

BroodMaker

Generated by Doxygen 1.9.3

1 Todo List	1
2 Module Index	3
2.1 Modules	3
3 Namespace Index	5
3.1 Namespace List	5
4 Hierarchical Index	7
4.1 Class Hierarchy	7
5 Class Index	9
5.1 Class List	9
6 File Index	13
6.1 File List	13
7 Module Documentation	17
7.1 Brood	17
7.1.1 Detailed Description	17
7.2 Application	17
7.2.1 Detailed Description	18
7.3 Data	18
7.3.1 Detailed Description	19
7.4 Components	19
7.4.1 Detailed Description	19
7.5 BroodUI	19
7.5.1 Detailed Description	19
8 Namespace Documentation	21
8.1 Brood Namespace Reference	21
8.2 Brood::Application Namespace Reference	21
8.2.1 Detailed Description	22
8.3 Brood::Application::Components Namespace Reference	22
8.3.1 Enumeration Type Documentation	22
8.3.1.1 ENUM_MovementType	22
8.3.1.2 ENUM_TileType	23
8.4 Brood::Application::Data Namespace Reference	23
8.5 Brood::Application::StaticVariables Namespace Reference	23
8.6 Brood::BroodUI Namespace Reference	24
8.6.1 Enumeration Type Documentation	24
8.6.1.1 ENUM_UIType	24
8.7 Brood::UtilityFuncs Namespace Reference	25
8.7.1 Detailed Description	25
8.7.2 Function Documentation	25

8.7.2.1 LoadTextureFromFile() [1/2]	25
8.7.2.2 LoadTextureFromFile() [2/2]	26
9 Class Documentation	27
9.1 Brood::Application::Application Class Reference	27
9.1.1 Detailed Description	28
9.1.2 Constructor & Destructor Documentation	28
9.1.2.1 Application()	28
9.1.2.2 ~Application()	28
9.1.3 Member Function Documentation	28
9.1.3.1 CreateWorkSpace()	28
9.1.3.2 Debugger()	29
9.1.3.3 Draw()	29
9.1.3.4 PollEvents()	29
9.1.3.5 RunApplicaiton()	30
9.1.4 Member Data Documentation	30
9.1.4.1 m_events	30
9.1.4.2 m_mainWorkspace	30
9.1.4.3 m_window	30
9.1.4.4 myBoard	30
9.1.4.5 myDice	31
9.2 Board Class Reference	31
9.2.1 Detailed Description	31
9.3 Brood::Application::Components::Board Class Reference	31
9.3.1 Constructor & Destructor Documentation	33
9.3.1.1 Board()	33
9.3.1.2 ~Board()	33
9.3.2 Member Function Documentation	34
9.3.2.1 Debugger()	34
9.3.2.2 DecreaseNumCol()	34
9.3.2.3 DecreaseNumRow()	34
9.3.2.4 Draw()	34
9.3.2.5 GetBoardPathList()	36
9.3.2.6 GetBoardPos()	36
9.3.2.7 GetBoardSize()	36
9.3.2.8 GetCurrentActivePath()	37
9.3.2.9 GetDataToSave()	37
9.3.2.10 GetNumCol()	37
9.3.2.11 GetNumRow()	37
9.3.2.12 IncreaseNumCol()	37
9.3.2.13 IncreaseNumRow()	38
9.3.2.14 InitializeBoard() [1/2]	38

9.3.2.15 InitializeBoard() [2/2]	38
9.3.2.16 LoadDataFromFile()	39
9.3.2.17 SaveDataToFile()	39
9.3.2.18 SetBoardPos() [1/2]	39
9.3.2.19 SetBoardPos() [2/2]	40
9.3.2.20 SetBoardSize() [1/2]	40
9.3.2.21 SetBoardSize() [2/2]	40
9.3.2.22 SetCurrentActiveTilePtr()	41
9.3.2.23 SetNumCol()	41
9.3.2.24 SetNumRow()	41
9.3.2.25 ToggleDrawLine()	42
9.3.2.26 UpdateBoardPath()	42
9.3.3 Member Data Documentation	42
9.3.3.1 m_boardBody	42
9.3.3.2 m_boardPaths	42
9.3.3.3 m_currActivePathPtr	43
9.3.3.4 m_numCols	43
9.3.3.5 m_numRows	43
9.4 BoardEditor Class Reference	43
9.4.1 Detailed Description	43
9.5 Brood::Application::BoardEditor Class Reference	44
9.5.1 Constructor & Destructor Documentation	46
9.5.1.1 BoardEditor()	46
9.5.1.2 ~BoardEditor()	46
9.5.2 Member Function Documentation	46
9.5.2.1 Debugger()	46
9.5.2.2 Draw()	46
9.5.2.3 InitializeWorkSpace()	47
9.5.2.4 Update()	47
9.5.2.5 UpdateAllDisplayElement()	47
9.5.2.6 UpdateBoardColPanelElement()	48
9.5.2.7 UpdateBoardRowPanelElement()	48
9.5.2.8 UpdateBoardXPosPanelElement()	48
9.5.2.9 UpdateBoardXSizePanelElement()	48
9.5.2.10 UpdateBoardYPosPanelElement()	49
9.5.2.11 UpdateBoardYSizePanelElement()	49
9.5.3 Member Data Documentation	49
9.5.3.1 m_btnBoardDecCol	49
9.5.3.2 m_btnBoardDecRow	49
9.5.3.3 m_btnBoardIncCol	50
9.5.3.4 m_btnBoardIncRow	50
9.5.3.5 m_btnBoardPosDecX	50

9.5.3.6 m_btnBoardPosDecY	50
9.5.3.7 m_btnBoardPosIncX	50
9.5.3.8 m_btnBoardPosIncY	50
9.5.3.9 m_btnBoardSizeDecX	51
9.5.3.10 m_btnBoardSizeDecY	51
9.5.3.11 m_btnBoardSizeIncX	51
9.5.3.12 m_btnBoardSizeIncY	51
9.5.3.13 m_panelBodyPtr	51
9.5.3.14 m_txtBoardCol	51
9.5.3.15 m_txtBoardColPromt	52
9.5.3.16 m_txtBoardPosPromtX	52
9.5.3.17 m_txtBoardPosPromtY	52
9.5.3.18 m_txtBoardPosX	52
9.5.3.19 m_txtBoardPosY	52
9.5.3.20 m_txtBoardRow	52
9.5.3.21 m_txtBoardRowPromt	53
9.5.3.22 m_txtBoardSizePromtX	53
9.5.3.23 m_txtBoardSizePromtY	53
9.5.3.24 m_txtBoardSizeX	53
9.5.3.25 m_txtBoardSizeY	53
9.5.3.26 m_txtSettingTitle	53
9.6 Brood::BroodUI::Button Class Reference	54
9.6.1 Constructor & Destructor Documentation	55
9.6.1.1 Button() [1/2]	55
9.6.1.2 ~Button()	55
9.6.1.3 Button() [2/2]	55
9.6.2 Member Function Documentation	56
9.6.2.1 Debugger()	56
9.6.2.2 DoElement()	56
9.6.2.3 DyCreateButton() [1/2]	56
9.6.2.4 DyCreateButton() [2/2]	57
9.6.2.5 GetSpriteBody()	57
9.6.2.6 operator=()	58
9.6.2.7 SetSelected()	58
9.6.3 Member Data Documentation	58
9.6.3.1 m_bodySprite	59
9.7 Button Class Reference	59
9.7.1 Detailed Description	59
9.8 Brood::Application::CardEditor Class Reference	60
9.8.1 Constructor & Destructor Documentation	72
9.8.1.1 CardEditor()	72
9.8.1.2 ~CardEditor()	72

9.8.2 Member Function Documentation	73
9.8.2.1 Debugger()	73
9.8.2.2 Draw()	73
9.8.2.3 DrawCardDisplayBcompPanel()	73
9.8.2.4 DrawCardDisplayFcompPanel()	74
9.8.2.5 DrawCardDisplayPanel()	74
9.8.2.6 DrawCardInfoPanel()	74
9.8.2.7 InitializeCardDisplayBcompPanel()	76
9.8.2.8 InitializeCardDisplayFcompPanel()	76
9.8.2.9 InitializeCardDisplayPanel()	76
9.8.2.10 InitializeCardInfoPanel()	76
9.8.2.11 InitializeSettingSelectionDDI()	77
9.8.2.12 InitializeWorkSpace()	77
9.8.2.13 Update()	77
9.8.2.14 UpdateAllDisplayElement()	77
9.8.2.15 UpdateCardBackFileTexture()	78
9.8.2.16 UpdateCardDisplayBcompPanel()	78
9.8.2.17 UpdateCardDisplayFcompPanel()	78
9.8.2.18 UpdateCardDisplayPanel()	79
9.8.2.19 UpdateCardFrontFileTexture()	79
9.8.2.20 UpdateCardInfoPanel()	79
9.8.2.21 UpdateCurrCardAnswerInput()	79
9.8.2.22 UpdateCurrCardInfoDown()	80
9.8.2.23 UpdateCurrCardInfoTime()	80
9.8.2.24 UpdateCurrCardInfoUp()	80
9.8.2.25 UpdateCurrCardQuestionInput()	80
9.8.2.26 UpdateCurrSelectedCardIdx()	81
9.8.2.27 UpdateCurrSelectedDeckIdx()	81
9.8.2.28 UpdateSettingSelectionDDI()	81
9.8.2.29 UpdateTurnCard()	81
9.8.3 Member Data Documentation	82
9.8.3.1 m_btnCardAnswerPromptDecPosX	82
9.8.3.2 m_btnCardAnswerPromptDecPosY	82
9.8.3.3 m_btnCardAnswerPromptDecSizeX	82
9.8.3.4 m_btnCardAnswerPromptDecSizeY	82
9.8.3.5 m_btnCardAnswerPromptIncPosX	82
9.8.3.6 m_btnCardAnswerPromptIncPosY	83
9.8.3.7 m_btnCardAnswerPromptIncSizeX	83
9.8.3.8 m_btnCardAnswerPromptIncSizeY	83
9.8.3.9 m_btnCardAnswerValueDecPosX	83
9.8.3.10 m_btnCardAnswerValueDecPosY	83
9.8.3.11 m_btnCardAnswerValueDecSizeX	83

9.8.3.12 m_btnCardAnswerValueDecSizeY	84
9.8.3.13 m_btnCardAnswerValueIncPosX	84
9.8.3.14 m_btnCardAnswerValueIncPosY	84
9.8.3.15 m_btnCardAnswerValueIncSizeX	84
9.8.3.16 m_btnCardAnswerValueIncSizeY	84
9.8.3.17 m_btnCardBackBgOpenFile	84
9.8.3.18 m_btnCardBackTimePromptDecPosX	85
9.8.3.19 m_btnCardBackTimePromptDecPosY	85
9.8.3.20 m_btnCardBackTimePromptDecSizeX	85
9.8.3.21 m_btnCardBackTimePromptDecSizeY	85
9.8.3.22 m_btnCardBackTimePromptIncPosX	85
9.8.3.23 m_btnCardBackTimePromptIncPosY	85
9.8.3.24 m_btnCardBackTimePromptIncSizeX	86
9.8.3.25 m_btnCardBackTimePromptIncSizeY	86
9.8.3.26 m_btnCardBackTimeValueDecPosX	86
9.8.3.27 m_btnCardBackTimeValueDecPosY	86
9.8.3.28 m_btnCardBackTimeValueDecSizeX	86
9.8.3.29 m_btnCardBackTimeValueDecSizeY	86
9.8.3.30 m_btnCardBackTimeValueIncPosX	87
9.8.3.31 m_btnCardBackTimeValueIncPosY	87
9.8.3.32 m_btnCardBackTimeValueIncSizeX	87
9.8.3.33 m_btnCardBackTimeValueIncSizeY	87
9.8.3.34 m_btnCardDecPosX	87
9.8.3.35 m_btnCardDecPosY	87
9.8.3.36 m_btnCardDecSizeX	88
9.8.3.37 m_btnCardDecSizeY	88
9.8.3.38 m_btnCardDownPromptDecPosX	88
9.8.3.39 m_btnCardDownPromptDecPosY	88
9.8.3.40 m_btnCardDownPromptDecSizeX	88
9.8.3.41 m_btnCardDownPromptDecSizeY	88
9.8.3.42 m_btnCardDownPromptIncPosX	89
9.8.3.43 m_btnCardDownPromptIncPosY	89
9.8.3.44 m_btnCardDownPromptIncSizeX	89
9.8.3.45 m_btnCardDownPromptIncSizeY	89
9.8.3.46 m_btnCardDownValueDecPosX	89
9.8.3.47 m_btnCardDownValueDecPosY	89
9.8.3.48 m_btnCardDownValueDecSizeX	90
9.8.3.49 m_btnCardDownValueDecSizeY	90
9.8.3.50 m_btnCardDownValueIncPosX	90
9.8.3.51 m_btnCardDownValueIncPosY	90
9.8.3.52 m_btnCardDownValueIncSizeX	90
9.8.3.53 m_btnCardDownValueIncSizeY	90

9.8.3.54 m_btnCardFrontBgOpenFile	91
9.8.3.55 m_btnCardFrontTimePromptDecPosX	91
9.8.3.56 m_btnCardFrontTimePromptDecPosY	91
9.8.3.57 m_btnCardFrontTimePromptDecSizeX	91
9.8.3.58 m_btnCardFrontTimePromptDecSizeY	91
9.8.3.59 m_btnCardFrontTimePromptIncPosX	91
9.8.3.60 m_btnCardFrontTimePromptIncPosY	92
9.8.3.61 m_btnCardFrontTimePromptIncSizeX	92
9.8.3.62 m_btnCardFrontTimePromptIncSizeY	92
9.8.3.63 m_btnCardFrontTimeValueDecPosX	92
9.8.3.64 m_btnCardFrontTimeValueDecPosY	92
9.8.3.65 m_btnCardFrontTimeValueDecSizeX	92
9.8.3.66 m_btnCardFrontTimeValueDecSizeY	93
9.8.3.67 m_btnCardFrontTimeValueIncPosX	93
9.8.3.68 m_btnCardFrontTimeValueIncPosY	93
9.8.3.69 m_btnCardFrontTimeValueIncSizeX	93
9.8.3.70 m_btnCardFrontTimeValueIncSizeY	93
9.8.3.71 m_btnCardIncPosX	93
9.8.3.72 m_btnCardIncPosY	94
9.8.3.73 m_btnCardIncSizeX	94
9.8.3.74 m_btnCardIncSizeY	94
9.8.3.75 m_btnCardQuestionPromptDecPosX	94
9.8.3.76 m_btnCardQuestionPromptDecPosY	94
9.8.3.77 m_btnCardQuestionPromptDecSizeX	94
9.8.3.78 m_btnCardQuestionPromptDecSizeY	95
9.8.3.79 m_btnCardQuestionPromptIncPosX	95
9.8.3.80 m_btnCardQuestionPromptIncPosY	95
9.8.3.81 m_btnCardQuestionPromptIncSizeX	95
9.8.3.82 m_btnCardQuestionPromptIncSizeY	95
9.8.3.83 m_btnCardQuestionValueDecPosX	95
9.8.3.84 m_btnCardQuestionValueDecPosY	96
9.8.3.85 m_btnCardQuestionValueDecSizeX	96
9.8.3.86 m_btnCardQuestionValueDecSizeY	96
9.8.3.87 m_btnCardQuestionValueIncPosX	96
9.8.3.88 m_btnCardQuestionValueIncPosY	96
9.8.3.89 m_btnCardQuestionValueIncSizeX	96
9.8.3.90 m_btnCardQuestionValueIncSizeY	97
9.8.3.91 m_btnCardSubmitDecPosX	97
9.8.3.92 m_btnCardSubmitDecPosY	97
9.8.3.93 m_btnCardSubmitDecSizeX	97
9.8.3.94 m_btnCardSubmitDecSizeY	97
9.8.3.95 m_btnCardSubmitIncPosX	97

9.8.3.96 m_btnCardSubmitIncPosY	98
9.8.3.97 m_btnCardSubmitIncSizeX	98
9.8.3.98 m_btnCardSubmitIncSizeY	98
9.8.3.99 m_btnCardTurnCardDecPosX	98
9.8.3.100 m_btnCardTurnCardDecPosY	98
9.8.3.101 m_btnCardTurnCardDecSizeX	98
9.8.3.102 m_btnCardTurnCardDecSizeY	99
9.8.3.103 m_btnCardTurnCardIncPosX	99
9.8.3.104 m_btnCardTurnCardIncPosY	99
9.8.3.105 m_btnCardTurnCardIncSizeX	99
9.8.3.106 m_btnCardTurnCardIncSizeY	99
9.8.3.107 m_btnCardUpPromptDecPosX	99
9.8.3.108 m_btnCardUpPromptDecPosY	100
9.8.3.109 m_btnCardUpPromptDecSizeX	100
9.8.3.110 m_btnCardUpPromptDecSizeY	100
9.8.3.111 m_btnCardUpPromptIncPosX	100
9.8.3.112 m_btnCardUpPromptIncPosY	100
9.8.3.113 m_btnCardUpPromptIncSizeX	100
9.8.3.114 m_btnCardUpPromptIncSizeY	101
9.8.3.115 m_btnCardUpValueDecPosX	101
9.8.3.116 m_btnCardUpValueDecPosY	101
9.8.3.117 m_btnCardUpValueDecSizeX	101
9.8.3.118 m_btnCardUpValueDecSizeY	101
9.8.3.119 m_btnCardUpValueIncPosX	101
9.8.3.120 m_btnCardUpValueIncPosY	102
9.8.3.121 m_btnCardUpValueIncSizeX	102
9.8.3.122 m_btnCardUpValueIncSizeY	102
9.8.3.123 m_btnDecCurrCardDown	102
9.8.3.124 m_btnDecCurrCardNum	102
9.8.3.125 m_btnDecCurrCardTime	102
9.8.3.126 m_btnDecCurrCardUp	103
9.8.3.127 m_btnDecDeckNum	103
9.8.3.128 m_btnIncCurrCardDown	103
9.8.3.129 m_btnIncCurrCardNum	103
9.8.3.130 m_btnIncCurrCardTime	103
9.8.3.131 m_btnIncCurrCardUp	103
9.8.3.132 m_btnIncDeckNum	104
9.8.3.133 m_btnTurnCardAround	104
9.8.3.134 m_ddiSettingSelection	104
9.8.3.135 m_panelBodyPtr	104
9.8.3.136 m_selectedSettingIdx	104
9.8.3.137 m_txtCardAnswerPromptPosX	104

9.8.3.138 m_txtCardAnswerPromptPosXPrompt	104
9.8.3.139 m_txtCardAnswerPromptPosY	105
9.8.3.140 m_txtCardAnswerPromptPosYPrompt	105
9.8.3.141 m_txtCardAnswerPromptSizeX	105
9.8.3.142 m_txtCardAnswerPromptSizeXPrompt	105
9.8.3.143 m_txtCardAnswerPromptSizeY	105
9.8.3.144 m_txtCardAnswerPromptSizeYPrompt	105
9.8.3.145 m_txtCardAnswerValuePosX	106
9.8.3.146 m_txtCardAnswerValuePosXPrompt	106
9.8.3.147 m_txtCardAnswerValuePosY	106
9.8.3.148 m_txtCardAnswerValuePosYPrompt	106
9.8.3.149 m_txtCardAnswerValueSizeX	106
9.8.3.150 m_txtCardAnswerValueSizeXPrompt	106
9.8.3.151 m_txtCardAnswerValueSizeY	107
9.8.3.152 m_txtCardAnswerValueSizeYPrompt	107
9.8.3.153 m_txtCardBackBgFileInput	107
9.8.3.154 m_txtCardBackBgFileNamePrompt	107
9.8.3.155 m_txtCardBackTimePromptPosX	107
9.8.3.156 m_txtCardBackTimePromptPosXPrompt	107
9.8.3.157 m_txtCardBackTimePromptPosY	108
9.8.3.158 m_txtCardBackTimePromptPosYPrompt	108
9.8.3.159 m_txtCardBackTimePromptSizeX	108
9.8.3.160 m_txtCardBackTimePromptSizeXPrompt	108
9.8.3.161 m_txtCardBackTimePromptSizeY	108
9.8.3.162 m_txtCardBackTimePromptSizeYPrompt	108
9.8.3.163 m_txtCardBackTimeValuePosX	109
9.8.3.164 m_txtCardBackTimeValuePosXPrompt	109
9.8.3.165 m_txtCardBackTimeValuePosY	109
9.8.3.166 m_txtCardBackTimeValuePosYPrompt	109
9.8.3.167 m_txtCardBackTimeValueSizeX	109
9.8.3.168 m_txtCardBackTimeValueSizeXPrompt	109
9.8.3.169 m_txtCardBackTimeValueSizeY	110
9.8.3.170 m_txtCardBackTimeValueSizeYPrompt	110
9.8.3.171 m_txtCardDownPromptPosX	110
9.8.3.172 m_txtCardDownPromptPosXPrompt	110
9.8.3.173 m_txtCardDownPromptPosY	110
9.8.3.174 m_txtCardDownPromptPosYPrompt	110
9.8.3.175 m_txtCardDownPromptSizeX	111
9.8.3.176 m_txtCardDownPromptSizeXPrompt	111
9.8.3.177 m_txtCardDownPromptSizeY	111
9.8.3.178 m_txtCardDownPromptSizeYPrompt	111
9.8.3.179 m_txtCardDownValuePosX	111

9.8.3.180 m_txtCardDownValuePosXPrompt	111
9.8.3.181 m_txtCardDownValuePosY	112
9.8.3.182 m_txtCardDownValuePosYPrompt	112
9.8.3.183 m_txtCardDownValueSizeX	112
9.8.3.184 m_txtCardDownValueSizeXPrompt	112
9.8.3.185 m_txtCardDownValueSizeY	112
9.8.3.186 m_txtCardDownValueSizeYPrompt	112
9.8.3.187 m_txtCardFrontBgFileInput	113
9.8.3.188 m_txtCardFrontBgFileNamePrompt	113
9.8.3.189 m_txtCardFrontTimePromptPosX	113
9.8.3.190 m_txtCardFrontTimePromptPosXPrompt	113
9.8.3.191 m_txtCardFrontTimePromptPosY	113
9.8.3.192 m_txtCardFrontTimePromptPosYPrompt	113
9.8.3.193 m_txtCardFrontTimePromptSizeX	114
9.8.3.194 m_txtCardFrontTimePromptSizeXPrompt	114
9.8.3.195 m_txtCardFrontTimePromptSizeY	114
9.8.3.196 m_txtCardFrontTimePromptSizeYPrompt	114
9.8.3.197 m_txtCardFrontTimeValuePosX	114
9.8.3.198 m_txtCardFrontTimeValuePosXPrompt	114
9.8.3.199 m_txtCardFrontTimeValuePosY	115
9.8.3.200 m_txtCardFrontTimeValuePosYPrompt	115
9.8.3.201 m_txtCardFrontTimeValueSizeX	115
9.8.3.202 m_txtCardFrontTimeValueSizeXPrompt	115
9.8.3.203 m_txtCardFrontTimeValueSizeY	115
9.8.3.204 m_txtCardFrontTimeValueSizeYPrompt	115
9.8.3.205 m_txtCardPosX	116
9.8.3.206 m_txtCardPosXPrompt	116
9.8.3.207 m_txtCardPosY	116
9.8.3.208 m_txtCardPosYPrompt	116
9.8.3.209 m_txtCardQuestionPromptPosX	116
9.8.3.210 m_txtCardQuestionPromptPosXPrompt	116
9.8.3.211 m_txtCardQuestionPromptPosY	117
9.8.3.212 m_txtCardQuestionPromptPosYPrompt	117
9.8.3.213 m_txtCardQuestionPromptSizeX	117
9.8.3.214 m_txtCardQuestionPromptSizeXPrompt	117
9.8.3.215 m_txtCardQuestionPromptSizeY	117
9.8.3.216 m_txtCardQuestionPromptSizeYPrompt	117
9.8.3.217 m_txtCardQuestionValuePosX	118
9.8.3.218 m_txtCardQuestionValuePosXPrompt	118
9.8.3.219 m_txtCardQuestionValuePosY	118
9.8.3.220 m_txtCardQuestionValuePosYPrompt	118
9.8.3.221 m_txtCardQuestionValueSizeX	118

9.8.3.222 m_txtCardQuestionValueSizeXPrompt	118
9.8.3.223 m_txtCardQuestionValueSizeY	119
9.8.3.224 m_txtCardQuestionValueSizeYPrompt	119
9.8.3.225 m_txtCardSizeX	119
9.8.3.226 m_txtCardSizeXPrompt	119
9.8.3.227 m_txtCardSizeY	119
9.8.3.228 m_txtCardSizeYPrompt	119
9.8.3.229 m_txtCardSubmitPosX	120
9.8.3.230 m_txtCardSubmitPosXPrompt	120
9.8.3.231 m_txtCardSubmitPosY	120
9.8.3.232 m_txtCardSubmitPosYPrompt	120
9.8.3.233 m_txtCardSubmitSizeX	120
9.8.3.234 m_txtCardSubmitSizeXPrompt	120
9.8.3.235 m_txtCardSubmitSizeY	121
9.8.3.236 m_txtCardSubmitSizeYPrompt	121
9.8.3.237 m_txtCardTurnCardPosX	121
9.8.3.238 m_txtCardTurnCardPosXPrompt	121
9.8.3.239 m_txtCardTurnCardPosY	121
9.8.3.240 m_txtCardTurnCardPosYPrompt	121
9.8.3.241 m_txtCardTurnCardSizeX	122
9.8.3.242 m_txtCardTurnCardSizeXPrompt	122
9.8.3.243 m_txtCardTurnCardSizeY	122
9.8.3.244 m_txtCardTurnCardSizeYPrompt	122
9.8.3.245 m_txtCardUpPromptPosX	122
9.8.3.246 m_txtCardUpPromptPosXPrompt	122
9.8.3.247 m_txtCardUpPromptPosY	123
9.8.3.248 m_txtCardUpPromptPosYPrompt	123
9.8.3.249 m_txtCardUpPromptSizeX	123
9.8.3.250 m_txtCardUpPromptSizeXPrompt	123
9.8.3.251 m_txtCardUpPromptSizeY	123
9.8.3.252 m_txtCardUpPromptSizeYPrompt	123
9.8.3.253 m_txtCardUpValuePosX	124
9.8.3.254 m_txtCardUpValuePosXPrompt	124
9.8.3.255 m_txtCardUpValuePosY	124
9.8.3.256 m_txtCardUpValuePosYPrompt	124
9.8.3.257 m_txtCardUpValueSizeX	124
9.8.3.258 m_txtCardUpValueSizeXPrompt	124
9.8.3.259 m_txtCardUpValueSizeY	125
9.8.3.260 m_txtCardUpValueSizeYPrompt	125
9.8.3.261 m_txtCurrCardAnswerInput	125
9.8.3.262 m_txtCurrCardAnswerPrompt	125
9.8.3.263 m_txtCurrCardDown	125

9.8.3.264 m_txtCurrCardDownPrompt	125
9.8.3.265 m_txtCurrCardNum	126
9.8.3.266 m_txtCurrCardNumPrompt	126
9.8.3.267 m_txtCurrCardQuestionInput	126
9.8.3.268 m_txtCurrCardQuestionPrompt	126
9.8.3.269 m_txtCurrCardTime	126
9.8.3.270 m_txtCurrCardTimePrompt	126
9.8.3.271 m_txtCurrCardUp	127
9.8.3.272 m_txtCurrCardUpPrompt	127
9.8.3.273 m_txtDeckNum	127
9.8.3.274 m_txtDeckNumPrompt	127
9.9 CardEditor Class Reference	127
9.9.1 Detailed Description	127
9.10 Brood::Application::Components::CardInfo Class Reference	128
9.10.1 Constructor & Destructor Documentation	129
9.10.1.1 CardInfo() [1/2]	129
9.10.1.2 ~CardInfo()	129
9.10.1.3 CardInfo() [2/2]	129
9.10.2 Member Function Documentation	129
9.10.2.1 GetCorrectAnswer()	129
9.10.2.2 GetCorrectNumSteps()	130
9.10.2.3 GetDataToSave()	130
9.10.2.4 GetIncorrectNumSteps()	130
9.10.2.5 GetQuestion()	130
9.10.2.6 GetTime()	131
9.10.2.7 InitializeCard()	131
9.10.2.8 operator=()	131
9.10.2.9 SetCorrectAnswer()	131
9.10.2.10 SetCorrectNumSteps()	132
9.10.2.11 SetIncorrectNumSteps()	132
9.10.2.12 SetQuestion()	132
9.10.2.13 SetTime()	132
9.10.3 Member Data Documentation	133
9.10.3.1 m_correctAnswer	133
9.10.3.2 m_correctNumSteps	133
9.10.3.3 m_incorrectNumSteps	133
9.10.3.4 m_question	133
9.10.3.5 m_time	133
9.11 Brood::Application::Components::Deck Class Reference	134
9.11.1 Constructor & Destructor Documentation	135
9.11.1.1 Deck() [1/2]	135
9.11.1.2 ~Deck()	135

9.11.1.3 Deck() [2/2]	136
9.11.2 Member Function Documentation	136
9.11.2.1 AddCardInfoToDeck()	136
9.11.2.2 DealCard()	136
9.11.2.3 GetCardInitFileName()	136
9.11.2.4 GetCardInitFilePath()	137
9.11.2.5 GetCardList()	137
9.11.2.6 GetCardPtrAtIdx()	137
9.11.2.7 GetCurrActiveCardIdx()	137
9.11.2.8 GetCurrActiveCardPtr()	138
9.11.2.9 GetDataToSave()	138
9.11.2.10 InitializeDeck()	138
9.11.2.11 LoadCardFromInitFile()	138
9.11.2.12 LoadDataFromFile()	139
9.11.2.13 operator=()	139
9.11.2.14 ResetDeck()	139
9.11.2.15 SaveCardToInitFile()	139
9.11.2.16 SaveDataToFile()	140
9.11.2.17 SetCardListSize()	140
9.11.2.18 SetCurrActiveCardIdx()	140
9.11.2.19 Shuffel()	141
9.11.3 Member Data Documentation	141
9.11.3.1 m_cardInitFileName	141
9.11.3.2 m_cardInitFilePath	141
9.11.3.3 m_cardList	141
9.11.3.4 m_currActiveCardIdx	141
9.11.3.5 m_currUndealtCardIdx	141
9.12 Brood::Application::DeckEditor Class Reference	142
9.12.1 Constructor & Destructor Documentation	144
9.12.1.1 DeckEditor()	144
9.12.1.2 ~DeckEditor()	144
9.12.2 Member Function Documentation	144
9.12.2.1 Debugger()	145
9.12.2.2 Draw()	145
9.12.2.3 InitializeWorkSpace()	145
9.12.2.4 Update()	146
9.12.2.5 UpdateAllDisplayElement()	146
9.12.2.6 UpdateCardInitFile()	146
9.12.2.7 UpdateCardNumber()	147
9.12.2.8 UpdateCurrSelectedDeckIdx()	147
9.12.2.9 UpdateDeckFileTexture()	147
9.12.2.10 UpdateDeckPosX()	148

9.12.2.11 UpdateDeckPosY()	148
9.12.2.12 UpdateDeckSizeX()	148
9.12.2.13 UpdateDeckSizeY()	148
9.12.3 Member Data Documentation	149
9.12.3.1 m_btnCardInitOpenFile	149
9.12.3.2 m_btnDecCardNum	149
9.12.3.3 m_btnDeckBgOpenFile	149
9.12.3.4 m_btnDeckDecCurrlIdx	149
9.12.3.5 m_btnDeckDecPosX	149
9.12.3.6 m_btnDeckDecPosY	150
9.12.3.7 m_btnDeckDecSizeX	150
9.12.3.8 m_btnDeckDecSizeY	150
9.12.3.9 m_btnDeckIncCurrlIdx	150
9.12.3.10 m_btnDeckIncPosX	150
9.12.3.11 m_btnDeckIncPosY	150
9.12.3.12 m_btnDeckIncSizeX	151
9.12.3.13 m_btnDeckIncSizeY	151
9.12.3.14 m_btnIncCardNum	151
9.12.3.15 m_panelBodyPtr	151
9.12.3.16 m_txtCardInitFileInput	151
9.12.3.17 m_txtCardInitFileNamePrompt	151
9.12.3.18 m_txtCardNum	152
9.12.3.19 m_txtCardNumPrompt	152
9.12.3.20 m_txtDeckBgFileInput	152
9.12.3.21 m_txtDeckBgFileNamePrompt	152
9.12.3.22 m_txtDeckCurrlIdx	152
9.12.3.23 m_txtDeckCurrlIdxPrompt	152
9.12.3.24 m_txtDeckPosX	153
9.12.3.25 m_txtDeckPosXPrompt	153
9.12.3.26 m_txtDeckPosY	153
9.12.3.27 m_txtDeckPosYPrompt	153
9.12.3.28 m_txtDeckSizeX	153
9.12.3.29 m_txtDeckSizeXPrompt	153
9.12.3.30 m_txtDeckSizeY	154
9.12.3.31 m_txtDeckSizeYPrompt	154
9.12.3.32 m_txtSettingTitle	154
9.13 Brood::Application::Components::DeckManager Class Reference	154
9.13.1 Constructor & Destructor Documentation	155
9.13.1.1 DeckManager() [1/2]	155
9.13.1.2 ~DeckManager()	156
9.13.1.3 DeckManager() [2/2]	156
9.13.2 Member Function Documentation	156

9.13.2.1 AddCardToDeckAtCurrlIdx()	156
9.13.2.2 Debugger()	156
9.13.2.3 Draw()	157
9.13.2.4 GetCurrActiveDeck()	157
9.13.2.5 GetCurrActiveDeckIdx()	157
9.13.2.6 GetDataToSave()	157
9.13.2.7 GetDeckAtIdx()	157
9.13.2.8 GetDeckList()	158
9.13.2.9 GetIncorrectPenalty()	158
9.13.2.10 GetMovementType()	158
9.13.2.11 InitializeDeckManager()	158
9.13.2.12 LoadDataFromFile()	159
9.13.2.13 operator=()	159
9.13.2.14 SaveDataToFile()	159
9.13.2.15 SetCurrActiveDeckIdx()	160
9.13.2.16 SetDeckSize()	160
9.13.2.17 SetIncorrectPenalty()	160
9.13.2.18 SetMovementType()	160
9.13.3 Member Data Documentation	161
9.13.3.1 m_currDeckIdx	161
9.13.3.2 m_deckList	161
9.13.3.3 m_incorrectPenalty	161
9.13.3.4 m_movementType	161
9.14 Brood::Application::Components::Dice Class Reference	161
9.14.1 Constructor & Destructor Documentation	162
9.14.1.1 Dice() [1/3]	162
9.14.1.2 Dice() [2/3]	163
9.14.1.3 ~Dice()	163
9.14.1.4 Dice() [3/3]	163
9.14.2 Member Function Documentation	164
9.14.2.1 GetDataToSave()	164
9.14.2.2 GetNumSides()	164
9.14.2.3 InitializeDice()	164
9.14.2.4 LoadDataFromFile()	165
9.14.2.5 operator=()	165
9.14.2.6 RollDice()	165
9.14.2.7 SaveDataToFile()	165
9.14.2.8 SetBodySize() [1/2]	166
9.14.2.9 SetBodySize() [2/2]	166
9.14.2.10 SetNumSides()	166
9.14.2.11 SetTexture()	167
9.14.3 Member Data Documentation	167

9.14.3.1 m_numSides	167
9.15 Dice Class Reference	167
9.15.1 Detailed Description	167
9.16 Brood::Application::DiceEditor Class Reference	168
9.16.1 Constructor & Destructor Documentation	170
9.16.1.1 DiceEditor()	170
9.16.1.2 ~DiceEditor()	170
9.16.2 Member Function Documentation	170
9.16.2.1 Debugger()	170
9.16.2.2 Draw()	171
9.16.2.3 InitializeWorkSpace()	171
9.16.2.4 Update()	171
9.16.2.5 UpdateAllDisplayElement()	172
9.16.2.6 UpdateDiceFileTexture()	172
9.16.2.7 UpdateDicePosX()	172
9.16.2.8 UpdateDicePosY()	173
9.16.2.9 UpdateDiceSideNum()	173
9.16.2.10 UpdateDiceSizeX()	173
9.16.2.11 UpdateDiceSizeY()	173
9.16.2.12 UpdateRollDice()	174
9.16.3 Member Data Documentation	174
9.16.3.1 m_boardPtr	174
9.16.3.2 m_btnDiceDecPosX	174
9.16.3.3 m_btnDiceDecPosY	174
9.16.3.4 m_btnDiceDecSide	175
9.16.3.5 m_btnDiceDecSizeX	175
9.16.3.6 m_btnDiceDecSizeY	175
9.16.3.7 m_btnDiceIncPosX	175
9.16.3.8 m_btnDiceIncPosY	175
9.16.3.9 m_btnDiceIncSide	175
9.16.3.10 m_btnDiceIncSizeX	176
9.16.3.11 m_btnDiceIncSizeY	176
9.16.3.12 m_btnDiceOpenFile	176
9.16.3.13 m_btnRollDice	176
9.16.3.14 m_dicePtr	176
9.16.3.15 m_panelBodyPtr	176
9.16.3.16 m_txtDiceFileInput	177
9.16.3.17 m_txtDiceFileNamePrompt	177
9.16.3.18 m_txtDicePosX	177
9.16.3.19 m_txtDicePosXPrompt	177
9.16.3.20 m_txtDicePosY	177
9.16.3.21 m_txtDicePosYPrompt	177

9.16.3.22 m_txtDiceSide	178
9.16.3.23 m_txtDiceSidePrompt	178
9.16.3.24 m_txtDiceSizeX	178
9.16.3.25 m_txtDiceSizeXPrompt	178
9.16.3.26 m_txtDiceSizeY	178
9.16.3.27 m_txtDiceSizeYPrompt	178
9.16.3.28 m_txtSettingTitle	179
9.17 DiceEditor Class Reference	179
9.17.1 Detailed Description	179
9.18 Brood::Application::Components::DisplayCard Class Reference	179
9.18.1 Constructor & Destructor Documentation	182
9.18.1.1 DisplayCard() [1/2]	182
9.18.1.2 ~DisplayCard()	182
9.18.1.3 DisplayCard() [2/2]	182
9.18.2 Member Function Documentation	182
9.18.2.1 Debugger()	182
9.18.2.2 Draw()	183
9.18.2.3 GetAnswerPromptPtr()	183
9.18.2.4 GetAnswerValuePtr()	183
9.18.2.5 GetBackBgFileName()	183
9.18.2.6 GetBackTimePromptPtr()	184
9.18.2.7 GetBackTimeValuePtr()	184
9.18.2.8 GetCompUpperPos()	184
9.18.2.9 GetDataToSave()	184
9.18.2.10 GetDownPromptPtr()	185
9.18.2.11 GetDownValuePtr()	185
9.18.2.12 GetFrontBgFileName()	185
9.18.2.13 GetFrontTimePromptPtr()	185
9.18.2.14 GetFrontTimeValuePtr()	186
9.18.2.15 GetFurtherstBottomRightCordinate()	186
9.18.2.16 GetNearestTopLeftCordinate()	186
9.18.2.17 GetQuestionPromptPtr()	186
9.18.2.18 GetQuestionValuePtr()	187
9.18.2.19 GetSubmitPtr()	187
9.18.2.20 GetTurnCardPtr()	187
9.18.2.21 GetUpPromptPtr()	187
9.18.2.22 GetUpValuePtr()	188
9.18.2.23 InializeDisplayCard()	188
9.18.2.24 InitializeDisplayCard()	188
9.18.2.25 LoadDataFromFile()	188
9.18.2.26 operator=()	189
9.18.2.27 SaveDataToFile()	189

9.18.2.28 SetBackBgFileName()	189
9.18.2.29 SetCardFront()	190
9.18.2.30 SetCardInfoToDisplay()	190
9.18.2.31 SetFrontBgFileName()	190
9.18.2.32 ToggleCardFace()	190
9.18.2.33 UpdateDisplayedText()	191
9.18.2.34 UpdateLargestUpperPos()	191
9.18.2.35 UpdateSmallestLowerPos()	191
9.18.3 Member Data Documentation	191
9.18.3.1 m_backBgFileName	191
9.18.3.2 m_BtnSubmit	192
9.18.3.3 m_BtnTurnCard	192
9.18.3.4 m_cardInfoToDisplay	192
9.18.3.5 m_frontBgFileName	192
9.18.3.6 m_isCurrFront	192
9.18.3.7 m_TxtBackTimePromt	192
9.18.3.8 m_TxtBackTimeValue	192
9.18.3.9 m_TxtDownPrompt	193
9.18.3.10 m_TxtDownValue	193
9.18.3.11 m_TxtFrontTimePromt	193
9.18.3.12 m_TxtFrontTimeValue	193
9.18.3.13 m_TxtQuestionPrompt	193
9.18.3.14 m_TxtQuestionValue	193
9.18.3.15 m_TxtUpPromt	193
9.18.3.16 m_TxtUpValue	194
9.18.3.17 m_TxtUserAnswerPrompt	194
9.18.3.18 m_TxtUserAnswerValue	194
9.19 Brood::BroodUI::DropDownInput Class Reference	194
9.19.1 Constructor & Destructor Documentation	195
9.19.1.1 DropDownInput() [1/2]	195
9.19.1.2 ~DropDownInput()	195
9.19.1.3 DropDownInput() [2/2]	195
9.19.2 Member Function Documentation	195
9.19.2.1 AddItemToMenu() [1/2]	196
9.19.2.2 AddItemToMenu() [2/2]	196
9.19.2.3 operator=()	196
9.20 DropDownInput Class Reference	197
9.20.1 Detailed Description	197
9.21 Brood::BroodUI::DropDownMenu Class Reference	198
9.21.1 Constructor & Destructor Documentation	200
9.21.1.1 DropDownMenu() [1/2]	200
9.21.1.2 ~DropDownMenu()	200

9.21.1.3 DropDownMenu() [2/2]	200
9.21.2 Member Function Documentation	201
9.21.2.1 AddItemToMenu() [1/2]	201
9.21.2.2 AddItemToMenu() [2/2]	201
9.21.2.3 Debugger()	201
9.21.2.4 DoElement()	202
9.21.2.5 Draw()	202
9.21.2.6 GetItemList()	202
9.21.2.7 operator=()	203
9.21.2.8 SetActiveOverlayColor()	203
9.21.2.9 SetBodyColor()	203
9.21.2.10 SetBodyPosition() [1/2]	204
9.21.2.11 SetBodyPosition() [2/2]	204
9.21.2.12 SetBodySize() [1/2]	204
9.21.2.13 SetBodySize() [2/2]	205
9.21.2.14 SetEachItemPos()	205
9.21.2.15 SetEachItemSize()	205
9.21.2.16SetFont()	205
9.21.2.17 SetFontColor()	206
9.21.2.18 SetFontSize()	206
9.21.2.19 SetHotOverlayColor()	206
9.21.2.20 SetItemPos()	207
9.21.2.21 SetItemSize()	207
9.21.2.22 SetText()	207
9.21.3 Member Data Documentation	207
9.21.3.1 m_items	207
9.21.3.2 m_maxItemLength	208
9.22 DropDownMenu Class Reference	208
9.22.1 Detailed Description	208
9.23 Brood::Application::EditorWorkspace Class Reference	209
9.23.1 Constructor & Destructor Documentation	210
9.23.1.1 EditorWorkspace()	210
9.23.1.2 ~EditorWorkspace()	210
9.23.2 Member Function Documentation	210
9.23.2.1 Debugger()	210
9.23.2.2 Draw()	211
9.23.2.3 InitializeEditModeTabs()	211
9.23.2.4 InitializeWorkSpace()	211
9.23.2.5 SetGameDataManager()	212
9.23.2.6 Update()	212
9.23.2.7 UpdateActiveEditorIdx()	212
9.23.2.8 UpdateAllDisplayElement()	212

9.23.2.9 UpdateEditModeTabs()	213
9.23.3 Member Data Documentation	213
9.23.3.1 m_activeEditorIdx	213
9.23.3.2 m_editModesTabs	213
9.23.3.3 m_editorWorkspaceList	213
9.23.3.4 m_sidePanel	213
9.24 Brood::BroodUI::ElementSelection Class Reference	214
9.24.1 Member Function Documentation	214
9.24.1.1 Debugger()	215
9.24.1.2 DebugPrintSelectedElementID()	215
9.24.1.3 GetAlmostActiveElementIdPtr()	215
9.24.1.4 GetCurrActiveElementIdPtr()	215
9.24.1.5 GetHotElementIdFlag()	215
9.24.1.6 GetHotElementIdPtr()	216
9.24.1.7 GetLastActiveElementIdPtr()	216
9.24.1.8 SetAlmostActiveElementIdPtr()	216
9.24.1.9 SetCurrActiveElementIdPtr()	216
9.24.1.10 SetHotElementIdPtr()	217
9.24.1.11 SetHotElementIdPtrFlag()	217
9.24.1.12 SetLastActiveElementIdPtr()	217
9.24.2 Member Data Documentation	217
9.24.2.1 m_almostActiveElementIdPtr	218
9.24.2.2 m_currActiveElementIdPtr	218
9.24.2.3 m_hotElementFlag	218
9.24.2.4 m_hotElementIdPtr	218
9.24.2.5 m_lastActiveElementIdPtr	218
9.25 ElementSelection Class Reference	218
9.25.1 Detailed Description	219
9.26 Brood::Application:: FileAccess Class Reference	219
9.26.1 Constructor & Destructor Documentation	220
9.26.1.1 FileAccess()	220
9.26.1.2 ~FileAccess()	220
9.26.2 Member Function Documentation	220
9.26.2.1 CheckEOF()	220
9.26.2.2 CreateFile()	220
9.26.2.3 GetNextLine()	220
9.26.2.4 OpenFile()	221
9.26.2.5 RemoveAllContent()	221
9.26.2.6 Rewind()	221
9.26.2.7 WriteOneLineToFile()	221
9.26.3 Member Data Documentation	222
9.26.3.1 m_fileHandler	222

9.26.3.2 <code>m_filePath</code>	222
9.27 <code>Brood::Application::Components::GameDataManager</code> Class Reference	222
9.27.1 Constructor & Destructor Documentation	223
9.27.1.1 <code>GameManager()</code>	223
9.27.1.2 <code>~GameManager()</code>	223
9.27.2 Member Function Documentation	224
9.27.2.1 <code>Debugger()</code>	224
9.27.2.2 <code>Draw()</code>	224
9.27.2.3 <code>GetBoardPtr()</code>	224
9.27.2.4 <code>GetDataToSave()</code>	224
9.27.2.5 <code>GetDeckManagerPtr()</code>	225
9.27.2.6 <code>GetDicePtr()</code>	225
9.27.2.7 <code>GetDisplayCardPtr()</code>	225
9.27.2.8 <code>GetGameTitle()</code>	225
9.27.2.9 <code>GetPlayerManagerPtr()</code>	226
9.27.2.10 <code>InitializeGameManager()</code>	226
9.27.2.11 <code>InitializeGameDataManager()</code>	226
9.27.2.12 <code>LoadDataFromFile()</code>	226
9.27.2.13 <code>SaveDataToFile()</code>	227
9.27.2.14 <code>SetGameTitle()</code>	227
9.27.3 Member Data Documentation	227
9.27.3.1 <code>m_board</code>	227
9.27.3.2 <code>m_deckManager</code>	227
9.27.3.3 <code>m_dice</code>	228
9.27.3.4 <code>m_displayCard</code>	228
9.27.3.5 <code>m_gameTitle</code>	228
9.27.3.6 <code>m_playerManager</code>	228
9.28 <code>GameDataManager</code> Class Reference	228
9.28.1 Detailed Description	228
9.29 <code>Brood::Application::GameEditor</code> Class Reference	229
9.29.1 Constructor & Destructor Documentation	230
9.29.1.1 <code>GameEditor()</code>	231
9.29.1.2 <code>~GameEditor()</code>	231
9.29.2 Member Function Documentation	231
9.29.2.1 <code>Debugger()</code>	231
9.29.2.2 <code>Draw()</code>	231
9.29.2.3 <code>InitializeWorkSpace()</code>	232
9.29.2.4 <code>Update()</code>	232
9.29.2.5 <code>UpdateAllDisplayElement()</code>	232
9.29.2.6 <code>UpdateDeckNumber()</code>	232
9.29.2.7 <code>UpdateGameTitleInput()</code>	233
9.29.2.8 <code>UpdateIncorrectPenaltyPanelElement()</code>	233

9.29.2.9 UpdateMaximumPlayerNumber()	233
9.29.2.10 UpdateMinimumPlayerNumber()	233
9.29.2.11 UpdateMovementTypePanelElement()	234
9.29.3 Member Data Documentation	234
9.29.3.1 m_btnDeckDecNum	234
9.29.3.2 m_btnDeckIncNum	234
9.29.3.3 m_btnPlayerDecMaxnNum	234
9.29.3.4 m_btnPlayerDecMinNum	234
9.29.3.5 m_btnPlayerIncMaxNum	235
9.29.3.6 m_btnPlayerIncMinNum	235
9.29.3.7 m_ddiIncorrectPenalty	235
9.29.3.8 m_ddiMovementType	235
9.29.3.9 m_panelBodyPtr	235
9.29.3.10 m_titleScreenBtn	235
9.29.3.11 m_txtDeckNum	236
9.29.3.12 m_txtDeckNumPrompt	236
9.29.3.13 m_txtGameInput	236
9.29.3.14 m_txtGameNamePrompt	236
9.29.3.15 m_txtIncorrectPenaltyPromt	236
9.29.3.16 m_txtMovementTypePromt	236
9.29.3.17 m_txtPlayerMaxNum	237
9.29.3.18 m_txtPlayerMaxNumPrompt	237
9.29.3.19 m_txtPlayerMinNum	237
9.29.3.20 m_txtPlayerMinNumPrompt	237
9.29.3.21 m_txtSettingTitle	237
9.30 Brood::Application::HomeWorkSpace Class Reference	237
9.30.1 Constructor & Destructor Documentation	239
9.30.1.1 HomeWorkSpace()	239
9.30.1.2 ~HomeWorkSpace()	240
9.30.2 Member Function Documentation	240
9.30.2.1 Debugger()	240
9.30.2.2 Draw()	240
9.30.2.3 ExecuteCreateNewEditDialogBox()	240
9.30.2.4 ExecuteLoadPreviousEditDialogBox()	240
9.30.2.5 ExecuteMenuItem()	240
9.30.2.6 InitializeRibbonTabs()	241
9.30.2.7 InitializeWorkSpace()	241
9.30.2.8 LoadGameData()	241
9.30.2.9 ResetGameData()	242
9.30.2.10 SaveGameData()	242
9.30.2.11 SwitchToHomeWorkspace()	242
9.30.2.12 Update()	242

9.30.2.13 UpdateAllDisplayElement()	242
9.30.2.14 UpdateFileNameFeildDialog()	243
9.30.2.15 UpdateHomeWorkspace()	243
9.30.2.16 UpdateRibbonTabs()	243
9.30.3 Member Data Documentation	243
9.30.3.1 m_cancleBtn	243
9.30.3.2 m_createNewEditBtn	243
9.30.3.3 m_currGameInfoIdx	244
9.30.3.4 m_currWorkSpaceIdx	244
9.30.3.5 m_enterFileNameFeildTxt	244
9.30.3.6 m_fileNameTxt	244
9.30.3.7 m_gameInfoFilePathList	244
9.30.3.8 m_gameListTexture	244
9.30.3.9 m_gameOrEditorBtn	245
9.30.3.10 m_gameOrEditorForeground	245
9.30.3.11 m_importGameBtn	245
9.30.3.12 m_isHomeWorkspace	245
9.30.3.13 m_loadPreviousEditsBtn	245
9.30.3.14 m_openFileBtn	245
9.30.3.15 m_openFileBtnIdx	245
9.30.3.16 m_ribbonTabs	246
9.30.3.17 m_showEnterFilenameDialogBox	246
9.30.3.18 m_showImportDialogBox	246
9.30.3.19 m_toggleGameInfoLeftBtn	246
9.30.3.20 m_toggleGameInfoRightBtn	246
9.30.3.21 m_workSpacesList	246
9.31 HomeWorkSpace Class Reference	247
9.31.1 Detailed Description	247
9.32 Brood::BroodUI::Id Class Reference	247
9.32.1 Constructor & Destructor Documentation	248
9.32.1.1 Id() [1/2]	248
9.32.1.2 ~Id()	249
9.32.1.3 Id() [2/2]	249
9.32.2 Member Function Documentation	249
9.32.2.1 AddChild()	249
9.32.2.2 DeleteChildIdAtIdx()	249
9.32.2.3 GetChildIdAtIdx()	251
9.32.2.4 GetChildIdx()	251
9.32.2.5 GetElementID()	251
9.32.2.6 GetParentID()	252
9.32.2.7 GetParentIDPtr() [1/2]	252
9.32.2.8 GetParentIDPtr() [2/2]	252

9.32.2.9 GetTotalChildNum()	253
9.32.2.10 HasChild()	253
9.32.2.11 HasParent()	253
9.32.2.12 operator=()	253
9.32.2.13 SetParent()	254
9.32.3 Member Data Documentation	254
9.32.3.1 GLOBAL_ID_NUM	254
9.32.3.2 m_allChildPtrs	254
9.32.3.3 m_elementID	254
9.32.3.4 m_hasChilds	255
9.32.3.5 m_parentID	255
9.32.3.6 m_parentIDPtr	255
9.33 Id Class Reference	255
9.33.1 Detailed Description	255
9.34 Brood::BroodUI::MapIdToElement Class Reference	255
9.34.1 Member Function Documentation	256
9.34.1.1 AddToMap()	256
9.34.1.2 GetElementPtrFromMap()	256
9.34.1.3 GetMap()	257
9.34.1.4 RemoveFromMap()	257
9.34.2 Member Data Documentation	257
9.34.2.1 stm_mapper	257
9.35 MapIdToElement Class Reference	258
9.35.1 Detailed Description	258
9.36 Brood::BroodUI::MenuBar Class Reference	258
9.36.1 Constructor & Destructor Documentation	260
9.36.1.1 MenuBar() [1/2]	260
9.36.1.2 ~MenuBar()	260
9.36.1.3 MenuBar() [2/2]	260
9.36.2 Member Function Documentation	261
9.36.2.1 AddItemToMenu() [1/2]	261
9.36.2.2 AddItemToMenu() [2/2]	261
9.36.2.3 AddMenuToMenuBar() [1/2]	262
9.36.2.4 AddMenuToMenuBar() [2/2]	262
9.36.2.5 Debugger()	263
9.36.2.6 Draw()	263
9.36.2.7 GetMenuList()	263
9.36.2.8 operator=()	263
9.36.2.9 SetActiveOverlayColor()	264
9.36.2.10 SetBodyColor()	264
9.36.2.11 SetBodyPosition() [1/2]	264
9.36.2.12 SetBodyPosition() [2/2]	265

9.36.2.13 SetBodySize() [1/2]	265
9.36.2.14 SetBodySize() [2/2]	266
9.36.2.15 SetFont()	266
9.36.2.16 SetFontColor()	267
9.36.2.17 SetFontSize()	267
9.36.2.18 SetHotOverlayColor()	267
9.36.2.19 SetMenuBodySize()	268
9.36.2.20 SetMenuPos()	268
9.36.3 Member Data Documentation	268
9.36.3.1 m_menus	268
9.37 MenuBar Class Reference	268
9.37.1 Detailed Description	269
9.38 Brood::MouseHandler Class Reference	270
9.38.1 Member Enumeration Documentation	271
9.38.1.1 ENUM_MouseCursorType	271
9.38.2 Member Function Documentation	271
9.38.2.1 ChangeCursorTo()	271
9.38.2.2 Debugger()	272
9.38.2.3 GetCurrFrameLMSStatus()	272
9.38.2.4 GetCurrFrameMousePos()	272
9.38.2.5 GetCurrFrameRMSSStatus()	273
9.38.2.6 GetlastFrameLMSStatus()	273
9.38.2.7 GetLastFrameMousePos()	273
9.38.2.8 GetlastFrameRMSSStatus()	273
9.38.2.9 InitializeCursor()	273
9.38.2.10 IsLeftButtonHold()	274
9.38.2.11 IsLeftButtonPressed()	274
9.38.2.12 IsLeftButtonReleased()	274
9.38.2.13 IsRightButtonHold()	274
9.38.2.14 IsRightButtonPressed()	275
9.38.2.15 IsRightButtonReleased()	275
9.38.2.16 UpdateMouseButtonStatus()	275
9.38.2.17 UpdateMousePos()	275
9.38.3 Member Data Documentation	276
9.38.3.1 m_currFrameLeftMouseButtonPressed	276
9.38.3.2 m_currFrameMousePos	276
9.38.3.3 m_currFrameRightMouseButtonPressed	276
9.38.3.4 m_cursor	276
9.38.3.5 m_lastFrameLeftMouseButtonPressed	276
9.38.3.6 m_lastFrameMousePos	276
9.38.3.7 m_lastFrameRightMouseButtonPressed	276
9.39 Brood::Application::Components::Path Class Reference	277

9.39.1 Constructor & Destructor Documentation	279
9.39.1.1 Path() [1/2]	279
9.39.1.2 ~Path()	279
9.39.1.3 Path() [2/2]	279
9.39.2 Member Function Documentation	279
9.39.2.1 AddPlayerToList()	279
9.39.2.2 DeletePlayerFromList()	280
9.39.2.3 Draw()	280
9.39.2.4 DrawPath()	280
9.39.2.5 GetBridgeEndPathPtr()	281
9.39.2.6 GetDataToSave()	281
9.39.2.7 GetDeckIdx()	281
9.39.2.8 GetDeckPtr()	281
9.39.2.9 GetDrawLine()	282
9.39.2.10 GetForceDiceRoll()	282
9.39.2.11 GetNextPathPtr()	282
9.39.2.12 GetNumCardDraw()	282
9.39.2.13 GetPlayerListPtr()	283
9.39.2.14 GetPreviousPathPtr()	283
9.39.2.15 GetTileCenter()	283
9.39.2.16 GetTilePtr() [1/2]	283
9.39.2.17 GetTilePtr() [2/2]	284
9.39.2.18 GetTileType()	284
9.39.2.19 InitializePath()	284
9.39.2.20 operator=(())	284
9.39.2.21 SetBridgeEndPathPtr()	285
9.39.2.22 SetDeckIdx()	285
9.39.2.23 SetDeckPtr()	285
9.39.2.24 SetForceDiceRoll()	286
9.39.2.25 SetNextPathPtr()	286
9.39.2.26 SetNumCardDraw()	286
9.39.2.27 SetPreviousPathPtr()	286
9.39.2.28 SetTilePtr()	287
9.39.2.29 SetTileType()	287
9.39.2.30 ToggleDrawLine()	287
9.39.2.31 UpdatePathLines()	287
9.39.3 Member Data Documentation	288
9.39.3.1 m_bridgeEndPathPtr	288
9.39.3.2 m_bridgePathLine	288
9.39.3.3 m_deckIdx	288
9.39.3.4 m_deckPtr	288
9.39.3.5 m_drawLine	288

9.39.3.6 m_forceDiceRoll	289
9.39.3.7 m_nextPathLine	289
9.39.3.8 m_nextPathPtr	289
9.39.3.9 m_numCardDraw	289
9.39.3.10 m_playerPtrList	289
9.39.3.11 m_previousPathPtr	289
9.39.3.12 m_tilePtr	290
9.39.3.13 m_tileType	290
9.40 Path Class Reference	290
9.40.1 Detailed Description	290
9.41 Brood::Application::Components::Player Class Reference	290
9.41.1 Constructor & Destructor Documentation	292
9.41.1.1 Player() [1/2]	292
9.41.1.2 ~Player()	292
9.41.1.3 Player() [2/2]	292
9.41.2 Member Function Documentation	293
9.41.2.1 Draw()	293
9.41.2.2 GetCurrPathPtr()	293
9.41.2.3 GetDataToSave()	293
9.41.2.4 GetPlayerBody()	293
9.41.2.5 GetPlayerSpriteBody()	294
9.41.2.6 GetPositionOffsetX()	294
9.41.2.7 GetPositionOffsetY()	294
9.41.2.8 InitializePlayer()	294
9.41.2.9 LoadDataFromFile()	295
9.41.2.10 operator=()	295
9.41.2.11 SaveDataToFile()	295
9.41.2.12 SetPlayerSizeX()	295
9.41.2.13 SetPlayerSizeY()	296
9.41.2.14 SetPositionOffsetX()	296
9.41.2.15 SetPositionOffsetY()	296
9.41.2.16 UpdatePathptr()	297
9.41.2.17 UpdatePosition()	297
9.41.3 Member Data Documentation	297
9.41.3.1 m_playerBody	297
9.41.3.2 m_playerCurrPathPtr	297
9.41.3.3 m_positionOffsetX	298
9.41.3.4 m_positionOffsetY	298
9.41.3.5 m_spriteBody	298
9.42 Player Class Reference	298
9.42.1 Detailed Description	298
9.43 Brood::Application::PlayerEditor Class Reference	299

9.43.1 Constructor & Destructor Documentation	301
9.43.1.1 PlayerEditor()	301
9.43.1.2 ~PlayerEditor()	301
9.43.2 Member Function Documentation	302
9.43.2.1 Debugger()	302
9.43.2.2 Draw()	302
9.43.2.3 InitializeWorkSpace()	302
9.43.2.4 Update()	303
9.43.2.5 UpdateAllDisplayElement()	303
9.43.2.6 UpdateCurrPlayerFileTexture()	303
9.43.2.7 UpdateCurrPlayerOffsetX()	304
9.43.2.8 UpdateCurrPlayerOffsetY()	304
9.43.2.9 UpdateCurrPlayerSizeX()	304
9.43.2.10 UpdateCurrPlayerSizeY()	304
9.43.2.11 UpdateCurrSelectedPlayerIdx()	305
9.43.2.12 UpdateStartColNumber()	305
9.43.2.13 UpdateStartRowNumber()	305
9.43.3 Member Data Documentation	305
9.43.3.1 m_boardPtr	306
9.43.3.2 m_btnPlayerDecCurrIdx	306
9.43.3.3 m_btnPlayerDecCurrIdxOffsetX	306
9.43.3.4 m_btnPlayerDecCurrIdxOffsetY	306
9.43.3.5 m_btnPlayerDecCurrIdxSizeX	306
9.43.3.6 m_btnPlayerDecCurrIdxSizeY	306
9.43.3.7 m_btnPlayerDecStartColNum	307
9.43.3.8 m_btnPlayerDecStartRowNum	307
9.43.3.9 m_btnPlayerIncCurrIdx	307
9.43.3.10 m_btnPlayerIncCurrIdxOffsetX	307
9.43.3.11 m_btnPlayerIncCurrIdxOffsetY	307
9.43.3.12 m_btnPlayerIncCurrIdxSizeX	307
9.43.3.13 m_btnPlayerIncCurrIdxSizeY	308
9.43.3.14 m_btnPlayerIncStartColNum	308
9.43.3.15 m_btnPlayerIncStartRowNum	308
9.43.3.16 m_btnPlayerOpenFile	308
9.43.3.17 m_ddiDrawPlayerAtCurrIdx	308
9.43.3.18 m_panelBodyPtr	308
9.43.3.19 m_playerManagerPtr	309
9.43.3.20 m_txtDrawPlayerAtCurrIdxPromt	309
9.43.3.21 m_txtPlayerCurrIdx	309
9.43.3.22 m_txtPlayerCurrIdxOffsetX	309
9.43.3.23 m_txtPlayerCurrIdxOffsetXPrompt	309
9.43.3.24 m_txtPlayerCurrIdxOffsetY	309

9.43.3.25 m_txtPlayerCurrlIdxOffsetYPrompt	310
9.43.3.26 m_txtPlayerCurrlIdxPrompt	310
9.43.3.27 m_txtPlayerCurrlIdxSizeX	310
9.43.3.28 m_txtPlayerCurrlIdxSizeXPrompt	310
9.43.3.29 m_txtPlayerCurrlIdxSizeY	310
9.43.3.30 m_txtPlayerCurrlIdxSizeYPrompt	310
9.43.3.31 m_txtPlayerFileInput	311
9.43.3.32 m_txtplayerFileNamePrompt	311
9.43.3.33 m_txtPlayerStartColNum	311
9.43.3.34 m_txtPlayerStartColPrompt	311
9.43.3.35 m_txtPlayerStartRowNum	311
9.43.3.36 m_txtPlayerStartRowPrompt	311
9.43.3.37 m_txtSettingTitle	312
9.44 PlayerEditor Class Reference	312
9.44.1 Detailed Description	312
9.45 Brood::Application::Components::PlayerManager Class Reference	312
9.45.1 Constructor & Destructor Documentation	314
9.45.1.1 PlayerManager() [1/2]	314
9.45.1.2 ~PlayerManager()	314
9.45.1.3 PlayerManager() [2/2]	314
9.45.2 Member Function Documentation	314
9.45.2.1 Draw()	314
9.45.2.2 GetAllPlayerBegin()	315
9.45.2.3 GetAllPlayerEnd()	315
9.45.2.4 GetCurrActivePlayerIdx()	315
9.45.2.5 GetDataToSave()	316
9.45.2.6 GetMaxPlayer()	316
9.45.2.7 GetMinPlayer()	316
9.45.2.8 GetNextPlayer()	316
9.45.2.9 GetPlayerAtCurrlIdx()	317
9.45.2.10 GetPlayerStartPath()	317
9.45.2.11 InitializePlayerManager() [1/2]	317
9.45.2.12 InitializePlayerManager() [2/2]	317
9.45.2.13 LoadDataFromFile()	318
9.45.2.14 operator=()	318
9.45.2.15 ReplacaePlayerAt()	318
9.45.2.16 SaveDataToFile()	319
9.45.2.17 SetCurrActivePlayerIdx()	319
9.45.2.18 SetMaxPlayer()	319
9.45.2.19 SetMinPlayer()	319
9.45.2.20 SetPathForPlayerAtCurrlIdx()	320
9.45.2.21 SetPlayerStartPath()	320

9.45.3 Member Data Documentation	320
9.45.3.1 m_allPlayers	320
9.45.3.2 m_currActivePlayerIdx	320
9.45.3.3 m_maxPlayer	321
9.45.3.4 m_minPlayer	321
9.45.3.5 m_startPathPtr	321
9.46 PlayerManager Class Reference	321
9.46.1 Detailed Description	321
9.47 Brood::SpriteHandler Class Reference	321
9.47.1 Constructor & Destructor Documentation	323
9.47.1.1 SpriteHandler() [1/2]	323
9.47.1.2 ~SpriteHandler()	323
9.47.1.3 SpriteHandler() [2/2]	323
9.47.2 Member Function Documentation	323
9.47.2.1 Debugger()	323
9.47.2.2 GetcurrSpriteIndex()	324
9.47.2.3 GetSpritHeight()	324
9.47.2.4 GetSpritLenght()	324
9.47.2.5 GetTextureDirectoryPath()	324
9.47.2.6 GetTextureFileName()	325
9.47.2.7 operator=()	325
9.47.2.8 RemoveTexture()	325
9.47.2.9 SetSpriteFromTexture()	325
9.47.2.10 SetSpriteHeight()	326
9.47.2.11 SetSpriteLength()	326
9.47.2.12 SetTextureFromFilePath() [1/2]	326
9.47.2.13 SetTextureFromFilePath() [2/2]	327
9.47.2.14 SetTextureFromSavedFilePath()	327
9.47.3 Member Data Documentation	327
9.47.3.1 m_body	327
9.47.3.2 m_currSpriteIndex	328
9.47.3.3 m_spriteHeight	328
9.47.3.4 m_spriteLength	328
9.47.3.5 m_texture	328
9.47.3.6 m_textureDirectoryPath	328
9.47.3.7 m_textureFileName	329
9.48 SpriteHandler Class Reference	329
9.48.1 Detailed Description	329
9.49 ST_ApplicationData Struct Reference	329
9.49.1 Detailed Description	330
9.50 Brood::Application::Data::ST_BoardData Struct Reference	330
9.50.1 Member Function Documentation	330

9.50.1.1 <code>GetString()</code>	331
9.50.1.2 <code>PopulateFromString()</code>	331
9.50.2 Member Data Documentation	331
9.50.2.1 <code>stm_boardPosX</code>	331
9.50.2.2 <code>stm_boardPosY</code>	331
9.50.2.3 <code>stm_boardSizeX</code>	331
9.50.2.4 <code>stm_boardSizeY</code>	332
9.50.2.5 <code>stm_currActiveNumCol</code>	332
9.50.2.6 <code>stm_currActiveNumRow</code>	332
9.50.2.7 <code>stm_numCol</code>	332
9.50.2.8 <code>stm_numRow</code>	332
9.51 <code>ST_BoardData</code> Struct Reference	332
9.51.1 Detailed Description	333
9.52 <code>St_CardInfo</code> Class Reference	333
9.52.1 Detailed Description	333
9.53 <code>Brood::Application::Data::ST_CardInfoPrefabData</code> Struct Reference	333
9.53.1 Member Function Documentation	334
9.53.1.1 <code>GetString()</code>	334
9.53.1.2 <code>PopulateFromString()</code>	334
9.53.2 Member Data Documentation	334
9.53.2.1 <code>stm_correctAnswer</code>	334
9.53.2.2 <code>stm_correctNumSteps</code>	334
9.53.2.3 <code>stm_incorrectNumSteps</code>	335
9.53.2.4 <code>stm_question</code>	335
9.53.2.5 <code>stm_time</code>	335
9.54 <code>ST_CardPrefabData</code> Struct Reference	335
9.54.1 Detailed Description	335
9.55 <code>Brood::Application::StaticVariables::ST_ColorVariables</code> Struct Reference	335
9.55.1 Member Function Documentation	336
9.55.1.1 <code>GetRandomColor()</code>	336
9.55.2 Member Data Documentation	336
9.55.2.1 <code>stm_AppPrimaryColor</code>	336
9.55.2.2 <code>stm_AppSecondaryColor</code>	337
9.55.2.3 <code>stm_Black</code>	337
9.55.2.4 <code>stm_CurrActiveOverlay</code>	337
9.55.2.5 <code>stm_ErrorColor</code>	337
9.55.2.6 <code>stm_HotOverlay</code>	337
9.55.2.7 <code>stm_MainMenu</code>	337
9.55.2.8 <code>stm_White</code>	338
9.56 <code>ST_ColorVariables</code> Struct Reference	338
9.56.1 Detailed Description	338
9.57 <code>Brood::Application::Data::ST_DeckManagerData</code> Struct Reference	338

9.57.1 Member Function Documentation	339
9.57.1.1 GetString()	339
9.57.1.2 PopulateFromString()	339
9.57.2 Member Data Documentation	339
9.57.2.1 stm_currDecksIdx	339
9.57.2.2 stm_incorrectPenalty	339
9.57.2.3 stm_movementType	340
9.57.2.4 stm_numDecks	340
9.58 ST_DeckManagerData Struct Reference	340
9.58.1 Detailed Description	340
9.59 Brood::Application::Data::ST_DeckPrefabData Struct Reference	340
9.59.1 Member Function Documentation	341
9.59.1.1 GetString()	341
9.59.1.2 PopulateFromString()	341
9.59.2 Member Data Documentation	342
9.59.2.1 stm_cardInitFilename	342
9.59.2.2 stm_currActiveCardIdx	342
9.59.2.3 stm_deckPosX	342
9.59.2.4 stm_deckPosY	342
9.59.2.5 stm_deckSizeX	342
9.59.2.6 stm_deckSizeY	343
9.59.2.7 stm_numTotalCard	343
9.59.2.8 stm_textureFileName	343
9.59.2.9 stm_undealtCardIdx	343
9.60 ST_DeckPrefabData Struct Reference	343
9.60.1 Detailed Description	343
9.61 Brood::Application::Data::ST_DicePrefabData Struct Reference	344
9.61.1 Member Function Documentation	344
9.61.1.1 GetString()	344
9.61.1.2 PopulateFromString()	344
9.61.2 Member Data Documentation	345
9.61.2.1 stm_dicePosX	345
9.61.2.2 stm_dicePosY	345
9.61.2.3 stm_diceSizeX	345
9.61.2.4 stm_diceSizeY	345
9.61.2.5 stm_numSides	345
9.61.2.6 stm_textureFileName	346
9.62 ST_DicePrefabData Struct Reference	346
9.62.1 Detailed Description	346
9.63 Brood::Application::Data::ST_DisplayCardData Struct Reference	346
9.63.1 Member Function Documentation	349
9.63.1.1 GetString()	349

9.63.1.2 PopulateFromString()	349
9.63.2 Member Data Documentation	350
9.63.2.1 stm_answerPromptPositionX	350
9.63.2.2 stm_answerPromptPositonY	350
9.63.2.3 stm_answerPromptSizeX	350
9.63.2.4 stm_answerPromptSizeY	350
9.63.2.5 stm_answerValuePositionX	350
9.63.2.6 stm_answerValuePositonY	351
9.63.2.7 stm_answerValueSizeX	351
9.63.2.8 stm_answerValueSizeY	351
9.63.2.9 stm_backTimePromptPositionX	351
9.63.2.10 stm_backTimePromptPositonY	351
9.63.2.11 stm_backTimePromptSizeX	351
9.63.2.12 stm_backTimePromptSizeY	352
9.63.2.13 stm_backTimeValuePositionX	352
9.63.2.14 stm_backTimeValuePositonY	352
9.63.2.15 stm_backTimeValueSizeX	352
9.63.2.16 stm_backTimeValueSizeY	352
9.63.2.17 stm_displayCardBackTextureFilename	352
9.63.2.18 stm_displayCardFrontTextureFilename	353
9.63.2.19 stm_displayCardPositionX	353
9.63.2.20 stm_displayCardPositionY	353
9.63.2.21 stm_displayCardSizeX	353
9.63.2.22 stm_displayCardSizeY	353
9.63.2.23 stm_downPromptPositionX	353
9.63.2.24 stm_downPromptPositonY	354
9.63.2.25 stm_downPromptSizeX	354
9.63.2.26 stm_downPromptSizeY	354
9.63.2.27 stm_downValuePositionX	354
9.63.2.28 stm_downValuePositonY	354
9.63.2.29 stm_downValueSizeX	354
9.63.2.30 stm_downValueSizeY	355
9.63.2.31 stm_frontTimePromptPositionX	355
9.63.2.32 stm_frontTimePromptPositonY	355
9.63.2.33 stm_frontTimePromptSizeX	355
9.63.2.34 stm_frontTimePromptSizeY	355
9.63.2.35 stm_frontTimeValuePositionX	355
9.63.2.36 stm_frontTimeValuePositonY	356
9.63.2.37 stm_frontTimeValueSizeX	356
9.63.2.38 stm_frontTimeValueSizeY	356
9.63.2.39 stm_questionPromptPositionX	356
9.63.2.40 stm_questionPromptPositonY	356

9.63.2.41 <code>stm_questionPromptSizeX</code>	356
9.63.2.42 <code>stm_questionPromptSizeY</code>	357
9.63.2.43 <code>stm_questionValuePositionX</code>	357
9.63.2.44 <code>stm_questionValuePositonY</code>	357
9.63.2.45 <code>stm_questionValueSizeX</code>	357
9.63.2.46 <code>stm_questionValueSizeY</code>	357
9.63.2.47 <code>stm_submitButtonValuePositionX</code>	357
9.63.2.48 <code>stm_submitButtonValuePositonY</code>	358
9.63.2.49 <code>stm_submitButtonValueSizeX</code>	358
9.63.2.50 <code>stm_submitButtonValueSizeY</code>	358
9.63.2.51 <code>stm_turnCardPromptPositionX</code>	358
9.63.2.52 <code>stm_turnCardPromptPositonY</code>	358
9.63.2.53 <code>stm_turnCardPromptSizeX</code>	358
9.63.2.54 <code>stm_turnCardPromptSizeY</code>	359
9.63.2.55 <code>stm_upPromptPositionX</code>	359
9.63.2.56 <code>stm_upPromptPositonY</code>	359
9.63.2.57 <code>stm_upPromptSizeX</code>	359
9.63.2.58 <code>stm_upPromptSizeY</code>	359
9.63.2.59 <code>stm_upValuePositionX</code>	359
9.63.2.60 <code>stm_upValuePositonY</code>	360
9.63.2.61 <code>stm_upValueSizeX</code>	360
9.63.2.62 <code>stm_upValueSizeY</code>	360
9.64 <code>ST_DisplayCardData</code> Struct Reference	360
9.64.1 Detailed Description	360
9.65 <code>Brood::Application::StaticVariables::ST_Folders</code> Struct Reference	360
9.65.1 Member Data Documentation	361
9.65.1.1 <code>stm_assets</code>	361
9.65.1.2 <code>stm_cardTextures</code>	361
9.65.1.3 <code>stm_cwd</code>	362
9.65.1.4 <code>stm_data</code>	362
9.65.1.5 <code>stm_deckTextures</code>	362
9.65.1.6 <code>stm_diceTextures</code>	362
9.65.1.7 <code>stm_fonts</code>	362
9.65.1.8 <code>stm_playerTextures</code>	363
9.65.1.9 <code>stm_textures</code>	363
9.65.1.10 <code>stm_tileTextures</code>	363
9.66 <code>Brood::Application::Data::ST_GameData</code> Struct Reference	363
9.66.1 Member Function Documentation	364
9.66.1.1 <code>GetString()</code>	364
9.66.1.2 <code>PopulateFromString()</code>	364
9.66.2 Member Data Documentation	364
9.66.2.1 <code>stm_gameTitle</code>	364

9.67 Brood::Application::StaticVariables::ST_GlobalCoreVariables Struct Reference	364
9.67.1 Member Data Documentation	365
9.67.1.1 stm_exit	365
9.67.1.2 stm_font	365
9.67.1.3 stm_is_debug_mode	365
9.67.1.4 stm_panelPercentage	365
9.67.1.5 stm_seed	366
9.67.1.6 stm_window_height	366
9.67.1.7 stm_window_width	366
9.68 Brood::Application::Data::ST_PathPrefabData Struct Reference	366
9.68.1 Member Function Documentation	367
9.68.1.1 GetString()	367
9.68.1.2 PopulateFromString()	367
9.68.2 Member Data Documentation	368
9.68.2.1 stm_assignedDeckId	368
9.68.2.2 stm_ColNum	368
9.68.2.3 stm_endBridgeTileColNum	368
9.68.2.4 stm_endBridgeTileRowNum	368
9.68.2.5 stm_forceDiceRoll	368
9.68.2.6 stm_nextTileColNum	368
9.68.2.7 stm_nextTileRowNum	369
9.68.2.8 stm_numberCardDraw	369
9.68.2.9 stm_RowNum	369
9.68.2.10 stm_textureFileName	369
9.68.2.11 stm_tileType	369
9.69 Brood::Application::Data::ST_PlayerManagerData Struct Reference	369
9.69.1 Member Function Documentation	370
9.69.1.1 GetString()	370
9.69.1.2 PopulateFromString()	370
9.69.2 Member Data Documentation	370
9.69.2.1 stm_currPlayerIdx	371
9.69.2.2 stm_maxPlayer	371
9.69.2.3 stm_minPlayer	371
9.70 ST_PlayerManagerData Struct Reference	371
9.70.1 Detailed Description	371
9.71 Brood::Application::Data::ST_PlayerPrefabData Struct Reference	371
9.71.1 Member Function Documentation	372
9.71.1.1 GetString()	372
9.71.1.2 PopulateFromString()	372
9.71.2 Member Data Documentation	373
9.71.2.1 stm_currCol	373
9.71.2.2 stm_currRow	373

9.71.2.3 <code>stm_playerOffsetX</code>	373
9.71.2.4 <code>stm_playerOffsetY</code>	373
9.71.2.5 <code>stm_playerSizeX</code>	373
9.71.2.6 <code>stm_playerSizeY</code>	374
9.71.2.7 <code>stm_textureFileName</code>	374
9.72 <code>ST_PlayerPrefabData</code> Struct Reference	374
9.72.1 Detailed Description	374
9.73 <code>ST_TilePrefabData</code> Struct Reference	374
9.73.1 Detailed Description	374
9.74 <code>Brood::BroodUI::TextBox</code> Class Reference	375
9.74.1 Constructor & Destructor Documentation	376
9.74.1.1 <code>TextBox()</code> [1/2]	376
9.74.1.2 <code>~TextBox()</code>	377
9.74.1.3 <code>TextBox()</code> [2/2]	377
9.74.2 Member Function Documentation	377
9.74.2.1 <code>Debugger()</code>	377
9.74.2.2 <code>DeleteLastChar()</code>	377
9.74.2.3 <code>DoElement()</code>	378
9.74.2.4 <code>Draw()</code>	378
9.74.2.5 <code>DyCreateTextBox()</code> [1/2]	378
9.74.2.6 <code>DyCreateTextBox()</code> [2/2]	379
9.74.2.7 <code>InputLogic()</code>	380
9.74.2.8 <code>IsEditable()</code>	380
9.74.2.9 <code>IsEnterPressed()</code>	380
9.74.2.10 <code>IsSelected()</code>	380
9.74.2.11 <code>operator=()</code>	380
9.74.2.12 <code>SetEditable()</code>	381
9.74.2.13 <code>SetEnterPressedFalse()</code>	381
9.74.2.14 <code>SetLimit()</code>	381
9.74.2.15 <code>SetPlaceHolderText()</code>	382
9.74.2.16 <code>SetSelected()</code>	382
9.74.2.17 <code>SetText()</code>	382
9.74.2.18 <code>TypeOn()</code>	382
9.74.3 Member Data Documentation	383
9.74.3.1 <code>m_enterPressed</code>	383
9.74.3.2 <code>m_hasLimit</code>	383
9.74.3.3 <code>m_isEditable</code>	383
9.74.3.4 <code>m_limit</code>	383
9.74.3.5 <code>m_ossText</code>	384
9.74.3.6 <code>m_placeHolderText</code>	384
9.75 <code>TextBox</code> Class Reference	384
9.75.1 Detailed Description	384

9.76 Brood::Application::TileEditor Class Reference	386
9.76.1 Constructor & Destructor Documentation	388
9.76.1.1 TileEditor()	388
9.76.1.2 ~TileEditor()	389
9.76.2 Member Function Documentation	389
9.76.2.1 Debugger()	389
9.76.2.2 Draw()	389
9.76.2.3 InitializeWorkSpace()	390
9.76.2.4 Update()	390
9.76.2.5 UpdateAllDisplayElement()	390
9.76.2.6 UpdateAssinedDeckIdx()	390
9.76.2.7 UpdateCorrectCardNumber()	391
9.76.2.8 UpdateEndBridgeColNumber()	391
9.76.2.9 UpdateEndBridgeRowNumber()	391
9.76.2.10 UpdateForceDiceRoll()	391
9.76.2.11 UpdateNextPathColNumber()	392
9.76.2.12 UpdateNextPathRowNumber()	392
9.76.2.13 UpdateTileColNumber()	392
9.76.2.14 UpdateTileFileTexture()	392
9.76.2.15 UpdateTileRowNumber()	393
9.76.2.16 UpdateTileType()	393
9.76.3 Member Data Documentation	393
9.76.3.1 m_btnTileBgOpenFile	393
9.76.3.2 m_btnTileDecAssignedDeckIdx	393
9.76.3.3 m_btnTileDecCurrCol	394
9.76.3.4 m_btnTileDecCurrRow	394
9.76.3.5 m_btnTileDecEndBridgeCol	394
9.76.3.6 m_btnTileDecEndBridgeRow	394
9.76.3.7 m_btnTileDecNextPathCol	394
9.76.3.8 m_btnTileDecNextPathRow	394
9.76.3.9 m_btnTileDecNumCorrectCard	395
9.76.3.10 m_btnTileIncAssignedDeckIdx	395
9.76.3.11 m_btnTileIncCurrCol	395
9.76.3.12 m_btnTileIncCurrRow	395
9.76.3.13 m_btnTileIncEndBridgeCol	395
9.76.3.14 m_btnTileIncEndBridgeRow	395
9.76.3.15 m_btnTileIncNextPathCol	396
9.76.3.16 m_btnTileIncNextPathRow	396
9.76.3.17 m_btnTileIncNumCorrectCard	396
9.76.3.18 m_ddiTileForceDiceRoll	396
9.76.3.19 m_ddiTileType	396
9.76.3.20 m_panelBodyPtr	396

9.76.3.21 m_txtSettingTitle	397
9.76.3.22 m_txtTileAssignedDeckIdx	397
9.76.3.23 m_txtTileAssignedDeckIdxPromt	397
9.76.3.24 m_txtTileBgFileInput	397
9.76.3.25 m_txtTileBgFileNamePrompt	397
9.76.3.26 m_txtTileCurrCol	397
9.76.3.27 m_txtTileCurrColPromt	398
9.76.3.28 m_txtTileCurrRow	398
9.76.3.29 m_txtTileCurrRowPromt	398
9.76.3.30 m_txtTileEndBridgeCol	398
9.76.3.31 m_txtTileEndBridgeColPromt	398
9.76.3.32 m_txtTileEndBridgeRow	398
9.76.3.33 m_txtTileEndBridgeRowPromt	399
9.76.3.34 m_txtTileForceDiceRollPromt	399
9.76.3.35 m_txtTileNextPathCol	399
9.76.3.36 m_txtTileNextPathColPromt	399
9.76.3.37 m_txtTileNextPathRow	399
9.76.3.38 m_txtTileNextPathRowPromt	399
9.76.3.39 m_txtTileNumCorrectCard	400
9.76.3.40 m_txtTileNumCorrectCardPromt	400
9.76.3.41 m_txtTileTypePromt	400
9.77 TileEditor Class Reference	400
9.77.1 Detailed Description	400
9.78 Brood::Application::Components::Tiles Class Reference	401
9.78.1 Constructor & Destructor Documentation	402
9.78.1.1 Tiles() [1/2]	402
9.78.1.2 ~Tiles()	402
9.78.1.3 Tiles() [2/2]	402
9.78.2 Member Function Documentation	402
9.78.2.1 GetCol()	402
9.78.2.2 GetRow()	403
9.78.2.3 GetRowAndCol()	403
9.78.2.4 operator=()	403
9.78.2.5 SetCol()	403
9.78.2.6 SetRow()	404
9.78.2.7 SetRowAndCol()	404
9.78.2.8 UpdateTile()	404
9.78.3 Member Data Documentation	405
9.78.3.1 m_tileColNum	405
9.78.3.2 m_tileRowNum	405
9.79 Tiles Class Reference	405
9.79.1 Detailed Description	405

9.80 Brood::BroodUI::UIElement Class Reference	406
9.80.1 Constructor & Destructor Documentation	408
9.80.1.1 UIElement() [1/2]	408
9.80.1.2 ~UIElement()	408
9.80.1.3 UIElement() [2/2]	409
9.80.2 Member Function Documentation	409
9.80.2.1 Debugger()	409
9.80.2.2 DoElement()	409
9.80.2.3 Draw()	410
9.80.2.4 GetActiveOverlayColor()	411
9.80.2.5 GetBody()	411
9.80.2.6 GetBodyColor()	411
9.80.2.7 GetBodyPosition()	411
9.80.2.8 GetBodySize()	412
9.80.2.9 GetElementIdPtr()	412
9.80.2.10 GetElementType()	412
9.80.2.11 GetFontSize()	412
9.80.2.12 GetHotOverlayColor()	413
9.80.2.13 GetText()	413
9.80.2.14 IsAlmostActiveElement()	413
9.80.2.15 IsCurrActiveElement()	413
9.80.2.16 IsHotElement()	414
9.80.2.17 IsMouseOverElement()	414
9.80.2.18 IsSelected()	414
9.80.2.19 operator=(())	414
9.80.2.20 SetActiveOverlayColor()	415
9.80.2.21 SetBodyColor()	415
9.80.2.22 SetBodyPosition() [1/2]	415
9.80.2.23 SetBodyPosition() [2/2]	416
9.80.2.24 SetBodySize() [1/2]	416
9.80.2.25 SetBodySize() [2/2]	417
9.80.2.26 SetDrawOverlay()	417
9.80.2.27 SetFont()	417
9.80.2.28 SetFontColor()	417
9.80.2.29 SetFontSize()	417
9.80.2.30 SetHotOverlayColor()	418
9.80.2.31 SetSelected()	418
9.80.2.32 SetText()	418
9.80.2.33 SetTextPosition()	419
9.80.3 Member Data Documentation	419
9.80.3.1 m_activeOverlayColor	419
9.80.3.2 m_body	419

9.80.3.3 m_bodyOverLay	419
9.80.3.4 m_debugTextSave	420
9.80.3.5 m_drawOverlay	420
9.80.3.6 m_drawText	420
9.80.3.7 m_elementId	420
9.80.3.8 m_elementType	420
9.80.3.9 m_font	420
9.80.3.10 m_fontSize	421
9.80.3.11 m_hotOverlayColor	421
9.80.3.12 m_isSelected	421
9.80.3.13 m_text	421
9.80.3.14 m_textContent	421
9.81 UIElement Class Reference	421
9.81.1 Detailed Description	422
9.82 Brood::Application::WorkSpace Class Reference	422
9.82.1 Constructor & Destructor Documentation	424
9.82.1.1 WorkSpace()	424
9.82.1.2 ~WorkSpace()	424
9.82.2 Member Function Documentation	425
9.82.2.1 Debugger()	425
9.82.2.2 Draw()	425
9.82.2.3 DyCreateButton() [1/2]	425
9.82.2.4 DyCreateButton() [2/2]	426
9.82.2.5 DyCreateDeclncPannelElement()	427
9.82.2.6 DyCreateDropDownInput() [1/2]	427
9.82.2.7 DyCreateDropDownInput() [2/2]	428
9.82.2.8 DyCreateDropdownInputPannelElement()	428
9.82.2.9 DyCreateFileInputPannelElement()	429
9.82.2.10 DyCreateTextBox() [1/2]	430
9.82.2.11 DyCreateTextBox() [2/2]	431
9.82.2.12 DyCreateTextInputPannelElement()	431
9.82.2.13 InitializeWorkSpace()	432
9.82.2.14 SetFont()	432
9.82.2.15 SetGameDataManager()	432
9.82.2.16 Update()	433
9.82.2.17 UpdateAllDisplayElement()	433
9.82.2.18 UpdateDeclncPosX()	433
9.82.2.19 UpdateDeclncPosY()	434
9.82.2.20 UpdateDeclncSizeX()	434
9.82.2.21 UpdateDeclncSizeY()	435
9.82.3 Member Data Documentation	435
9.82.3.1 m_font	435

9.82.3.2 mGameData	435
9.82.3.3 mIsDebugger	436
9.82.3.4 mUnNamedUIList	436
9.83 WorkSpace Class Reference	436
9.83.1 Detailed Description	436
10 File Documentation	437
10.1 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Application.cpp File Reference	437
10.1.1 Detailed Description	437
10.2 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Application.h File Reference	437
10.2.1 Detailed Description	438
10.3 Application.h	438
10.4 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Board.cpp File Reference	439
10.4.1 Detailed Description	439
10.5 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Board.h File Reference	439
10.5.1 Detailed Description	440
10.6 Board.h	440
10.7 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/BoardEditor.cpp File Reference	442
10.7.1 Detailed Description	442
10.8 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/BoardEditor.h File Reference	442
10.8.1 Detailed Description	443
10.9 BoardEditor.h	443
10.10 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Button.cpp File Reference	445
10.10.1 Detailed Description	445
10.11 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Button.h File Reference	445
10.11.1 Detailed Description	446
10.12 Button.h	446
10.13 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/CardEditor.cpp File Reference	447
10.13.1 Detailed Description	447
10.14 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/CardEditor.h File Reference	447
10.14.1 Detailed Description	448
10.15 CardEditor.h	448
10.16 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/CardInfo.cpp File Reference	458
10.17 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/CardInfo.h File Reference	458
10.17.1 Detailed Description	458
10.18 CardInfo.h	459
10.19 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Data.cpp File Reference	460
10.20 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Data.h File Reference	460

10.20.1 Detailed Description	461
10.21 Data.h	461
10.22 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Deck.cpp File Reference	467
10.23 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Deck.h File Reference	467
10.23.1 Detailed Description	467
10.24 Deck.h	468
10.25 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DeckEditor.cpp File Reference	469
10.25.1 Detailed Description	469
10.26 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DeckEditor.h File Reference	470
10.26.1 Detailed Description	470
10.27 DeckEditor.h	470
10.28 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DeckManager.cpp File Reference	472
10.28.1 Detailed Description	473
10.29 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DeckManager.h File Reference	473
10.29.1 Detailed Description	473
10.30 DeckManager.h	474
10.31 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Dice.cpp File Reference	475
10.32 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Dice.h File Reference	475
10.32.1 Detailed Description	476
10.33 Dice.h	476
10.34 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.cpp File Reference	477
10.34.1 Detailed Description	477
10.35 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.h File Reference	478
10.35.1 Detailed Description	478
10.36 DiceEditor.h	478
10.37 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DisplayCard.cpp File Reference	480
10.37.1 Detailed Description	480
10.38 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DisplayCard.h File Reference	481
10.39 DisplayCard.h	481
10.40 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DropDownInput.cpp File Reference	484
10.40.1 Detailed Description	484
10.41 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DropDownInput.h File Reference	484
10.41.1 Detailed Description	484
10.42 DropDownInput.h	485
10.43 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DropDownMenu.cpp File Reference	485

10.43.1 Detailed Description	485
10.44 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DropDownMenu.h File Reference	486
10.44.1 Detailed Description	486
10.45 DropDownMenu.h	486
10.46 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/EditorWorkspace.cpp File Reference	488
10.46.1 Detailed Description	488
10.47 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/EditorWorkspace.h File Reference	488
10.47.1 Detailed Description	489
10.48 EditorWorkspace.h	489
10.49 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/ElementSelection.cpp File Reference	490
10.49.1 Detailed Description	490
10.50 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/ElementSelection.h File Reference	490
10.50.1 Detailed Description	491
10.51 ElementSelection.h	491
10.52 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/FileAccess.cpp File Reference	492
10.53 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/FileAccess.h File Reference	492
10.53.1 Detailed Description	492
10.54 FileAccess.h	493
10.55 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GameDataManager.cpp File Reference	493
10.55.1 Detailed Description	494
10.56 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GameDataManager.h File Reference	494
10.56.1 Detailed Description	494
10.57 GameDataManager.h	495
10.58 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GameEditor.cpp File Reference	496
10.58.1 Detailed Description	496
10.59 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GameEditor.h File Reference	496
10.59.1 Detailed Description	497
10.60 GameEditor.h	497
10.61 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GlobalVariables.cpp File Reference	499
10.61.1 Detailed Description	499
10.62 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/GlobalVariables.h File Reference	499
10.62.1 Detailed Description	499
10.63 GlobalVariables.h	500

10.64 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/HomeWorkSpace.cpp File Reference	501
10.65 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/HomeWorkSpace.h File Reference	501
10.65.1 Detailed Description	502
10.66 HomeWorkSpace.h	502
10.67 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Main.cpp File Reference	504
10.67.1 Detailed Description	504
10.67.2 Function Documentation	504
10.67.2.1 main()	504
10.68 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MapIdToElement.cpp File Reference	504
10.68.1 Detailed Description	504
10.69 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MapIdToElement.h File Reference	505
10.69.1 Detailed Description	505
10.70 MapIdToElement.h	505
10.71 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MenuBar.cpp File Reference	506
10.71.1 Detailed Description	506
10.72 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MenuBar.h File Reference	506
10.72.1 Detailed Description	507
10.73 MenuBar.h	507
10.74 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MouseHandler.cpp File Reference	508
10.74.1 Detailed Description	509
10.75 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MouseHandler.h File Reference	509
10.75.1 Detailed Description	509
10.76 MouseHandler.h	509
10.77 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Path.cpp File Reference	511
10.77.1 Function Documentation	511
10.77.1.1 operator==()	511
10.78 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Path.h File Reference	512
10.78.1 Function Documentation	512
10.78.1.1 operator==()	512
10.79 Path.h	513
10.80 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Player.cpp File Reference	515
10.80.1 Detailed Description	515
10.81 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Player.h File Reference	515
10.81.1 Detailed Description	516
10.82 Player.h	516
10.83 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/PlayerEditor.cpp File Reference	517

10.84 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/PlayerEditor.h File Reference	518
10.84.1 Detailed Description	518
10.85 PlayerEditor.h	518
10.86 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/PlayerManager.cpp File Reference	521
10.86.1 Detailed Description	521
10.87 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/PlayerManager.h File Reference	521
10.87.1 Detailed Description	521
10.88 PlayerManager.h	522
10.89 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/SpriteHandler.cpp File Reference	523
10.89.1 Detailed Description	523
10.90 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/SpriteHandler.h File Reference	524
10.90.1 Detailed Description	524
10.91 SpriteHandler.h	524
10.92 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/stdafx.cpp File Reference	525
10.92.1 Detailed Description	526
10.93 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/stdafx.h File Reference	526
10.93.1 Detailed Description	526
10.93.2 Macro Definition Documentation	526
10.93.2.1 WIN32_LEAN_AND_MEAN	527
10.94 stdafx.h	527
10.95 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.cpp File Reference	527
10.95.1 Detailed Description	527
10.96 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.h File Reference	527
10.96.1 Detailed Description	528
10.96.2 Macro Definition Documentation	528
10.96.2.1 DELETE_KEY	528
10.96.2.2 ENTER_KEY	528
10.96.2.3 ESCAPE_KEY	528
10.97 TextBox.h	529
10.98 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TileEditor.cpp File Reference	530
10.98.1 Detailed Description	530
10.99 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TileEditor.h File Reference	531
10.99.1 Detailed Description	531
10.100 TileEditor.h	531
10.101 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Tiles.cpp File Reference	534
10.101.1 Detailed Description	534
10.102 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Tiles.h File Reference	534
10.102.1 Detailed Description	535

10.103 Tiles.h	535
10.104 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UI_ID.cpp File Reference	536
10.104.1 Detailed Description	536
10.105 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UI_ID.h File Reference	536
10.105.1 Detailed Description	537
10.106 UI_ID.h	537
10.107 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UIElement.cpp File Reference	538
10.107.1 Detailed Description	538
10.108 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UIElement.h File Reference	539
10.108.1 Detailed Description	539
10.109 UIElement.h	540
10.110 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UtilityFunctions.cpp File Reference	542
10.111 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UtilityFunctions.h File Reference	542
10.112 UtilityFunctions.h	543
10.113 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/WorkSpace.cpp File Reference	543
10.113.1 Detailed Description	543
10.114 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/WorkSpace.h File Reference	543
10.114.1 Detailed Description	544
10.115 WorkSpace.h	544
Index	547

Chapter 1

Todo List

Class BoardEditor

allowd to enter number in the txt feild
allowd to enter number in the txt feild

Member Brood::Application::Application::Application ()

: delete me
delete me

Member Brood::Application::Application::Draw ()

delete me
delete me

Member Brood::Application::Application::myBoard

delete me

Member Brood::Application::Application::RunApplicaiton ()

delete me

Member Brood::Application::Components::Board::Draw (sf::RenderWindow &a_window)

add copy constuctor and copy assignment

Member Brood::Application::Components::Tiles::GetRowAndCol () const

add copy constructor and assignment

Member Brood::Application::HomeWorkSpace::UpdateRibbonTabs ()

deleteME
deleteme

Member Brood::SpriteHandler::SetSpriteFromTexture (unsigned a_num)

make the adaptable for number of rows and column

Class Player

make it so that if two or more player are in the same tile then offset the player so that they are visually. One way to do this could be to add a vector of playerPtr to tile where if a player get to the tile player will push itself back to this list and will save ther player index to itself. and once a player moves on it will call a function to update itself leaving the tile thus updating other player position in the tile list

Class SpriteHandler

make this animatable. next Sprit() which get the next sprite and loops back to idx 0

Class TextBox

add a way to add place holder text
add view to the textbox feild that gets scrolled when the type text go of window
add a actual cursor instead of _

Class [TileEditor](#)

allowd to enter number in the txt feild

Class [UIElement](#)

create a verticalmenus for ma [UIElement](#). This could be used to for Drop down menus.

create a font class whose job is to load and save font and replace all the m_font to object of that file

Encapsulate the BroodUI better.

create a log class that logs all the errors

change the setter funciton to return bool, errorMessage

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

Brood	17
Application	17
Data	18
Components	19
BroodUI	19

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Brood	21
Brood::Application	
TODO initialize all of the data here to match the default values	21
Brood::Application::Components	22
Brood::Application::Data	23
Brood::Application::StaticVariables	23
Brood::BroodUI	24
Brood::UtilityFuncs	
It contains all the utility functions for which making a stand alone class would not make sense	25

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Brood::Application::Application	27
Board	31
Brood::Application::Components::Board	31
BoardEditor	43
Button	59
CardEditor	127
Brood::Application::Components::CardInfo	128
Brood::Application::Components::DeckManager	154
Dice	167
DiceEditor	179
DropDownInput	197
DropDownMenu	208
Brood::BroodUI::ElementSelection	214
ElementSelection	218
Brood::Application::FileAccess	219
Brood::Application::Components::GameDataManager	222
GameDataManager	228
HomeWorkSpace	247
Brood::BroodUI::Id	247
Id	255
Brood::BroodUI::MapIdToElement	255
MapIdToElement	258
MenuBar	268
Brood::MouseHandler	270
Brood::Application::Components::Path	277
Path	290
Brood::Application::Components::Player	290
Player	298
PlayerEditor	312
Brood::Application::Components::PlayerManager	312
PlayerManager	321
Brood::SpriteHandler	321
SpriteHandler	329
ST_ApplicationData	329
Brood::Application::Data::ST_BoardData	330

ST_BoardData	332
St_CardInfo	333
Brood::Application::Data::ST_CardInfoPrefabData	333
ST_CardPrefabData	335
Brood::Application::StaticVariables::ST_ColorVariables	335
ST_ColorVariables	338
Brood::Application::Data::ST_DeckManagerData	338
ST_DeckManagerData	340
Brood::Application::Data::ST_DeckPrefabData	340
ST_DeckPrefabData	343
Brood::Application::Data::ST_DicePrefabData	344
ST_DicePrefabData	346
Brood::Application::Data::ST_DisplayCardData	346
ST_DisplayCardData	360
Brood::Application::StaticVariables::ST_Folders	360
Brood::Application::Data::ST_GameData	363
Brood::Application::StaticVariables::ST_GlobalCoreVariables	364
Brood::Application::Data::ST_PathPrefabData	366
Brood::Application::Data::ST_PlayerManagerData	369
ST_PlayerManagerData	371
Brood::Application::Data::ST_PlayerPrefabData	371
ST_PlayerPrefabData	374
ST_TilePrefabData	374
TextBox	384
TileEditor	400
Tiles	405
Brood::BroodUI::UIElement	406
Brood::BroodUI::Button	54
Brood::Application::Components::Deck	134
Brood::Application::Components::Dice	161
Brood::Application::Components::DisplayCard	179
Brood::Application::Components::Tiles	401
Brood::BroodUI::DropDownMenu	198
Brood::BroodUI::DropDownInput	194
Brood::BroodUI::MenuBar	258
Brood::BroodUI::TextBox	375
UIElement	421
Brood::Application::WorkSpace	422
Brood::Application::BoardEditor	44
Brood::Application::CardEditor	60
Brood::Application::DeckEditor	142
Brood::Application::DiceEditor	168
Brood::Application::EditorWorkspace	209
Brood::Application::GameEditor	229
Brood::Application::HomeWorkSpace	237
Brood::Application::PlayerEditor	299
Brood::Application::TileEditor	386
WorkSpace	436

Chapter 5

Class Index

5.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Brood::Application::Application	27
Applicaiton class provides the interface for the applicaiton	
Board	31
A container Class to create a board	
Brood::Application::Components::Board	31
BoardEditor	
It allows the users to tweek the general game setting and game rules	43
Brood::Application::BoardEditor	44
Brood::BroodUI::Button	54
Button	
Uielemnt to work as a Button	59
Brood::Application::CardEditor	60
CardEditor	
It allows the users to tweek the card setting	127
Brood::Application::Components::CardInfo	128
Brood::Application::Components::Deck	134
Brood::Application::DeckEditor	142
Brood::Application::Components::DeckManager	154
Brood::Application::Components::Dice	161
Dice	
A class to create a Deck	167
Brood::Application::DiceEditor	168
DiceEditor	
It allows the users to tweek the dice setting	179
Brood::Application::Components::DisplayCard	179
Brood::BroodUI::DropDownInput	194
DropDownInput	
A UI elemnt that represent a drop down menu	197
Brood::BroodUI::DropDownMenu	198
DropDownMenu	
A UI elemnt that represent a drop down menu	208
Brood::Application::EditorWorkspace	209
Brood::BroodUI::ElementSelection	214
ElementSelection	
A static class to store which element is selected and what could be selected next	218

Brood::Application::FileAccess	219
Brood::Application::Components::GameDataManager	222
GameDataManager	
A GameDataManager Class manages the game data	228
Brood::Application::GameEditor	229
Brood::Application::HomeWorkSpace	237
HomeWorkSpace	
It is the underlying entry to the application. It is the true entry point for the application	247
Brood::BroodUI::Id	247
Id	
Class to provide a unique ID to the different UI elements	255
Brood::BroodUI::MapIdToElement	255
MapIdToElement	
A static struct to handel the mapping of unique UI_ID to its element for quick acceess	258
Brood::BroodUI::MenuBar	258
MenuBar	
A class that represents a menu bar	268
Brood::MouseHandler	270
Brood::Application::Components::Path	277
Path	
Path is a wraper for tile class	290
Brood::Application::Components::Player	290
Player	
A Player Class	298
Brood::Application::PlayerEditor	299
PlayerEditor	
It allows the users to tweek the player setting	312
Brood::Application::Components::PlayerManager	312
PlayerManager	
A PlayerManager Class manages all player realted things	321
Brood::SpriteHandler	321
SpriteHandler	
Class to handel the sprite	329
ST_ApplicationData	
All the general data about the game	329
Brood::Application::Data::ST_BoardData	330
ST_BoardData	
Data about the board	332
St_CardInfo	
A class to store the information that is to be displayed in the card	333
Brood::Application::Data::ST_CardInfoPrefabData	333
ST_CardPrefabData	
Data in a card and other use full data	335
Brood::Application::StaticVariables::ST_ColorVariables	335
ST_ColorVariables	
Static struct for all the colors used in the applicaiton	338
Brood::Application::Data::ST_DeckManagerData	338
ST_DeckManagerData	
Deck manager data for one deck prefab	340
Brood::Application::Data::ST_DeckPrefabData	340
ST_DeckPrefabData	
Deck data for one deck prefab	343
Brood::Application::Data::ST_DicePrefabData	344
ST_DicePrefabData	
Dice prefab data	346
Brood::Application::Data::ST_DisplayCardData	346
ST_DisplayCardData	
Display card data	360

Brood::Application::StaticVariables::ST_Folders	360
Brood::Application::Data::ST_GameData	363
Brood::Application::StaticVariables::ST_GlobalCoreVariables	364
Brood::Application::Data::ST_PathPrefabData	366
Brood::Application::Data::ST_PlayerManagerData	369
ST_PlayerManagerData	
Player manager data	371
Brood::Application::Data::ST_PlayerPrefabData	371
ST_PlayerPrefabData	
Player prefab data	374
ST_TilePrefabData	
Tile data for one tile prefab	374
Brood::BroodUI::TextBox	375
TextBox	
Uielemnt that represent the text in the UI	384
Brood::Application::TileEditor	386
TileEditor	
It allows the users to tweek the deck setting	400
Brood::Application::Components::Tiles	401
Tiles	
Building block of a borad	405
Brood::BroodUI::UIElement	406
UIElement	
Abstract base class that is parent to all the different UI elements	421
Brood::Application::WorkSpace	422
WorkSpace	
Abstract base class that is parent to all the different workspace	436

Chapter 6

File Index

6.1 File List

Here is a list of all files with brief descriptions:

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Application.cpp	437
This file is a source file for Application class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Application.h	437
This file is a header file for Application class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Board.cpp	439
This file is a source file for Board class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Board.h	439
This file is a header file for Board class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.cpp	442
This file is a source file for BoardEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h	442
This file is a header file for BoardEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Button.cpp	445
This file is a source file for Button class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Button.h	445
This file is a header file for Button class which is in BroodUI namespace	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/CardEditor.cpp	447
This file is a source file for CardEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/CardEditor.h	447
This file is a header file for CardEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/CardInfo.cpp	458
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/CardInfo.h	458
This file is a header file for Dice class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Data.cpp	460
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Data.h	460
This file is a header file for all the Data that the application uses	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Deck.cpp	467
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Deck.h	467
This file is a source file for Deck class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DeckEditor.cpp	469
This file is a source file for DeckEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DeckEditor.h	470
This file is a header file for DeckEditor class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DeckManager.cpp	472
This file is a source file for DeckManager class	

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DeckManager.h	473
This file is a header file for DeckManager class	
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Dice.cpp	475
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Dice.h	
This file is a source file for Dice class	475
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DiceEditor.cpp	
This file is a source file for DiceEditor class	477
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DiceEditor.h	
This file is a header file for DiceEditor class	478
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DisplayCard.cpp	
This file is a source file for DisplayCard class	480
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DisplayCard.h	481
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DropDownInput.cpp	
This file is a source file for DropDownInput class	484
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DropDownInput.h	
This file is a header file for DropDownInput class	484
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DropDownMenu.cpp	
This file is a source file for DropDownMenu class	485
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/DropDownMenu.h	
This file is a header file for DropDownMenu class	486
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/EditorWorkspace.cpp	
This file is a source file for EditorWorkspace class	488
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/EditorWorkspace.h	
This file is a header file for InitialWorkSpace class	488
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/ElementSelection.cpp	
This file is a source file for ElementSelection class which is in BroodUI namespace	490
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/ElementSelection.h	
This file is a header file for ElementSelection class	490
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/FileAccess.cpp	492
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/FileAccess.h	
This file is a header file for FileAccess class	492
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GameDataManager.cpp	
This file is a source file for GameDataManager class	493
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GameDataManager.h	
This file is a header file for GameDataManager class	494
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GameEditor.cpp	
This file is a source file for GameEditor class	496
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GameEditor.h	
This file is a header file for GameEditor class	496
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GlobalVariables.cpp	
This file is a source file for GlobalVariables stuct and ColorVariable struct	499
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/GlobalVariables.h	
This file is a header file for color variables	499
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/HomeWorkSpace.cpp	501
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/HomeWorkSpace.h	
This file is a header file for InitialWorkSpace class	501
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Main.cpp	
This file is conatains the main function	504
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MapIdToElement.cpp	
This file is a source file for MapIdToElement class	504
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MapIdToElement.h	
This file is a header file for MapIdToElement which are declared in BroodUI namespace	505
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MenuBar.cpp	
This file is a source file for MenuBar class	506
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MenuBar.h	
This file is a header file for MenuBar class	506
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MouseHandler.cpp	
This file is a source file for MouseHandler class	508

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/MouseHandler.h This file is a header file for MouseHandler class	509
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Path.cpp C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Path.h C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Player.cpp This file is a source file for Player class	511
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Player.h This file is a header file for Player class	512
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/PlayerEditor.cpp C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/PlayerEditor.h This file is a header file for PlayerEditor class	515
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/PlayerManager.cpp This file is a source file for PlayerManager class	517
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/PlayerManager.h This file is a header file for PlayerManager class	518
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/SpriteHandler.cpp This file is a source file for SpriteHandler class	521
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/SpriteHandler.h This file is a header file for SpriteHandler class which is in Brood namespace	523
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/stdafx.cpp This file is a source file that includes the standard includes	524
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/stdafx.h This file is a include file for include files which are not changed infrequently	525
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/TextBox.cpp This file is a source file for TextBox class	526
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/TextBox.h This file is a header file for TextBox class whicih is in BroodUI namespace	527
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/TileEditor.cpp This file is a source file for TileEditor class	528
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/TileEditor.h This file is a header file for TileEditor class	530
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Tiles.cpp This file is a source file for Tiles class	531
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Tiles.h This file is a header file for Tiles class	534
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UI_ID.cpp This file is a source file for Id class which is in BroodUI namespace	536
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UI_ID.h This file is a header file for Id class which is in BroodUI namespace	536
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UIElement.cpp This file is a source file for UIElement class and struct MapIdToElement which is in BroodUI namespace	538
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UIElement.h This file is a header file for UIElement class and it also contains a enum ENUM_UIType, struct ST_UIElementCtorParam, and struct MapIdToElement which are declared in BroodUI namespace	539
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UtilityFunctions.cpp C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/UtilityFunctions.h C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/WorkSpace.cpp This file is a source file for WorkSpace class	542
C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/WorkSpace.h This file is a header file for WorkSpace class	543

Chapter 7

Module Documentation

7.1 Brood

Modules

- [Application](#)
- [BroodUI](#)

Namespaces

- namespace [Brood::UtilityFuncs](#)

It contains all the utility functions for which making a stand alone class would not make sense.

Classes

- class **MouseHandler**
A class to handle all mouse changes.
- class **SpriteHandler**
a class to handle the sprite

7.1.1 Detailed Description

7.2 Application

Modules

- [Data](#)
- [Components](#)

Classes

- class [Brood::Application::Application](#)
Applicaiton class provides the interface for the applicaiton.
- class [BoardEditor](#)
It allows the users to tweek the general game setting and game rules.
- class [CardEditor](#)
It allows the users to tweek the card setting.
- class [TileEditor](#)
It allows the users to tweek the deck setting.
- class [DiceEditor](#)
It allows the users to tweek the dice setting.
- class [HomeWorkSpace](#)
It is the underlying entry to the application. It is the true entry point for the application.
- class [PlayerEditor](#)
It allows the users to tweek the player setting.
- class [WorkSpace](#)
a abstract base class that is parent to all the different workspace.

7.2.1 Detailed Description

7.3 Data

Classes

- struct [ST_ApplicationData](#)
contains all the general data about the game
- struct [ST_BoardData](#)
contains data about the board
- struct [ST_CardPrefabData](#)
data in a card and other use full data
- struct [ST_DeckPrefabData](#)
contains deck data for one deck prefab
- struct [ST_DeckManagerData](#)
contains deck manager data for one deck prefab
- struct [ST_DicePrefabData](#)
contains dice prefab data
- struct [ST_DisplayCardData](#)
contains display card data
- struct [ST_PlayerPrefabData](#)
contains player prefab data
- struct [ST_PlayerManagerData](#)
contains player manager data
- struct [ST_TilePrefabData](#)
contains tile data for one tile prefab

7.3.1 Detailed Description

7.4 Components

Classes

- class [Board](#)
A container Class to create a board.
- class [PlayerManager](#)
A [PlayerManager](#) Class manages all player realted things.
- class [Dice](#)
A class to create a Deck.
- class [GameDataManager](#)
A [GameDataManager](#) Class manages the game data.
- class [Path](#)
[Path](#) is a wraper for tile class.
- class [Player](#)
A [Player](#) Class.
- class [Tiles](#)
building block of a borad.

7.4.1 Detailed Description

7.5 BroodUI

Classes

- class [Button](#)
a uielment to work as a [Button](#).
- class [DropDownInput](#)
A UI elemnt that represent a drop down menu.
- class [DropDownMenu](#)
A UI elemnt that represent a drop down menu.
- class [ElementSelection](#)
A static class to store which element is selected and what could be selected next.
- class [MapIdToElement](#)
A static struct to handel the mapping of unique UI_ID to its element for quick acceess.
- class [MenuBar](#)
A class that represents a menu bar.
- class [TextBox](#)
a uielment that represent the text in the UI
- class [Id](#)
Class to provide a unique ID to the different UI elements.
- class [UIElement](#)
a abstract base class that is parent to all the different UI elements

7.5.1 Detailed Description

Chapter 8

Namespace Documentation

8.1 Brood Namespace Reference

Namespaces

- namespace [Application](#)

TODO initialize all of the data here to match the default values.
- namespace [BroodUI](#)
- namespace [UtilityFuncs](#)

It contains all the utility functions for which making a stand alone class would not make sense.

Classes

- class [MouseHandler](#)
- class [SpriteHandler](#)

8.2 Brood::Application Namespace Reference

TODO initialize all of the data here to match the default values.

Namespaces

- namespace [Components](#)
- namespace [Data](#)
- namespace [StaticVariables](#)

Classes

- class [Application](#)
Applicaiton class provides the interface for the applicaiton.
- class [BoardEditor](#)
- class [CardEditor](#)
- class [DeckEditor](#)
- class [DiceEditor](#)
- class [EditorWorkspace](#)
- class [FileAccess](#)
- class [GameEditor](#)
- class [HomeWorkSpace](#)
- class [PlayerEditor](#)
- class [TileEditor](#)
- class [WorkSpace](#)

8.2.1 Detailed Description

TODO initialize all of the data here to match the default values.

@TODO make a better image handler @TODO add a image to [ST_TilePrefabData](#), [ST_DeckPrefabData](#), [ST_PlayerPrefabData](#), [ST_DicePrefabData](#), and [ST_CardPrefabData](#) @TODO create a system data that adds fonts, colors, size, screen height/length and, current game status other things that the applicaiton might use here
 Basically have all the static system variables here...

8.3 Brood::Application::Components Namespace Reference

Classes

- class [Board](#)
- class [CardInfo](#)
- class [Deck](#)
- class [DeckManager](#)
- class [Dice](#)
- class [DisplayCard](#)
- class [GameDataManager](#)
- class [Path](#)
- class [Player](#)
- class [PlayerManager](#)
- class [Tiles](#)

Enumerations

- enum class [ENUM_MovementType](#) { [MOVEMENT_diceThenCard](#) , [MOVEMENT_diceOnly](#) , [MOVEMENT_cardOnly](#) }
- enum class [ENUM_TileType](#) {
[TILE_blank](#) , [TILE_tile](#) , [TILE_start](#) , [TILE_end](#) ,
[TILE_bridge](#) }

8.3.1 Enumeration Type Documentation

8.3.1.1 ENUM_MovementType

```
enum class Brood::Application::Components::ENUM\_MovementType [strong]
```

Enumerator

MOVEMENT_diceThenCard	
MOVEMENT_diceOnly	when the dice if forced for on the start
MOVEMENT_cardOnly	when the dice is only used

8.3.1.2 ENUM_TileType

```
enum class Brood::Application::Components::ENUM_TileType [strong]
```

Enumerator

TILE_blank	
TILE_tile	
TILE_start	
TILE_end	
TILE_bridge	

8.4 Brood::Application::Data Namespace Reference**Classes**

- struct [ST_BoardData](#)
- struct [ST_CardInfoPrefabData](#)
- struct [ST_DeckManagerData](#)
- struct [ST_DeckPrefabData](#)
- struct [ST_DicePrefabData](#)
- struct [ST_DisplayCardData](#)
- struct [ST_GameData](#)
- struct [ST_PathPrefabData](#)
- struct [ST_PlayerManagerData](#)
- struct [ST_PlayerPrefabData](#)

8.5 Brood::Application::StaticVariables Namespace Reference**Classes**

- struct [ST_ColorVariables](#)
- struct [ST_Folders](#)
- struct [ST_GlobalCoreVariables](#)

8.6 Brood::BroodUI Namespace Reference

Classes

- class [Button](#)
- class [DropDownInput](#)
- class [DropDownMenu](#)
- class [ElementSelection](#)
- class [Id](#)
- class [MapIdToElement](#)
- class [MenuBar](#)
- class [TextBox](#)
- class [UIElement](#)

Enumerations

- enum class [ENUM_UIType](#) {

 [UI_textBox](#) , [UI_button](#) , [UI_dropDownMenu](#) , [UI_dropDownInput](#) ,

 [UI_menuBar](#) , [UI_scrollBar](#) }

8.6.1 Enumeration Type Documentation

8.6.1.1 ENUM_UIType

```
enum class Brood::BroodUI::ENUM\_UIType [strong]
```

Enumerator

UI_textBox	
UI_button	container for text -> derived from UI_Element
UI_dropDownMenu	a button -> derived from UI_Element
UI_dropDownInput	drop down menu; has items inside it -> derived from UI_Button
UI_menuBar	drop down input; has items inside it -> derived from UI_Button
UI_scrollBar	menubar is a list of drop down menu stacked horizontally @TODO add latter

8.7 Brood::UtilityFuncs Namespace Reference

It contains all the utility functions for which making a stand alone class would not make sense.

Functions

- bool [LoadTextureFromFile](#) (sf::Texture &a_texture, const std::string &a_cwd, std::string a_filePath)
It opens a texture form a file and loads it into the texture.
- bool [LoadTextureFromFile](#) (sf::Texture &a_texture, std::string a_filePath)
It opens a texture form a file and loads it into the texture.

8.7.1 Detailed Description

It contains all the utility functions for which making a stand alone class would not make sense.

8.7.2 Function Documentation

8.7.2.1 LoadTextureFromFile() [1/2]

```
bool Brood::UtilityFuncs::LoadTextureFromFile (
    sf::Texture & a_texture,
    const std::string & a_cwd,
    std::string a_filePath )
```

It opens a texture form a file and loads it into the texture.

@overload

It also log to console if texture loading was unsuccesful

Parameters

a_texture	reference to the texture object where the loaded texture is to be stored
a_cwd	current working directory
a_filePath	file path that points to the texture to be loaded

Returns

true if the texture was loaded successfully; else returns false

8.7.2.2 LoadTextureFromFile() [2/2]

```
bool Brood::UtilityFuncs::LoadTextureFromFile (
    sf::Texture & a_texture,
    std::string a_filePath )
```

It opens a texture from a file and loads it into the texture.

It also log to console if texture loading was unsuccessful

Parameters

<i>a_texture</i>	reference to the texture object where the loaded texture is to be stored
<i>a_filePath</i>	file path that points to the texture to be loaded

Returns

true if the texture was loaded successfully; else returns false

Chapter 9

Class Documentation

9.1 Brood::Application::Application Class Reference

Applicaiton class provides the interface for the applicaiton.

```
#include <Application.h>
```

Public Member Functions

- [Application \(\)](#)
default constructor
- [~Application \(\)](#)
default destructor
- [void RunApplicaiton \(\)](#)
Contains the main loop for the applicaiton.

Private Member Functions

- [void PollEvents \(\)](#)
polls for any event
- [void Draw \(\)](#)
draws to screen
- [void CreateWorkSpace \(\)](#)
- [void Debugger \(\)](#)
calls the UI Element Debugger

Private Attributes

- [sf::RenderWindow m_window](#)
window obj
- [sf::Event m_events](#)
event object
- [Brood::Application::HomeWorkSpace m_mainWorkspace](#)
initial workspace object
- [Brood::Application::Components::Board myBoard](#)
- [Brood::Application::Components::Dice myDice](#)

9.1.1 Detailed Description

Applicaiton class provides the interface for the applicaiton.

9.1.2 Constructor & Destructor Documentation

9.1.2.1 Application()

```
Brood::Application::Application::Application ( )
```

default constructor

Inializes the application and sets it up

Todo : delete me

Todo delete me

9.1.2.2 ~Application()

```
Brood::Application::Application::~Application ( )
```

default destructor

cleans up after the applicaiton

9.1.3 Member Function Documentation

9.1.3.1 CreateWorkSpace()

```
void Brood::Application::Application::CreateWorkSpace ( ) [private]
```

9.1.3.2 Debugger()

```
void Brood::Application::Application::Debugger () [private]
```

calls the UI Element Debugger

changes the text of the element to its the element ID

This function helps in debugging the UI elements. Called once at the end of Initialization.

9.1.3.3 Draw()

```
void Brood::Application::Application::Draw () [private]
```

draws to screen

Draws object to screen.

@protecteed

Note

the elements should be drawn form bottom to top, right to left.

See also

[Brood::BroodUI::UIElement](#) class for more explation.

Todo delete me

Todo delete me

9.1.3.4 PollEvents()

```
void Brood::Application::Application::PollEvents () [private]
```

polls for any event

This function polls the event object for any event.

@protecteed

9.1.3.5 RunApplicaiton()

```
void Brood::Application::Application::RunApplicaiton ( )
```

Contains the main loop for the applicaiton.

Todo delete me

9.1.4 Member Data Documentation

9.1.4.1 m_events

```
sf::Event Brood::Application::Application::m_events [private]
```

event object

9.1.4.2 m_mainWorkspace

```
Brood::Application::HomeWorkSpace Brood::Application::Application::m_mainWorkspace [private]
```

initial workspace object

9.1.4.3 m_window

```
sf::RenderWindow Brood::Application::Application::m_window [private]
```

window obj

9.1.4.4 myBoard

```
Brood::Application::Components::Board Brood::Application::Application::myBoard [private]
```

Todo delete me

9.1.4.5 myDice

```
Brood::Application::Components::Dice Brood::Application::myDice [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Application.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Application.cpp](#)

9.2 Board Class Reference

A container Class to create a board.

```
#include "Board.h"
```

9.2.1 Detailed Description

A container Class to create a board.

Note

that board origin is set in the upper left corner; which will be consistent through out the application

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Board.h](#)

9.3 Brood::Application::Components::Board Class Reference

```
#include <Board.h>
```

Public Member Functions

- **Board** (unsigned a_numRows=1, unsigned a_numCols=1, float a_boardSizeX=50.f, float a_boardSizeY=50.f, float a_boardPosX=50.f, float a_boardPosY=50.f)
default Constructor
- **~Board** ()
Destructor.
- void **Draw** (sf::RenderWindow &a_window)
Draw function draws the board to the screen.
- void **InitializeBoard** (unsigned a_numRows=0, unsigned a_numCols=0, float a_boardSizeX=0.f, float a_boardSizeY=0.f, float a_boardPosX=0.f, float a_boardPosY=0.f)
sets up the board
- void **InitializeBoard** (Brood::Application::Data::ST_BoardData &a_boardData)
sets up the board
- **Brood::Application::Data::ST_BoardData GetDataToSave** ()

- creates and returns boardData struct
- void [SaveDataToFile](#) (Brood::Application::FileAccess *a_fileAccessPtr)

saves the board and its path data to passed file
- void [LoadDataFromFile](#) (Brood::Application::FileAccess *a_fileAccessPtr, Brood::Application::Components::DeckManager *a_deckMangerPtr)

loads the board and its path data from passed file
- void [SetNumRow](#) (unsigned a_numRows, Brood::Application::Components::Deck *a_deckPtr)

Setter funciton to set the number of row in the board.
- void [SetNumCol](#) (unsigned a_numCols, Brood::Application::Components::Deck *a_deckPtr)

Setter funciton to set the number of col in the board.
- void [SetBoardSize](#) (sf::Vector2f a_boardSize)

Setter funciton to set the board size.
- void [SetBoardSize](#) (float a_boardSizeX, float a_boardSizeY)

Setter funciton to set the board size.
- void [SetBoardPos](#) (sf::Vector2f a_boardPos)

Setter funciton to set the board position.
- void [SetBoardPos](#) (float a_boardPosX, float a_boardPosY)

Setter funciton to set the board position.
- void [SetCurrentActiveTilePtr](#) (Brood::Application::Components::Path *a_newActivePathPtr)

Setter funciton to set the current active path.
- const unsigned [GetNumRow](#) () const

Getter funciton to get the number of rows in the board.
- const unsigned [GetNumCol](#) () const

Getter funciton to get the number of columns in the board.
- const sf::Vector2f [GetBoardSize](#) () const

Getter funciton to get the board size.
- const sf::Vector2f [GetBoardPos](#) () const

Getter funciton to get the board position.
- const std::vector< std::vector< Brood::Application::Components::Path * > > & [GetBoardPathList](#) () const

Getter funciton to get the board path list by reference.
- Brood::Application::Components::Path * [GetCurrentActivePath](#) ()

Getter funciton to get the current active path pointer.
- void [ToggleDrawLine](#) ()

Toggles draw line variable for all the path.
- void [Debugger](#) ()

Draw funciton.

Private Member Functions

- void [IncreaseNumRow](#) (unsigned a_numRows, Brood::Application::Components::Deck *a_deckPtr)

Increases and populates the number of row in the board and updates the old tiles.
- void [DecreaseNumRow](#) (unsigned a_numRows)

Decreases the number of row in the board and updates the old tiles and deallocates the removed row.
- void [IncreaseNumCol](#) (unsigned a_numCols, Brood::Application::Components::Deck *a_deckPtr)

Increases and populates the number of columns in the board and updates the old tiles.
- void [DecreaseNumCol](#) (unsigned a_numCols)

Decreases the number of columns in the board and updates the old tiles and deallocates the removed columns.
- void [UpdateBoardPath](#) (unsigned a_rowBegin, unsigned a_rowEnd, unsigned a_colBegin, unsigned a_colEnd, bool a_createNew=false, Brood::Application::Components::Deck *a_deckPtr=nullptr)

updates the path specified by passed param in board by using the member variables.

Private Attributes

- sf::RectangleShape [m_boardBody](#)
main board
- unsigned [m_numRows](#)
number of cell rows in the board
- unsigned [m_numCols](#)
number of cell columns in the board
- std::vector< std::vector< [Brood::Application::Components::Path](#) * > > [m_boardPaths](#)
contains the board [Path](#) whic in turn contains tile
- [Brood::Application::Components::Path](#) * [m_currActivePathPtr](#)

9.3.1 Constructor & Destructor Documentation

9.3.1.1 Board()

```
Board::Board (
    unsigned a numRows = 1,
    unsigned a numCols = 1,
    float a boardSizeX = 50.f,
    float a boardSizeY = 50.f,
    float a boardPosX = 50.f,
    float a boardPosY = 50.f )
```

default Constructor

It initializes the board as well as the tiles. [Tiles](#) are set up according to the

Parameters

a numRows	number of tile rows in a board -> default 1
a numCols	number of tile columns in a board-> default 1
a boardSizeX	board's width -> default 50.f
a boardSizeY	board's length -> default 50.f
a boardPosX	board's x-position on screen; relative to the render window -> default 50.f
a boardPosY	board's y-position on screen; realtive to the render window -> default 50.f

9.3.1.2 ~Board()

```
Board::~Board ( )
```

Destructor.

9.3.2 Member Function Documentation

9.3.2.1 Debugger()

```
void Board::Debugger ( )
```

Draw funciton.

@virtual

This function helps in debugging the UI elements. Called once at the end of Inialization.

9.3.2.2 DecreaseNumCol()

```
void Board::DecreaseNumCol (
    unsigned a_numCols ) [private]
```

Decreases the number of columns in the board and updates the old tiles and deallocates the removed columns.

Parameters

<i>a_numCols</i>	number of tile columns to set in a board
------------------	--

9.3.2.3 DecreaseNumRow()

```
void Board::DecreaseNumRow (
    unsigned a_numRows ) [private]
```

Decreases the number of row in the board and updates the old tiles and deallocate the removed row.

Parameters

<i>a_numRows</i>	number of tile rows to set in a board
------------------	---------------------------------------

9.3.2.4 Draw()

```
void Board::Draw (
    sf::RenderWindow & a_window )
```

Draw funciton draws the board to the screen.

Todo add copy constructor and copy assignment

It also invokes the tiles', dice's, and playermanger's draw functions.

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.3.2.5 GetBoardPathList()

```
const std::vector< std::vector< Brood::Application::Components::Path * > > & Board::Get<BoardPathList> ( ) const
```

Getter funciton to get the board path list by reference.

Returns

board path list by reference

9.3.2.6 GetBoardPos()

```
const sf::Vector2f Board::GetBoardPos ( ) const
```

Getter funciton to get the board position.

Returns

board position

9.3.2.7 GetBoardSize()

```
const sf::Vector2f Board::GetBoardSize ( ) const
```

Getter funciton to get the board size.

Returns

board size

9.3.2.8 GetCurrentActivePath()

```
Brood::Application::Components::Path * Board::GetCurrentActivePath ( )
```

Getter funciton to get the current active path pointer.

Returns

pointer to the current active path

9.3.2.9 GetDataToSave()

```
Brood::Application::Data::ST_BoardData Board::GetDataToSave ( )
```

creates and returns boardData struct

Returns

boardData struct with the board data in it

9.3.2.10 GetNumCol()

```
const unsigned Board::GetNumCol ( ) const
```

Getter funciton to get the number of columns in the board.

Returns

number of columns in the board

9.3.2.11 GetNumRow()

```
const unsigned Board::GetNumRow ( ) const
```

Getter funciton to get the number of rows in the board.

Returns

number of rows in the board

9.3.2.12 IncreaseNumCol()

```
void Board::IncreaseNumCol (  
    unsigned a_numCols,  
    Brood::Application::Components::Deck * a_deckPtr ) [private]
```

Increases and populates the number of columns in the board and updates the old tiles.

Parameters

<code>a_numCols</code>	number of tile cols to set in a board
------------------------	---------------------------------------

9.3.2.13 IncreaseNumRow()

```
void Board::IncreaseNumRow (
    unsigned a_numRows,
    Brood::Application::Components::Deck * a_deckPtr ) [private]
```

Increases and populates the number of row in the board and updates the old tiles.

Parameters

<code>a_numRows</code>	number of tile rows to set in a board
------------------------	---------------------------------------

9.3.2.14 InitializeBoard() [1/2]

```
void Board::InitializeBoard (
    Brood::Application::Data::ST_BoardData & a_boardData )
```

sets up the board

It initializes the board as well as the tiles.

Parameters

<code>a_boardData</code>	struct containing all the board data
--------------------------	--------------------------------------

9.3.2.15 InitializeBoard() [2/2]

```
void Board::InitializeBoard (
    unsigned a numRows = 0,
    unsigned a numCols = 0,
    float a boardSizeX = 0.f,
    float a boardSizeY = 0.f,
    float a boardPosX = 0.f,
    float a boardPosY = 0.f )
```

sets up the board

It initializes the board as well as the tiles.

Parameters

<i>a_numRows</i>	number of tile rows in a board -> default 0
<i>a_numCols</i>	number of tile columns in a board-> default 0
<i>a_boardSizeX</i>	board's width -> default 0.f
<i>a_boardSizeY</i>	board's length -> default 0.f
<i>a_boardPosX</i>	board's x-position on screen; relative to the render window -> default 0.f
<i>a_boardPosY</i>	board's y-position on screen; realtive to the render window -> default 0.f

assigning the path color

9.3.2.16 LoadDataFromFile()

```
void Board::LoadDataFromFile (
    Brood::Application::FileAccess * a_fileAccessPtr,
    Brood::Application::Components::DeckManager * a_deckMangerPtr )
```

loads the board and its path data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.3.2.17 SaveDataToFile()

```
void Board::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

saves the board and its path data to passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.3.2.18 SetBoardPos() [1/2]

```
void Board::SetBoardPos (
    float a_boardPosX,
    float a_boardPosY )
```

Setter funciton to set the board position.

It also updates the tiles by moving it

Parameters

<i>a_boardPosX</i>	board's width
<i>a_boardPosY</i>	board's length

9.3.2.19 SetBoardPos() [2/2]

```
void Board::SetBoardPos (
    sf::Vector2f a_boardPos )
```

Setter function to set the board position.

It also updates the tiles by moving it

Parameters

<i>a_boardPos</i>	position of the board
-------------------	-----------------------

9.3.2.20 SetBoardSize() [1/2]

```
void Board::SetBoardSize (
    float a_boardSizeX,
    float a_boardSizeY )
```

Setter function to set the board size.

It also updates the tiles by decreasing the size to keep the number of rows and number of columns constant

Parameters

<i>a_boardSizeX</i>	board's width
<i>a_boardSizeY</i>	board's length

9.3.2.21 SetBoardSize() [2/2]

```
void Board::SetBoardSize (
    sf::Vector2f a_boardSize )
```

Setter function to set the board size.

It also updates the tiles by decreasing the size to keep the number of rows and number of columns constant

Parameters

<i>a_boardSize</i>	size of the board
--------------------	-------------------

9.3.2.22 SetCurrentActiveTilePtr()

```
void Board::SetCurrentActiveTilePtr (
    Brood::Application::Components::Path * a_newActivePathPtr )
```

Setter funciton to set the current active path.

Parameters

<i>a_newActiveTilePtr</i>	pointer to the new active path
---------------------------	--------------------------------

9.3.2.23 SetNumCol()

```
void Board::SetNumCol (
    unsigned a_numCols,
    Brood::Application::Components::Deck * a_deckPtr )
```

Setter funciton to set the number of col in the board.

Parameters

<i>a_numCols</i>	number of tile col to set in a board
------------------	--------------------------------------

9.3.2.24 SetNumRow()

```
void Board::SetNumRow (
    unsigned a_numRows,
    Brood::Application::Components::Deck * a_deckPtr )
```

Setter funciton to set the number of row in the board.

Parameters

<i>a_numRows</i>	number of tile rows to set in a board
------------------	---------------------------------------

9.3.2.25 ToggleDrawLine()

```
void Board::ToggleDrawLine( )
```

Toggles draw line variable for all the path.

9.3.2.26 UpdateBoardPath()

```
void Board::UpdateBoardPath(
    unsigned a_rowBegin,
    unsigned a_rowEnd,
    unsigned a_colBegin,
    unsigned a_colEnd,
    bool a_createNew = false,
    Brood::Application::Components::Deck * a_deckPtr = nullptr ) [private]
```

updates the path specified by passed param in board by using the member variables.

It updates the tile starting from a_rowBegin to including a_rowEnd for the rows and for the column starting from a_colBegin to including a_colEnd. It calls [Brood::Application::Components::Tiles::UpdateTile\(\)](#) on each tile.

Parameters

<i>a_rowBegin</i>	index of row from where the tiles update should start
<i>a_rowEnd</i>	index of row from where the tiles update should end
<i>a_colBegin</i>	index of column from where the tiles update should start
<i>a_colEnd</i>	index of column from where the tiles update should end
<i>a_createNew</i>	true if new tiles should be created before updating the tiles -> default false

9.3.3 Member Data Documentation

9.3.3.1 m_boardBody

```
sf::RectangleShape Brood::Application::Components::Board::m_boardBody [private]
```

main board

9.3.3.2 m_boardPaths

```
std::vector<std::vector<Brood::Application::Components::Path*>> Brood::Application::Components::Board::m_boardPaths [private]
```

contains the board [Path](#) whic in turn contains tile

9.3.3.3 m_currActivePathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::Board::m_currActivePath←  
Ptr [private]
```

holds the pointer of the current active path used by the tile editor

9.3.3.4 m_numCols

```
unsigned Brood::Application::Components::Board::m_numCols [private]
```

number of cell columns in the board

9.3.3.5 m_numRows

```
unsigned Brood::Application::Components::Board::m_numRows [private]
```

number of cell rows in the board

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Board.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Board.cpp](#)

9.4 BoardEditor Class Reference

It allows the users to tweek the general game setting and game rules.

```
#include "BoardEditor.h"
```

9.4.1 Detailed Description

It allows the users to tweek the general game setting and game rules.

It is a container that bundles and control all the different editor workspace, namely [Board](#) Editor, Tile Editor, [Path](#) Editor, and Deck Editor.

It is a container that bundles elements that is present in [BoardEditor](#) workspace

It contains the following panel element: board X-position, board Y-position, boardX-size, board Y-Size, board row, board column

Derived from Workspace class

Todo allowd to enter number in the txt feild

Editor [WorkSpace](#) is where the user edit or tweek different setting to make a game project

Derived from the [WorkSpace](#) class

It is a container that bundles elements that is present in GameEditor workspace. It contains the following element:

Derived from Workspace class

Todo allowd to enter number in the txt feild

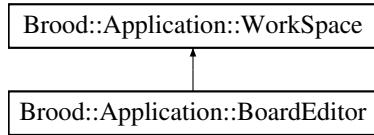
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[BoardEditor.h](#)

9.5 Brood::Application::BoardEditor Class Reference

```
#include <BoardEditor.h>
```

Inheritance diagram for Brood::Application::BoardEditor:



Public Member Functions

- `BoardEditor (Brood::Application::Components::GameManager *aGameData, sf::RectangleShape *a←_panelPtr)`
default constructor
- `~BoardEditor ()`
default destructor
- `void InitializeWorkSpace ()`
Initializes the work space.
- `void Update ()`
Updates function.
- `void UpdateAllDisplayElement ()`
calls the updates all the display element for the current active editor
- `void Draw (sf::RenderWindow &a_window)`
Draw funciton.
- `void Debugger ()`
debugger funciton
- `void UpdateBoardXSizePanelElement ()`
checks if the user interacted with the x size panel
- `void UpdateBoardYSizePanelElement ()`
checks if the user interacted with the y size panel
- `void UpdateBoardXPosPanelElement ()`
checks if the user interacted with the x position panel
- `void UpdateBoardYPosPanelElement ()`
checks if the user interacted with the y position panel
- `void UpdateBoardRowPanelElement ()`
checks if the user interacted with the row panel
- `void UpdateBoardColPanelElement ()`
checks if the user interacted with the column panel

Private Attributes

- sf::RectangleShape * `m_panelBodyPtr`
holds the panel body
- Brood::BroodUI::TextBox * `m_txtSettingTitle`
- Brood::BroodUI::TextBox * `m_txtBoardSizePromtX`
holds the text `Board X` size prompt
- Brood::BroodUI::Button * `m_btnBoardSizeDecX`
clicking it will decrease the board x size by 5
- Brood::BroodUI::TextBox * `m_txtBoardSizeX`
displays the board's x size
- Brood::BroodUI::Button * `m_btnBoardSizeIncX`
clicking it will increase the board x size by 5
- Brood::BroodUI::TextBox * `m_txtBoardSizePromtY`
holds the text `Board y` size prompt
- Brood::BroodUI::Button * `m_btnBoardSizeDecY`
clicking it will decrease the board y size by 5
- Brood::BroodUI::TextBox * `m_txtBoardSizeY`
displays the board's y size
- Brood::BroodUI::Button * `m_btnBoardSizeIncY`
clicking it will increase the board x size by 5
- Brood::BroodUI::TextBox * `m_txtBoardPosPromtX`
holds the text `Board X` position prompt
- Brood::BroodUI::Button * `m_btnBoardPosDecX`
clicking it will decrease the board x position by 5
- Brood::BroodUI::TextBox * `m_txtBoardPosX`
displays the board's x position
- Brood::BroodUI::Button * `m_btnBoardPosIncX`
clicking it will increase the board x position by 5
- Brood::BroodUI::TextBox * `m_txtBoardPosPromtY`
holds the text `Board y` position prompt
- Brood::BroodUI::Button * `m_btnBoardPosDecY`
clicking it will decrease the board y position by 5
- Brood::BroodUI::TextBox * `m_txtBoardPosY`
displays the board's y position
- Brood::BroodUI::Button * `m_btnBoardPosIncY`
clicking it will increase the board y position by 5
- Brood::BroodUI::TextBox * `m_txtBoardRowPromt`
holds the text `Board Row` number prompt
- Brood::BroodUI::Button * `m_btnBoardDecRow`
clicking it will decrease the board's row by 1
- Brood::BroodUI::TextBox * `m_txtBoardRow`
displays the board's Row number
- Brood::BroodUI::Button * `m_btnBoardIncRow`
clicking it will increase the board's row by 1
- Brood::BroodUI::TextBox * `m_txtBoardColPromt`
holds the text `Board Column` number prompt
- Brood::BroodUI::Button * `m_btnBoardDecCol`
clicking it will decrease the board's Column by 1
- Brood::BroodUI::TextBox * `m_txtBoardCol`
displays the board's Column number
- Brood::BroodUI::Button * `m_btnBoardIncCol`
clicking it will increase the board's Column by 1

Additional Inherited Members

9.5.1 Constructor & Destructor Documentation

9.5.1.1 BoardEditor()

```
BoardEditor::BoardEditor (
    Brood::Application::Components::GameDataManager * aGameData,
    sf::RectangleShape * aPanelPtr )
```

default constructor

Parameters

<i>aGameData</i>	pointer to the game data object
<i>aPanelPtr</i>	pointer to the panel body object

9.5.1.2 ~BoardEditor()

```
BoardEditor::~BoardEditor ( )
```

default destructor

9.5.2 Member Function Documentation

9.5.2.1 Debugger()

```
void BoardEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.5.2.2 Draw()

```
void BoardEditor::Draw (
    sf::RenderWindow & aWindow ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.5.2.3 InitializeWorkSpace()

```
void BoardEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the following panel element: board X-position, board Y-position, boardX-size, board Y-Size, board row, board column

Implements [Brood::Application::WorkSpace](#).

9.5.2.4 Update()

```
void BoardEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace. It Update the following panel element: board X-position, board Y-position, boardX-size, board Y-Size, board row, board column

Implements [Brood::Application::WorkSpace](#).

9.5.2.5 UpdateAllDisplayElement()

```
void BoardEditor::UpdateAllDisplayElement ( ) [virtual]
```

calls the updates all the display element for the current active editor

@virtual

Implements [Brood::Application::WorkSpace](#).

9.5.2.6 UpdateBoardColPanelElement()

```
void BoardEditor::UpdateBoardColPanelElement ( )
```

checks if the user interacted with the column panel

Column panel contains **Board** column number prompt textbox, **Board** column number value text box, button to increase the **Board** column number, and button to decrease the **Board** column number.

Only the button to increase the **Board** column number, and button to decrease the **Board** column number are interactable

If the interactable button was pressed then the **Board** column number is increased or decreased by 1. Column number should be more than 1 but less than 100

9.5.2.7 UpdateBoardRowPanelElement()

```
void BoardEditor::UpdateBoardRowPanelElement ( )
```

checks if the user interacted with the row panel

Row panel contains **Board** row number prompt textbox, **Board** row number value text box, button to increase the **Board** row number, and button to decrease the **Board** row number.

Only the button to increase the **Board** row number, and button to decrease the **Board** row number are interactable

If the interactable button was pressed then the **Board** row number is increased or decreased by 1. Row number should be more than 1 but less than 100

9.5.2.8 UpdateBoardXPosPanelElement()

```
void BoardEditor::UpdateBoardXPosPanelElement ( )
```

checks if the user interacted with the x position panel

X position panel contains **Board** X position prompt textbox, **Board** X position value text box, button to increase the **Board** X position, and button to decrease the **Board** X position.

Only the button to increase the **Board** X position, and button to decrease the **Board** X position are interactable

If the interactable button was pressed then the X pos is increased or decreased by 5 if the board does not go outside the board window

9.5.2.9 UpdateBoardXSizePanelElement()

```
void BoardEditor::UpdateBoardXSizePanelElement ( )
```

checks if the user interacted with the x size panel

X size panel contains **Board** X size prompt textbox, **Board** X size value text box, button to increase the **Board** X size, and button to decrease the **Board** X size.

Only the button to increase the **Board** X size, and button to decrease the **Board** X size are interactable

If the interactable button was pressed then the X size is increased or decreased by 5 if the board does not go outside the board window

9.5.2.10 UpdateBoardYPosPanelElement()

```
void BoardEditor::UpdateBoardYPosPanelElement ( )
```

checks if the user interacted with the y position panel

Y position panel contains **Board** Y position prompt textbox, **Board** Y position value text box, button to increase the **Board** Y position, and button to decrease the **Board** Y position.

Only the button to increase the **Board** Y position, and button to decrease the **Board** Y position are interactable

If the interactable button was pressed then the Y pos is increased or decreased by 5 if the board does not go outside the board window

9.5.2.11 UpdateBoardYSizePanelElement()

```
void BoardEditor::UpdateBoardYSizePanelElement ( )
```

checks if the user interacted with the y size panel

Y Size panel contains **Board** Y size prompt textbox, **Board** Y size value text box, button to increase the **Board** Y size, and button to decrease the **Board** Y size.

Only the button to increase the **Board** Y size, and button to decrease the **Board** Y size are interactable

If the interactable button was pressed then the Y size is increased or decreased by 5 if the board does not go outside the board window

9.5.3 Member Data Documentation

9.5.3.1 m_btnBoardDecCol

```
Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardDecCol [private]
```

clicking it will decrease the board's Column by 1

9.5.3.2 m_btnBoardDecRow

```
Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardDecRow [private]
```

clicking it will decrease the board's row by 1

9.5.3.3 m_btnBoardIncCol

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardIncCol [private]`

clicking it will increase the board's Column by 1

9.5.3.4 m_btnBoardIncRow

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardIncRow [private]`

clicking it will increase the board's row by 1

9.5.3.5 m_btnBoardPosDecX

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardPosDecX [private]`

clicking it will decrease the board x position by 5

9.5.3.6 m_btnBoardPosDecY

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardPosDecY [private]`

clicking it will decrease the board y position by 5

9.5.3.7 m_btnBoardPosIncX

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardPosIncX [private]`

clicking it will increase the board x position by 5

9.5.3.8 m_btnBoardPosIncY

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardPosIncY [private]`

clicking it will increase the board y position by 5

9.5.3.9 m_btnBoardSizeDecX

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardSizeDecX [private]`

clicking it will decrease the board x size by 5

9.5.3.10 m_btnBoardSizeDecY

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardSizeDecY [private]`

clicking it will decrease the board y size by 5

9.5.3.11 m_btnBoardSizeIncX

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardSizeIncX [private]`

clicking it will increase the board x size by 5

9.5.3.12 m_btnBoardSizeIncY

`Brood::BroodUI::Button* Brood::Application::BoardEditor::m_btnBoardSizeIncY [private]`

clicking it will increase the board x size by 5

9.5.3.13 m_panelBodyPtr

`sf::RectangleShape* Brood::Application::BoardEditor::m_panelBodyPtr [private]`

holds the panel body

9.5.3.14 m_txtBoardCol

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardCol [private]`

displays the board's Column number

9.5.3.15 m_txtBoardColPromt

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardColPromt [private]

holds the text **Board** Column number prompt

9.5.3.16 m_txtBoardPosPromtX

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardPosPromtX [private]

holds the text **Board** X position prompt

9.5.3.17 m_txtBoardPosPromtY

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardPosPromtY [private]

holds the text **Board** y position prompt

9.5.3.18 m_txtBoardPosX

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardPosX [private]

displays the board's x position

9.5.3.19 m_txtBoardPosY

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardPosY [private]

displays the board's y position

9.5.3.20 m_txtBoardRow

Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardRow [private]

displays the board's Row number

9.5.3.21 m_txtBoardRowPromt

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardRowPromt [private]`

holds the text **Board** Row number prompt

9.5.3.22 m_txtBoardSizePromtX

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardSizePromtX [private]`

holds the text **Board** X size prompt

9.5.3.23 m_txtBoardSizePromtY

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardSizePromtY [private]`

holds the text **Board** y size prompt

9.5.3.24 m_txtBoardSizeX

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardSizeX [private]`

displays the board's x size

9.5.3.25 m_txtBoardSizeY

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtBoardSizeY [private]`

displays the board's y size

9.5.3.26 m_txtSettingTitle

`Brood::BroodUI::TextBox* Brood::Application::BoardEditor::m_txtSettingTitle [private]`

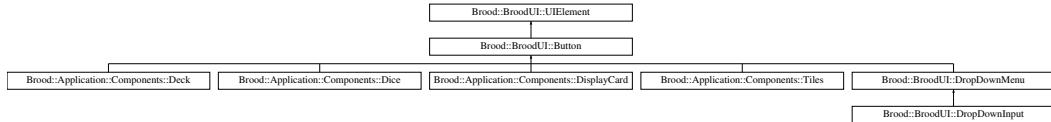
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[BoardEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[BoardEditor.cpp](#)

9.6 Brood::BroodUI::Button Class Reference

```
#include <Button.h>
```

Inheritance diagram for Brood::BroodUI::Button:



Public Member Functions

- `Button (Brood::BroodUI::UIElement *a_parentPtr=nullptr, Brood::BroodUI::ENUM_UIType a_enum←Type=Brood::BroodUI::ENUM_UIType::UI_button)`
Default Constructor.
- `virtual ~Button ()`
default destructor
- `Button (const Button &a_otherElement)`
Copy constructor.
- `Brood::BroodUI::Button & operator= (const Button &a_otherButton)`
assignment operator
- `Brood::SpriteHandler & GetSpriteBody ()`
Getter function to get the sprite body.
- `void SetSelected (bool a_selected)`
setter function to set the state of the element i.e. if it is selected or not
- `virtual bool DoElement () override`
checks if the logics of the element is to be executed or not
- `virtual void Debugger ()`
changes the text/texture of the element to its the element ID when debug is on and changes it back to the text/texture it had before debugging when debug is off.

Static Public Member Functions

- static `Brood::BroodUI::Button * DyCreateButton (sf::Vector2f a_size, sf::Vector2f a_pos, std::string a_text="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
helper function to dynamically create a button element initializes it, and adds it to m_unNameList
- static `Brood::BroodUI::Button * DyCreateButton (float a_sizeX, float a_sizeY, float a_posX, float a_posY, std::string a_text="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
helper function to dynamically create a button element initializes it, and adds it to m_unNameList

Protected Attributes

- `Brood::SpriteHandler m_bodySprite`
object to set texture

Additional Inherited Members

9.6.1 Constructor & Destructor Documentation

9.6.1.1 Button() [1/2]

```
Button::Button (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr,
    Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_button )
```

Default Constructor.

Initializes the button object

Note

by default it sets the text that is displayed on the button to empty string

Parameters

<code>m_parentPtr</code>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------------	---

9.6.1.2 ~Button()

```
Button::~Button ( ) [virtual]
```

default destructor

9.6.1.3 Button() [2/2]

```
Button::Button (
    const Button & a_otherElement )
```

Copy constructor.

Parameters

<code>a_otherElement</code>	reference to the uiElement which is used to copy the data form
-----------------------------	--

9.6.2 Member Function Documentation

9.6.2.1 Debugger()

```
void Button::Debugger ( ) [virtual]
```

changes the text/texture of the element to its the element ID when debug is on and changes it back to the text/texture it had before debugging when debug is off.

Implements [Brood::BroodUI::UIElement](#).

Reimplemented in [Brood::Application::Components::DisplayCard](#), and [Brood::BroodUI::DropDownMenu](#).

9.6.2.2 DoElement()

```
bool Button::DoElement ( ) [override], [virtual]
```

checks if the logics of the element is to be executed or not

It checks the mouse position and button state to determine if to execute the elements logic or not. It does this by manipulating the element selection class. It also set the overlay

Returns

true if the element's funciton is to be executed; else false

Reimplemented from [Brood::BroodUI::UIElement](#).

Reimplemented in [Brood::BroodUI::DropDownMenu](#).

9.6.2.3 DyCreateButton() [1/2]

```
Brood::BroodUI::Button * Button::DyCreateButton (
    float a_sizeX,
    float a_sizeY,
    float a_posX,
    float a_posY,
    std::string a_text = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
) [static]
```

helper function to dynamically create a button element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_sizeX</i>	length of the button element
<i>a_sizeY</i>	width of the button element
<i>a_posX</i>	x-position of the button element
<i>a_posY</i>	y-position of the button element
<i>a_text</i>	text to display in the button element ; default ""
<i>a_color</i>	color of the button element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created [Button](#) element

9.6.2.4 DyCreateButton() [2/2]

```
Brood::BroodUI::Button * Button::DyCreateButton (
    sf::Vector2f a_size,
    sf::Vector2f a_pos,
    std::string a_text = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
) [static]
```

helper function to dynamically create a button element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_size</i>	size of the button element
<i>a_pos</i>	position of the button element
<i>a_text</i>	text to display in the button element ; default ""
<i>a_color</i>	color of the button element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created [Button](#) element

9.6.2.5 GetSpriteBody()

```
Brood::SpriteHandler & Button::GetSpriteBody ( )
```

Getter function to get the sprite body.

Sprite body allows to manipulate the sprite/texture

Returns

reference to the spritebody object

9.6.2.6 operator=()

```
Brood::BroodUI::Button & Button::operator= (
    const Button & a_otherButton )
```

assignment operator

Parameters

<i>a_otherButton</i>	reference to the button which is used to copy the data form
----------------------	---

Returns

pointer to this element

9.6.2.7 SetSelected()

```
void Button::SetSelected (
    bool a_selected ) [virtual]
```

setter function to set the state of the element i.e. if it is selected or not

Note

this is used to display the active overlay or not

Parameters

<i>a_selected</i>	true if current button is selected or not
-------------------	---

Reimplemented from [Brood::BroodUI::UIElement](#).

9.6.3 Member Data Documentation

9.6.3.1 m_bodySprite

```
Brood::SpriteHandler Brood::BroodUI::Button::m_bodySprite [protected]
```

object to set texture

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Button.h
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Button.cpp

9.7 Button Class Reference

a uiemnt to work as a [Button](#).

```
#include "Button.h"
```

9.7.1 Detailed Description

a uiemnt to work as a [Button](#).

A class derived from [TextBox](#) class.

```
9.7.1.0.1 Example Case // window object
sf::RenderWindow window( sf::VideoMode( 500, 500 ), "BroodMaker" );
// event object
sf::Event events;
// creating a Button object
Brood::BroodUI::Button myButton;
myButton.SetBodySize( 100, 50 );
myButton.SetBodyPosition( 55, 0 );
myButton.SetBodyColor( sf::Color::Red );
myButtonSetFont( font );
// if you want to set a text
myButton.SetText( "----" );
// if you want to set a texture
myButton.GetSpriteBody().SetSpriteLength( 50 );
myButton.GetSpriteBody().SetSpriteHeight( 50 );
// loading the texture
myButton.GetSpriteBody().SetTextureFromFilePath( fileName );
myButton.GetSpriteBody().SetSpriteFromTexture( 0 );
//app loop
while( !exit )
{
    // event loop

    // logic
    if (myButton.DoElement()) {
        std::cout << "myButton Pressed" << std::endl;
    }

    // render
    myButton.Draw( window );
}
```

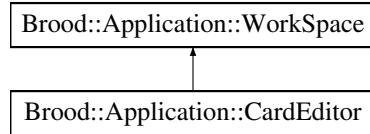
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Button.h

9.8 Brood::Application::CardEditor Class Reference

```
#include <CardEditor.h>
```

Inheritance diagram for Brood::Application::CardEditor:



Public Member Functions

- [CardEditor \(Brood::Application::Components::GameDataManager *aGameData, sf::RectangleShape *aPanelPtr\)](#)
default constructor
- [~CardEditor \(\)](#)
default destructor
- [void InitializeWorkSpace \(\)](#)
Initializes the work space.
- [void Update \(\)](#)
Updates function.
- [void UpdateAllDisplayElement \(\)](#)
updates the display element
- [void Draw \(sf::RenderWindow &a_window\)](#)
Draw funciton.
- [void Debugger \(\)](#)
debugger funciton

Private Member Functions

- [void InitializeSettingSelectionDDI \(\)](#)
Initializes the setting Section dropdown input which is at the top of the setting pannel.
- [void UpdateSettingSelectionDDI \(\)](#)
checks if the user interacted with setting Selection and updates accordingly
- [void InitializeCardDisplayPanel \(\)](#)
Initializes the panel element of the Card Display setting.
- [void UpdateCardDisplayPanel \(\)](#)
Updates the panel element of the Card Display setting.
- [void DrawCardDisplayPanel \(sf::RenderWindow &a_window\)](#)
Draws the panel element of the Card Display setting.
- [void InitializeCardDisplayFcompPanel \(\)](#)
Initializes the panel element of the Card Display front component setting.
- [void UpdateCardDisplayFcompPanel \(\)](#)
Updates the panel element of the Card Display front component setting.
- [void DrawCardDisplayFcompPanel \(sf::RenderWindow &a_window\)](#)
Draws the panel element of the Card Display front component setting.
- [void InitializeCardDisplayBcompPanel \(\)](#)

- void [UpdateCardDisplayBcompPanel \(\)](#)
Initializes the panel element of the Card Display back component setting.
- void [DrawCardDisplayBcompPanel \(sf::RenderWindow &a_window\)](#)
Updates the panel element of the Card Display back component setting.
- void [InializeCardInfoPanel \(\)](#)
Draws the panel element of the Card Display back component setting.
- void [UpdateCardInfoPanel \(\)](#)
Initializes the panel element of the Card Display back component setting.
- void [DrawCardInfoPanel \(sf::RenderWindow &a_window\)](#)
Updates the panel element of the Card Display back component setting.
- void [UpdateCardFrontFileTexture \(\)](#)
Draws the panel element of the Card Display back component setting.
- void [UpdateCardBackFileTexture \(\)](#)
checks if the user interacted with the laod Card texture panel
- void [UpdateCurrSelectedDeckIdx \(\)](#)
checks if the user interacted with the laod Card texture panel
- void [UpdateCurrSelectedCardIdx \(\)](#)
checks if the user interacted with the current selceted deck index panel
- void [UpdateCurrCardInfoTime \(\)](#)
checks if the user interacted with the current selceted Card index panel
- void [UpdateCurrCardInfoUp \(\)](#)
checks if the user interacted with the current selceted Card Up panel
- void [UpdateCurrCardInfoDown \(\)](#)
checks if the user interacted with the current selceted Card Down panel
- void [UpdateCurrCardQuestionInput \(\)](#)
checks if the user interacted with the set Curr Card Question panel
- void [UpdateCurrCardAnswerInput \(\)](#)
checks if the user interacted with the set Curr Card Answer panel
- void [UpdateTurnCard \(\)](#)
checks if the user interacted with the turn card panel

Private Attributes

- sf::RectangleShape * [m_panelBodyPtr](#)
holds the panel body
- unsigned [m_selectedSettingIdx](#)
- Brood::BroodUI::DropDownInput * [m_ddiSettingSelection](#)
- Brood::BroodUI::TextBox * [m_txtCardSizeXPrompt](#)
holds the text Card size x
- Brood::BroodUI::Button * [m_btnCardDecSizeX](#)
clicking it will decrease the Card size x number by 1
- Brood::BroodUI::TextBox * [m_txtCardSizeX](#)
displays the current Card size X
- Brood::BroodUI::Button * [m_btnCardIncSizeX](#)
clicking it will increase the Card size x number by 1
- Brood::BroodUI::TextBox * [m_txtCardSizeYPrompt](#)
holds the text Card size y
- Brood::BroodUI::Button * [m_btnCardDecSizeY](#)
clicking it will decrease the Card size y number by 1

- Brood::BroodUI::TextBox * m_txtCardSizeY
displays the current Card size y
- Brood::BroodUI::Button * m_btnCardIncSizeY
clicking it will increase the Card size y number by 1
- Brood::BroodUI::TextBox * m_txtCardPosXPrompt
holds the text Card position x
- Brood::BroodUI::Button * m_btnCardDecPosX
clicking it will decrease the Card Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardPosX
displays the current Card Position X
- Brood::BroodUI::Button * m_btnCardIncPosX
clicking it will increase the Card Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardPosYPrompt
holds the text Card Position y
- Brood::BroodUI::Button * m_btnCardDecPosY
clicking it will decrease the Card Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardPosY
displays the current Card Position y
- Brood::BroodUI::Button * m_btnCardIncPosY
clicking it will increase the Card Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontBgFileNamePrompt
holds the CardFront texture filename prompt
- Brood::BroodUI::TextBox * m_txtCardFrontBgFileInput
holds the file name user enters
- Brood::BroodUI::Button * m_btnCardFrontBgOpenFile
clicking it will load the CardFront file
- Brood::BroodUI::TextBox * m_txtCardBackBgFileNamePrompt
holds the CardBack texture filename prompt
- Brood::BroodUI::TextBox * m_txtCardBackBgFileInput
holds the file name user enters
- Brood::BroodUI::Button * m_btnCardBackBgOpenFile
clicking it will load the CardBack file
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptPosXPrompt
holds the text CardFrontTimePrompt position x
- Brood::BroodUI::Button * m_btnCardFrontTimePromptDecPosX
clicking it will decrease the CardFrontTimePrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptPosX
displays the current CardFrontTimePrompt Position X
- Brood::BroodUI::Button * m_btnCardFrontTimePromptIncPosX
clicking it will increase the CardFrontTimePrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptPosYPrompt
holds the text CardFrontTimePrompt Position y
- Brood::BroodUI::Button * m_btnCardFrontTimePromptDecPosY
clicking it will decrease the CardFrontTimePrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptPosY
displays the current CardFrontTimePrompt Position y
- Brood::BroodUI::Button * m_btnCardFrontTimePromptIncPosY
clicking it will increase the CardFrontTimePrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValuePosXPrompt
holds the text CardFrontTimeValue position x
- Brood::BroodUI::Button * m_btnCardFrontTimeValueDecPosX

- clicking it will decrease the CardFrontTimeValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValuePosX
 - displays the current CardFrontTimeValue Position X
- Brood::BroodUI::Button * m_btnCardFrontTimeValueIncPosX
 - clicking it will increase the CardFrontTimeValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValuePosYPrompt
 - holds the text CardFrontTimeValue Position y
- Brood::BroodUI::Button * m_btnCardFrontTimeValueDecPosY
 - clicking it will decrease the CardFrontTimeValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValuePosY
 - displays the current CardFrontTimeValue Position y
- Brood::BroodUI::Button * m_btnCardFrontTimeValueIncPosY
 - clicking it will increase the CardFrontTimeValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardUpPromptPosXPrompt
 - holds the text CardUpPrompt position x
- Brood::BroodUI::Button * m_btnCardUpPromptDecPosX
 - clicking it will decrease the CardUpPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardUpPromptPosX
 - displays the current CardUpPrompt Position X
- Brood::BroodUI::Button * m_btnCardUpPromptIncPosX
 - clicking it will increase the CardUpPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardUpPromptPosYPrompt
 - holds the text CardUpPrompt Position y
- Brood::BroodUI::Button * m_btnCardUpPromptDecPosY
 - clicking it will decrease the CardUpPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardUpPromptPosY
 - displays the current CardUpPrompt Position y
- Brood::BroodUI::Button * m_btnCardUpPromptIncPosY
 - clicking it will increase the CardUpPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardUpValuePosXPrompt
 - holds the text CardUpValue position x
- Brood::BroodUI::Button * m_btnCardUpValueDecPosX
 - clicking it will decrease the CardUpValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardUpValuePosX
 - displays the current CardUpValue Position X
- Brood::BroodUI::Button * m_btnCardUpValueIncPosX
 - clicking it will increase the CardUpValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardUpValuePosYPrompt
 - holds the text CardUpValue Position y
- Brood::BroodUI::Button * m_btnCardUpValueDecPosY
 - clicking it will decrease the CardUpValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardUpValuePosY
 - displays the current CardUpValue Position y
- Brood::BroodUI::Button * m_btnCardUpValueIncPosY
 - clicking it will increase the CardUpValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptPosXPrompt
 - holds the text CardDownPrompt position x
- Brood::BroodUI::Button * m_btnCardDownPromptDecPosX
 - clicking it will decrease the CardDownPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptPosX
 - displays the current CardDownPrompt Position X

- Brood::BroodUI::Button * m_btnCardDownPromptIncPosX
clicking it will increase the CardDownPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptPosYPrompt
holds the text CardDownPrompt Position y
- Brood::BroodUI::Button * m_btnCardDownPromptDecPosY
clicking it will decrease the CardDownPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptPosY
displays the current CardDownPrompt Position y
- Brood::BroodUI::Button * m_btnCardDownPromptIncPosY
clicking it will increase the CardDownPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValuePosXPrompt
holds the text CardDownValue position x
- Brood::BroodUI::Button * m_btnCardDownValueDecPosX
clicking it will decrease the CardDownValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValuePosX
displays the current CardDownValue Position X
- Brood::BroodUI::Button * m_btnCardDownValueIncPosX
clicking it will increase the CardDownValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValuePosYPrompt
holds the text CardDownValue Position y
- Brood::BroodUI::Button * m_btnCardDownValueDecPosY
clicking it will decrease the CardDownValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValuePosY
displays the current CardDownValue Position y
- Brood::BroodUI::Button * m_btnCardDownValueIncPosY
clicking it will increase the CardDownValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardPosXPrompt
holds the text CardTurnCard position x
- Brood::BroodUI::Button * m_btnCardTurnCardDecPosX
clicking it will decrease the CardTurnCard Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardPosX
displays the current CardTurnCard Position X
- Brood::BroodUI::Button * m_btnCardTurnCardIncPosX
clicking it will increase the CardTurnCard Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardPosYPrompt
holds the text CardTurnCard Position y
- Brood::BroodUI::Button * m_btnCardTurnCardDecPosY
clicking it will decrease the CardTurnCard Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardPosY
displays the current CardTurnCard Position y
- Brood::BroodUI::Button * m_btnCardTurnCardIncPosY
clicking it will increase the CardTurnCard Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptSizeXPrompt
holds the text CardFrontTimePrompt size x
- Brood::BroodUI::Button * m_btnCardFrontTimePromptDecSizeX
clicking it will decrease the CardFrontTimePrompt size x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptSizeX
displays the current CardFrontTimePrompt size X
- Brood::BroodUI::Button * m_btnCardFrontTimePromptIncSizeX
clicking it will increase the CardFrontTimePrompt size x number by 5
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptSizeYPrompt

- holds the text CardFrontTimePrompt size y*
- Brood::BroodUI::Button * m_btnCardFrontTimePromptDecSizeY
 - clicking it will decrease the CardFrontTimePrompt size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardFrontTimePromptSizeY
 - displays the current CardFrontTimePrompt size y*
- Brood::BroodUI::Button * m_btnCardFrontTimePromptIncSizeY
 - clicking it will increase the CardFrontTimePrompt size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValueSizeXPrompt
 - holds the text CardFrontTimeValue size x*
- Brood::BroodUI::Button * m_btnCardFrontTimeValueDecSizeX
 - clicking it will decrease the CardFrontTimeValue size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValueSizeX
 - displays the current CardFrontTimeValue size X*
- Brood::BroodUI::Button * m_btnCardFrontTimeValueIncSizeX
 - clicking it will increase the CardFrontTimeValue size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValueSizeYPrompt
 - holds the text CardFrontTimeValue size y*
- Brood::BroodUI::Button * m_btnCardFrontTimeValueDecSizeY
 - clicking it will decrease the CardFrontTimeValue size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardFrontTimeValueSizeY
 - displays the current CardFrontTimeValue size y*
- Brood::BroodUI::Button * m_btnCardFrontTimeValueIncSizeY
 - clicking it will increase the CardFrontTimeValue size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpPromptSizeXPrompt
 - holds the text CardUpPrompt size x*
- Brood::BroodUI::Button * m_btnCardUpPromptDecSizeX
 - clicking it will decrease the CardUpPrompt size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpPromptSizeX
 - displays the current CardUpPrompt size X*
- Brood::BroodUI::Button * m_btnCardUpPromptIncSizeX
 - clicking it will increase the CardUpPrompt size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpPromptSizeYPrompt
 - holds the text CardUpPrompt size y*
- Brood::BroodUI::Button * m_btnCardUpPromptDecSizeY
 - clicking it will decrease the CardUpPrompt size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpPromptSizeY
 - displays the current CardUpPrompt size y*
- Brood::BroodUI::Button * m_btnCardUpPromptIncSizeY
 - clicking it will increase the CardUpPrompt size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpValueSizeXPrompt
 - holds the text CardUpValue size x*
- Brood::BroodUI::Button * m_btnCardUpValueDecSizeX
 - clicking it will decrease the CardUpValue size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpValueSizeX
 - displays the current CardUpValue size X*
- Brood::BroodUI::Button * m_btnCardUpValueIncSizeX
 - clicking it will increase the CardUpValue size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardUpValueSizeYPrompt
 - holds the text CardUpValue size y*
- Brood::BroodUI::Button * m_btnCardUpValueDecSizeY
 - clicking it will decrease the CardUpValue size y number by 5*

- Brood::BroodUI::TextBox * m_txtCardUpValueSizeY
displays the current CardUpValue size y
- Brood::BroodUI::Button * m_btnCardUpValueIncSizeY
clicking it will increase the CardUpValue size y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptSizeXPrompt
holds the text CardDownPrompt size x
- Brood::BroodUI::Button * m_btnCardDownPromptDecSizeX
clicking it will decrease the CardDownPrompt size x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptSizeX
displays the current CardDownPrompt size X
- Brood::BroodUI::Button * m_btnCardDownPromptIncSizeX
clicking it will increase the CardDownPrompt size x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptSizeYPrompt
holds the text CardDownPrompt size y
- Brood::BroodUI::Button * m_btnCardDownPromptDecSizeY
clicking it will decrease the CardDownPrompt size y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownPromptSizeY
displays the current CardDownPrompt size y
- Brood::BroodUI::Button * m_btnCardDownPromptIncSizeY
clicking it will increase the CardDownPrompt size y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValueSizeXPrompt
holds the text CardDownValue size x
- Brood::BroodUI::Button * m_btnCardDownValueDecSizeX
clicking it will decrease the CardDownValue size x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValueSizeX
displays the current CardDownValue size X
- Brood::BroodUI::Button * m_btnCardDownValueIncSizeX
clicking it will increase the CardDownValue size x number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValueSizeYPrompt
holds the text CardDownValue size y
- Brood::BroodUI::Button * m_btnCardDownValueDecSizeY
clicking it will decrease the CardDownValue size y number by 5
- Brood::BroodUI::TextBox * m_txtCardDownValueSizeY
displays the current CardDownValue size y
- Brood::BroodUI::Button * m_btnCardDownValueIncSizeY
clicking it will increase the CardDownValue size y number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardSizeXPrompt
holds the text CardTurnCard size x
- Brood::BroodUI::Button * m_btnCardTurnCardDecSizeX
clicking it will decrease the CardTurnCard size x number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardSizeX
displays the current CardTurnCard size X
- Brood::BroodUI::Button * m_btnCardTurnCardIncSizeX
clicking it will increase the CardTurnCard size x number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardSizeYPrompt
holds the text CardTurnCard size y
- Brood::BroodUI::Button * m_btnCardTurnCardDecSizeY
clicking it will decrease the CardTurnCard size y number by 5
- Brood::BroodUI::TextBox * m_txtCardTurnCardSizeY
displays the current CardTurnCard size y
- Brood::BroodUI::Button * m_btnCardTurnCardIncSizeY
clicking it will increase the CardTurnCard size y number by 5

- clicking it will increase the CardTurnCard size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptPosXPrompt
 holds the text CardBackTimePrompt position x
- Brood::BroodUI::Button * m_btnCardBackTimePromptDecPosX
 clicking it will decrease the CardBackTimePrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptPosX
 displays the current CardBackTimePrompt Position X
- Brood::BroodUI::Button * m_btnCardBackTimePromptIncPosX
 clicking it will increase the CardBackTimePrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptPosYPrompt
 holds the text CardBackTimePrompt Position y
- Brood::BroodUI::Button * m_btnCardBackTimePromptDecPosY
 clicking it will decrease the CardBackTimePrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptPosY
 displays the current CardBackTimePrompt Position y
- Brood::BroodUI::Button * m_btnCardBackTimePromptIncPosY
 clicking it will increase the CardBackTimePrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValuePosXPrompt
 holds the text CardBackTimeValue position x
- Brood::BroodUI::Button * m_btnCardBackTimeValueDecPosX
 clicking it will decrease the CardBackTimeValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValuePosX
 displays the current CardBackTimeValue Position X
- Brood::BroodUI::Button * m_btnCardBackTimeValueIncPosX
 clicking it will increase the CardBackTimeValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValuePosYPrompt
 holds the text CardBackTimeValue Position y
- Brood::BroodUI::Button * m_btnCardBackTimeValueDecPosY
 clicking it will decrease the CardBackTimeValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValuePosY
 displays the current CardBackTimeValue Position y
- Brood::BroodUI::Button * m_btnCardBackTimeValueIncPosY
 clicking it will increase the CardBackTimeValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptPosXPrompt
 holds the text CardQuestionPrompt position x
- Brood::BroodUI::Button * m_btnCardQuestionPromptDecPosX
 clicking it will decrease the CardQuestionPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptPosX
 displays the current CardQuestionPrompt Position X
- Brood::BroodUI::Button * m_btnCardQuestionPromptIncPosX
 clicking it will increase the CardQuestionPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptPosYPrompt
 holds the text CardQuestionPrompt Position y
- Brood::BroodUI::Button * m_btnCardQuestionPromptDecPosY
 clicking it will decrease the CardQuestionPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptPosY
 displays the current CardQuestionPrompt Position y
- Brood::BroodUI::Button * m_btnCardQuestionPromptIncPosY
 clicking it will increase the CardQuestionPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionValuePosXPrompt
 holds the text CardQuestionValue position x

- Brood::BroodUI::Button * m_btnCardQuestionValueDecPosX
clicking it will decrease the CardQuestionValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionValuePosX
displays the current CardQuestionValue Position X
- Brood::BroodUI::Button * m_btnCardQuestionValueIncPosX
clicking it will increase the CardQuestionValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionValuePosYPrompt
holds the text CardQuestionValue Position y
- Brood::BroodUI::Button * m_btnCardQuestionValueDecPosY
clicking it will decrease the CardQuestionValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionValuePosY
displays the current CardQuestionValue Position y
- Brood::BroodUI::Button * m_btnCardQuestionValueIncPosY
clicking it will increase the CardQuestionValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptPosXPrompt
holds the text CardAnswerPrompt position x
- Brood::BroodUI::Button * m_btnCardAnswerPromptDecPosX
clicking it will decrease the CardAnswerPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptPosX
displays the current CardAnswerPrompt Position X
- Brood::BroodUI::Button * m_btnCardAnswerPromptIncPosX
clicking it will increase the CardAnswerPrompt Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptPosYPrompt
holds the text CardAnswerPrompt Position y
- Brood::BroodUI::Button * m_btnCardAnswerPromptDecPosY
clicking it will decrease the CardAnswerPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptPosY
displays the current CardAnswerPrompt Position y
- Brood::BroodUI::Button * m_btnCardAnswerPromptIncPosY
clicking it will increase the CardAnswerPrompt Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerValuePosXPrompt
holds the text CardAnswerValue position x
- Brood::BroodUI::Button * m_btnCardAnswerValueDecPosX
clicking it will decrease the CardAnswerValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerValuePosX
displays the current CardAnswerValue Position X
- Brood::BroodUI::Button * m_btnCardAnswerValueIncPosX
clicking it will increase the CardAnswerValue Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerValuePosYPrompt
holds the text CardAnswerValue Position y
- Brood::BroodUI::Button * m_btnCardAnswerValueDecPosY
clicking it will decrease the CardAnswerValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardAnswerValuePosY
displays the current CardAnswerValue Position y
- Brood::BroodUI::Button * m_btnCardAnswerValueIncPosY
clicking it will increase the CardAnswerValue Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitPosXPrompt
holds the text CardSubmit position x
- Brood::BroodUI::Button * m_btnCardSubmitDecPosX
clicking it will decrease the CardSubmit Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitPosX

- displays the current CardSubmit Position X
- Brood::BroodUI::Button * m_btnCardSubmitIncPosX
 - clicking it will increase the CardSubmit Position x number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitPosYPrompt
 - holds the text CardSubmit Position y
- Brood::BroodUI::Button * m_btnCardSubmitDecPosY
 - clicking it will decrease the CardSubmit Position y number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitPosY
 - displays the current CardSubmit Position y
- Brood::BroodUI::Button * m_btnCardSubmitIncPosY
 - clicking it will increase the CardSubmit Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptSizeXPrompt
 - holds the text CardBackTimePrompt position x
- Brood::BroodUI::Button * m_btnCardBackTimePromptDecSizeX
 - clicking it will decrease the CardBackTimePrompt Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptSizeX
 - displays the current CardBackTimePrompt Size X
- Brood::BroodUI::Button * m_btnCardBackTimePromptIncSizeX
 - clicking it will increase the CardBackTimePrompt Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptSizeYPrompt
 - holds the text CardBackTimePrompt Size y
- Brood::BroodUI::Button * m_btnCardBackTimePromptDecSizeY
 - clicking it will decrease the CardBackTimePrompt Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimePromptSizeY
 - displays the current CardBackTimePrompt Size y
- Brood::BroodUI::Button * m_btnCardBackTimePromptIncSizeY
 - clicking it will increase the CardBackTimePrompt Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValueSizeXPrompt
 - holds the text CardBackTimeValue position x
- Brood::BroodUI::Button * m_btnCardBackTimeValueDecSizeX
 - clicking it will decrease the CardBackTimeValue Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValueSizeX
 - displays the current CardBackTimeValue Size X
- Brood::BroodUI::Button * m_btnCardBackTimeValueIncSizeX
 - clicking it will increase the CardBackTimeValue Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValueSizeYPrompt
 - holds the text CardBackTimeValue Size y
- Brood::BroodUI::Button * m_btnCardBackTimeValueDecSizeY
 - clicking it will decrease the CardBackTimeValue Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardBackTimeValueSizeY
 - displays the current CardBackTimeValue Size y
- Brood::BroodUI::Button * m_btnCardBackTimeValueIncSizeY
 - clicking it will increase the CardBackTimeValue Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptSizeXPrompt
 - holds the text CardQuestionPrompt position x
- Brood::BroodUI::Button * m_btnCardQuestionPromptDecSizeX
 - clicking it will decrease the CardQuestionPrompt Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptSizeX
 - displays the current CardQuestionPrompt Size X
- Brood::BroodUI::Button * m_btnCardQuestionPromptIncSizeX
 - clicking it will increase the CardQuestionPrompt Size x number by 5

- Brood::BroodUI::TextBox * m_txtCardQuestionPromptSizeYPrompt
 - holds the text CardQuestionPrompt Size y*
- Brood::BroodUI::Button * m_btnCardQuestionPromptDecSizeY
 - clicking it will decrease the CardQuestionPrompt Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardQuestionPromptSizeY
 - displays the current CardQuestionPrompt Size y*
- Brood::BroodUI::Button * m_btnCardQuestionPromptIncSizeY
 - clicking it will increase the CardQuestionPrompt Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardQuestionValueSizeXPrompt
 - holds the text CardQuestionValue position x*
- Brood::BroodUI::Button * m_btnCardQuestionValueDecSizeX
 - clicking it will decrease the CardQuestionValue Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardQuestionValueSizeX
 - displays the current CardQuestionValue Size X*
- Brood::BroodUI::Button * m_btnCardQuestionValueIncSizeX
 - clicking it will increase the CardQuestionValue Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardQuestionValueSizeYPrompt
 - holds the text CardQuestionValue Size y*
- Brood::BroodUI::Button * m_btnCardQuestionValueDecSizeY
 - clicking it will decrease the CardQuestionValue Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardQuestionValueSizeY
 - displays the current CardQuestionValue Size y*
- Brood::BroodUI::Button * m_btnCardQuestionValueIncSizeY
 - clicking it will increase the CardQuestionValue Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptSizeXPrompt
 - holds the text CardAnswerPrompt position x*
- Brood::BroodUI::Button * m_btnCardAnswerPromptDecSizeX
 - clicking it will decrease the CardAnswerPrompt Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptSizeX
 - displays the current CardAnswerPrompt Size X*
- Brood::BroodUI::Button * m_btnCardAnswerPromptIncSizeX
 - clicking it will increase the CardAnswerPrompt Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptSizeYPrompt
 - holds the text CardAnswerPrompt Size y*
- Brood::BroodUI::Button * m_btnCardAnswerPromptDecSizeY
 - clicking it will decrease the CardAnswerPrompt Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerPromptSizeY
 - displays the current CardAnswerPrompt Size y*
- Brood::BroodUI::Button * m_btnCardAnswerPromptIncSizeY
 - clicking it will increase the CardAnswerPrompt Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerValueSizeXPrompt
 - holds the text CardAnswerValue position x*
- Brood::BroodUI::Button * m_btnCardAnswerValueDecSizeX
 - clicking it will decrease the CardAnswerValue Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerValueSizeX
 - displays the current CardAnswerValue Size X*
- Brood::BroodUI::Button * m_btnCardAnswerValueIncSizeX
 - clicking it will increase the CardAnswerValue Size x number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerValueSizeYPrompt
 - holds the text CardAnswerValue Size y*
- Brood::BroodUI::Button * m_btnCardAnswerValueDecSizeY
 - clicking it will decrease the CardAnswerValue Size y number by 5*

- clicking it will decrease the CardAnswerValue Size y number by 5*
- Brood::BroodUI::TextBox * m_txtCardAnswerValueSizeY
 displays the current CardAnswerValue Size y
- Brood::BroodUI::Button * m_btnCardAnswerValueIncSizeY
 clicking it will increase the CardAnswerValue Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitSizeXPrompt
 holds the text CardSubmit position x
- Brood::BroodUI::Button * m_btnCardSubmitDecSizeX
 clicking it will decrease the CardSubmit Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitSizeX
 displays the current CardSubmit Size X
- Brood::BroodUI::Button * m_btnCardSubmitIncSizeX
 clicking it will increase the CardSubmit Size x number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitSizeYPrompt
 holds the text CardSubmit Size y
- Brood::BroodUI::Button * m_btnCardSubmitDecSizeY
 clicking it will decrease the CardSubmit Size y number by 5
- Brood::BroodUI::TextBox * m_txtCardSubmitSizeY
 displays the current CardSubmit Size y
- Brood::BroodUI::Button * m_btnCardSubmitIncSizeY
 clicking it will increase the CardSubmit Size y number by 5
- Brood::BroodUI::TextBox * m_txtDeckNumPrompt
 holds the text Curr Deck Num
- Brood::BroodUI::Button * m_btnDecDeckNum
 clicking it will decrease the Curr Deck Num number by 1
- Brood::BroodUI::TextBox * m_txtDeckNum
 displays the current CardSubmit Size y
- Brood::BroodUI::Button * m_btnIncDeckNum
 clicking it will increase the Curr Deck Num number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardNumPrompt
 holds the text Curr card Num
- Brood::BroodUI::Button * m_btnDecCurrCardNum
 clicking it will decrease the Curr card number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardNum
 displays the current CardSubmit Size y
- Brood::BroodUI::Button * m_btnIncCurrCardNum
 clicking it will increase the Curr card Num number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardTimePrompt
 holds the text Curr card time
- Brood::BroodUI::Button * m_btnDecCurrCardTime
 clicking it will decrease the Curr card time number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardTime
 displays the current Curr card time
- Brood::BroodUI::Button * m_btnIncCurrCardTime
 clicking it will increase the Curr card Time number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardUpPrompt
 holds the text Curr card up
- Brood::BroodUI::Button * m_btnDecCurrCardUp
 clicking it will decrease the Curr card up number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardUp
 displays the current Curr card up

- Brood::BroodUI::Button * m_btnIncCurrCardUp
clicking it will increase the Curr card up number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardDownPrompt
holds the text Curr card down
- Brood::BroodUI::Button * m_btnDecCurrCardDown
clicking it will decrease the Curr card down number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardDown
displays the current Curr card down
- Brood::BroodUI::Button * m_btnIncCurrCardDown
clicking it will increase the Curr card down number by 1
- Brood::BroodUI::TextBox * m_txtCurrCardQuestionPrompt
holds the curr Card Question prompt
- Brood::BroodUI::TextBox * m_txtCurrCardQuestionInput
holds the question user enters
- Brood::BroodUI::TextBox * m_txtCurrCardAnswerPrompt
holds the curr Card Answer prompt
- Brood::BroodUI::TextBox * m_txtCurrCardAnswerInput
holds the question user enters
- Brood::BroodUI::Button * m_btnTurnCardAround

Additional Inherited Members

9.8.1 Constructor & Destructor Documentation

9.8.1.1 CardEditor()

```
CardEditor::CardEditor (
    Brood::Application::Components::GameManager * aGameData,
    sf::RectangleShape * aPanelPtr )
```

default constructor

Parameters

aGameData	pointer to the game data object
aPanelPtr	pointer to the panel body object

9.8.1.2 ~CardEditor()

```
CardEditor::~CardEditor ( )
```

default destructor

9.8.2 Member Function Documentation

9.8.2.1 Debugger()

```
void CardEditor::Debugger ( ) [virtual]  
  
debugger funciton  
  
@virtual
```

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.8.2.2 Draw()

```
void CardEditor::Draw (sf::RenderWindow & a_window) [virtual]  
  
Draw funciton.  
  
@virtual
```

Draws all the component to the screen

Parameters

a_window	reference to the render window
----------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.8.2.3 DrawCardDisplayBcompPanel()

```
void CardEditor::DrawCardDisplayBcompPanel (sf::RenderWindow & a_window) [private]
```

Draws the panel element of the Card Display back component setting.

It draws the following panel element: card back time prompt X-position, card back time prompt Y-position, card back time value X-position, card back time value Y-position, card question prompt X-position, card question prompt Y-position, card question value X-position, card question value Y-position, card answer prompt X-position, card answer prompt Y-position, card answer value X-position, card answer value Y-position, card submit X-position, card submit Y-position, card back time prompt X-size, card back time prompt Y-size, card back time value X-size, card back time value Y-size, card question prompt X-size, card question prompt Y-size, card question value X-size, card question value Y-size, card answer prompt X-size, card answer prompt Y-size, card answer value X-size, card answer value Y-size, card submit X-size, card submit Y-size

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.8.2.4 DrawCardDisplayFcompPanel()

```
void CardEditor::DrawCardDisplayFcompPanel (
    sf::RenderWindow & a_window ) [private]
```

Draws the panel element of the Card Display front component setting.

It draws the following panel element: card front time prompt X-position, card front time prompt Y-position, card front time value X-position, card front time value Y-position, card up prompt X-position, card up prompt Y-position, card up value X-position, card up value Y-position, card down prompt X-position, card down prompt Y-position, card down value X-position, card down value Y-position, card turn card X-position, card turn card Y-position card front time prompt X-size, card front time prompt Y-size, card front time value X-size, card front time value Y-size, card up prompt X-size, card up prompt Y-size, card up value X-size, card up value Y-size, card down prompt X-size, card down prompt Y-size, card down value X-size, card down value Y-size, card turn card X-size, card turn card Y-size

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.8.2.5 DrawCardDisplayPanel()

```
void CardEditor::DrawCardDisplayPanel (
    sf::RenderWindow & a_window ) [private]
```

Draws the panel element of the Card Display setting.

It draws the following panel element: card X-position, card Y-position, card X-size, card Y-Size, CardFrontBg filename texture, CardFrontBg filename texture

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.8.2.6 DrawCardInfoPanel()

```
void CardEditor::DrawCardInfoPanel (
    sf::RenderWindow & a_window ) [private]
```

Draws the panel element of the Card Display back component setting.

It draws the following panel element: Curr Deck idx prompt, Curr Card idx prompt, Curr Card Time prompt, Curr Card up prompt, Curr Card down prompt, Curr Card question prompt, Curr Card answer prompt, turn card around

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.8.2.7 InitializeCardDisplayBcompPanel()

```
void CardEditor::InitializeCardDisplayBcompPanel ( ) [private]
```

Initializes the panel element of the Card Display back component setting.

It initializes the following panel element: card back time prompt X-position, card back time prompt Y-position, card back time value X-position, card back time value Y-position, card question prompt X-position, card question prompt Y-position, card question value X-position, card question value Y-position, card answer prompt X-position, card answer prompt Y-position, card answer value X-position, card answer value Y-position, card submit X-position, card submit Y-position, card back time prompt X-size, card back time prompt Y-size, card back time value X-size, card back time value Y-size, card question prompt X-size, card question prompt Y-size, card question value X-size, card question value Y-size, card answer prompt X-size, card answer prompt Y-size, card answer value X-size, card answer value Y-size, card submit X-size, card submit Y-size

9.8.2.8 InitializeCardDisplayFcompPanel()

```
void CardEditor::InitializeCardDisplayFcompPanel ( ) [private]
```

Initializes the panel element of the Card Display front component setting.

It initializes the following panel element: card front time prompt X-position, card front time prompt Y-position, card front time value X-position, card front time value Y-position, card up prompt X-position, card up prompt Y-position, card up value X-position, card up value Y-position, card down prompt X-position, card down prompt Y-position, card down value X-position, card down value Y-position, card turn card X-position, card turn card Y-position card front time prompt X-size, card front time prompt Y-size, card front time value X-size, card front time value Y-size, card up prompt X-size, card up prompt Y-size, card up value X-size, card up value Y-size, card down prompt X-size, card down prompt Y-size, card down value X-size, card down value Y-size, card turn card X-size, card turn card Y-size

9.8.2.9 InitializeCardDisplayPanel()

```
void CardEditor::InitializeCardDisplayPanel ( ) [private]
```

Initializes the panel element of the Card Display setting.

It initializes the following panel element: card X-position, card Y-position, card X-size, card Y-Size, CardFrontBg filename texture, CardFrontBg filename texture

9.8.2.10 InitializeCardInfoPanel()

```
void CardEditor::InitializeCardInfoPanel ( ) [private]
```

Initializes the panel element of the Card Display back component setting.

It initializes the following panel element: Curr Deck Num prompt, Curr Card Num prompt, Curr Card Time prompt, Curr Card up prompt, Curr Card down prompt, Curr Card question prompt, Curr Card answer prompt, turn card around

9.8.2.11 InitailizeSettingSelectionDDI()

```
void CardEditor::InitailizeSettingSelectionDDI ( ) [private]
```

Initializes the setting Section dropdown input which is at the top of the setting pannel.

It has the 4 settings : Display card Setting, Display card front componet Setting, Display card back componet Setting Existing info Card Setting

9.8.2.12 InitializeWorkSpace()

```
void CardEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the board, setting section Dropdown Input.

Implements [Brood::Application::WorkSpace](#).

9.8.2.13 Update()

```
void CardEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.8.2.14 UpdateAllDisplayElement()

```
void CardEditor::UpdateAllDisplayElement ( ) [virtual]
```

updates the display element

@virtual

Implements [Brood::Application::WorkSpace](#).

9.8.2.15 UpdateCardBackFileTexture()

```
void CardEditor::UpdateCardBackFileTexture ( ) [private]
```

checks if the user interacted with the laod Card texture panel

Load Card Back texture file panel contains load Card Back texture file promt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\CardTexture\ as it loads them from there.

9.8.2.16 UpdateCardDisplayBcompPanel()

```
void CardEditor::UpdateCardDisplayBcompPanel ( ) [private]
```

Updates the panel element of the Card Display back component setting.

It Updates the folloing panel element: card back time prompt X-position, card back time prompt Y-position, card back time value X-position, card back time value Y-position, card question prompt X-position, card question prompt Y-position, card question value X-position, card question value Y-position, card answer prompt X-position, card answer prompt Y-position, card answer value X-position, card answer value Y-position, card submit X-position, card submit Y-position, card back time prompt X-size, card back time prompt Y-size, card back time value X-size, card back time value Y-size, card question prompt X-size, card question prompt Y-size, card question value X-size, card question value Y-size, card answer prompt X-size, card answer prompt Y-size, card answer value X-size, card answer value Y-size, card submit X-size, card submit Y-size

9.8.2.17 UpdateCardDisplayFcompPanel()

```
void CardEditor::UpdateCardDisplayFcompPanel ( ) [private]
```

Updates the panel element of the Card Display front component setting.

It Updates the folloing panel element: card front time prompt X-position, card front time prompt Y-position, card front time value X-position, card front time value Y-position, card up prompt X-position, card up prompt Y-position, card up value X-position, card up value Y-position, card down prompt X-position, card down prompt Y-position, card down value X-position, card down value Y-position, card turn card X-position, card turn card Y-position card front time prompt X-size, card front time prompt Y-size, card front time value X-size, card front time value Y-size, card up prompt X-size, card up prompt Y-size, card up value X-size, card up value Y-size, card down prompt X-size, card down prompt Y-size, card down value X-size, card down value Y-size, card turn card X-size, card turn card Y-size

TODO make it so that card component are bound inside

9.8.2.18 UpdateCardDisplayPanel()

```
void CardEditor::UpdateCardDisplayPanel ( ) [private]
```

Updates the panel element of the Card Display setting.

It Updates the following panel element: card X-position, card Y-position, card X-size, card Y-Size, CardFrontBg file-name texture, CardFrontBg filename texture

TODO make it so that when the display card pos changes the pos of its component also changes

9.8.2.19 UpdateCardFrontFileTexture()

```
void CardEditor::UpdateCardFrontFileTexture ( ) [private]
```

checks if the user interacted with the load Card texture panel

Load Card front texture file panel contains load Card front texture file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\CardTexture\ as it loads them from there.

9.8.2.20 UpdateCardInfoPanel()

```
void CardEditor::UpdateCardInfoPanel ( ) [private]
```

Updates the panel element of the Card Display back component setting.

It Updates the following panel element: Curr Deck Num prompt, Curr Card Num prompt, Curr Card Time prompt, Curr Card up prompt, Curr Card down prompt, Curr Card question prompt, Curr Card answer prompt, turn card around

9.8.2.21 UpdateCurrCardAnswerInput()

```
void CardEditor::UpdateCurrCardAnswerInput ( ) [private]
```

checks if the user interacted with the set Curr Card Answer panel

Curr Card Answer panel contains Curr Card Answer prompt textbox, and textbox to enter the Curr Card Answer.

Only the textbox to enter the Curr Card Answer is interactable

If the interactable textbox was pressed then it allows the user to set the game title

9.8.2.22 UpdateCurrCardInfoDown()

```
void CardEditor::UpdateCurrCardInfoDown ( ) [private]
```

checks if the user interacted with the current selected Card Down panel

Current selected Card index panel contains current selected Card Down prompt textbox, current selected Card Down value text box, button to increase the current selected Card Down, and button to decrease the current selected Card Down

Only the button to increase the current selected Card Down, and button to decrease the current selected Card Down are interactable

If the interactable button was pressed then the current selected Card Down is increased or decreased by 1. current selected Card index should be more than or equal to 0

9.8.2.23 UpdateCurrCardInfoTime()

```
void CardEditor::UpdateCurrCardInfoTime ( ) [private]
```

checks if the user interacted with the current selected Card time panel

Current selected Card index panel contains current selected Card time prompt textbox, current selected Card time value text box, button to increase the current selected Card time, and button to decrease the current selected Card time

Only the button to increase the current selected Card time, and button to decrease the current selected Card time are interactable

If the interactable button was pressed then the current selected Card time is increased or decreased by 1. current selected Card index should be more than or equal to 0

9.8.2.24 UpdateCurrCardInfoUp()

```
void CardEditor::UpdateCurrCardInfoUp ( ) [private]
```

checks if the user interacted with the current selected Card Up panel

Current selected Card index panel contains current selected Card Up prompt textbox, current selected Card Up value text box, button to increase the current selected Card Up, and button to decrease the current selected Card Up

Only the button to increase the current selected Card Up, and button to decrease the current selected Card Up are interactable

If the interactable button was pressed then the current selected Card Up is increased or decreased by 1. current selected Card index should be more than or equal to 0

9.8.2.25 UpdateCurrCardQuestionInput()

```
void CardEditor::UpdateCurrCardQuestionInput ( ) [private]
```

checks if the user interacted with the set Curr Card Question panel

Curr Card Question panel contains Curr Card Question prompt textbox, and textbox to enter the Curr Card Question.

Only the textbox to enter the Curr Card Question is interactable

If the interactable textbox was pressed then it allows the user to set the game title

9.8.2.26 UpdateCurrSelectedCardIdx()

```
void CardEditor::UpdateCurrSelectedCardIdx ( ) [private]
```

checks if the user interacted with the current selected Card index panel

Current selected Card index panel contains current selected Card index prompt textbox, current selected Card index value text box, button to increase the current selected Card index, and button to decrease the current selected Card index

Only the button to increase the current selected Card index, and button to decrease the current selected Card index are interactable

If the interactable button was pressed then the current selected Card index is increased or decreased by 1. current selected Card index should be more than or equal to 0 but less than minimum Card number

9.8.2.27 UpdateCurrSelectedDeckIdx()

```
void CardEditor::UpdateCurrSelectedDeckIdx ( ) [private]
```

checks if the user interacted with the current selected deck index panel

Current selected deck index panel contains current selected deck index prompt textbox, current selected deck index value text box, button to increase the current selected deck index, and button to decrease the current selected deck index

Only the button to increase the current selected deck index, and button to decrease the current selected deck index are interactable

If the interactable button was pressed then the current selected deck index is increased or decreased by 1. current selected deck index should be more than or equal to 0 but less than minimum deck number

9.8.2.28 UpdateSettingSelectionDDI()

```
void CardEditor::UpdateSettingSelectionDDI ( ) [private]
```

checks if the user interacted with setting Selection and updates accordingly

9.8.2.29 UpdateTurnCard()

```
void CardEditor::UpdateTurnCard ( ) [private]
```

checks if the user interacted with the turn card panel

turn card panel contains only turn card button to turn the display card.

Only the button to turn the display card is interactable.

If the interactable button was pressed then it allows the user to turn the display card.

9.8.3 Member Data Documentation

9.8.3.1 m_btnCardAnswerPromptDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptDecPosX [private]`

clicking it will decrease the CardAnswerPrompt Position x number by 5

9.8.3.2 m_btnCardAnswerPromptDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptDecPosY [private]`

clicking it will decrease the CardAnswerPrompt Position y number by 5

9.8.3.3 m_btnCardAnswerPromptDecSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptDecSizeX [private]`

clicking it will decrease the CardAnswerPrompt Size x number by 5

9.8.3.4 m_btnCardAnswerPromptDecSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptDecSizeY [private]`

clicking it will decrease the CardAnswerPrompt Size y number by 5

9.8.3.5 m_btnCardAnswerPromptIncPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptIncPosX [private]`

clicking it will increase the CardAnswerPrompt Position x number by 5

9.8.3.6 m_btnCardAnswerPromptIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptIncPosY [private]
```

clicking it will increase the CardAnswerPrompt Position y number by 5

9.8.3.7 m_btnCardAnswerPromptIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptIncSizeX [private]
```

clicking it will increase the CardAnswerPrompt Size x number by 5

9.8.3.8 m_btnCardAnswerPromptIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerPromptIncSizeY [private]
```

clicking it will increase the CardAnswerPrompt Size y number by 5

9.8.3.9 m_btnCardAnswerValueDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueDecPosX [private]
```

clicking it will decrease the CardAnswerValue Position x number by 5

9.8.3.10 m_btnCardAnswerValueDecPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueDecPosY [private]
```

clicking it will decrease the CardAnswerValue Position y number by 5

9.8.3.11 m_btnCardAnswerValueDecSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueDecSizeX [private]
```

clicking it will decrease the CardAnswerValue Size x number by 5

9.8.3.12 m_btnCardAnswerValueDecSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueDecSizeY [private]

clicking it will decrease the CardAnswerValue Size y number by 5

9.8.3.13 m_btnCardAnswerValueIncPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueIncPosX [private]

clicking it will increase the CardAnswerValue Position x number by 5

9.8.3.14 m_btnCardAnswerValueIncPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueIncPosY [private]

clicking it will increase the CardAnswerValue Position y number by 5

9.8.3.15 m_btnCardAnswerValueIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueIncSizeX [private]

clicking it will increase the CardAnswerValue Size x number by 5

9.8.3.16 m_btnCardAnswerValueIncSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardAnswerValueIncSizeY [private]

clicking it will increase the CardAnswerValue Size y number by 5

9.8.3.17 m_btnCardBackBgOpenFile

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackBgOpenFile [private]

clicking it will load the CardBack file

9.8.3.18 m_btnCardBackTimePromptDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptDecPosX [private]`

clicking it will decrease the CardBackTimePrompt Position x number by 5

9.8.3.19 m_btnCardBackTimePromptDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptDecPosY [private]`

clicking it will decrease the CardBackTimePrompt Position y number by 5

9.8.3.20 m_btnCardBackTimePromptDecSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptDecSizeX [private]`

clicking it will decrease the CardBackTimePrompt Size x number by 5

9.8.3.21 m_btnCardBackTimePromptDecSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptDecSizeY [private]`

clicking it will decrease the CardBackTimePrompt Size y number by 5

9.8.3.22 m_btnCardBackTimePromptIncPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptIncPosX [private]`

clicking it will increase the CardBackTimePrompt Position x number by 5

9.8.3.23 m_btnCardBackTimePromptIncPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptIncPosY [private]`

clicking it will increase the CardBackTimePrompt Position y number by 5

9.8.3.24 m_btnCardBackTimePromptIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptIncSizeX [private]

clicking it will increase the CardBackTimePrompt Size x number by 5

9.8.3.25 m_btnCardBackTimePromptIncSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimePromptIncSizeY [private]

clicking it will increase the CardBackTimePrompt Size y number by 5

9.8.3.26 m_btnCardBackTimeValueDecPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueDecPosX [private]

clicking it will decrease the CardBackTimeValue Position x number by 5

9.8.3.27 m_btnCardBackTimeValueDecPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueDecPosY [private]

clicking it will decrease the CardBackTimeValue Position y number by 5

9.8.3.28 m_btnCardBackTimeValueDecSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueDecSizeX [private]

clicking it will decrease the CardBackTimeValue Size x number by 5

9.8.3.29 m_btnCardBackTimeValueDecSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueDecSizeY [private]

clicking it will decrease the CardBackTimeValue Size y number by 5

9.8.3.30 m_btnCardBackTimeValueIncPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueIncPosX [private]`

clicking it will increase the CardBackTimeValue Position x number by 5

9.8.3.31 m_btnCardBackTimeValueIncPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueIncPosY [private]`

clicking it will increase the CardBackTimeValue Position y number by 5

9.8.3.32 m_btnCardBackTimeValueIncSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueIncSizeX [private]`

clicking it will increase the CardBackTimeValue Size x number by 5

9.8.3.33 m_btnCardBackTimeValueIncSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardBackTimeValueIncSizeY [private]`

clicking it will increase the CardBackTimeValue Size y number by 5

9.8.3.34 m_btnCardDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDecPosX [private]`

clicking it will decrease the Card Position x number by 5

9.8.3.35 m_btnCardDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDecPosY [private]`

clicking it will decrease the Card Position y number by 5

9.8.3.36 m_btnCardDecSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDecSizeX [private]`

clicking it will decrease the Card size x number by 1

9.8.3.37 m_btnCardDecSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDecSizeY [private]`

clicking it will decrease the Card size y number by 1

9.8.3.38 m_btnCardDownPromptDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptDecPosX [private]`

clicking it will decrease the CardDownPrompt Position x number by 5

9.8.3.39 m_btnCardDownPromptDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptDecPosY [private]`

clicking it will decrease the CardDownPrompt Position y number by 5

9.8.3.40 m_btnCardDownPromptDecSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptDecSizeX [private]`

clicking it will decrease the CardDownPrompt size x number by 5

9.8.3.41 m_btnCardDownPromptDecSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptDecSizeY [private]`

clicking it will decrease the CardDownPrompt size y number by 5

9.8.3.42 m_btnCardDownPromptIncPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptIncPosX [private]`

clicking it will increase the CardDownPrompt Position x number by 5

9.8.3.43 m_btnCardDownPromptIncPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptIncPosY [private]`

clicking it will increase the CardDownPrompt Position y number by 5

9.8.3.44 m_btnCardDownPromptIncSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptIncSizeX [private]`

clicking it will increase the CardDownPrompt size x number by 5

9.8.3.45 m_btnCardDownPromptIncSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownPromptIncSizeY [private]`

clicking it will increase the CardDownPrompt size y number by 5

9.8.3.46 m_btnCardDownValueDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueDecPosX [private]`

clicking it will decrease the CardDownValue Position x number by 5

9.8.3.47 m_btnCardDownValueDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueDecPosY [private]`

clicking it will decrease the CardDownValue Position y number by 5

9.8.3.48 m_btnCardDownValueDecSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueDecSizeX [private]

clicking it will decrease the CardDownValue size x number by 5

9.8.3.49 m_btnCardDownValueDecSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueDecSizeY [private]

clicking it will decrease the CardDownValue size y number by 5

9.8.3.50 m_btnCardDownValueIncPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueIncPosX [private]

clicking it will increase the CardDownValue Position x number by 5

9.8.3.51 m_btnCardDownValueIncPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueIncPosY [private]

clicking it will increase the CardDownValue Position y number by 5

9.8.3.52 m_btnCardDownValueIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueIncSizeX [private]

clicking it will increase the CardDownValue size x number by 5

9.8.3.53 m_btnCardDownValueIncSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardDownValueIncSizeY [private]

clicking it will increase the CardDownValue size y number by 5

9.8.3.54 m_btnCardFrontBgOpenFile

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontBgOpenFile [private]
```

clicking it will load the CardFront file

9.8.3.55 m_btnCardFrontTimePromptDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptDecPosX [private]
```

clicking it will decrease the CardFrontTimePrompt Position x number by 5

9.8.3.56 m_btnCardFrontTimePromptDecPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptDecPosY [private]
```

clicking it will decrease the CardFrontTimePrompt Position y number by 5

9.8.3.57 m_btnCardFrontTimePromptDecSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptDecSizeX [private]
```

clicking it will decrease the CardFrontTimePrompt size x number by 5

9.8.3.58 m_btnCardFrontTimePromptDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptDecSizeY [private]
```

clicking it will decrease the CardFrontTimePrompt size y number by 5

9.8.3.59 m_btnCardFrontTimePromptIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptIncPosX [private]
```

clicking it will increase the CardFrontTimePrompt Position x number by 5

9.8.3.60 m_btnCardFrontTimePromptIncPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptIncPosY [private]

clicking it will increase the CardFrontTimePrompt Position y number by 5

9.8.3.61 m_btnCardFrontTimePromptIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptIncSizeX [private]

clicking it will increase the CardFrontTimePrompt size x number by 5

9.8.3.62 m_btnCardFrontTimePromptIncSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimePromptIncSizeY [private]

clicking it will increase the CardFrontTimePrompt size y number by 5

9.8.3.63 m_btnCardFrontTimeValueDecPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueDecPosX [private]

clicking it will decrease the CardFrontTimeValue Position x number by 5

9.8.3.64 m_btnCardFrontTimeValueDecPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueDecPosY [private]

clicking it will decrease the CardFrontTimeValue Position y number by 5

9.8.3.65 m_btnCardFrontTimeValueDecSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueDecSizeX [private]

clicking it will decrease the CardFrontTimeValue size x number by 5

9.8.3.66 m_btnCardFrontTimeValueDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueDecSizeY [private]
```

clicking it will decrease the CardFrontTimeValue size y number by 5

9.8.3.67 m_btnCardFrontTimeValueIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueIncPosX [private]
```

clicking it will increase the CardFrontTimeValue Position x number by 5

9.8.3.68 m_btnCardFrontTimeValueIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueIncPosY [private]
```

clicking it will increase the CardFrontTimeValue Position y number by 5

9.8.3.69 m_btnCardFrontTimeValueIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueIncSizeX [private]
```

clicking it will increase the CardFrontTimeValue size x number by 5

9.8.3.70 m_btnCardFrontTimeValueIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardFrontTimeValueIncSizeY [private]
```

clicking it will increase the CardFrontTimeValue size y number by 5

9.8.3.71 m_btnCardIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardIncPosX [private]
```

clicking it will increase the Card Position x number by 5

9.8.3.72 m_btnCardIncPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardIncPosY [private]`

clicking it will increase the Card Position y number by 5

9.8.3.73 m_btnCardIncSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardIncSizeX [private]`

clicking it will increase the Card size x number by 1

9.8.3.74 m_btnCardIncSizeY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardIncSizeY [private]`

clicking it will increase the Card size y number by 1

9.8.3.75 m_btnCardQuestionPromptDecPosX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptDecPosX [private]`

clicking it will decrease the CardQuestionPrompt Position x number by 5

9.8.3.76 m_btnCardQuestionPromptDecPosY

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptDecPosY [private]`

clicking it will decrease the CardQuestionPrompt Position y number by 5

9.8.3.77 m_btnCardQuestionPromptDecSizeX

`Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptDecSizeX [private]`

clicking it will decrease the CardQuestionPrompt Size x number by 5

9.8.3.78 m_btnCardQuestionPromptDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptDecSizeY [private]
```

clicking it will decrease the CardQuestionPrompt Size y number by 5

9.8.3.79 m_btnCardQuestionPromptIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptIncPosX [private]
```

clicking it will increase the CardQuestionPrompt Position x number by 5

9.8.3.80 m_btnCardQuestionPromptIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptIncPosY [private]
```

clicking it will increase the CardQuestionPrompt Position y number by 5

9.8.3.81 m_btnCardQuestionPromptIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptIncSizeX [private]
```

clicking it will increase the CardQuestionPrompt Size x number by 5

9.8.3.82 m_btnCardQuestionPromptIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionPromptIncSizeY [private]
```

clicking it will increase the CardQuestionPrompt Size y number by 5

9.8.3.83 m_btnCardQuestionValueDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueDecPosX [private]
```

clicking it will decrease the CardQuestionValue Position x number by 5

9.8.3.84 m_btnCardQuestionValueDecPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueDecPosY [private]

clicking it will decrease the CardQuestionValue Position y number by 5

9.8.3.85 m_btnCardQuestionValueDecSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueDecSizeX [private]

clicking it will decrease the CardQuestionValue Size x number by 5

9.8.3.86 m_btnCardQuestionValueDecSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueDecSizeY [private]

clicking it will decrease the CardQuestionValue Size y number by 5

9.8.3.87 m_btnCardQuestionValueIncPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueIncPosX [private]

clicking it will increase the CardQuestionValue Position x number by 5

9.8.3.88 m_btnCardQuestionValueIncPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueIncPosY [private]

clicking it will increase the CardQuestionValue Position y number by 5

9.8.3.89 m_btnCardQuestionValueIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueIncSizeX [private]

clicking it will increase the CardQuestionValue Size x number by 5

9.8.3.90 m_btnCardQuestionValueIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardQuestionValueIncSizeY [private]
```

clicking it will increase the CardQuestionValue Size y number by 5

9.8.3.91 m_btnCardSubmitDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitDecPosX [private]
```

clicking it will decrease the CardSubmit Position x number by 5

9.8.3.92 m_btnCardSubmitDecPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitDecPosY [private]
```

clicking it will decrease the CardSubmit Position y number by 5

9.8.3.93 m_btnCardSubmitDecSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitDecSizeX [private]
```

clicking it will decrease the CardSubmit Size x number by 5

9.8.3.94 m_btnCardSubmitDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitDecSizeY [private]
```

clicking it will decrease the CardSubmit Size y number by 5

9.8.3.95 m_btnCardSubmitIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitIncPosX [private]
```

clicking it will increase the CardSubmit Position x number by 5

9.8.3.96 m_btnCardSubmitIncPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitIncPosY [private]

clicking it will increase the CardSubmit Size y number by 5

9.8.3.97 m_btnCardSubmitIncSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitIncSizeX [private]

clicking it will increase the CardSubmit Size x number by 5

9.8.3.98 m_btnCardSubmitIncSizeY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardSubmitIncSizeY [private]

clicking it will increase the CardSubmit Size y number by 5

9.8.3.99 m_btnCardTurnCardDecPosX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardDecPosX [private]

clicking it will decrease the CardTurnCard Position x number by 5

9.8.3.100 m_btnCardTurnCardDecPosY

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardDecPosY [private]

clicking it will decrease the CardTurnCard Position y number by 5

9.8.3.101 m_btnCardTurnCardDecSizeX

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardDecSizeX [private]

clicking it will decrease the CardTurnCard size x number by 5

9.8.3.102 m_btnCardTurnCardDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardDecSizeY [private]
```

clicking it will decrease the CardTurnCard size y number by 5

9.8.3.103 m_btnCardTurnCardIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardIncPosX [private]
```

clicking it will increase the CardTurnCard Position x number by 5

9.8.3.104 m_btnCardTurnCardIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardIncPosY [private]
```

clicking it will increase the CardTurnCard Position y number by 5

9.8.3.105 m_btnCardTurnCardIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardIncSizeX [private]
```

clicking it will increase the CardTurnCard size x number by 5

9.8.3.106 m_btnCardTurnCardIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardTurnCardIncSizeY [private]
```

clicking it will increase the CardTurnCard size y number by 5

9.8.3.107 m_btnCardUpPromptDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptDecPosX [private]
```

clicking it will decrease the CardUpPrompt Position x number by 5

9.8.3.108 m_btnCardUpPromptDecPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptDecPosY [private]
```

clicking it will decrease the CardUpPrompt Position y number by 5

9.8.3.109 m_btnCardUpPromptDecSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptDecSizeX [private]
```

clicking it will decrease the CardUpPrompt size x number by 5

9.8.3.110 m_btnCardUpPromptDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptDecSizeY [private]
```

clicking it will decrease the CardUpPrompt size y number by 5

9.8.3.111 m_btnCardUpPromptIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptIncPosX [private]
```

clicking it will increase the CardUpPrompt Position x number by 5

9.8.3.112 m_btnCardUpPromptIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptIncPosY [private]
```

clicking it will increase the CardUpPrompt Position y number by 5

9.8.3.113 m_btnCardUpPromptIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptIncSizeX [private]
```

clicking it will increase the CardUpPrompt size x number by 5

9.8.3.114 m_btnCardUpPromptIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpPromptIncSizeY [private]
```

clicking it will increase the CardUpPrompt size y number by 5

9.8.3.115 m_btnCardUpValueDecPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueDecPosX [private]
```

clicking it will decrease the CardUpValue Position x number by 5

9.8.3.116 m_btnCardUpValueDecPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueDecPosY [private]
```

clicking it will decrease the CardUpValue Position y number by 5

9.8.3.117 m_btnCardUpValueDecSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueDecSizeX [private]
```

clicking it will decrease the CardUpValue size x number by 5

9.8.3.118 m_btnCardUpValueDecSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueDecSizeY [private]
```

clicking it will decrease the CardUpValue size y number by 5

9.8.3.119 m_btnCardUpValueIncPosX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueIncPosX [private]
```

clicking it will increase the CardUpValue Position x number by 5

9.8.3.120 m_btnCardUpValueIncPosY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueIncPosY [private]
```

clicking it will increase the CardUpValue Position y number by 5

9.8.3.121 m_btnCardUpValueIncSizeX

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueIncSizeX [private]
```

clicking it will increase the CardUpValue size x number by 5

9.8.3.122 m_btnCardUpValueIncSizeY

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnCardUpValueIncSizeY [private]
```

clicking it will increase the CardUpValue size y number by 5

9.8.3.123 m_btnDecCurrCardDown

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnDecCurrCardDown [private]
```

clicking it will decrease the Curr card down number by 1

9.8.3.124 m_btnDecCurrCardNum

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnDecCurrCardNum [private]
```

clicking it will decrease the Curr card number by 1

9.8.3.125 m_btnDecCurrCardTime

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnDecCurrCardTime [private]
```

clicking it will decrease the Curr card time number by 1

9.8.3.126 m_btnDecCurrCardUp

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnDecCurrCardUp [private]
```

clicking it will decrease the Curr card up number by 1

9.8.3.127 m_btnDecDeckNum

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnDecDeckNum [private]
```

clicking it will decrease the Curr Deck Num number by 1

9.8.3.128 m_btnIncCurrCardDown

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnIncCurrCardDown [private]
```

clicking it will increase the Curr card down number by 1

9.8.3.129 m_btnIncCurrCardNum

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnIncCurrCardNum [private]
```

clicking it will increase the Curr card Num number by 1

9.8.3.130 m_btnIncCurrCardTime

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnIncCurrCardTime [private]
```

clicking it will increase the Curr card Time number by 1

9.8.3.131 m_btnIncCurrCardUp

```
Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnIncCurrCardUp [private]
```

clicking it will increase the Curr card up number by 1

9.8.3.132 m_btnIncDeckNum

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnIncDeckNum [private]

clicking it will increase the Curr Deck Num number by 1

9.8.3.133 m_btnTurnCardAround

Brood::BroodUI::Button* Brood::Application::CardEditor::m_btnTurnCardAround [private]

9.8.3.134 m_ddiSettingSelection

Brood::BroodUI::DropDownInput* Brood::Application::CardEditor::m_ddiSettingSelection [private]

holds which catagory of setting the displayed user can choose between: Display card Setting, Display card front componet Setting, Display card back componet Setting Existing info Card Setting

9.8.3.135 m_panelBodyPtr

sf::RectangleShape* Brood::Application::CardEditor::m_panelBodyPtr [private]

holds the panel body

9.8.3.136 m_selectedSettingIdx

unsigned Brood::Application::CardEditor::m_selectedSettingIdx [private]

9.8.3.137 m_txtCardAnswerPromptPosX

Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptPosX [private]

displays the current CardAnswerPrompt Position X

9.8.3.138 m_txtCardAnswerPromptPosXPrompt

Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptPosXPrompt [private]

holds the text CardAnswerPrompt position x

9.8.3.139 m_txtCardAnswerPromptPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptPosY [private]
```

displays the current CardAnswerPrompt Position y

9.8.3.140 m_txtCardAnswerPromptPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptPosYPrompt [private]
```

holds the text CardAnswerPrompt Position y

9.8.3.141 m_txtCardAnswerPromptSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptSizeX [private]
```

displays the current CardAnswerPrompt Size X

9.8.3.142 m_txtCardAnswerPromptSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptSizeXPrompt  
[private]
```

holds the text CardAnswerPrompt position x

9.8.3.143 m_txtCardAnswerPromptSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptSizeY [private]
```

displays the current CardAnswerPrompt Size y

9.8.3.144 m_txtCardAnswerPromptSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerPromptSizeYPrompt  
[private]
```

holds the text CardAnswerPrompt Size y

9.8.3.145 m_txtCardAnswerValuePosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValuePosX [private]
```

displays the current CardAnswerValue Position X

9.8.3.146 m_txtCardAnswerValuePosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValuePosXPrompt [private]
```

holds the text CardAnswerValue position x

9.8.3.147 m_txtCardAnswerValuePosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValuePosY [private]
```

displays the current CardAnswerValue Position y

9.8.3.148 m_txtCardAnswerValuePosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValuePosYPrompt [private]
```

holds the text CardAnswerValue Position y

9.8.3.149 m_txtCardAnswerValueSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValueSizeX [private]
```

displays the current CardAnswerValue Size X

9.8.3.150 m_txtCardAnswerValueSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValueSizeXPrompt [private]
```

holds the text CardAnswerValue position x

9.8.3.151 m_txtCardAnswerValueSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValueSizeY [private]
```

displays the current CardAnswerValue Size y

9.8.3.152 m_txtCardAnswerValueSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardAnswerValueSizeYPrompt [private]
```

holds the text CardAnswerValue Size y

9.8.3.153 m_txtCardBackBgFileInput

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackBgFileInput [private]
```

holds the file name user enters

9.8.3.154 m_txtCardBackBgFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackBgFileNamePrompt [private]
```

holds the CardBack texture filename prompt

9.8.3.155 m_txtCardBackTimePromptPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptPosX [private]
```

displays the current CardBackTimePrompt Position X

9.8.3.156 m_txtCardBackTimePromptPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptPosXPrompt [private]
```

holds the text CardBackTimePrompt position x

9.8.3.157 m_txtCardBackTimePromptPosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptPosY [private]`

displays the current CardBackTimePrompt Position y

9.8.3.158 m_txtCardBackTimePromptPosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptPosYPrompt [private]`

holds the text CardBackTimePrompt Position y

9.8.3.159 m_txtCardBackTimePromptSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptSizeX [private]`

displays the current CardBackTimePrompt Size X

9.8.3.160 m_txtCardBackTimePromptSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptSizeXPrompt [private]`

holds the text CardBackTimePrompt position x

9.8.3.161 m_txtCardBackTimePromptSizeY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptSizeY [private]`

displays the current CardBackTimePrompt Size y

9.8.3.162 m_txtCardBackTimePromptSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimePromptSizeYPrompt [private]`

holds the text CardBackTimePrompt Size y

9.8.3.163 m_txtCardBackTimeValuePosX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValuePosX [private]`

displays the current CardBackTimeValue Position X

9.8.3.164 m_txtCardBackTimeValuePosXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValuePosXPrompt [private]`

holds the text CardBackTimeValue position x

9.8.3.165 m_txtCardBackTimeValuePosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValuePosY [private]`

displays the current CardBackTimeValue Position y

9.8.3.166 m_txtCardBackTimeValuePosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValuePosYPrompt [private]`

holds the text CardBackTimeValue Position y

9.8.3.167 m_txtCardBackTimeValueSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValueSizeX [private]`

displays the current CardBackTimeValue Size X

9.8.3.168 m_txtCardBackTimeValueSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValueSizeXPrompt [private]`

holds the text CardBackTimeValue position x

9.8.3.169 m_txtCardBackTimeValueSizeY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValueSizeY [private]`

displays the current CardBackTimeValue Size y

9.8.3.170 m_txtCardBackTimeValueSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardBackTimeValueSizeYPrompt [private]`

holds the text CardBackTimeValue Size y

9.8.3.171 m_txtCardDownPromptPosX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptPosX [private]`

displays the current CardDownPrompt Position X

9.8.3.172 m_txtCardDownPromptPosXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptPosXPrompt [private]`

holds the text CardDownPrompt position x

9.8.3.173 m_txtCardDownPromptPosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptPosY [private]`

displays the current CardDownPrompt Position y

9.8.3.174 m_txtCardDownPromptPosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptPosYPrompt [private]`

holds the text CardDownPrompt Position y

9.8.3.175 m_txtCardDownPromptSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptSizeX [private]
```

displays the current CardDownPrompt size X

9.8.3.176 m_txtCardDownPromptSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptSizeXPrompt [private]
```

holds the text CardDownPrompt size x

9.8.3.177 m_txtCardDownPromptSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptSizeY [private]
```

displays the current CardDownPrompt size y

9.8.3.178 m_txtCardDownPromptSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownPromptSizeYPrompt [private]
```

holds the text CardDownPrompt size y

9.8.3.179 m_txtCardDownValuePosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValuePosX [private]
```

displays the current CardDownValue Position X

9.8.3.180 m_txtCardDownValuePosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValuePosXPrompt [private]
```

holds the text CardDownValue position x

9.8.3.181 m_txtCardDownValuePosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValuePosY [private]
```

displays the current CardDownValue Position y

9.8.3.182 m_txtCardDownValuePosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValuePosYPrompt [private]
```

holds the text CardDownValue Position y

9.8.3.183 m_txtCardDownValueSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValueSizeX [private]
```

displays the current CardDownValue size X

9.8.3.184 m_txtCardDownValueSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValueSizeXPrompt [private]
```

holds the text CardDownValue size x

9.8.3.185 m_txtCardDownValueSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValueSizeY [private]
```

displays the current CardDownValue size y

9.8.3.186 m_txtCardDownValueSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardDownValueSizeYPrompt [private]
```

holds the text CardDownValue size y

9.8.3.187 m_txtCardFrontBgFileInput

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontBgFileInput [private]
```

holds the file name user enters

9.8.3.188 m_txtCardFrontBgFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontBgFileNamePrompt [private]
```

holds the CardFront texture filename prompt

9.8.3.189 m_txtCardFrontTimePromptPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptPosX [private]
```

displays the current CardFrontTimePrompt Position X

9.8.3.190 m_txtCardFrontTimePromptPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptPosXPrompt [private]
```

holds the text CardFrontTimePrompt position x

9.8.3.191 m_txtCardFrontTimePromptPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptPosY [private]
```

displays the current CardFrontTimePrompt Position y

9.8.3.192 m_txtCardFrontTimePromptPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptPosYPrompt [private]
```

holds the text CardFrontTimePrompt Position y

9.8.3.193 m_txtCardFrontTimePromptSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptSizeX [private]`

displays the current CardFrontTimePrompt size X

9.8.3.194 m_txtCardFrontTimePromptSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptSizeXPrompt [private]`

holds the text CardFrontTimePrompt size x

9.8.3.195 m_txtCardFrontTimePromptSizeY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptSizeY [private]`

displays the current CardFrontTimePrompt size y

9.8.3.196 m_txtCardFrontTimePromptSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimePromptSizeYPrompt [private]`

holds the text CardFrontTimePrompt size y

9.8.3.197 m_txtCardFrontTimeValuePosX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValuePosX [private]`

displays the current CardFrontTimeValue Position X

9.8.3.198 m_txtCardFrontTimeValuePosXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValuePosXPrompt [private]`

holds the text CardFrontTimeValue position x

9.8.3.199 m_txtCardFrontTimeValuePosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValuePosY [private]`

displays the current CardFrontTimeValue Position y

9.8.3.200 m_txtCardFrontTimeValuePosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValuePosYPrompt [private]`

holds the text CardFrontTimeValue Position y

9.8.3.201 m_txtCardFrontTimeValueSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValueSizeX [private]`

displays the current CardFrontTimeValue size X

9.8.3.202 m_txtCardFrontTimeValueSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValueSizeXPrompt [private]`

holds the text CardFrontTimeValue size x

9.8.3.203 m_txtCardFrontTimeValueSizeY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValueSizeY [private]`

displays the current CardFrontTimeValue size y

9.8.3.204 m_txtCardFrontTimeValueSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardFrontTimeValueSizeYPrompt [private]`

holds the text CardFrontTimeValue size y

9.8.3.205 m_txtCardPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardPosX [private]
```

displays the current Card Position X

9.8.3.206 m_txtCardPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardPosXPrompt [private]
```

holds the text Card position x

9.8.3.207 m_txtCardPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardPosY [private]
```

displays the current Card Position y

9.8.3.208 m_txtCardPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardPosYPrompt [private]
```

holds the text Card Position y

9.8.3.209 m_txtCardQuestionPromptPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptPosX [private]
```

displays the current CardQuestionPrompt Position X

9.8.3.210 m_txtCardQuestionPromptPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptPosXPrompt [private]
```

holds the text CardQuestionPrompt position x

9.8.3.211 m_txtCardQuestionPromptPosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptPosY [private]`

displays the current CardQuestionPrompt Position y

9.8.3.212 m_txtCardQuestionPromptPosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptPosYPrompt [private]`

holds the text CardQuestionPrompt Position y

9.8.3.213 m_txtCardQuestionPromptSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptSizeX [private]`

displays the current CardQuestionPrompt Size X

9.8.3.214 m_txtCardQuestionPromptSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptSizeXPrompt [private]`

holds the text CardQuestionPrompt position x

9.8.3.215 m_txtCardQuestionPromptSizeY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptSizeY [private]`

displays the current CardQuestionPrompt Size y

9.8.3.216 m_txtCardQuestionPromptSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionPromptSizeYPrompt [private]`

holds the text CardQuestionPrompt Size y

9.8.3.217 m_txtCardQuestionValuePosX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValuePosX [private]`

displays the current CardQuestionValue Position X

9.8.3.218 m_txtCardQuestionValuePosXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValuePosXPrompt [private]`

holds the text CardQuestionValue position x

9.8.3.219 m_txtCardQuestionValuePosY

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValuePosY [private]`

displays the current CardQuestionValue Position y

9.8.3.220 m_txtCardQuestionValuePosYPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValuePosYPrompt [private]`

holds the text CardQuestionValue Position y

9.8.3.221 m_txtCardQuestionValueSizeX

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValueSizeX [private]`

displays the current CardQuestionValue Size X

9.8.3.222 m_txtCardQuestionValueSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValueSizeXPrompt [private]`

holds the text CardQuestionValue position x

9.8.3.223 m_txtCardQuestionValueSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValueSizeY [private]
```

displays the current CardQuestionValue Size y

9.8.3.224 m_txtCardQuestionValueSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardQuestionValueSizeYPrompt [private]
```

holds the text CardQuestionValue Size y

9.8.3.225 m_txtCardSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSizeX [private]
```

displays the current Card size X

9.8.3.226 m_txtCardSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSizeXPrompt [private]
```

holds the text Card size x

9.8.3.227 m_txtCardSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSizeY [private]
```

displays the current Card size y

9.8.3.228 m_txtCardSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSizeYPrompt [private]
```

holds the text Card size y

9.8.3.229 m_txtCardSubmitPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitPosX [private]
```

displays the current CardSubmit Position X

9.8.3.230 m_txtCardSubmitPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitPosXPrompt [private]
```

holds the text CardSubmit position x

9.8.3.231 m_txtCardSubmitPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitPosY [private]
```

displays the current CardSubmit Position y

9.8.3.232 m_txtCardSubmitPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitPosYPrompt [private]
```

holds the text CardSubmit Position y

9.8.3.233 m_txtCardSubmitSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitSizeX [private]
```

displays the current CardSubmit Size X

9.8.3.234 m_txtCardSubmitSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitSizeXPrompt [private]
```

holds the text CardSubmit position x

9.8.3.235 m_txtCardSubmitSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitSizeY [private]
```

displays the current CardSubmit Size y

9.8.3.236 m_txtCardSubmitSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardSubmitSizeYPrompt [private]
```

holds the text CardSubmit Size y

9.8.3.237 m_txtCardTurnCardPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardPosX [private]
```

displays the current CardTurnCard Position X

9.8.3.238 m_txtCardTurnCardPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardPosXPrompt [private]
```

holds the text CardTurnCard position x

9.8.3.239 m_txtCardTurnCardPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardPosY [private]
```

displays the current CardTurnCard Position y

9.8.3.240 m_txtCardTurnCardPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardPosYPrompt [private]
```

holds the text CardTurnCard Position y

9.8.3.241 m_txtCardTurnCardSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardSizeX [private]
```

displays the current CardTurnCard size X

9.8.3.242 m_txtCardTurnCardSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardSizeXPrompt [private]
```

holds the text CardTurnCard size x

9.8.3.243 m_txtCardTurnCardSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardSizeY [private]
```

displays the current CardTurnCard size y

9.8.3.244 m_txtCardTurnCardSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardTurnCardSizeYPrompt [private]
```

holds the text CardTurnCard size y

9.8.3.245 m_txtCardUpPromptPosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptPosX [private]
```

displays the current CardUpPrompt Position X

9.8.3.246 m_txtCardUpPromptPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptPosXPrompt [private]
```

holds the text CardUpPrompt position x

9.8.3.247 m_txtCardUpPromptPosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptPosY [private]
```

displays the current CardUpPrompt Position y

9.8.3.248 m_txtCardUpPromptPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptPosYPrompt [private]
```

holds the text CardUpPrompt Position y

9.8.3.249 m_txtCardUpPromptSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptSizeX [private]
```

displays the current CardUpPrompt size X

9.8.3.250 m_txtCardUpPromptSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptSizeXPrompt [private]
```

holds the text CardUpPrompt size x

9.8.3.251 m_txtCardUpPromptSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptSizeY [private]
```

displays the current CardUpPrompt size y

9.8.3.252 m_txtCardUpPromptSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpPromptSizeYPrompt [private]
```

holds the text CardUpPrompt size y

9.8.3.253 m_txtCardUpValuePosX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValuePosX [private]
```

displays the current CardUpValue Position X

9.8.3.254 m_txtCardUpValuePosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValuePosXPrompt [private]
```

holds the text CardUpValue position x

9.8.3.255 m_txtCardUpValuePosY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValuePosY [private]
```

displays the current CardUpValue Position y

9.8.3.256 m_txtCardUpValuePosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValuePosYPrompt [private]
```

holds the text CardUpValue Position y

9.8.3.257 m_txtCardUpValueSizeX

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValueSizeX [private]
```

displays the current CardUpValue size X

9.8.3.258 m_txtCardUpValueSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValueSizeXPrompt [private]
```

holds the text CardUpValue size x

9.8.3.259 m_txtCardUpValueSizeY

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValueSizeY [private]
```

displays the current CardUpValue size y

9.8.3.260 m_txtCardUpValueSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCardUpValueSizeYPrompt [private]
```

holds the text CardUpValue size y

9.8.3.261 m_txtCurrCardAnswerInput

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardAnswerInput [private]
```

holds the question user enters

9.8.3.262 m_txtCurrCardAnswerPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardAnswerPrompt [private]
```

holds the curr Card Answer prompt

9.8.3.263 m_txtCurrCardDown

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardDown [private]
```

displays the current Curr card down

9.8.3.264 m_txtCurrCardDownPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardDownPrompt [private]
```

holds the text Curr card down

9.8.3.265 m_txtCurrCardNum

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardNum [private]
```

displays the current CardSubmit Size y

9.8.3.266 m_txtCurrCardNumPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardNumPrompt [private]
```

holds the text Curr card Num

9.8.3.267 m_txtCurrCardQuestionInput

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardQuestionInput [private]
```

holds the question user enters

9.8.3.268 m_txtCurrCardQuestionPrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardQuestionPrompt [private]
```

holds the curr Card Question prompt

9.8.3.269 m_txtCurrCardTime

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardTime [private]
```

displays the current Curr card time

9.8.3.270 m_txtCurrCardTimePrompt

```
Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardTimePrompt [private]
```

holds the text Curr card time

9.8.3.271 m_txtCurrCardUp

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardUp [private]`

displays the current Curr card up

9.8.3.272 m_txtCurrCardUpPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtCurrCardUpPrompt [private]`

holds the text Curr card up

9.8.3.273 m_txtDeckNum

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtDeckNum [private]`

displays the current CardSubmit Size y

9.8.3.274 m_txtDeckNumPrompt

`Brood::BroodUI::TextBox* Brood::Application::CardEditor::m_txtDeckNumPrompt [private]`

holds the text Curr Deck Num

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardEditor.cpp](#)

9.9 CardEditor Class Reference

It allows the users to tweek the card setting.

```
#include "CardEditor.h"
```

9.9.1 Detailed Description

It allows the users to tweek the card setting.

It is a container that bundels elements that is present in [CardEditor](#) workspace

Derived form [Workspace](#) class

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardEditor.h](#)

9.10 Brood::Application::Components::CardInfo Class Reference

```
#include <CardInfo.h>
```

Public Member Functions

- `CardInfo` (unsigned `a_time`=0, unsigned `a_correctNumSteps`=0, unsigned `a_incorrectNumSteps`=0, std::string `a_question`="", std::string `a_correctAnswer`="")

Default Constructor.
- `~CardInfo` ()

Default destructor.
- `CardInfo` (const `CardInfo` &`a_other`)

copy constructor
- `CardInfo & operator=` (const `CardInfo` &`a_other`)

assignment operator
- `void InitializeCard` (`Brood::Application::Data::ST_CardInfoPrefabData` &`a_cardinfoData`)

initializes the card info with the passed data
- `Brood::Application::Data::ST_CardInfoPrefabData GetDataToSave` ()

creates and returns cardInfo data struct
- `unsigned GetTime` ()

Getter funciton to get time to complete the question.
- `unsigned GetCorrectNumSteps` ()

getter funciton to get correct number of steps
- `unsigned GetIncorrectNumSteps` ()

Getter funciton to get incorrect number of steps.
- `std::string GetQuestion` ()

Getter function to get question.
- `std::string GetCorrectAnswer` ()

Getter funciton to get correct answer.
- `void SetTime` (unsigned `a_time`)

Setter funciton to set time to complete the question.
- `void SetCorrectNumSteps` (unsigned `a_correctNumSteps`)

setter funciton to get correct number of steps
- `void SetIncorrectNumSteps` (unsigned `a_incorrectNumSteps`)

Setter funciton to get incorrect number of steps.
- `void SetQuestion` (std::string `a_question`)

Setter function to get question.
- `void SetCorrectAnswer` (std::string `a_correctAnswer`)

setter funciton to get correct answer

Private Attributes

- `unsigned m_time`

the time the player has to answer the question
- `unsigned m_correctNumSteps`

number steps the player move forward if answered correctly
- `unsigned m_incorrectNumSteps`

number steps the player move backward if answered incorrectly
- `std::string m_question`

stores the question
- `std::string m_correctAnswer`

9.10.1 Constructor & Destructor Documentation

9.10.1.1 CardInfo() [1/2]

```
Brood::Application::Components::CardInfo::CardInfo (
    unsigned a_time = 0,
    unsigned a_correctNumSteps = 0,
    unsigned a_incorrectNumSteps = 0,
    std::string a_question = "",
    std::string a_correctAnswer = "" )
```

Default Constructor.

9.10.1.2 ~CardInfo()

```
Brood::Application::Components::CardInfo::~CardInfo ( )
```

Default destructor.

9.10.1.3 CardInfo() [2/2]

```
Brood::Application::Components::CardInfo::CardInfo (
    const CardInfo & a_other )
```

copy constructor

Parameters

a_other	reference to the cardinfo which is used to copy the data form
---------	---

9.10.2 Member Function Documentation

9.10.2.1 GetCorrectAnswer()

```
std::string Brood::Application::Components::CardInfo::GetCorrectAnswer ( )
```

Getter funciton to get correct answer.

Returns

correct answer

9.10.2.2 GetCorrectNumSteps()

```
unsigned Brood::Application::Components::CardInfo::GetCorrectNumSteps ( )
```

getter funciton to get correct number of steps

Returns

number steps the player move forward if answered correctly

9.10.2.3 GetDataToSave()

```
Brood::Application::Data::ST_CardInfoPrefabData Brood::Application::Components::CardInfo::GetDataToSave ( )
```

creates and returns cardInfo data struct

Returns

cardInfo data struct with the cardInfo data in it

9.10.2.4 GetIncorrectNumSteps()

```
unsigned Brood::Application::Components::CardInfo::GetIncorrectNumSteps ( )
```

Getter funciton to get incorrect number of steps.

Returns

number steps the player move backward if answered incorrectly

9.10.2.5 GetQuestion()

```
std::string Brood::Application::Components::CardInfo::GetQuestion ( )
```

Getter function to get question.

Returns

question

9.10.2.6 GetTime()

```
unsigned Brood::Application::Components::CardInfo::GetTime ( )
```

Getter funciton to get time to complete the question.

Returns

the time the player has to answer the question

9.10.2.7 InitializeCard()

```
void Brood::Application::Components::CardInfo::InitializeCard (
    Brood::Application::Data::ST_CardInfoPrefabData & a_cardinfoData )
```

initializes the card info with the passed data

Parameters

a_cardinfoData	reference of the card info data
----------------	---------------------------------

9.10.2.8 operator=()

```
Brood::Application::Components::CardInfo & Brood::Application::Components::CardInfo::operator=
(
    const CardInfo & a_other )
```

assignment operator

Parameters

a_other	reference to the cardinfo which is used to copy the data form
---------	---

Returns

pointer to this cardinfo

9.10.2.9 SetCorrectAnswer()

```
void Brood::Application::Components::CardInfo::SetCorrectAnswer (
    std::string a_correctAnswer )
```

setter funciton to get correct answer

Parameters

<i>a_correctAnswer</i>	correct answer
------------------------	----------------

9.10.2.10 SetCorrectNumSteps()

```
void Brood::Application::Components::CardInfo::SetCorrectNumSteps (  
    unsigned a_correctNumSteps )
```

setter funciton to get correct number of steps

Parameters

<i>a_correctNumSteps</i>	number steps the player move forward if answered correctly
--------------------------	--

9.10.2.11 SetIncorrectNumSteps()

```
void Brood::Application::Components::CardInfo::SetIncorrectNumSteps (   
    unsigned a_incorrectNumSteps )
```

Setter funciton to get incorrect number of steps.

Parameters

<i>a_incorrectNumSteps</i>	number steps the player move backward if answered incorrectly
----------------------------	---

9.10.2.12 SetQuestion()

```
void Brood::Application::Components::CardInfo::SetQuestion (   
    std::string a_question )
```

Setter function to get question.

Returns

a_question question

9.10.2.13 SetTime()

```
void Brood::Application::Components::CardInfo::SetTime (   
    unsigned a_time )
```

Setter funciton to set time to complete the question.

Parameters

<code>a_time</code>	the time the player has to answer the question
---------------------	--

9.10.3 Member Data Documentation

9.10.3.1 `m_correctAnswer`

```
std::string Brood::Application::Components::CardInfo::m_correctAnswer [private]
```

9.10.3.2 `m_correctNumSteps`

```
unsigned Brood::Application::Components::CardInfo::m_correctNumSteps [private]
```

number steps the player move forward if answered correctly

9.10.3.3 `m_incorrectNumSteps`

```
unsigned Brood::Application::Components::CardInfo::m_incorrectNumSteps [private]
```

number steps the player move backward if answered incorrectly

9.10.3.4 `m_question`

```
std::string Brood::Application::Components::CardInfo::m_question [private]
```

stores the question

9.10.3.5 `m_time`

```
unsigned Brood::Application::Components::CardInfo::m_time [private]
```

the time the player has to answer the question

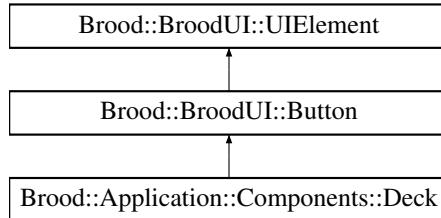
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardInfo.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardInfo.cpp](#)

9.11 Brood::Application::Components::Deck Class Reference

```
#include <Deck.h>
```

Inheritance diagram for Brood::Application::Components::Deck:



Public Member Functions

- `Deck (Brood::BroodUI::UIElement *a_parentPtr=nullptr)`
Default Constructor.
- `~Deck ()`
destructor
- `Deck (const Brood::Application::Components::Deck &a_other)`
copy constructor
- `Deck & operator= (const Brood::Application::Components::Deck &a_other)`
assignment operator
- `void InitializeDeck (Brood::Application::Data::ST_DeckPrefabData &a_deckData)`
initializes the deck with the passed data
- `Brood::Application::Data::ST_DeckPrefabData GetDataToSave ()`
creates and returns deck data struct
- `void SaveDataToFile (Brood::Application::FileAccess *a_fileAccessPtr, std::string a_gameTitle, unsigned a_idx)`
initializes the deck with the passed data
- `void LoadDataFromFile (Brood::Application::FileAccess *a_fileAccessPtr)`
loads the deck and its path data from passed file
- `const std::vector< Brood::Application::Components::CardInfo * > & GetCardList () const`
Getter function to get the cardList.
- `Brood::Application::Components::CardInfo * GetCardPtrAtIdx (unsigned a_cardIdx)`
Gets a card info pointer to the passed index.
- `std::string GetCardInitFileName ()`
gets card Init file name
- `std::string GetCardInitFilePath ()`
gets card Init file path
- `unsigned GetCurrActiveCardIdx ()`
gets card current active card index
- `CardInfo * GetCurrActiveCardPtr ()`
- `void SetCurrActiveCardIdx (unsigned a_currActiveCardIdx)`
sets card current active card index
- `void SetCardListSize (unsigned a_cardListSize)`
resizes the card list
- `bool LoadCardFromInitFile (std::string a_fileInitPath)`
loads cards from a card Init file path

- bool [SaveCardToInitFile](#) (std::string a_fileInitPath)
saves cards to a card Init file path
- const [CardInfo DealCard](#) ()
Function to deal the next undeal card.
- void [Shuffel](#) ()
shuffles the deck
- void [ResetDeck](#) ()
resets the m_currUndealt card index to zero and suffels the deck
- void [AddCardInfoToDeck](#) (Brood::Application::Components::CardInfo a_cardToAdd)
adds the passed card info into the list

Private Attributes

- std::vector< [Brood::Application::Components::CardInfo](#) * > m_cardList
- unsigned m_currUndealtCardIdx
- std::string m_cardInitFilePath
- std::string m_cardInitFileName
- unsigned m_currActiveCardIdx

Additional Inherited Members

9.11.1 Constructor & Destructor Documentation

9.11.1.1 Deck() [1/2]

```
Brood::Application::Components::Deck::Deck (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr )
```

Default Constructor.

Initializes the [Deck](#) object

Parameters

<i>m_parentPtr</i>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------	---

9.11.1.2 ~Deck()

```
Brood::Application::Components::Deck::~Deck ( )
```

destructor

9.11.1.3 Deck() [2/2]

```
Brood::Application::Components::Deck::Deck (
    const Brood::Application::Components::Deck & a_other )
```

copy constructor

Parameters

<i>a_other</i>	reference to the the deck object that is being copied from
----------------	--

9.11.2 Member Function Documentation

9.11.2.1 AddCardInfoToDeck()

```
void Brood::Application::Components::Deck::AddCardInfoToDeck (
    Brood::Application::Components::CardInfo a_cardToAdd )
```

adds the passed card info into the list

Parameters

<i>a_cardToAdd</i>	card to add to the deck
--------------------	-------------------------

9.11.2.2 DealCard()

```
const Brood::Application::Components::CardInfo Brood::Application::Components::Deck::DealCard
( )
```

Function to deal the next undeal card.

If all the card are dealt then it resets the dealt and shuffles it

Returns

a cardInfo object with the dealt card

9.11.2.3 GetCardInitFileName()

```
std::string Brood::Application::Components::Deck::GetCardInitFileName ( )
```

gets card Init file name

Returns

card Init file name

9.11.2.4 GetCardInitFilePath()

```
std::string Brood::Application::Components::Deck::GetCardInitFilePath ( )
```

gets card Init file path

Returns

card Init file path

9.11.2.5 GetCardList()

```
const std::vector< Brood::Application::Components::CardInfo * > & Brood::Application::Components::Deck::GetCardList ( ) const
```

Getter function to get the cardInfoList.

Returns

constance reference to the card Info List

9.11.2.6 GetCardPtrAtIdx()

```
Brood::Application::Components::CardInfo * Brood::Application::Components::Deck::GetCardPtrAt<br>Idx (  
    unsigned a_cardInfoIdx )
```

Gets a card info pointer to the passed index.

Parameters

<i>a_cardInfoIdx</i>	card info at passed index
----------------------	---------------------------

Returns

pointer to the card given index

9.11.2.7 GetCurrActiveCardIdx()

```
unsigned Brood::Application::Components::Deck::GetCurrActiveCardIdx ( )
```

gets card current active card index

Returns

card Init file path

9.11.2.8 GetCurrActiveCardPtr()

```
Brood::Application::Components::CardInfo * Brood::Application::Components::Deck::GetCurrActiveCardPtr ( )
```

9.11.2.9 GetDataToSave()

```
Brood::Application::Data::ST_DeckPrefabData Brood::Application::Components::Deck::GetDataToSave ( )
```

creates and returns deck data struct

Returns

cardInfo data struct with the deck data in it

setting up the deck

9.11.2.10 InitializeDeck()

```
void Brood::Application::Components::Deck::InitializeDeck (
    Brood::Application::Data::ST_DeckPrefabData & a_deckData )
```

initializes the deck with the passed data

Parameters

<i>a_deckData</i>	reference of the deck data
-------------------	----------------------------

setting up the deck

9.11.2.11 LoadCardFromInitFile()

```
bool Brood::Application::Components::Deck::LoadCardFromInitFile (
    std::string a_fileInitPath )
```

loads cards from a card Init file path

Parameters

<code>a_fileInitPath</code>	path to a file from which cards can be loaded into
-----------------------------	--

9.11.2.12 LoadDataFromFile()

```
void Brood::Application::Components::Deck::LoadDataFromFile ( 
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the deck and its path data from passed file

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
------------------------------	---------------------------------

9.11.2.13 operator=()

```
Brood::Application::Components::Deck & Brood::Application::Components::Deck::operator= ( 
    const Brood::Application::Components::Deck & a_other )
```

assignmnet operator

Parameters

<code>a_other</code>	reference to the the deck object that is being copied from
----------------------	--

9.11.2.14 ResetDeck()

```
void Brood::Application::Components::Deck::ResetDeck ( )
```

resets the `m_currUndealt` card index to zero and suffels the deck

9.11.2.15 SaveCardToInitFile()

```
bool Brood::Application::Components::Deck::SaveCardToInitFile ( 
    std::string a_fileInitPath )
```

saves cards to a card Init file path

Parameters

<code>a_fileInitPath</code>	path to a file to which cards can be saved into
-----------------------------	---

9.11.2.16 SaveDataToFile()

```
void Brood::Application::Components::Deck::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr,
    std::string a_gameTitle,
    unsigned a_idx )
```

initializes the deck with the passed data

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
<code>a_gameTitle</code>	title of the game
<code>a_idx</code>	this deck index in the deck manager

9.11.2.17 SetCardListSize()

```
void Brood::Application::Components::Deck::SetCardListSize (
    unsigned a_cardListSize )
```

resizes the card list

Parameters

<code>a_cardListSize</code>	
-----------------------------	--

9.11.2.18 SetCurrActiveCardIdx()

```
void Brood::Application::Components::Deck::SetCurrActiveCardIdx (
    unsigned a_currActiveCardIdx )
```

sets card current active card index

Returns

card Init file path

9.11.2.19 Shuffel()

```
void Brood::Application::Components::Deck::Shuffel ( )
```

shuffles the deck

9.11.3 Member Data Documentation

9.11.3.1 m_cardInitFileName

```
std::string Brood::Application::Components::Deck::m_cardInitFileName [private]
```

9.11.3.2 m_cardInitFilePath

```
std::string Brood::Application::Components::Deck::m_cardInitFilePath [private]
```

9.11.3.3 m_cardList

```
std::vector<Brood::Application::Components::CardInfo*> Brood::Application::Components::Deck::m_cardList [private]
```

9.11.3.4 m_currActiveCardIdx

```
unsigned Brood::Application::Components::Deck::m_currActiveCardIdx [private]
```

9.11.3.5 m_currUndealtCardIdx

```
unsigned Brood::Application::Components::Deck::m_currUndealtCardIdx [private]
```

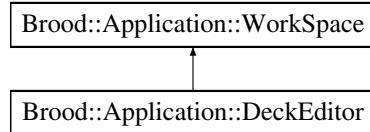
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Deck.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Deck.cpp](#)

9.12 Brood::Application::DeckEditor Class Reference

```
#include <DeckEditor.h>
```

Inheritance diagram for Brood::Application::DeckEditor:



Public Member Functions

- `DeckEditor (Brood::Application::Components::GameManager *aGameData, sf::RectangleShape *a←panelPtr)`
default constructor
- `~DeckEditor ()`
default destructor
- `void InitializeWorkSpace ()`
Initializes the work space.
- `void Update ()`
Updates function.
- `void UpdateAllDisplayElement ()`
updates the display element
- `void Draw (sf::RenderWindow &a_window)`
Draw funciton.
- `void Debugger ()`
debugger funciton

Private Member Functions

- `void UpdateCurrSelectedDeckIdx ()`
checks if the user interacted with the current selceted deck index panel
- `void UpdateDeckSizeX ()`
checks if the user interacted with the Deck x size panel
- `void UpdateDeckSizeY ()`
checks if the user interacted with the Deck y size panel
- `void UpdateDeckPosX ()`
checks if the user interacted with the Deck x position panel
- `void UpdateDeckPosY ()`
checks if the user interacted with the player Y Offset panel
- `void UpdateDeckFileTexture ()`
checks if the user interacted with the laod Deck texture panel
- `void UpdateCardInitFile ()`
checks if the user interacted with the laod card init panel
- `void UpdateCardNumber ()`
checks if the user interacted with the Card number panel

Private Attributes

- sf::RectangleShape * `m_panelBodyPtr`
holds the panel body
- Brood::BroodUI::TextBox * `m_txtSettingTitle`
- Brood::BroodUI::TextBox * `m_txtDeckCurrlIdxPrompt`
holds the text deck current index
- Brood::BroodUI::Button * `m_btnDeckDecCurrlIdx`
clicking it will decrease the deck current index number by 1
- Brood::BroodUI::TextBox * `m_txtDeckCurrlIdx`
displays the current deck current index
- Brood::BroodUI::Button * `m_btnDeckIncCurrlIdx`
clicking it will increase the deck current index number by 1
- Brood::BroodUI::TextBox * `m_txtDeckSizeXPrompt`
holds the text deck size x
- Brood::BroodUI::Button * `m_btnDeckDecSizeX`
clicking it will decrease the deck size x number by 1
- Brood::BroodUI::TextBox * `m_txtDeckSizeX`
displays the current deck size X
- Brood::BroodUI::Button * `m_btnDeckIncSizeX`
clicking it will increase the deck size x number by 1
- Brood::BroodUI::TextBox * `m_txtDeckSizeYPrompt`
holds the text deck size y
- Brood::BroodUI::Button * `m_btnDeckDecSizeY`
clicking it will decrease the deck size y number by 1
- Brood::BroodUI::TextBox * `m_txtDeckSizeY`
displays the current deck size y
- Brood::BroodUI::Button * `m_btnDeckIncSizeY`
clicking it will increase the deck size y number by 1
- Brood::BroodUI::TextBox * `m_txtDeckPosXPrompt`
holds the text deck position x
- Brood::BroodUI::Button * `m_btnDeckDecPosX`
clicking it will decrease the deck Position x number by 5
- Brood::BroodUI::TextBox * `m_txtDeckPosX`
displays the current deck Position X
- Brood::BroodUI::Button * `m_btnDeckIncPosX`
clicking it will increase the deck Position x number by 5
- Brood::BroodUI::TextBox * `m_txtDeckPosYPrompt`
holds the text deck Position y
- Brood::BroodUI::Button * `m_btnDeckDecPosY`
clicking it will decrease the deck Position y number by 5
- Brood::BroodUI::TextBox * `m_txtDeckPosY`
displays the current deck Position y
- Brood::BroodUI::Button * `m_btnDeckIncPosY`
clicking it will increase the deck Position y number by 5
- Brood::BroodUI::TextBox * `m_txtDeckBgFileNamePrompt`
holds the Deck texture filename prompt
- Brood::BroodUI::TextBox * `m_txtDeckBgFileInput`
holds the file name user enters
- Brood::BroodUI::Button * `m_btnDeckBgOpenFile`
clicking it will load the Deck file

- Brood::BroodUI::TextBox * m_txtCardInitFileNamePrompt
holds the CardInit texture filename prompt
- Brood::BroodUI::TextBox * m_txtCardInitFileInput
holds the file name user enters
- Brood::BroodUI::Button * m_btnCardInitOpenFile
clicking it will load the Cardinit file
- Brood::BroodUI::TextBox * m_txtCardNumPrompt
holds the text number of cards
- Brood::BroodUI::Button * m_btnDecCardNum
clicking it will decrease the number of cards by 1
- Brood::BroodUI::TextBox * m_txtCardNum
displays the current number of cards in the deck
- Brood::BroodUI::Button * m_btnIncCardNum
clicking it will increase the number of cards by 1

Additional Inherited Members

9.12.1 Constructor & Destructor Documentation

9.12.1.1 DeckEditor()

```
Brood::Application::DeckEditor::DeckEditor (
    Brood::Application::Components::GameManager * aGameData,
    sf::RectangleShape * aPanelPtr )
```

default constructor

Parameters

aGameData	pointer to the game data object
aPanelPtr	pointer to the panel body object

9.12.1.2 ~DeckEditor()

```
Brood::Application::DeckEditor::~DeckEditor ( )
```

default destructor

9.12.2 Member Function Documentation

9.12.2.1 Debugger()

```
void Brood::Application::DeckEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.12.2.2 Draw()

```
void Brood::Application::DeckEditor::Draw ( sf::RenderWindow & a_window ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Parameters

<i>a_window</i>	reference to the render window
-----------------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.12.2.3 InitializeWorkSpace()

```
void Brood::Application::DeckEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the board, setting section Dropdown Input.

Implements [Brood::Application::WorkSpace](#).

9.12.2.4 Update()

```
void Brood::Application::DeckEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.12.2.5 UpdateAllDisplayElement()

```
void Brood::Application::DeckEditor::UpdateAllDisplayElement ( ) [virtual]
```

updates the display element

@virtual

Implements [Brood::Application::WorkSpace](#).

9.12.2.6 UpdateCardInitFile()

```
void Brood::Application::DeckEditor::UpdateCardInitFile ( ) [private]
```

checks if the user interacted with the load card init panel

Load card init file panel contains load card init file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \data\ as it loads them from there.

9.12.2.7 UpdateCardNumber()

```
void Brood::Application::DeckEditor::UpdateCardNumber ( ) [private]
```

checks if the user interacted with the Card number panel

Card number panel contains Card number prompt textbox, Card number value text box, button to increase the Card number number, and button to decrease the Card number

Only the button to increase the Card number, and button to decrease the Card number are interactable

If the interactable button was pressed then the Card number is increased or decreased by 1. Card number should be more than or equal to 1

9.12.2.8 UpdateCurrSelectedDeckIdx()

```
void Brood::Application::DeckEditor::UpdateCurrSelectedDeckIdx ( ) [private]
```

checks if the user interacted with the current selected deck index panel

Current selected deck index panel contains current selected Deck index prompt textbox, current selected Deck index value text box, button to increase the current selected Deck index, and button to decrease the current selected Deck index

Only the button to increase the current selected Deck index, and button to decrease the current selected Deck index are interactable

If the interactable button was pressed then the current selected Deck index is increased or decreased by 1. current selected Deck index should be more than or equal to 1 but less than Deck number

9.12.2.9 UpdateDeckFileTexture()

```
void Brood::Application::DeckEditor::UpdateDeckFileTexture ( ) [private]
```

checks if the user interacted with the load Deck texture panel

Load Deck texture file panel contains load Deck texture file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\DeckTexture\ as it loads them from there.

9.12.2.10 UpdateDeckPosX()

```
void Brood::Application::DeckEditor::UpdateDeckPosX ( ) [private]
```

checks if the user interacted with the Deck x position panel

Deck X position panel contains current Deck X position promt textbox, current Deck X position value text box, button to increase the current Deck X position, and button to decrease the current Deck X position.

Only the button to increase the current Deck X position, and button to decrease the current Deck X position are interactable

If the interactable button was pressed then the current Deck X position is increased or decreased by 1 if the current Deck does not go outside the board window

9.12.2.11 UpdateDeckPosY()

```
void Brood::Application::DeckEditor::UpdateDeckPosY ( ) [private]
```

checks if the user interacted with the player Y Offset panel

Player Y Offset panel contains current player Y Offset promt textbox, current player Y Offset value text box, button to increase the current player Y Offset, and button to decrease the current player Y Offset.

Only the button to increase the current player Y Offset, and button to decrease the current player Y Offset are interactable

If the interactable button was pressed then the current player Y Offset is increased or decreased by 1 if the current player does not go outside the board window

9.12.2.12 UpdateDeckSizeX()

```
void Brood::Application::DeckEditor::UpdateDeckSizeX ( ) [private]
```

checks if the user interacted with the Deck x size panel

Deck X size panel contains current Deck X size promt textbox, current Deck X size value text box, button to increase the current Deck X size, and button to decrease the current Deck X size.

Only the button to increase the current Deck X size, and button to decrease the current Deck X size are interactable

If the interactable button was pressed then the current Deck X size is increased or decreased by 5 if the current Deck does not go outside the board window

9.12.2.13 UpdateDeckSizeY()

```
void Brood::Application::DeckEditor::UpdateDeckSizeY ( ) [private]
```

checks if the user interacted with the Deck y size panel

Deck Y size panel contains current Deck y size promt textbox, current Deck y size value text box, button to increase the current Deck y size, and button to decrease the current Deck y size.

Only the button to increase the current Deck y size, and button to decrease the current Deck y size are interactable

If the interactable button was pressed then the current Deck y size is increased or decreased by 5 if the current Deck does not go outside the board window

9.12.3 Member Data Documentation

9.12.3.1 m_btnCardInitOpenFile

`Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnCardInitOpenFile [private]`

clicking it will load the CardInit file

9.12.3.2 m_btnDecCardNum

`Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDecCardNum [private]`

clicking it will decrease the number of cards by 1

9.12.3.3 m_btnDeckBgOpenFile

`Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckBgOpenFile [private]`

clicking it will load the Deck file

9.12.3.4 m_btnDeckDecCurrIdx

`Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckDecCurrIdx [private]`

clicking it will decrease the deck current index number by 1

9.12.3.5 m_btnDeckDecPosX

`Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckDecPosX [private]`

clicking it will decrease the deck Position x number by 5

9.12.3.6 **m_btnDeckDecPosY**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckDecPosY [private]
```

clicking it will decrease the deck Position y number by 5

9.12.3.7 **m_btnDeckDecSizeX**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckDecSizeX [private]
```

clicking it will decrease the deck size x number by 1

9.12.3.8 **m_btnDeckDecSizeY**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckDecSizeY [private]
```

clicking it will decrease the deck size y number by 1

9.12.3.9 **m_btnDeckIncCurrlIdx**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckIncCurrlIdx [private]
```

clicking it will increase the deck current index number by 1

9.12.3.10 **m_btnDeckIncPosX**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckIncPosX [private]
```

clicking it will increase the deck Position x number by 5

9.12.3.11 **m_btnDeckIncPosY**

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckIncPosY [private]
```

clicking it will increase the deck Position y number by 5

9.12.3.12 m_btnDeckIncSizeX

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckIncSizeX [private]
```

clicking it will increase the deck size x number by 1

9.12.3.13 m_btnDeckIncSizeY

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnDeckIncSizeY [private]
```

clicking it will increase the deck size y number by 1

9.12.3.14 m_btnIncCardNum

```
Brood::BroodUI::Button* Brood::Application::DeckEditor::m_btnIncCardNum [private]
```

clicking it will increase the number of cards by 1

9.12.3.15 m_panelBodyPtr

```
sf::RectangleShape* Brood::Application::DeckEditor::m_panelBodyPtr [private]
```

holds the panel body

9.12.3.16 m_txtCardInitFileInput

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtCardInitFileInput [private]
```

holds the file name user enters

9.12.3.17 m_txtCardInitFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtCardInitFileNamePrompt [private]
```

holds the CardInit texture filename prompt

9.12.3.18 m_txtCardNum

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtCardNum [private]
```

displays the current number of cards in the deck

9.12.3.19 m_txtCardNumPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtCardNumPrompt [private]
```

holds the text number of cards

9.12.3.20 m_txtDeckBgFileInput

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckBgFileInput [private]
```

holds the file name user enters

9.12.3.21 m_txtDeckBgFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckBgFileNamePrompt [private]
```

holds the Deck texture filename prompt

9.12.3.22 m_txtDeckCurrIdx

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckCurrIdx [private]
```

displays the current deck current index

9.12.3.23 m_txtDeckCurrIdxPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckCurrIdxPrompt [private]
```

holds the text deck current index

9.12.3.24 m_txtDeckPosX

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckPosX [private]
```

displays the current deck Position X

9.12.3.25 m_txtDeckPosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckPosXPrompt [private]
```

holds the text deck position x

9.12.3.26 m_txtDeckPosY

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckPosY [private]
```

displays the current deck Position y

9.12.3.27 m_txtDeckPosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckPosYPrompt [private]
```

holds the text deck Position y

9.12.3.28 m_txtDeckSizeX

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckSizeX [private]
```

displays the current deck size X

9.12.3.29 m_txtDeckSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckSizeXPrompt [private]
```

holds the text deck size x

9.12.3.30 m_txtDeckSizeY

`Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckSizeY [private]`

displays the current deck size y

9.12.3.31 m_txtDeckSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtDeckSizeYPrompt [private]`

holds the text deck size y

9.12.3.32 m_txtSettingTitle

`Brood::BroodUI::TextBox* Brood::Application::DeckEditor::m_txtSettingTitle [private]`

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckEditor.cpp](#)

9.13 Brood::Application::Components::DeckManager Class Reference

```
#include <DeckManager.h>
```

Public Member Functions

- [DeckManager \(\)](#)
default constructor
- [~DeckManager \(\)](#)
default destructor
- [DeckManager \(const Brood::Application::Components::DeckManager &a_otherDeckManager\)](#)
copy constructor
- [DeckManager & operator= \(const Brood::Application::Components::DeckManager &a_otherDeckManager\)](#)
assignmnet operator
- [void InitializeDeckManager \(Brood::Application::Data::ST_DeckManagerData &a_deckManagerData\)](#)
initializes the deck manager with the passed data
- [Brood::Application::Data::ST_DeckManagerData GetDataToSave \(\)](#)
creates and returns deck manager data struct
- [void SaveDataToFile \(Brood::Application:: FileAccess *a_fileAccessPtr, std::string a_gameTitle\)](#)
saves the deck manager and its path data to passed file
- [void LoadDataFromFile \(Brood::Application:: FileAccess *a_fileAccessPtr\)](#)
loads the deck manager from passed file

- `unsigned GetCurrActiveDeckIdx ()`
`getter funciton to get the current deck index`
- `Brood::Application::Components::Deck * GetDeckAtIdx (unsigned a_deckIdx)`
`getter function to get the deck at the passed index`
- `std::vector< Brood::Application::Components::Deck * > & GetDeckList ()`
`getter function to get the deck list`
- `Brood::Application::Components::Deck * GetCurrActiveDeck ()`
`getter function to get the current active deck`
- `bool GetIncorrectPenalty ()`
`getter function to get current incorrect penalty value`
- `Brood::Application::Components::ENUM_MovementType GetMovementType ()`
`getter function to get current movement type value`
- `void SetIncorrectPenalty (bool a_incorrectPenalty)`
`setter function to set current incorrect penalty value`
- `void SetMovementType (Brood::Application::Components::ENUM_MovementType a_movementType)`
`setter function to set current movement type value`
- `void SetDeckSize (unsigned a_deckSize)`
`Resizes the deck.`
- `void SetCurrActiveDeckIdx (unsigned a_currDeckIdx)`
`Setter funciton to set the current deck index.`
- `void AddCardToDeckAtCurrlIdx (Brood::Application::Components::CardInfo a_cardToAdd)`
`Adds a card to the deck at current index.`
- `void Draw (sf::RenderWindow &a_window)`
`Draw funciton.`
- `void Debugger ()`
`debugger funciton`

Private Attributes

- `bool m_incorrectPenalty`
- `Brood::Application::Components::ENUM_MovementType m_movementType`
- `unsigned m_currDeckIdx`
- `std::vector< Brood::Application::Components::Deck * > m_deckList`

9.13.1 Constructor & Destructor Documentation

9.13.1.1 DeckManager() [1/2]

```
Brood::Application::Components::DeckManager::DeckManager ( )
```

default constructor

9.13.1.2 ~DeckManager()

```
Brood::Application::Components::DeckManager::~DeckManager ( )
```

default destructor

9.13.1.3 DeckManager() [2/2]

```
Brood::Application::Components::DeckManager::DeckManager ( 
    const Brood::Application::Components::DeckManager & a_otherDeckManager )
```

copy constructor

Parameters

<i>a_otherDeckManager</i>	reference to the the deck manager object that is being copied from
---------------------------	--

9.13.2 Member Function Documentation

9.13.2.1 AddCardToDeckAtCurrlIdx()

```
void Brood::Application::Components::DeckManager::AddCardToDeckAtCurrlIdx ( 
    Brood::Application::Components::CardInfo a_cardToAdd )
```

Adds a card to the deck at current index.

Parameters

<i>a_cardToAdd</i>	card info to add the current deck
--------------------	-----------------------------------

9.13.2.2 Debugger()

```
void Brood::Application::Components::DeckManager::Debugger ( )
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

9.13.2.3 Draw()

```
void Brood::Application::Components::DeckManager::Draw (
    sf::RenderWindow & a_window )
```

Draw funciton.

@virtual

Draws all the component to the screen

9.13.2.4 GetCurrActiveDeck()

```
Brood::Application::Components::Deck * Brood::Application::Components::DeckManager::GetCurrActiveDeck ( )
```

getter function to get the current active deck

Returns

pointer to current active deck

9.13.2.5 GetCurrActiveDeckIdx()

```
unsigned Brood::Application::Components::DeckManager::GetCurrActiveDeckIdx ( )
```

getter funciton to get the current deck index

Returns

current deck index

9.13.2.6 GetDataToSave()

```
Brood::Application::Data::ST_DeckManagerData Brood::Application::Components::DeckManager::GetDataToSave ( )
```

creates and returns deck manager data struct

Returns

cardInfo data struct with the deck manager data in it

9.13.2.7 GetDeckAtIdx()

```
Brood::Application::Components::Deck * Brood::Application::Components::DeckManager::GetDeckAtIndex (
    unsigned a_deckIdx )
```

getter function to get the deck at the passed index

Parameters

<code>a_deckIdx</code>	index from which the deck is to be returned
------------------------	---

Returns

pointer to the deck at passed index if valid; else nullptr

9.13.2.8 GetDeckList()

```
std::vector< Brood::Application::Components::Deck * > & Brood::Application::Components::DeckManager::GetDeckList ( )
```

getter function to get the deck list

Returns

reference to the decklist

9.13.2.9 GetIncorrectPenalty()

```
bool Brood::Application::Components::DeckManager::GetIncorrectPenalty ( )
```

getter function to get current incorrect penalty value

Returns

current incorrect penalty value

9.13.2.10 GetMovementType()

```
Brood::Application::Components::ENUM_MovementType Brood::Application::Components::DeckManager::GetMovementType ( )
```

getter function to get current movement type value

Returns

current current movement type value

9.13.2.11 InitializeDeckManager()

```
void Brood::Application::Components::DeckManager::InitializeDeckManager (
    Brood::Application::Data::ST_DeckManagerData & a_deckManagerData )
```

initializes the deck manager with the passed data

Parameters

<i>a_deckManagerData</i>	reference of the deck manager data
--------------------------	------------------------------------

9.13.2.12 LoadDataFromFile()

```
void Brood::Application::Components::DeckManager::LoadDataFromFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the deck manager from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.13.2.13 operator=()

```
Brood::Application::Components::DeckManager & Brood::Application::Components::DeckManager::operator=
( const Brood::Application::Components::DeckManager & a_otherDeckManager )
```

assignmnet operator

Parameters

<i>a_otherDeckManager</i>	reference to the the deck manager object that is being copied from
---------------------------	--

9.13.2.14 SaveDataToFile()

```
void Brood::Application::Components::DeckManager::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr,
    std::string a_gameTitle )
```

saves the deck manager and its path data to passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
<i>a_gameTitle</i>	title of the game

9.13.2.15 SetCurrActiveDeckIdx()

```
void Brood::Application::Components::DeckManager::SetCurrActiveDeckIdx (  
    unsigned a_currDeckIdx )
```

Setter function to set the current deck index.

Parameters

a_currDeckIdx	deck index
---------------	------------

9.13.2.16 SetDeckSize()

```
void Brood::Application::Components::DeckManager::SetDeckSize (   
    unsigned a_deckSize )
```

Resizes the deck.

Parameters

a_deckSize	deck size
------------	-----------

9.13.2.17 SetIncorrectPenalty()

```
void Brood::Application::Components::DeckManager::SetIncorrectPenalty (   
    bool a_incorrectPenalty )
```

setter function to set current incorrect penalty value

Parameters

a_incorrectPenalty	incorrect penalty value to set
--------------------	--------------------------------

9.13.2.18 SetMovementType()

```
void Brood::Application::Components::DeckManager::SetMovementType (   
    Brood::Application::Components::ENUM_MovementType a_movementType )
```

setter function to set current movement type value

Parameters

<i>a_movementType</i>	current movement type value to set
-----------------------	------------------------------------

9.13.3 Member Data Documentation**9.13.3.1 m_currDeckIdx**

```
unsigned Brood::Application::Components::DeckManager::m_currDeckIdx [private]
```

9.13.3.2 m_deckList

```
std::vector<Brood::Application::Components::Deck*> Brood::Application::Components::DeckManager::m_deckList [private]
```

9.13.3.3 m_incorrectPenalty

```
bool Brood::Application::Components::DeckManager::m_incorrectPenalty [private]
```

9.13.3.4 m_movementType

```
Brood::Application::Components::ENUM_MovementType Brood::Application::Components::DeckManager::m_movementType [private]
```

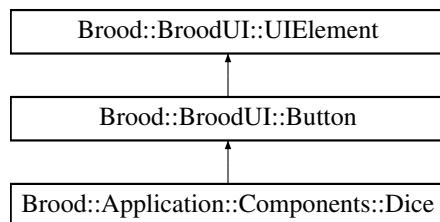
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckManager.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckManager.cpp](#)

9.14 Brood::Application::Components::Dice Class Reference

```
#include <Dice.h>
```

Inheritance diagram for Brood::Application::Components::Dice:



Public Member Functions

- `Dice (int a_numSides=6, Brood::BroodUI::UIElement *a_parentPtr=nullptr)`
Default Constructor.
- `Dice (std::string a_texturePath, int a_numSides=6, Brood::BroodUI::UIElement *a_parentPtr=nullptr)`
Initializes the button object.
- `virtual ~Dice ()`
destructor
- `Dice (const Dice &a_otherDice)`
copy constructor
- `Dice & operator= (const Dice &a_otherDice)`
assignment operator
- `void InitializeDice (Brood::Application::Data::ST_DicePrefabData &a_diceData)`
initializes the dice with the passed data
- `Brood::Application::Data::ST_DicePrefabData GetDataToSave ()`
creates and returns dice data struct
- `void SaveDataToFile (Brood::Application:: FileAccess *a_fileAccessPtr)`
loads the dice data from passed file
- `void LoadDataFromFile (Brood::Application:: FileAccess *a_fileAccessPtr)`
loads the dice and its path data from passed file
- `void SetBodySize (sf::Vector2f a_size) override`
Setter function to set the dice's Size.
- `void SetBodySize (float a_sizeX, float a_sizeY) override`
Setter function to set the dice's Size.
- `void SetNumSides (unsigned a_numSides)`
Sets number of side the dice has.
- `bool SetTexture (std::string a_texturePath)`
Sets texture for the dice.
- `const unsigned GetNumSides ()`
Getter function to get the dice's number of sides.
- `unsigned RollDice ()`
Rolls the dice to get a number between 0 and m_numSides.

Private Attributes

- `unsigned m_numSides`
number of side in a die

Additional Inherited Members

9.14.1 Constructor & Destructor Documentation

9.14.1.1 Dice() [1/3]

```
Dice::Dice (
    int a_numSides = 6,
    Brood::BroodUI::UIElement * a_parentPtr = nullptr )
```

Default Constructor.

Initializes the button object

Parameters

<i>m_parentPtr</i>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
<i>a_index</i>	the nth child of the parent; if parent does not exist then -1 -> default value -1
<i>a_numSides</i>	number of a side that the dice has; -> default 6

9.14.1.2 Dice() [2/3]

```
Dice::Dice (
    std::string a_texturePath,
    int a_numSides = 6,
    Brood::BroodUI::UIElement * a_parentPtr = nullptr )
```

Initializes the button object.

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Warning

The sprite for the face of the should be lined linearly. That is only one row of sprite should be present.

Assumes that the sprites are square.

This constructor is called when a texture is passed is passed

Parameters

<i>a_texturePath</i>	file path to the texture containing face of the dice; it is a single texture that has texture for num faces of the dice
<i>m_parentPtr</i>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
<i>a_index</i>	the nth child of the parent; if parent does not exist then -1 -> default value -1
<i>a_numSides</i>	number of a side that the dice has; -> default 6

9.14.1.3 ~Dice()

```
Dice::~Dice ( ) [virtual]
```

destructor

9.14.1.4 Dice() [3/3]

```
Dice::Dice (
    const Dice & a_otherDice )
```

copy constructor

Parameters

<code>a_otherDice</code>	reference to the the dice structue that is being copied from
--------------------------	--

9.14.2 Member Function Documentation

9.14.2.1 GetDataToSave()

```
Brood::Application::Data::ST_DicePrefabData Dice::GetDataToSave ( )
```

creates and returns dice data struct

Returns

dice data struct with the dice data in it

9.14.2.2 GetNumSides()

```
const unsigned Dice::GetNumSides ( )
```

Getter function to get the dice's number of sides.

Returns

dice's saved number of side

9.14.2.3 InitializeDice()

```
void Dice::InitializeDice (  
    Brood::Application::Data::ST_DicePrefabData & a_diceData )
```

initializes the dice with the passed data

Parameters

<code>a_deckData</code>	reference of the dice data
-------------------------	----------------------------

9.14.2.4 LoadDataFromFile()

```
void Dice::LoadDataFromFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the dice and its path data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.14.2.5 operator=()

```
Brood::Application::Components::Dice & Dice::operator= (
    const Dice & a_otherDice )
```

assignmnet operator

Parameters

<i>a_otherDice</i>	reference to the the path structue that is being copied from
--------------------	--

9.14.2.6 RollDice()

```
unsigned Dice::RollDice ( )
```

Rolls the dice to get a number between 0 and m_numSides.

This also sets the sprite from the loaded texture

Returns

random number between 0 and m_numSides

9.14.2.7 SaveDataToFile()

```
void Dice::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the dice data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.14.2.8 SetBodySize() [1/2]

```
void Dice::SetBodySize (
    float a_sizeX,
    float a_sizeY ) [override], [virtual]
```

Setter function to set the dice's Size.

@virtual This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_sizeX</i>	length of the element
<i>a_sizeY</i>	width of the element

Reimplemented from [Brood::BroodUI::UIElement](#).

9.14.2.9 SetBodySize() [2/2]

```
void Dice::SetBodySize (
    sf::Vector2f a_size ) [override], [virtual]
```

Setter function to set the dice's Size.

@virtual

Parameters

<i>a_size</i>	size of the element
---------------	---------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.14.2.10 SetNumSides()

```
void Dice::SetNumSides (
    unsigned a_numSides )
```

Sets number of side the dice has.

Parameters

<code>a_numSides</code>	number of sides the dice has
-------------------------	------------------------------

9.14.2.11 SetTexture()

```
bool Dice::SetTexture (
    std::string a_texturePath )
```

Sets texture for the dice.

Parameters

<code>a_texturePathfile</code>	path to to the texture containing face of the dice; it is a single texture that has texture for num faces of the dice
--------------------------------	---

9.14.3 Member Data Documentation**9.14.3.1 m_numSides**

`unsigned Brood::Application::Components::Dice::m_numSides [private]`

number of side in a die

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Dice.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Dice.cpp](#)

9.15 Dice Class Reference

A class to create a Deck.

```
#include "Deck.h"
```

9.15.1 Detailed Description

A class to create a Deck.

A class to create a dice.

It has all the card info init

TODO add a function and ctor to load from a file

This class is derived form the [Button](#) class. This [Dice](#) is a unbiased [Dice](#).

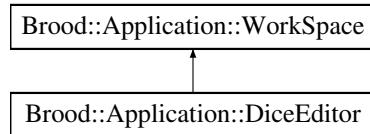
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Deck.h](#)

9.16 Brood::Application::DiceEditor Class Reference

```
#include <DiceEditor.h>
```

Inheritance diagram for Brood::Application::DiceEditor:



Public Member Functions

- [DiceEditor \(Brood::Application::Components::GameDataManager *aGameData, sf::RectangleShape *aPanelPtr\)](#)
default constructor
- [~DiceEditor \(\)](#)
default destructor
- [void InitializeWorkSpace \(\)](#)
Initializes the work space.
- [void Update \(\)](#)
Updates function.
- [void UpdateAllDisplayElement \(\)](#)
updates the display element
- [void Draw \(sf::RenderWindow &aWindow\)](#)
Draw funciton.
- [void Debugger \(\)](#)
debugger funciton

Private Member Functions

- [void UpdateDiceSideNum \(\)](#)
checks if the user interacted with the dice side number panel
- [void UpdateDiceSizeX \(\)](#)
checks if the user interacted with the dice x size panel
- [void UpdateDiceSizeY \(\)](#)
checks if the user interacted with the dice y size panel
- [void UpdateDicePosX \(\)](#)
checks if the user interacted with the dice x position panel
- [void UpdateDicePosY \(\)](#)
checks if the user interacted with the player Y Offset panel
- [void UpdateDiceFileTexture \(\)](#)
checks if the user interacted with the laod dice texture panel
- [void UpdateRollDice \(\)](#)
checks if the user interacted with the roll dice panel

Private Attributes

- sf::RectangleShape * `m_panelBodyPtr`
holds the panel body
- Brood::Application::Components::Board * `m_boardPtr`
board for the game
- Brood::Application::Components::Dice * `m_dicePtr`
pointer to the deck manager
- Brood::BroodUI::TextBox * `m_txtSettingTitle`
- Brood::BroodUI::TextBox * `m_txtDiceSidePrompt`
holds the text `Dice` side
- Brood::BroodUI::Button * `m_btnDiceDecSide`
clicking it will decrease the Diceside number by 1
- Brood::BroodUI::TextBox * `m_txtDiceSide`
displays the `Dice` Side
- Brood::BroodUI::Button * `m_btnDiceIncSide`
clicking it will increase the `Dice` Side number by 1
- Brood::BroodUI::TextBox * `m_txtDiceSizeXPrompt`
holds the text `Dice` X-Size
- Brood::BroodUI::Button * `m_btnDiceDecSizeX`
clicking it will decrease the `Dice` x-size number by 5
- Brood::BroodUI::TextBox * `m_txtDiceSizeX`
displays the `Dice` x-size
- Brood::BroodUI::Button * `m_btnDiceIncSizeX`
clicking it will increase the `Dice` x-size number by 5
- Brood::BroodUI::TextBox * `m_txtDiceSizeYPrompt`
holds the text `Dice` y-Size
- Brood::BroodUI::Button * `m_btnDiceDecSizeY`
clicking it will decrease the `Dice` y-size number by 5
- Brood::BroodUI::TextBox * `m_txtDiceSizeY`
displays the `Dice` y-size
- Brood::BroodUI::Button * `m_btnDiceIncSizeY`
clicking it will increase the `Dice` y-size number by 5
- Brood::BroodUI::TextBox * `m_txtDicePosXPrompt`
holds the text `Dice` X-Pos
- Brood::BroodUI::Button * `m_btnDiceDecPosX`
clicking it will decrease the `Dice` X-Pos number by 5
- Brood::BroodUI::TextBox * `m_txtDicePosX`
displays the `Dice` X-Pos
- Brood::BroodUI::Button * `m_btnDiceIncPosX`
clicking it will increase the `Dice` X-Pos number by 5
- Brood::BroodUI::TextBox * `m_txtDicePosYPrompt`
holds the text `Dice` y-Pos prompt
- Brood::BroodUI::Button * `m_btnDiceDecPosY`
clicking it will decrease the `Dice` y-Pos number by 5
- Brood::BroodUI::TextBox * `m_txtDicePosY`
displays the `Dice` y-Pos
- Brood::BroodUI::Button * `m_btnDiceIncPosY`
clicking it will increase the `Dice` y-Pos number by 5
- Brood::BroodUI::TextBox * `m_txtDiceFileNamePrompt`
holds the `Dice` texture filename prompt

- Brood::BroodUI::TextBox * m_txtDiceFileInput
holds the file name user enters
- Brood::BroodUI::Button * m_btnDiceOpenFile
clicking it will load the [Dice](#) file
- Brood::BroodUI::Button * m_btnRollDice
Rolls the dice.

Additional Inherited Members

9.16.1 Constructor & Destructor Documentation

9.16.1.1 DiceEditor()

```
DiceEditor::DiceEditor (
    Brood::Application::Components::GameDataManager * aGameData,
    sf::RectangleShape * aPanelPtr )
```

default constructor

Parameters

<code>aGameData</code>	pointer to the game data object
<code>aPanelPtr</code>	pointer to the panel body object

9.16.1.2 ~DiceEditor()

```
DiceEditor::~DiceEditor ( )
```

default destructor

9.16.2 Member Function Documentation

9.16.2.1 Debugger()

```
void DiceEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.16.2.2 Draw()

```
void DiceEditor::Draw (
    sf::RenderWindow & a_window ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Parameters

<i>a_window</i>	reference to the render window
-----------------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.16.2.3 InitializeWorkSpace()

```
void DiceEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It initializes the folloing panel element: dice side number, dice x-size, dice y-size, dice x-pos, dice-ypos, dice texture filename input, and roll a dice to check if the texture works

Implements [Brood::Application::WorkSpace](#).

9.16.2.4 Update()

```
void DiceEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the dice editor workspace

Updates the panel element of the dice setting

It Updates the following panel element: dice side number, dice x-size, dice y-size, dice x-pos, dice-ypos, dice texture filename input, and roll a dice to check if the texture works

Implements [Brood::Application::WorkSpace](#).

9.16.2.5 UpdateAllDisplayElement()

```
void DiceEditor::UpdateAllDisplayElement ( ) [virtual]
```

updates the display element

@virtual

Implements [Brood::Application::WorkSpace](#).

9.16.2.6 UpdateDiceFileDialog()

```
void DiceEditor::UpdateDiceFileDialog ( ) [private]
```

checks if the user interacted with the load dice texture panel

Load dice texture file panel contains load dice texture file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\DiceTexture\ as it loads them from there.

9.16.2.7 UpdateDicePosX()

```
void DiceEditor::UpdateDicePosX ( ) [private]
```

checks if the user interacted with the dice x position panel

dice X position panel contains current dice X position prompt textbox, current dice X position value text box, button to increase the current dice X position, and button to decrease the current dice X position.

Only the button to increase the current dice X position, and button to decrease the current dice X position are interactable

If the interactable button was pressed then the current dice X position is increased or decreased by 1 if the current dice does not go outside the board window

9.16.2.8 UpdateDicePosY()

```
void DiceEditor::UpdateDicePosY ( ) [private]
```

checks if the user interacted with the player Y Offset panel

Player Y Offset panel contains current player Y Offset prompt textbox, current player Y Offset value text box, button to increase the current player Y Offset, and button to decrease the current player Y Offset.

Only the button to increase the current player Y Offset, and button to decrease the current player Y Offset are interactable

If the interactable button was pressed then the current player Y Offset is increased or decreased by 1 if the current player does not go outside the board window

9.16.2.9 UpdateDiceSideNum()

```
void DiceEditor::UpdateDiceSideNum ( ) [private]
```

checks if the user interacted with the dice side number panel

dice side number panel contains current diceside number prompt textbox, current dice side number value text box, button to increase the current dice side number, and button to decrease the current dice side number.

Only the button to increase the current dice side number, and button to decrease the current dice side number are interactable

If the interactable button was pressed then the current dice side number is increased or decreased by 1. The side number can be between 1 and 100

9.16.2.10 UpdateDiceSizeX()

```
void DiceEditor::UpdateDiceSizeX ( ) [private]
```

checks if the user interacted with the dice x size panel

dice X size panel contains current dice X size prompt textbox, current dice X size value text box, button to increase the current dice X size, and button to decrease the current dice X size.

Only the button to increase the current dice X size, and button to decrease the current dice X size are interactable

If the interactable button was pressed then the current dice X size is increased or decreased by 5 if the current dice does not go outside the board window

9.16.2.11 UpdateDiceSizeY()

```
void DiceEditor::UpdateDiceSizeY ( ) [private]
```

checks if the user interacted with the dice y size panel

dice Y size panel contains current dice y size prompt textbox, current dice y size value text box, button to increase the current dice y size, and button to decrease the current dice y size.

Only the button to increase the current dice y size, and button to decrease the current dice y size are interactable

If the interactable button was pressed then the current dice y size is increased or decreased by 5 if the current dice does not go outside the board window

9.16.2.12 UpdateRollDice()

```
void DiceEditor::UpdateRollDice ( ) [private]
```

checks if the user interacted with the roll dice panel

Roll dice contains a button to roll a dice

Only the button to roll a dice is interactable

If the interactable button was pressed then it rolls a dice

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\DiceTexture\ as it loads them from there. Do not include .png

9.16.3 Member Data Documentation

9.16.3.1 m_boardPtr

```
Brood::Application::Components::Board* Brood::Application::DiceEditor::m_boardPtr [private]
```

board for the game

9.16.3.2 m_btnDiceDecPosX

```
Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceDecPosX [private]
```

clicking it will decrease the **Dice** X-Pos number by 5

9.16.3.3 m_btnDiceDecPosY

```
Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceDecPosY [private]
```

clicking it will decrease the **Dice** y-Pos number by 5

9.16.3.4 m_btnDiceDecSide

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceDecSide [private]`

clicking it will decrease the Diceside number by 1

9.16.3.5 m_btnDiceDecSizeX

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceDecSizeX [private]`

clicking it will decrease the [Dice](#) x-size number by 5

9.16.3.6 m_btnDiceDecSizeY

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceDecSizeY [private]`

clicking it will decrease the [Dice](#) y-size number by 5

9.16.3.7 m_btnDiceIncPosX

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceIncPosX [private]`

clicking it will increase the [Dice](#) X-Pos number by 5

9.16.3.8 m_btnDiceIncPosY

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceIncPosY [private]`

clicking it will increase the [Dice](#) y-Pos number by 5

9.16.3.9 m_btnDiceIncSide

`Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceIncSide [private]`

clicking it will increase the [Dice](#) Side number by 1

9.16.3.10 m_btnDiceIncSizeX

Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceIncSizeX [private]

clicking it will increase the Dice x-size number by 5

9.16.3.11 m_btnDiceIncSizeY

Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceIncSizeY [private]

clicking it will increase the Dice y-size number by 5

9.16.3.12 m_btnDiceOpenFile

Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnDiceOpenFile [private]

clicking it will load the Dice file

9.16.3.13 m_btnRollDice

Brood::BroodUI::Button* Brood::Application::DiceEditor::m_btnRollDice [private]

Rolls the dice.

clicking it will decrease the Dice y-Pos number by 5

9.16.3.14 m_dicePtr

Brood::Application::Components::Dice* Brood::Application::DiceEditor::m_dicePtr [private]

pointer to the deck manager

9.16.3.15 m_panelBodyPtr

sf::RectangleShape* Brood::Application::DiceEditor::m_panelBodyPtr [private]

holds the panel body

9.16.3.16 m_txtDiceFileInput

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceFileInput [private]
```

holds the file name user enters

9.16.3.17 m_txtDiceFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceFileNamePrompt [private]
```

holds the **Dice** texture filename prompt

9.16.3.18 m_txtDicePosX

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDicePosX [private]
```

displays the **Dice** X-Pos

9.16.3.19 m_txtDicePosXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDicePosXPrompt [private]
```

holds the text **Dice** X-Pos

9.16.3.20 m_txtDicePosY

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDicePosY [private]
```

displays the **Dice** y-Pos

9.16.3.21 m_txtDicePosYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDicePosYPrompt [private]
```

holds the text **Dice** y-Pos prompt

9.16.3.22 m_txtDiceSide

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSide [private]
```

displays the [Dice](#) Side

9.16.3.23 m_txtDiceSidePrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSidePrompt [private]
```

holds the text [Dice](#) side

9.16.3.24 m_txtDiceSizeX

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSizeX [private]
```

displays the [Dice](#) x-size

9.16.3.25 m_txtDiceSizeXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSizeXPrompt [private]
```

holds the text [Dice](#) X-Size

9.16.3.26 m_txtDiceSizeY

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSizeY [private]
```

displays the [Dice](#) y-size

9.16.3.27 m_txtDiceSizeYPrompt

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtDiceSizeYPrompt [private]
```

holds the text [Dice](#) y-Size

9.16.3.28 m_txtSettingTitle

```
Brood::BroodUI::TextBox* Brood::Application::DiceEditor::m_txtSettingTitle [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.h
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.cpp

9.17 DiceEditor Class Reference

It allows the users to tweek the dice setting.

```
#include "DiceEditor.h"
```

9.17.1 Detailed Description

It allows the users to tweek the dice setting.

It is a container that bundles elements that is present in DiceEditor workspace

It has the following panel element: dice side number, dice x-size, dice y-size, dice x-pos, dice-ypos, dice texture filename input, and roll a dice to check if the texture works

Derived from Workspace class

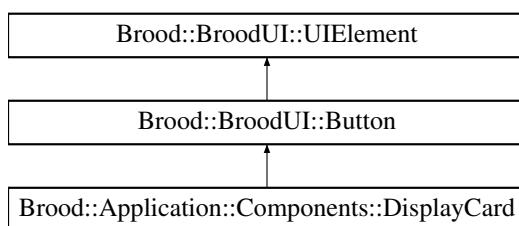
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.h

9.18 Brood::Application::Components::DisplayCard Class Reference

```
#include <DisplayCard.h>
```

Inheritance diagram for Brood::Application::Components::DisplayCard:



Public Member Functions

- `DisplayCard ()`
`Default Constructor.`
- `~DisplayCard ()`
`default destrctor`
- `DisplayCard (const DisplayCard &a_other)`
`Copy constructor.`
- `DisplayCard & operator= (const DisplayCard &a_other)`
`assignment operator`
- `void InitializeDisplayCard ()`
`Initializes the display card.`
- `void InitializeDisplayCard (Brood::Application::Data::ST_DisplayCardData &a_displayCardData)`
`initializes the displayCard with the passed data`
- `Brood::Application::Data::ST_DisplayCardData GetDataToSave ()`
`creates and returns displayCard data struct`
- `void SaveDataToFile (Brood::Application::FileAccess *a_fileAccessPtr)`
`loads the DisplayCard data from passed file`
- `void LoadDataFromFile (Brood::Application::FileAccess *a_fileAccessPtr)`
`loads the DisplayCard and its path data from passed file`
- `std::string GetFrontBgFileName ()`
`Getter function to get a front background filename.`
- `std::string GetBackBgFileName ()`
`Getter function to get a back background filename.`
- `Brood::BroodUI::TextBox * GetFrontTimePromptPtr ()`
`Getter function to get the pointer to the frontTime prompt ui element.`
- `Brood::BroodUI::TextBox * GetFrontTimeValuePtr ()`
`Getter function to get the pointer to the frontTime value ui element.`
- `Brood::BroodUI::TextBox * GetUpPromptPtr ()`
`Getter function to get the pointer to the up prompt ui element.`
- `Brood::BroodUI::TextBox * GetUpValuePtr ()`
`Getter function to get the pointer to the up value ui element.`
- `Brood::BroodUI::TextBox * GetDownPromptPtr ()`
`Getter function to get the pointer to the down prompt ui element.`
- `Brood::BroodUI::TextBox * GetDownValuePtr ()`
`Getter function to get the pointer to the down value ui element.`
- `Brood::BroodUI::Button * GetTurnCardPtr ()`
`Getter function to get the pointer to the turn card ui element.`
- `Brood::BroodUI::TextBox * GetBackTimePromptPtr ()`
`Getter function to get the pointer to the back time prompt ui element.`
- `Brood::BroodUI::TextBox * GetBackTimeValuePtr ()`
`Getter function to get the pointer to the back time value ui element.`
- `Brood::BroodUI::TextBox * GetQuestionPromptPtr ()`
`Getter function to get the pointer to the answer prompt ui element.`
- `Brood::BroodUI::TextBox * GetQuestionValuePtr ()`
`Getter function to get the pointer to the answer value ui element.`
- `Brood::BroodUI::TextBox * GetAnswerPromptPtr ()`
`Getter function to get the pointer to the user answer prompt ui element.`
- `Brood::BroodUI::TextBox * GetAnswerValuePtr ()`
`Getter function to get the pointer to the user answer value ui element.`
- `Brood::BroodUI::Button * GetSubmitPtr ()`

- `sf::Vector2f GetFurtherstBottomRightCordinate ()`
gets the bottom right cordinate of the box that encloses all the child component
- `sf::Vector2f GetNearestTopLeftCordinate ()`
gets top left cordinate of the box that encloses all the child component
- `bool SetFrontBgFileName (std::string a_frontBgFilename)`
Setter function to set a front background filename.
- `bool SetBackBgFileName (std::string a_backBgFilename)`
Setter function to set a back background filename.
- `void SetCardInfoToDisplay (Brood::Application::Components::CardInfo *a_cardInfoToDisplay)`
setter funciton to set the card info that will be used to display
- `void ToggleCardFace ()`
toggles the background
- `void SetCardFront (bool a_currFront)`
sets m_currFront
- `void UpdateDisplayedText ()`
updates the displayed text from the saved cardInfo
- `void Draw (sf::RenderWindow &a_window)`
Draws the body to the render window.
- `void Debugger ()`
debugger

Private Member Functions

- `sf::Vector2f GetCompUpperPos (Brood::BroodUI::UIElement *a_comp)`
Getter function to get the bottom right coridinate of the passed component.
- `void UpdateLargestUpperPos (sf::Vector2f &a_largestUpperPos, Brood::BroodUI::UIElement *a_comp)`
updates the passed largest upper pos by comparaing it with the upper position of the passed component
- `void UpdateSmallestLowerPos (sf::Vector2f &a_smallestLowerPos, Brood::BroodUI::UIElement *a_comp)`
updates the passed smallest lower pos by comparaing it with the lower position of the passed component

Private Attributes

- `bool m_isCurrFront`
- `std::string m_frontBgFileName`
- `std::string m_backBgFileName`
- `Brood::Application::Components::CardInfo * m_cardInfoToDisplay`
- `Brood::BroodUI::TextBox * m_TxtFrontTimePromt`
- `Brood::BroodUI::TextBox * m_TxtFrontTimeValue`
- `Brood::BroodUI::TextBox * m_TxtUpPromt`
- `Brood::BroodUI::TextBox * m_TxtUpValue`
- `Brood::BroodUI::TextBox * m_TxtDownPromt`
- `Brood::BroodUI::TextBox * m_TxtDownValue`
- `Brood::BroodUI::Button * m_BtnTurnCard`
- `Brood::BroodUI::TextBox * m_TxtBackTimePromt`
- `Brood::BroodUI::TextBox * m_TxtBackTimeValue`
- `Brood::BroodUI::TextBox * m_TxtQuestionPromt`
- `Brood::BroodUI::TextBox * m_TxtQuestionValue`
- `Brood::BroodUI::TextBox * m_TxtUserAnswerPromt`
- `Brood::BroodUI::TextBox * m_TxtUserAnswerValue`
- `Brood::BroodUI::Button * m_BtnSubmit`

Additional Inherited Members

9.18.1 Constructor & Destructor Documentation

9.18.1.1 `DisplayCard()` [1/2]

```
Brood::Application::Components::DisplayCard::DisplayCard ( )
```

Default Constructor.

9.18.1.2 `~DisplayCard()`

```
Brood::Application::Components::DisplayCard::~DisplayCard ( )
```

default destructor

9.18.1.3 `DisplayCard()` [2/2]

```
Brood::Application::Components::DisplayCard::DisplayCard ( const DisplayCard & a_other )
```

Copy constructor.

Parameters

<code>a_other</code>	reference to the other display card which is used to copy the data form
----------------------	---

9.18.2 Member Function Documentation

9.18.2.1 `Debugger()`

```
void Brood::Application::Components::DisplayCard::Debugger ( ) [virtual]
```

debugger

@virtual

Reimplemented from [Brood::BroodUI::Button](#).

9.18.2.2 Draw()

```
void Brood::Application::Components::DisplayCard::Draw (
    sf::RenderWindow & a_window ) [virtual]
```

Draws the body to the render window.

@virtual

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.18.2.3 GetAnswerPromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetAnswerPromptPtr ( )
```

Getter function to get the pointer to the user answer prompt ui element.

Returns

pointer to the user answer prompt ui element

9.18.2.4 GetAnswerValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetAnswerValuePtr ( )
```

Getter function to get the pointer to the user answer value ui element.

Returns

pointer to the user answer value ui element

9.18.2.5 GetBackBgFileName()

```
std::string Brood::Application::Components::DisplayCard::GetBackBgFileName ( )
```

Getter function to get a back background filename.

Returns

back background filename

9.18.2.6 GetBackTimePromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetBackTimePromptPtr ( )
```

Getter function to get the pointer to the back time prompt ui element.

Returns

pointer to the back time prompt ui element

9.18.2.7 GetBackTimeValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetBackTimeValuePtr ( )
```

Getter function to get the pointer to the back time value ui element.

Returns

pointer to the back time value ui element

9.18.2.8 GetCompUpperPos()

```
sf::Vector2f Brood::Application::Components::DisplayCard::GetCompUpperPos ( Brood::BroodUI::UIElement * a_comp ) [private]
```

Getter function to get the bottom right coridinate of the passed component.

Parameters

<i>a_comp</i>	component whose bottom right coridinate is to be found
---------------	--

Returns

the bottom right coridinate of the passed component

9.18.2.9 GetDataToSave()

```
Brood::Application::Data::ST_DisplayCardData Brood::Application::Components::DisplayCard::GetDataToSave ( )
```

creates and returns displayCard data struct

Returns

displayCard data struct with the displayCard data in it

9.18.2.10 GetDownPromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetDownPromptPtr ( )
```

Getter function to get the pointer to the down prompt ui element.

Returns

pointer to the down prompt ui element

9.18.2.11 GetDownValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetDownValuePtr ( )
```

Getter function to get the pointer to the down value ui element.

Returns

pointer to the down value ui element

9.18.2.12 GetFrontBgFileName()

```
std::string Brood::Application::Components::DisplayCard::GetFrontBgFileName ( )
```

Getter function to get a front background filename.

Returns

front background filename

9.18.2.13 GetFrontTimePromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetFrontTimePromptPtr ( )
```

Getter function to get the pointer to the frontTime prompt ui element.

Returns

pointer to the frontTime prompt ui element

9.18.2.14 GetFrontTimeValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetFrontTimeValuePtr ( )
```

Getter function to get the pointer to the frontTime value ui element.

Returns

pointer to the frontTime value ui element

9.18.2.15 GetFurtherstBottomRightCordinate()

```
sf::Vector2f Brood::Application::Components::DisplayCard::GetFurtherstBottomRightCordinate ( )
```

gets the bottom right cordinate of the box that encloses all the child component

Returns

gets the bottom right cordinate of the box that encloses all the child component

9.18.2.16 GetNearestTopLeftCordinate()

```
sf::Vector2f Brood::Application::Components::DisplayCard::GetNearestTopLeftCordinate ( )
```

gets top left cordinate of the box that encloses all the child component

Returns

gets top left cordinate of the box that encloses all the child component

9.18.2.17 GetQuestionPromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetQuestionPromptPtr ( )
```

Getter function to get the pointer to the answer prompt ui element.

Returns

pointer to the answer prompt ui element

9.18.2.18 GetQuestionValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetQuestionValuePtr ( )
```

Getter function to get the pointer to the answer value ui element.

Returns

pointer to the answer value ui element

9.18.2.19 GetSubmitPtr()

```
Brood::BroodUI::Button * Brood::Application::Components::DisplayCard::GetSubmitPtr ( )
```

Getter function to get the pointer to the submit ui element.

Returns

pointer to the submit ui element

9.18.2.20 GetTurnCardPtr()

```
Brood::BroodUI::Button * Brood::Application::Components::DisplayCard::GetTurnCardPtr ( )
```

Getter function to get the pointer to the turn card ui element.

Returns

pointer to the turn card ui element

9.18.2.21 GetUpPromptPtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetUpPromptPtr ( )
```

Getter function to get the pointer to the up prompt ui element.

Returns

pointer to the up prompt ui element

9.18.2.22 GetUpValuePtr()

```
Brood::BroodUI::TextBox * Brood::Application::Components::DisplayCard::GetUpValuePtr ( )
```

Getter function to get the pointer to the up value ui element.

Returns

pointer to the up value ui element

9.18.2.23 InializeDisplayCard()

```
void Brood::Application::Components::DisplayCard::InializeDisplayCard ( )
```

Initializes the display card.

9.18.2.24 InitializeDisplayCard()

```
void Brood::Application::Components::DisplayCard::InitializeDisplayCard (   
    Brood::Application::Data::ST_DisplayCardData & a_displayCardData )
```

initializes the displayCard with the passed data

Parameters

<i>a_displayCardData</i>	reference of the card info data
--------------------------	---------------------------------

9.18.2.25 LoadDataFromFile()

```
void Brood::Application::Components::DisplayCard::LoadDataFromFile (   
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the [DisplayCard](#) and its path data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.18.2.26 operator=()

```
Brood::Application::Components::DisplayCard & Brood::Application::Components::DisplayCard<-
::operator= (
    const DisplayCard & a_other )
```

assignment operator

Parameters

<i>a_other</i>	reference to the other display card which is used to copy the data form
----------------	---

Returns

pointer to this display card

9.18.2.27 SaveDataToFile()

```
void Brood::Application::Components::DisplayCard::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the [DisplayCard](#) data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.18.2.28 SetBackBgFileName()

```
bool Brood::Application::Components::DisplayCard::SetBackBgFileName (
    std::string a_backBgFilename )
```

Setter function to set a back background filename.

Parameters

<i>a_backBgFilename</i>	back background filename
-------------------------	--------------------------

Returns

true if texture was successfully set

9.18.2.29 SetCardFront()

```
void Brood::Application::Components::DisplayCard::SetCardFront (
    bool a_currFront )
```

sets m_currFront

Parameters

<i>a_currFront</i>	If current bg is using front then toggles it to use back
--------------------	--

9.18.2.30 SetCardInfoToDisplay()

```
void Brood::Application::Components::DisplayCard::SetCardInfoToDisplay (
    Brood::Application::Components::CardInfo * a_cardInfoToDisplay )
```

setter function to set the card info that will be used to display

Parameters

<i>a_cardInfoToDisplay</i>	pointer to the cardInfo object
----------------------------	--------------------------------

9.18.2.31 SetFrontBgFileName()

```
bool Brood::Application::Components::DisplayCard::SetFrontBgFileName (
    std::string a_frontBgFilename )
```

Setter function to set a front background filename.

Parameters

<i>a_frontBgFilename</i>	front background filename
--------------------------	---------------------------

Returns

true if texture was successfully set

9.18.2.32 ToggleCardFace()

```
void Brood::Application::Components::DisplayCard::ToggleCardFace ( )
```

toggles the background

If current bg is using front then toggles it to use back

9.18.2.33 UpdateDisplayedText()

```
void Brood::Application::Components::DisplayCard::UpdateDisplayedText ( )
```

updates the displayed text from the saved cardInfo

9.18.2.34 UpdateLargestUpperPos()

```
void Brood::Application::Components::DisplayCard::UpdateLargestUpperPos (
    sf::Vector2f & a_largestUpperPos,
    Brood::BroodUI::UIElement * a_comp ) [private]
```

updates the passed largest upper pos by comparaing it with the upper position of the passed component

Parameters

<i>a_largestUpperPos</i>	largest upper pos passed by reference
<i>a_comp</i>	component whose upperPos is to be compared to

9.18.2.35 UpdateSmallestLowerPos()

```
void Brood::Application::Components::DisplayCard::UpdateSmallestLowerPos (
    sf::Vector2f & a_smallestLowerPos,
    Brood::BroodUI::UIElement * a_comp ) [private]
```

updates the passed smallest lower pos by comparaing it with the lower position of the passed component

Parameters

<i>a_smallestLowerPos</i>	smallest lower pos passed by reference
<i>a_comp</i>	component whose lowerPos is to be compared to

9.18.3 Member Data Documentation**9.18.3.1 m_backBgFileName**

```
std::string Brood::Application::Components::DisplayCard::m_backBgFileName [private]
```

9.18.3.2 m_BtnSubmit

```
Brood::BroodUI::Button* Brood::Application::Components::DisplayCard::m_BtnSubmit [private]
```

9.18.3.3 m_BtnTurnCard

```
Brood::BroodUI::Button* Brood::Application::Components::DisplayCard::m_BtnTurnCard [private]
```

9.18.3.4 m_cardInfoToDisplay

```
Brood::Application::Components::CardInfo* Brood::Application::Components::DisplayCard::m_card←  
InfoToDisplay [private]
```

9.18.3.5 m_frontBgFileName

```
std::string Brood::Application::Components::DisplayCard::m_frontBgFileName [private]
```

9.18.3.6 m_isCurrFront

```
bool Brood::Application::Components::DisplayCard::m_isCurrFront [private]
```

9.18.3.7 m_TxtBackTimePromt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtBackTimePromt [private]
```

9.18.3.8 m_TxtBackTimeValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtBackTimeValue [private]
```

9.18.3.9 m_TxtDownPrompt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtDownPrompt [private]
```

9.18.3.10 m_TxtDownValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtDownValue [private]
```

9.18.3.11 m_TxtFrontTimePromt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtFrontTimePromt [private]
```

9.18.3.12 m_TxtFrontTimeValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtFrontTimeValue [private]
```

9.18.3.13 m_TxtQuestionPrompt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtQuestionPrompt [private]
```

9.18.3.14 m_TxtQuestionValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtQuestionValue [private]
```

9.18.3.15 m_TxtUpPromt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtUpPromt [private]
```

9.18.3.16 m_TxtUpValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtUpValue [private]
```

9.18.3.17 m_TxtUserAnswerPrompt

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtUserAnswerPrompt [private]
```

9.18.3.18 m_TxtUserAnswerValue

```
Brood::BroodUI::TextBox* Brood::Application::Components::DisplayCard::m_TxtUserAnswerValue [private]
```

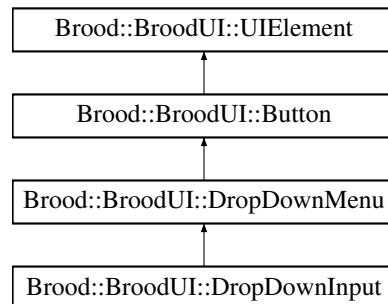
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DisplayCard.h
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DisplayCard.cpp

9.19 Brood::BroodUI::DropDownInput Class Reference

```
#include <DropDownInput.h>
```

Inheritance diagram for Brood::BroodUI::DropDownInput:



Public Member Functions

- `DropDownInput (Brood::BroodUI::UIElement *a_parentPtr=nullptr, Brood::BroodUI::ENUM_UIType a_enumType=Brood::BroodUI::ENUM_UIType::UI_dropDownInput)`
Default Constructor.
- `virtual ~DropDownInput ()`
Default Destructor.
- `DropDownInput (const DropDownInput &a_otherElement)`
Copy constructor.
- `Brood::BroodUI::DropDownInput & operator= (const DropDownInput &a_otherButton)`
assignment operator
- `virtual void AddItemToMenu (std::string a_menuName, sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::white)`
Function to add menu item to the dropdown menu.
- `virtual void AddItemToMenu (Brood::BroodUI::Button *a_buttonPtrToAdd, bool a_createNew=true)`
Function to add menu item to the dropdown menu.

Additional Inherited Members

9.19.1 Constructor & Destructor Documentation

9.19.1.1 DropDownInput() [1/2]

```
DropDownInput::DropDownInput (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr,
    Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_dropDownInput
)
```

Default Constructor.

Initializes the [DropDownInput](#) object

Parameters

<i>m_parentPtr</i>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------	---

9.19.1.2 ~DropDownInput()

```
DropDownInput::~DropDownInput ( ) [virtual]
```

Default Destructor.

9.19.1.3 DropDownInput() [2/2]

```
DropDownInput::DropDownInput (
    const DropDownInput & a_otherElement )
```

Copy constructor.

Parameters

<i>a_otherElement</i>	reference to the dropdownInput which is used to copy the data form
-----------------------	--

9.19.2 Member Function Documentation

9.19.2.1 AddItemToMenu() [1/2]

```
void DropDownInput::AddItemToMenu (
    Brood::BroodUI::Button * a_buttonPtrToAdd,
    bool a_createNew = true ) [virtual]
```

Function to add menu item to the dropdown menu.

if a_createNew is true then create a new button using the passed item then adds the new button

It dynamically allocates memory for the newly created item

Parameters

<i>a_buttonPtrToAdd</i>	pointer to the button element which is used to add
<i>a_createNew</i>	if a_createNew is true then create a new button using the passed item then adds the new button

Reimplemented from [Brood::BroodUI::DropDownMenu](#).

9.19.2.2 AddItemToMenu() [2/2]

```
void DropDownInput::AddItemToMenu (
    std::string a_menuName,
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_White
) [virtual]
```

Function to add menu item to the dropdown menu.

It dynamically allocates memory for the item

Parameters

<i>a_menuName</i>	name of the item
<i>a_color</i>	font color -> default sf::Color::White

Reimplemented from [Brood::BroodUI::DropDownMenu](#).

9.19.2.3 operator=()

```
Brood::BroodUI::DropDownInput & DropDownInput::operator= (
    const DropDownInput & a_otherButton )
```

assignment operator

Parameters

<i>a_otherElement</i>	reference to the dropdown menu which is used to copy the data form
-----------------------	--

Returns

pointer to this element

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownInput.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownInput.cpp](#)

9.20 DropDownInput Class Reference

A UI elemnt that represent a drop down menu.

```
#include "DropDownInput.h"
```

9.20.1 Detailed Description

A UI elemnt that represent a drop down menu.

A class derived from [DropDownMenu](#) class. It contains a list of menus inside a single element. The list of items are left aligned and below the menu title. the size of the items is depended on the longest item name.

Note

This class differs from the [DropDownMenu](#) class as for this class the title that is used in [DropDownMenu](#) is changed to reflect the selected item. So it is important to add line of code below when your item's DoElement() returns true.

```
// see the example case for more information
myDropDownInput.SetText( itemList.at( i )->GetText() );
```

9.20.1.0.1 Example Case // window object

```
sf::RenderWindow window( sf::VideoMode( 500, 500 ), "BroodMaker" );
// event object
sf::Event events;
// creating a DropDownInput object
Brood::BroodUI::DropDownInput myDropDownInput;
// setting size and position
myDropDownInput.SetBodySize( 150, 50 );
myDropDownInput.SetBodyPosition( 265, 120 );
// setting the font which is required to display the text
myDropDownInput.SetFont( &font );
// setting the font size. It should be less than the body height by 2 units.
// @see Brood::BroodUI::Textbox::SetFont() for more information
myDropDownInput.SetFontSize( 20 );
// setting body color
myDropDownInput.SetBodyColor( sf::Color::Green );
// adding 2 items to the menu
myDropDownInput.AddItemToMenu( "1st item" );
myDropDownInput.AddItemToMenu( "2st item" );
myDropDownInput.AddItemToMenu( "longest of all item" );
//app loop
while( !exit )
{
    // event loop

    // logic
    // checking if the logics of the element is to be executed or not
```

```

if(myDropDownInput.DoElement())
{
    std::cout << "myDropDownInput Pressed" << std::endl;
}
if( myDropDownInput.IsSelected() )
{
    auto itemList = myDropDownInput.GetItemList();
    if( !itemList.empty() )
    {
        // checking if the logics of the items is to be executed or not
        for( int i = 0; i < itemList.size(); ++i )
        {
            if( itemList.at( i )->DoElement() )
            {
                // setting the elements name as the item's name
                // this is important so that the changes can be reflected in the
                // title
                myDropDownInput.SetText( itemList.at( i )->GetText() );

                // executing the function
                std::cerr << "item at " << i << " Pressed with ID " <<
                    itemList.at( i )->GetText() << std::endl;
            }
        }
    }
    // rendering
    myDropDown.Draw( window );
}

```

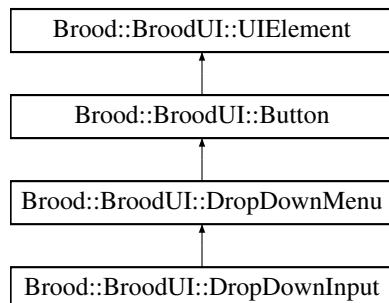
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownInput.h](#)

9.21 Brood::BroodUI::DropDownMenu Class Reference

```
#include <DropDownMenu.h>
```

Inheritance diagram for Brood::BroodUI::DropDownMenu:



Public Member Functions

- [DropDownMenu \(Brood::BroodUI::UIElement *a_parentPtr=nullptr, Brood::BroodUI::ENUM_UIType a_enumType=Brood::BroodUI::ENUM_UIType::UI_dropDownMenu\)](#)
Default Constructor.
- virtual [~DropDownMenu \(\)](#)
Default Destructor.
- [DropDownMenu \(const DropDownMenu &a_otherElement\)](#)
Copy constructor.
- [Brood::BroodUI::DropDownMenu & operator= \(const DropDownMenu &a_otherElement\)](#)

- `assignment operator`
- `std::vector< Brood::BroodUI::Button * > & GetItemList ()`

Getter funciton to get the item List.
- `void SetBodySize (sf::Vector2f a_eachItemSize) override`

Setter function to set the DropDownMenu's Size.
- `void SetBodySize (float a_itemSizeX, float a_itemSizeY) override`

Setter function to set the DropDownMenu's size.
- `void SetBodyPosition (sf::Vector2f a_pos, bool a_relativeToParent=false) override`

Setter function to set the DropDownMenu's Position.
- `void SetBodyPosition (float a_posX, float a_posY, bool a_relativeToParent=false) override`

Setter function to set the DropDownMenu's Position.
- `void SetFont (sf::Font *a_font)`

Setter function to set the font for the menu.
- `virtual void SetFontColor (sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_White)`

setter function to set the Font color
- `void SetFontSize (unsigned a_fontSize=12)`

setter funciton to set the Font size of the menu title and its items
- `voidSetText (std::string a_text="")`

Setter function to set the text for DropDownMenu.
- `virtual void SetBodyColor (sf::Color a_bodyColor)`

Setter function to set the element body color.
- `virtual void SetActiveOverlayColor (sf::Color a_color)`

Setter function to set the element's active color.
- `virtual void SetHotOverlayColor (sf::Color a_color)`

Setter function to set the element's hot color.
- `virtual void AddItemToMenu (std::string a_menuName, sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_White)`

Function to add menu item to the dropdown menu.
- `virtual void AddItemToMenu (Brood::BroodUI::Button *a_buttonPtrToAdd, bool a_createNew=true)`

Function to add menu item to the dropdown menu.
- `virtual bool DoElement () override`

checks if the logics of the element is to be executed or not
- `virtual void Draw (sf::RenderWindow &a_window) override`

Draw function.
- `virtual void Debugger ()`

changes the text/texture of the element to its the element ID when debug is on and changes it back to the text/texture it had before debugging when debug is off.

Protected Member Functions

- `void SetItemPos (int a_itemIndex)`

helper funciton to position the item at given index correctly in the drop down list.
- `void SetItemSize (int a_itemIndex)`

helper funciton to size the items correctly in the drop down list
- `void SetEachItemSize ()`

Setter function to set the size of each item in the drop down menus.
- `void SetEachItemPos ()`

Setter function to set the postiong of each item in the drop down menus.

Protected Attributes

- std::vector< Brood::BroodUI::Button * > m_items
stores the list of menus
- size_t m_maxItemLength
holds the length of the longest menu item

Additional Inherited Members

9.21.1 Constructor & Destructor Documentation

9.21.1.1 DropDownMenu() [1/2]

```
DropDownMenu::DropDownMenu (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr,
    Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_dropDownMenu
)
```

Default Constructor.

Initializes the `DropDownMenu` object

Parameters

<code>m_parentPtr</code>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------------	---

9.21.1.2 ~DropDownMenu()

```
DropDownMenu::~DropDownMenu ( ) [virtual]
```

Default Destructor.

9.21.1.3 DropDownMenu() [2/2]

```
DropDownMenu::DropDownMenu (
    const DropDownMenu & a_otherElement )
```

Copy constructor.

Parameters

<code>a_otherElement</code>	reference to the dropdownMenu which is used to copy the data form
-----------------------------	---

9.21.2 Member Function Documentation

9.21.2.1 AddItemToMenu() [1/2]

```
void DropDownMenu::AddItemToMenu (
    Brood::BroodUI::Button * a_buttonPtrToAdd,
    bool a_createNew = true ) [virtual]
```

Function to add menu item to the dropdown menu.

If `a_createNew` is true then create a new button using the passed item then adds the new button

It dynamically allocates memory for the newly created item

Parameters

<code>a_buttonPtrToAdd</code>	pointer to the button element which is used to add
<code>a_createNew</code>	if <code>a_createNew</code> is true then create a new button using the passed item then adds the new button

Reimplemented in [Brood::BroodUI::DropDownInput](#).

9.21.2.2 AddItemToMenu() [2/2]

```
void DropDownMenu::AddItemToMenu (
    std::string a_menuName,
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_White
) [virtual]
```

Function to add menu item to the dropdown menu.

It dynamically allocates memory for the item

Parameters

<code>a_menuName</code>	name of the item
<code>a_color</code>	font color -> default sf::Color::White

Reimplemented in [Brood::BroodUI::DropDownInput](#).

9.21.2.3 Debugger()

```
void DropDownMenu::Debugger () [virtual]
```

changes the text/texture of the element to its the element ID when debug is on and changes it back to the text/texture it had before debugging when debug is off.

Reimplemented from [Brood::BroodUI::Button](#).

9.21.2.4 DoElement()

```
bool DropDownMenu::DoElement ( ) [override], [virtual]
```

checks if the logics of the element is to be executed or not

It checks the mouse position and button state to determine if to execute the elements logic or not. It does this by manipulating the element selection class. It also set the overlay

Returns

true if the element's funciton is to be executed; else false

Reimplemented from [Brood::BroodUI::Button](#).

9.21.2.5 Draw()

```
void DropDownMenu::Draw ( sf::RenderWindow & a_window ) [override], [virtual]
```

Draw function.

Parameters

a_window	reference to render window
----------	----------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.6 GetItemList()

```
std::vector< Brood::BroodUI::Button * > & DropDownMenu::GetItemList ( )
```

Getter funciton to get the item List.

Returns

reference to the item list

9.21.2.7 operator=()

```
Brood::BroodUI::DropDownMenu & DropDownMenu::operator= (
    const DropDownMenu & a_otherElement )
```

assignment operator

Parameters

<i>a_otherElement</i>	reference to the dropdown menu which is used to copy the data form
-----------------------	--

Returns

pointer to this element

9.21.2.8 SetActiveOverlayColor()

```
void DropDownMenu::SetActiveOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's active color.

Parameters

<i>a_color</i>	color of the body when it is active
----------------	-------------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.9 SetBodyColor()

```
void DropDownMenu::SetBodyColor (
    sf::Color a_bodyColor ) [virtual]
```

Setter function to set the element body color.

Parameters

<i>a_bodyColor</i>	color of the body
--------------------	-------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.10 SetBodyPosition() [1/2]

```
void DropDownMenu::SetBodyPosition (
    float a_posX,
    float a_posY,
    bool a_relativeToParent = false ) [override], [virtual]
```

Setter function to set the [DropDownMenu](#)'s Position.

@virtual

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_posX</i>	x-position of the element
<i>a_posY</i>	y-position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.11 SetBodyPosition() [2/2]

```
void DropDownMenu::SetBodyPosition (
    sf::Vector2f a_pos,
    bool a_relativeToParent = false ) [override], [virtual]
```

Setter function to set the [DropDownMenu](#)'s Position.

@virtual

Parameters

<i>a_pos</i>	position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.12 SetBodySize() [1/2]

```
void DropDownMenu::SetBodySize (
    float a_sizeX,
    float a_sizeY ) [override], [virtual]
```

Setter function to set the [DropDownMenu](#)'s size.

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_sizeX</i>	length of the DropDownMenu
<i>a_sizeY</i>	width of the DropDownMenu

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.13 SetBodySize() [2/2]

```
void DropDownMenu::SetBodySize (
    sf::Vector2f a_size )  [override], [virtual]
```

Setter function to set the [DropDownMenu](#)'s Size.

Parameters

<i>a_size</i>	size of the DropDownMenu
---------------	--

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.14 SetEachItemPos()

```
void DropDownMenu::SetEachItemPos ( )  [protected]
```

Setter function to set the postiong of each item in the drop down menus.

9.21.2.15 SetEachItemSize()

```
void DropDownMenu::SetEachItemSize ( )  [protected]
```

Setter function to set the size of each item in the drop down menus.

9.21.2.16 SetFont()

```
void DropDownMenu::SetFont (
    sf::Font * a_font )
```

Setter function to set the font for the menu.

Parameters

<i>a_font</i>	pointer to a loaded font
---------------	--------------------------

9.21.2.17 SetFontColor()

```
void DropDownMenu::SetFontColor (
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_White
) [virtual]
```

setter function to set the Font color

param a_color font color -> default sf::Color::White

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.18 SetFontSize()

```
void DropDownMenu::SetFontSize (
    unsigned a_fontSize = 12 )
```

setter funciton to set the Font size of the menu title and its items

Parameters

<i>a_charSize</i>	-> size of indivisual character in the SetEditablename -> deafult 12
-------------------	--

9.21.2.19 SetHotOverlayColor()

```
void DropDownMenu::SetHotOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's hot color.

Parameters

<i>a_color</i>	color of the body when it is hot
----------------	----------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.2.20 SetItemPos()

```
void DropDownMenu::SetItemPos (
    int a_itemIndex ) [protected]
```

helper function to position the item at given index correctly in the drop down list.

Parameters

a_itemIndex	index of the item in the drop down list
-------------	---

9.21.2.21 SetItemSize()

```
void DropDownMenu::SetItemSize (
    int a_itemIndex ) [protected]
```

helper function to size the items correctly in the drop down list

Parameters

a_itemIndex	index of the item in the drop down list
-------------	---

9.21.2.22 SetText()

```
void DropDownMenu::SetText (
    std::string a_text = "" ) [virtual]
```

Setter function to set the text for [DropDownMenu](#).

Parameters

a_text	text to set the text for DropDownMenu
--------	---

Reimplemented from [Brood::BroodUI::UIElement](#).

9.21.3 Member Data Documentation

9.21.3.1 m_items

```
std::vector<Brood::BroodUI::Button*> Brood::BroodUI::DropDownMenu::m_items [protected]
```

stores the list of menus

9.21.3.2 m_maxItemLength

```
size_t Brood::BroodUI::DropDownMenu::m_maxItemLength [protected]
```

holds the length of the longest menu item

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownMenu.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownMenu.cpp](#)

9.22 DropDownMenu Class Reference

A UI element that represents a drop down menu.

```
#include "DropDownMenu.h"
```

9.22.1 Detailed Description

A UI element that represents a drop down menu.

A class derived from [Button](#) class. It contains a list of menus inside a single element. The list of items are left aligned and below the menu title. The size of the items is depended on the longest item name.

9.22.1.0.1 Example Case

```
// window object
sf::RenderWindow window( sf::VideoMode( 500, 500 ), "BroodMaker" );
// event object
sf::Event events;
// creating a dropdown object
Brood::BroodUI::DropDownMenu myDropDown;
// setting size and position
myDropDown.SetBodySize( 20, 50 );
myDropDown.SetBodyPosition( 265, 0 );
// setting the font which is required to display the text
myDropDownSetFont( &font );
// setting the font size. It should be less than the body height by 2 units.
// @see Brood::BroodUI::Textbox::SetFont()
myDropDown.SetFontSize( 20 );
// setting the menu title to "a"
myDropDown.SetText( "a" );
// setting body color
myDropDown.SetBodyColor( sf::Color::Green );
// adding 2 items to the menu
myDropDown.AddItemToMenu( "1st item" );
myDropDown.AddItemToMenu( "2st item" );
//app loop
while( !exit )
{
    // event loop

    // logic
    // checking if the logics of the element is to be executed or not
    myDropDown.DoElement();

    // checking its item if the dropdown menu was selected
    if( myDropDown.IsSelected() )
    {
        auto itemList = myDropDown.GetItemList();
        // draw its items
        // positioning the items
        if( !itemList.empty() )
        {
            // checking if the logics of the items is to be executed or not
            for( int i = 0; i < itemList.size(); ++i )
            {
                if( itemList.at( i )->DoElement() )
                {

```

```

        std::cout << "item at " << i << " Pressed" << std::endl;
    }
}
}
// rendering
myDropDown.Draw( window );
}
}

```

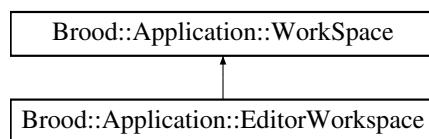
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DropDownMenu.h](#)

9.23 Brood::Application::EditorWorkspace Class Reference

```
#include <EditorWorkspace.h>
```

Inheritance diagram for Brood::Application::EditorWorkspace:



Public Member Functions

- [EditorWorkspace \(Brood::Application::Components::GameManager *aGameData\)](#)
default constructor
- [virtual ~EditorWorkspace \(\)](#)
default destructor
- [void InitializeWorkSpace \(\)](#)
initializes the component of a workspace
- [void Update \(\)](#)
all the logic for the workspace is here
- [void UpdateAllDisplayElement \(\)](#)
calls the updates all the display element for the current active editor
- [void Draw \(sf::RenderWindow &a_window\)](#)
renders to the screen
- [void Debugger \(\)](#)
Calls Debugger for all UI elements in this class.
- [void SetGameManager \(Brood::Application::Components::GameManager *aGameData\)](#)
setter function for game data

Private Member Functions

- [void InitializeEditModeTabs \(\)](#)
Initializes the edit mode tabs.
- [void UpdateEditModeTabs \(\)](#)
checks if the ribbon tabs was pressed
- [void UpdateActiveEditorIdx \(unsigned a_idx\)](#)
updates the edit Tab and selects the proper tab

Private Attributes

- sf::RectangleShape `m_sidePanel`
panel body
- unsigned `m_activeEditorIdx`
holds the index of current active editor workspace
- std::vector< Brood::BroodUI::Button * > `m_editModesTabs`
Ui for edit modes tabs.
- std::vector< Brood::Application::WorkSpace * > `m_editorWorkspaceList`
list of editor workspace

Additional Inherited Members

9.23.1 Constructor & Destructor Documentation

9.23.1.1 EditorWorkspace()

```
Brood::Application::EditorWorkspace::EditorWorkspace (
    Brood::Application::Components::GameManager * aGameData )
```

default constructor

9.23.1.2 ~EditorWorkspace()

```
Brood::Application::EditorWorkspace::~EditorWorkspace ( ) [virtual]
```

default destructor

9.23.2 Member Function Documentation

9.23.2.1 Debugger()

```
void Brood::Application::EditorWorkspace::Debugger ( ) [virtual]
```

Calls Debugger for all UI elements in this class.

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements Brood::Application::WorkSpace.

9.23.2.2 Draw()

```
void Brood::Application::EditorWorkspace::Draw (
    sf::RenderWindow & a_window ) [virtual]
```

renders to the screen

Draw function.

@virtual

Draws all the component to the screen

Implements [Brood::Application::WorkSpace](#).

9.23.2.3 InitializeEditModeTabs()

```
void Brood::Application::EditorWorkspace::InitializeEditModeTabs ( ) [private]
```

Initializes the edit mode tabs.

Initializes the edit mode tab bar which is at the top of the edit window.

It has the 4 edit modes in it: [Board](#) Edit, [Tiles](#) Edit, [Path](#) Edit, and Deck Edit

[Board](#) Edit allows for the general settings to be edited.

[Tile](#) Edit allows for tiles setting to be changed for individual tiles.

[Path](#) Edit allows for the path that the players take to be changed.

Deck Edit allows for the deck and card setting to be changed

9.23.2.4 InitializeWorkSpace()

```
void Brood::Application::EditorWorkspace::InitializeWorkSpace ( ) [virtual]
```

initializes the component of a workspace

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the EditMode tabs and the editor workspaces

Implements [Brood::Application::WorkSpace](#).

9.23.2.5 SetGameDataManager()

```
void Brood::Application::EditorWorkspace::SetGameDataManager ( 
    Brood::Application::Components::GameDataManager * a_gameDataManagerPtr ) [virtual]
```

setter function for game data

@virtual

Reimplemented from [Brood::Application::WorkSpace](#).

9.23.2.6 Update()

```
void Brood::Application::EditorWorkspace::Update ( ) [virtual]
```

all the logic for the workspace is here

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.23.2.7 UpdateActiveEditorIdx()

```
void Brood::Application::EditorWorkspace::UpdateActiveEditorIdx ( 
    unsigned a_idx ) [private]
```

updates the edit Tab and selects the proper tab

9.23.2.8 UpdateAllDisplayElement()

```
void Brood::Application::EditorWorkspace::UpdateAllDisplayElement ( ) [virtual]
```

calls the updates all the display element for the current active editor

@virtual

Implements [Brood::Application::WorkSpace](#).

9.23.2.9 UpdateEditModeTabs()

```
void Brood::Application::EditorWorkspace::UpdateEditModeTabs ( ) [private]
```

checks if the ribbon tabs was pressed

Checks to see if the ribbon tabs's logic is to be executed or not.

9.23.3 Member Data Documentation

9.23.3.1 m_activeEditorIdx

```
unsigned Brood::Application::EditorWorkspace::m_activeEditorIdx [private]
```

holds the index of current active editor workspace

9.23.3.2 m_editModesTabs

```
std::vector<Brood::BroodUI::Button*> Brood::Application::EditorWorkspace::m_editModesTabs  
[private]
```

Ui for edit modes tabs.

9.23.3.3 m_editorWorkspaceList

```
std::vector<Brood::Application::WorkSpace*> Brood::Application::EditorWorkspace::m_editorWork-  
spaceList [private]
```

list of editor workspace

9.23.3.4 m_sidePanel

```
sf::RectangleShape Brood::Application::EditorWorkspace::m_sidePanel [private]
```

panel body

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[EditorWorkspace.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[EditorWorkspace.cpp](#)

9.24 Brood::BroodUI::ElementSelection Class Reference

```
#include <ElementSelection.h>
```

Static Public Member Functions

- static void `SetHotElementIdPtr (Brood::BroodUI::Id *a_idPtr)`
Setter function to set passed UI id as the hot element.
- static void `SetAlmostActiveElementIdPtr (Brood::BroodUI::Id *a_idPtr)`
Setter function to set the passed element id as active element.
- static void `SetCurrActiveElementIdPtr (Brood::BroodUI::Id *a_idPtr)`
Setter function sets the passed element id as current active element.
- static void `SetLastActiveElementIdPtr (Brood::BroodUI::Id *a_idPtr)`
Setter function sets the passed element id as last active element.
- static void `SetHotElementIdPtrFlag (bool a_found)`
Setter function to hot element flag if a hot element is found for a given frame.
- static `Brood::BroodUI::Id * GetHotElementIdPtr ()`
Getter function to get the pointer to hot element.
- static `Brood::BroodUI::Id * GetAlmostActiveElementIdPtr ()`
Getter function to gets the pointer to active element.
- static `Brood::BroodUI::Id * GetCurrActiveElementIdPtr ()`
Getter function to gets the pointer to current active element.
- static `Brood::BroodUI::Id * GetLastActiveElementIdPtr ()`
Getter function to gets the pointer to current active element.
- static bool `GetHotElementIdFlag ()`
Getter function gets the state of the m_hotElementFlag.
- static void `Debugger ()`
Function to print the id of all the element selection pointer, namely: HotElement, AlmostActiveElement, CurrActiveElement, and LastActiveElement.
- static void `DebugPrintSelectedElementID ()`
Print the selected Element ID if any.

Static Private Attributes

- static `Brood::BroodUI::Id * m_hotElementIdPtr = nullptr`
stores the ptr to element id when the mouse is hovering over it
- static bool `m_hotElementFlag = false`
is set to true as soon as hot element is found for a given frame
- static `Brood::BroodUI::Id * m_almostActiveElementIdPtr = nullptr`
stores the ptr element id when the mouse is pressing on it
- static `Brood::BroodUI::Id * m_currActiveElementIdPtr = nullptr`
- static `Brood::BroodUI::Id * m_lastActiveElementIdPtr = nullptr`
stores the pointer to the last active element

9.24.1 Member Function Documentation

9.24.1.1 Debugger()

```
void ElementSelection::Debugger ( ) [static]
```

Function to print the id of all the element selection pointer, namely: HotElement, AlmostActiveElement, CurrActive←Element, and LastActiveElement.

9.24.1.2 DebugPrintSelectedElementID()

```
void ElementSelection::DebugPrintSelectedElementID ( ) [static]
```

Print the selected Element ID if any.

9.24.1.3 GetAlmostActiveElementIdPtr()

```
Brood::BroodUI::Id * ElementSelection::GetAlmostActiveElementIdPtr ( ) [static]
```

Getter function to gets the pointer to active element.

Returns

the pointer to active element

9.24.1.4 GetCurrActiveElementIdPtr()

```
Brood::BroodUI::Id * ElementSelection::GetCurrActiveElementIdPtr ( ) [static]
```

Getter function to gets the pointer to current active element.

Returns

the pointer to current active element

9.24.1.5 GetHotElementIdFlag()

```
bool ElementSelection::GetHotElementIdFlag ( ) [static]
```

Getter function gets the state of the m_hotElementFlag.

Returns

state of the m_hotElementFlag

9.24.1.6 GetHotElementIdPtr()

```
Brood::BroodUI::Id * ElementSelection::GetHotElementIdPtr ( ) [static]
```

Getter function to get the pointer to hot element.

Returns

the pointer to hot element

9.24.1.7 GetLastActiveElementIdPtr()

```
Brood::BroodUI::Id * ElementSelection::GetLastActiveElementIdPtr ( ) [static]
```

Getter function to gets the pointer to current active element.

Returns

the pointer to current active element

9.24.1.8 SetAlmostActiveElementIdPtr()

```
void ElementSelection::SetAlmostActiveElementIdPtr ( Brood::BroodUI::Id * a_idPtr ) [static]
```

Setter function to set the passed element id as active element.

Parameters

a_idPtr	pointer to UI_ID of the selected element
---------	--

9.24.1.9 SetCurrActiveElementIdPtr()

```
void ElementSelection::SetCurrActiveElementIdPtr ( Brood::BroodUI::Id * a_idPtr ) [static]
```

Setter function sets the passed element id as current active element.

Parameters

a_idPtr	pointer to UI_ID of the selected element
---------	--

9.24.1.10 SetHotElementIdPtr()

```
void ElementSelection::SetHotElementIdPtr (
    Brood::BroodUI::Id * a_idPtr ) [static]
```

Setter function to set passed UI id as the hot element.

Parameters

a_idPtr	pointer to UI_ID of the element the mouse is hovering over
---------	--

9.24.1.11 SetHotElementIdPtrFlag()

```
void ElementSelection::SetHotElementIdPtrFlag (
    bool a_found ) [static]
```

Setter function to hot element flag if a hot element is found for a given frame.

It is set only once for any given frame.

Parameters

a_found	true if a hot element found for a given frame
---------	---

9.24.1.12 SetLastActiveElementIdPtr()

```
void ElementSelection::SetLastActiveElementIdPtr (
    Brood::BroodUI::Id * a_idPtr ) [static]
```

Setter function sets the passed element id as last active element.

Parameters

a_idPtr	pointer to UI_ID of the selected element
---------	--

9.24.2 Member Data Documentation

9.24.2.1 m_almostActiveElementIdPtr

```
Brood::BroodUI::Id * ElementSelection::m_almostActiveElementIdPtr = nullptr [static], [private]
```

stores the ptr element id when the mouse is pressing on it

9.24.2.2 m_currActiveElementIdPtr

```
Brood::BroodUI::Id * ElementSelection::m_currActiveElementIdPtr = nullptr [static], [private]
```

stores the ptr to element id which should be selected even when the mouse is not over it. untill other elements are pressed after the LMB is released and util new element is selected

9.24.2.3 m_hotElementFlag

```
bool ElementSelection::m_hotElementFlag = false [static], [private]
```

is set to true as soon as hot element is found for a given frame

9.24.2.4 m_hotElementIdPtr

```
Brood::BroodUI::Id * ElementSelection::m_hotElementIdPtr = nullptr [static], [private]
```

stores the ptr to element id when the mouse is hovering over it

9.24.2.5 m_lastActiveElementIdPtr

```
Brood::BroodUI::Id * ElementSelection::m_lastActiveElementIdPtr = nullptr [static], [private]
```

stores the pointer to the last active element

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[ElementSelection.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[ElementSelection.cpp](#)

9.25 ElementSelection Class Reference

A static class to store which element is selected and what could be selected next.

```
#include "ElementSelection.h"
```

9.25.1 Detailed Description

A static class to store which element is selected and what could be selected next.

It is a static class and can be accessed for anywhere in the code base. It keeps stores the currently active element—`m_currActiveElement`, which element is in the verge of being active—`m_activeElement`, and which element is the mouse cursor over—`m_hotElement`.

The main difference between `currActiveElement` and `almostActiveElement` is that a element can be set as `currActiveElement` only when the left mouse button is released on top the element, an `almostActiveElement` is set when the left mouse button is pressed but has not be released yet.

To prevent multiple element selection—which can happen when more than one element overlaps, for example open menubar may cover other elements below it—it has a `m_hotElementFlag` member variable which gets set to true as soon as the first element under the mouse cursor is found. So it is suggested to check for elements starting from the top to bottom and going form left to right.

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[ElementSelection.h](#)

9.26 Brood::Application:: FileAccess Class Reference

```
#include <FileAccess.h>
```

Public Member Functions

- `FileAccess ()`
default constructor
- `~FileAccess ()`
default destructor
- `bool OpenFile (std::string a_fullfileName)`
opens the passed file in read and write mode
- `bool CreateFile (std::string a_fullfileName)`
- `bool GetNextLine (std::string &a_line)`
Get the next line from the file.
- `void Rewind ()`
points the pointer to beginning of the file
- `void RemoveAllContent ()`
removes all content form the file and points the pointer to beginning of the file
- `void WriteOneLineToFile (const std::string &a_line)`
writes a line to the the file.
- `bool CheckEOF ()`
checks if the end of file is reached or not

Private Attributes

- `std::fstream m_fileHandler`
file handler
- `std::filesystem::path m_filePath`

9.26.1 Constructor & Destructor Documentation

9.26.1.1 FileAccess()

```
Brood::Application::FileAccess::FileAccess ( )
```

default constructor

9.26.1.2 ~FileAccess()

```
Brood::Application::FileAccess::~FileAccess ( )
```

default destructor

9.26.2 Member Function Documentation

9.26.2.1 CheckEOF()

```
bool Brood::Application::FileAccess::CheckEOF ( )
```

checks if the end of file is reached or not

Returns

true if file pointer is pointed at end of file

9.26.2.2 CreateFile()

```
bool Brood::Application::FileAccess::CreateFile (
    std::string a_fullfileName )
```

9.26.2.3 GetNextLine()

```
bool Brood::Application::FileAccess::GetNextLine (
    std::string & a_line )
```

Get the next line from the file.

Parameters

<i>is</i>	the string where the extracted line is stored
-----------	---

Returns

true if there was one.

9.26.2.4 OpenFile()

```
bool Brood::Application::FileAccess::OpenFile (
    std::string a_fullfileName )
```

opens the passed file in read and write mode

Parameters

<i>a_fullfileName</i>	filepath to the file that is to be opened
-----------------------	---

Returns

true on successfull opening; else false

9.26.2.5 RemoveAllContent()

```
void Brood::Application::FileAccess::RemoveAllContent ( )
```

removes all content form the file and points the pointer to beginning of the file

9.26.2.6 Rewind()

```
void Brood::Application::FileAccess::Rewind ( )
```

points the pointer to beginning of the file

9.26.2.7 WriteOneLineToFile()

```
void Brood::Application::FileAccess::WriteOneLineToFile (
    const std::string & a_line )
```

writes a line to the the file.

Parameters

<i>is</i>	the string that is added to the file
-----------	--------------------------------------

9.26.3 Member Data Documentation**9.26.3.1 m_fileHandler**

```
std::fstream Brood::Application::FileAccess::m_fileHandler [private]
```

file handler

9.26.3.2 m_filePath

```
std::filesystem::path Brood::Application::FileAccess::m_filePath [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[FileAccess.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[FileAccess.cpp](#)

9.27 Brood::Application::Components::GameDataManager Class Reference

```
#include <GameDataManager.h>
```

Public Member Functions

- [GameDataManager \(\)](#)
default constructor
- [~GameDataManager \(\)](#)
default destructor
- void [InitializeGameDataManager \(\)](#)
Initializes the member variables.
- void [Draw \(sf::RenderWindow &a_window\)](#)
Draws all the component to the screen.
- void [Debugger \(\)](#)
debugger
- void [InitializeGameManager \(Brood::Application::Data::ST_GameData &a_gameDataManager\)](#)
initializes the game data manager with the passed data
- [Brood::Application::Data::ST_GameData GetDataToSave \(\)](#)

- void [SaveDataToFile \(Brood::Application:: FileAccess *a_fileAccessPtr\)](#)
saves the game data manager data to passed file
- void [LoadDataFromFile \(Brood::Application:: FileAccess *a_fileAccessPtr\)](#)
loads the game data manager data from passed file
- void [SetGameTitle \(std::string a_gameTitle\)](#)
setter funciton to set the game title for the game
- std::string [GetGameTitle \(\)](#)
Getter funciton to get a saved game title of the game.
- Brood::Application::Components::Board * [GetBoardPtr \(\)](#)
Getter funciton to get a pointer to board of the game.
- Brood::Application::Components::Dice * [GetDicePtr \(\)](#)
Getter funciton to get a pointer to dice of the game.
- Brood::Application::Components::PlayerManager * [GetPlayerManagerPtr \(\)](#)
Getter funciton to get a pointer to player mangaer of the game.
- Brood::Application::Components::DeckManager * [GetDeckManagerPtr \(\)](#)
Getter funciton to get a pointer to Deck mangaer of the game.
- Brood::Application::Components::DisplayCard * [GetDisplayCardPtr \(\)](#)
Getter funciton to get a pointer to display card of the game.

Private Attributes

- std::string [m_gameTitle](#)
- Brood::Application::Components::Board [m_board](#)
board for the game
- Brood::Application::Components::Dice [m_dice](#)
dice for the game
- Brood::Application::Components::PlayerManager [m_playerManager](#)
player mangaer for the game
- Brood::Application::Components::DeckManager [m_deckManager](#)
Deck mangaer for the game.
- Brood::Application::Components::DisplayCard [m_displayCard](#)

9.27.1 Constructor & Destructor Documentation

9.27.1.1 GameManager()

```
GameManager::GameManager ( )
```

default constructor

9.27.1.2 ~GameManager()

```
GameManager::~GameManager ( )
```

default destructor

9.27.2 Member Function Documentation

9.27.2.1 Debugger()

```
void GameDataManager::Debugger ( )
```

debugger

debugger funciton

@virtual

This function helps in debugging the UI elements.

9.27.2.2 Draw()

```
void GameDataManager::Draw ( sf::RenderWindow & a_window )
```

Draws all the component to the screen.

Draw funciton.

@virtual

Draws all the component to the screen

9.27.2.3 GetBoardPtr()

```
Brood::Application::Components::Board * GameDataManager::GetBoardPtr ( )
```

Getter funciton to get a pointer to board of the game.

Returns

pointer pointer to the board

9.27.2.4 GetDataToSave()

```
Brood::Application::Data::ST_GameData GameDataManager::GetDataToSave ( )
```

9.27.2.5 GetDeckManagerPtr()

```
Brood::Application::Components::DeckManager * GameDataManager::GetDeckManagerPtr ( )
```

Getter funciton to get a pointer to `Deck` mangaer of the game.

Getter funciton to get a pointer to deck mangaer of the game.

Returns

pointer pointer to the deck mangaer

9.27.2.6 GetDicePtr()

```
Brood::Application::Components::Dice * GameDataManager::GetDicePtr ( )
```

Getter funciton to get a pointer to dice of the game.

Returns

pointer pointer to the dice

9.27.2.7 GetDisplayCardPtr()

```
Brood::Application::Components::DisplayCard * GameDataManager::GetDisplayCardPtr ( )
```

Getter funciton to get a pointer to display card of the game.

Returns

pointer pointer to the deck mangaer

9.27.2.8 GetGameTitle()

```
std::string GameDataManager::GetGameTitle ( )
```

Getter funciton to get a saved game title of the game.

Returns

game title

9.27.2.9 GetPlayerManagerPtr()

```
Brood::Application::Components::PlayerManager * GameDataManager::GetPlayerManagerPtr ( )
```

Getter funciton to get a pointer to player mangaer of the game.

Returns

pointer pointer to the player mangaer

9.27.2.10 InitializeGameDataManager()

```
void GameDataManager::InitializeGameDataManager (   
    Brood::Application::Data::ST_GameData & a_gameDataManager )
```

initializes the game data manager with the passed data

Parameters

<code>a_gameDataManager</code>	reference of the game data
--------------------------------	----------------------------

9.27.2.11 InitializeGameDataManager()

```
void GameDataManager::InitializeGameDataManager ( )
```

Initializes the member variables.

9.27.2.12 LoadDataFromFile()

```
void GameDataManager::LoadDataFromFile (   
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the game data manager data from passed file

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
------------------------------	---------------------------------

9.27.2.13 SaveDataToFile()

```
void GameManager::SaveDataToFile (  
    Brood::Application::FileAccess * a_fileAccessPtr )
```

saves the game data manager data to passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.27.2.14 SetGameTitle()

```
void GameManager::SetGameTitle (   
    std::string a_gameTitle )
```

setter funciton to set the game title for the game

Parameters

<i>a_gameTitle</i>	game title
--------------------	------------

9.27.3 Member Data Documentation

9.27.3.1 m_board

`Brood::Application::Components::Board` `Brood::Application::Components::GameManager::m_board`
[private]

board for the game

9.27.3.2 m_deckManager

`Brood::Application::Components::DeckManager` `Brood::Application::Components::GameManager::m_deckManager` [private]

Deck mangaer for the game.

9.27.3.3 m_dice

```
Brood::Application::Components::Dice Brood::Application::Components::GameDataManager::m_dice
[private]
```

dice for the game

9.27.3.4 m_displayCard

```
Brood::Application::Components::DisplayCard Brood::Application::Components::GameDataManager<-
::m_displayCard [private]
```

9.27.3.5 m_gameTitle

```
std::string Brood::Application::Components::GameDataManager::m_gameTitle [private]
```

9.27.3.6 m_playerManager

```
Brood::Application::Components::PlayerManager Brood::Application::Components::GameDataManager<-
::m_playerManager [private]
```

player manager for the game

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GameManager.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GameManager.cpp](#)

9.28 GameDataManager Class Reference

A [GameManager](#) Class manages the game data.

```
#include "GameManager.h"
```

9.28.1 Detailed Description

A [GameManager](#) Class manages the game data.

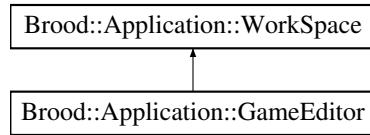
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GameManager.h](#)

9.29 Brood::Application::GameEditor Class Reference

```
#include <GameEditor.h>
```

Inheritance diagram for Brood::Application::GameEditor:



Public Member Functions

- [GameEditor \(Brood::Application::Components::GameManager *aGameData, sf::RectangleShape *a←_panelPtr\)](#)
default constructor
- [~GameEditor \(\)](#)
default destructor
- [void InitializeWorkSpace \(\)](#)
Initializes the work space.
- [void Update \(\)](#)
Updates function.
- [void UpdateAllDisplayElement \(\)](#)
updates the display element
- [void Draw \(sf::RenderWindow &a_window\)](#)
Draw funciton.
- [void Debugger \(\)](#)
debugger funciton

Private Member Functions

- [void UpdateGameTitleInput \(\)](#)
checks if the user interacted with the set Game Title name panel
- [void UpdateMovementTypePanelElement \(\)](#)
checks if the user interacted with the movement type panel
- [void UpdateIncorrectPenaltyPanelElement \(\)](#)
checks if the user interacted with the incorrect penalty panel
- [void UpdateMinimumPlayerNumber \(\)](#)
checks if the user interacted with the minumum player panel
- [void UpdateMaximumPlayerNumber \(\)](#)
checks if the user interacted with the deck panel
- [void UpdateDeckNumber \(\)](#)
checks if the user interacted with the deck number panel

Private Attributes

- sf::RectangleShape * [m_panelBodyPtr](#)
holds the panel body
- Brood::BroodUI::Button * [m_titleScreenBtn](#)
shows the title screen of game
- Brood::BroodUI::TextBox * [m_txtSettingTitle](#)
- Brood::BroodUI::TextBox * [m_txtGameNamePrompt](#)
holds the Game name prompt
- Brood::BroodUI::TextBox * [m_txtGameInput](#)
holds the Game name user enters
- Brood::BroodUI::TextBox * [m_txtMovementTypePromt](#)
holds the text Movement type prompt
- Brood::BroodUI::DropDownInput * [m_ddiMovementType](#)
- Brood::BroodUI::TextBox * [m_txtIncorectPenaltyPromt](#)
holds the text Movement type prompt
- Brood::BroodUI::DropDownInput * [m_ddiIncorectPenalty](#)
holds yes or no
- Brood::BroodUI::TextBox * [m_txtPlayerMinNumPrompt](#)
holds the text minimum player prompt
- Brood::BroodUI::Button * [m_btnPlayerDecMinNum](#)
clicking it will decrease the minium required player number by 1
- Brood::BroodUI::TextBox * [m_txtPlayerMinNum](#)
displays the Current minium required player number
- Brood::BroodUI::Button * [m_btnPlayerIncMinNum](#)
clicking it will increase the minium required player number by 1
- Brood::BroodUI::TextBox * [m_txtPlayerMaxNumPrompt](#)
holds the text maximum player prompt
- Brood::BroodUI::Button * [m_btnPlayerDecMaxnNum](#)
clicking it will decrease the maximum required player number by 1
- Brood::BroodUI::TextBox * [m_txtPlayerMaxNum](#)
displays the Current maximum required player number
- Brood::BroodUI::Button * [m_btnPlayerIncMaxNum](#)
clicking it will increase the maximum required player number by 1
- Brood::BroodUI::TextBox * [m_txtDeckNumPrompt](#)
holds the text number of deck prompt
- Brood::BroodUI::Button * [m_btnDeckDecNum](#)
clicking it will decrease the maximum required player number by 1
- Brood::BroodUI::TextBox * [m_txtDeckNum](#)
displays the Current maximum required player number
- Brood::BroodUI::Button * [m_btnDeckIncNum](#)
clicking it will increase the maximum required player number by 1

Additional Inherited Members

9.29.1 Constructor & Destructor Documentation

9.29.1.1 GameEditor()

```
Brood::Application::GameEditor::GameEditor (
    Brood::Application::Components::GameDataManager * aGameData,
    sf::RectangleShape * aPanelPtr )
```

default constructor

Parameters

<i>aGameData</i>	pointer to the game data object
<i>aPanelPtr</i>	pointer to the panel body object

9.29.1.2 ~GameEditor()

```
Brood::Application::GameEditor::~GameEditor ( )
```

default destructor

9.29.2 Member Function Documentation

9.29.2.1 Debugger()

```
void Brood::Application::GameEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.29.2.2 Draw()

```
void Brood::Application::GameEditor::Draw (
    sf::RenderWindow & aWindow ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Implements [Brood::Application::WorkSpace](#).

9.29.2.3 InitializeWorkSpace()

```
void Brood::Application::GameEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the EditMode tabs and the editor workspaces

Implements [Brood::Application::WorkSpace](#).

9.29.2.4 Update()

```
void Brood::Application::GameEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.29.2.5 UpdateAllDisplayElement()

```
void Brood::Application::GameEditor::UpdateAllDisplayElement ( ) [virtual]
```

updates the display element

@virtual

Implements [Brood::Application::WorkSpace](#).

9.29.2.6 UpdateDeckNumber()

```
void Brood::Application::GameEditor::UpdateDeckNumber ( ) [private]
```

checks if the user interacted with the deck number panel

Deck number panel contains deck number prompt textbox, deck number number value text box, button to increase the deck number number, and button to decrease the deck number

Only the button to increase the deck number, and button to decrease the deck number are interactable

If the interactable button was pressed then the deck number is increased or decreased by 1. deck number should be more than or equal to 1 but less than 10

9.29.2.7 UpdateGameTitleInput()

```
void Brood::Application::GameEditor::UpdateGameTitleInput ( ) [private]
```

checks if the user interacted with the set Game Title name panel

Game title panel contains game title prompt textbox, and textbox to enter the game title.

Only the textbox to enter the game title is interactable

If the interactable textbox was pressed then it allows the user to set the game title

9.29.2.8 UpdateIncorrectPenaltyPanelElement()

```
void Brood::Application::GameEditor::UpdateIncorrectPenaltyPanelElement ( ) [private]
```

checks if the user interacted with the incorrect penalty panel

Incorrect penalty panel contains [Board](#) incorrect penalty prompt textbox, and drop down input to select a yes or no for incorrect penalty

Only the drop down input to select the incorrect penalty is interactable

If the interactable drop down input was pressed then it expand to show the two option avialble Yes and No

9.29.2.9 UpdateMaximumPlayerNumber()

```
void Brood::Application::GameEditor::UpdateMaximumPlayerNumber ( ) [private]
```

checks if the user interacted with the deck panel

Minimum player panel contains deck number prompt textbox, deck number value text box, button to increase the deck number, and button to decrease the maximum player number

Only the button to increase the deck number, and button to decrease the deck number are interactable

If the interactable button was pressed then the deck number is increased or decreased by 1. Maximum player number should be more than or equal to minimum player number but less than 10

9.29.2.10 UpdateMinimumPlayerNumber()

```
void Brood::Application::GameEditor::UpdateMinimumPlayerNumber ( ) [private]
```

checks if the user interacted with the minumum player panel

Minumum player panel contains minimum player number prompt textbox, minimum player number value text box, button to increase the minimum player number, and button to decrease the minimum player number

Only the button to increase the minimum player number, and button to decrease the minimum player number are interactable

If the interactable button was pressed then the minimum player number is increased or decreased by 1. Minimum player number should be more than or equal to 0 but less than Maximum player number

9.29.2.11 UpdateMovementTypePanelElement()

```
void Brood::Application::GameEditor::UpdateMovementTypePanelElement ( ) [private]
```

checks if the user interacted with the movement type panel

Movement type panel contains [Board](#) movement type promt textbox, and drop down input to select a movement type

Only the drop down input to select the movement type is interactable

If the interactable drop down input was pressed then it expand to show the three option avialble dice, start with dice and then card, and card no dice

9.29.3 Member Data Documentation

9.29.3.1 m_btnDeckDecNum

```
Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnDeckDecNum [private]
```

clicking it will decrease the maximum required player number by 1

9.29.3.2 m_btnDeckIncNum

```
Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnDeckIncNum [private]
```

clicking it will increase the maximum required player number by 1

9.29.3.3 m_btnPlayerDecMaxnNum

```
Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnPlayerDecMaxnNum [private]
```

clicking it will decrease the maximum required player number by 1

9.29.3.4 m_btnPlayerDecMinNum

```
Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnPlayerDecMinNum [private]
```

clicking it will decrease the minium required player number by 1

9.29.3.5 m_btnPlayerIncMaxNum

`Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnPlayerIncMaxNum [private]`

clicking it will increase the maximum required player number by 1

9.29.3.6 m_btnPlayerIncMinNum

`Brood::BroodUI::Button* Brood::Application::GameEditor::m_btnPlayerIncMinNum [private]`

clicking it will increase the minium required player number by 1

9.29.3.7 m_ddiIncorrectPenalty

`Brood::BroodUI::DropDownInput* Brood::Application::GameEditor::m_ddiIncorrectPenalty [private]`

holds yes or no

9.29.3.8 m_ddiMovementType

`Brood::BroodUI::DropDownInput* Brood::Application::GameEditor::m_ddiMovementType [private]`

holds how the player can move: dice, start with dice and then card, and card no dice

9.29.3.9 m_panelBodyPtr

`sf::RectangleShape* Brood::Application::GameEditor::m_panelBodyPtr [private]`

holds the panel body

9.29.3.10 m_titleScreenBtn

`Brood::BroodUI::Button* Brood::Application::GameEditor::m_titleScreenBtn [private]`

shows the tile screen of game

9.29.3.11 m_txtDeckNum

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtDeckNum [private]
```

displays the Current maximum required player number

9.29.3.12 m_txtDeckNumPrompt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtDeckNumPrompt [private]
```

holds the text number of deck prompt

9.29.3.13 m_txtGameInput

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtGameInput [private]
```

holds the Game name user enters

9.29.3.14 m_txtGameNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtGameNamePrompt [private]
```

holds the Game name prompt

9.29.3.15 m_txtIncorrectPenaltyPromt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtIncorrectPenaltyPromt [private]
```

holds the text Movement type prompt

9.29.3.16 m_txtMovementTypePromt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtMovementTypePromt [private]
```

holds the text Movement type prompt

9.29.3.17 m_txtPlayerMaxNum

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtPlayerMaxNum [private]
```

displays the Current maximum required player number

9.29.3.18 m_txtPlayerMaxNumPrompt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtPlayerMaxNumPrompt [private]
```

holds the text maximum player prompt

9.29.3.19 m_txtPlayerMinNum

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtPlayerMinNum [private]
```

displays the Current minium required player number

9.29.3.20 m_txtPlayerMinNumPrompt

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtPlayerMinNumPrompt [private]
```

holds the text minimum player prompt

9.29.3.21 m_txtSettingTitle

```
Brood::BroodUI::TextBox* Brood::Application::GameEditor::m_txtSettingTitle [private]
```

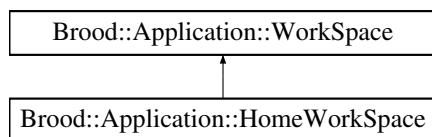
The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GameEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GameEditor.cpp](#)

9.30 Brood::Application::HomeWorkSpace Class Reference

```
#include <HomeWorkSpace.h>
```

Inheritance diagram for Brood::Application::HomeWorkSpace:



Public Member Functions

- `HomeWorkSpace ()`
@TODO delete me
- `~HomeWorkSpace ()`
default destructor
- virtual void `InitializeWorkSpace ()` override
Initializes the work space.
- virtual void `Update ()`
Updates function.
- void `UpdateAllDisplayElement ()`
Updating all the component.
- virtual void `Draw (sf::RenderWindow &a_window)`
Draw funciton.
- virtual void `Debugger ()`
debugger funciton

Private Member Functions

- void `InitializeRibbonTabs ()`
Initializes the ribbon tab bar which is at the top of the applicaiton.
- void `UpdateRibbonTabs ()`
Checks to see if the ribbon tabs's logic is to be executed or not.
- void `ExecuteMenuItem (unsigned a_idx, unsigned a_jidx)`
executes the funciton related to the menu item of the ribbon tabs
- void `UpdateHomeWorkspace ()`
Holds the logic for the elements in the HomeWorkspace.
- void `UpdateFileNameFeildDialog ()`
Holds the logic to execute open file button was pressed.
- void `SwitchToHomeWorkspace ()`
Swtichs to home workspace if it is not in it.
- void `ExecuteLoadPreviousEditDialogBox ()`
opening a dialog box so that user can enter file name to load previous edit
- void `ExecuteCreateNewEditDialogBox ()`
opening a dialog box so that user can enter file name to create new edit
- void `ResetGameData ()`
resets the game manager
- void `SaveGameData ()`
Seializes the mGameData with to a file with game title.
- bool `LoadGameData (std::filesystem::path a_gameOrEditorFile)`
Loads mGameData with the specified file.

Private Attributes

- unsigned `m_openFileBtnIdx`
- `Brood::BroodUI::MenuBar m_ribbonTabs`
Ui for ribbon tabs.
- `sf::RectangleShape m_gameOrEditorForeground`
- `Brood::BroodUI::Button * m_gameOrEditorBtn`
- `Brood::BroodUI::Button * m_toggleGameInfoLeftBtn`
button to toggle left through the games/edits list
- `Brood::BroodUI::Button * m_toggleGameInfoRightBtn`
button to toggle right through the games/edits list
- int `m_currGameInfoIdx`
index of the file that is loaded by the main button
- bool `m_showImportDialogBox`
true to show the import dialog box
- `Brood::BroodUI::Button * m_importGameBtn`
button to import games
- `Brood::BroodUI::Button * m_loadPreviousEditsBtn`
button to load previous edits
- `Brood::BroodUI::Button * m_createNewEditBtn`
button to create new edit
- `Brood::BroodUI::Button * m_cancleBtn`
button to close the dialog box
- bool `m_showEnterFilenameDialogBox`
true to show the filename dialog box
- `Brood::BroodUI::TextBox * m_fileNameTxt`
stores the filename prompt
- `Brood::BroodUI::TextBox * m_enterFileNameFeildTxt`
place to enter the filename
- `Brood::BroodUI::Button * m_openFileBtn`
button to open the entered file name
- `std::vector< std::string > m_gameInfoFilePathList`
store the list of file path that corresponds to the game or editor file
- `std::vector< sf::Texture > m_gameListTexture`
list of texture to show in the gamOrEditor button
- `std::vector< Brood::Application::WorkSpace * > m_workSpacesList`
- unsigned `m_currWorkSpaceIdx`
- bool `m_isHomeWorkspace`

Additional Inherited Members

9.30.1 Constructor & Destructor Documentation

9.30.1.1 HomeWorkSpace()

`HomeWorkSpace::HomeWorkSpace ()`

@TODO delete me

default constructor @TODO load the game info file path @TODO add the create games and exit to the file path
@TODO load the game texture

9.30.1.2 ~HomeWorkSpace()

```
HomeWorkSpace::~HomeWorkSpace ( )
default destructor
@virtual
```

9.30.2 Member Function Documentation

9.30.2.1 Debugger()

```
void HomeWorkSpace::Debugger ( ) [virtual]
debugger funciton
@virtual
This function helps in debugging the UI elements.
Implements Brood::Application::WorkSpace.
```

9.30.2.2 Draw()

```
void HomeWorkSpace::Draw (
    sf::RenderWindow & a_window ) [virtual]
Draw funciton.
@virtual
Draws all the component to the screen @TODO deleteme
Implements Brood::Application::WorkSpace.
```

9.30.2.3 ExecuteCreateNewEditDialogBox()

```
void HomeWorkSpace::ExecuteCreateNewEditDialogBox ( ) [private]
opening a dialog box so that user can enter file name to create new edit
```

9.30.2.4 ExecuteLoadPreviousEditDialogBox()

```
void HomeWorkSpace::ExecuteLoadPreviousEditDialogBox ( ) [private]
opening a dialog box so that user can enter file name to load previous edit
```

9.30.2.5 ExecuteMenuItem()

```
void HomeWorkSpace::ExecuteMenuItem (
    unsigned a_iIdx,
    unsigned a_jIdx ) [private]
executes the funciton related to the menu item of the ribbon tabs
It contains all the funciton to execute for each menu item of the ribbon tabs
```

Parameters

<i>a_idx</i>	index of menu to execute
<i>a_jidx</i>	index of menu item to execute for given menu index i

9.30.2.6 InitializeRibbonTabs()

```
void HomeWorkSpace::InitializeRibbonTabs ( ) [private]
```

Initializes the ribbon tab bar which is at the top of the applicaiton.

It has the file and help menus in it.

File menu has Create new, Import games, Load previous edits, and Save option under it.

Help menu has How to Use, Documentation, and About BroodMaker option under it.

9.30.2.7 InitializeWorkSpace()

```
void HomeWorkSpace::InitializeWorkSpace ( ) [override], [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It inializes the ribbon tab, creates the left and right toggle button, game/edit secllection card, importGames, loadPreviousEdits, CreateMew edit, cancle button, and create new button. It also creates the fileName, and enter FineName text feild

Implements [Brood::Application::WorkSpace](#).

9.30.2.8 LoadGameData()

```
bool HomeWorkSpace::LoadGameData (
    std::filesystem::path a_gameOrEditorFile ) [private]
```

Loads mGameData with the specified file.

Returns

flase if the passed file does not exist

9.30.2.9 ResetGameData()

```
void HomeWorkSpace::ResetGameData ( ) [private]
```

resets the game manager

9.30.2.10 SaveGameData()

```
void HomeWorkSpace::SaveGameData ( ) [private]
```

Serializes the mGameData with to a file with game title.

9.30.2.11 SwitchToHomeWorkspace()

```
void HomeWorkSpace::SwitchToHomeWorkspace ( ) [private]
```

Switches to home workspace if it is not in it.

9.30.2.12 Update()

```
void HomeWorkSpace::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.30.2.13 UpdateAllDisplayElement()

```
void HomeWorkSpace::UpdateAllDisplayElement ( ) [virtual]
```

Updating all the component.

@virtual

Updating all the component to the screen

Implements [Brood::Application::WorkSpace](#).

9.30.2.14 UpdateFileNameFeildDialog()

```
void HomeWorkSpace::UpdateFileNameFeildDialog ( ) [private]
```

Holds the logic to execute open file button was pressed.

9.30.2.15 UpdateHomeWorkspace()

```
void HomeWorkSpace::UpdateHomeWorkspace ( ) [private]
```

Holds the logic for the elements in the HomeWorkspace.

9.30.2.16 UpdateRibbonTabs()

```
void HomeWorkSpace::UpdateRibbonTabs ( ) [private]
```

Checks to see if the ribbon tabs's logic is to be executed or not.

Todo deleteME

Todo deleteme

list of function to execute for each menu item

9.30.3 Member Data Documentation**9.30.3.1 m_cancleBtn**

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_cancleBtn [private]
```

button to close the dialog box

9.30.3.2 m_createNewEditBtn

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_createNewEditBtn [private]
```

button to create new edit

9.30.3.3 m_currGameInfoIdx

```
int Brood::Application::HomeWorkSpace::m_currGameInfoIdx [private]
```

index of the file that is loaded by the main button

9.30.3.4 m_currWorkSpaceIdx

```
unsigned Brood::Application::HomeWorkSpace::m_currWorkSpaceIdx [private]
```

holds which workspace the application is currently on, 0 means editor workspace, 1 means game workspace

9.30.3.5 m_enterFileNameFeildTxt

```
Brood::BroodUI::TextBox* Brood::Application::HomeWorkSpace::m_enterFileNameFeildTxt [private]
```

place to enter the filename

9.30.3.6 m_fileNameTxt

```
Brood::BroodUI::TextBox* Brood::Application::HomeWorkSpace::m_fileNameTxt [private]
```

stores the filename prompt

9.30.3.7 m_gameInfoFilePathList

```
std::vector<std::string> Brood::Application::HomeWorkSpace::m_gameInfoFilePathList [private]
```

store the list of file path that corresponds to the game or editor file

9.30.3.8 m_gameListTexture

```
std::vector<sf::Texture> Brood::Application::HomeWorkSpace::m_gameListTexture [private]
```

list of texture to show in the gamOrEditor button

9.30.3.9 m_gameOrEditorBtn

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_gameOrEditorBtn [private]
```

main card that is in the middle of the screen that shows the current game/edit list

9.30.3.10 m_gameOrEditorForeground

```
sf::RectangleShape Brood::Application::HomeWorkSpace::m_gameOrEditorForeground [private]
```

main ui element which shows the list of stored game files and editor files

9.30.3.11 m_importGameBtn

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_importGameBtn [private]
```

button to import games

9.30.3.12 m_isHomeWorkspace

```
bool Brood::Application::HomeWorkSpace::m_isHomeWorkspace [private]
```

holds wheter current active workspace is initial workspace or game/edit workspace

9.30.3.13 m_loadPreviousEditsBtn

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_loadPreviousEditsBtn [private]
```

button to load previous edits

9.30.3.14 m_openFileBtn

```
Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_openFileBtn [private]
```

button to open the entered file name

9.30.3.15 m_openFileBtnIdx

```
unsigned Brood::Application::HomeWorkSpace::m_openFileBtnIdx [private]
```

9.30.3.16 m_ribbonTabs

Brood::BroodUI::MenuBar Brood::Application::HomeWorkSpace::m_ribbonTabs [private]

Ui for ribbon tabs.

9.30.3.17 m_showEnterFilenameDialogBox

bool Brood::Application::HomeWorkSpace::m_showEnterFilenameDialogBox [private]

true to show the filename dialog box

9.30.3.18 m_showImportDialogBox

bool Brood::Application::HomeWorkSpace::m_showImportDialogBox [private]

true to show the import dialog box

9.30.3.19 m_toggleGameInfoLeftBtn

Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_toggleGameInfoLeftBtn [private]

button to toggle left through the games/edits list

9.30.3.20 m_toggleGameInfoRightBtn

Brood::BroodUI::Button* Brood::Application::HomeWorkSpace::m_toggleGameInfoRightBtn [private]

button to toggle right through the games/edits list

9.30.3.21 m_workSpacesList

std::vector<Brood::Application::WorkSpace*> Brood::Application::HomeWorkSpace::m_workSpaces
List [private]

list of workspace that the application uses. It holds 2 workspace, namely, editorWorSpace, and game workspace.

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[HomeWorkSpace.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[HomeWorkSpace.cpp](#)

9.31 HomeWorkSpace Class Reference

It is the underlying entry to the application. It is the true entry point for the application.

```
#include "HomeWorkSpace.h"
```

9.31.1 Detailed Description

It is the underlying entry to the application. It is the true entry point for the application.

It creates ribbon tabs which is accessable from anywhere in the applicaiton.

It comprises of 3 workspace, namely Initial [WorkSpace](#), Editor workspace, and game workspace.

Initial [WorkSpace](#) is the homeworkspace. that is to say that the it controls wheter to switch to Editor workspace, or game workspace, ot to itself. It is where the user can chose weather to load a game file and go into game mode to play previously made games or alternatively to go into editor mode and work on previous game edits or start a new game project.

Editor workspace is where user can make new games and export the created game to play in the game workspace.

Game workspace is where the user can play the created game.

Derived from the [WorkSpace](#) class

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[HomeWorkSpace.h](#)

9.32 Brood::BroodUI::Id Class Reference

```
#include <UI_ID.h>
```

Public Member Functions

- `Id (Id *a_parentIDPtr=nullptr)`
default constructor
- `~Id ()`
- `Id (const Id &a_otherId)`
Copy constructor.
- `Id & operator= (const Id &a_otherId)`
assignment operator
- `const int GetParentID () const`
Getter function to get the parent's ID.
- `const Brood::BroodUI::Id * GetParentIDPtr () const`
Getter function to get the pointer to its parent.
- `Brood::BroodUI::Id * GetParentIDPtr ()`
Getter function to get the pointer to its parent.
- `const int GetElementID () const`

- const int `GetChildIdx` (const `Brood::BroodUI::Id` *`a_childIDPtr`) const

Getter function to get the element's ID.
- const int `GetTotalChildNum` () const

Getter function to get how many child the element has.
- `Id * GetChildIdAtIndex` (const int `a_index`) const

Getter function to get pointer of the passed child's index.
- bool `HasChild` () const

Getter function to check if the element has a child or not.
- bool `HasParent` () const

Getter function to check if the element has a parent or not.
- void `SetParent` (`Id *a_parentID=nullptr`)

Setter function to set the element's parent.
- void `AddChild` (`Brood::BroodUI::Id *a_childIDPtr`)

adds the child to its child list
- void `DeleteChildAtIndex` (const int `a_index`)

deletes the child at passed `a_index` from the child list

Private Attributes

- `Brood::BroodUI::Id * m_parentIDPtr`

stores the parents UI ID pointer
- int `m_parentID`

stores the parents UI ID
- int `m_elementID`

stores its UI ID
- bool `m_hasChildren`

stores if it has children or not
- `std::vector< Brood::BroodUI::Id * > m_allChildPtrs`

stores all the pointers to its children

Static Private Attributes

- static unsigned `GLOBAL_ID_NUM` = 0

unique id is assigned based on this

9.32.1 Constructor & Destructor Documentation

9.32.1.1 `Id()` [1/2]

```
Id::Id (
    Id * a_parentIDPtr = nullptr )
```

default constructor

Initializes the data member of the class.

Note

If no parent is supplied, it initializes `m_parentId` to -1.

Parameters

<code>a_parentID</code>	pointer to the <code>Id</code> of the parent if any; default -> nullPtr
-------------------------	---

9.32.1.2 ~Id()

```
Id::~Id ( )
```

@breif default destructor

9.32.1.3 Id() [2/2]

```
Id::Id (
    const Id & a_otherId )
```

Copy constructor.

Parameters

<code>a_<-> otherId</code>	reference to the <code>Id</code> from which <code>Id</code> is to be copied from
----------------------------------	--

9.32.2 Member Function Documentation**9.32.2.1 AddChild()**

```
void Id::AddChild (
    Brood::BroodUI::Id * a_childIdPtr )
```

adds the child to its child list

Parameters

<code>a_childIdPtr</code>	pointer to the child id
---------------------------	-------------------------

9.32.2.2 DeleteChildIdAtIndex()

```
void Id::DeleteChildIdAtIndex (
    const int a_index )
```

deletes the child at passed a_index from the child list

Parameters

<i>a_index</i>	index child's pointer in the child list to be deleted
----------------	---

9.32.2.3 GetChildIdAtIndex()

```
Brood::BroodUI::Id * Id::GetChildIdAtIndex (
    const int a_index ) const
```

Getter function to get pointer of the passed child's index.

Note

if there index supplied is out of bound then, it logs the error and returns nullptr

Parameters

<i>a_index</i>	index of the child in the elements's child list
----------------	---

Returns

pointer to child at the given index if present; else returns nullptr

9.32.2.4 GetChildIndex()

```
const int Id::GetChildIndex (
    const Brood::BroodUI::Id * a_childIDPtr ) const
```

Getter function to get its child element's index in its list.

Parameters

<i>a_childIDPtr</i>	pointer to the child whose index we are trying to find
---------------------	--

Returns

child element's index in its child list; if no such child exists return -1

9.32.2.5 GetElementID()

```
const int Id::GetElementID ( ) const
```

Getter function to get the element's ID.

Returns

element's unique UI ID

9.32.2.6 GetParentID()

```
const int Id::GetParentID ( ) const
```

Getter function to get the parent's ID.

Returns

parent's unique UI ID if presents, else returns -1

9.32.2.7 GetParentIDPtr() [1/2]

```
Brood::BroodUI::Id * Id::GetParentIDPtr ( )
```

Getter function to get the pointer to its parent.

Returns**9.32.2.8 GetParentIDPtr() [2/2]**

```
const Brood::BroodUI::Id * Id::GetParentIDPtr ( ) const
```

Getter function to get the pointer to its parent.

Returns

9.32.2.9 GetTotalChildNum()

```
const int Id::GetTotalChildNum ( ) const
```

Getter function to get how many child the element has.

Returns

element's child list size

9.32.2.10 HasChild()

```
bool Id::HasChild ( ) const
```

Getter function to check if the element has a child or not.

Returns

true if the element has child/s; else returns false

9.32.2.11 HasParent()

```
bool Id::HasParent ( ) const
```

Getter function to check if the element has a parent or not.

Returns

true if the element has parent; else returns false

9.32.2.12 operator=(*)

```
Brood::BroodUI::Id & Id::operator= (
    const Id & a_otherId )
```

assignment operator

Parameters

<i>a_<- otherId</i>	reference to the <code>Id</code> from which <code>Id</code> is to be copied from
------------------------	--

Returns

pointer to itself

9.32.2.13 SetParent()

```
void Id::SetParent (
    Id * a_parentID = nullptr )
```

Setter function to set the element's parent.

Parameters

<i>a_parentID</i>	pointer to the parent element id object
-------------------	---

9.32.3 Member Data Documentation**9.32.3.1 GLOBAL_ID_NUM**

```
unsigned Id::GLOBAL_ID_NUM = 0 [static], [private]
```

unique id is assigned based on this

assigning a location for global id.

It starts with 0 and only goes up.

9.32.3.2 m_allChildPtrs

```
std::vector<Brood::BroodUI::Id*> Brood::BroodUI::Id::m_allChildPtrs [private]
```

stores all the pointer to its childs

9.32.3.3 m_elementID

```
int Brood::BroodUI::Id::m_elementID [private]
```

stores its UI ID

9.32.3.4 m_hasChilds

```
bool Brood::BroodUI::Id::m_hasChilds [private]
```

stores if it has childs or not

9.32.3.5 m_parentID

```
int Brood::BroodUI::Id::m_parentID [private]
```

stores the parents UI ID

9.32.3.6 m_parentIDPtr

```
Brood::BroodUI::Id* Brood::BroodUI::Id::m_parentIDPtr [private]
```

stores the parents UI ID pointer

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UI_ID.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UI_ID.cpp](#)

9.33 Id Class Reference

Class to provide a unique ID to the different UI elements.

9.33.1 Detailed Description

Class to provide a unique ID to the different UI elements.

All the UI elements has a unique ID. The [Id](#) class also keeps track of its parent and childs. This ID system is the core of the intermediate GUI method. This ID helps us to keep track of which element is selected, or about to be selected and what action we should take

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UI_ID.h](#)

9.34 Brood::BroodUI::MapIdToElement Class Reference

```
#include <MapIdToElement.h>
```

Static Public Member Functions

- static std::map< const int, Brood::BroodUI::UIElement * > & **GetMap** ()

Getter Function to get a reference to the map.
- static Brood::BroodUI::UIElement * **GetElementPtrFromMap** (int a_id)

Getter Function to get the element that is mapped to from ui id.
- static bool **AddToMap** (int a_id, Brood::BroodUI::UIElement *a_elementPtr)

adds the id and element to the map
- static bool **RemoveFromMap** (int a_id)

removes a entry with the id as a key from the map

Static Private Attributes

- static std::map< const int, Brood::BroodUI::UIElement * > **stm_mapper**

maps the unique UI ID to the element

9.34.1 Member Function Documentation

9.34.1.1 AddToMap()

```
bool MapIdToElement::AddToMap (
    int a_id,
    Brood::BroodUI::UIElement * a_elementPtr ) [static]
```

adds the id and element to the map

Parameters

<i>a_id</i>	unique id of the element that is mapped to its element pointer
<i>a_elementPtr</i>	pointer to the element that has the passed unique UI_ID

Returns

true if adding was successfull; else false

9.34.1.2 GetElementPtrFromMap()

```
Brood::BroodUI::UIElement * MapIdToElement::GetElementPtrFromMap (
    int a_id ) [static]
```

Getter Function to get the element that is mapped to from ui id.

Parameters

<i>a_id</i>	unique id of the element that maps to its element pointer
-------------	---

Returns

pointer to the element whose unique id was mapped to; else returns nullptr if not matched

9.34.1.3 GetMap()

```
std::map< const int, Brood::BroodUI::UIElement * > & MapIdToElement::GetMap ( ) [static]
```

Getter Funciton to get a reference to the map.

Returns

reference to the map

9.34.1.4 RemoveFromMap()

```
bool MapIdToElement::RemoveFromMap (
    int a_id ) [static]
```

removes a entry with the id as a key from the map

Parameters

<i>a_id</i>	unique id of the element that is mapped to its element pointer
-------------	--

Returns

true if removal was successfull; else false

9.34.2 Member Data Documentation**9.34.2.1 stm_mapper**

```
std::map< const int, Brood::BroodUI::UIElement * > MapIdToElement::stm_mapper [static], [private]
```

maps the unique UI ID to the element

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MapIdToElement.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MapIdToElement.cpp](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UIElement.cpp](#)

9.35 MapIdToElement Class Reference

A static struct to handle the mapping of unique UI_ID to its element for quick access.

```
#include <MapIdToElement.h>
```

9.35.1 Detailed Description

A static struct to handle the mapping of unique UI_ID to its element for quick access.

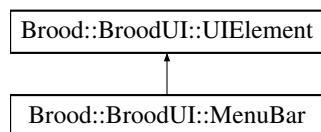
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MapIdToElement.h](#)

9.36 Brood::BroodUI::MenuBar Class Reference

```
#include <MenuBar.h>
```

Inheritance diagram for Brood::BroodUI::MenuBar:



Public Member Functions

- **MenuBar (Brood::BroodUI::UIElement *a_parentPtr=nullptr)**
Default Constructor.
- **virtual ~MenuBar ()**
Default Destructor.
- **MenuBar (const MenuBar &a_otherElement)**
Copy constructor.
- **Brood::BroodUI::MenuBar & operator= (const MenuBar &a_otherElement)**
assignment operator
- **std::vector< Brood::BroodUI::DropDownMenu * > & GetMenuList ()**
Getter funciton to get the menu List.
- **void SetBodySize (sf::Vector2f a_eachItemSize) override**
Setter function to set the `MenuBar`'s size.
- **void SetBodySize (float a_itemSizeX, float a_itemSizeY) override**
Setter function to set the `MenuBar`'s size.
- **void SetBodyPosition (sf::Vector2f a_pos, bool a_relativeToParent=false) override**
Setter function to set the menubar's Position.
- **void SetBodyPosition (float aPosX, float aPosY, bool a_relativeToParent=false) override**
Setter function to set the menubar's Position.
- **void SetFont (sf::Font *a_font)**
Setter function to set the font for the menu.
- **virtual void SetFontColor (sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_White)**
setter function to set the Font color
- **void SetFontSize (unsigned a_fontSize)**
Setter funtion to set the size of the font.
- **virtual void SetBodyColor (sf::Color a_bodyColor)**
Setter function to set the element body color.
- **virtual void SetActiveOverlayColor (sf::Color a_color)**
Setter function to set the element's active color.
- **virtual void SetHotOverlayColor (sf::Color a_color)**
Setter function to set the element's hot color.
- **void AddMenuToMenuBar (std::string a_menuName)**
Function to add menu to the menubar.
- **void AddMenuToMenuBar (Brood::BroodUI::DropDownMenu *a_dropdownMenuPtrToAdd, bool a_create←New=true)**
Function to adds the passed `DropDownMenu` at back of menubar.
- **void AddItemToMenu (unsigned a_index, std::string a_menuName)**
Function to add menu item to menu in menubar.
- **void AddItemToMenu (unsigned a_index, Brood::BroodUI::Button *a_buttonPtrToAdd, bool a_create←New=true)**
Function to adds the passed button to dropdown menu at passed index.
- **virtual void Draw (sf::RenderWindow &a_window) override**
Draw function.
- **virtual void Debugger ()**
changes the text of the element to its the element ID when debug is on and changes it back to the text it had before debugging when debug is off.

Private Member Functions

- void [SetMenuBodySize](#) (int a_itemIndex)
helper function to set the body size of the menus correctly in the menubar
- void [SetMenuPos](#) (int a_itemIndex)
helper function to position the menus correctly in the menubar

Private Attributes

- std::vector< [Brood::BroodUI::DropDownMenu](#) * > m_menus
stores the list of drop down menus

Additional Inherited Members

9.36.1 Constructor & Destructor Documentation

9.36.1.1 [MenuBar\(\)](#) [1/2]

```
MenuBar::MenuBar (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr )
```

Default Constructor.

Initializes the [MenuBar](#) object

Parameters

<code>m_parentPtr</code>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------------	---

9.36.1.2 [~MenuBar\(\)](#)

```
MenuBar::~MenuBar ( ) [virtual]
```

Default Destructor.

9.36.1.3 [MenuBar\(\)](#) [2/2]

```
MenuBar::MenuBar (
    const MenuBar & a_otherElement )
```

Copy constructor.

Parameters

<i>a_otherElement</i>	reference to the MenuBar which is used to copy the data form
-----------------------	--

9.36.2 Member Function Documentation**9.36.2.1 AddItemToMenu() [1/2]**

```
void MenuBar::AddItemToMenu (
    unsigned a_index,
    Brood::BroodUI::Button * a_buttonPtrToAdd,
    bool a_createNew = true )
```

Function to adds the passed button to dropdown menu at passed index.

if *a_createNew* is true then create a new button using the passed item then adds the new button to the [DropDownMenu](#)

It dynamically allocates memory for the newly created item

Parameters

<i>a_index</i>	menu bar which dropdown the item is to be added to
<i>a_buttonPtrToAdd</i>	pointer to the button element which is used to add
<i>a_createNew</i>	if <i>a_createNew</i> is true then create a new button using the passed item then adds the new button

9.36.2.2 AddItemToMenu() [2/2]

```
void MenuBar::AddItemToMenu (
    unsigned a_index,
    std::string a_menuItemName )
```

Function to add menu item to menu in menubar.

Parameters

<i>a_index</i>	menu bar which dropdown the item is to be added to
<i>a_menuItemName</i>	name of the item

9.36.2.3 AddMenuToMenuBar() [1/2]

```
void MenuBar::AddMenuToMenuBar (
    Brood::BroodUI::DropDownMenu * a_dropdownMenuPtrToAdd,
    bool a_createNew = true )
```

Function to adds the passed [DropDownMenu](#) at back of menubar.

If `a_createNew` is true then create a new [DropDownMenu](#) using the passed item then adds the new [DropDownMenu](#) to the menubar It dynamically allocates memory for the newly created item

Parameters

<code>a_dropdownMenuPtrToAdd</code>	pointer to the dropdownMenu element which is used to add at the back
<code>a_createNew</code>	if <code>a_createNew</code> is true then create a new dropdownMenu using the passed item then adds the new button

9.36.2.4 AddMenuToMenuBar() [2/2]

```
void MenuBar::AddMenuToMenuBar (
    std::string a_menuName )
```

Function to add menu to the menubar.

It dynamically allocates memory for the menu and adds it to the list.

Note

assumes that menu bar is already set. For more information

See also

[Brood::BroodUI::MenuBar::SetMenuBodySize\(\)](#)

Note

assumes font is set. For more information

See also

[Brood::BroodUI::UIElement::SetFont\(\)](#)

Note

assumes font size is set. For more information

See also

[Brood::BroodUI::UIElement::SetCharacterSize\(\)](#)

Parameters

<code>a_menuName</code>	name of the item
-------------------------	------------------

9.36.2.5 Debugger()

```
void MenuBar::Debugger ( ) [virtual]
```

changes the text of the element to its the element ID when debug is on and changes it back to the text it had before debugging when debug is off.

Implements [Brood::BroodUI::UIElement](#).

9.36.2.6 Draw()

```
void MenuBar::Draw (
    sf::RenderWindow & a_window ) [override], [virtual]
```

Draw function.

Parameters

<code>a_window</code>	reference to render window
-----------------------	----------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.7 GetMenuList()

```
std::vector< Brood::BroodUI::DropDownMenu * > & MenuBar::GetMenuList ( )
```

Getter funciton to get the menu List.

Returns

reference to the menu list

9.36.2.8 operator=()

```
Brood::BroodUI::MenuBar & MenuBar::operator= (
    const MenuBar & a_otherElement )
```

assignment operator

Parameters

<i>a_otherElement</i>	reference to the dropdown menu which is used to copy the data form
-----------------------	--

Returns

pointer to this element

9.36.2.9 SetActiveOverlayColor()

```
void MenuBar::SetActiveOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's active color.

Parameters

<i>a_color</i>	color of the body when it is active
----------------	-------------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.10 SetBodyColor()

```
void MenuBar::SetBodyColor (
    sf::Color a_bodyColor ) [virtual]
```

Setter function to set the element body color.

Parameters

<i>a_bodyColor</i>	color of the body
--------------------	-------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.11 SetBodyPosition() [1/2]

```
void MenuBar::SetBodyPosition (
    float a_posX,
    float a_posY,
    bool a_relativeToParent = false ) [override], [virtual]
```

Setter function to set the menubar's Position.

@virtual

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_posX</i>	x-position of the element
<i>a_posY</i>	y-position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.12 SetBodyPosition() [2/2]

```
void MenuBar::SetBodyPosition (
    sf::Vector2f a_pos,
    bool a_relativeToParent = false ) [override], [virtual]
```

Setter function to set the menubar's Position.

@virtual

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_pos</i>	position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.13 SetBodySize() [1/2]

```
void MenuBar::SetBodySize (
    float a_sizeX,
    float a_sizeY ) [override], [virtual]
```

Setter function to set the [MenuBar](#)'s size.

@virtual

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Note

this is the menu bar size and not the menus size. Menu size is determined by its content.

Parameters

<i>a_itemSizeX</i>	length of the menubar
<i>a_itemSizeY</i>	height of the menubar

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.14 SetBodySize() [2/2]

```
void MenuBar::SetBodySize (
    sf::Vector2f a_size )  [override], [virtual]
```

Setter function to set the [MenuBar](#)'s size.

@virtual

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Note

this is the menu bar size and not the menu itself

Warning

The menu height cannot be smaller than font size + 2 px.

Parameters

<i>a_size</i>	length and height the menubar -> sf::vector2f
---------------	---

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.15 SetFont()

```
void MenuBar::SetFont (
    sf::Font * a_font )
```

Setter function to set the font for the menu.

Parameters

<i>a_font</i>	pointer to a loaded font
---------------	--------------------------

9.36.2.16 SetFontColor()

```
void MenuBar::SetFontColor (
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_White
) [virtual]
```

setter function to set the Font color

param a_color font color -> default sf::Color::White

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.17 SetFontSize()

```
void MenuBar::SetFontSize (
    unsigned a_fontSize )
```

Setter funtion to set the size of the font.

This does not affect the menubar height at all.

Warning

Before setting the font size set the menubar height

Font size cannot be bigger than menu height - 2. If a a_fontSize is bigger than the menu height by menu height - 2. then sets the font size to menu height - 2

Parameters

<i>a_fontSize</i>	size of the font in pixel
-------------------	---------------------------

9.36.2.18 SetHotOverlayColor()

```
void MenuBar::SetHotOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's hot color.

Parameters

<i>a_color</i>	color of the body when it is hot
----------------	----------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.36.2.19 SetMenuBodySize()

```
void MenuBar::SetMenuBodySize (
    int a_itemIndex ) [private]
```

helper funciton to set the body size of the menus correctly in the menubar

Parameters

<i>a_itemIndex</i>	index of the menu in the menu list
--------------------	------------------------------------

9.36.2.20 SetMenuPos()

```
void MenuBar::SetMenuPos (
    int a_itemIndex ) [private]
```

helper funciton to position the menus correctly in the menubar

Parameters

<i>a_itemIndex</i>	index of the menu in the menu list
--------------------	------------------------------------

9.36.3 Member Data Documentation

9.36.3.1 m_menus

```
std::vector<Brood::BroodUI::DropDownMenu*> Brood::BroodUI::MenuBar::m_menus [private]
```

stores the list of drop down menus

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MenuBar.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MenuBar.cpp](#)

9.37 MenuBar Class Reference

A class that represents a menu bar.

```
#include "MenuBar.h"
```

9.37.1 Detailed Description

A class that represents a menu bar.

It is a class derived from [UIElement](#) class. It acts a a container that bundles DropDownmeus horizontoally.

9.37.1.0.1 Example Case

```
// window object
sf::RenderWindow window( sf::VideoMode( 500, 500 ), "BroodMaker" );
// event object
sf::Event events;
// creating a MenuBar object
Brood::BroodUI::MenuBar myMenu;
// setting size and position
myMenu.SetBodySize( WINDOW_WIDTH, 30 );
myMenu.SetBodyPosition( 0, 0 );
// setting the font which is required to display the text
myMenuSetFont( &font );
// setting the font size. It should be less than the body height by 2 units.
// @see Brood::BroodUI::UIElement::SetFontSize() for more information
myMenu.SetCharacterSize( 20 );
// setting body color
myMenu.SetBodyColor( sf::Color::Magenta );
// Adding 3 menu to the menubar
// and setting the menu title to "hello w", "b", and "c"
myMenu.AddMenuToMenuBar( "hello w" );
myMenu.AddMenuToMenuBar( "b" );
myMenu.AddMenuToMenuBar( "c" );
// adding 4 items to the first menu of the menu bar
myMenu.GetMenuList().at( 0 )->AddItemToMenu( "aa" );
myMenu.GetMenuList().at( 0 )->AddItemToMenu( "ab" );
myMenu.GetMenuList().at( 0 )->AddItemToMenu( "ac" );
myMenu.GetMenuList().at( 0 )->AddItemToMenu( "ad" );
// adding 4 items to the second menu of the menu bar
myMenu.GetMenuList().at( 1 )->AddItemToMenu( "ba" );
myMenu.GetMenuList().at( 1 )->AddItemToMenu( "bb" );
myMenu.GetMenuList().at( 1 )->AddItemToMenu( "bc" );
myMenu.GetMenuList().at( 1 )->AddItemToMenu( "bd" );
//app loop
while( !exit )
{
    // event loop

    // logic
    if( !myMenu.GetMenuList().empty() )
    {
        auto menus = myMenu.GetMenuList();

        for( int i = 0; i < menus.size(); ++i )
        {
            menus.at( i )->DoElement();
            // checking if the logics of the element is to be executed or not
            if( menus.at( i )->IsSelected() )
            {
                auto itemList = menus.at( i )->GetItemList();
                if( !itemList.empty() )
                {
                    for( int i = 0; i < itemList.size(); ++i )
                    {
                        if( itemList.at( i )->DoElement() )
                        {
                            std::cout << "menu at " << i << " Pressed" << std::endl;
                            std::cout << "item at " << j << " Pressed" << std::endl;
                        }
                    }
                }
            }
        }
    }

    // rendering
    myDropDown.Draw( window );
}
```

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MenuBar.h](#)

9.38 Brood::MouseHandler Class Reference

```
#include <MouseHandler.h>
```

Public Types

- enum class **ENUM_MouseCursorType** {
 CURSOR_arrow , CURSOR_wait , CURSOR_text , CURSOR_hand ,
 CURSOR_sizeHorizontal , CURSOR_help , CURSOR_notAllowed }

Static Public Member Functions

- static void **InitializeCursor** (sf::RenderWindow &a_window)
 Initializes the cursor and sets it for the window.
- static void **UpdateMousePos** (sf::RenderWindow &a_window)
 a static function to keep track of the mouse Position
- static void **ChangeCursorTo** (sf::RenderWindow &a_window, const Brood::MouseHandler::ENUM_MouseCursorType a_cursorType)
 changes the cursor to specified cursor type
- static sf::Vector2i **GetLastFrameMousePos** ()
 Getter function to get the last frame mouse position.
- static sf::Vector2i **GetCurrFrameMousePos** ()
 Getter function to get the currrent frame mouse position.
- static bool **GetlastFrameLMSStatus** ()
 Getter function to get the last frame left mouse button press status.
- static bool **GetCurrFrameLMSStatus** ()
 Getter function to get the current frame left mouse button press status.
- static bool **GetlastFrameRMSStatus** ()
 Getter function to get the last frame right mouse button press status.
- static bool **GetCurrFrameRMSStatus** ()
 Getter function to get the current frame right mouse button press status.
- static void **UpdateMouseButtonStatus** ()
 Updates mouse button status and saves the mouse button status for this frame and last frame.
- static bool **IsLeftButtonHold** ()
 Check if the left mouse button is in hold state.
- static bool **IsLeftButtonPressed** ()
 Check if the left mouse button is pressed or not.
- static bool **IsLeftButtonReleased** ()
 Check if the left mouse button is released or not.
- static bool **IsRightButtonHold** ()
 Check if the right mouse button is in hold state.
- static bool **IsRightButtonPressed** ()
 Check if the right mouse button is pressed or not.
- static bool **IsRightButtonReleased** ()
 Check if the right mouse button is released or not.
- static void **Debugger** ()
 Prints the mouse states like button press and current position.

Static Private Attributes

- static sf::Cursor [m_cursor](#)
- static sf::Vector2i [m_lastFrameMousePos](#)
- static sf::Vector2i [m_currFrameMousePos](#)
- static bool [m_lastFrameLeftMouseButtonPressed](#)
- static bool [m_lastFrameRightMouseButtonPressed](#)
- static bool [m_currFrameLeftMouseButtonPressed](#)
- static bool [m_currFrameRightMouseButtonPressed](#)

9.38.1 Member Enumeration Documentation

9.38.1.1 ENUM_MouseCursorType

```
enum class Brood::MouseHandler::ENUM\_MouseCursorType [strong]
```

Enumerator

CURSOR_arrow	
CURSOR_wait	default cursor
CURSOR_text	busy cursor
CURSOR_hand	I-beam cursor, when hovering over a field allowing text entry
CURSOR_sizeHorizontal	Pointing hand cursor
CURSOR_help	horizontal cursor with arrow head at both ends
CURSOR_notAllowed	help cursor with ? on it action not allowed cursor

9.38.2 Member Function Documentation

9.38.2.1 ChangeCursorTo()

```
void MouseHandler::ChangeCursorTo (
    sf::RenderWindow & a_window,
```

```
const Brood::MouseHandler::ENUM_MouseCursorType a_cursorType ) [static]
```

changes the cursor to specified cursor type

Parameters

<i>a_window</i>	reference to render window
<i>a_cursorType</i>	cursor type to change to

9.38.2.2 Debugger()

```
void MouseHandler::Debugger ( ) [static]
```

Prints the mouse states like button press and current position.

9.38.2.3 GetCurrFrameLMSStatus()

```
bool MouseHandler::GetCurrFrameLMSStatus ( ) [static]
```

Getter function to get the current frame left mouse button press status.

Returns

current current frame left mouse button press status

9.38.2.4 GetCurrFrameMousePos()

```
sf::Vector2i MouseHandler::GetCurrFrameMousePos ( ) [static]
```

Getter function to get the current frame mouse position.

Returns

current frame mouse position

9.38.2.5 GetCurrFrameRMSStatus()

```
bool MouseHandler::GetCurrFrameRMSStatus ( ) [static]
```

Getter function to get the current frame right mouse button press status.

Returns

currrent current frame right mouse button press status

9.38.2.6 GetlastFrameLMSStatus()

```
bool MouseHandler::GetlastFrameLMSStatus ( ) [static]
```

Getter function to get the last frame left mouse button press status.

Returns

currrent last frame left mouse button press status

9.38.2.7 GetLastFrameMousePos()

```
sf::Vector2i MouseHandler::GetLastFrameMousePos ( ) [static]
```

Getter function to get the last frame mouse position.

Returns

last frame mouse position

9.38.2.8 GetlastFrameRMSStatus()

```
bool MouseHandler::GetlastFrameRMSStatus ( ) [static]
```

Getter function to get the last frame right mouse button press status.

///

Returns

currrent last frame right mouse button press status

9.38.2.9 InitializeCursor()

```
void MouseHandler::InitializeCursor (
    sf::RenderWindow & a_window ) [static]
```

Initializes the cursor and sets it for the window.

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.38.2.10 IsLeftButtonHold()

```
bool MouseHandler::IsLeftButtonHold ( ) [static]
```

Check if the left mouse button is in hold state.

Returns

true if the left mouse button is in hold state

9.38.2.11 IsLeftButtonPressed()

```
bool MouseHandler::IsLeftButtonPressed ( ) [static]
```

Check if the left mouse button is pressed or not.

Returns

true if the left mouse button is pressed

9.38.2.12 IsLeftButtonReleased()

```
bool MouseHandler::IsLeftButtonReleased ( ) [static]
```

Check if the left mouse button is released or not.

Returns

true if the left mouse button is released

9.38.2.13 IsRightButtonHold()

```
bool MouseHandler::IsRightButtonHold ( ) [static]
```

Check if the right mouse button is in hold state.

Returns

true if the right mouse button is in hold state

9.38.2.14 IsRightButtonPressed()

```
bool MouseHandler::IsRightButtonPressed ( ) [static]
```

Check if the right mouse button is pressed or not.

Returns

true if the right mouse button is pressed

9.38.2.15 IsRightButtonReleased()

```
bool MouseHandler::IsRightButtonReleased ( ) [static]
```

Check if the right mouse button is released or not.

Returns

true if the right mouse button is released

9.38.2.16 UpdateMouseButtonStatus()

```
void MouseHandler::UpdateMouseButtonStatus ( ) [static]
```

Updates mouse button status and saves the mouse button status for this frame and last frame.

Note

only consideres left and right mouse button

9.38.2.17 UpdateMousePos()

```
void MouseHandler::UpdateMousePos ( sf::RenderWindow & a_window ) [static]
```

a static function to keep track of the mouse Position

It keep track of mouse position last frame and this frame

Parameters

<i>a_window</i>	reference to the render window
-----------------	--------------------------------

9.38.3 Member Data Documentation

9.38.3.1 m_currFrameLeftMouseButtonPressed

```
bool MouseHandler::m_currFrameLeftMouseButtonPressed [static], [private]
```

9.38.3.2 m_currFrameMousePos

```
sf::Vector2i MouseHandler::m_currFrameMousePos [static], [private]
```

9.38.3.3 m_currFrameRightMouseButtonPressed

```
bool MouseHandler::m_currFrameRightMouseButtonPressed [static], [private]
```

9.38.3.4 m_cursor

```
sf::Cursor MouseHandler::m_cursor [static], [private]
```

9.38.3.5 m_lastFrameLeftMouseButtonPressed

```
bool MouseHandler::m_lastFrameLeftMouseButtonPressed [static], [private]
```

9.38.3.6 m_lastFrameMousePos

```
sf::Vector2i MouseHandler::m_lastFrameMousePos [static], [private]
```

9.38.3.7 m_lastFrameRightMouseButtonPressed

```
bool MouseHandler::m_lastFrameRightMouseButtonPressed [static], [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MouseHandler.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[MouseHandler.cpp](#)

9.39 Brood::Application::Components::Path Class Reference

```
#include <Path.h>
```

Public Member Functions

- **Path** (*Brood::Application::Components::Tiles *a_tilePtr=nullptr, Brood::Application::Components::Deck *a_deckPtr=nullptr, Brood::Application::Components::ENUM_TileType a_tileType=Brood::Application::Components::ENUM_TileType*)
default constructor
- **~Path** ()
destructor
- **Path** (const **Path** &*a_otherPath*)
copy constructor
- **Path** & **operator=** (const **Path** &*a_otherPath*)
assignment operator
- void **InitializePath** (*Brood::Application::Data::ST_PathPrefabData &a_pathData, Brood::Application::Components::Path *a_nextPathPtr, Brood::Application::Components::Path *a_bridgeEndPathPtr, Brood::Application::Components::Deck *a_deckPtr*)
initializes the player with the passed data
- **Brood::Application::Data::ST_PathPrefabData** **GetDataToSave** ()
creates and returns player data struct
- **Brood::Application::Components::ENUM_TileType** **GetTileType** ()
Getter function to get tile type of the tile.
- **Brood::Application::Components::Tiles *** **GetTilePtr** ()
Getter function to get the pointer to the tile.
- const **Brood::Application::Components::Tiles *** **GetTilePtr** () const
Getter function to get the pointer to the tile.
- **Brood::Application::Components::Path *** **GetNextPathPtr** ()
Getter function to get the pointer to the next path.
- **Brood::Application::Components::Path *** **GetPreviousPathPtr** ()
Getter function to get the pointer to the previous path.
- **Brood::Application::Components::Path *** **GetBridgeEndPathPtr** ()
Getter function to get the pointer to the bridge end path.
- const std::vector< **Brood::Application::Components::Player** *> & **GetPlayerListPtr** () const
Getter function to get a reference to the player list.
- const **Brood::Application::Components::Deck *** **GetDeckPtr** () const
Getter function to get the pointer to the deck.
- unsigned **GetDeckIdx** ()
Getter function to get the index of the assigned deck.
- unsigned **GetNumCardDraw** ()
Getter function to get the number of card to be drawn.
- bool **GetForceDiceRoll** ()
getter function to get force dice roll
- sf::Vector2f **GetTileCenter** ()
Getter function to get the mid position of the tile in the screen space.
- bool **GetDrawLine** ()
Getter function to get the draw line.
- void **SetTileType** (*Brood::Application::Components::ENUM_TileType a_tileType*)
Setter function to Set tile type of the tile.
- void **SetTilePtr** (*Brood::Application::Components::Tiles *a_tilePtr*)

- **void SetNextPathPtr (Brood::Application::Components::Path *a_nextPathPtr)**
Setter funciton to set the pointer to the tile.
- **void SetPreviousPathPtr (Brood::Application::Components::Path *a_previousPathPtr)**
Setter funciton to set the pointer the next path.
- **void SetBridgeEndPathPtr (Brood::Application::Components::Path *a_bridgeEndPathPtr)**
Setter funciton to set the pointer the previous path.
- **void SetDeckPtr (Brood::Application::Components::Deck *a_deckPtr)**
Setter funciton to set the assigned deck pointer.
- **void SetDeckIdx (unsigned a_deckIdx)**
Setter funciton to set the assigned deck index.
- **void SetNumCardDraw (unsigned a_numCardDraw)**
Setter funciton to Set the number of card to be drawn.
- **void SetForceDiceRoll (bool a_diceRoll)**
Setter funciton to Set force dice roll.
- **void ToggleDrawLine ()**
Function to toggle the drawLine variable.
- **bool AddPlayerToList (Brood::Application::Components::Player *a_playerPtr)**
adds the player from to the player list
- **bool DeletePlayerFromList (Brood::Application::Components::Player *a_playerPtr)**
deletes the player from its list and set's the player tile pointer to nullptr
- **void UpdatePathLines ()**
Updates path lines based on nextpath, and bridgeEnd path.
- **void Draw (sf::RenderWindow &a_window)**
Draw funciton draws the tile to the screen.
- **void DrawPath (sf::RenderWindow &a_window)**
Draw funciton draws the path line to the screen.

Private Attributes

- **Brood::Application::Components::ENUM_TileType m_tileType**
holds the type of the tile this path holds
- **Brood::Application::Components::Tiles * m_tilePtr**
holds a tile inside of it
- **Brood::Application::Components::Path * m_nextPathPtr**
holds a pointer to the previous path
- **Brood::Application::Components::Path * m_previousPathPtr**
holds a pointer to the bridge end path
- **std::vector< Brood::Application::Components::Player * > m_playerPtrList**
list of player who are assigned to this path
- **Brood::Application::Components::Deck * m_deckPtr**
deck that is assigned to this path
- **unsigned m_deckIdx**
index of the assinged deck
- **unsigned m_numCardDraw**
number of card that the player are forced to draw
- **bool m_forceDiceRoll**
forces dice roll
- **bool m_drawLine**
line is drawn for tile editor
- **sf::VertexArray m_nextPathLine**
- **sf::VertexArray m_bridgePathLine**

9.39.1 Constructor & Destructor Documentation

9.39.1.1 Path() [1/2]

```
Path::Path (
    Brood::Application::Components::Tiles * a_tilePtr = nullptr,
    Brood::Application::Components::Deck * a_deckPtr = nullptr,
    Brood::Application::Components::ENUM_TileType a_tileType = Brood::Application::Components::ENUM_TileType::Blank
)
```

default constructor

Parameters

<i>a_tilePtr</i>	pointer to a tile
<i>a_tileType</i>	type of the tile; default blank

9.39.1.2 ~Path()

```
Path::~Path ( )
```

destructor

9.39.1.3 Path() [2/2]

```
Path::Path (
    const Path & a_otherPath )
```

copy constructor

Parameters

<i>a_otherPath</i>	reference to the the path object that is being copied from
--------------------	--

9.39.2 Member Function Documentation

9.39.2.1 AddPlayerToList()

```
bool Path::AddPlayerToList (
    Brood::Application::Components::Player * a_playerPtr )
```

adds the player from to the player list

Parameters

<code>a_playerPtr</code>	pointer to the player which is to be added from the list
--------------------------	--

Returns

true if the addition was successfull

9.39.2.2 DeletePlayerFromList()

```
bool Path::DeletePlayerFromList (
    Brood::Application::Components::Player * a_playerPtr )
```

deletes the player from its list and set's the player tile pointer to nullptr

Parameters

<code>a_playerPtr</code>	pointer to the player which is to be deleted from the list
--------------------------	--

Returns

true if the deletion was successfull

9.39.2.3 Draw()

```
void Path::Draw (
    sf::RenderWindow & a_window )
```

Draw funciton draws the tile to the screen.

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.39.2.4 DrawPath()

```
void Path::DrawPath (
    sf::RenderWindow & a_window )
```

Draw funciton draws the path line to the screen.

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

9.39.2.5 GetBridgeEndPathPtr()

```
Brood::Application::Components::Path * Path::GetBridgeEndPathPtr ( )
```

Getter funciton to get the pointer to the bridge end path.

Returns

pointer to bridge end path

9.39.2.6 GetDataToSave()

```
Brood::Application::Data::ST_PathPrefabData Path::GetDataToSave ( )
```

creates and returns player data struct

Returns

player data struct with the dice data in it

9.39.2.7 GetDeckIdx()

```
unsigned Path::GetDeckIdx ( )
```

Getter funciton to get the index of the assinged deck.

Returns

the index of the assinged deck

9.39.2.8 GetDeckPtr()

```
const Brood::Application::Components::Deck * Path::GetDeckPtr ( ) const
```

Getter funciton to get the pointer to the deck.

Returns

pointer to the deck

9.39.2.9 GetDrawLine()

```
bool Path::GetDrawLine ( )
```

Getter funciton to get the draw line.

Returns

draw line

9.39.2.10 GetForceDiceRoll()

```
bool Path::GetForceDiceRoll ( )
```

getter funciton to get force dice roll

Returns

force dice roll boolean

9.39.2.11 GetNextPathPtr()

```
Brood::Application::Components::Path * Path::GetNextPathPtr ( )
```

Getter funciton to get the pointer to the next path.

Returns

pointer to the next path

9.39.2.12 GetNumCardDraw()

```
unsigned Path::GetNumCardDraw ( )
```

Getter funciton to get the number of card to be drawn.

Returns

get the number of card to be drawn

9.39.2.13 GetPlayerListPtr()

```
const std::vector< Brood::Application::Components::Player * > & Path::GetPlayerListPtr ( )  
const
```

Getter function to get a reference to the player list.

Returns

const reference to the player list

9.39.2.14 GetPreviousPathPtr()

```
Brood::Application::Components::Path * Path::GetPreviousPathPtr ( )
```

Getter function to get the pointer to the previous path.

Returns

pointer to the previous path

9.39.2.15 GetTileCenter()

```
sf::Vector2f Path::GetTileCenter ( )
```

Getter function to get the mid position of the tile in the screen space.

Returns

mid position of the tile in the screen space

9.39.2.16 GetTilePtr() [1/2]

```
Brood::Application::Components::Tiles * Path::GetTilePtr ( )
```

Getter function to get the pointer to the tile.

Returns

pointer to the tile

9.39.2.17 GetTilePtr() [2/2]

```
const Brood::Application::Components::Tiles * Path::GetTilePtr ( ) const
```

Getter function to get the pointer to the tile.

Returns

pointer to the tile

9.39.2.18 GetTileType()

```
Brood::Application::Components::ENUM_TileType Path::GetTileType ( )
```

Getter function to get tile type of the tile.

Returns

tile type of the tile

9.39.2.19 InitializePath()

```
void Path::InitializePath (
    Brood::Application::Data::ST_PathPrefabData & a_pathData,
    Brood::Application::Components::Path * a_nextPathPtr,
    Brood::Application::Components::Path * a_bridgeEndPathPtr,
    Brood::Application::Components::Deck * a_deckPtr )
```

initializes the player with the passed data

Parameters

<i>a_deckData</i>	reference of the player data
<i>a_nextPathPtr</i>	pointer to the next path
<i>a_bridgeEndPathPtr</i>	pointer to the bridge end path
<i>a_deckPtr</i>	pointer to the assigned deck

9.39.2.20 operator=()

```
Brood::Application::Components::Path & Path::operator= (
    const Path & a_otherPath )
```

assignment operator

Parameters

<i>a_otherPath</i>	reference to the the path structue that is being copied from
--------------------	--

9.39.2.21 SetBridgeEndPathPtr()

```
void Path::SetBridgeEndPathPtr (
    Brood::Application::Components::Path * a_bridgeEndPathPtr )
```

Setter funciton to set the pointer the bridge end path.

Parameters

<i>a_bridgeEndPathPtr</i>	pointer to the bridge end path
---------------------------	--------------------------------

9.39.2.22 SetDeckIdx()

```
void Path::SetDeckIdx (
    unsigned a_deckIdx )
```

Setter funciton to set the assigned deck index.

Parameters

<i>a_deckPtr</i>	index of the assigned deck
------------------	----------------------------

9.39.2.23 SetDeckPtr()

```
void Path::SetDeckPtr (
    Brood::Application::Components::Deck * a_deckPtr )
```

Setter funciton to set the assigned deck pointer.

Parameters

<i>a_deckPtr</i>	pointer to the assigned deck pointer
------------------	--------------------------------------

9.39.2.24 SetForceDiceRoll()

```
void Path::SetForceDiceRoll (
    bool a_diceRoll )
```

Setter funciton to Set force dice roll.

Parameters

<i>a_diceRoll</i>	true if it is too force dice roll
-------------------	-----------------------------------

9.39.2.25 SetNextPathPtr()

```
void Path::SetNextPathPtr (
    Brood::Application::Components::Path * a_nextPathPtr )
```

Setter funciton to set the pointer the next path.

Parameters

<i>a_nextPathPtr</i>	pointer to the next path
----------------------	--------------------------

9.39.2.26 SetNumCardDraw()

```
void Path::SetNumCardDraw (
    unsigned a_numCardDraw )
```

Setter funciton to Set the number of card to be drawn.

Parameters

<i>a_deckPtr</i>	index of the assigned deck
------------------	----------------------------

9.39.2.27 SetPreviousPathPtr()

```
void Path::SetPreviousPathPtr (
    Brood::Application::Components::Path * a_previousPathPtr )
```

Setter funciton to set the pointer the previous path.

Parameters

<i>a_previousPathPtr</i>	pointer to the previous path
--------------------------	------------------------------

9.39.2.28 SetTilePtr()

```
void Path::SetTilePtr (
    Brood::Application::Components::Tiles * a_tilePtr )
```

Setter function to set the pointer to the tile.

Parameters

<i>a_tilePtr</i>	pointer to the tile
------------------	---------------------

9.39.2.29 SetTileType()

```
void Path::SetTileType (
    Brood::Application::Components::ENUM_TileType a_tileType )
```

Setter function to set tile type of the tile.

Parameters

<i>a_tileType</i>	tile type of the tile
-------------------	-----------------------

9.39.2.30 ToggleDrawLine()

```
void Path::ToggleDrawLine ( )
```

Function to toggle the drawLine variable.

9.39.2.31 UpdatePathLines()

```
void Path::UpdatePathLines ( )
```

Updates path lines based on nextpath, and bridgeEnd path.

9.39.3 Member Data Documentation

9.39.3.1 m_bridgeEndPathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::Path::m_bridgeEndPathPtr  
[private]
```

holds a pointer to the bridge end path

9.39.3.2 m_bridgePathLine

```
sf::VertexArray Brood::Application::Components::Path::m_bridgePathLine [private]
```

line to the bridge path this is always going to be an array of two vertex and the vertext at index 0 is midPoint of the tile and the vertext at index 1 is midPoint of the nextTile

9.39.3.3 m_deckIdx

```
unsigned Brood::Application::Components::Path::m_deckIdx [private]
```

index of the assinged deck

9.39.3.4 m_deckPtr

```
Brood::Application::Components::Deck* Brood::Application::Components::Path::m_deckPtr [private]
```

deck that is assigned to this path

9.39.3.5 m_drawLine

```
bool Brood::Application::Components::Path::m_drawLine [private]
```

line is drawn for tile editor

9.39.3.6 m_forceDiceRoll

```
bool Brood::Application::Components::Path::m_forceDiceRoll [private]
```

forces dice roll

9.39.3.7 m_nextPathLine

```
sf::VertexArray Brood::Application::Components::Path::m_nextPathLine [private]
```

line to the next path; this is always going to be an array of two vertex and the vertext at index 0 is midPoint of the tile and the vertext at index 1 is midPoint of the nextTile

9.39.3.8 m_nextPathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::Path::m_nextPathPtr  
[private]
```

holds a pointer to the next path next path can only be one step away form the tile

9.39.3.9 m_numCardDraw

```
unsigned Brood::Application::Components::Path::m_numCardDraw [private]
```

number of card that the player are forced to draw

9.39.3.10 m_playerPtrList

```
std::vector<Brood::Application::Components::Player*> Brood::Application::Components::Path::m←  
_playerPtrList [private]
```

list of player who are assigned to this path

9.39.3.11 m_previousPathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::Path::m_previousPathPtr  
[private]
```

holds a pointer to the previous path

9.39.3.12 m_tilePtr

```
Brood::Application::Components::Tiles* Brood::Application::Components::Path::m_tilePtr [private]
```

holds a tile inside of it

9.39.3.13 m_tileType

```
Brood::Application::Components::ENUM_TileType Brood::Application::Components::Path::m_tileType [private]
```

holds the type of the tile this path holds

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Path.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Path.cpp](#)

9.40 Path Class Reference

[Path](#) is a wrapper for tile class.

```
#include <Path.h>
```

9.40.1 Detailed Description

[Path](#) is a wrapper for tile class.

It stores a pointer to a tile and other information necessary

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Path.h](#)

9.41 Brood::Application::Components::Player Class Reference

```
#include <Player.h>
```

Public Member Functions

- `Player (Brood::Application::Components::Path *a_pathPtr=nullptr, unsigned a_positionOffsetX=0, unsigned a_positionOffsetY=0)`
Default Constructor.
- `~Player ()`
Default Desturctor.
- `Player (const Player &a_otherPlayer)`
Copy constructor.
- `Player & operator= (const Player &a_otherPlayer)`
assignment operator
- `void InitializePlayer (Brood::Application::Data::ST_PlayerPrefabData &a_playerData, Brood::Application::Components::Path *a_playerCurrPath)`
initializes the player with the passed data
- `Brood::Application::Data::ST_PlayerPrefabData GetDataToSave ()`
creates and returns player data struct
- `void SaveDataToFile (Brood::Application::FileAccess *a_fileAccessPtr)`
loads the Player data from passed file
- `void LoadDataFromFile (Brood::Application::FileAccess *a_fileAccessPtr, Brood::Application::Components::Board *a_gameBoard)`
loads the Player and its path data from passed file
- `void UpdatePathptr (Brood::Application::Components::Path *a_playerNewPathPtr)`
Funciton to update the pointer to the path.
- `void SetPlayerSizeX (float a_sizeX)`
Setter funciton to set the x-size.
- `void SetPlayerSizeY (float a_sizeY)`
Setter funciton to set the x-size.
- `void SetPositionOffsetX (int a_positionOffsetX)`
Setter funciton to set the x-offset for player positon.
- `void SetPositionOffsetY (int a_positionOffsetY)`
Setter funciton to set the y-offset for player positon.
- `const Brood::Application::Components::Path * GetCurrPathPtr () const`
Getter function to get the pointer to the tile that the player is currently on.
- `sf::RectangleShape & GetPlayerBody ()`
Getter function to get a reference to player body.
- `Brood::SpriteHandler & GetPlayerSpriteBody ()`
Getter function to get a reference to the player sprite body.
- `int GetPositionOffsetX ()`
Getter function to get the x-offset.
- `int GetPositionOffsetY ()`
Getter function to get the y-offset.
- `void Draw (sf::RenderWindow &a_window)`
Draw funciton draws tiles to the screen.

Private Member Functions

- `void UpdatePosition ()`
updates the player position with respect to tile

Private Attributes

- sf::RectangleShape `m_playerBody`
- `Brood::SpriteHandler m_spriteBody`
- int `m_positionOffsetX`
- int `m_positionOffsetY`
- `Brood::Application::Components::Path * m_playerCurrPathPtr`

9.41.1 Constructor & Destructor Documentation

9.41.1.1 Player() [1/2]

```
Player::Player (
    Brood::Application::Components::Path * a_pathPtr = nullptr,
    unsigned a_positionOffsetX = 0,
    unsigned a_positionOffsetY = 0 )
```

Default Constructor.

Parameters

<code>a_pathite</code>	a iterator that holds to Brood::Application::Components::st_path node which contains the start row and column number inside it
<code>a_positionOffsetX</code>	position offset of the path relative to tile's x position
<code>a_positionOffsetY</code>	position offset of the path relative to tile's Y position

9.41.1.2 ~Player()

```
Player::~Player ( )
```

Default Desturctor.

9.41.1.3 Player() [2/2]

```
Player::Player (
    const Player & a_otherPlayer )
```

Copy constructor.

Parameters

<code>a_otherPlayer</code>	reference to the the other player object that is being copied from
----------------------------	--

9.41.2 Member Function Documentation

9.41.2.1 Draw()

```
void Player::Draw (
    sf::RenderWindow & a_window )
```

Draw function draws tiles to the screen.

Parameters

a_window	reference to the render window
----------	--------------------------------

9.41.2.2 GetCurrPathPtr()

```
const Brood::Application::Components::Path * Player::GetCurrPathPtr ( ) const
```

Getter function to get the pointer to the tile that the player is currently on.

Returns

pointer to the tile that the player is currently on

9.41.2.3 GetDataToSave()

```
Brood::Application::Data::ST_PlayerPrefabData Player::GetDataToSave ( )
```

creates and returns player data struct

Returns

player data struct with the dice data in it

9.41.2.4 GetPlayerBody()

```
sf::RectangleShape & Player::GetPlayerBody ( )
```

Getter function to get a reference to player body.

Returns

a reference to player body

9.41.2.5 GetPlayerSpriteBody()

```
Brood::SpriteHandler & Player::GetPlayerSpriteBody ( )
```

Getter function to get a reference to the player sprite body.

Returns

a reference to player body

9.41.2.6 GetPositionOffsetX()

```
int Player::GetPositionOffsetX ( )
```

Getter function to get the x-offset.

Returns

a reference to player body

9.41.2.7 GetPositionOffsetY()

```
int Player::GetPositionOffsetY ( )
```

Getter function to get the y-offset.

Returns

a reference to player body

9.41.2.8 InitializePlayer()

```
void Player::InitializePlayer (
    Brood::Application::Data::ST_PlayerPrefabData & a_playerData,
    Brood::Application::Components::Path * a_playerCurrPath )
```

initializes the player with the passed data

Parameters

<i>a_deckData</i>	reference of the player data
<i>a_playerCurrPath</i>	path that teh player is currently on

9.41.2.9 LoadDataFromFile()

```
void Player::LoadDataFromFile (
    Brood::Application::FileAccess * a_fileAccessPtr,
    Brood::Application::Components::Board * a_gameBoard )
```

loads the [Player](#) and its path data from passed file

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
------------------------------	---------------------------------

9.41.2.10 operator=()

```
Brood::Application::Components::Player & Player::operator= (
    const Player & a_otherPlayer )
```

assignment operator

Parameters

<code>a_otherPlayer</code>	reference to the the other player object that is being copied from
----------------------------	--

9.41.2.11 SaveDataToFile()

```
void Player::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the [Player](#) data from passed file

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
------------------------------	---------------------------------

9.41.2.12 SetPlayerSizeX()

```
void Player::SetPlayerSizeX (
    float a_sizeX )
```

Setter function to set the x-size.

The offset is relative to the tile the player is currently on. It also updates the players position

Parameters

<i>a_sizeX</i>	player's x size
----------------	-----------------

9.41.2.13 SetPlayerSizeY()

```
void Player::SetPlayerSizeY (
    float a_sizeY )
```

Setter function to set the x-size.

The offset is relative to the tile the player is currently on. It also updates the players position

Parameters

<i>a_sizeX</i>	player's x size
----------------	-----------------

9.41.2.14 SetPositionOffsetX()

```
void Player::SetPositionOffsetX (
    int a_positionOffsetX )
```

Setter function to set the x-offset for player position.

The offset is relative to the tile the player is currently on. It also updates the players position

Parameters

<i>a_positionOffsetX</i>	position offset of the path relative to tile's x position
--------------------------	---

9.41.2.15 SetPositionOffsetY()

```
void Player::SetPositionOffsetY (
    int a_positionOffsetY )
```

Setter function to set the y-offset for player position.

The offset is relative to the tile the player is currently on. It also updates the players position

Parameters

<i>a_positionOffsetY</i>	position offset of the path relative to tile's Y position
--------------------------	---

9.41.2.16 UpdatePathptr()

```
void Player::UpdatePathptr (
    Brood::Application::Components::Path * a_playerNewPathPtr )
```

Funciton to update the pointer to the path.

It also updates the position of the player body. If there are already player in the tile space it offsets the player position by 3 px to the left and down

Parameters

<i>a_playerNewPathPtr</i>	pointer to the new path
---------------------------	-------------------------

9.41.2.17 UpdatePosition()

```
void Player::UpdatePosition ( ) [private]
```

updates the player position with respect to tile

9.41.3 Member Data Documentation**9.41.3.1 m_playerBody**

```
sf::RectangleShape Brood::Application::Components::Player::m_playerBody [private]
```

9.41.3.2 m_playerCurrPathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::Player::m_playerCurr←
PathPtr [private]
```

9.41.3.3 m_positionOffsetX

```
int Brood::Application::Components::Player::m_positionOffsetX [private]
```

9.41.3.4 m_positionOffsetY

```
int Brood::Application::Components::Player::m_positionOffsetY [private]
```

9.41.3.5 m_spriteBody

[Brood::SpriteHandler](#) Brood::Application::Components::Player::m_spriteBody [private]

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Player.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Player.cpp](#)

9.42 Player Class Reference

A [Player](#) Class.

```
#include "Player.h"
```

9.42.1 Detailed Description

A [Player](#) Class.

Todo make it so that if two or more player are in the same tile then offset the player so that they are visually. One way to do this could be to add a vector of playerPtr to tile where if a player get to the tile player will push itself back to this list and will save ther player index to itself. and once a player moves on it will call a function to update itself leaving the tile thus updating other player position in the tile list

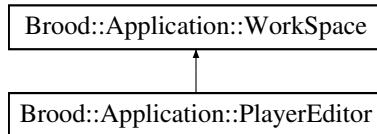
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Player.h](#)

9.43 Brood::Application::PlayerEditor Class Reference

```
#include <PlayerEditor.h>
```

Inheritance diagram for Brood::Application::PlayerEditor:



Public Member Functions

- `PlayerEditor (Brood::Application::Components::GameDataManager *aGameData, sf::RectangleShape *a←_panelPtr)`
default constructor
- `~PlayerEditor ()`
default destructor
- `void InitializeWorkSpace ()`
Initializes the work space.
- `void Update ()`
Updates function.
- `void UpdateAllDisplayElement ()`
changes the displayed data on the panel for current active player
- `void Draw (sf::RenderWindow &a_window)`
Draw funciton.
- `void Debugger ()`
debugger funciton

Private Member Functions

- `void UpdateStartRowNumber ()`
checks if the user interacted with the start row panel
- `void UpdateStartColNumber ()`
checks if the user interacted with the start column panel
- `void UpdateCurrSelectedPlayerIdx ()`
checks if the user interacted with the current selcted player index panel
- `void UpdateCurrPlayerSizeX ()`
checks if the user interacted with the player x size panel
- `void UpdateCurrPlayerSizeY ()`
checks if the user interacted with the player y size panel
- `void UpdateCurrPlayerOffsetX ()`
checks if the user interacted with the player x Offset panel
- `void UpdateCurrPlayerOffsetY ()`
checks if the user interacted with the player Y Offset panel
- `void UpdateCurrPlayerFileTexture ()`
checks if the user interacted with the laod player texture panel

Private Attributes

- sf::RectangleShape * `m_panelBodyPtr`
holds the panel body
- Brood::Application::Components::Board * `m_boardPtr`
board for the game
- Brood::Application::Components::PlayerManager * `m_playerManagerPtr`
pointer to the player manager
- Brood::BroodUI::TextBox * `m_txtSettingTitle`
- Brood::BroodUI::TextBox * `m_txtPlayerStartRowPrompt`
holds the text start row prompt
- Brood::BroodUI::Button * `m_btnPlayerDecStartRowNum`
clicking it will decrease the start row of player number by 1
- Brood::BroodUI::TextBox * `m_txtPlayerStartRowNum`
displays the Current start row number of the player
- Brood::BroodUI::Button * `m_btnPlayerIncStartRowNum`
clicking it will increase the start row of the player number by 1
- Brood::BroodUI::TextBox * `m_txtPlayerStartColPrompt`
holds the text start column prompt
- Brood::BroodUI::Button * `m_btnPlayerDecStartColNum`
clicking it will decrease the start column of player number by 1
- Brood::BroodUI::TextBox * `m_txtPlayerStartColNum`
displays the Current start column number of the player
- Brood::BroodUI::Button * `m_btnPlayerIncStartColNum`
clicking it will increase the start column of the player number by 1
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdxPrompt`
holds the text selected Index prompt
- Brood::BroodUI::Button * `m_btnPlayerDecCurrlIdx`
clicking it will decrease the current selected player Index number by 1
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdx`
displays the current selected player Index
- Brood::BroodUI::Button * `m_btnPlayerIncCurrlIdx`
clicking it will increase the current selected player Index number by 1
- Brood::BroodUI::TextBox * `m_txtDrawPlayerAtCurrlIdxPromt`
*holds the text Current *Player* Index prompt*
- Brood::BroodUI::DropDownInput * `m_ddiDrawPlayerAtCurrlIdx`
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdxSizeXPrompt`
*holds the text *Player* X-Size*
- Brood::BroodUI::Button * `m_btnPlayerDecCurrlIdxSizeX`
clicking it will decrease the current selected player x-size number by 5
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdxSizeX`
displays the current selected player x-size
- Brood::BroodUI::Button * `m_btnPlayerIncCurrlIdxSizeX`
clicking it will increase the current selected player x-size number by 5
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdxSizeYPrompt`
*holds the text *Player* y-Size*
- Brood::BroodUI::Button * `m_btnPlayerDecCurrlIdxSizeY`
clicking it will decrease the current selected player y-size number by 5
- Brood::BroodUI::TextBox * `m_txtPlayerCurrlIdxSizeY`
displays the current selected player y-size
- Brood::BroodUI::Button * `m_btnPlayerIncCurrlIdxSizeY`

- clicking it will increase the current selected player y-size number by 5
- Brood::BroodUI::TextBox * m_txtPlayerCurrlIdxOffsetXPrompt
 - holds the text Player X-Offset*
- Brood::BroodUI::Button * m_btnPlayerDecCurrlIdxOffsetX
 - clicking it will decrease the current selected player X-Offset number by 5*
- Brood::BroodUI::TextBox * m_txtPlayerCurrlIdxOffsetX
 - displays the current selected player X-Offset*
- Brood::BroodUI::Button * m_btnPlayerIncCurrlIdxOffsetX
 - clicking it will increase the current selected player X-Offset number by 5*
- Brood::BroodUI::TextBox * m_txtPlayerCurrlIdxOffsetYPrompt
 - holds the text Player y-Offset prompt*
- Brood::BroodUI::Button * m_btnPlayerDecCurrlIdxOffsetY
 - clicking it will decrease the current selected player y-Offset number by 5*
- Brood::BroodUI::TextBox * m_txtPlayerCurrlIdxOffsetY
 - displays the current selected player y-Offset*
- Brood::BroodUI::Button * m_btnPlayerIncCurrlIdxOffsetY
 - clicking it will increase the current selected player y-Offset number by 5*
- Brood::BroodUI::TextBox * m_txtpPlayerFileNamePrompt
 - holds the Player texture filename prompt*
- Brood::BroodUI::TextBox * m_txtpPlayerFileInput
 - holds the file name user enters*
- Brood::BroodUI::Button * m_btnPlayerOpenFile
 - clicking it will load the player file*

Additional Inherited Members

9.43.1 Constructor & Destructor Documentation

9.43.1.1 PlayerEditor()

```
PlayerEditor::PlayerEditor (
    Brood::Application::Components::GameDataManager * aGameData,
    sf::RectangleShape * a_panelPtr )
```

default constructor

Parameters

aGameData	pointer to the game data object
a_panelPtr	pointer to the panel body object

9.43.1.2 ~PlayerEditor()

```
PlayerEditor::~PlayerEditor ( )
```

default destructor

9.43.2 Member Function Documentation

9.43.2.1 Debugger()

```
void PlayerEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.43.2.2 Draw()

```
void PlayerEditor::Draw ( sf::RenderWindow & a_window ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Parameters

a_window	reference to the render window
----------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.43.2.3 InitializeWorkSpace()

```
void PlayerEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the board, setting section Dropdown Input.

Implements [Brood::Application::WorkSpace](#).

9.43.2.4 Update()

```
void PlayerEditor::Update ( ) [virtual]
```

Updates function.

@virtual

It Updates the following panel element : [Player](#) start row, [Player](#) start column, Current player index, Draw current player, Current player X-Size, current player Y-size, Current player X-offset, Current player Y-offset, and Current player texture file name

Implements [Brood::Application::WorkSpace](#).

9.43.2.5 UpdateAllDisplayElement()

```
void PlayerEditor::UpdateAllDisplayElement ( ) [virtual]
```

changes the displayed data on the panel for current active player

@virtual

Implements [Brood::Application::WorkSpace](#).

9.43.2.6 UpdateCurrPlayerFileTexture()

```
void PlayerEditor::UpdateCurrPlayerFileTexture ( ) [private]
```

checks if the user interacted with the load player texture panel

Load player texture file panel contains load player texture file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

the Texture for the file should be in "cwd\Assets\Textures\PlayerTexture\" as it loads them from there

9.43.2.7 UpdateCurrPlayerOffsetX()

```
void PlayerEditor::UpdateCurrPlayerOffsetX ( ) [private]
```

checks if the user interacted with the player x Offset panel

Player X Offset panel contains current player X Offset prompt textbox, current player X Offset value text box, button to increase the current player X Offset, and button to decrease the current player X Offset.

Only the button to increase the current player X Offset, and button to decrease the current player X Offset are interactable

If the interactable button was pressed then the current player X Offset is increased or decreased by 1 if the current player does not go outside the board window

9.43.2.8 UpdateCurrPlayerOffsetY()

```
void PlayerEditor::UpdateCurrPlayerOffsetY ( ) [private]
```

checks if the user interacted with the player Y Offset panel

Player Y Offset panel contains current player Y Offset prompt textbox, current player Y Offset value text box, button to increase the current player Y Offset, and button to decrease the current player Y Offset.

Only the button to increase the current player Y Offset, and button to decrease the current player Y Offset are interactable

If the interactable button was pressed then the current player Y Offset is increased or decreased by 1 if the current player does not go outside the board window

9.43.2.9 UpdateCurrPlayerSizeX()

```
void PlayerEditor::UpdateCurrPlayerSizeX ( ) [private]
```

checks if the user interacted with the player x size panel

Player X size panel contains current player X size prompt textbox, current player X size value text box, button to increase the current player X size, and button to decrease the current player X size.

Only the button to increase the current player X size, and button to decrease the current player X size are interactable

If the interactable button was pressed then the current player X size is increased or decreased by 5 if the current player does not go outside the board window

9.43.2.10 UpdateCurrPlayerSizeY()

```
void PlayerEditor::UpdateCurrPlayerSizeY ( ) [private]
```

checks if the user interacted with the player y size panel

Player Y size panel contains current player y size prompt textbox, current player y size value text box, button to increase the current player y size, and button to decrease the current player y size.

Only the button to increase the current player y size, and button to decrease the current player y size are interactable

If the interactable button was pressed then the current player y size is increased or decreased by 5 if the current player does not go outside the board window

9.43.2.11 UpdateCurrSelectedPlayerIdx()

```
void PlayerEditor::UpdateCurrSelectedPlayerIdx ( ) [private]
```

checks if the user interacted with the current selected player index panel

Current selected player index panel contains current selected player index prompt textbox, current selected player index value text box, button to increase the current selected player index, and button to decrease the current selected player index

Only the button to increase the current selected player index, and button to decrease the current selected player index are interactable

If the interactable button was pressed then the current selected player index is increased or decreased by 1. current selected player index should be more than or equal to 0 but less than minimum player number

9.43.2.12 UpdateStartColNumber()

```
void PlayerEditor::UpdateStartColNumber ( ) [private]
```

checks if the user interacted with the start column panel

Start column panel contains the start column number prompt textbox, start column number value text box, button to increase the start column number, and button to decrease the start column number

Only the button to increase the start column number, and button to decrease the start column number are interactable

If the interactable button was pressed then the start column is increased or decreased by 1. Start column number should be more than or equal to 0 but less than board column number

9.43.2.13 UpdateStartRowNumber()

```
void PlayerEditor::UpdateStartRowNumber ( ) [private]
```

checks if the user interacted with the start row panel

Start row panel contains the start row number prompt textbox, start row number value text box, button to increase the start row number, and button to decrease the start row number

Only the button to increase the start row number, and button to decrease the start row number are interactable

If the interactable button was pressed then the start row is increased or decreased by 1. Start row number should be more than or equal to 0 but less than board row number

9.43.3 Member Data Documentation

9.43.3.1 m_boardPtr

```
Brood::Application::Components::Board* Brood::Application::PlayerEditor::m_boardPtr [private]
```

board for the game

9.43.3.2 m_btnPlayerDecCurrIdx

```
Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecCurrIdx [private]
```

clicking it will decrease the current selected player Index number by 1

9.43.3.3 m_btnPlayerDecCurrIdxOffsetX

```
Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecCurrIdxOffsetX [private]
```

clicking it will decrease the current selected player X-Offset number by 5

9.43.3.4 m_btnPlayerDecCurrIdxOffsetY

```
Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecCurrIdxOffsetY [private]
```

clicking it will decrease the current selected player y-Offset number by 5

9.43.3.5 m_btnPlayerDecCurrIdxSizeX

```
Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecCurrIdxSizeX [private]
```

clicking it will decrease the current selected player x-size number by 5

9.43.3.6 m_btnPlayerDecCurrIdxSizeY

```
Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecCurrIdxSizeY [private]
```

clicking it will decrease the current selected player y-size number by 5

9.43.3.7 m_btnPlayerDecStartColNum

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecStartColNum [private]`

clicking it will decrease the start column of player number by 1

9.43.3.8 m_btnPlayerDecStartRowNum

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerDecStartRowNum [private]`

clicking it will decrease the start row of player number by 1

9.43.3.9 m_btnPlayerIncCurrlIdx

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncCurrlIdx [private]`

clicking it will increase the current selected player Indexr number by 1

9.43.3.10 m_btnPlayerIncCurrlIdxOffsetX

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncCurrlIdxOffsetX [private]`

clicking it will increase the current selected player X-Offset number by 5

9.43.3.11 m_btnPlayerIncCurrlIdxOffsetY

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncCurrlIdxOffsetY [private]`

clicking it will increase the current selected player y-Offset number by 5

9.43.3.12 m_btnPlayerIncCurrlIdxSizeX

`Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncCurrlIdxSizeX [private]`

clicking it will increase the current selected player x-size number by 5

9.43.3.13 m_btnPlayerIncCurrIdxSizeY

Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncCurrIdxSizeY [private]

clicking it will increase the current selected player y-size number by 5

9.43.3.14 m_btnPlayerIncStartColNum

Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncStartColNum [private]

clicking it will increase the start column of the player number by 1

9.43.3.15 m_btnPlayerIncStartRowNum

Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerIncStartRowNum [private]

clicking it will increase the start row of the player number by 1

9.43.3.16 m_btnPlayerOpenFile

Brood::BroodUI::Button* Brood::Application::PlayerEditor::m_btnPlayerOpenFile [private]

clicking it will load the player file

9.43.3.17 m_ddiDrawPlayerAtCurrIdx

Brood::BroodUI::DropDownInput* Brood::Application::PlayerEditor::m_ddiDrawPlayerAtCurrIdx [private]

holds if the player at current index is to be draw or not

9.43.3.18 m_panelBodyPtr

sf::RectangleShape* Brood::Application::PlayerEditor::m_panelBodyPtr [private]

holds the panel body

9.43.3.19 m_playerManagerPtr

```
Brood::Application::Components::PlayerManager* Brood::Application::PlayerEditor::m_player←
ManagerPtr [private]
```

pointer to the player manager

9.43.3.20 m_txtDrawPlayerAtCurrlIdxPromt

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtDrawPlayerAtCurrlIdxPromt [private]
```

holds the text Current [Player](#) Index prompt

9.43.3.21 m_txtPlayerCurrlIdx

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdx [private]
```

displays the current selected player Index

9.43.3.22 m_txtPlayerCurrlIdxOffsetX

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxOffsetX [private]
```

displays the current selected player X-Offset

9.43.3.23 m_txtPlayerCurrlIdxOffsetXPrompt

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxOffsetXPrompt
[private]
```

holds the text [Player](#) X-Offset

9.43.3.24 m_txtPlayerCurrlIdxOffsetY

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxOffsetY [private]
```

displays the current selected player y-Offset

9.43.3.25 m_txtPlayerCurrlIdxOffsetYPrompt

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxOffsetYPrompt [private]`

holds the text **Player** y-Offset prompt

9.43.3.26 m_txtPlayerCurrlIdxPrompt

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxPrompt [private]`

holds the text selected Index prompt

9.43.3.27 m_txtPlayerCurrlIdxSizeX

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxSizeX [private]`

displays the current selected player x-size

9.43.3.28 m_txtPlayerCurrlIdxSizeXPrompt

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxSizeXPrompt [private]`

holds the text **Player** X-Size

9.43.3.29 m_txtPlayerCurrlIdxSizeY

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxSizeY [private]`

displays the current selected player y-size

9.43.3.30 m_txtPlayerCurrlIdxSizeYPrompt

`Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerCurrlIdxSizeYPrompt [private]`

holds the text **Player** y-Size

9.43.3.31 m_txtPlayerFileInput

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerFileInput [private]
```

holds the file name user enters

9.43.3.32 m_txtplayerFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtplayerFileNamePrompt [private]
```

holds the [Player](#) texture filename prompt

9.43.3.33 m_txtPlayerStartColNum

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerStartColNum [private]
```

displays the Current start column number of the player

9.43.3.34 m_txtPlayerStartColPrompt

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerStartColPrompt [private]
```

holds the text start column prompt

9.43.3.35 m_txtPlayerStartRowNum

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerStartRowNum [private]
```

displays the Current start row number of the player

9.43.3.36 m_txtPlayerStartRowPrompt

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtPlayerStartRowPrompt [private]
```

holds the text start row prompt

9.43.3.37 m_txtSettingTitle

```
Brood::BroodUI::TextBox* Brood::Application::PlayerEditor::m_txtSettingTitle [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[PlayerEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[PlayerEditor.cpp](#)

9.44 PlayerEditor Class Reference

It allows the users to tweek the player setting.

```
#include "PlayerEditor.h"
```

9.44.1 Detailed Description

It allows the users to tweek the player setting.

It is a container that bundles elements that is present in [PlayerEditor](#) workspace

It allows to tweek the following settings: [Player](#) start row, [Player](#) start column, Current player index, Draw current player, Current player X-Size, current player Y-size, Current player X-offset, Current player Y-offset, and Current player texture file name

Derived form Workspace class

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[PlayerEditor.h](#)

9.45 Brood::Application::Components::PlayerManager Class Reference

```
#include <PlayerManager.h>
```

Public Member Functions

- `PlayerManager ()`
Default Constructor.
- `~PlayerManager ()`
Default destructor.
- `PlayerManager (const Brood::Application::Components::PlayerManager &a_otherPlayerManager)`
Copy constructor.
- `PlayerManager & operator= (const Brood::Application::Components::PlayerManager &a_otherPlayerManager)`
assignment operator
- `void InitializePlayerManager (Brood::Application::Components::Path *a_pathPtr)`
Initializes the player manager.
- `void InitializePlayerManager (Brood::Application::Data::ST_PlayerManagerData &a_playerData, Brood::Application::Components::Path *a_playerCurrPath)`
initializes the player manager with the passed data
- `Brood::Application::Data::ST_PlayerManagerData GetDataToSave ()`
creates and returns player data struct
- `void SaveDataToFile (Brood::Application::FileAccess *a_fileAccessPtr)`
loads the PlayerManager data from passed file
- `void LoadDataFromFile (Brood::Application::FileAccess *a_fileAccessPtr, Brood::Application::Components::Board *a_gameBoard)`
loads the PlayerManager and its path data from passed file
- `const unsigned GetMaxPlayer () const`
Getter function to get the maximum number of player.
- `const unsigned GetMinPlayer () const`
Getter function to get the minimum number of player.
- `const unsigned GetCurrActivePlayerIdx () const`
Getter function to get the current Player index.
- `const Brood::Application::Components::Path * GetPlayerStartPath () const`
Getter function to get the start row of player.
- `std::vector< Brood::Application::Components::Player * >::iterator GetAllPlayerBegin ()`
Getter function; for begin of player list.
- `std::vector< Brood::Application::Components::Player * >::iterator GetAllPlayerEnd ()`
Getter function; for end of path list.
- `void SetMaxPlayer (unsigned a_maxPlayer)`
Setter function to set the maximum number of player.
- `void SetMinPlayer (unsigned a_minPlayer)`
Setter function to set the minimum number of player.
- `void SetCurrActivePlayerIdx (unsigned a_currActivePlayerIdx)`
Setter function to set the current player Index.
- `void SetPlayerStartPath (Brood::Application::Components::Path *a_startPathPtr)`
Setter function to set the start path for new additional player.
- `void SetPathForPlayerAtCurrlIdx (Brood::Application::Components::Path *a_pathPtr)`
Setter function to set the path for current active player.
- `Brood::Application::Components::Player * GetPlayerAtCurrlIdx ()`
Getter function to get the current active player from the list.
- `Brood::Application::Components::Player * GetNextPlayer ()`
Getter function to get next player in the list.
- `void ReplacePlayerAt (unsigned a_index, Brood::Application::Components::Player *a_playerPtr=nullptr)`
replaces the a new player at passed index with a copy of passed index
- `void Draw (sf::RenderWindow &a_window)`
Draws the player to the screen.

Private Attributes

- std::vector< Brood::Application::Components::Player * > m_allPlayers
- unsigned m_currActivePlayerIdx
- unsigned m_maxPlayer
- unsigned m_minPlayer
- Brood::Application::Components::Path * m_startPathPtr

9.45.1 Constructor & Destructor Documentation

9.45.1.1 PlayerManager() [1/2]

```
PlayerManager::PlayerManager ( )
```

Default Constructor.

9.45.1.2 ~PlayerManager()

```
PlayerManager::~PlayerManager ( )
```

Default destructor.

9.45.1.3 PlayerManager() [2/2]

```
PlayerManager::PlayerManager (
    const Brood::Application::Components::PlayerManager & a_otherPlayerManager )
```

Copy constructor.

Parameters

a_otherPlayerManager	reference to the the other playermanager object that is being copied from
----------------------	---

9.45.2 Member Function Documentation

9.45.2.1 Draw()

```
void PlayerManager::Draw (
    sf::RenderWindow & a_window )
```

Draws the player to the screen.

Parameters

<code>a_window</code>	reference to the render screen
-----------------------	--------------------------------

9.45.2.2 GetAllPlayerBegin()

```
std::vector< Brood::Application::Components::Player * >::iterator PlayerManager::GetAll<->  
PlayerBegin ( )
```

Getter function; for begin of player list.

Returns

a iterator that points to the begining of the player list

9.45.2.3 GetAllPlayerEnd()

```
std::vector< Brood::Application::Components::Player * >::iterator PlayerManager::GetAll<->  
PlayerEnd ( )
```

Getter function; for end of path list.

Returns

a iterator that points to the end of the path list

9.45.2.4 GetCurrActivePlayerIdx()

```
const unsigned PlayerManager::GetCurrActivePlayerIdx ( ) const
```

Getter function to get the current [Player](#) index.

Returns

maximum number of player

9.45.2.5 GetDataToSave()

```
Brood::Application::Data::ST_PlayerManagerData PlayerManager::GetDataToSave ( )
```

creates and returns player data struct

Returns

player data struct with the dice data in it

9.45.2.6 GetMaxPlayer()

```
const unsigned PlayerManager::GetMaxPlayer ( ) const
```

Getter function to get the maximum number of player.

Returns

maximum number of player

9.45.2.7 GetMinPlayer()

```
const unsigned PlayerManager::GetMinPlayer ( ) const
```

Getter function to get the minimum number of player.

Returns

minimum number of player

9.45.2.8 GetNextPlayer()

```
Brood::Application::Components::Player * PlayerManager::GetNextPlayer ( )
```

Getter function to get next player in the list.

Returns

next player who is supposed to be players if list is not empty ; nullptr if the list is empty

9.45.2.9 GetPlayerAtCurrlIdx()

```
Brood::Application::Components::Player * PlayerManager::GetPlayerAtCurrlIdx ( )
```

Getter function to get the current active player from the list.

Returns

currentActive player who is supposed to be players if list is not empty ; nullptr if the list is empty

9.45.2.10 GetPlayerStartPath()

```
const Brood::Application::Components::Path * PlayerManager::GetPlayerStartPath ( ) const
```

Getter function to get the start row of player.

Returns

start row for the players

9.45.2.11 InitializePlayerManager() [1/2]

```
void PlayerManager::InitializePlayerManager (
    Brood::Application::Components::Path * a_pathPtr )
```

Initializes the player manager.

Parameters

<i>a_pathPtr</i>	a iterator that holds to Brood::Application::Components::st_path node which contains the start row and column number inside it
------------------	--

9.45.2.12 InitializePlayerManager() [2/2]

```
void PlayerManager::InitializePlayerManager (
    Brood::Application::Data::ST_PlayerManagerData & a_playerManagerData,
    Brood::Application::Components::Path * a_playerCurrPath )
```

initializes the player manager with the passed data

Parameters

<i>a_deckData</i>	reference of the player manager data
<i>a_playerCurrPath</i>	path that the new player starts on

9.45.2.13 LoadDataFromFile()

```
void PlayerManager::LoadDataFromFile (
    Brood::Application::FileAccess * a_fileAccessPtr,
    Brood::Application::Components::Board * a_gameBoard )
```

loads the [PlayerManager](#) and its path data from passed file

Parameters

<i>a_fileAccessPtr</i>	pointer to a file Access object
------------------------	---------------------------------

9.45.2.14 operator=()

```
Brood::Application::Components::PlayerManager & PlayerManager::operator= (
    const Brood::Application::Components::PlayerManager & a_otherPlayerManager )
```

assignment operator

Parameters

<i>a_otherPlayerManager</i>	reference to the the other playermanager object that is being copied from
-----------------------------	---

9.45.2.15 ReplacePlayerAt()

```
void PlayerManager::ReplacePlayerAt (
    unsigned a_index,
    Brood::Application::Components::Player * a_playerPtr = nullptr )
```

replaces the a new player at passed index with a copy of passed index

Note

if the passed index does not exist then it does nothing to increase the index, increase the maximum number of player by

Parameters

<i>a_index</i>	index at which he player is to be replaced
<i>a_playerPtr</i>	Pointer to the player whose data is used to make a copy

9.45.2.16 SaveDataToFile()

```
void PlayerManager::SaveDataToFile (
    Brood::Application::FileAccess * a_fileAccessPtr )
```

loads the [PlayerManager](#) data from passed file

Parameters

<code>a_fileAccessPtr</code>	pointer to a file Access object
------------------------------	---------------------------------

9.45.2.17 SetCurrActivePlayerIdx()

```
void PlayerManager::SetCurrActivePlayerIdx (
    unsigned a_currActivePlayerIdx )
```

Setter function to set the current player Index.

Parameters

<code>a_currActivePlayerIdx</code>	minimum number of player
------------------------------------	--------------------------

9.45.2.18 SetMaxPlayer()

```
void PlayerManager::SetMaxPlayer (
    unsigned a_maxPlayer )
```

Setter function to set the maximum number of player.

Parameters

<code>a_maxPlayer</code>	maximum number of player
--------------------------	--------------------------

9.45.2.19 SetMinPlayer()

```
void PlayerManager::SetMinPlayer (
    unsigned a_minPlayer )
```

Setter function to set the minimum number of player.

Parameters

<code>a_minPlayer</code>	minimum number of player
--------------------------	--------------------------

9.45.2.20 SetPathForPlayerAtCurrlIdx()

```
void PlayerManager::SetPathForPlayerAtCurrlIdx (
    Brood::Application::Components::Path * a_pathPtr )
```

Setter function to set the path for current active player.

Parameters

<code>a_pathPtr</code>	pointer to the path the current active player is currently on
------------------------	---

9.45.2.21 SetPlayerStartPath()

```
void PlayerManager::SetPlayerStartPath (
    Brood::Application::Components::Path * a_startPathPtr )
```

Setter function to set the start path for new additional player.

Parameters

<code>a_startPathPtr</code>	pointer to the start path for the player
-----------------------------	--

9.45.3 Member Data Documentation**9.45.3.1 m_allPlayers**

```
std::vector<Brood::Application::Components::Player*> Brood::Application::Components::PlayerManager::m_allPlayers [private]
```

9.45.3.2 m_currActivePlayerIdx

```
unsigned Brood::Application::Components::PlayerManager::m_currActivePlayerIdx [private]
```

9.45.3.3 m_maxPlayer

```
unsigned Brood::Application::Components::PlayerManager::m_maxPlayer [private]
```

9.45.3.4 m_minPlayer

```
unsigned Brood::Application::Components::PlayerManager::m_minPlayer [private]
```

9.45.3.5 m_startPathPtr

```
Brood::Application::Components::Path* Brood::Application::Components::PlayerManager::m_start←  
PathPtr [private]
```

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[PlayerManager.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[PlayerManager.cpp](#)

9.46 PlayerManager Class Reference

A [PlayerManager](#) Class manages all player realted things.

```
#include "DeckManager.h"
```

9.46.1 Detailed Description

A [PlayerManager](#) Class manages all player realted things.

m_minPlayer

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckManager.h](#)

9.47 Brood::SpriteHandler Class Reference

```
#include <SpriteHandler.h>
```

Public Member Functions

- `SpriteHandler (sf::RectangleShape *a_body)`
`Default Constructor.`
- `virtual ~SpriteHandler ()`
`default destructor`
- `SpriteHandler (const SpriteHandler &a_otherSpriteHandler)`
`Copy Constructor.`
- `SpriteHandler & operator= (const SpriteHandler &a_otherSpriteHandler)`
`assignment operator`
- `void SetSpriteLength (float a_spriteLength)`
`setter function to set the sprite length`
- `void SetSpriteHeight (float a_spriteHeight)`
`setter function to set the sprite height`
- `bool SetTextureFromFilePath (std::string a_texturePath)`
`loads and sets the texture from the file path`
- `bool SetTextureFromFilePath (std::string a_textureDirectoryPath, std::string a_textureFileName)`
`loads and sets the texture from the file path`
- `bool SetTextureFromSavedFilePath ()`
`load and sets a texture from the stored texturePath`
- `void SetSpriteFromTexture (unsigned a_num)`
`Setter for texture.`
- `const std::string GetTextureFileName ()`
`Getter function to get the saved texture path.`
- `const std::string GetTextureDirectoryPath ()`
`Getter function to get the saved texture path.`
- `const float GetSpriteLength ()`
`Getter function to get the saved sprite length.`
- `const float GetSpriteHeight ()`
`Getter function to get the saved sprite height.`
- `const unsigned GetcurrSpriteIndex ()`
`Getter function to get the index of current Sprite.`
- `void RemoveTexture ()`
`Removes the texture.`
- `void Debugger ()`
`changes the texture of the element to its the element ID when debug is on and changes it back to the texture it had before debugging when debug is off.`

Private Attributes

- `sf::RectangleShape * m_body`
`pointer to the body`
- `std::string m_textureDirectoryPath`
`saves the directory path to the texture`
- `std::string m_textureFileName`
`saves the path to the texture`
- `sf::Texture m_texture`
`reference to the the texture opened from the path`
- `float m_spriteLength`
`sprite length in the texture file`
- `float m_spriteHeight`
`sprite height in the texture file`
- `unsigned m_currSpriteIndex`
`sprite height in the texture file`

9.47.1 Constructor & Destructor Documentation

9.47.1.1 SpriteHandler() [1/2]

```
SpriteHandler::SpriteHandler (
    sf::RectangleShape * a_body )
```

Default Constructor.

Initializes the [SpriteHandler](#) object

Parameters

<i>a_body</i>	pointer to the body of the element
---------------	------------------------------------

9.47.1.2 ~SpriteHandler()

```
SpriteHandler::~SpriteHandler ( ) [virtual]
```

default destructor

9.47.1.3 SpriteHandler() [2/2]

```
SpriteHandler::SpriteHandler (
    const SpriteHandler & a_otherSpriteHandler )
```

Copy Constructor.

Parameters

<i>a_otherSpriteHandler</i>	reference to the other sprite handler object that is being copied from
-----------------------------	--

9.47.2 Member Function Documentation

9.47.2.1 Debugger()

```
void SpriteHandler::Debugger ( )
```

changes the texture of the element to its the element ID when debug is on and changes it back to the texture it had before debugging when debug is off.

9.47.2.2 GetcurrSpriteIndex()

```
const unsigned SpriteHandler::GetcurrSpriteIndex ( )
```

Getter function to get the index of current Sprite.

Returns

float value containing the sprite length

9.47.2.3 GetSpriteHeight()

```
const float SpriteHandler::GetSpriteHeight ( )
```

Getter function to get the saved sprite height.

Returns

float value containing the sprite length

9.47.2.4 GetSpriteLength()

```
const float SpriteHandler::GetSpriteLength ( )
```

Getter function to get the saved sprite length.

Returns

float value containing the sprite length

9.47.2.5 GetTextureDirectoryPath()

```
const std::string SpriteHandler::GetTextureDirectoryPath ( )
```

Getter function to get the saved texture path.

Returns

string containing the texture directory path

9.47.2.6 GetTextureFileName()

```
const std::string SpriteHandler::GetTextureFileName ( )
```

Getter function to get the saved texture path.

Returns

string containing the texture directory path

9.47.2.7 operator=()

```
Brood::SpriteHandler & SpriteHandler::operator= (
    const SpriteHandler & a_otherSpriteHandler )
```

assignment operator

Parameters

<code>a_otherSpriteHandler</code>	reference to the the other sprite handler object that is being copied from
-----------------------------------	--

9.47.2.8 RemoveTexture()

```
void SpriteHandler::RemoveTexture ( )
```

Removes the texture.

9.47.2.9 SetSpriteFromTexture()

```
void SpriteHandler::SetSpriteFromTexture (
    unsigned a_num )
```

Setter for texture.

sets the texture from the texture vector if it is not empty. If either the sprite height and length is equal to zero then, it sets them to body's height and length

Warning

Assumes that The sprite for the faces are lined linearly. That is assumes that there is only one row of sprite.

Todo make the adaptable for number of rows and column

Parameters

<i>a_num</i>	index of sprite in the texture
--------------	--------------------------------

9.47.2.10 SetSpriteHeight()

```
void SpriteHandler::SetSpriteHeight (
    float a_spriteHeight )
```

setter funciton to set the sprite height

Parameters

<i>a_spriteLength</i>	height of the sprite
-----------------------	----------------------

9.47.2.11 SetSpriteLength()

```
void SpriteHandler::SetSpriteLength (
    float a_spriteLength )
```

setter funciton to set the sprite length

Parameters

<i>a_spriteLength</i>	length of the sprite
-----------------------	----------------------

9.47.2.12 SetTextureFromFilePath() [1/2]

```
bool SpriteHandler::SetTextureFromFilePath (
    std::string a_textureDirectoryPath,
    std::string a_textureFileName )
```

loads and sets the texture from the file path

It also saves the textureFilepath. By default uses the body size as the sprite size

Parameters

<i>a_textureDirectoryPath</i>	file path to the texture
<i>a_textureFileName</i>	file name of the texture

Returns

returns true if texture was successfully set; else false

9.47.2.13 SetTextureFromFilePath() [2/2]

```
bool SpriteHandler::SetTextureFromFilePath (
    std::string a_texturePath )
```

loads and sets the texture from the file path

It also saves the textureFilepath. By default uses the body size as the sprite size

Parameters

<i>a_texturePath</i>	file path to to the texture
----------------------	-----------------------------

Returns

returns true if texture was successfully set; else false

9.47.2.14 SetTextureFromSavedFilePath()

```
bool SpriteHandler::SetTextureFromSavedFilePath ( )
```

load and sets a texture from the stored texturePath

Returns

returns true if texture was successfully set; else false

9.47.3 Member Data Documentation**9.47.3.1 m_body**

```
sf::RectangleShape* Brood::SpriteHandler::m_body [private]
```

pointer to the body

9.47.3.2 m_currSpriteIndex

```
unsigned Brood::SpriteHandler::m_currSpriteIndex [private]
```

sprite height in the texture file

saves the current index of sprite in the texture

9.47.3.3 m_spriteHeight

```
float Brood::SpriteHandler::m_spriteHeight [private]
```

9.47.3.4 m_spriteLength

```
float Brood::SpriteHandler::m_spriteLength [private]
```

sprite length in the texture file

9.47.3.5 m_texture

```
sf::Texture Brood::SpriteHandler::m_texture [private]
```

reference to the the texture opened from the path

9.47.3.6 m_textureDirectoryPath

```
std::string Brood::SpriteHandler::m_textureDirectoryPath [private]
```

saves the directory path to the texture

9.47.3.7 m_textureFileName

```
std::string Brood::SpriteHandler::m_textureFileName [private]
```

saves the path to the texture

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[SpriteHandler.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[SpriteHandler.cpp](#)

9.48 SpriteHandler Class Reference

a class to handel the sprite

```
#include "SpriteHandler.h"
```

9.48.1 Detailed Description

a class to handel the sprite

This class has function to manupulate the sprite and textures. It allows to load texture from file.

Note

A sprite sheet should always be horizontal

```
// creating a button
Brood::BroodUI::Button mySpriteButton;
mySpriteButton.SetBodySize( 50, 50 );
mySpriteButton.SetBodyPosition( 55, 100 );
mySpriteButtonSetFont( font );
// loading the texture
mySpriteButton.GetSpriteBody().SetTextureFromFilePath( fileName );
// setting the second sprite from the texture file
mySpriteButton.GetSpriteBody().SetSpriteFromTexture( 1 );
```

Todo make this animatable. next Sprit() which get the next sprite and loops back to idx 0

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[SpriteHandler.h](#)

9.49 ST_ApplicationData Struct Reference

contains all the general data about the game

```
#include <Data.h>
```

9.49.1 Detailed Description

contains all the general data about the game

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.50 Brood::Application::Data::ST_BoardData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- float [stm_boardSizeX](#) = 0
length of board
- float [stm_boardSizeY](#) = 0
height of board
- float [stm_boardPosX](#) = 0
x-cordinate of board's upper left corner
- float [stm_boardPosY](#) = 0
y-cordinate of board's upper left corner
- unsigned [stm_numRow](#) = 1
number of tile Row in the board
- unsigned [stm_numCol](#) = 1
number of tile column in the board
- unsigned [stm_currActiveNumRow](#) = 0
tile row number of the active tile
- unsigned [stm_currActiveNumCol](#) = 0
tile column number of the active tile

9.50.1 Member Function Documentation

9.50.1.1 GetString()

```
const std::string ST_BoardData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.50.1.2 PopulateFromString()

```
void ST_BoardData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

a_string	string containing all of the data seperated by ''
----------	---

9.50.2 Member Data Documentation

9.50.2.1 stm_boardPosX

```
float Brood::Application::Data::ST_BoardData::stm_boardPosX = 0
```

x-cordinate of board's upper left corner

9.50.2.2 stm_boardPosY

```
float Brood::Application::Data::ST_BoardData::stm_boardPosY = 0
```

y-cordinate of board's upper left corner

9.50.2.3 stm_boardSizeX

```
float Brood::Application::Data::ST_BoardData::stm_boardSizeX = 0
```

length of board

9.50.2.4 **stm_boardSizeY**

```
float Brood::Application::Data::ST_BoardData::stm_boardSizeY = 0
```

height of board

9.50.2.5 **stm_currActiveNumCol**

```
unsigned Brood::Application::Data::ST_BoardData::stm_currActiveNumCol = 0
```

tile column number of the active tile

9.50.2.6 **stm_currActiveNumRow**

```
unsigned Brood::Application::Data::ST_BoardData::stm_currActiveNumRow = 0
```

tile row number of the active tile

9.50.2.7 **stm_numCol**

```
unsigned Brood::Application::Data::ST_BoardData::stm_numCol = 1
```

number of tile column in the board

9.50.2.8 **stm_numRow**

```
unsigned Brood::Application::Data::ST_BoardData::stm_numRow = 1
```

number of tile Row in the board

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.51 ST_BoardData Struct Reference

contains data about the board

```
#include <Data.h>
```

9.51.1 Detailed Description

contains data about the board

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.52 St_CardInfo Class Reference

A class to store the information that is to be displayed in the card.

```
#include "CardInfo.h"
```

9.52.1 Detailed Description

A class to store the information that is to be displayed in the card.

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[CardInfo.h](#)

9.53 Brood::Application::Data::ST_CardInfoPrefabData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- unsigned [stm_time](#) = 0
number of second that the user has to answer the question
- unsigned [stm_correctNumSteps](#) = 0
number of steps the user takes when their answer is correct
- unsigned [stm_incorrectNumSteps](#) = 0
number of steps the user takes when their answer is incorrect
- std::string [stm_question](#) = ""
question in the card
- std::string [stm_correctAnswer](#) = ""
correct answer to the quesiton

9.53.1 Member Function Documentation

9.53.1.1 GetString()

```
const std::string Brood::Application::Data::ST_CardInfoPrefabData::GetString ( ) const  
creates a string and appends each data to its back seperated by ''
```

Returns

string containg all the data appended to it seperated by ''

9.53.1.2 PopulateFromString()

```
void Brood::Application::Data::ST_CardInfoPrefabData::PopulateFromString (   
    std::string a_string )
```

populates the struct from passed string

Parameters

a_string	string containing all of the data seperated by ''
----------	---

9.53.2 Member Data Documentation

9.53.2.1 stm_correctAnswer

```
std::string Brood::Application::Data::ST_CardInfoPrefabData::stm_correctAnswer = ""  
correct answer to the quesiton
```

9.53.2.2 stm_correctNumSteps

```
unsigned Brood::Application::Data::ST_CardInfoPrefabData::stm_correctNumSteps = 0  
number of steps the user takes when their answer is correct
```

9.53.2.3 stm_incorrectNumSteps

```
unsigned Brood::Application::Data::ST_CardInfoPrefabData::stm_incorrectNumSteps = 0
```

number of steps the user takes when their answer is incorrect

9.53.2.4 stm_question

```
std::string Brood::Application::Data::ST_CardInfoPrefabData::stm_question = ""
```

question in the card

9.53.2.5 stm_time

```
unsigned Brood::Application::Data::ST_CardInfoPrefabData::stm_time = 0
```

number of second that the user has to answer the question

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.54 ST_CardPrefabData Struct Reference

data in a card and other use full data

```
#include <Data.h>
```

9.54.1 Detailed Description

data in a card and other use full data

card data is loaded into this from a file / stored into a file

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.55 Brood::Application::StaticVariables::ST_ColorVariables Struct Reference

```
#include <GlobalVariables.h>
```

Static Public Member Functions

- static sf::Color [GetRandomColor \(\)](#)
generates random color

Static Public Attributes

- static sf::Color [stm_Black](#) = sf::Color::Black
sf::Color::Black
- static sf::Color [stm_White](#) = sf::Color::White
sf::Color::White
- static sf::Color [stm_HotOverlay](#) = sf::Color(186, 189, 201, 50)
Lavender Gray with alpha of 50.
- static sf::Color [stm_CurrActiveOverlay](#) = sf::Color(36, 38, 46, 200)
Raisin Black with alpha of 200.
- static sf::Color [stm_MainMenu](#) = sf::Color(64, 73, 79, 255)
Charcoal.
- static sf::Color [stm_AppPrimaryColor](#) = sf::Color(73, 85, 123, 255)
Purple Navy.
- static sf::Color [stm_AppSecondaryColor](#) = sf::Color(99, 123, 168, 255)
Pastel Pink.
- static sf::Color [stm_ErrorColor](#) = sf::Color(191, 33, 30, 255)
Venetian Red.

9.55.1 Member Function Documentation

9.55.1.1 [GetRandomColor\(\)](#)

```
sf::Color ST_ColorVariables::GetRandomColor ( ) [static]
```

generates random color

Generates a random color.

9.55.2 Member Data Documentation

9.55.2.1 [stm_AppPrimaryColor](#)

```
sf::Color ST_ColorVariables::stm_AppPrimaryColor = sf::Color( 73, 85, 123, 255 ) [static]
```

Purple Navy.

9.55.2.2 **stm_AppSecondaryColor**

```
sf::Color ST_ColorVariables::stm_AppSecondaryColor = sf::Color( 99, 123, 168, 255 ) [static]
```

Pastel Pink.

9.55.2.3 **stm_Black**

```
sf::Color ST_ColorVariables::stm_Black = sf::Color::Black [static]
```

sf::Color::Black

9.55.2.4 **stm_CurrActiveOverlay**

```
sf::Color ST_ColorVariables::stm_CurrActiveOverlay = sf::Color( 36, 38, 46, 200 ) [static]
```

Raisin Black with alpha of 200.

9.55.2.5 **stm_ErrorColor**

```
sf::Color ST_ColorVariables::stm_ErrorColor = sf::Color( 191, 33, 30, 255 ) [static]
```

Venetian Red.

9.55.2.6 **stm_HotOverlay**

```
sf::Color ST_ColorVariables::stm_HotOverlay = sf::Color( 186, 189, 201, 50 ) [static]
```

Lavender Gray with alpha of 50.

9.55.2.7 **stm_MainMenu**

```
sf::Color ST_ColorVariables::stm_MainMenu = sf::Color( 64, 73, 79, 255 ) [static]
```

Charcoal.

9.55.2.8 stm_White

```
sf::Color ST_ColorVariables::stm_White = sf::Color::White [static]
sf::Color::White
```

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.cpp](#)

9.56 ST_ColorVariables Struct Reference

static struct for all the colors used in the applicaiton

```
#include "GlobalVariables.h"
```

9.56.1 Detailed Description

static struct for all the colors used in the applicaiton

static struct for all path to folder

static struct for all the global Core Variables

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.h](#)

9.57 Brood::Application::Data::ST_DeckManagerData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- unsigned [stm_movementType](#) = 0
- bool [stm_incorrectPenalty](#) = false
incorrect penalty for the game
- unsigned [stm_numDecks](#) = 1
total number of deck
- unsigned [stm_currDecksIdx](#) = 0
current deck idx

9.57.1 Member Function Documentation

9.57.1.1 GetString()

```
const std::string ST_DeckManagerData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.57.1.2 PopulateFromString()

```
void ST_DeckManagerData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

a_string	string containing all of the data seperated by ''
----------	---

9.57.2 Member Data Documentation

9.57.2.1 stm_currDecksIdx

```
unsigned Brood::Application::Data::ST_DeckManagerData::stm_currDecksIdx = 0
```

current deck idx

9.57.2.2 stm_incorrectPenalty

```
bool Brood::Application::Data::ST_DeckManagerData::stm_incorrectPenalty = false
```

incorrect penalty for the game

9.57.2.3 `stm_movementType`

```
unsigned Brood::Application::Data::ST_DeckManagerData::stm_movementType = 0
```

movementType of the deck; [Brood::Application::Components::ENUM_MovementType](#) stored as unsigned

9.57.2.4 `stm_numDecks`

```
unsigned Brood::Application::Data::ST_DeckManagerData::stm_numDecks = 1
```

total number of deck

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.58 `ST_DeckManagerData` Struct Reference

contains deck manager data for one deck prefab

```
#include <Data.h>
```

9.58.1 Detailed Description

contains deck manager data for one deck prefab

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.59 `Brood::Application::Data::ST_DeckPrefabData` Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- float `stm_deckSizeX` = 0
length of deck
- float `stm_deckSizeY` = 0
height of deck
- float `stm_deckPosX` = 0
x-coordinate of deck's upper left corner
- float `stm_deckPosY` = 0
y-coordinate of deck's upper left corner
- std::string `stm_textureFileName` = ""
filename for the texture
- std::string `stm_cardInitFilename` = ""
filename for the card init
- unsigned `stm_numTotalCard` = 1
number of card in this deck
- unsigned `stm_undealtCardIdx` = 0
idx of undealt card in this deck
- unsigned `stm_currActiveCardIdx` = 0
idx of current active card in this deck

9.59.1 Member Function Documentation

9.59.1.1 GetString()

```
const std::string ST_DeckPrefabData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.59.1.2 PopulateFromString()

```
void ST_DeckPrefabData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

<code>a_string</code>	string containing all of the data seperated by ''
-----------------------	---

9.59.2 Member Data Documentation

9.59.2.1 **stm_cardInitFilename**

```
std::string Brood::Application::Data::ST_DeckPrefabData::stm_cardInitFilename = ""  
  
filename for the card init
```

9.59.2.2 **stm_currActiveCardIdx**

```
unsigned Brood::Application::Data::ST_DeckPrefabData::stm_currActiveCardIdx = 0  
  
idx of current active card in this deck
```

9.59.2.3 **stm_deckPosX**

```
float Brood::Application::Data::ST_DeckPrefabData::stm_deckPosX = 0  
  
x-coordinate of deck's upper left corner
```

9.59.2.4 **stm_deckPosY**

```
float Brood::Application::Data::ST_DeckPrefabData::stm_deckPosY = 0  
  
y-coordinate of deck's upper left corner
```

9.59.2.5 **stm_deckSizeX**

```
float Brood::Application::Data::ST_DeckPrefabData::stm_deckSizeX = 0  
  
length of deck
```

9.59.2.6 stm_deckSizeY

```
float Brood::Application::Data::ST_DeckPrefabData::stm_deckSizeY = 0
```

height of deck

9.59.2.7 stm_numTotalCard

```
unsigned Brood::Application::Data::ST_DeckPrefabData::stm_numTotalCard = 1
```

number of card in this deck

9.59.2.8 stm_textureFileName

```
std::string Brood::Application::Data::ST_DeckPrefabData::stm_textureFileName = ""
```

filename for the texture

9.59.2.9 stm_undealtCardIdx

```
unsigned Brood::Application::Data::ST_DeckPrefabData::stm_undealtCardIdx = 0
```

idx of undealt card in this deck

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.60 ST_DeckPrefabData Struct Reference

contains deck data for one deck prefab

```
#include <Data.h>
```

9.60.1 Detailed Description

contains deck data for one deck prefab

Decks are unique in the released game. As the position of where they are in in the screen will be different

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.61 Brood::Application::Data::ST_DicePrefabData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- unsigned [stm_numSides](#) = 6
number of side in the dice
- float [stm_diceSizeX](#) = 0
length of dice
- float [stm_diceSizeY](#) = 0
height of dice
- float [stm_dicePosX](#) = 0
x-coordinate of dice's upper left corner
- float [stm_dicePosY](#) = 0
y-coordinate of dice's upper left corner
- std::string [stm_textureFileName](#) = ""
filename for the texture

9.61.1 Member Function Documentation

9.61.1.1 [GetString\(\)](#)

```
const std::string ST_DicePrefabData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.61.1.2 [PopulateFromString\(\)](#)

```
void ST_DicePrefabData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

<code>a_string</code>	string containing all of the data seperated by ''
-----------------------	---

9.61.2 Member Data Documentation

9.61.2.1 `stm_dicePosX`

```
float Brood::Application::Data::ST_DicePrefabData::stm_dicePosX = 0
```

x-cordinate of dice's upper left corner

9.61.2.2 `stm_dicePosY`

```
float Brood::Application::Data::ST_DicePrefabData::stm_dicePosY = 0
```

y-cordinate of dice's upper left corner

9.61.2.3 `stm_diceSizeX`

```
float Brood::Application::Data::ST_DicePrefabData::stm_diceSizeX = 0
```

length of dice

9.61.2.4 `stm_diceSizeY`

```
float Brood::Application::Data::ST_DicePrefabData::stm_diceSizeY = 0
```

height of dice

9.61.2.5 `stm_numSides`

```
unsigned Brood::Application::Data::ST_DicePrefabData::stm_numSides = 6
```

number of side in the dice

9.61.2.6 `stm_textureFileName`

```
std::string Brood::Application::Data::ST_DicePrefabData::stm_textureFileName = ""
```

filename for the texture

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.62 `ST_DicePrefabData` Struct Reference

contains dice prefab data

```
#include <Data.h>
```

9.62.1 Detailed Description

contains dice prefab data

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.63 `Brood::Application::Data::ST_DisplayCardData` Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- float `stm_displayCardSizeX` = 0
length of display card
- float `stm_displayCardSizeY` = 0
width of display card
- float `stm_displayCardPositionX` = 0
x-cordinate of display card's upper left corner
- float `stm_displayCardPositionY` = 0
y-cordinate of display card's upper left corner
- std::string `stm_displayCardFrontTextureFilename` = ""
filename for the front card texture
- std::string `stm_displayCardBackTextureFilename` = ""
filename for the back card texture
- float `stm_frontTimePromptSizeX` = 0
length of front Time Prompt
- float `stm_frontTimePromptSizeY` = 0
width of front Time Prompt
- float `stm_frontTimePromptPositionX` = 0
x-cordinate of front Time Prompt's upper left corner
- float `stm_frontTimePromptPositionY` = 0
y-cordinate of front Time Prompt's upper left corner
- float `stm_frontTimeValueSizeX` = 0
length of front time value
- float `stm_frontTimeValueSizeY` = 0
width of front time value
- float `stm_frontTimeValuePositionX` = 0
y-cordinate of front time value's upper left corner
- float `stm_frontTimeValuePositionY` = 0
x-cordinate of front time value's upper left corner
- float `stm_upPromptSizeX` = 0
length of up prompt
- float `stm_upPromptSizeY` = 0
width of up prompt
- float `stm_upPromptPositionX` = 0
x-cordinate of up prompt's upper left corner
- float `stm_upPromptPositionY` = 0
y-cordinate of up prompt's upper left corner
- float `stm_upValueSizeX` = 0
length of up value
- float `stm_upValueSizeY` = 0
width of up value
- float `stm_upValuePositionX` = 0
x-cordinate of up value's upper left corner
- float `stm_upValuePositionY` = 0
y-cordinate of up value's upper left corner
- float `stm_downPromptSizeX` = 0
length of down prompt
- float `stm_downPromptSizeY` = 0
width of down prompt
- float `stm_downPromptPositionX` = 0

- float `stm_downPromptPositonY` = 0
y-cordinate of down prompt's upper left corner
- float `stm_downValueSizeX` = 0
length of down value
- float `stm_downValueSizeY` = 0
width of down value
- float `stm_downValuePositionX` = 0
x-cordinate of down value's upper left corner
- float `stm_downValuePositionY` = 0
y-cordinate of down value's upper left corner
- float `stm_turnCardPromptSizeX` = 0
length of turn card prompt
- float `stm_turnCardPromptSizeY` = 0
width of turn card prompt
- float `stm_turnCardPromptPositionX` = 0
x-cordinate of turn card prompt's upper left corner
- float `stm_turnCardPromptPositionY` = 0
y-cordinate of turn card prompt's upper left corner
- float `stm_backTimePromptSizeX` = 0
length of back time prompt
- float `stm_backTimePromptSizeY` = 0
width of back time prompt
- float `stm_backTimePromptPositionX` = 0
x-cordinate of back time prompt's upper left corner
- float `stm_backTimePromptPositionY` = 0
y-cordinate of back time prompt's upper left corner
- float `stm_backTimeValueSizeX` = 0
length of back time value
- float `stm_backTimeValueSizeY` = 0
width of back time value
- float `stm_backTimeValuePositionX` = 0
x-cordinate of back time value's upper left corner
- float `stm_backTimeValuePositionY` = 0
y-cordinate of back time value's upper left corner
- float `stm_questionPromptSizeX` = 0
length of question prompt
- float `stm_questionPromptSizeY` = 0
width of question prompt
- float `stm_questionPromptPositionX` = 0
x-cordinate of question prompt's upper left corner
- float `stm_questionPromptPositionY` = 0
y-cordinate of question prompt's upper left corner
- float `stm_questionValueSizeX` = 0
length of question value
- float `stm_questionValueSizeY` = 0
width of question value
- float `stm_questionValuePositionX` = 0
x-cordinate of question value's upper left corner
- float `stm_questionValuePositionY` = 0
y-cordinate of question value's upper left corner

- float `stm_answerPromptSizeX` = 0
length of answer prompt
- float `stm_answerPromptSizeY` = 0
width of answer prompt
- float `stm_answerPromptPositionX` = 0
x-cordinate of answer prompt's upper left corner
- float `stm_answerPromptPositionY` = 0
y-cordinate of answer prompt's upper left corner
- float `stm_answerValueSizeX` = 0
length of answer value
- float `stm_answerValueSizeY` = 0
width of answer value
- float `stm_answerValuePositionX` = 0
x-cordinate of answer value's upper left corner
- float `stm_answerValuePositionY` = 0
y-cordinate of answer value's upper left corner
- float `stm_submitButtonValueSizeX` = 0
length of submit button value
- float `stm_submitButtonValueSizeY` = 0
width of submit button value
- float `stm_submitButtonValuePositionX` = 0
x-cordinate of submit button value's upper left corner
- float `stm_submitButtonValuePositionY` = 0
y-cordinate of submit button value's upper left corner

9.63.1 Member Function Documentation

9.63.1.1 GetString()

```
const std::string ST_DisplayCardData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.63.1.2 PopulateFromString()

```
void ST_DisplayCardData::PopulateFromString (   
    std::string a_string )
```

populates the struct from passed string

Parameters

<i>a_string</i>	string containing all of the data seperated by ''
-----------------	---

9.63.2 Member Data Documentation

9.63.2.1 **stm_answerPromptPositionX**

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerPromptPositionX = 0
```

x-coordinate of answer prompt's upper left corner

9.63.2.2 **stm_answerPromptPositionY**

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerPromptPositionY = 0
```

y-coordinate of answer prompt's upper left corner

9.63.2.3 **stm_answerPromptSizeX**

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerPromptSizeX = 0
```

length of answer prompt

9.63.2.4 **stm_answerPromptSizeY**

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerPromptSizeY = 0
```

width of answer prompt

9.63.2.5 **stm_answerValuePositionX**

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerValuePositionX = 0
```

x-coordinate of answer value's upper left corner

9.63.2.6 stm_answerValuePositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerValuePositonY = 0
```

y-coordinate of answer value's upper left corner

9.63.2.7 stm_answerValueSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerValueSizeX = 0
```

length of answer value

9.63.2.8 stm_answerValueSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_answerValueSizeY = 0
```

width of answer value

9.63.2.9 stm_backTimePromptPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimePromptPositionX = 0
```

x-coordinate of back time prompt's upper left corner

9.63.2.10 stm_backTimePromptPositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimePromptPositonY = 0
```

y-coordinate of back time prompt's upper left corner

9.63.2.11 stm_backTimePromptSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimePromptSizeX = 0
```

length of back time prompt

9.63.2.12 `stm_backTimePromptSizeY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimePromptSizeY = 0
```

width of back time prompt

9.63.2.13 `stm_backTimeValuePositionX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimeValuePositionX = 0
```

x-coordinate of back time value's upper left corner

9.63.2.14 `stm_backTimeValuePositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimeValuePositonY = 0
```

y-coordinate of back time value's upper left corner

9.63.2.15 `stm_backTimeValueSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimeValueSizeX = 0
```

length of back time value

9.63.2.16 `stm_backTimeValueSizeY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_backTimeValueSizeY = 0
```

width of back time value

9.63.2.17 `stm_displayCardBackTextureFilename`

```
std::string Brood::Application::Data::ST_DisplayCardData::stm_displayCardBackTextureFilename =
""
```

filename for the back card texture

9.63.2.18 stm_displayCardFrontTextureFilename

```
std::string Brood::Application::Data::ST_DisplayCardData::stm_displayCardFrontTextureFilename  
= ""
```

filename for the front card texture

9.63.2.19 stm_displayCardPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_displayCardPositionX = 0
```

x-coordinate of display card's upper left corner

9.63.2.20 stm_displayCardPositionY

```
float Brood::Application::Data::ST_DisplayCardData::stm_displayCardPositionY = 0
```

y-coordinate of display card's upper left corner

9.63.2.21 stm_displayCardSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_displayCardSizeX = 0
```

length of display card

9.63.2.22 stm_displayCardSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_displayCardSizeY = 0
```

width of display card

9.63.2.23 stm_downPromptPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_downPromptPositionX = 0
```

x-coordinate of down prompt's upper left corner

9.63.2.24 `stm_downPromptPositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downPromptPositonY = 0
```

y-coordinate of down prompt's upper left corner

9.63.2.25 `stm_downPromptSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downPromptSizeX = 0
```

length of down prompt

9.63.2.26 `stm_downPromptSizeY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downPromptSizeY = 0
```

width of down prompt

9.63.2.27 `stm_downValuePositionX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downValuePositionX = 0
```

x-coordinate of down value's upper left corner

9.63.2.28 `stm_downValuePositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downValuePositonY = 0
```

y-coordinate of down value's upper left corner

9.63.2.29 `stm_downValueSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_downValueSizeX = 0
```

length of down value

9.63.2.30 stm_downValueSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_downValueSizeY = 0
```

width of down value

9.63.2.31 stm_frontTimePromptPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimePromptPositionX = 0
```

x-coordinate of front Time Prompt's upper left corner

9.63.2.32 stm_frontTimePromptPositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimePromptPositonY = 0
```

y-coordinate of front Time Prompt's upper left corner

9.63.2.33 stm_frontTimePromptSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimePromptSizeX = 0
```

length of front Time Prompt

9.63.2.34 stm_frontTimePromptSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimePromptSizeY = 0
```

width of front Time Prompt

9.63.2.35 stm_frontTimeValuePositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimeValuePositionX = 0
```

y-coordinate of front time value's upper left corner

9.63.2.36 stm_frontTimeValuePositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimeValuePositonY = 0
```

x-coordinate of front time value's upper left corner

9.63.2.37 stm_frontTimeValueSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimeValueSizeX = 0
```

length of front time value

9.63.2.38 stm_frontTimeValueSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_frontTimeValueSizeY = 0
```

width of front time value

9.63.2.39 stm_questionPromptPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionPromptPositionX = 0
```

x-coordinate of question prompt's upper left corner

9.63.2.40 stm_questionPromptPositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionPromptPositonY = 0
```

y-coordinate of question prompt's upper left corner

9.63.2.41 stm_questionPromptSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionPromptSizeX = 0
```

length of question prompt

9.63.2.42 stm_questionPromptSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionPromptSizeY = 0
```

width of question prompt

9.63.2.43 stm_questionValuePositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionValuePositionX = 0
```

x-coordinate of question value's upper left corner

9.63.2.44 stm_questionValuePositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionValuePositonY = 0
```

y-coordinate of question value's upper left corner

9.63.2.45 stm_questionValueSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionValueSizeX = 0
```

length of question value

9.63.2.46 stm_questionValueSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_questionValueSizeY = 0
```

width of question value

9.63.2.47 stm_submitButtonValuePositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_submitButtonValuePositionX = 0
```

x-coordinate of submit button value's upper left corner

9.63.2.48 `stm_submitButtonValuePositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_submitButtonValuePositonY = 0
```

y-coordinate of submit button value's upper left corner

9.63.2.49 `stm_submitButtonValueSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_submitButtonValueSizeX = 0
```

length of submit button value

9.63.2.50 `stm_submitButtonValueSizeY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_submitButtonValueSizeY = 0
```

width of submit button value

9.63.2.51 `stm_turnCardPromptPositionX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_turnCardPromptPositionX = 0
```

x-coordinate of turn card prompt's upper left corner

9.63.2.52 `stm_turnCardPromptPositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_turnCardPromptPositonY = 0
```

y-coordinate of turn card prompt's upper left corner

9.63.2.53 `stm_turnCardPromptSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_turnCardPromptSizeX = 0
```

length of turn card prompt

9.63.2.54 stm_turnCardPromptSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_turnCardPromptSizeY = 0
```

width of turn card prompt

9.63.2.55 stm_upPromptPositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_upPromptPositionX = 0
```

x-coordinate of up prompt's upper left corner

9.63.2.56 stm_upPromptPositonY

```
float Brood::Application::Data::ST_DisplayCardData::stm_upPromptPositonY = 0
```

y-coordinate of up prompt's upper left corner

9.63.2.57 stm_upPromptSizeX

```
float Brood::Application::Data::ST_DisplayCardData::stm_upPromptSizeX = 0
```

length of up prompt

9.63.2.58 stm_upPromptSizeY

```
float Brood::Application::Data::ST_DisplayCardData::stm_upPromptSizeY = 0
```

width of up prompt

9.63.2.59 stm_upValuePositionX

```
float Brood::Application::Data::ST_DisplayCardData::stm_upValuePositionX = 0
```

x-coordinate of up value's upper left corner

9.63.2.60 `stm_upValuePositonY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_upValuePositonY = 0
```

y-coordinate of up value's upper left corner

9.63.2.61 `stm_upValueSizeX`

```
float Brood::Application::Data::ST_DisplayCardData::stm_upValueSizeX = 0
```

length of up value

9.63.2.62 `stm_upValueSizeY`

```
float Brood::Application::Data::ST_DisplayCardData::stm_upValueSizeY = 0
```

width of up value

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.64 ST_DisplayCardData Struct Reference

contains display card data

```
#include <Data.h>
```

9.64.1 Detailed Description

contains display card data

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.65 Brood::Application::StaticVariables::ST_Folders Struct Reference

```
#include <GlobalVariables.h>
```

Static Public Attributes

- static std::filesystem::path `stm_cwd` = std::filesystem::current_path()
current working directory of the application
- static std::filesystem::path `stm_assets` = `Brood::Application::StaticVariables::ST_Folders::stm_cwd / std::filesystem::path{ "Assets" }`
path to assets folder. It is a sub folder of CWD
- static std::filesystem::path `stm_fonts` = `Brood::Application::StaticVariables::ST_Folders::stm_assets / std::filesystem::path{ "Fonts" }`
path to font folder. It is a sub folder of Assets folder
- static std::filesystem::path `stm_textures` = `Brood::Application::StaticVariables::ST_Folders::stm_assets / std::filesystem::path{ "Textures" }`
path to Texture folder. It is a sub folder of Assets folder
- static std::filesystem::path `stm_diceTextures` = `Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "DiceTexture" }`
path to diceTexture folder. It is a sub folder of Textures folder
- static std::filesystem::path `stm_playerTextures` = `Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "PlayerTexture" }`
path to playerTextures folder. It is a sub folder of Textures folder
- static std::filesystem::path `stm_tileTextures` = `Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "TileTexture" }`
path to tileTextures folder. It is a sub folder of Textures folder
- static std::filesystem::path `stm_deckTextures` = `Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "DeckTexture" }`
path to tileTextures folder. It is a sub folder of Textures folder
- static std::filesystem::path `stm_data` = `Brood::Application::StaticVariables::ST_Folders::stm_cwd / std::filesystem::path{ "Data" }`
path to data folder. It is a sub folder of CWD

9.65.1 Member Data Documentation

9.65.1.1 `stm_assets`

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_assets = Brood::Application::StaticVariables::ST_Folders::stm_cwd / std::filesystem::path{ "Assets" } [static]
```

path to assets folder. It is a sub folder of CWD

9.65.1.2 `stm_cardTextures`

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_cardTextures = Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "CardTexture" } [static]
```

path to tileTextures folder. It is a sub folder of Textures folder

9.65.1.3 **stm_cwd**

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_cwd = std::filesystem::current_path() [static]
```

current working directory of the applicaiton

9.65.1.4 **stm_data**

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_data = Brood::Application::StaticVariables::ST_Folders::stm_data / std::filesystem::path{ "Data" } [static]
```

path to data folder. It is a sub folder of CWD

9.65.1.5 **stm_deckTextures**

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_deckTextures = Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "DeckTexture" } [static]
```

path to tileTextures folder. It is a sub folder of Textures folder

9.65.1.6 **stm_diceTextures**

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_diceTextures = Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "DiceTexture" } [static]
```

path to diceTexture folder. It is a sub folder of Textures folder

9.65.1.7 **stm_fonts**

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_fonts = Brood::Application::StaticVariables::ST_Folders::stm_fonts / std::filesystem::path{ "Fonts" } [static]
```

path to font folder. It is a sub folder of Assets folder

9.65.1.8 stm_playerTextures

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_playerTextures =
Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "Player<-
Texture" } [static]
```

path to playerTextures folder. It is a sub folder of Textures folder

9.65.1.9 stm_textures

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_textures = Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "Textures" } [static]
```

path to Texture folder. It is a sub folder of Assets folder

9.65.1.10 stm_tileTextures

```
std::filesystem::path Brood::Application::StaticVariables::ST_Folders::stm_tileTextures =
Brood::Application::StaticVariables::ST_Folders::stm_textures / std::filesystem::path{ "Tile<-
Texture" } [static]
```

path to tileTextures folder. It is a sub folder of Textures folder

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.cpp](#)

9.66 Brood::Application::Data::ST_GameData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- std::string [stm_gameTitle](#) = ""
name of the game; std::string

9.66.1 Member Function Documentation

9.66.1.1 GetString()

```
const std::string Brood::Application::Data::ST_GameData::GetString ( ) const
creates a string and appends each data to its back seperated by ''
```

Returns

string containg all the data appended to it seperated by ''

9.66.1.2 PopulateFromString()

```
void Brood::Application::Data::ST_GameData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

<code>a_string</code>	string containing all of the data seperated by ''
-----------------------	---

9.66.2 Member Data Documentation

9.66.2.1 stm_gameTitle

```
std::string Brood::Application::Data::ST_GameData::stm_gameTitle = ""
name of the game; std::string
```

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.67 Brood::Application::StaticVariables::ST_GlobalCoreVariables Struct Reference

```
#include <GlobalVariables.h>
```

Static Public Attributes

- static uint32_t **stm_window_height** = 900
 - holds the application height*
- static uint32_t **stm_window_width** = 900
 - holds the application width*
- static sf::Font **stm_font**
 - stores the font that is used through out the applicaiton*
- static bool **stm_is_debug_mode** = false
 - holds the state of debugger; true if debug mode is on*
- static bool **stm_exit** = false
- static uint32_t **stm_panelPercentage** = 35
- static unsigned **stm_seed** = std::chrono::system_clock::now().time_since_epoch().count()
 - holds the seed for random value*

9.67.1 Member Data Documentation

9.67.1.1 **stm_exit**

```
bool Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_exit = false [static]
```

stores the state of the application; false if the application is running

9.67.1.2 **stm_font**

```
sf::Font Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_font [static]
```

stores the font that is used through out the applicaiton

9.67.1.3 **stm_is_debug_mode**

```
bool Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_is_debug_mode = false [static]
```

holds the state of debugger; true if debug mode is on

9.67.1.4 **stm_panelPercentage**

```
uint32_t Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_panelPercentage = 35 [static]
```

holds the percentage of size of the application side pannel compare to the applicaiton window

9.67.1.5 `stm_seed`

```
unsigned Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_seed = std::chrono::system_clock::now().time_since_epoch().count() [static]
```

holds the seed for random value

9.67.1.6 `stm_window_height`

```
uint32_t Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_window_height = 900 [static]
```

holds the application height

9.67.1.7 `stm_window_width`

```
uint32_t Brood::Application::StaticVariables::ST_GlobalCoreVariables::stm_window_width = 900 [static]
```

holds the application width

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[GlobalVariables.cpp](#)

9.68 `Brood::Application::Data::ST_PathPrefabData` Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void [PopulateFromString](#) (std::string a_string)
populates the struct from passed string
- const std::string [GetString](#) () const
creates a string and appends each data to its back seperated by '' by ''

Public Attributes

- `unsigned stm_tileType = 0`
tile type
- `unsigned stm_RowNum = 0`
tile row num
- `unsigned stm_ColNum = 0`
tile column num
- `unsigned stm_nextTileRowNum = 0`
next tile row num
- `unsigned stm_nextTileColNum = 0`
next tile column num
- `unsigned stm_endBridgeTileRowNum = 0`
end bridge tile row num
- `unsigned stm_endBridgeTileColNum = 0`
end bridge tile column num
- `unsigned stm_numberCardDraw = 0`
number of card to draw
- `unsigned stm_assignedDeckId = 0`
- `bool stm_forceDiceRoll = false`
if true rolls dice instead of drawing a card
- `std::string stm_textureFileName = ""`
filename for the texture; std::string

9.68.1 Member Function Documentation

9.68.1.1 GetString()

```
const std::string Brood::Application::Data::ST_PathPrefabData::GetString ( ) const
```

creates a string and appends each data to its back seperated by '' by ''

9.68.1.2 PopulateFromString()

```
void Brood::Application::Data::ST_PathPrefabData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

<code>a_string</code>	string containing all of the data seperated by ''
-----------------------	---

9.68.2 Member Data Documentation

9.68.2.1 **stm_assignedDeckId**

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_assignedDeckId = 0
```

deck index that this tile is assigned to. when player lands on this tile they will draw a draw card from it.

9.68.2.2 **stm_ColNum**

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_ColNum = 0
```

tile column num

9.68.2.3 **stm_endBridgeTileColNum**

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_endBridgeTileColNum = 0
```

end bridge tile column num

9.68.2.4 **stm_endBridgeTileRowNum**

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_endBridgeTileRowNum = 0
```

end bridge tile row num

9.68.2.5 **stm_forceDiceRoll**

```
bool Brood::Application::Data::ST_PathPrefabData::stm_forceDiceRoll = false
```

if true rolls dice instead of drawing a card

9.68.2.6 **stm_nextTileColNum**

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_nextTileColNum = 0
```

next tile column num

9.68.2.7 stm_nextTileRowNum

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_nextTileRowNum = 0  
next tile row num
```

9.68.2.8 stm_numberCardDraw

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_numberCardDraw = 0  
number of card to draw
```

9.68.2.9 stm_RowNum

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_RowNum = 0  
tile row num
```

9.68.2.10 stm_textureFileName

```
std::string Brood::Application::Data::ST_PathPrefabData::stm_textureFileName = ""  
filename for the texture; std::string
```

9.68.2.11 stm_tileType

```
unsigned Brood::Application::Data::ST_PathPrefabData::stm_tileType = 0  
type of the tile has number; Brood::Application::Components::ENUM\_TileType stored as unsigned  
The documentation for this struct was generated from the following files:
```

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.69 Brood::Application::Data::ST_PlayerManagerData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void **PopulateFromString** (std::string a_string)
populates the struct from passed string
- const std::string **GetString** () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- unsigned **stm_minPlayer** = 1
minimum number of player; unsinged
- unsigned **stm_maxPlayer** = 1
maximum number of player= 0; unsinged
- unsigned **stm_currPlayerIdx** = 0
current player idx= 0; unsinged

9.69.1 Member Function Documentation

9.69.1.1 GetString()

```
const std::string ST_PlayerManagerData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.69.1.2 PopulateFromString()

```
void ST_PlayerManagerData::PopulateFromString (
    std::string a_string )
```

populates the struct from passed string

Parameters

a_string	string containing all of the data seperated by ''
-----------------	---

9.69.2 Member Data Documentation

9.69.2.1 stm_currPlayerIdx

```
unsigned Brood::Application::Data::ST_PlayerManagerData::stm_currPlayerIdx = 0  
current player idx= 0; unsinged
```

9.69.2.2 stm_maxPlayer

```
unsigned Brood::Application::Data::ST_PlayerManagerData::stm_maxPlayer = 1  
maximum number of player= 0; unsinged
```

9.69.2.3 stm_minPlayer

```
unsigned Brood::Application::Data::ST_PlayerManagerData::stm_minPlayer = 1  
minimum number of player; unsinged
```

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.70 ST_PlayerManagerData Struct Reference

contains player manager data

```
#include <Data.h>
```

9.70.1 Detailed Description

contains player manager data

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.71 Brood::Application::Data::ST_PlayerPrefabData Struct Reference

```
#include <Data.h>
```

Public Member Functions

- void `PopulateFromString` (std::string a_string)
populates the struct from passed string
- const std::string `GetString` () const
creates a string and appends each data to its back seperated by ''

Public Attributes

- float `stm_currRow` = 0
tile row where the player is
- float `stm_currCol` = 0
tile column where the player is
- float `stm_playerSizeX` = 0
length of player
- float `stm_playerSizeY` = 0
height of player
- int `stm_playerOffsetX` = 0
x-offset of player
- int `stm_playerOffsetY` = 0
y-offset of player
- std::string `stm_textureFileName` = ""
filename for the texture

9.71.1 Member Function Documentation

9.71.1.1 `GetString()`

```
const std::string ST_PlayerPrefabData::GetString ( ) const
```

creates a string and appends each data to its back seperated by ''

Returns

string containg all the data appended to it seperated by ''

9.71.1.2 `PopulateFromString()`

```
void ST_PlayerPrefabData::PopulateFromString (
    std::string a_string)
```

populates the struct from passed string

Parameters

<i>a_string</i>	string containing all of the data seperated by ''
-----------------	---

9.71.2 Member Data Documentation

9.71.2.1 **stm_currCol**

```
float Brood::Application::Data::ST_PlayerPrefabData::stm_currCol = 0
```

tile column where the player is

9.71.2.2 **stm_currRow**

```
float Brood::Application::Data::ST_PlayerPrefabData::stm_currRow = 0
```

tile row where the player is

9.71.2.3 **stm_playerOffsetX**

```
int Brood::Application::Data::ST_PlayerPrefabData::stm_playerOffsetX = 0
```

x-offset of player

9.71.2.4 **stm_playerOffsetY**

```
int Brood::Application::Data::ST_PlayerPrefabData::stm_playerOffsetY = 0
```

y-offset of player

9.71.2.5 **stm_playerSizeX**

```
float Brood::Application::Data::ST_PlayerPrefabData::stm_playerSizeX = 0
```

length of player

9.71.2.6 **stm_playerSizeY**

```
float Brood::Application::Data::ST_PlayerPrefabData::stm_playerSizeY = 0  
height of player
```

9.71.2.7 **stm_textureFileName**

```
std::string Brood::Application::Data::ST_PlayerPrefabData::stm_textureFileName = ""  
filename for the texture
```

The documentation for this struct was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.cpp](#)

9.72 ST_PlayerPrefabData Struct Reference

contains player prefab data

```
#include <Data.h>
```

9.72.1 Detailed Description

contains player prefab data

The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.73 ST_TilePrefabData Struct Reference

contains tile data for one tile prefab

```
#include <Data.h>
```

9.73.1 Detailed Description

contains tile data for one tile prefab

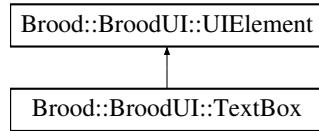
The documentation for this struct was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Data.h](#)

9.74 Brood::BroodUI::TextBox Class Reference

```
#include <TextBox.h>
```

Inheritance diagram for Brood::BroodUI::TextBox:



Public Member Functions

- `TextBox (Brood::BroodUI::UIElement *a_parentPtr=nullptr, Brood::BroodUI::ENUM_UIType a_enum←Type=Brood::BroodUI::ENUM_UIType::UI_textBox)`
Default Constructor.
- `virtual ~TextBox ()`
default destructor
- `TextBox (const TextBox &a_otherTextBox)`
Copy constructor.
- `Brood::BroodUI::TextBox & operator= (const TextBox &a_otherTextBox)`
assignment operator
- `const bool IsEditable () const`
getter funciton to get its editability
- `const bool IsSelected () const`
getter funciton to get if the element is currently selected or not
- `const bool IsEnterPressed () const`
getter funciton to get value stored in of enter button
- `void SetText (std::string a_text="") override`
setter function to set the text that is displayed in the button
- `void SetLimit (bool a_hasLimit=false, int a_limit=0)`
setter funciton to set the limit
- `void SetSelected (bool a_selected)`
setter function to set the state of the element i.e. if it is seleected or not
- `void SetEditable (bool a_isEditable)`
setter function to set if the element is edutable or not
- `void SetEnterPressedFalse ()`
setter function to set enterPressed to false
- `void SetPlaceHolderText (std::string a_placeHolderText)`
setter function to set enterPressed to false
- `void TypeOn (sf::Event a_input)`
Called when a character is typed a.
- `virtual bool DoElement () override`
checks if the logics of the element is to be executed or not
- `virtual void Draw (sf::RenderWindow &a_window) override`
Draws the body to the render window.
- `virtual void Debugger () override`
debugger

Static Public Member Functions

- static `Brood::BroodUI::TextBox * DyCreateTextBox` (`sf::Vector2f a_size, sf::Vector2f a_pos, std::string a_text="", bool a_isEditable=false, std::string a_placeholderText="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColoredTextColor`)
helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList
- static `Brood::BroodUI::TextBox * DyCreateTextBox` (`float a_sizeX, float a_sizeY, float a_posX, float a_posY, std::string a_text="", bool a_isEditable=false, std::string a_placeholderText="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor`)
helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList

Private Member Functions

- void `InputLogic` (`int charTyped`)
logic of what happens when different keys are pressed
- void `DeleteLastChar` ()
delete the last char form the user enterd string

Private Attributes

- `std::ostringstream m_ossText`
text buffer
- `std::string m_placeHolderText`
- `bool m_isEditable`
is true if the element's text is editable
- `bool m_hasLimit`
is true if the element has number of character limit
- `int m_limit`
max number of character

Static Private Attributes

- static `bool m_enterPressed = false`
saves if enter was pressed or not

Additional Inherited Members

9.74.1 Constructor & Destructor Documentation

9.74.1.1 TextBox() [1/2]

```
TextBox::TextBox (
    Brood::BroodUI::UIElement * a_parentPtr = nullptr,
    Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_textBox
)
```

Default Constructor.

Initializes the textBox object

It also sets the default value of isEditable, isSelected, and hasLimit to false. It sets the limit to -1.

Parameters

<code>m_parentPtr</code>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr
--------------------------	---

9.74.1.2 ~TextBox()

```
TextBox::~TextBox ( ) [virtual]
```

default destructor

9.74.1.3 TextBox() [2/2]

```
TextBox::TextBox ( const TextBox & a_otherTextBox )
```

Copy constructor.

Parameters

<code>a_otherTextBox</code>	reference to the uiElement which is used to copy the data form
-----------------------------	--

9.74.2 Member Function Documentation**9.74.2.1 Debugger()**

```
void TextBox::Debugger ( ) [override], [virtual]
```

debugger

Implements [Brood::BroodUI::UIElement](#).

9.74.2.2 DeleteLastChar()

```
void TextBox::DeleteLastChar ( ) [private]
```

delete the last char form the user enterd string

9.74.2.3 DoElement()

```
bool TextBox::DoElement ( ) [override], [virtual]
```

checks if the logics of the element is to be executed or not

In the process updates the element selector –that is current active and hot element. It also set the overlay

Returns

true if the element's function is to be executed

Reimplemented from [Brood::BroodUI::UIElement](#).

9.74.2.4 Draw()

```
void TextBox::Draw (
    sf::RenderWindow & a_window ) [override], [virtual]
```

Draws the body to the render window.

@virtual

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

Reimplemented from [Brood::BroodUI::UIElement](#).

9.74.2.5 DyCreateTextBox() [1/2]

```
Brood::BroodUI::TextBox * TextBox::DyCreateTextBox (
    float a_sizeX,
    float a_sizeY,
    float a_posX,
    float a_posY,
    std::string a_text = "",
    bool a_isEditable = false,
    std::string a_placeholderText = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
) [static]
```

helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_sizeX</i>	length of the textbox element
<i>a_sizeY</i>	width of the textbox element
<i>a_posX</i>	x-position of the textbox element
<i>a_posY</i>	y-position of the textbox element
<i>a_text</i>	text to display in the textbox element ; default ""
<i>a_isEditable</i>	true if the text box is editable; default false
<i>a_color</i>	color of the textbox element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created textbox element

9.74.2.6 DyCreateTextBox() [2/2]

```
Brood::BroodUI::TextBox * TextBox::DyCreateTextBox (
    sf::Vector2f a_size,
    sf::Vector2f a_pos,
    std::string a_text = "",
    bool a_isEditable = false,
    std::string a_placeholderText = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
) [static]
```

helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_size</i>	size of the textbox element
<i>a_pos</i>	position of the textbox element
<i>a_text</i>	text to display in the textbox element ; default ""
<i>a_isEditable</i>	true if the text box is editable; default false
<i>a_color</i>	color of the textbox element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created textbox element

9.74.2.7 InputLogic()

```
void TextBox::InputLogic (
    int charTyped ) [private]
```

logic of what happens when different keys are pressed

@reparam charTyped ascii number that represent the char

9.74.2.8 IsEditable()

```
const bool TextBox::IsEditable () const
```

getter funciton to get its editibility

Returns

true if the textbox is editable; else false

9.74.2.9 IsEnterPressed()

```
const bool TextBox::IsEnterPressed () const
```

getter funciton to get value stored in of enter button

Returns

value stored in of enter button

9.74.2.10 IsSelected()

```
const bool TextBox::IsSelected () const
```

getter funciton to get if the element is currently selected or not

Returns

true if it is currently selected; else false

9.74.2.11 operator=()

```
Brood::BroodUI::TextBox & TextBox::operator= (
    const TextBox & a_otherTextBox )
```

assignment operator

Parameters

<i>a_otherTextBox</i>	reference to the text box which is used to copy the data form
-----------------------	---

Returns

pointer to this element

9.74.2.12 SetEditable()

```
void TextBox::SetEditable (
    bool a_isEditable )
```

setter function to set if the element is edutable or not

Parameters

<i>a_isEditable</i>	true if text box is editable
---------------------	------------------------------

9.74.2.13 SetEnterPressedFalse()

```
void TextBox::SetEnterPressedFalse ( )
```

setter function to set enterPressed to false

9.74.2.14 SetLimit()

```
void TextBox::SetLimit (
    bool a_hasLimit = false,
    int a_limit = 0 )
```

setter funciton to set the limit

Note

setting the limit to zero will have 1 char limit

Parameters

<i>a_hasLimit</i>	true if the text box has a limit attached to it
<i>a_limit</i>	max number of character allowed

9.74.2.15 SetPlaceHolderText()

```
void TextBox::SetPlaceHolderText (
    std::string a_placeHolderText )
```

setter function to set enterPressed to false

9.74.2.16 SetSelected()

```
void TextBox::SetSelected (
    bool a_selected ) [virtual]
```

setter function to set the state of the element i.e. if it is selected or not

Parameters

<i>a_selected</i>	true if current text box is selected or not
-------------------	---

Reimplemented from [Brood::BroodUI::UIElement](#).

9.74.2.17 SetText()

```
void TextBox::SetText (
    std::string a_text = "" ) [override], [virtual]
```

setter function to set the text that is displayed in the button

Warning

It assumes that the font for the text is already set

Parameters

<i>a_text</i>	text to show on the button -> default empty string
---------------	--

Reimplemented from [Brood::BroodUI::UIElement](#).

9.74.2.18 TypeOn()

```
void TextBox::TypeOn (
    sf::Event a_input )
```

Called when a character is typed a.

Parameters

<i>a_input</i>	a copy of sf::Event::TextEntered
----------------	----------------------------------

9.74.3 Member Data Documentation

9.74.3.1 m_enterPressed

```
bool TextBox::m_enterPressed = false [static], [private]
```

saves if enter was pressed or not

@TODO fix the limit bug

allocating space and initializing the m_enterPressed

9.74.3.2 m_hasLimit

```
bool Brood::BroodUI::TextBox::m_hasLimit [private]
```

is true if the element has number of character limit

9.74.3.3 m_isEditable

```
bool Brood::BroodUI::TextBox::m_isEditable [private]
```

is true if the element's text is editable

9.74.3.4 m_limit

```
int Brood::BroodUI::TextBox::m_limit [private]
```

max number of character

9.74.3.5 m_ossText

```
std::ostringstream Brood::BroodUI::TextBox::m_ossText [private]
```

text buffer

9.74.3.6 m_placeHolderText

```
std::string Brood::BroodUI::TextBox::m_placeHolderText [private]
```

placeholder Text that gets displaced if the textbox is editable and has no text in the text buffer

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.h
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.cpp

9.75 TextBox Class Reference

a uielment that represent the text in the UI

```
#include "TextBox.h"
```

9.75.1 Detailed Description

a uielment that represent the text in the UI

A class derived from UI elements class. It can be set as mutable or immutable text.

Warning

The following code needs to be present in the event switch for the textbox to be editable:

```
case sf::Event::TextEntered:
{
    if( Brood::BroodUI::ElementSelection::GetCurrActiveElementIdPtr() == nullptr )
    {
        break;
    }
    // checking if the current active element is a editable textbox
    int currActiveElementId =
    Brood::BroodUI::ElementSelection::GetCurrActiveElementIdPtr()->GetElementID();
    Brood::BroodUI::UIElement* currActiveElement =
    Brood::BroodUI::MapIdToElement::GetElementPtrFromMap( currActiveElementId );
    if( currActiveElement->GetType() == Brood::BroodUI::ENUM_UIType::UI_textBox )
    {
        // checking if it is editable or not
        Brood::BroodUI::TextBox* currActiveTextBox = ( Brood::BroodUI::TextBox* )currActiveElement;
        if( currActiveTextBox->IsEditable() )
        {
            // if yes then sent the textEntered event to the element
            currActiveTextBox->TypeOn( events );
        }
    }
    break;
}
```

```

9.75.1.0.1 Example Case // window object
sf::RenderWindow window( sf::VideoMode( 500, 500 ), "BroodMaker" );
// event object
sf::Event events;
// creating a textbox object
myTextBox.SetBodySize( 100, 50 );
myTextBox.SetBodyPosition( 160, 350 );
myTextBox.SetBodyColor( sf::Color::White );
myTextBoxSetFont( font );
myTextBox.SetFontSize( 25 );
myTextBox.SetFontColor( sf::Color::Black );
myTextBox.SetText( "hello is this god" );
myTextBox.setEditable( true );
myTextBox.setLimit( true, 4 );
//app loop
while( !exit )
{
    // events
    while( window.pollEvent( events ) )
    {
        switch( events.type )
        {
            // other events
            // == in the even polling loop ==
            case sf::Event::TextEntered:
            {
                if( Brood::BroodUI::ElementSelection::GetCurrActiveElementIdPtr() == nullptr )
                {
                    break;
                }

                // checking if the current active element is a editable textbox
                int currActiveElementId =
                    Brood::BroodUI::ElementSelection::GetCurrActiveElementIdPtr()->GetElementID();
                Brood::BroodUI::UIElement* currActiveElement =
                    Brood::BroodUI::MapIdToElement::GetElementPtrFromMap( currActiveElementId );
                if( currActiveElement->GetElementType() == Brood::BroodUI::ENUM_UIType::UI_textBox )
                {
                    // checking if it is editable or not
                    Brood::BroodUI::TextBox* currActiveTextBox = ( Brood::BroodUI::TextBox*
                    )currActiveElement;
                    if( currActiveTextBox->IsEditable() )
                    {
                        // if yes then sent the textEntered event to the element
                        currActiveTextBox->TypeOn( events );
                    }
                }
                break;
            }
        }
    }
    // logics
    myTextBox.DoElement();
    // code to detect enter press then get the string user entered and process it
    if( myTextBox->IsEnterPressed() &&
        myTextBox->GetElementIdPtr() == Brood::BroodUI::ElementSelection::GetLastActiveElementIdPtr() )
    {
        std::cerr << myTextBox->GetText() << std::endl;

        // resetting the m_enterPressed
        myTextBox->SetEnterPressedFalse();
    }
    // render
    myTextBox.Draw( window );
}

```

Todo add a way to add place holder text

add view to the textbox feild that gets scrolled when the type text go of window

add a actual cursor instead of _

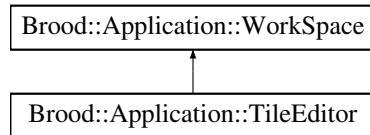
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[TextBox.h](#)

9.76 Brood::Application::TileEditor Class Reference

```
#include <TileEditor.h>
```

Inheritance diagram for Brood::Application::TileEditor:



Public Member Functions

- **TileEditor** (Brood::Application::Components::GameManager *aGameData, sf::RectangleShape *aPanelPtr)
default constructor
- **~TileEditor** ()
default destructor
- void **InitializeWorkSpace** ()
Initializes the work space.
- void **Update** ()
Updates function.
- void **UpdateAllDisplayElement** ()
updates the display element
- void **Draw** (sf::RenderWindow &a_window)
Draw funciton.
- void **Debugger** ()
debugger funciton

Private Member Functions

- void **UpdateTileRowNumber** ()
checks if the user interacted with the tile row panel
- void **UpdateTileColNumber** ()
checks if the user interacted with the tile column panel
- void **UpdateTileType** ()
checks if the user interacted with the tile type panel
- void **UpdateNextPathRowNumber** ()
checks if the user interacted with the TileNextPath row panel
- void **UpdateNextPathColNumber** ()
checks if the user interacted with the TileNextPath col panel
- void **UpdateEndBridgeRowNumber** ()
checks if the user interacted with the TileEndBridge row panel
- void **UpdateEndBridgeColNumber** ()
checks if the user interacted with the TileEndBridge col panel
- void **UpdateCorrectCardNumber** ()
checks if the user interacted with the Correct card number
- void **UpdateAssinedDeckIdx** ()
checks if the user interacted with the Assigned deck number
- void **UpdateForceDiceRoll** ()
checks if the user interacted with the force dice roll panel
- void **UpdateTileFileTexture** ()
checks if the user interacted with the laod tile texture panel

Private Attributes

- sf::RectangleShape * `m_panelBodyPtr`
holds the panel body
- Brood::BroodUI::TextBox * `m_txtSettingTitle`
- Brood::BroodUI::TextBox * `m_txtTileCurrRowPromt`
holds the text `Board` Row number prompt
- Brood::BroodUI::Button * `m_btnTileDecCurrRow`
clicking it will decrease the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileCurrRow`
displays the board's Row number
- Brood::BroodUI::Button * `m_btnTileIncCurrRow`
clicking it will increase the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileCurrColPromt`
holds the text `Board` Col number prompt
- Brood::BroodUI::Button * `m_btnTileDecCurrCol`
clicking it will decrease the board's Col by 1
- Brood::BroodUI::TextBox * `m_txtTileCurrCol`
displays the board's Col number
- Brood::BroodUI::Button * `m_btnTileIncCurrCol`
clicking it will increase the board's Col by 1
- Brood::BroodUI::TextBox * `m_txtTileTypePromt`
holds the text Tile type prompt
- Brood::BroodUI::DropDownInput * `m_ddiTileType`
- Brood::BroodUI::TextBox * `m_txtTileNextPathRowPromt`
holds the text `Board` Row number prompt
- Brood::BroodUI::Button * `m_btnTileDecNextPathRow`
clicking it will decrease the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileNextPathRow`
displays the board's Row number
- Brood::BroodUI::Button * `m_btnTileIncNextPathRow`
clicking it will increase the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileNextPathColPromt`
holds the text `Board` Col number prompt
- Brood::BroodUI::Button * `m_btnTileDecNextPathCol`
clicking it will decrease the board's Col by 1
- Brood::BroodUI::TextBox * `m_txtTileNextPathCol`
displays the board's Col number
- Brood::BroodUI::Button * `m_btnTileIncNextPathCol`
clicking it will increase the board's Col by 1
- Brood::BroodUI::TextBox * `m_txtTileEndBridgeRowPromt`
holds the text `Board` Row number prompt
- Brood::BroodUI::Button * `m_btnTileDecEndBridgeRow`
clicking it will decrease the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileEndBridgeRow`
displays the board's Row number
- Brood::BroodUI::Button * `m_btnTileIncEndBridgeRow`
clicking it will increase the board's row by 1
- Brood::BroodUI::TextBox * `m_txtTileEndBridgeColPromt`
holds the text `Board` Col number prompt
- Brood::BroodUI::Button * `m_btnTileDecEndBridgeCol`

- *clicking it will decrease the board's Col by 1*
 Brood::BroodUI::TextBox * m_txtTileEndBridgeCol
displays the board's Col number
- Brood::BroodUI::Button * m_btnTileIncEndBridgeCol
clicking it will increase the board's Col by 1
- Brood::BroodUI::TextBox * m_txtTileNumCorrectCardPromt
holds the text tile's NumCorrectCard prompt
- Brood::BroodUI::Button * m_btnTileDecNumCorrectCard
clicking it will decrease the tile's NumCorrectCard by 1
- Brood::BroodUI::TextBox * m_txtTileNumCorrectCard
displays the tile's NumCorrectCard number
- Brood::BroodUI::Button * m_btnTileIncNumCorrectCard
clicking it will increase the tile's NumCorrectCard by 1
- Brood::BroodUI::TextBox * m_txtTileAssignedDeckIdxPromt
holds the text tile's assigned DeckIdx prompt
- Brood::BroodUI::Button * m_btnTileDecAssignedDeckIdx
clicking it will decrease the tile's DeckIdx by 1
- Brood::BroodUI::TextBox * m_txtTileAssignedDeckIdx
displays the tile's DeckIdx
- Brood::BroodUI::Button * m_btnTileIncAssignedDeckIdx
clicking it will increase the tile's DeckIdx by 1
- Brood::BroodUI::TextBox * m_txtTileForceDiceRollPromt
holds the text Force dice prompt
- Brood::BroodUI::DropDownInput * m_ddiTileForceDiceRoll
holds yes or no
- Brood::BroodUI::TextBox * m_txtTileBgFileNamePrompt
holds the Tile texture filename prompt
- Brood::BroodUI::TextBox * m_txtTileBgFileInput
holds the file name user enters
- Brood::BroodUI::Button * m_btnTileBgOpenFile
clicking it will load the Tile file

Additional Inherited Members

9.76.1 Constructor & Destructor Documentation

9.76.1.1 TileEditor()

```
TileEditor::TileEditor (
    Brood::Application::Components::GameManager * aGameData,
    sf::RectangleShape * a_panelPtr )
```

default constructor

Parameters

<i>aGameData</i>	pointer to the game data object
<i>aPanelPtr</i>	pointer to the panel body object

9.76.1.2 ~TileEditor()

```
TileEditor::~TileEditor ( )
```

default destructor

9.76.2 Member Function Documentation**9.76.2.1 Debugger()**

```
void TileEditor::Debugger ( ) [virtual]
```

debugger funciton

@virtual

This function helps in debugging the UI elements.

Implements [Brood::Application::WorkSpace](#).

9.76.2.2 Draw()

```
void TileEditor::Draw (
    sf::RenderWindow & aWindow ) [virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Parameters

<i>aWindow</i>	reference to the render window
----------------	--------------------------------

Implements [Brood::Application::WorkSpace](#).

9.76.2.3 InitializeWorkSpace()

```
void TileEditor::InitializeWorkSpace ( ) [virtual]
```

Initializes the work space.

@virtual

It creates all the component of the workspace. It initializes the board, setting section Dropdown Input.

Implements [Brood::Application::WorkSpace](#).

9.76.2.4 Update()

```
void TileEditor::Update ( ) [virtual]
```

Updates function.

@virtual

Contains all the logic for the initial workspace

Implements [Brood::Application::WorkSpace](#).

9.76.2.5 UpdateAllDisplayElement()

```
void TileEditor::UpdateAllDisplayElement ( ) [virtual]
```

updates the display element

@virtual

Implements [Brood::Application::WorkSpace](#).

9.76.2.6 UpdateAssinedDeckIdx()

```
void TileEditor::UpdateAssinedDeckIdx ( ) [private]
```

checks if the user interacted with the Assigned deck number

Assigned deck number panel contains the Assigned deck number prompt textbox, Assigned deck number value text box, button to increase the Assigned deck number number, and button to decrease the Assigned deck number number

Only the button to increase the Assigned deck number, and button to decrease the Assigned deck number are interactable

If the interactable button was pressed then the Assigned deck number is increased or decreased by 1. Assigned deck number should be more than or equal to 0 but less than number of cards in the assigned deck

9.76.2.7 UpdateCorrectCardNumber()

```
void TileEditor::UpdateCorrectCardNumber ( ) [private]
```

checks if the user interacted with the Correct card number

Correct card number panel contains the Correct card number prompt textbox, Correct card number value text box, button to increase the Correct card number number, and button to decrease the Correct card number number

Only the button to increase the Correct card number, and button to decrease the Correct card number are interactable

If the interactable button was pressed then the Correct card number is increased or decreased by 1. Correct card number should be more than or equal to 1 but less than number of cards in the assigned deck

9.76.2.8 UpdateEndBridgeColNumber()

```
void TileEditor::UpdateEndBridgeColNumber ( ) [private]
```

checks if the user interacted with the TileEndBridge col panel

TileEndBridge col panel contains the TileEndBridge col number prompt textbox, TileEndBridge col number value text box, button to increase the TileEndBridge col number, and button to decrease the TileEndBridge col number

Only the button to increase the TileEndBridge col number, and button to decrease the TileEndBridge col number are interactable

If the interactable button was pressed then the TileEndBridge col is increased or decreased by 1. TileEndBridge col number should be more than or equal to 0 but less than board col number

9.76.2.9 UpdateEndBridgeRowNumber()

```
void TileEditor::UpdateEndBridgeRowNumber ( ) [private]
```

checks if the user interacted with the TileEndBridge row panel

TileEndBridge row panel contains the TileEndBridge row number prompt textbox, TileEndBridge row number value text box, button to increase the TileEndBridge row number, and button to decrease the TileEndBridge row number

Only the button to increase the TileEndBridge row number, and button to decrease the TileEndBridge row number are interactable

If the interactable button was pressed then the TileEndBridge row is increased or decreased by 1. TileEndBridge row number should be more than or equal to 0 but less than board row number

9.76.2.10 UpdateForceDiceRoll()

```
void TileEditor::UpdateForceDiceRoll ( ) [private]
```

checks if the user interacted with the force dice roll panel

Tile type panel contains force dice roll panel prompt textbox, and drop down input to select a yes or no

Only the drop down input to select the yes or no is interactable

If the interactable drop down input was pressed then it expands to show the two options available yes and no

9.76.2.11 UpdateNextPathColNumber()

```
void TileEditor::UpdateNextPathColNumber ( ) [private]
```

checks if the user interacted with the TileNextPath col panel

TileNextPath col panel contains the TileNextPath col number prompt textbox, TileNextPath col number value text box, button to increase the TileNextPath col number, and button to decrease the TileNextPath col number

Only the button to increase the TileNextPath col number, and button to decrease the TileNextPath col number are interactable

If the interactable button was pressed then the TileNextPath col is increased or decreased by 1. TileNextPath col number should be more than or equal to 0 but less than board col number

9.76.2.12 UpdateNextPathRowNumber()

```
void TileEditor::UpdateNextPathRowNumber ( ) [private]
```

checks if the user interacted with the TileNextPath row panel

TileNextPath row panel contains the TileNextPath row number prompt textbox, TileNextPath row number value text box, button to increase the TileNextPath row number, and button to decrease the TileNextPath row number

Only the button to increase the TileNextPath row number, and button to decrease the TileNextPath row number are interactable

If the interactable button was pressed then the TileNextPath row is increased or decreased by 1. TileNextPath row number should be more than or equal to 0 and can only be smaller or bigger than current tile row by 1

9.76.2.13 UpdateTileColNumber()

```
void TileEditor::UpdateTileColNumber ( ) [private]
```

checks if the user interacted with the tile column panel

Tile column panel contains the tile column number prompt textbox, tile column number value text box, button to increase the tile column number, and button to decrease the tile column number

Only the button to increase the tile column number, and button to decrease the tile column number are interactable

If the interactable button was pressed then the tile column is increased or decreased by 1. Tile column number should be more than or equal to 0 but less than board column number

9.76.2.14 UpdateTileFileTexture()

```
void TileEditor::UpdateTileFileTexture ( ) [private]
```

checks if the user interacted with the load tile texture panel

Load tile texture file panel contains load tile texture file prompt textbox, textbox to enter the file name, and button to open and load the file.

Only the textbox to enter the file name, and button to open and load the file are interactable

If the interactable button was pressed then it opens the filename entered

Warning

Texture should always be in .png form

the Texture for the file should be in \Assets\Textures\TileTexture\ as it loads them from there.

9.76.2.15 UpdateTileRowNumber()

```
void TileEditor::UpdateTileRowNumber ( ) [private]
```

checks if the user interacted with the tile row panel

Tile row panel contains the Tile row number prompt textbox, Tile row number value text box, button to increase the Tile row number, and button to decrease the Tile row number

Only the button to increase the Tile row number, and button to decrease the Tile row number are interactable

If the interactable button was pressed then the Tile row is increased or decreased by 1. Tile row number should be more than or equal to 0 but less than board row number

9.76.2.16 UpdateTileType()

```
void TileEditor::UpdateTileType ( ) [private]
```

checks if the user interacted with the tile type panel

Tile type panel contains **Board** tile type prompt textbox, and drop down input to select a tile type

Only the drop down input to select the tile type is interactable

If the interactable drop down input was pressed then it expands to show the five options available: blank tile, tile tile, start tile, end tile, bridge tile

9.76.3 Member Data Documentation

9.76.3.1 m_btnTileBgOpenFile

```
Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileBgOpenFile [private]
```

clicking it will load the Tile file

9.76.3.2 m_btnTileDecAssignedDeckIdx

```
Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecAssignedDeckIdx [private]
```

clicking it will decrease the tile's DeckIdx by 1

9.76.3.3 m_btnTileDecCurrCol

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecCurrCol [private]

clicking it will decrease the board's Col by 1

9.76.3.4 m_btnTileDecCurrRow

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecCurrRow [private]

clicking it will decrease the board's row by 1

9.76.3.5 m_btnTileDecEndBridgeCol

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecEndBridgeCol [private]

clicking it will decrease the board's Col by 1

9.76.3.6 m_btnTileDecEndBridgeRow

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecEndBridgeRow [private]

clicking it will decrease the board's row by 1

9.76.3.7 m_btnTileDecNextPathCol

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecNextPathCol [private]

clicking it will decrease the board's Col by 1

9.76.3.8 m_btnTileDecNextPathRow

Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecNextPathRow [private]

clicking it will decrease the board's row by 1

9.76.3.9 m_btnTileDecNumCorrectCard

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileDecNumCorrectCard [private]`

clicking it will decrease the tile's NumCorrectCard by 1

9.76.3.10 m_btnTileIncAssignedDeckIdx

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncAssignedDeckIdx [private]`

clicking it will increase the tile's DeckIdx by 1

9.76.3.11 m_btnTileIncCurrCol

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncCurrCol [private]`

clicking it will increase the board's Col by 1

9.76.3.12 m_btnTileIncCurrRow

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncCurrRow [private]`

clicking it will increase the board's row by 1

9.76.3.13 m_btnTileIncEndBridgeCol

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncEndBridgeCol [private]`

clicking it will increase the board's Col by 1

9.76.3.14 m_btnTileIncEndBridgeRow

`Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncEndBridgeRow [private]`

clicking it will increase the board's row by 1

9.76.3.15 m_btnTileIncNextPathCol

```
Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncNextPathCol [private]
```

clicking it will increase the board's Col by 1

9.76.3.16 m_btnTileIncNextPathRow

```
Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncNextPathRow [private]
```

clicking it will increase the board's row by 1

9.76.3.17 m_btnTileIncNumCorrectCard

```
Brood::BroodUI::Button* Brood::Application::TileEditor::m_btnTileIncNumCorrectCard [private]
```

clicking it will increase the tile's NumCorrectCard by 1

9.76.3.18 m_ddiTileForceDiceRoll

```
Brood::BroodUI::DropDownInput* Brood::Application::TileEditor::m_ddiTileForceDiceRoll [private]
```

holds yes or no

9.76.3.19 m_ddiTileType

```
Brood::BroodUI::DropDownInput* Brood::Application::TileEditor::m_ddiTileType [private]
```

type of the tiles: start tile, end tile, bridge Start tile, blank tile

9.76.3.20 m_panelBodyPtr

```
sf::RectangleShape* Brood::Application::TileEditor::m_panelBodyPtr [private]
```

holds the panel body

9.76.3.21 m_txtSettingTitle

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtSettingTitle [private]
```

9.76.3.22 m_txtTileAssignedDeckIdx

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileAssignedDeckIdx [private]
```

displays the tile's DeckIdx

9.76.3.23 m_txtTileAssignedDeckIdxPromt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileAssignedDeckIdxPromt [private]
```

holds the text tile's assigned DeckIdx prompt

9.76.3.24 m_txtTileBgFileInput

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileBgFileInput [private]
```

holds the file name user enters

9.76.3.25 m_txtTileBgFileNamePrompt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileBgFileNamePrompt [private]
```

holds the Tile texture filename prompt

9.76.3.26 m_txtTileCurrCol

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileCurrCol [private]
```

displays the board's Col number

9.76.3.27 m_txtTileCurrColPromt

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileCurrColPromt [private]

holds the text **Board** Col number prompt

9.76.3.28 m_txtTileCurrRow

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileCurrRow [private]

displays the board's Row number

9.76.3.29 m_txtTileCurrRowPromt

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileCurrRowPromt [private]

holds the text **Board** Row number prompt

9.76.3.30 m_txtTileEndBridgeCol

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileEndBridgeCol [private]

displays the board's Col number

9.76.3.31 m_txtTileEndBridgeColPromt

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileEndBridgeColPromt [private]

holds the text **Board** Col number prompt

9.76.3.32 m_txtTileEndBridgeRow

Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileEndBridgeRow [private]

displays the board's Row number

9.76.3.33 m_txtTileEndBridgeRowPromt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileEndBridgeRowPromt [private]
```

holds the text **Board** Row number prompt

9.76.3.34 m_txtTileForceDiceRollPromt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileForceDiceRollPromt [private]
```

holds the text Force dice prompt

9.76.3.35 m_txtTileNextPathCol

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNextPathCol [private]
```

displays the board's Col number

9.76.3.36 m_txtTileNextPathColPromt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNextPathColPromt [private]
```

holds the text **Board** Col number prompt

9.76.3.37 m_txtTileNextPathRow

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNextPathRow [private]
```

displays the board's Row number

9.76.3.38 m_txtTileNextPathRowPromt

```
Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNextPathRowPromt [private]
```

holds the text **Board** Row number prompt

9.76.3.39 m_txtTileNumCorrectCard

`Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNumCorrectCard [private]`

displays the tile's NumCorrectCard number

9.76.3.40 m_txtTileNumCorrectCardPromt

`Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileNumCorrectCardPromt [private]`

holds the text tile's NumCorrectCard prompt

9.76.3.41 m_txtTileTypePromt

`Brood::BroodUI::TextBox* Brood::Application::TileEditor::m_txtTileTypePromt [private]`

holds the text Tile type prompt

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[TileEditor.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[TileEditor.cpp](#)

9.77 TileEditor Class Reference

It allows the users to tweek the deck setting.

```
#include "DeckEditor.h"
```

9.77.1 Detailed Description

It allows the users to tweek the deck setting.

It allows the users to tweek the tile setting.

It is a container that bundels elements that is present in DeckEditor workspace

Derived form Workspace class

It is a container that bundels elements that is present in [TileEditor](#) workspace

Derived form Workspace class

Todo allowd to enter number in the txt feild

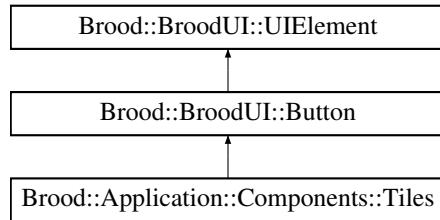
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[DeckEditor.h](#)

9.78 Brood::Application::Components::Tiles Class Reference

```
#include <Tiles.h>
```

Inheritance diagram for Brood::Application::Components::Tiles:



Public Member Functions

- [Tiles \(\)](#)
Default Constructor.
- [~Tiles \(\)](#)
Default Destructor.
- [Tiles \(const Tiles &a_otherTile\)](#)
Copy contrutor.
- [Tiles & operator= \(const Tiles &a_other\)](#)
assignment operator
- const sf::Vector2u [GetRowAndCol \(\) const](#)
Getter Function to get the postion of the tile relative to the board's column and row.
- const unsigned [GetRow \(\) const](#)
Getter Function to get the row number of the tile relative to the board's row.
- const unsigned [GetCol \(\) const](#)
Getter Function to get the col number of the tile relative to the board's col.
- void [SetRowAndCol \(sf::Vector2u a_rowCol\)](#)
Setter Function to set the postion of the tile relative to the board's column and row.
- void [SetRow \(unsigned a_row\)](#)
Setter Function to set the row number of the tile relative to the board's row.
- void [SetCol \(unsigned a_col\)](#)
Setter Function to set the col number of the tile relative to the board's col.
- void [UpdateTile \(unsigned a_tileRowNum=0, unsigned a_tileColNum=0, float a_tileSizeX=0.f, float a_tileSizeY=0.f, float a_boardPosX=0.f, float a_boardPosY=0.f\)](#)
updates the member variable and size of this tile

Private Attributes

- unsigned [m_tileRowNum](#)
row position in a board
- unsigned [m_tileColNum](#)
column position in a board

Additional Inherited Members

9.78.1 Constructor & Destructor Documentation

9.78.1.1 Tiles() [1/2]

```
Tiles::Tiles ( )
```

Default Constructor.

9.78.1.2 ~Tiles()

```
Tiles::~Tiles ( )
```

Default Destructors.

9.78.1.3 Tiles() [2/2]

```
Tiles::Tiles ( const Tiles & a_otherTile )
```

Copy contrutor.

9.78.2 Member Function Documentation

9.78.2.1 GetCol()

```
const unsigned Tiles::GetCol ( ) const
```

Getter Function to get the col number of the tile relative to the board's col.

Returns

col number of the tile relative to the board's col

9.78.2.2 GetRow()

```
const unsigned Tiles::GetRow ( ) const
```

Getter Function to get the row number of the tile relative to the board's row.

Returns

row number of the tile relative to the board's row

9.78.2.3 GetRowAndCol()

```
const sf::Vector2u Tiles::GetRowAndCol ( ) const
```

Getter Function to get the position of the tile relative to the board's column and row.

Todo add copy constructor and assignment

Returns

a vector2i containing the position of the tile; .x-> is the tile's column number; .y-> is the tile's row number.

9.78.2.4 operator=()

```
Brood::Application::Components::Tiles & Tiles::operator= (
    const Tiles & a_other )
```

assignment operator

9.78.2.5 SetCol()

```
void Tiles::SetCol (
    unsigned a_col )
```

Setter Function to set the col number of the tile relative to the board's col.

Parameters

<code>a_row</code>	row number of the tile relative to the board's col
--------------------	--

9.78.2.6 SetRow()

```
void Tiles::SetRow (
    unsigned a_row )
```

Setter Function to set the row number of the tile relative to the board's row.

Parameters

<i>a_row</i>	row number of the tile relative to the board's row
--------------	--

9.78.2.7 SetRowAndCol()

```
void Tiles::SetRowAndCol (
    sf::Vector2u a_rowCol )
```

Setter Function to set the position of the tile relative to the board's column and row.

Parameters

<i>a_rowCol</i>	a vector2u containing the position of the tile; .x-> is the tile's column number; .y-> is the tile's row number.
-----------------	--

9.78.2.8 UpdateTile()

```
void Tiles::UpdateTile (
    unsigned a_tileRowNum = 0,
    unsigned a_tileColNum = 0,
    float a_tileSizeX = 0.f,
    float a_tileSizeY = 0.f,
    float a_boardPosX = 0.f,
    float a_boardPosY = 0.f )
```

updates the member variable and size of this tile

Parameters

<i>a_tileRowNum</i>	represents row number of the board -> default 0
<i>a_tileColNum</i>	represents column of the board -> default 0
<i>a_tileSizeX</i>	size of each tile in x-asix -> default 0.f
<i>a_tileSizeY</i>	size of each tile in y-asix -> default 0.f
<i>a_boardPosX</i>	board's x-position on screen -> default 0.f relative to the render window -> default 0.f
<i>a_boardPosY</i>	board's y-position on screen; realtive to the render window -> default 0.f

9.78.3 Member Data Documentation

9.78.3.1 m_tileColNum

```
unsigned Brood::Application::Components::Tiles::m_tileColNum [private]
```

column position in a board

9.78.3.2 m_tileRowNum

```
unsigned Brood::Application::Components::Tiles::m_tileRowNum [private]
```

row position in a board

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Tiles.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Tiles.cpp](#)

9.79 Tiles Class Reference

building block of a board.

```
#include "Tiles.h"
```

9.79.1 Detailed Description

building block of a board.

[Tiles](#) are the basic building block of the board. This class provides the functionality to alter and customize each tiles. Thus give the ability to customizing the whole board.

Derived from [Button](#) Class

Setting up a tile is similar to setting up a button. But as the tile is used as a building block for a board it has tile row and tile number. For more information

See also

[Brood::BroodUI::Button](#) class.

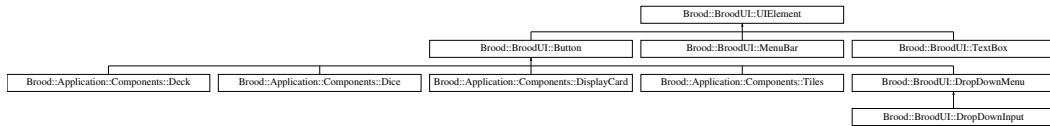
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[Tiles.h](#)

9.80 Brood::BroodUI::UIElement Class Reference

```
#include <UIElement.h>
```

Inheritance diagram for Brood::BroodUI::UIElement:



Public Member Functions

- `UIElement (Brood::BroodUI::ENUM_UIType a_elementType, Brood::BroodUI::UIElement *a_parent ← Ptr=nullptr)`
default Constructor
- `virtual ~UIElement ()`
virtual destructor
- `UIElement (const UIElement &a_otherElement)`
Copy constructor.
- `Brood::BroodUI::UIElement & operator= (const UIElement &a_otherElement)`
assignment operator
- `const Brood::BroodUI::ENUM_UIType GetElementType () const`
getter function to get the element type
- `const sf::RectangleShape & GetBody () const`
Getter function to get a reference to the element.
- `const sf::Vector2f GetBodyPosition () const`
Getter function to get the positon of the element.
- `const sf::Vector2f GetBodySize () const`
Getter function to get the size of element.
- `const sf::Color GetBodyColor ()`
Getter function to get the element's body color.
- `const sf::Color GetActiveOverlayColor ()`
Getter function to get the element's active overlay color.
- `const sf::Color GetHotOverlayColor ()`
Getter function to get the element's hot overlay color.
- `Brood::BroodUI::Id * GetElementIdPtr ()`
Getter function to get the pointer to the element Id object.
- `const bool IsSelected () const`
getter funciton to get if the element is currently selected or not
- `std::string GetText () const`
getter funciton to get the user typed text that is stored in the element
- `unsigned int GetFontSize () const`
getter funciton to get the font size
- `virtual void SetBodySize (sf::Vector2f a_size)`
Setter function to set the element's Size.
- `virtual void SetBodySize (float a_sizeX, float a_sizeY)`
Setter function to set the element's Size.
- `virtual void SetBodyPosition (sf::Vector2f a_pos, bool a_relativeToParent=false)`
Setter function to set the element's Position.

- virtual void [SetBodyPosition](#) (float a_posX, float a_posY, bool a_relativeToParent=false)
Setter function to set the element's Position.
- virtual void [SetFont](#) (sf::Font &a_font)
setter function to set the font size
- virtual void [SetFontColor](#) (sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_White)
setter function to set the Font color
- virtual void [SetFontSize](#) (int a_fontSize=12)
setter function to set the Font size
- virtual virtual void [SetText](#) (std::string a_text="")
setter function to set the text that is displayed in the button
- virtual virtual void [SetSelected](#) (bool a_selected)
setter function to set the state of the element i.e. if it is selected or not
- virtual void [SetBodyColor](#) (sf::Color a_bodyColor)
Setter function to set the element body color.
- virtual void [SetActiveOverlayColor](#) (sf::Color a_color)
Setter function to set the element's active color.
- virtual void [SetHotOverlayColor](#) (sf::Color a_color)
Setter function to set the element's hot color.
- bool [IsMouseOverElement](#) ()
Check if the mouse is hovering over the element.
- bool [IsAlmostActiveElement](#) ()
Checks if the element is the active element or not.
- bool [IsCurrActiveElement](#) ()
Checks if the element is the current active element or not.
- bool [IsHotElement](#) ()
Checks if the element is the hot element or not.
- virtual bool [DoElement](#) ()
checks if the logics of the element is to be executed or not
- virtual void [Draw](#) (sf::RenderWindow &a_window)
Draws the body to the render window.
- virtual void [Debugger](#) ()=0
changes the text of the element to its the element ID when debug is on and changes it back to the text it had before debugging when debug is off.

Protected Member Functions

- void [SetTextPosition](#) ()
setter funciton to set the position of the text.
- void [SetDrawOverlay](#) ()
Setter function to set if the overlay is to be drawn or not.

Protected Attributes

- Brood::BroodUI::ENUM_UIType m_elementType
stores the type of element stored
- Brood::BroodUI::Id m_elementId
unique id of the element
- sf::RectangleShape m_body
element body

- sf::RectangleShape **m_bodyOverLay**
element body
- sf::Color **m_hotOverlayColor**
color when mouse is over it
- sf::Color **m_activeOverlayColor**
color when it is active element
- bool **m_drawOverlay**
true if overlay is to be drawn
- bool **m_isSelected**
is true if the element is selected
- sf::Text **m_text**
text content
- std::string **m_textContent**
saves the text content
- std::string **m_debugTextSave**
saves the text content before replacing by uid
- sf::Font * **m_font**
element's font
- unsigned **m_fontSize**
font size
- bool **m_drawText**
is true if text is set and hence draw to screen

9.80.1 Constructor & Destructor Documentation

9.80.1.1 UIElement() [1/2]

```
UIElement::UIElement (
    Brood::BroodUI::ENUM_UIType a_elementType,
    Brood::BroodUI::UIElement * a_parentPtr = nullptr )
```

default Constructor

Initializes the member variables and adds it to the id to element map.

Sets hotelement overlay color to lighter overlay and sets activeOvelay to a darker overlay.

Parameters

<i>a_elementType</i>	Type of UI element
<i>m_parentPtr</i>	pointer to the parent element; if parent does not exist then nullptr -> default value nullptr

9.80.1.2 ~UIElement()

```
UIElement::~UIElement ( ) [virtual]
```

virtual destructor

@virtual

Deletes element form the parent's child list if parent exist and assigns the its child element to the parent element if parent exist. And finally deletes its id from the id to element map.

9.80.1.3 UIElement() [2/2]

```
UIElement::UIElement (
    const UIElement & a_otherElement )
```

Copy constructor.

Parameters

<i>a_otherElement</i>	reference to the uiElement which is used to copy the data form
-----------------------	--

9.80.2 Member Function Documentation

9.80.2.1 Debugger()

```
void UIElement::Debugger ( ) [pure virtual]
```

changes the text of the element to its the element ID when debug is on and changes it back to the text it had before debugging when debug is off.

Implemented in [Brood::BroodUI::Button](#), [Brood::Application::Components::DisplayCard](#), [Brood::BroodUI::DropDownMenu](#), [Brood::BroodUI::MenuBar](#), and [Brood::BroodUI::TextBox](#).

9.80.2.2 DoElement()

```
bool UIElement::DoElement ( ) [virtual]
```

checks if the logics of the element is to be executed or not

It checks the mouse position and button state to determine if to execute the elements logic or not. It does this by manipulating the element selection class. It also set the overlay

Returns

true if the element's funciton is to be executed; else false

Reimplemented in [Brood::BroodUI::Button](#), [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::TextBox](#).

9.80.2.3 Draw()

```
void UIElement::Draw (
    sf::RenderWindow & a_window ) [virtual]
```

Draws the body to the render window.

@virtual

Parameters

<code>a_window</code>	reference to the render window
-----------------------	--------------------------------

Reimplemented in [Brood::Application::Components::DisplayCard](#), [Brood::BroodUI::DropDownMenu](#), [Brood::BroodUI::MenuBar](#), and [Brood::BroodUI::TextBox](#).

9.80.2.4 GetActiveOverlayColor()

```
const sf::Color UIElement::GetActiveOverlayColor ()
```

Getter function to get the element's active overlay color.

@reutrn color of the body when it is active

9.80.2.5 GetBody()

```
const sf::RectangleShape & UIElement::GetBody () const
```

Getter function to get a reference to the element.

Returns

constant reference to the element

9.80.2.6 GetBodyColor()

```
const sf::Color UIElement::GetBodyColor ()
```

Getter function to get the element's body color.

@reutrn color of the body

9.80.2.7 GetBodyPosition()

```
const sf::Vector2f UIElement::GetBodyPosition () const
```

Getter function to get the positon of the element.

Returns

position of the element

9.80.2.8 GetBodySize()

```
const sf::Vector2f UIElement::GetBodySize() const
```

Getter function to get the size of element.

Returns

size of the body

9.80.2.9 GetElementIdPtr()

```
Brood::BroodUI::Id * UIElement::GetElementIdPtr()
```

Getter function to get the pointer to the element [Id](#) object.

Returns

pointer to the element [Id](#)

9.80.2.10 GetElementType()

```
const Brood::BroodUI::ENUM_UIType UIElement::GetElementType() const
```

getter function to get the element type

Returns

the element type

9.80.2.11 GetFontSize()

```
unsigned int UIElement::GetFontSize() const
```

getter funciton to get the font size

Returns

size of the font

9.80.2.12 GetHotOverlayColor()

```
const sf::Color UIElement::GetHotOverlayColor() 
```

Getter function to get the element's hot overlay color.

@return color of the body when it is hot

9.80.2.13 GetText()

```
std::string UIElement::GetText() const 
```

getter function to get the user typed text that is stored in the element

Returns

all the user typed text that is stored in the element

9.80.2.14 IsAlmostActiveElement()

```
bool UIElement::IsAlmostActiveElement() 
```

Checks if the element is the active element or not.

Returns

true if it is the active element; else false

9.80.2.15 IsCurrActiveElement()

```
bool UIElement::IsCurrActiveElement() 
```

Checks if the element is the current active element or not.

Returns

true if it is the curr active element; else false

9.80.2.16 IsHotElement()

```
bool UIElement::IsHotElement ( )
```

Checks if the element is the hot element or not.

Returns

true if it is the hot element; else false

9.80.2.17 IsMouseOverElement()

```
bool UIElement::IsMouseOverElement ( )
```

Check if the mouse is hovering over the element.

Returns

true if the mouse is over the element; else false

9.80.2.18 IsSelected()

```
const bool UIElement::IsSelected ( ) const
```

getter funciton to get if the element is currently selected or not

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Returns

true if it is currently selected; else false

9.80.2.19 operator=()

```
Brood::BroodUI::UIElement & UIElement::operator= (
    const UIElement & a_otherElement )
```

assignment operator

Parameters

<i>a_otherElement</i>	reference to the uiElement which is used to copy the data form
-----------------------	--

Returns

pointer to this element

9.80.2.20 SetActiveOverlayColor()

```
void UIElement::SetActiveOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's active color.

Parameters

<i>a_color</i>	color of the body when it is active
----------------	-------------------------------------

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.21 SetBodyColor()

```
void UIElement::SetBodyColor (
    sf::Color a_bodyColor ) [virtual]
```

Setter function to set the element body color.

Parameters

<i>a_bodyColor</i>	color of the body
--------------------	-------------------

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.22 SetBodyPosition() [1/2]

```
void UIElement::SetBodyPosition (
    float a_posX,
    float a_posY,
    bool a_relativeToParent = false ) [virtual]
```

Setter function to set the element's Position.

@virtual This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_posX</i>	x-position of the element
<i>a_posY</i>	y-position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.23 SetBodyPosition() [2/2]

```
void UIElement::SetBodyPosition (
    sf::Vector2f a_pos,
    bool a_relativeToParent = false ) [virtual]
```

Setter function to set the element's Position.

@virtual

It can also set the position relative to its parent if exist by passing a second parameter to true. If second parameter is pass as true and na parent does not exist then it sets the position as it is

Parameters

<i>a_pos</i>	position of the element
<i>a_relativeToParent</i>	is true if the passed position is relative to its parent; default -> false.

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.24 SetBodySize() [1/2]

```
void UIElement::SetBodySize (
    float a_sizeX,
    float a_sizeY ) [virtual]
```

Setter function to set the element's Size.

@virtual This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

<i>a_sizeX</i>	length of the element
<i>a_sizeY</i>	width of the element

Reimplemented in [Brood::BroodUI::DropDownMenu](#), [Brood::BroodUI::MenuBar](#), and [Brood::Application::Components::Dice](#).

9.80.2.25 SetBodySize() [2/2]

```
void UIElement::SetBodySize (
    sf::Vector2f a_size )  [virtual]
```

Setter function to set the element's Size.

@virtual

Parameters

<i>a_size</i>	size of the element
---------------	---------------------

Reimplemented in [Brood::BroodUI::DropDownMenu](#), [Brood::BroodUI::MenuBar](#), and [Brood::Application::Components::Dice](#).

9.80.2.26 SetDrawOverlay()

```
void UIElement::SetDrawOverlay ( )  [protected]
```

Setter function to set if the overlay is to be drawn or not.

9.80.2.27 SetFont()

```
void UIElement::SetFont (
    sf::Font & a_font )  [virtual]
```

setter function to set the font size

9.80.2.28 SetFontColor()

```
void UIElement::SetFontColor (
    sf::Color a_color = Brood::Application::StaticVariables::ST\_ColorVariables::stm\_White
)  [virtual]
```

setter function to set the Font color

param a_color font color -> default sf::Color::White

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.29 SetFontSize()

```
void UIElement::SetFontSize (
    int a_fontSize = 12 )  [virtual]
```

setter function to set the Font size

Note

If the font size is more than body height by

Parameters

<code>a_charSize</code>	-> size of individual character in the SetEditabtext -> default 12
-------------------------	--

9.80.2.30 SetHotOverlayColor()

```
void UIElement::SetHotOverlayColor (
    sf::Color a_color ) [virtual]
```

Setter function to set the element's hot color.

Parameters

<code>a_color</code>	color of the body when it is hot
----------------------	----------------------------------

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::MenuBar](#).

9.80.2.31 SetSelected()

```
void UIElement::SetSelected (
    bool a_selected ) [virtual]
```

setter function to set the state of the element i.e. if it is selected or not

Parameters

<code>a_selected</code>	true if current text box is selected or not
-------------------------	---

Reimplemented in [Brood::BroodUI::Button](#), and [Brood::BroodUI::TextBox](#).

9.80.2.32 SetText()

```
void UIElement::SetText (
    std::string a_text = "" ) [virtual]
```

setter function to set the text that is displayed in the button

Warning

It assumes that the font for the text is already set

Parameters

<i>a_text</i>	text to show on the button -> default empty string
---------------	--

Reimplemented in [Brood::BroodUI::DropDownMenu](#), and [Brood::BroodUI::TextBox](#).

9.80.2.33 SetTextPosition()

```
void UIElement::SetTextPosition ( ) [protected]
```

setter funciton to set the position of the text.

Sets the position of the text such that it is always centered

9.80.3 Member Data Documentation

9.80.3.1 m_activeOverlayColor

```
sf::Color Brood::BroodUI::UIElement::m_activeOverlayColor [protected]
```

color when it is active element

9.80.3.2 m_body

```
sf::RectangleShape Brood::BroodUI::UIElement::m_body [protected]
```

element body

9.80.3.3 m_bodyOverLay

```
sf::RectangleShape Brood::BroodUI::UIElement::m_bodyOverLay [protected]
```

element body

9.80.3.4 m_debugTextSave

```
std::string Brood::BroodUI::UIElement::m_debugTextSave [protected]
```

saves the text content before replacing by uiid

9.80.3.5 m_drawOverlay

```
bool Brood::BroodUI::UIElement::m_drawOverlay [protected]
```

true if overlay is to be drawn

9.80.3.6 m_drawText

```
bool Brood::BroodUI::UIElement::m_drawText [protected]
```

is true if text is set and hence draw to screen

9.80.3.7 m_elementId

```
Brood::BroodUI::Id Brood::BroodUI::UIElement::m_elementId [protected]
```

unique id of the element

9.80.3.8 m_elementType

```
Brood::BroodUI::ENUM_UIType Brood::BroodUI::UIElement::m_elementType [protected]
```

stores the type of element stored

9.80.3.9 m_font

```
sf::Font* Brood::BroodUI::UIElement::m_font [protected]
```

element's font

9.80.3.10 m_fontSize

```
unsigned Brood::BroodUI::UIElement::m_fontSize [protected]
```

font size

9.80.3.11 m_hotOverlayColor

```
sf::Color Brood::BroodUI::UIElement::m_hotOverlayColor [protected]
```

color when mouse is over it

9.80.3.12 m_isSelected

```
bool Brood::BroodUI::UIElement::m_isSelected [protected]
```

is true if the element is selected

9.80.3.13 m_text

```
sf::Text Brood::BroodUI::UIElement::m_text [protected]
```

text content

9.80.3.14 m_textContent

```
std::string Brood::BroodUI::UIElement::m_textContent [protected]
```

saves the text content

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UIElement.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UIElement.cpp](#)

9.81 UIElement Class Reference

a abstract base class that is parent to all the different UI elements

```
#include "UIElement.h"
```

9.81.1 Detailed Description

a abstract base class that is parent to all the different UI elements

For the UI element the order of update logic and rending matters.

Update Logic: We first check if we should execute the funciton for the element with DoElement() function which returns a boolean value. Another method for some but not all of the UI Element is to check if the m_isSelected value is true or false. On the same note the order for checking update logic is from top left to bottom right- like reading a english book.

Render Logic: The order for render logic is to go from bottom to top starting form bottom-right to left. This is because of how rendering works where things drawn last is always on top.

Todo create a verticalmenus for ma [UIElement](#). This could be used to for Drop down menus.

create a font class whose job is to load and save font and replace all the m_font to object of that file

Encapsulate the BroodUI better.

create a log class that logs all the errors

change the setter funciton to return bool, errorMessage

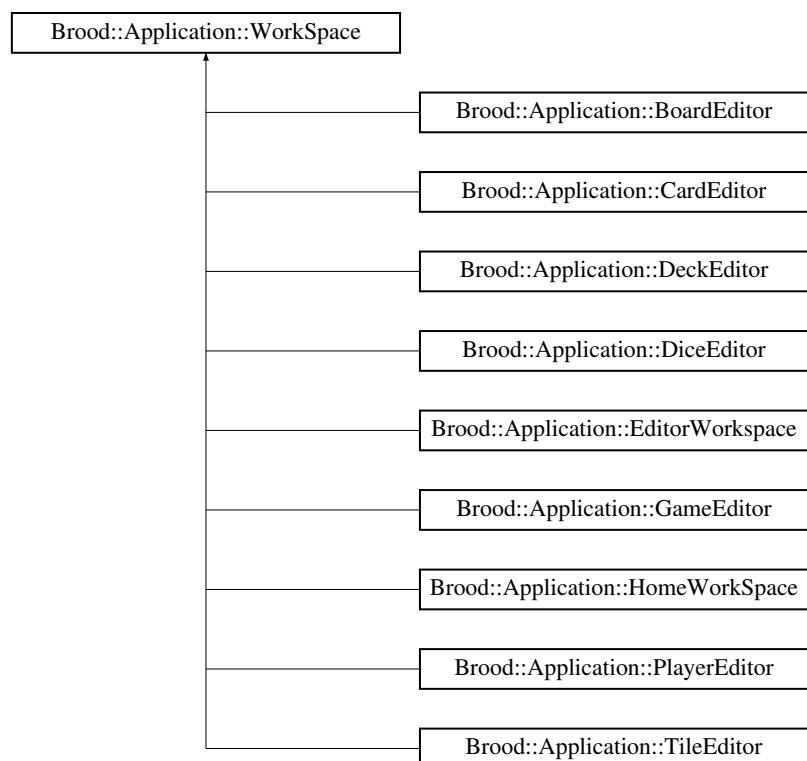
The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[UIElement.h](#)

9.82 Brood::Application::WorkSpace Class Reference

```
#include <WorkSpace.h>
```

Inheritance diagram for Brood::Application::WorkSpace:



Public Member Functions

- `WorkSpace ()`
`default constructor`
- `virtual ~WorkSpace ()`
`default destructor`
- `void SetFont (sf::Font *a_font)`
- `virtual void InitializeWorkSpace ()=0`
- `virtual void Update ()=0`
- `virtual void UpdateAllDisplayElement ()=0`
- `virtual void Draw (sf::RenderWindow &a_window)=0`
`Draw function.`
- `virtual void Debugger ()=0`
`debugger function`
- `virtual void SetGameDataManager (Brood::Application::Components::GameDataManager *aGameDataManagerPtr)`
`setter function for game data`
- `Brood::BroodUI::Button * DyCreateButton (sf::Vector2f a_size, sf::Vector2f a_pos, std::string a_text="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`@TODO: set the editor or game workspace varialbe here`
- `Brood::BroodUI::Button * DyCreateButton (float a_sizeX, float a_sizeY, float a_posX, float a_posY, std::string a_text="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`helper function to dynamically create a button element initializes it, and adds it to m_unNameList`
- `Brood::BroodUI::TextBox * DyCreateTextBox (sf::Vector2f a_size, sf::Vector2f a_pos, std::string a_text="", bool a_isEditable=false, std::string a_placeholderText="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList`
- `Brood::BroodUI::TextBox * DyCreateTextBox (float a_sizeX, float a_sizeY, float a_posX, float a_posY, std::string a_text="", bool a_isEditable=false, std::string a_placeholderText="", sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList`
- `Brood::BroodUI::DropDownInput * DyCreateDropDownInput (sf::Vector2f a_size, sf::Vector2f a_pos, sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`helper function to dynamically create a DropDownList element initializes it, and adds it to m_unNameList`
- `Brood::BroodUI::DropDownInput * DyCreateDropDownInput (float a_sizeX, float a_sizeY, float a_posX, float a_posY, sf::Color a_color=Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor)`
`helper function to dynamically create a DropDownList element initializes it, and adds it to m_unNameList`
- `void DyCreateDropdownInputElement (sf::RectangleShape *a_panelBodyPtr, Brood::BroodUI::TextBox **a_txtPromptPtr, Brood::BroodUI::DropDownInput **a_ddiPtr, std::string a_promptToDisplay, std::vector<std::string > a_itemsToAdd, unsigned a_firstItemLength=0, bool a_createFromTop=false)`
`Dynamically creates a dropdown input Panel Element.`
- `void DyCreateDeclncPannelElement (sf::RectangleShape *a_panelBodyPtr, Brood::BroodUI::TextBox **a_txtPromptPtr, Brood::BroodUI::Button **a_btnDecPtr, Brood::BroodUI::TextBox **a_txtValuePtr, Brood::BroodUI::Button **a_btlnIncPtr, std::string a_promptToDisplay, std::string a_valueToDisplay, bool a_createFromTop=false)`
`Dynamically creates a Panel Element that can increase or decrease a parameter.`
- `void DyCreateFileInputPannelElement (sf::RectangleShape *a_panelBodyPtr, Brood::BroodUI::TextBox **a_txtPromptPtr, Brood::BroodUI::TextBox **a_txtFileInputPtr, Brood::BroodUI::Button **a_btnOpenFilePtr, std::string a_promptToDisplay, std::string a_valueToDisplay, bool a_createFromTop=false)`
`Dynamically creates a Panel Element that user can enter a filename.`
- `void DyCreateTextInputPannelElement (sf::RectangleShape *a_panelBodyPtr, Brood::BroodUI::TextBox **a_txtPromptPtr, Brood::BroodUI::TextBox **a_txtTextInputPtr, std::string a_promptToDisplay, std::string a_valueToDisplay, std::string a_placeHolderText, bool a_createFromTop=false)`
`Dynamically creates a Panel Element that user can enter a text.`

- void `UpdateDeclncSizeX` (`Brood::BroodUI::Button *a_btnDecSizeX, Brood::BroodUI::TextBox *a_txtSizeX, Brood::BroodUI::Button *a_btnIncSizeX, Brood::BroodUI::UIElement *a_elemnetToChangeSizeX, unsigned a_unit, unsigned a_lowerLimit, unsigned a_upperLimit)`
checks if the user interacted with the element SizeX panel
- void `UpdateDeclncSizeY` (`Brood::BroodUI::Button *a_btnDecSizeY, Brood::BroodUI::TextBox *a_txtSizeY, Brood::BroodUI::Button *a_btnIncSizeY, Brood::BroodUI::UIElement *a_elemnetToChangeSizeY, unsigned a_unit, unsigned a_lowerLimit, unsigned a_upperLimit)`
checks if the user interacted with the element SizeY panel
- void `UpdateDeclncPosX` (`Brood::BroodUI::Button *a_btnDecPosX, Brood::BroodUI::TextBox *a_txtPosX, Brood::BroodUI::Button *a_btnIncPosX, Brood::BroodUI::UIElement *a_elemnetToChangePosX, unsigned a_unit, unsigned a_lowerLimit, unsigned a_upperLimit)`
checks if the user interacted with the element PosX panel
- void `UpdateDeclncPosY` (`Brood::BroodUI::Button *a_btnDecPosY, Brood::BroodUI::TextBox *a_txtPosY, Brood::BroodUI::Button *a_btnIncPosY, Brood::BroodUI::UIElement *a_elemnetToChangePosY, unsigned a_unit, unsigned a_lowerLimit, unsigned a_upperLimit)`
checks if the user interacted with the player Y Offset panel

Protected Attributes

- `sf::Font * m_font`
font
- `std::vector< Brood::BroodUI::UIElement * > m_unNamedUIList`
stores list of all UI Elements pointer
- `bool m_isDebugger`
true if debugger is on

Static Protected Attributes

- static `Brood::Application::Components::GameDataManager * mGameData = nullptr`
game data

9.82.1 Constructor & Destructor Documentation

9.82.1.1 WorkSpace()

`WorkSpace::WorkSpace ()`

default constructor

9.82.1.2 ~WorkSpace()

`WorkSpace::~WorkSpace () [virtual]`

default destructor

@virtual

9.82.2 Member Function Documentation

9.82.2.1 Debugger()

```
void WorkSpace::Debugger ( ) [pure virtual]

debugger funciton

@virtual
```

This function helps in debugging the UI elements.

Implemented in [Brood::Application::BoardEditor](#), [Brood::Application::CardEditor](#), [Brood::Application::DeckEditor](#), [Brood::Application::DiceEditor](#), [Brood::Application::EditorWorkspace](#), [Brood::Application::GameEditor](#), [Brood::Application::HomeWork](#), [Brood::Application::PlayerEditor](#), and [Brood::Application::TileEditor](#).

9.82.2.2 Draw()

```
void WorkSpace::Draw (
    sf::RenderWindow & a_window ) [pure virtual]
```

Draw funciton.

@virtual

Draws all the component to the screen

Implemented in [Brood::Application::BoardEditor](#), [Brood::Application::CardEditor](#), [Brood::Application::DeckEditor](#), [Brood::Application::DiceEditor](#), [Brood::Application::EditorWorkspace](#), [Brood::Application::GameEditor](#), [Brood::Application::HomeWork](#), [Brood::Application::PlayerEditor](#), and [Brood::Application::TileEditor](#).

9.82.2.3 DyCreateButton() [1/2]

```
Brood::BroodUI::Button * WorkSpace::DyCreateButton (
    float a_sizeX,
    float a_sizeY,
    float a_posX,
    float a_posY,
    std::string a_text = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
)
```

helper function to dynamically create a button element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_sizeX</i>	length of the button element
<i>a_sizeY</i>	width of the button element
<i>a_posX</i>	x-position of the button element
<i>a_posY</i>	y-position of the button element
<i>a_text</i>	text to display in the button element ; default ""
<i>a_color</i>	color of the button element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created [Button](#) element

9.82.2.4 DyCreateButton() [2/2]

```
Brood::BroodUI::Button * WorkSpace::DyCreateButton (
    sf::Vector2f a_size,
    sf::Vector2f a_pos,
    std::string a_text = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
)
```

@TODO: set the editor or game workspace varialbe here

helper function to dynamically create a button element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_size</i>	size of the button element
<i>a_pos</i>	postion of the button element
<i>a_text</i>	text to display in the button element ; default ""
<i>a_color</i>	color of the button element ; default <i>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</i>

Returns

pointer to dynamically created [Button](#) element

9.82.2.5 DyCreateDecIncPannelElement()

```
void WorkSpace::DyCreateDecIncPannelElement (
    sf::RectangleShape * a_panelBodyPtr,
    Brood::BroodUI::TextBox ** a_txtPromptPtr,
    Brood::BroodUI::Button ** a_btnDecPtr,
    Brood::BroodUI::TextBox ** a_txtValuePtr,
    Brood::BroodUI::Button ** a_btnIncPtr,
    std::string a_promptToDisplay,
    std::string a_valueToDisplay,
    bool a_createFromTop = false )
```

Dynamically creates a Panel Element that can increase or decrease a parameter.

It has 4 UI element and its layout is < textBox prompt> <button '<'> <textbox value> <button '>'>

For example: boardSize X < 20 >.

It alternates the color of the panel element. Color used are: [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppSecondaryColor](#) and [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor](#)

Parameters

<i>a_panelBodyPtr</i>	pointer to panel body.
<i>a_txtPromptPtr</i>	pointer to textbox pointer where the dynamically created textbox to display the prompt is stored
<i>a_btnDecPtr</i>	pointer to button pointer where the dynamically created button to display the '<' is stored
<i>a_txtValuePtr</i>	pointer to textbox pointer where the dynamically created textbox to display the value is stored
<i>a_btnIncPtr</i>	pointer to button pointer where the dynamically created button to display the '>' is stored
<i>a_promptToDisplay</i>	text to display in the prompt textbox element
<i>a_valueToDisplay</i>	text to display in the value textbox element
<i>a_createFromTop</i>	is true if the created element should be positioned second element from the top

9.82.2.6 DyCreateDropDownInput() [1/2]

```
Brood::BroodUI::DropDownInput * WorkSpace::DyCreateDropDownInput (
    float a_sizeX,
    float a_sizeY,
    float a_posX,
    float a_posY,
    sf::Color a_color = Brood::Application::StaticVariables::ST\_ColorVariables::stm\_AppPrimaryColor
)
```

helper function to dynamically create a [DropDownInput](#) element initializes it, and adds it to *m_unNameList*

@overlaoded

Note

assumes that *m_font* is loaded

Parameters

<i>a_sizeX</i>	length of the <code>DropDownInput</code> element
<i>a_sizeY</i>	width of the <code>DropDownInput</code> element
<i>a_posX</i>	x-position of the <code>DropDownInput</code> element
<i>a_posY</i>	y-position of the <code>DropDownInput</code> element
<i>a_color</i>	color of the textbox element ; default <code>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</code>

Returns

pointer to dynamically created textbox element

9.82.2.7 DyCreateDropDownInput() [2/2]

```
Brood::BroodUI::DropDownInput * WorkSpace::DyCreateDropDownInput (
    sf::Vector2f a_size,
    sf::Vector2f a_pos,
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
)
```

helper function to dynamically create a `DropDownInput` element initializes it, and adds it to `m_unNameList`

Note

assumes that `m_font` is loaded

Parameters

<i>a_size</i>	size of the <code>DropDownInput</code> element
<i>a_pos</i>	position of the <code>DropDownInput</code> element
<i>a_color</i>	color of the textbox element ; default <code>Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor</code>

Returns

pointer to dynamically created `DropDownInput` element

9.82.2.8 DyCreateDropdownInputElement()

```
void WorkSpace::DyCreateDropdownInputElement (
    sf::RectangleShape * a_panelBodyPtr,
    Brood::BroodUI::TextBox ** a_txtPromptPtr,
    Brood::BroodUI::DropDownInput ** a_ddiPtr,
```

```
    std::string a_promptToDisplay,
    std::vector< std::string > a_itemsToAdd,
    unsigned a_firstItemLength = 0,
    bool a_createFromTop = false )
```

Dynamically creates a dropdown input Panel Element.

It occupies the whole pannel length.

It can be position at the top of the panel if a_topElement is true, as the second element if a_createFromTop is true, else below the last creasted UI element.

Note

both a_topElement and a_createFromTop can not be true at the same time.

It alternates the color of the panel element. Color used are: [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppSecondaryColor](#) and [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor](#)

Parameters

<i>a_panelBodyPtr</i>	pointer to panel body.
<i>a_txtPromptPtr</i>	pointer to textbox pointer where the dynamically created textbox to display the prompt is stored
<i>a_ddiPtr</i>	pointer to drop down input pointer where the dynamically created drop down input is stored
<i>a_promptToDisplay</i>	text to display in the prompt textbox element
<i>a_itemsToAdd</i>	list of item to add to the drop down input
<i>a_firstItemLength</i>	the length of the string to display on the as the selected element -> default 0 do nothing
<i>a_createFromTop</i>	is true if the created element should be positioned second element form the top -> default false

9.82.2.9 DyCreateFileInputPannelElement()

```
void WorkSpace::DyCreateFileInputPannelElement (
    sf::RectangleShape * a_panelBodyPtr,
    Brood::BroodUI::TextBox ** a_txtPromptPtr,
    Brood::BroodUI::TextBox ** a_txtFileInputPtr,
    Brood::BroodUI::Button ** a_btnOpenFilePtr,
    std::string a_promptToDisplay,
    std::string a_valueToDisplay,
    bool a_createFromTop = false )
```

Dynamically creates a Panel Element that user can enter a filename.

It has 3 UI element and its layout is

```
<textBox prompt>
<textbox>
<open button>
```

For example: [Player](#) texture

```
(enter file name) <open>
```

It alternates the color of the panel element. Color used are: [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppSecondaryColor](#) and [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor](#)

Parameters

<i>a_panelBodyPtr</i>	pointer to panel body.
<i>a_txtPromptPtr</i>	pointer to textbox pointer where the dynamically created textbox to display the prompt is stored
<i>a_txtFileInputPtr</i>	pointer to textbox pointer where the dynamically created textbox to enter file name is stored
<i>a_btnOpenFilePtr</i>	pointer to button pointer where the dynamically created button to display the 'open' is stored
<i>a_promptToDisplay</i>	text to display in the prompt textbox element
<i>a_valueToDisplay</i>	text to display in the value textbox element
<i>a_createFromTop</i>	is true if the created element should be positioned second element form the top

9.82.2.10 DyCreateTextBox() [1/2]

```
Brood::BroodUI::TextBox * WorkSpace::DyCreateTextBox (
    float a_sizeX,
    float a_sizeY,
    float a_posX,
    float a_posY,
    std::string a_text = "",
    bool a_isEditable = false,
    std::string a_placeholderText = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
)
```

helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_sizeX</i>	length of the textbox element
<i>a_sizeY</i>	width of the textbox element
<i>a_posX</i>	x-position of the textbox element
<i>a_posY</i>	y-position of the textbox element
<i>a_text</i>	text to display in the textbox element ; default ""
<i>a_isEditable</i>	true if the text box is editable; default false
<i>a_color</i>	color of the textbox element ; default Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor

Returns

pointer to dynamically created textbox element

9.82.2.11 DyCreateTextBox() [2/2]

```
Brood::BroodUI::TextBox * WorkSpace::DyCreateTextBox (
    sf::Vector2f a_size,
    sf::Vector2f a_pos,
    std::string a_text = "",
    bool a_isEditable = false,
    std::string a_placeholderText = "",
    sf::Color a_color = Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor
)
```

helper function to dynamically create a textbox element initializes it, and adds it to m_unNameList

@overlaoded

Note

assumes that m_font is loaded

Parameters

<i>a_size</i>	size of the textbox element
<i>a_pos</i>	postion of the textbox element
<i>a_text</i>	text to display in the textbox element ; default ""
<i>a_isEditable</i>	true if the text box is editable; default false
<i>a_color</i>	color of the textbox element ; default Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor

Returns

pointer to dynamically created textbox element

9.82.2.12 DyCreateTextInputPanelElement()

```
void WorkSpace::DyCreateTextInputPanelElement (
    sf::RectangleShape * a_panelBodyPtr,
    Brood::BroodUI::TextBox ** a_txtPromptPtr,
    Brood::BroodUI::TextBox ** a_txtTextInputttPtr,
    std::string a_promptToDisplay,
    std::string a_valueToDisplay,
    std::string a_placeHolderText,
    bool a_createFromTop = false )
```

Dynamically creates a Panel Element that user can enter a text.

It has 2 UI element and its layout is

```
<textBox prompt> <textbox>
```

For example: Game Name (enter file name)

It alternates the color of the panel element. Color used are: [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppSecondaryColor](#) and [Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor](#)

Parameters

<i>a_panelBodyPtr</i>	pointer to panel body.
<i>a_txtPromptPtr</i>	pointer to textbox pointer where the dynamically created textbox to display the prompt is stored
<i>a_txtTextInputtPtr</i>	pointer to textbox pointer where the dynamically created textbox to enter the text is stored
<i>a_promptToDisplay</i>	text to display in the prompt textbox element
<i>a_valueToDisplay</i>	text to display in the value textbox element
<i>a_placeHolderText</i>	text to display in the value textbox element
<i>a_createFromTop</i>	is true if the created element should be positioned second element form the top

9.82.2.13 InitializeWorkSpace()

```
virtual void Brood::Application::WorkSpace::InitializeWorkSpace ( ) [pure virtual]
```

Implemented in [Brood::Application::BoardEditor](#), [Brood::Application::CardEditor](#), [Brood::Application::DeckEditor](#), [Brood::Application::DiceEditor](#), [Brood::Application::EditorWorkspace](#), [Brood::Application::GameEditor](#), [Brood::Application::PlayerEditor](#), [Brood::Application::TileEditor](#), and [Brood::Application::HomeWorkSpace](#).

9.82.2.14 SetFont()

```
void Brood::Application::WorkSpace::SetFont (
    sf::Font * a_font )
```

9.82.2.15 SetGameDataManager()

```
void WorkSpace::SetGameDataManager (
    Brood::Application::Components::GameManager * a_gameDataManagerPtr ) [virtual]
```

setter function for game data

@virtual

Reimplemented in [Brood::Application::EditorWorkspace](#).

9.82.2.16 Update()

```
virtual void Brood::Application::WorkSpace::Update ( ) [pure virtual]
```

Implemented in [Brood::Application::BoardEditor](#), [Brood::Application::CardEditor](#), [Brood::Application::DeckEditor](#), [Brood::Application::DiceEditor](#), [Brood::Application::EditorWorkspace](#), [Brood::Application::GameEditor](#), [Brood::Application::HomeWork](#), [Brood::Application::PlayerEditor](#), and [Brood::Application::TileEditor](#).

9.82.2.17 UpdateAllDisplayElement()

```
virtual void Brood::Application::WorkSpace::UpdateAllDisplayElement ( ) [pure virtual]
```

Implemented in [Brood::Application::BoardEditor](#), [Brood::Application::CardEditor](#), [Brood::Application::DeckEditor](#), [Brood::Application::DiceEditor](#), [Brood::Application::EditorWorkspace](#), [Brood::Application::GameEditor](#), [Brood::Application::HomeWork](#), [Brood::Application::PlayerEditor](#), and [Brood::Application::TileEditor](#).

9.82.2.18 UpdateDecIncPosX()

```
void WorkSpace::UpdateDecIncPosX (
    Brood::BroodUI::Button * a_btnDecPosX,
    Brood::BroodUI::TextBox * a_txtPosX,
    Brood::BroodUI::Button * a_btnIncPosX,
    Brood::BroodUI::UIElement * a_elemnetToChangePosX,
    unsigned a_unit,
    unsigned a_lowerLimit,
    unsigned a_upperLimit )
```

checks if the user interacted with the element PosX panel

PosX panel contains current element PosX prompt textbox, current element PosX value text box, button to increase the current element PosX, and button to decrease the element PosX.

Only the button to increase the PosX, and button to decrease PosX are interactable

If the interactable button was pressed then the PosX is increased or decreased by specified units if the current element does not go outside the upper limit

Parameters

<i>a_btnDecPosX</i>	pointer to the element's panel decPosX button
<i>a_txtPosX</i>	pointer to the element's panel PosX value textbox
<i>a_btnIncPosX</i>	pointer to the element's panel incPosX button
<i>a_elemnetToChangePosX</i>	pointer to the element whose size being controlled
<i>a_unit</i>	unit to increase or decrease
<i>a_lowerLimit</i>	the minimum PosX of the element
<i>a_upperLimit</i>	the maximum PosX of the element

9.82.2.19 UpdateDecIncPosY()

```
void WorkSpace::UpdateDecIncPosY (
    Brood::BroodUI::Button * a_btnDecPosY,
    Brood::BroodUI::TextBox * a_txtPosY,
    Brood::BroodUI::Button * a_btnIncPosY,
    Brood::BroodUI::UIElement * a_elemnetToChangePosX,
    unsigned a_unit,
    unsigned a_lowerLimit,
    unsigned a_upperLimit )
```

checks if the user interacted with the player Y Offset panel

Player Y Offset panel contains current player Y Offset prompt textbox, current player Y Offset value text box, button to increase the current player Y Offset, and button to decrease the current player Y Offset.

Only the button to increase the current player Y Offset, and button to decrease the current player Y Offset are interactable

If the interactable button was pressed then the current player Y Offset is increased or decreased by 1 if the current player does not go outside the board window

9.82.2.20 UpdateDecIncSizeX()

```
void WorkSpace::UpdateDecIncSizeX (
    Brood::BroodUI::Button * a_btnDecSizeX,
    Brood::BroodUI::TextBox * a_txtSizeX,
    Brood::BroodUI::Button * a_btnIncSizeX,
    Brood::BroodUI::UIElement * a_elemnetToChangeSizeX,
    unsigned a_unit,
    unsigned a_lowerLimit,
    unsigned a_upperLimit )
```

checks if the user interacted with the element SizeX panel

SizeX panel contains current element SizeX prompt textbox, current element SizeX value text box, button to increase the current element SizeX, and button to decrease the element SizeX.

Only the button to increase the SizeX, and button to decrease SizeX are interactable

If the interactable button was pressed then the SizeX is increased or decreased by specified units if the current element does not go outside the upper limit

Parameters

<i>a_btnDecSizeX</i>	pointer to the element's panel decSizeX button
<i>a_txtSizeX</i>	pointer to the element's panel sizeX value textbox
<i>a_btnIncSizeX</i>	pointer to the element's panel incSizeX button
<i>a_elemnetToChangeSizeX</i>	pointer to the element whose size being controlled
<i>a_unit</i>	unit to increase or decrease
<i>a_lowerLimit</i>	the minimum SizeX of the element
<i>a_upperLimit</i>	the maximum SizeX of the element

9.82.2.21 UpdateDecIncSizeY()

```
void WorkSpace::UpdateDecIncSizeY (
    Brood::BroodUI::Button * a_btnDecSizeY,
    Brood::BroodUI::TextBox * a_txtSizeY,
    Brood::BroodUI::Button * a_btnIncSizeY,
    Brood::BroodUI::UIElement * a_elemnetToChangeSizeY,
    unsigned a_unit,
    unsigned a_lowerLimit,
    unsigned a_upperLimit )
```

checks if the user interacted with the element SizeY panel

SizeY panel contains current element SizeY prompt textbox, current element SizeY value text box, button to increase the current element SizeY, and button to decrease the element SizeY.

Only the button to increase the SizeY, and button to decrease SizeY are interactable

If the interactable button was pressed then the SizeY is increased or decreased by specified units if the current element does not go outside the upper limit

Parameters

<i>a_btnDecSizeY</i>	pointer to the element's panel decSizeY button
<i>a_txtSizeY</i>	pointer to the element's panel sizeY value textbox
<i>a_btnIncSizeY</i>	pointer to the element's panel incSizeY button
<i>a_elemnetToChangeSizeY</i>	pointer to the element whose size being controlled
<i>a_unit</i>	unit to increase or decrease
<i>a_lowerLimit</i>	the minimum SizeY of the element
<i>a_upperLimit</i>	the maximum SizeY of the element

9.82.3 Member Data Documentation

9.82.3.1 m_font

```
sf::Font* Brood::Application::WorkSpace::m_font [protected]  
font
```

9.82.3.2 mGameData

```
Brood::Application::Components::GameManager * WorkSpace::mGameData = nullptr [static],  
[protected]  
game data
```

9.82.3.3 m_isDebugger

```
bool Brood::Application::WorkSpace::m_isDebugger [protected]
```

true if debugger is on

9.82.3.4 m_unNamedUIList

```
std::vector<Brood::BroodUI::UIElement*> Brood::Application::WorkSpace::m_unNamedUIList [protected]
```

stores list of all UI Elements pointer

The documentation for this class was generated from the following files:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[WorkSpace.h](#)
- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[WorkSpace.cpp](#)

9.83 WorkSpace Class Reference

a abstract base class that is parent to all the different workspace.

```
#include "WorkSpace.h"
```

9.83.1 Detailed Description

a abstract base class that is parent to all the different workspace.

The documentation for this class was generated from the following file:

- C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/[WorkSpace.h](#)

Chapter 10

File Documentation

10.1 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Application.cpp File Reference

This file is a source file for Application class.

```
#include "stdafx.h"
#include "Application.h"
```

10.1.1 Detailed Description

This file is a source file for Application class.

It contains all of the defination of the member funciton of Application class.

10.2 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Application.h File Reference

This file is a header file for Application class.

```
#include <filesystem>
#include "Board.h"
#include "Dice.h"
#include "Player.h"
#include "CardInfo.h"
#include "MouseHandler.h"
#include "TextBox.h"
#include "MenuBar.h"
#include "DropDownInput.h"
#include "HomeWorkSpace.h"
#include "Path.h"
#include "GlobalVariables.h"
```

Classes

- class **Brood::Application::Application**
Applicaiton class provides the interface for the applicaiton.

Namespaces

- namespace **Brood**
- namespace **Brood::Application**
TODO initialize all of the data here to match the default values.

10.2.1 Detailed Description

This file is a header file for Application class.

It contains all of the declaration of the member funciton of Application class.

10.3 Application.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // ****
11
12 // =====
13 // ===== included files =====
14 // =====
15 #pragma once
16 #include <filesystem>
17
18 #include "Board.h"
19 #include "Dice.h"
20 #include "Player.h"
21 #include "CardInfo.h"
22 #include "MouseHandler.h"
23
24 #include "TextBox.h"
25 #include "MenuBar.h"
26 #include "DropDownInput.h"
27
28 #include "HomeWorkSpace.h"
29
30 #include "Path.h"
31 #include "GlobalVariables.h"
32
33 // =====
34 // ===== defining namespace =====
35 // =====
36 namespace Brood
37 {
38     namespace Application
39     {
40         class Application;
41     }
42 }
43
44 // =====
45 // ===== end of namespace defintion =====
46 // =====
47
48 // ===== start of Application class =====
49 // =====
50
51 // =====
52 // ===== start of Application class =====
53 // =====
54
55 class Brood::Application::Application
56
57 {
58     // ===== public member function =====
59 public:
60     Application(); // default constructor

```

```
65     ~Application(); // default destructor
66
67     void RunApplicaiton(); // main loop of the program
68
69 protected:
70     // ===== private member function =====
71 private:
72
74     void PollEvents();
75
77     void Draw();
78
79     void CreateWorkSpace();
80
82     void Debugger();
83
84     // ===== private member variables =====
85 private:
86
88     sf::RenderWindow m_window;
89
91     sf::Event m_events;
92
94     Brood::Application::HomeWorkSpace m_mainWorkspace;
95
97     Brood::Application::Components::Board myBoard;
98     Brood::Application::Components::Dice myDice;
99
100 };
101
102 // =====
103 // ===== end of button class =====
104 // =====
```

10.4 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Board.cpp File Reference

This file is a source file for [Board](#) class.

```
#include "stdafx.h"
#include "Board.h"
```

10.4.1 Detailed Description

This file is a source file for [Board](#) class.

It contains all of the defination of the member funciton of [Board](#) class.

10.5 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Board.h File Reference

This file is a header file for [Board](#) class.

```
#include "GlobalVariables.h"
#include "DeckManager.h"
#include "Path.h"
#include "Data.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::Board](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.5.1 Detailed Description

This file is a header file for [Board](#) class.

It contains all of the declaration of the member function of [Board](#) class. It contains all of the inline functions definition of the member function of [Board](#) class

10.6 Board.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // **** included files ****
3 // ****
4 // =====
5 // ===== defining namespace =====
6 // =====
7
8 #pragma once
9 #include "GlobalVariables.h"
10 #include "DeckManager.h"
11 #include "Path.h"
12 #include "Data.h"
13 #include "FileAccess.h"
14
15 // =====
16 // ===== end of namespace defination =====
17
18 namespace Brood::Application
19 {
20     namespace Components
21     {
22         class Board;
23         class Path;
24     }
25
26 // =====
27 // ===== start of Board class =====
28
29 class Brood::Application::Components::Board
30 {
31     // ===== public member function =====
32     public:
33
34     // default constructor
35     Board( unsigned a_numRows = 1, unsigned a_numCols = 1,
36             float a_boardSizeX = 50.f, float a_boardSizeY = 50.f,
37             float a_boardPosX = 50.f, float a_boardPosY = 50.f );
38
39     // Destructor
40     ~Board();
41
42     void Draw( sf::RenderWindow& a_window );
43 }
```

```

67     // initialize the board
68     void InitializeBoard( unsigned a_numRows = 0, unsigned a_numCols = 0,
69                           float a_boardSizeX = 0.f, float a_boardSizeY = 0.f,
70                           float a_boardPosX = 0.f, float a_boardPosY = 0.f );
71
72     // initialize the board
73     void InitializeBoard(Brood::Application::Data::ST_BoardData& a_boardData);
74
75     // returns the baord data to save
76     Brood::Application::Data::ST_BoardData GetDataToSave();
77
78     // appends the the baord data to passed file
79     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
80
81     // appends the the baord data to passed file
82     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr,
83                           Brood::Application::Components::DeckManager* a_deckMangerPtr );
84
85     // ===== setter funcitons =====
86
87     // setter function to set the nubmer of rows in the board
88     void SetNumRow( unsigned a_numRows, Brood::Application::Components::Deck* a_deckPtr );
89
90     // setter function to set the nubmer of coulumns in the board
91     void SetNumCol( unsigned a_numCols, Brood::Application::Components::Deck* a_deckPtr );
92
93     // setter function to set the board size
94     void SetBoardSize( sf::Vector2f a_boardSize );
95
96     // setter function to set the board size
97     void SetBoardSize( float a_boardSizeX, float a_boardSizeY );
98
99     // setter function to set the board position
100    void SetBoardPos( sf::Vector2f a_boardPos );
101
102    // setter function to set the board position
103    void SetBoardPos( float a_boardPosX, float a_boardPosY );
104
105    // getter function to get the currentActiveTile
106    void SetCurrentActiveTilePtr( Brood::Application::Components::Path* a_newActivePathPtr );
107
108    // ===== getter funcitons =====
109
110    // getter function to get the nubmer of rows in the board
111    const unsigned GetNumRow() const;
112
113    // getter function to get the nubmer of coulumns in the board
114    const unsigned GetNumCol() const;
115
116    // getter function to get the board size
117    const sf::Vector2f GetBoardSize() const;
118
119    // getter function to get the board position
120    const sf::Vector2f GetBoardPos() const;
121
122    // getter function to get the board path list
123    const std::vector<std::vector<Brood::Application::Components::Path*>> GetBoardPathList() const;
124
125    // getter function to get the currentActive path
126    Brood::Application::Components::Path* GetCurrentActivePath();
127
128    // toggles the draw line for all the paths
129    void ToggleDrawLine();
130
131    void Debugger(); // debugger
132
133    //Brood::Dice& GetDice();
134    //void Update();
135    //void PlayerRollAndMove();
136    //void AddPlayer( float a_playerSizeX = 0.f, float a_playerSizeY = 0.f, int a_playerStartRow = 0,
137    int a_playerStartCol = 0 );
138
139
140    // ===== private member functions =====
141 private:
142
143    // increases the number of row and populates it
144    void IncreaseNumRow( unsigned a_numRows, Brood::Application::Components::Deck* a_deckPtr );
145
146    // decrease the number of row
147    void DecreaseNumRow( unsigned a_numRows );
148
149    // increases the number of col and populates it
150    void IncreaseNumCol( unsigned a_numCols, Brood::Application::Components::Deck* a_deckPtr );
151
152    // decrease the number of col

```

```

153     void DecreaseNumCol( unsigned a_numCols );
154
155     // updates the tile in board by using the member variables
156     void UpdateBoardPath( unsigned a_rowBegin, unsigned a_rowEnd,
157                           unsigned a_colBegin, unsigned a_colEnd,
158                           bool a_createNew = false,
159                           Brood::Application::Components::Deck* a_deckPtr = nullptr);
160
161     // ===== private member variables =====
162 private:
163
165     sf::RectangleShape m_boardBody;
166
168     unsigned m_numRows;
169
171     unsigned m_numCols;
172
174     std::vector<std::vector<Brood::Application::Components::Path*>> m_boardPaths;
175
178     Brood::Application::Components::Path* m_currActivePathPtr;
179 };
180
181 // =====
182 // ===== end of Brood class =====
183 // =====
184

```

10.7 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Brood← Maker/BoardEditor.cpp File Reference

This file is a source file for [BoardEditor](#) class.

```
#include "stdafhx.h"
#include "BoardEditor.h"
```

10.7.1 Detailed Description

This file is a source file for [BoardEditor](#) class.

It contains all of the defination of the member funciton of [BoardEditor](#) class.

10.8 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Brood← Maker/BoardEditor.h File Reference

This file is a header file for [BoardEditor](#) class.

```
#include "MenuBar.h"
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::BoardEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.8.1 Detailed Description

This file is a header file for [BoardEditor](#) class.

It contains all of the declaration of the member function of [BoardEditor](#) class.

10.9 BoardEditor.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ===== included files =====
3 // ===== defining namespace =====
4 // =====
5 #pragma once
6 #include "MenuBar.h" // includes all the other uiElements
7 #include "WorkSpace.h"
8
9 // =====
10 // ===== defining namespace =====
11 // =====
12 // =====
13 // ===== end of namespace defination =====
14 // =====
15 // =====
16 // ===== start of BoardEditor class =====
17 // =====
18 // =====
19 // =====
20 // =====
21 // =====
22 namespace Brood
23 {
24     namespace Application
25     {
26         class BoardEditor;
27     }
28 }
29 // =====
30 // ===== end of namespace defination =====
31 // =====
32 // =====
33 // =====
34 // ===== start of BoardEditor class =====
35 // =====
36 // =====
37
38 class Brood::Application::BoardEditor :
39     public Brood::Application::WorkSpace
40 {
41     // ===== public member function =====
42 public:
43     // default constructor
44     BoardEditor( Brood::Application::Components::GameDataManager* aGameData,
45                 sf::RectangleShape* a_panelPtr );
46
47     // default destructor
48     ~BoardEditor();
49     // ===== virtual funcitons ====
50
51     // initializes the component of a workspace
52     void InitializeWorkSpace();
53
54     // all the logic for the workspace is here
55     void Update();
56
57     // updates all the display elment
58     void UpdateAllDisplayElement();
59
60     // renders to the screen
61     void Draw( sf::RenderWindow& a_window );
62
63     // Calls Debugger for all UI elements in this class
64     void Debugger();
65
66     // =====
67     // ===== update for General Board setting =====
68 }
```

```
84 // =====
85 // checks if the user interacted with the x size
86 // panel Element
87 void UpdateBoardXSizePanelElement();
88
89 // checks if the user interacted with the y size
90 // panel Element
91 void UpdateBoardYSizePanelElement();
92
93 // checks if the user interacted with the x position
94 // panel Element
95 void UpdateBoardXPosPanelElement();
96
97 // checks if the user interacted with the y position
98 // panel Element
99 void UpdateBoardYPosPanelElement();
100
101 // checks if the user interacted with the row
102 // panel Element
103 void UpdateBoardRowPanelElement();
104
105 // checks if the user interacted with the column
106 // panel Element
107 void UpdateBoardColPanelElement();
108
109 // ===== private data member =====
110
111 private:
112
114 sf::RectangleShape* m_panelBodyPtr;
115 // =====
116 // ===== member variables for General board setting =====
117 // =====
118
119 // === UI elements to show setting title ===
120 Brood::BroodUI::TextBox* m_txtSettingTitle;
121
122 // === UI elements to change the board's x size ===
123
125 Brood::BroodUI::TextBox* m_txtBoardSizePromtX;
126
128 Brood::BroodUI::Button* m_btnBoardSizeDecX;
129
131 Brood::BroodUI::TextBox* m_txtBoardSizeX;
132
134 Brood::BroodUI::Button* m_btnBoardSizeIncX;
135
136 // === UI elements to change the board's y size ===
137
139 Brood::BroodUI::TextBox* m_txtBoardSizePromtY;
140
142 Brood::BroodUI::Button* m_btnBoardSizeDecY;
143
145 Brood::BroodUI::TextBox* m_txtBoardSizeY;
146
148 Brood::BroodUI::Button* m_btnBoardSizeIncY;
149
150 // === UI elements to change the board's x position ===
151
153 Brood::BroodUI::TextBox* m_txtBoardPosPromtX;
154
156 Brood::BroodUI::Button* m_btnBoardPosDecX;
157
159 Brood::BroodUI::TextBox* m_txtBoardPosX;
160
162 Brood::BroodUI::Button* m_btnBoardPosIncX;
163
164 // === UI elements to change the board's y position ===
165
167 Brood::BroodUI::TextBox* m_txtBoardPosPromtY;
168
170 Brood::BroodUI::Button* m_btnBoardPosDecY;
171
173 Brood::BroodUI::TextBox* m_txtBoardPosY;
174
176 Brood::BroodUI::Button* m_btnBoardPosIncY;
177
178 // === UI elements to change the board's Row ===
179
181 Brood::BroodUI::TextBox* m_txtBoardRowPromt;
182
184 Brood::BroodUI::Button* m_btnBoardDecRow;
185
187 Brood::BroodUI::TextBox* m_txtBoardRow;
188
190 Brood::BroodUI::Button* m_btnBoardIncRow;
191
```

```

192     // ===== UI elements to change the board's Column =====
193     Brood::BroodUI::TextBox* m_txtBoardColPromt;
194
195     Brood::BroodUI::Button* m_btnBoardDecCol;
196
197     Brood::BroodUI::TextBox* m_txtBoardCol;
198
199     Brood::BroodUI::Button* m_btnBoardIncCol;
200 }
201
202
203 // =====
204 // ===== end of BoardEditor class =====
205 // =====

```

10.10 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Button.cpp File Reference

This file is a source file for [Button](#) class.

```
#include "stdafx.h"
#include "Button.h"
```

10.10.1 Detailed Description

This file is a source file for [Button](#) class.

It contains all of the defination of the member funciton of [Button](#) class.

10.11 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/Button.h File Reference

This file is a header file for [Button](#) class which is in BroodUI namespace.

```
#include "MouseHandler.h"
#include "UtilityFunctions.h"
#include "SpriteHandler.h"
#include "UIElement.h"
```

Classes

- class [Brood::BroodUI::Button](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.11.1 Detailed Description

This file is a header file for [Button](#) class which is in BroodUI namespace.

It contains all of the declaration of the member funciton of [Button](#) class.

10.12 Button.h

[Go to the documentation of this file.](#)

```

1 // ****
11 // =====
12 // ===== included files =====
15 // =====
16 #pragma once
17 #include "MouseHandler.h"
18 #include "UtilityFunctions.h"
19 #include "SpriteHandler.h"
20 #include "UIElement.h"
21
22 // =====
23 // ===== defining namespace =====
24 // =====
25 namespace Brood
26 {
27     namespace BroodUI
28     {
29         class Button;
30     }
31 }
32 // =====
33 // ===== end of namespace defination =====
34 // =====
35
36 // =====
37 // ===== start of button class =====
38 // =====
39
90 class Brood::BroodUI::Button :
91     public Brood::BroodUI::UIElement
92 {
93     // ===== public member function =====
94 public:
95
96     // default constructor
97     Button( Brood::BroodUI::UIElement* a_parentPtr = nullptr,
98             Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_button );
99
100    // default destructor
101    virtual ~Button();
102
103    // copy constructor
104    Button( const Button& a_otherElement );
105
106    // assignment operator
107    Brood::BroodUI::Button& operator=( const Button& a_otherButton );
108
109    // Dynamically create a button, initializes it, and adds
110    // it to m_unNameList
111    static Brood::BroodUI::Button* DyCreateButton( sf::Vector2f a_size,
112                                                 sf::Vector2f a_pos,
113                                                 std::string a_text = "",
114                                                 sf::Color a_color =
115                                                 Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
116
117    static Brood::BroodUI::Button* DyCreateButton( float a_sizeX, float a_sizeY,
118                                                 float a_posX, float a_posY,
119                                                 std::string a_text = "",
120                                                 sf::Color a_color =
121                                                 Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
122
123    // ===== Getter funciton =====
124    // gets the sprite body
125    Brood::SpriteHandler& GetSpriteBody();
126
127    // ===== Setter funciton =====

```

```
127     // sets the state of the element.
128     void SetSelected( bool a_selected );
129
130
131     // checks to see if the logic for the element is to
132     // be executed or not
133     virtual bool DoElement() override;
134
135     // adds the element ID to the text
136     virtual void Debugger();
137
138     // ===== private member variables =====
139 protected:
140     Brood::SpriteHandler m_bodySprite;
141 };
142
143 // ===== end of button class =====
144 // =====
145
146 // =====
```

10.13 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/CardEditor.cpp File Reference

This file is a source file for [CardEditor](#) class.

```
#include "stdafx.h"
#include "CardEditor.h"
```

10.13.1 Detailed Description

This file is a source file for [CardEditor](#) class.

It contains all of the definition of the member function of [CardEditor](#) class.

10.14 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/CardEditor.h File Reference

This file is a header file for [CardEditor](#) class.

```
#include "MenuBar.h"
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::CardEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.14.1 Detailed Description

This file is a header file for [CardEditor](#) class.

It contains all of the declaration of the member funciton of [CardEditor](#) class.

10.15 CardEditor.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ===== included files =====
3 // =====
4 // =====
5 #pragma once
6 #include "MenuBar.h" // includes all the other uiElements
7 #include "WorkSpace.h"
8
9 // ===== defining namespace =====
10 // =====
11 // =====
12 // =====
13 // ===== end of namespace defination =====
14 // =====
15 // =====
16 // ===== start of CardEditor class =====
17 // =====
18 // =====
19 // =====
20 // =====
21 // =====
22 namespace Brood
23 {
24     namespace Application
25     {
26         class CardEditor;
27     }
28 }
29 // =====
30 // ===== end of namespace defination =====
31 // =====
32 // =====
33 // =====
34 // ===== start of CardEditor class =====
35 // =====
36 // =====
37 class Brood::Application::CardEditor :
38     public Brood::Application::WorkSpace
39 {
40     // ===== public member function =====
41 public:
42     // default constructor
43     CardEditor( Brood::Application::Components::GameDataManager* aGameData,
44                 sf::RectangleShape* a_panelPtr );
45
46     // default destructor
47     ~CardEditor();
48     // ===== virtual funcitons ====
49
50     // initializes the component of a workspace
51     void InitializeWorkSpace();
52
53     // all the logic for the workspace is here
54     void Update();
55
56     // updates all the display elment
57     void UpdateAllDisplayElement();
58
59     // reners to the screen
60     void Draw( sf::RenderWindow& a_window );
61
62     // Calls Debugger for all UI elements in this class
63     void Debugger();
64
65     // ===== private member function =====
66     // =====
67     // =====
68     // =====
69     // =====
70     // =====
71     // =====
72     // =====
73     // =====
74     // =====
75     // =====
76 private:

```

```
77 // initializes the setting DDM
78 void InitializeSettingDDI();
80
81 // updates the setting DDM
82 void UpdateSettingSelectionDDI();
83
84 // initializes the card display panel element
85 void InitializeCardDisplayPanel();
86
87 // Update the card display panel element
88 void UpdateCardDisplayPanel();
89
90 // Draw the card display panel element
91 void DrawCardDisplayPanel( sf::RenderWindow& a_window );
92
93 // initializes the card display front component panel element
94 void InitializeCardDisplayFcompPanel();
95
96 // Update the card display front component panel element
97 void UpdateCardDisplayFcompPanel();
98
99 // Draw the card display front component panel element
100 void DrawCardDisplayFcompPanel( sf::RenderWindow& a_window );
101
102 // initializes the card display back component panel element
103 void InitializeCardDisplayBcompPanel();
104
105 // Update the card display back component panel element
106 void UpdateCardDisplayBcompPanel();
107
108 // Draw the card display back component panel element
109 void DrawCardDisplayBcompPanel( sf::RenderWindow& a_window );
110
111 // initializes the card info panel element
112 void InitializeCardInfoPanel();
113
114 // Update the card info panel element
115 void UpdateCardInfoPanel();
116
117 // Draw the card info panel element
118 void DrawCardInfoPanel( sf::RenderWindow& a_window );
119
120
121 // =====
122 // ===== update functions for card display setting =====
123 // =====
124
125
126 // checks if the user interacted with the load
127 // Card texture panel Element
128 void UpdateCardFrontFileDialog();
129
130 // checks if the user interacted with the load
131 // Card texture panel Element
132 void UpdateCardBackFileDialog();
133
134 // checks if the user interacted with the
135 // current selected deck index panel
136 void UpdateCurrSelectedDeckIdx();
137
138 // checks if the user interacted with the
139 // current selected Card index panel
140 void UpdateCurrSelectedCardIdx();
141
142 // checks if the user interacted with the
143 // time panel
144 void UpdateCurrCardInfoTime();
145
146 // checks if the user interacted with the
147 // Up panel
148 void UpdateCurrCardInfoUp();
149
150 // checks if the user interacted with the
151 // Down panel
152 void UpdateCurrCardInfoDown();
153
154 // checks if the user interacted with the
155 // enter curr card question panel element
156 void UpdateCurrCardQuestionInput();
157
158 // checks if the user interacted with the
159 // enter curr card answer panel element
160 void UpdateCurrCardAnswerInput();
161
162 // checks if the user interacted with the
163 // turn card panel element
```

```
164     void UpdateTurnCard();
165
166     // ===== private member variable =====
167 private:
168
169     sf::RectangleShape* m_panelBodyPtr;
170
171     // holds which setting it is displaying
172     unsigned m_selectedSettingIdx;
173
174     Brood::BroodUI::DropDownInput* m_ddiSettingSelection;
175
176     // ===== member variables for card display setting =====
177
178     // ===== UI elements to change CardFrontTimePrompt size x ====
179
180     Brood::BroodUI::TextBox* m_txtCardSizeXPrompt;
181
182     Brood::BroodUI::Button* m_btnCardDecSizeX;
183
184     Brood::BroodUI::TextBox* m_txtCardSizeX;
185
186     Brood::BroodUI::Button* m_btnCardIncSizeX;
187
188     // ===== UI elements to change Card size y ====
189
190     Brood::BroodUI::TextBox* m_txtCardSizeYPrompt;
191
192     Brood::BroodUI::Button* m_btnCardDecSizeY;
193
194     Brood::BroodUI::TextBox* m_txtCardSizeY;
195
196     Brood::BroodUI::Button* m_btnCardIncSizeY;
197
198     // ===== UI elements to change Card position x ====
199
200     Brood::BroodUI::TextBox* m_txtCardPosXPrompt;
201
202     Brood::BroodUI::Button* m_btnCardDecPosX;
203
204     Brood::BroodUI::TextBox* m_txtCardPosX;
205
206     Brood::BroodUI::Button* m_btnCardIncPosX;
207
208     // ===== UI elements to change Card Position y ====
209
210     Brood::BroodUI::TextBox* m_txtCardPosYPrompt;
211
212     Brood::BroodUI::Button* m_btnCardDecPosY;
213
214     Brood::BroodUI::TextBox* m_txtCardPosY;
215
216     Brood::BroodUI::Button* m_btnCardIncPosY;
217
218     // ===== UI elements to enter a CardFrontBg filename texture ====
219
220     Brood::BroodUI::TextBox* m_txtCardFrontBgFileNamePrompt;
221
222     Brood::BroodUI::TextBox* m_txtCardFrontBgFileInput;
223
224     Brood::BroodUI::Button* m_btnCardFrontBgOpenFile;
225
226     // ===== UI elements to enter a CardBrontBg filename textur ====
227
228     Brood::BroodUI::TextBox* m_txtCardBackBgFileNamePrompt;
229
230     Brood::BroodUI::TextBox* m_txtCardBackBgFileInput;
231
232     Brood::BroodUI::Button* m_btnCardBackBgOpenFile;
233
234     // ===== member variables for front of the card =====
235
236     // ===== UI elements to change CardFrontTimePrompt position x ====
237
238     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptPosXPrompt;
239
240     Brood::BroodUI::Button* m_btnCardFrontTimePromptDecPosX;
241
242     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptPosX;
243
244     Brood::BroodUI::Button* m_btnCardFrontTimePromptIncPosX;
245
246     // ===== UI elements to change CardFrontTimePrompt Position y ====
247
248     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptPosYPrompt;
```

```
283     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptPosYPrompt;
284
285     Brood::BroodUI::Button* m_btnCardFrontTimePromptDecPosY;
286
287     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptPosY;
288
289     Brood::BroodUI::Button* m_btnCardFrontTimePromptIncPosY;
290
291 // ===== UI elements to change CardFrontTimeValue position x =====
292
293     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosXPrompt;
294
295     Brood::BroodUI::Button* m_btnCardFrontTimeValueDecPosX;
296
297     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosX;
298
299     Brood::BroodUI::Button* m_btnCardFrontTimeValueIncPosX;
300
301 // ===== UI elements to change CardFrontTimeValue Position y =====
302
303     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosYPrompt;
304
305     Brood::BroodUI::Button* m_btnCardFrontTimeValueDecPosY;
306
307     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosY;
308
309     Brood::BroodUI::Button* m_btnCardFrontTimeValueIncPosY;
310
311 // ===== UI elements to change CardFrontTimeValue Position y =====
312
313     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosYPrompt;
314
315     Brood::BroodUI::Button* m_btnCardFrontTimeValueDecPosY;
316
317     Brood::BroodUI::TextBox* m_txtCardFrontTimeValuePosY;
318
319     Brood::BroodUI::Button* m_btnCardFrontTimeValueIncPosY;
320
321 // ===== UI elements to change CardUpPrompt position x =====
322
323     Brood::BroodUI::TextBox* m_txtCardUpPromptPosXPrompt;
324
325     Brood::BroodUI::Button* m_btnCardUpPromptDecPosX;
326
327     Brood::BroodUI::TextBox* m_txtCardUpPromptPosX;
328
329     Brood::BroodUI::Button* m_btnCardUpPromptIncPosX;
330
331 // ===== UI elements to change CardUpPrompt Position y =====
332
333     Brood::BroodUI::TextBox* m_txtCardUpPromptPosYPrompt;
334
335     Brood::BroodUI::Button* m_btnCardUpPromptDecPosY;
336
337     Brood::BroodUI::TextBox* m_txtCardUpPromptPosY;
338
339     Brood::BroodUI::Button* m_btnCardUpPromptIncPosY;
340
341 // ===== UI elements to change CardUpValue position x =====
342
343     Brood::BroodUI::TextBox* m_txtCardUpValuePosXPrompt;
344
345     Brood::BroodUI::Button* m_btnCardUpValueDecPosX;
346
347     Brood::BroodUI::TextBox* m_txtCardUpValuePosX;
348
349     Brood::BroodUI::Button* m_btnCardUpValueIncPosX;
350
351 // ===== UI elements to change CardUpValue position y =====
352
353     Brood::BroodUI::TextBox* m_txtCardUpValuePosYPrompt;
354
355     Brood::BroodUI::Button* m_btnCardUpValueDecPosY;
356
357     Brood::BroodUI::TextBox* m_txtCardUpValuePosY;
358
359     Brood::BroodUI::Button* m_btnCardUpValueIncPosY;
360
361 // ===== UI elements to change CardUpValue Position y =====
362
363     Brood::BroodUI::TextBox* m_txtCardUpValuePosYPrompt;
364
365     Brood::BroodUI::Button* m_btnCardUpValueDecPosY;
366
367     Brood::BroodUI::TextBox* m_txtCardUpValuePosY;
368
369     Brood::BroodUI::Button* m_btnCardUpValueIncPosY;
370
371 // ===== UI elements to change CardDownPrompt position x =====
372
373     Brood::BroodUI::TextBox* m_txtCardDownPromptPosXPrompt;
374
375     Brood::BroodUI::Button* m_btnCardDownPromptDecPosX;
376
377     Brood::BroodUI::TextBox* m_txtCardDownPromptPosX;
378
379     Brood::BroodUI::Button* m_btnCardDownPromptIncPosX;
380
381 // ===== UI elements to change CardDownPrompt Position y =====
382
383     Brood::BroodUI::TextBox* m_txtCardDownPromptPosYPrompt;
384
385     Brood::BroodUI::Button* m_btnCardDownPromptDecPosY;
386
387     Brood::BroodUI::TextBox* m_txtCardDownPromptPosY;
388
389     Brood::BroodUI::Button* m_btnCardDownPromptIncPosY;
390
391 // ===== UI elements to change CardDownPrompt Position y =====
392
393     Brood::BroodUI::TextBox* m_txtCardDownPromptPosYPrompt;
394
395     Brood::BroodUI::Button* m_btnCardDownPromptDecPosY;
396
397     Brood::BroodUI::TextBox* m_txtCardDownPromptPosY;
```

```
406     Brood::BroodUI::Button* m_btnCardDownPromptIncPosY;
407
408 // ===== UI elements to change CardDownValue position x =====
409
410     Brood::BroodUI::TextBox* m_txtCardDownValuePosXPrompt;
411
412     Brood::BroodUI::Button* m_btnCardDownValueDecPosX;
413
414     Brood::BroodUI::TextBox* m_txtCardDownValuePosX;
415
416     Brood::BroodUI::Button* m_btnCardDownValueIncPosX;
417
418 // ===== UI elements to change CardDownValue Position y =====
419
420     Brood::BroodUI::TextBox* m_txtCardDownValuePosYPrompt;
421
422     Brood::BroodUI::Button* m_btnCardDownValueDecPosY;
423
424     Brood::BroodUI::TextBox* m_txtCardDownValuePosY;
425
426     Brood::BroodUI::Button* m_btnCardDownValueIncPosY;
427
428 // ===== UI elements to change CardTurnCard position x =====
429
430     Brood::BroodUI::TextBox* m_txtCardTurnCardPosXPrompt;
431
432     Brood::BroodUI::Button* m_btnCardTurnCardDecPosX;
433
434     Brood::BroodUI::TextBox* m_txtCardTurnCardPosX;
435
436     Brood::BroodUI::Button* m_btnCardTurnCardIncPosX;
437
438 // ===== UI elements to change CardTurnCard Position y =====
439
440     Brood::BroodUI::TextBox* m_txtCardTurnCardPosYPrompt;
441
442     Brood::BroodUI::Button* m_btnCardTurnCardDecPosY;
443
444     Brood::BroodUI::TextBox* m_txtCardTurnCardPosY;
445
446     Brood::BroodUI::Button* m_btnCardTurnCardIncPosY;
447
448 // ===== UI elements to change CardFrontTimePrompt size x =====
449
450     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptSizeXPrompt;
451
452     Brood::BroodUI::Button* m_btnCardFrontTimePromptDecSizeX;
453
454     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptSizeX;
455
456     Brood::BroodUI::Button* m_btnCardFrontTimePromptIncSizeX;
457
458 // ===== UI elements to change CardFrontTimePrompt size y =====
459
460     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptSizeYPrompt;
461
462     Brood::BroodUI::Button* m_btnCardFrontTimePromptDecSizeY;
463
464     Brood::BroodUI::TextBox* m_txtCardFrontTimePromptSizeY;
465
466     Brood::BroodUI::Button* m_btnCardFrontTimePromptIncSizeY;
467
468 // ===== UI elements to change CardFrontTimeValue size x =====
469
470     Brood::BroodUI::TextBox* m_txtCardFrontTimeValueSizeXPrompt;
471
472     Brood::BroodUI::Button* m_btnCardFrontTimeValueDecSizeX;
473
474     Brood::BroodUI::TextBox* m_txtCardFrontTimeValueSizeX;
475
476     Brood::BroodUI::Button* m_btnCardFrontTimeValueIncSizeX;
477
478 // ===== UI elements to change CardFrontTimeValue size y =====
479
480     Brood::BroodUI::TextBox* m_txtCardFrontTimeValueSizeYPrompt;
481
482     Brood::BroodUI::Button* m_btnCardFrontTimeValueDecSizeY;
483
484     Brood::BroodUI::TextBox* m_txtCardFrontTimeValueSizeY;
485
486     Brood::BroodUI::Button* m_btnCardFrontTimeValueIncSizeY;
487
488 // ===== UI elements to change CardUpPrompt size x =====
489
490     Brood::BroodUI::TextBox* m_txtCardUpPromptSizeXPrompt;
491
492     Brood::BroodUI::Button* m_btnCardUpPromptDecSizeX;
```

```
527     Brood::BroodUI::TextBox* m_txtCardUpPromptSizeX;
529
530     Brood::BroodUI::Button* m_btnCardUpPromptIncSizeX;
531
532     // ===== UI elements to change CardUpPrompt size y =====
533
534     Brood::BroodUI::TextBox* m_txtCardUpPromptSizeYPrompt;
535
536     Brood::BroodUI::Button* m_btnCardUpPromptDecSizeY;
537
538     Brood::BroodUI::TextBox* m_txtCardUpPromptSizeY;
539
540     Brood::BroodUI::Button* m_btnCardUpPromptIncSizeY;
541
542     // ===== UI elements to change CardUpValue size x =====
543
544     Brood::BroodUI::TextBox* m_txtCardUpValueSizeXPrompt;
545
546     Brood::BroodUI::Button* m_btnCardUpValueDecSizeX;
547
548     Brood::BroodUI::TextBox* m_txtCardUpValueSizeX;
549
550     Brood::BroodUI::Button* m_btnCardUpValueIncSizeX;
551
552     // ===== UI elements to change CardUpValue size y =====
553
554     Brood::BroodUI::TextBox* m_txtCardUpValueSizeYPrompt;
555
556     Brood::BroodUI::Button* m_btnCardUpValueDecSizeY;
557
558     Brood::BroodUI::TextBox* m_txtCardUpValueSizeY;
559
560     Brood::BroodUI::Button* m_btnCardUpValueIncSizeY;
561
562     // ===== UI elements to change CardDownPrompt size x =====
563
564     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeXPrompt;
565
566     Brood::BroodUI::Button* m_btnCardDownPromptDecSizeX;
567
568     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeX;
569
570     Brood::BroodUI::Button* m_btnCardDownPromptIncSizeX;
571
572     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeY;
573
574     Brood::BroodUI::Button* m_btnCardDownPromptIncSizeY;
575
576     // ===== UI elements to change CardDownPrompt size y =====
577
578     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeYPrompt;
579
580     Brood::BroodUI::Button* m_btnCardDownPromptDecSizeY;
581
582     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeX;
583
584     Brood::BroodUI::Button* m_btnCardDownPromptIncSizeX;
585
586     Brood::BroodUI::TextBox* m_txtCardDownPromptSizeY;
587
588     Brood::BroodUI::Button* m_btnCardDownPromptIncSizeY;
589
590     // ===== UI elements to change CardDownValue size x =====
591
592     Brood::BroodUI::TextBox* m_txtCardDownValueSizeXPrompt;
593
594     Brood::BroodUI::Button* m_btnCardDownValueDecSizeX;
595
596     Brood::BroodUI::TextBox* m_txtCardDownValueSizeX;
597
598     Brood::BroodUI::Button* m_btnCardDownValueIncSizeX;
599
600     Brood::BroodUI::TextBox* m_txtCardDownValueSizeY;
601
602     Brood::BroodUI::Button* m_btnCardDownValueIncSizeY;
603
604     // ===== UI elements to change CardDownValue size y =====
605
606     Brood::BroodUI::TextBox* m_txtCardDownValueSizeYPrompt;
607
608     Brood::BroodUI::Button* m_btnCardDownValueDecSizeY;
609
610     Brood::BroodUI::TextBox* m_txtCardDownValueSizeX;
611
612     Brood::BroodUI::Button* m_btnCardDownValueIncSizeX;
613
614     Brood::BroodUI::TextBox* m_txtCardDownValueSizeY;
615
616     Brood::BroodUI::Button* m_btnCardDownValueIncSizeY;
617
618     // ===== UI elements to change CardTurnCard size x =====
619
620     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeXPrompt;
621
622     Brood::BroodUI::Button* m_btnCardTurnCardDecSizeX;
623
624     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeX;
625
626     Brood::BroodUI::Button* m_btnCardTurnCardIncSizeX;
627
628     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeY;
629
630     Brood::BroodUI::Button* m_btnCardTurnCardIncSizeY;
631
632     // ===== UI elements to change CardTurnCard size y =====
633
634     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeYPrompt;
635
636     Brood::BroodUI::Button* m_btnCardTurnCardDecSizeY;
637
638     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeX;
639
640     Brood::BroodUI::Button* m_btnCardTurnCardIncSizeX;
641
642     Brood::BroodUI::TextBox* m_txtCardTurnCardSizeY;
643
644     Brood::BroodUI::Button* m_btnCardTurnCardIncSizeY;
645
646     // ===== UI elements to change CardTurnCard size y =====
647
```

```
649 Brood::BroodUI::TextBox* m_txtCardTurnCardSizeYPrompt;
650 Brood::BroodUI::Button* m_btnCardTurnCardDecSizeY;
651
652 Brood::BroodUI::TextBox* m_txtCardTurnCardSizeY;
653 Brood::BroodUI::Button* m_btnCardTurnCardIncSizeY;
654
655 // =====
656 // ===== member variables for back of the card =====
657 // =====
658
659 // === UI elements to change CardBackTimePrompt position x ===
660
661 Brood::BroodUI::TextBox* m_txtCardBackTimePromptPosXPrompt;
662 Brood::BroodUI::Button* m_btnCardBackTimePromptDecPosX;
663
664 Brood::BroodUI::TextBox* m_txtCardBackTimePromptPosX;
665 Brood::BroodUI::Button* m_btnCardBackTimePromptIncPosX;
666
667 // === UI elements to change CardBackTimePrompt Position y ===
668
669 Brood::BroodUI::TextBox* m_txtCardBackTimePromptPosYPrompt;
670 Brood::BroodUI::Button* m_btnCardBackTimePromptDecPosY;
671
672 Brood::BroodUI::TextBox* m_txtCardBackTimePromptPosY;
673 Brood::BroodUI::Button* m_btnCardBackTimePromptIncPosY;
674
675 // === UI elements to change CardBackTimeValue position x ===
676
677 Brood::BroodUI::TextBox* m_txtCardBackTimeValuePosXPrompt;
678 Brood::BroodUI::Button* m_btnCardBackTimeValueDecPosX;
679
680 Brood::BroodUI::TextBox* m_txtCardBackTimeValuePosX;
681 Brood::BroodUI::Button* m_btnCardBackTimeValueIncPosX;
682
683 // === UI elements to change CardBackTimeValue Position y ===
684
685 Brood::BroodUI::TextBox* m_txtCardBackTimeValuePosYPrompt;
686 Brood::BroodUI::Button* m_btnCardBackTimeValueDecPosY;
687
688 Brood::BroodUI::TextBox* m_txtCardBackTimeValuePosY;
689 Brood::BroodUI::Button* m_btnCardBackTimeValueIncPosY;
690
691 // === UI elements to change CardQuestionPrompt position x ===
692
693 Brood::BroodUI::TextBox* m_txtCardQuestionPromptPosXPrompt;
694 Brood::BroodUI::Button* m_btnCardQuestionPromptDecPosX;
695
696 Brood::BroodUI::TextBox* m_txtCardQuestionPromptPosX;
697 Brood::BroodUI::Button* m_btnCardQuestionPromptIncPosX;
698
699 // === UI elements to change CardQuestionPrompt Position y ===
700
701 Brood::BroodUI::TextBox* m_txtCardQuestionPromptPosYPrompt;
702 Brood::BroodUI::Button* m_btnCardQuestionPromptDecPosY;
703
704 Brood::BroodUI::TextBox* m_txtCardQuestionPromptPosY;
705 Brood::BroodUI::Button* m_btnCardQuestionPromptIncPosY;
706
707 // === UI elements to change CardQuestionValue position x ===
708
709 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosXPrompt;
710 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosX;
711
712 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosX;
713 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosX;
714
715 // === UI elements to change CardQuestionValue Position y ===
716
717 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosYPrompt;
718 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosY;
719
720 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosY;
721 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosY;
722
723 // === UI elements to change CardQuestionValue position x ===
724
725 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosXPrompt;
726 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosX;
727
728 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosX;
729 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosX;
730
731 // === UI elements to change CardQuestionValue Position y ===
732
733 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosYPrompt;
734 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosY;
735
736 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosY;
737 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosY;
738
739 // === UI elements to change CardQuestionValue position x ===
740
741 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosXPrompt;
742 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosX;
743
744 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosX;
745 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosX;
746
747 // === UI elements to change CardQuestionValue Position y ===
748
749 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosYPrompt;
750 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosY;
751
752 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosY;
753 Brood::BroodUI::Button* m_btnCardQuestionValueIncPosY;
754
755 // === UI elements to change CardQuestionValue Position y ===
756
757 Brood::BroodUI::TextBox* m_txtCardQuestionValuePosYPrompt;
758 Brood::BroodUI::Button* m_btnCardQuestionValueDecPosY;
```

```
769     Brood::BroodUI::TextBox* m_txtCardQuestionValuePosY;
770
771     Brood::BroodUI::Button* m_btnCardQuestionValueIncPosY;
772
773 // ===== UI elements to change CardAnswerPrompt position x =====
774
775     Brood::BroodUI::TextBox* m_txtCardAnswerPromptPosXPrompt;
776
777     Brood::BroodUI::Button* m_btnCardAnswerPromptDecPosX;
778
779     Brood::BroodUI::TextBox* m_txtCardAnswerPromptPosX;
780
781     Brood::BroodUI::Button* m_btnCardAnswerPromptIncPosX;
782
783 // ===== UI elements to change CardAnswerPrompt Position y =====
784
785     Brood::BroodUI::TextBox* m_txtCardAnswerPromptPosYPrompt;
786
787     Brood::BroodUI::Button* m_btnCardAnswerPromptDecPosY;
788
789     Brood::BroodUI::TextBox* m_txtCardAnswerPromptPosY;
790
791     Brood::BroodUI::Button* m_btnCardAnswerPromptIncPosY;
792
793 // ===== UI elements to change CardAnswerValue position x =====
794
795     Brood::BroodUI::TextBox* m_txtCardAnswerValuePosXPrompt;
796
797     Brood::BroodUI::Button* m_btnCardAnswerValueDecPosX;
798
799     Brood::BroodUI::TextBox* m_txtCardAnswerValuePosX;
800
801     Brood::BroodUI::Button* m_btnCardAnswerValueIncPosX;
802
803 // ===== UI elements to change CardAnswerValue Position y =====
804
805     Brood::BroodUI::TextBox* m_txtCardAnswerValuePosYPrompt;
806
807     Brood::BroodUI::Button* m_btnCardAnswerValueDecPosY;
808
809     Brood::BroodUI::TextBox* m_txtCardAnswerValuePosY;
810
811     Brood::BroodUI::Button* m_btnCardAnswerValueIncPosY;
812
813 // ===== UI elements to change CardSubmit position x =====
814
815     Brood::BroodUI::TextBox* m_txtCardSubmitPosXPrompt;
816
817     Brood::BroodUI::Button* m_btnCardSubmitDecPosX;
818
819     Brood::BroodUI::TextBox* m_txtCardSubmitPosX;
820
821     Brood::BroodUI::Button* m_btnCardSubmitIncPosX;
822
823 // ===== UI elements to change CardSubmit Position y =====
824
825     Brood::BroodUI::TextBox* m_txtCardSubmitPosYPrompt;
826
827     Brood::BroodUI::Button* m_btnCardSubmitDecPosY;
828
829     Brood::BroodUI::TextBox* m_txtCardSubmitPosY;
830
831     Brood::BroodUI::Button* m_btnCardSubmitIncPosY;
832
833 // ===== UI elements to change CardBackTimePrompt position x =====
834
835     Brood::BroodUI::TextBox* m_txtCardBackTimePromptSizeXPrompt;
836
837     Brood::BroodUI::Button* m_btnCardBackTimePromptDecSizeX;
838
839     Brood::BroodUI::TextBox* m_txtCardBackTimePromptSizeX;
840
841     Brood::BroodUI::Button* m_btnCardBackTimePromptIncSizeX;
842
843 // ===== UI elements to change CardBackTimePrompt Size y =====
844
845     Brood::BroodUI::TextBox* m_txtCardBackTimePromptSizeYPrompt;
846
847     Brood::BroodUI::Button* m_btnCardBackTimePromptDecSizeY;
848
849     Brood::BroodUI::TextBox* m_txtCardBackTimePromptSizeY;
850
851     Brood::BroodUI::Button* m_btnCardBackTimePromptIncSizeY;
852
853 // ===== UI elements to change CardBackTimeValue position x =====
854
855     Brood::BroodUI::TextBox* m_txtCardBackTimeValuePosXPrompt;
```

```
891 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeXPrompt;
892 Brood::BroodUI::Button* m_btnCardBackTimeValueDecSizeX;
893
894 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeX;
895 Brood::BroodUI::Button* m_btnCardBackTimeValueIncSizeX;
896
897 // ===== UI elements to change CardBackTimeValue Size y =====
898
899 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeYPrompt;
900 Brood::BroodUI::Button* m_btnCardBackTimeValueDecSizeY;
901
902 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeY;
903 Brood::BroodUI::Button* m_btnCardBackTimeValueIncSizeY;
904
905 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeY;
906 Brood::BroodUI::Button* m_btnCardBackTimeValueDecSizeY;
907
908 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeY;
909 Brood::BroodUI::Button* m_btnCardBackTimeValueIncSizeY;
910
911 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeY;
912 Brood::BroodUI::Button* m_btnCardBackTimeValueIncSizeY;
913
914 Brood::BroodUI::TextBox* m_txtCardBackTimeValueSizeY;
915 Brood::BroodUI::Button* m_btnCardBackTimeValueIncSizeY;
916
917 // ===== UI elements to change CardQuestionPrompt position x =====
918
919 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeXPrompt;
920 Brood::BroodUI::Button* m_btnCardQuestionPromptDecSizeX;
921
922 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeX;
923 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeX;
924
925 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeX;
926 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeX;
927
928 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeX;
929 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeX;
930
931 // ===== UI elements to change CardQuestionPrompt Size y =====
932
933 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeYPrompt;
934 Brood::BroodUI::Button* m_btnCardQuestionPromptDecSizeY;
935
936 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeY;
937 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeY;
938
939 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeY;
940 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeY;
941
942 Brood::BroodUI::TextBox* m_txtCardQuestionPromptSizeY;
943 Brood::BroodUI::Button* m_btnCardQuestionPromptIncSizeY;
944
945 // ===== UI elements to change CardQuestionValue position x =====
946
947 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeXPrompt;
948 Brood::BroodUI::Button* m_btnCardQuestionValueDecSizeX;
949
950 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeX;
951 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeX;
952
953 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeX;
954 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeX;
955
956 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeX;
957 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeX;
958
959 // ===== UI elements to change CardQuestionValue Size y =====
960
961 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeYPrompt;
962 Brood::BroodUI::Button* m_btnCardQuestionValueDecSizeY;
963
964 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeY;
965 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeY;
966
967 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeY;
968 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeY;
969
970 Brood::BroodUI::TextBox* m_txtCardQuestionValueSizeY;
971 Brood::BroodUI::Button* m_btnCardQuestionValueIncSizeY;
972
973 // ===== UI elements to change CardAnswerPrompt position x =====
974
975 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeXPrompt;
976 Brood::BroodUI::Button* m_btnCardAnswerPromptDecSizeX;
977
978 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeX;
979 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeX;
980
981 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeX;
982 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeX;
983
984 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeX;
985 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeX;
986
987 // ===== UI elements to change CardAnswerPrompt Size y =====
988
989 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeYPrompt;
990 Brood::BroodUI::Button* m_btnCardAnswerPromptDecSizeY;
991
992 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeY;
993 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeY;
994
995 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeY;
996 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeY;
997
998 Brood::BroodUI::TextBox* m_txtCardAnswerPromptSizeY;
999 Brood::BroodUI::Button* m_btnCardAnswerPromptIncSizeY;
1000
1001 // ===== UI elements to change CardAnswerValue position x =====
1002
1003 Brood::BroodUI::TextBox* m_txtCardAnswerValueSizeXPrompt;
1004 Brood::BroodUI::Button* m_btnCardAnswerValueDecSizeX;
1005
1006 Brood::BroodUI::TextBox* m_txtCardAnswerValueSizeX;
1007 Brood::BroodUI::Button* m_btnCardAnswerValueIncSizeX;
1008
1009 Brood::BroodUI::TextBox* m_txtCardAnswerValueSizeX;
1010 Brood::BroodUI::Button* m_btnCardAnswerValueIncSizeX;
```

```
1013 // === UI elements to change CardAnswerValue Size y ====
1014 Brood::BroodUI::TextBox* m_txtCardAnswerValueSizeYPrompt;
1015
1016 Brood::BroodUI::Button* m_btnCardAnswerValueDecSizeY;
1017
1018 Brood::BroodUI::TextBox* m_txtCardAnswerValueSizeY;
1019
1020 Brood::BroodUI::Button* m_btnCardAnswerValueIncSizeY;
1021
1022 // === UI elements to change CardSubmit position x ====
1023 Brood::BroodUI::TextBox* m_txtCardSubmitSizeXPrompt;
1024
1025 Brood::BroodUI::Button* m_btnCardSubmitDecSizeX;
1026
1027 Brood::BroodUI::TextBox* m_txtCardSubmitSizeX;
1028
1029 Brood::BroodUI::Button* m_btnCardSubmitIncSizeX;
1030
1031 // === UI elements to change CardSubmit Size y ====
1032 Brood::BroodUI::TextBox* m_txtCardSubmitSizeYPrompt;
1033
1034 Brood::BroodUI::Button* m_btnCardSubmitDecSizeY;
1035
1036 Brood::BroodUI::TextBox* m_txtCardSubmitSizeY;
1037
1038 Brood::BroodUI::Button* m_btnCardSubmitIncSizeY;
1039
1040 // ===== member variables for card info setting =====
1041 // =====
1042 // ===== UI elements to change current Deck number =====
1043 Brood::BroodUI::TextBox* m_txtDeckNumPrompt;
1044
1045 Brood::BroodUI::Button* m_btnDecDeckNum;
1046
1047 Brood::BroodUI::TextBox* m_txtDeckNum;
1048
1049 Brood::BroodUI::Button* m_btnIncDeckNum;
1050
1051 // ===== UI elements to change current Card number =====
1052 Brood::BroodUI::TextBox* m_txtCurrCardNumPrompt;
1053
1054 Brood::BroodUI::Button* m_btnDecCurrCardNum;
1055
1056 Brood::BroodUI::TextBox* m_txtCurrCardNum;
1057
1058 Brood::BroodUI::Button* m_btnIncCurrCardNum;
1059
1060 // ===== UI elements to change current Card Time =====
1061 Brood::BroodUI::TextBox* m_txtCurrCardTimePrompt;
1062
1063 Brood::BroodUI::Button* m_btnDecCurrCardTime;
1064
1065 Brood::BroodUI::TextBox* m_txtCurrCardTime;
1066
1067 Brood::BroodUI::Button* m_btnIncCurrCardTime;
1068
1069 // ===== UI elements to change current Card Up =====
1070 Brood::BroodUI::TextBox* m_txtCurrCardUpPrompt;
1071
1072 Brood::BroodUI::Button* m_btnDecCurrCardUp;
1073
1074 Brood::BroodUI::TextBox* m_txtCurrCardUp;
1075
1076 Brood::BroodUI::Button* m_btnIncCurrCardUp;
1077
1078 // ===== UI elements to change current Card Down =====
1079 Brood::BroodUI::TextBox* m_txtCurrCardDownPrompt;
1080
1081 Brood::BroodUI::Button* m_btnDecCurrCardDown;
1082
1083 Brood::BroodUI::TextBox* m_txtCurrCardDown;
1084
1085 Brood::BroodUI::Button* m_btnIncCurrCardDown;
1086
1087 // ===== UI elements to enter a curr Card Question =====
1088
```

```

1133     Brood::BroodUI::TextBox* m_txtCurrCardQuestionPrompt;
1134
1135     Brood::BroodUI::TextBox* m_txtCurrCardQuestionInput;
1136
1137     // ===== UI elements to enter a curr Card Answer =====
1138
1139     Brood::BroodUI::TextBox* m_txtCurrCardAnswerPrompt;
1140
1141     Brood::BroodUI::TextBox* m_txtCurrCardAnswerInput;
1142
1143     // ===== UI elements to turn the card =====
1144
1145     Brood::BroodUI::Button* m_btnTurnCardAround;
1146
1147 };
1148
1149 // =====
1150 // ===== end of CardEditor class =====
1151 // =====
1152 // =====
1153 // =====
1154
1155
1156

```

10.16 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/CardInfo.cpp File Reference

```
#include "stdafx.h"
#include "CardInfo.h"
```

10.17 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/CardInfo.h File Reference

This file is a header file for [Dice](#) class.

```
#include "Data.h"
```

Classes

- class [Brood::Application::Components::CardInfo](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
 - TODO initialize all of the data here to match the default values.*
- namespace [Brood::Application::Components](#)

10.17.1 Detailed Description

This file is a header file for [Dice](#) class.

It contains all of the declaration of the member function of [Dice](#) class.

10.18 CardInfo.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // ****
3 // =====
4 // ===== included files =====
5 // =====
6
7 #pragma once
8 #include "Data.h"
9
10 // =====
11 // ===== defining namespace =====
12 // =====
13
14 // =====
15 // =====
16
17 // =====
18 // =====
19
20 // =====
21 // ===== start of CardInfo class =====
22 // =====
23 namespace Brood::Application
24 {
25     namespace Components
26     {
27         class CardInfo;
28     }
29 }
30 // =====
31 // ===== end of namespace defination =====
32 // =====
33
34 // =====
35 // ===== start of CardInfo class =====
36 // =====
37
38 class Brood::Application::Components::CardInfo
39 {
40     // ===== public memeber function =====
41 public:
42     // default constructor
43     CardInfo( unsigned a_time = 0, unsigned a_correctNumSteps = 0,
44               unsigned a_incorrectNumSteps = 0, std::string a_question = "",
45               std::string a_correctAnswer = "" );
46
47     // default constructor
48     ~CardInfo();
49
50     // copy constructor
51     CardInfo( const CardInfo& a_other );
52
53     // assignment constructor
54     CardInfo& operator = ( const CardInfo& a_other );
55
56     // initialize the card info
57     void InitializeCard( Brood::Application::Data::ST_CardInfoPrefabData& a_cardinfoData );
58
59     // returns the card Info data to save
60     Brood::Application::Data::ST_CardInfoPrefabData GetDataToSave();
61
62     // ===== getter function =====
63
64     // getter funciton to get time
65     unsigned GetTime();
66
67     // getter funciton to get correct number of steps
68     unsigned GetCorrectNumSteps();
69
70     // getter funciton to get incorrect number of steps
71     unsigned GetIncorrectNumSteps();
72
73     // getter funciton to get question
74     std::string GetQuestion();
75
76     // getter funciton to get correct answer
77     std::string GetCorrectAnswer();
78
79     // ===== setter function =====
80
81     // setter funciton to get time
82     void SetTime( unsigned a_time );
83
84     // setter funciton to get correct number of steps
85     void SetCorrectNumSteps( unsigned a_correctNumSteps );
86
87     // setter funciton to get incorrect number of steps
88     void SetIncorrectNumSteps( unsigned a_incorrectNumSteps );
89
90     // setter funciton to get question
91
92
93
94
95
96
```

```

97     void SetQuestion( std::string a_question );
98
99     // setter funciton to get correct answer
100    void SetCorrectAnswer( std::string a_correctAnswer );
101
102    // ====== private member variables ======
103 private:
104
105    // ===== following are in the front of the card =====
106
108    unsigned m_time;
109
111    unsigned m_correctNumSteps;
112
114    unsigned m_incorrectNumSteps;
115
116    // ===== following are in the back of the card =====
117
119    std::string m_question;
120
121    // stores the correct answer
122    std::string m_correctAnswer;
123 };
124
125 // =====
126 // ===== end of CardInfo class =====
127 // =====

```

10.19 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Data.cpp File Reference

```
#include "stdafx.h"
#include "Data.h"
```

10.20 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Data.h File Reference

This file is a header file for all the Data that the application uses.

Classes

- struct [Brood::Application::Data::ST_GameData](#)
- struct [Brood::Application::Data::ST_BoardData](#)
- struct [Brood::Application::Data::ST_CardInfoPrefabData](#)
- struct [Brood::Application::Data::ST_DeckPrefabData](#)
- struct [Brood::Application::Data::ST_DeckManagerData](#)
- struct [Brood::Application::Data::ST_DicePrefabData](#)
- struct [Brood::Application::Data::ST_DisplayCardData](#)
- struct [Brood::Application::Data::ST_PlayerPrefabData](#)
- struct [Brood::Application::Data::ST_PlayerManagerData](#)
- struct [Brood::Application::Data::ST_PathPrefabData](#)

Namespaces

- namespace `Brood`
- namespace `Brood::Application`
TODO initialize all of the data here to match the default values.
- namespace `Brood::Application::Data`

10.20.1 Detailed Description

This file is a header file for all the Data that the application uses.

It contains enums declaration used by the Application namely: movement type, and incorect penalty

It also contains all of the data sturct declaration used by the application namely: applicaiton data, board data, path data, tile data, deck data card data, player data, and dice data

10.21 Data.h

[Go to the documentation of this file.](#)

```

1 // ****
16 // **** included files ****
17
18 // =====
19 // ===== defining namespace =====
20 // =====
21 #pragma once
22
26
33
35
36
37 // =====
38 // ===== end of namespace defination =====
39 // =====
40 namespace Brood::Application
41 {
42     namespace Data
43     {
44         struct ST_GameData;
45         struct ST_BoardData;
46         struct ST_CardInfoPrefabData;
47         struct ST_DeckPrefabData;
48         struct ST_DeckManagerData;
49         struct ST_DicePrefabData;
50         struct ST_DisplayCardData;
51         struct ST_PlayerPrefabData;
52         struct ST_PlayerManagerData;
53         struct ST_PathPrefabData;
54     }
55
56 }
57
58
59
60 // =====
61 // ===== start of ST_GameData struct defination =====
62 // =====
63
64 // =====
65 // ===== start of ST_BoardData struct defination =====
66 // =====
67
68 struct Brood::Application::Data::ST_GameData
69 {
70     std::string stm_gameTitle = "";
71
72     // ===== functions =====
73
74     // populates the struct from passed string
75     void PopulateFromString(std::string a_string);
76
77     // creates a string and appends each data to its back
78     // separated by '
79     const std::string GetString() const;
80
81
82
83
84
85
86

```

```

87     //int stm_totalTileTypes; /// total number of tile prefabs
88     //int stm_totalDeckTypes; /// total number of deck prefabs
89     //int stm_totalDiceTypes; /// total number of dice prefabs
90     //int stm_totalPPlayerTypes; /// total number of player prefabs
91     //int stm_playerNum; /// number of player set
92     //int stm_minPlayerNumer; /// minimum number of player required to play the game
93     //int stm_maxPlayerNumer; /// maximum number of player allowed for a game
94     //int stm_currPlayerTurn; /// index of player whose turn it is currently
105 };
106
107 // =====
108 // ===== end of ST_GameData struct defination =====
109 // =====
110
111 // =====
112 // ===== start of ST_BoardData struct defination =====
113 // =====
114
120 struct Brood::Application::Data::ST_BoardData
121 {
123     float stm_boardSizeX = 0;
124
126     float stm_boardSizeY = 0;
127
129     float stm_boardPosX = 0;
130
132     float stm_boardPosY = 0;
133
135     unsigned stm_numRow = 1;
136
138     unsigned stm_numCol = 1;
139
141     unsigned stm_currActiveNumRow = 0;
142
144     unsigned stm_currActiveNumCol = 0;
145
146     // ===== functions =====
147
148     // populates the struct from passed string
149     void PopulateFromString( std::string a_string );
150
151     // creates a string and appends each data to its back
152     // separated by ','
153     const std::string GetString() const;
154 };
155
156 // =====
157 // ===== end of ST_BoardData struct defination =====
158 // =====
159
160 // =====
161 // ===== start of ST_CardInfoPrefabData struct defination =====
162 // =====
163
171 struct Brood::Application::Data::ST_CardInfoPrefabData
172 {
173     // ===== data in the front of the card =====
174
176     unsigned stm_time = 0;
177
179     unsigned stm_correctNumSteps = 0;
180
182     unsigned stm_incorrectNumSteps = 0;
183
184     // ===== data in the back of the card =====
185
187     std::string stm_question = "";
188
190     std::string stm_correctAnswer = "";
191
192     // ===== functions =====
193
194     // populates the struct from passed string
195     void PopulateFromString( std::string a_string );
196
197     // creates a string and appends each data to its back
198     // separated by ','
199     const std::string GetString() const;
200 };
201
202 // =====
203 // ===== end of ST_CardInfoPrefabData struct defination =====
204 // =====
205
206 // =====
207 // ===== start of ST_DeckPrefabData struct defination =====
208 // =====

```

```
209 struct Brood::Application::Data::ST_DeckPrefabData
210 {
211     float stm_deckSizeX= 0;
212     float stm_deckSizeY= 0;
213     float stm_deckPosX= 0;
214     float stm_deckPosY= 0;
215
216     std::string stm_textureFileName = "";
217     std::string stm_cardInitFilename = "";
218
219     unsigned stm_numTotalCard= 1;
220
221     unsigned stm_undealtCardIdx= 0;
222
223     unsigned stm_currActiveCardIdx= 0;
224
225     // ===== functions =====
226
227     // populates the struct from passed string
228     void PopulateFromString( std::string a_string );
229
230     // creates a string and appends each data to its back
231     // seperated by ','
232     const std::string GetString() const;
233 };
234
235 // =====
236 // ===== end of ST_DeckPrefabData struct defination =====
237 // =====
238
239 // ===== start of ST_DeckManagerData struct defination =====
240 // =====
241
242 struct Brood::Application::Data::ST_DeckManagerData
243 {
244     unsigned stm_movementType= 0;
245
246     bool stm_incorrectPenalty= false;
247
248     unsigned stm_numDecks= 1;
249
250     unsigned stm_currDecksIdx= 0;
251
252     // ===== functions =====
253
254     // populates the struct from passed string
255     void PopulateFromString( std::string a_string );
256
257     // creates a string and appends each data to its back
258     // seperated by ','
259     const std::string GetString() const;
260 };
261
262 // =====
263 // ===== end of ST_DeckManagerData struct defination =====
264 // =====
265
266 // ===== start of ST_DicePrefabData struct defination =====
267 // =====
268
269 struct Brood::Application::Data::ST_DicePrefabData
270 {
271     unsigned stm_numSides= 6;
272
273     float stm_diceSizeX= 0;
274
275     float stm_diceSizeY= 0;
276
277     float stm_dicePosX= 0;
278
279     float stm_dicePosY= 0;
280
281     std::string stm_textureFileName = "";
282
283     // ===== functions =====
284
285     // populates the struct from passed string
286     void PopulateFromString( std::string a_string );
287
288     // creates a string and appends each data to its back
289     // seperated by ','
290     const std::string GetString() const;
291 };
292
293 // =====
294 // ===== end of ST_DicePrefabData struct defination =====
295 // =====
296
297 // ===== start of ST_DiceManagerData struct defination =====
298 // =====
299
300 struct Brood::Application::Data::ST_DiceManagerData
301 {
302     unsigned stm_numDice= 1;
303
304     float stm_diceSizeX= 0;
305
306     float stm_diceSizeY= 0;
307
308     float stm_dicePosX= 0;
309
310     float stm_dicePosY= 0;
311
312     std::string stm_textureFileName = "";
313
314     // ===== functions =====
315
316     // populates the struct from passed string
317     void PopulateFromString( std::string a_string );
318
319     // creates a string and appends each data to its back
320     // seperated by ','
321     const std::string GetString() const;
322 };
323
324 // =====
325 // ===== end of ST_DiceManagerData struct defination =====
326 // =====
```

```
334     // seperated by ' '
335     const std::string GetString() const;
336 };
337
338 // =====
339 // ===== end of ST_DicePrefabData struct defination =====
340 // =====
341
342 // =====
343 // ===== start of ST_DisplayCardData struct defination =====
344 // =====
345
346 struct Brood::Application::Data::ST_DisplayCardData
347 {
348     // ===== Display Card Setting =====
349
350     float stm_displayCardSizeX= 0;
351
352     float stm_displayCardSizeY= 0;
353
354     float stm_displayCardPositionX= 0;
355
356     float stm_displayCardPositionY= 0;
357
358     std::string stm_displayCardFrontTextureFilename = "";
359
360     std::string stm_displayCardBackTextureFilename = "";
361
362     // ===== display Card front component setting =====
363
364     float stm_frontTimePromptSizeX= 0;
365
366     float stm_frontTimePromptSizeY= 0;
367
368     float stm_frontTimePromptPositionX= 0;
369
370     float stm_frontTimePromptPositonY= 0;
371
372     float stm_frontTimeValueSizeX= 0;
373
374     float stm_frontTimeValueSizeY= 0;
375
376     float stm_frontTimeValuePositionX= 0;
377
378     float stm_frontTimeValuePositonY= 0;
379
380     float stm_upPromptSizeX= 0;
381
382     float stm_upPromptSizeY= 0;
383
384     float stm_upPromptPositionX= 0;
385
386     float stm_upPromptPositonY= 0;
387
388     float stm_upValueSizeX= 0;
389
390     float stm_upValueSizeY= 0;
391
392     float stm_upValuePositionX= 0;
393
394     float stm_upValuePositonY= 0;
395
396     float stm_downPromptSizeX= 0;
397
398     float stm_downPromptSizeY= 0;
399
400     float stm_downPromptPositionX= 0;
401
402     float stm_downPromptPositonY= 0;
403
404     float stm_downValueSizeX= 0;
405
406     float stm_downValueSizeY= 0;
407
408     float stm_downValuePositionX= 0;
409
410     float stm_downValuePositonY= 0;
411
412     float stm_turnCardPromptSizeX= 0;
413
414     float stm_turnCardPromptSizeY= 0;
415
416     float stm_turnCardPromptPositionX= 0;
417
418     float stm_turnCardPromptPositonY= 0;
419
420     // ===== display Card back component setting =====
```

```
460     float stm_backTimePromptSizeX= 0;
461     float stm_backTimePromptSizeY= 0;
462     float stm_backTimePromptPositionX= 0;
463     float stm_backTimePromptPositionY= 0;
464     float stm_backTimeValueSizeX= 0;
465     float stm_backTimeValueSizeY= 0;
466     float stm_backTimeValuePositionX= 0;
467     float stm_backTimeValuePositionY= 0;
468     float stm_questionPromptSizeX= 0;
469     float stm_questionPromptSizeY= 0;
470     float stm_questionPromptPositionX= 0;
471     float stm_questionPromptPositionY= 0;
472     float stm_questionValueSizeX= 0;
473     float stm_questionValueSizeY= 0;
474     float stm_questionValuePositionX= 0;
475     float stm_questionValuePositionY= 0;
476     float stm_answerPromptSizeX= 0;
477     float stm_answerPromptSizeY= 0;
478     float stm_answerPromptPositionX= 0;
479     float stm_answerPromptPositionY= 0;
480     float stm_answerValueSizeX= 0;
481     float stm_answerValueSizeY= 0;
482     float stm_answerValuePositionX= 0;
483     float stm_answerValuePositionY= 0;
484     float stm_submitButtonValueSizeX= 0;
485     float stm_submitButtonValueSizeY= 0;
486     float stm_submitButtonValuePositionX= 0;
487     float stm_submitButtonValuePositionY= 0;
488 // ===== functions =====
489 // populates the struct from passed string
490 void PopulateFromString( std::string a_string );
491 // creates a string and appends each data to its back
492 // seperated by ','
493 const std::string GetString() const;
494 };
495 // =====
496 // ===== end of ST_DisplayCardData struct defination =====
497 // =====
498 // ===== start of ST_PlayerPrefabData struct defination =====
499 // =====
500 // =====
501 // =====
502 // =====
503 // =====
504 // =====
505 // =====
506 // =====
507 // =====
508 // =====
509 // =====
510 // =====
511 // =====
512 // =====
513 // =====
514 // =====
515 // =====
516 // =====
517 // =====
518 // =====
519 // =====
520 // =====
521 // =====
522 // =====
523 // =====
524 // =====
525 // =====
526 // =====
527 // =====
528 // =====
529 // =====
530 // =====
531 // =====
532 // =====
533 // =====
534 // =====
535 // =====
536 // =====
537 // =====
538 // =====
539 // =====
540 // =====
541 // =====
542 // =====
543 // =====
544 // =====
545 // =====
546 // =====
547 // =====
548 // =====
549 // =====
550 // =====
551 // =====
552 // =====
553 // =====
554 // =====
555 // =====
556 // =====
557 // =====
558 // =====
559 // =====
560 // =====
561 // =====
562 // =====
563 // =====
564 // =====
565 // =====
566 // =====
567 // =====
568 struct Brood::Application::Data::ST_PlayerPrefabData
569 {
570     float stm_currRow= 0;
571     float stm_currCol= 0;
572     float stm_playerSizeX= 0;
573     float stm_playerSizeY= 0;
574     int stm_playerOffsetX= 0;
575
576
577
578
579
580
581
582
583
584 }
```

```
586     int stm_playerOffsetY= 0;
587
588     std::string stm_textureFileName = "";
589
590     // ===== functions =====
591
592     // populates the struct from passed string
593     void PopulateFromString( std::string a_string );
594
595     // creates a string and appends each data to its back
596     // separated by ','
597     const std::string GetString() const;
598
599 };
600
601 // =====
602 // ===== end of ST_PlayerPrefabData struct defination =====
603 // =====
604
605
606 // =====
607 // ===== start of ST_PlayerManagerData struct defination =====
608 // =====
609
610
611 struct Brood::Application::Data::ST_PlayerManagerData
612 {
613     unsigned stm_minPlayer= 1;
614
615     unsigned stm_maxPlayer= 1;
616
617     unsigned stm_currPlayerIdx= 0;
618
619     // ===== functions =====
620
621     // populates the struct from passed string
622     void PopulateFromString( std::string a_string );
623
624     // creates a string and appends each data to its back
625     // separated by ','
626     const std::string GetString() const;
627
628 };
629
630
631 // =====
632 // ===== end of ST_PlayerManagerData struct defination =====
633 // =====
634
635
636 // =====
637 // ===== start of ST_PathPrefabData struct defination =====
638 // =====
639
640
641 // =====
642 // ===== start of ST_PathPrefabData struct defination =====
643 // =====
644
645 struct Brood::Application::Data::ST_PathPrefabData
646 {
647     unsigned stm_tileType= 0;
648
649     unsigned stm_RowNum= 0;
650
651     unsigned stm_ColNum= 0;
652
653     unsigned stm_nextTileRowNum= 0;
654
655     unsigned stm_nextTileColNum= 0;
656
657     unsigned stm_endBridgeTileRowNum= 0;
658
659     unsigned stm_endBridgeTileColNum= 0;
660
661     unsigned stm_numberCardDraw= 0;
662
663     unsigned stm_assignedDeckId= 0;
664
665     bool stm_forceDiceRoll= false;
666
667     std::string stm_textureFileName = "";
668
669     // ===== functions =====
670
671     // populates the struct from passed string
672     void PopulateFromString( std::string a_string );
673
674     // creates a string and appends each data to its back
675     // separated by ','
676     const std::string GetString() const;
677
678 };
679
680
681 // =====
682 // ===== end of ST_PathPrefabData struct defination =====
683 // =====
```

10.22 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Deck.cpp File Reference

```
#include "stdafx.h"
#include "Deck.h"
```

10.23 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Deck.h File Reference

This file is a source file for Deck class.

```
#include "stdafx.h"
#include "Button.h"
#include "CardInfo.h"
#include "GlobalVariables.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::Deck](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.23.1 Detailed Description

This file is a source file for Deck class.

This file is a header file for Deck class.

It contains all of the definition of the member function of Deck class.

It contains all of the declaration of the member function of Deck class.

10.24 Deck.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // =====
11 // ===== included files =====
12 // =====
13 // =====
14 // =====
15
16 #pragma once
17 #include "stdafx.h"
18 #include "Button.h"
19 #include "CardInfo.h"
20 #include "GlobalVariables.h"
21 #include "FileAccess.h"
22
23 // =====
24 // ===== defining namespace =====
25 // =====
26 namespace Brood::Application
27 {
28     namespace Components
29     {
30         class Deck;
31     }
32 }
33 // =====
34 // ===== end of namespace defination =====
35 // =====
36
37 // =====
38 // ===== start of Deck class =====
39 // =====
40
51 class Brood::Application::Components::Deck :
52     public Brood::BroodUI::Button
53 {
54     // ===== public member funciton =====
55 public:
56
57     // default construtor
58     Deck( Brood::BroodUI::UIElement* a_parentPtr = nullptr );
59
60     // default destructor
61     ~Deck();
62
63     // copy constructor
64     Deck( const Brood::Application::Components::Deck& a_other );
65
66     // assignment constructor
67     Deck& operator = ( const Brood::Application::Components::Deck& a_other );
68
69     // initialize the deck
70     void InitializeDeck( Brood::Application::Data::ST_DeckPrefabData& a_deckData );
71
72     // returns the deck data to save
73     Brood::Application::Data::ST_DeckPrefabData GetDataToSave();
74
75     // appends the the deck data to passed file
76     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr, std::string a_gameTitle,
77     unsigned a_idx );
78
79     // appends the the deck data to passed file
80     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr );
81
82     // ===== getter functions =====
83
84     // getter funciton to get the card list
85     const std::vector< Brood::Application::Components::CardInfo*>& GetCardList() const;
86
87     // gets pointer to the card at given index
88     Brood::Application::Components::CardInfo* GetCardPtrAtIndex( unsigned a_cardInfoIdx );
89
90     // gets card Init file name
91     std::string GetCardInitFileName();
92
93     // gets card Init file path
94     std::string GetCardInitFilePath();
95
96     // gets card current active card index
97     unsigned GetCurrActiveCardIdx();
98
99     // gets pointer to card at current active card index
100    CardInfo* GetCurrActiveCardPtr();

```

```

101 // ====== setter functions ======
102 // sets card current active card index
103 void SetCurrActiveCardIdx( unsigned a_currActiveCardIdx );
104
105 // sets deck size
106 void SetCardListSize( unsigned a_cardListSize );
107
108 // loads cards from a card Init file path
109 bool LoadCardFromInitFile( std::string a_fileInitPath );
110
111 // loads cards from a card Init file path
112 bool SaveCardToInitFile( std::string a_fileInitPath );
113
114 // returns the first undealt card and increases the counter by 1
115 const CardInfo DealCard();
116
117 // shuffles the cards in m_stock
118 void Shuffel();
119
120 // restes the currentCardidx to zero and reshuffles the deck
121 void ResetDeck();
122
123 // adds the passed card to the deck
124 void AddCardInfoToDeck(
125     Brood::Application::Components::CardInfo a_cardToAdd );
126
127 // ====== private member variable ======
128 private:
129     // stack of undelt cards
130     std::vector<Brood::Application::Components::CardInfo*> m_cardList;
131
132     // holds the index of first undealt card
133     unsigned m_currUndealtCardIdx;
134
135     // holds the card init file Path
136     std::string m_cardInitFilePath;
137
138     // holds the card init file Path
139     std::string m_cardInitFileName;
140
141     // holds the index of card that is currently active
142     unsigned m_currActiveCardIdx;
143
144 };
145
146
147 // ======
148 // ====== start of Deck class ======
149 // ======
150 // ======

```

10.25 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DeckEditor.cpp File Reference

This file is a source file for DeckEditor class.

```
#include "stdafx.h"
#include "DeckEditor.h"
```

10.25.1 Detailed Description

This file is a source file for DeckEditor class.

It contains all of the defination of the member funciton of DeckEditor class.

10.26 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DeckEditor.h File Reference

This file is a header file for DeckEditor class.

```
#include "MenuBar.h"
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::DeckEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.26.1 Detailed Description

This file is a header file for DeckEditor class.

It contains all of the declaration of the member funciton of DeckEditor class.

10.27 DeckEditor.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // =====
3 // ===== included files =====
4 // =====
5 #pragma once
6 #include "MenuBar.h" // includes all the other uiElements
7 #include "WorkSpace.h"
8
9
10 // =====
11 // ===== defining namespace =====
12 // =====
13 namespace Brood
14 {
15     namespace Application
16     {
17         class DeckEditor;
18     }
19
20 // =====
21 // ===== end of namespace defination =====
22 // =====
23
24
25 // ===== start of DeckEditor class =====
26 // =====
27 // =====
28 // =====
29 // =====
30 // =====
31 // =====
32 // =====
33
34 // =====
35 // =====
36 // =====
37
38 class Brood::Application::DeckEditor :
39     public Brood::Application::WorkSpace
40 {
```

```
51     // ====== public member function ======
52 public:
53     // default constructor
54     DeckEditor( Brood::Application::Components::GameDataManager* aGameData,
55                 sf::RectangleShape* a_panelPtr );
56
57     // default destructor
58     ~DeckEditor();
59     // ===== virtual funcitons ====
60
61     // initializes the component of a workspace
62     void InitializeWorkSpace();
63
64     // all the logic for the workspace is here
65     void Update();
66
67     // updates all the display elment
68     void UpdateAllDispalyElement();
69
70     // reners to the screen
71     void Draw( sf::RenderWindow& a_window );
72
73     // Calls Debugger for all UI elements in this class
74     void Debugger();
75
76     // ===== private member function =====
77 private:
78
79     // checks if the user interacted current deck index
80     // panel Element
81     void UpdateCurrSelectedDeckIdx();
82
83     // checks if the user interacted with the Deck
84     // x size panel Element
85     void UpdateDeckSizeX();
86
87     // checks if the user interacted with the Deck
88     // y Pos panel Element
89     void UpdateDeckSizeY();
90
91     // checks if the user interacted with the Deck
92     // x size panel Element
93     void UpdateDeckPosX();
94
95     // checks if the user interacted with the Deck
96     // y Pos panel Element
97     void UpdateDeckPosY();
98
99     // checks if the user interacted with the load
100    // deck texture panel Element
101    void UpdateDeckFileTexture();
102
103    // checks if the user interacted with the load
104    // card init file panel Element
105    void UpdateCardInitFile();
106
107    // checks if the user interacted with the number
108    // of card panel Element
109    void UpdateCardNumber();
110
111    // ===== private member variable =====
112 private:
113
114     sf::RectangleShape* m_panelBodyPtr;
115
116     // ===== member variables for deck setting =====
117     // ===== member variables for deck setting =====
118     // ===== member variables for deck setting =====
119
120     // === UI elements to show setting title ===
121     Brood::BroodUI::TextBox* m_txtSettingTitle;
122
123     // === UI elements to change deck current index ===
124     Brood::BroodUI::TextBox* m_txtDeckCurrIdxPrompt;
125
126     Brood::BroodUI::Button* m_btnDeckDecCurrIdx;
127
128     Brood::BroodUI::TextBox* m_txtDeckCurrIdx;
129
130     Brood::BroodUI::Button* m_btnDeckIncCurrIdx;
131
132     // === UI elements to change deck size x ===
133     Brood::BroodUI::TextBox* m_txtDeckSizeXPrompt;
134
135     Brood::BroodUI::Button* m_btnDeckDecSizeX;
```

```

145     Brood::BroodUI::TextBox* m_txtDeckSizeX ;
146
147     Brood::BroodUI::Button* m_btnDeckIncSizeX;
148
149 // ===== UI elements to change deck size y =====
150
151     Brood::BroodUI::TextBox* m_txtDeckSizeYPrompt;
152
153     Brood::BroodUI::Button* m_btnDeckDecSizeY;
154
155     Brood::BroodUI::TextBox* m_txtDeckSizeY;
156
157     Brood::BroodUI::Button* m_btnDeckIncSizeY;
158
159 // ===== UI elements to change deck position x =====
160
161     Brood::BroodUI::TextBox* m_txtDeckPosXPrompt;
162
163     Brood::BroodUI::Button* m_btnDeckDecPosX;
164
165     Brood::BroodUI::TextBox* m_txtDeckPosX;
166
167     Brood::BroodUI::Button* m_btnDeckIncPosX;
168
169 // ===== UI elements to change deck Position y =====
170
171     Brood::BroodUI::TextBox* m_txtDeckPosYPrompt;
172
173     Brood::BroodUI::Button* m_btnDeckDecPosY;
174
175     Brood::BroodUI::TextBox* m_txtDeckPosY;
176
177     Brood::BroodUI::Button* m_btnDeckIncPosY;
178
179 // ===== UI elements to enter a deck filename texture =====
180
181     Brood::BroodUI::TextBox* m_txtDeckBgFileNamePrompt;
182
183     Brood::BroodUI::TextBox* m_txtDeckBgFileInput;
184
185     Brood::BroodUI::Button* m_btnDeckBgOpenFile;
186
187 // ===== UI elements to enter a filename for card init =====
188
189     Brood::BroodUI::TextBox* m_txtCardInitFileNamePrompt;
190
191     Brood::BroodUI::TextBox* m_txtCardInitFileInput;
192
193     Brood::BroodUI::Button* m_btnCardInitOpenFile;
194
195 // ===== UI elements to change number of cards in the deck =====
196
197     Brood::BroodUI::TextBox* m_txtCardNumPrompt;
198
199     Brood::BroodUI::Button* m_btnDecCardNum;
200
201     Brood::BroodUI::TextBox* m_txtCardNum;
202
203     Brood::BroodUI::Button* m_btnIncCardNum;
204
205 };
206
207 // ===== end of DeckEditor class =====
208
209 // =====

```

10.28 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DeckManager.cpp File Reference

This file is a source file for DeckManager class.

```
#include "stdafx.h"
#include "DeckManager.h"
```

10.28.1 Detailed Description

This file is a source file for DeckManager class.

It contains all of the definition of the member function of DeckManager class.

10.29 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DeckManager.h File Reference

This file is a header file for DeckManager class.

```
#include "Deck.h"
#include "Data.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::DeckManager](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

Enumerations

- enum class [Brood::Application::Components::ENUM_MovementType](#) { [Brood::Application::Components::MOVEMENT_diceTheatre](#) , [Brood::Application::Components::MOVEMENT_diceOnly](#) , [Brood::Application::Components::MOVEMENT_cardOnly](#) }

10.29.1 Detailed Description

This file is a header file for DeckManager class.

It contains all of the declaration of the member function of DeckManager class.

10.30 DeckManager.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // =====
11 // ===== included files =====
12 // =====
13 // =====
14 // =====
15 #pragma once
16 #include "Deck.h"
17 #include "Data.h"
18 #include "FileAccess.h"
19
20 // =====
21 // ===== defining namespace =====
22 // =====
23 namespace Brood::Application
24 {
25     namespace Components
26     {
27         enum class ENUM_MovementType;
28         class DeckManager;
29     }
30 }
31 // =====
32 // ===== end of namespace defination =====
33 // =====
34
35
36 // =====
37 // ===== start of ENUM_MovementType Enum class defination ===
38 // =====
39
45 enum class Brood::Application::Components::ENUM_MovementType
46 {
47     MOVEMENT_diceThenCard,
48     MOVEMENT_diceOnly,
49     MOVEMENT_cardOnly,
50 };
51
52 // =====
53 // ===== end of ENUM_UIType Enum class defination =====
54 // =====
55 // =====
56 // ===== start of DeckManager class =====
57 // =====
58
65 class Brood::Application::Components::DeckManager
66 {
67     // ===== private member function =====
68 public:
69     // default constructor
70     DeckManager();
71
72     // destructor
73     ~DeckManager();
74
75     // copy constructor
76     DeckManager( const Brood::Application::Components::DeckManager& a_otherDeckManager );
77
78     // assignment operator
79     DeckManager& operator=( const Brood::Application::Components::DeckManager& a_otherDeckManager );
80
81     // initialize the Deck Manager
82     void InitializeDeckManager( Brood::Application::Data::ST_DeckManagerData& a_deckManagerData );
83
84     // returns the deck manager data to save
85     Brood::Application::Data::ST_DeckManagerData GetDataToSave();
86
87     // appends the the deck manager data to passed file
88     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr, std::string a_gameTile );
89
90     // appends the the deck manager data to passed file
91     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr );
92
93     // ===== getter functions =====
94
95     // getter funciton to get the current deck index
96     unsigned GetCurrActiveDeckIdx( );
97
98     // getter funciton to get the deck at the passed index
99     Brood::Application::Components::Deck* GetDeckAtIndex( unsigned a_deckIdx );
100
101    // getter function to get the decklist

```

```
102     std::vector<Brood::Application::Components::Deck*>& GetDeckList();
103
104     // getter function to get the current active deck
105     Brood::Application::Components::Deck* GetCurrActiveDeck();
106
107     // getter function to get incorrect penalty
108     bool GetIncorrectPenalty();
109
110     // getter function to get movement type
111     Brood::Application::Components::ENUM_MovementType GetMovementType();
112
113     // ===== setter functions =====
114
115     // getter function to get incorrect penalty
116     void SetIncorrectPenalty( bool a_incorrectPenalty );
117
118     // getter function to get movement type
119     void SetMovementType( Brood::Application::Components::ENUM_MovementType a_movementType );
120
121     // sets deck size
122     void SetDeckSize( unsigned a_deckSize );
123
124     // sets current deck index
125     void SetCurrActiveDeckIdx( unsigned a_currDeckIdx );
126
127     // adds card to the back of current selected deck
128     void AddCardToDeckAtCurrIdx( Brood::Application::Components::CardInfo a_cardToAdd );
129
130     // draws all the deck to the screen
131     void Draw( sf::RenderWindow& a_window );
132
133     // debugger
134     void Debugger();
135
136     // ===== private member variables =====
137 private:
138
139     // ture if the incorrect penalty rule should be enforced or not
140     bool m_incorrectPenalty;
141
142     // stores the movement type used
143     Brood::Application::Components::ENUM_MovementType m_movementType;
144
145     // current deck index
146     unsigned m_currDeckIdx;
147
148     // deck list
149     std::vector<Brood::Application::Components::Deck*> m_deckList;
150 };
151
152 // =====
153 // ===== end of DeckManager class =====
154 // =====
```

10.31 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Dice.cpp File Reference

```
#include "stdafx.h"
#include "Dice.h"
```

10.32 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Dice.h File Reference

This file is a source file for [Dice](#) class.

```
#include "Button.h"
#include "Data.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::Dice](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.32.1 Detailed Description

This file is a source file for [Dice](#) class.

This file is a header file for [Dice](#) class.

It contains all of the defination of the member funciton of [Dice](#) class.

It contains all of the declaration of the member funciton of [Dice](#) class.

10.33 Dice.h

[Go to the documentation of this file.](#)

```
1 // ****
10 // =====
11
12 // ===== included files =====
13 // =====
14 // =====
15 #pragma once
16 #include "Button.h"
17 #include "Data.h"
18 #include "FileAccess.h"
19
20 // =====
21 // ===== defining namespace =====
22 // =====
23 namespace Brood::Application
24 {
25     namespace Components
26     {
27         class Dice;
28     }
29 }
30
31 // =====
32 // ===== end of namespace defination =====
33 // =====
34
35 // =====
36 // ===== start of Dice class =====
37 // =====
38
47 class Brood::Application::Components::Dice :
48     public BroodUI::Button
49 {
```

```

50     // ====== public member funciton ======
51 public:
52     // default constructor
53     Dice( int a_numSides = 6, Brood::BroodUI::UIElement* a_parentPtr = nullptr );
54
55     // constructor when a texture is passed
56     Dice( std::string a_texturePath, int a_numSides = 6,
57           Brood::BroodUI::UIElement* a_parentPtr = nullptr );
58
59     // default destructor
60     virtual ~Dice();
61
62     // copy constructor
63     Dice( const Dice& a_otherDice );
64
65     // assignment operator
66     Dice& operator=( const Dice& a_otherDice );
67
68     // initialize the dice
69     void InitializeDice( Brood::Application::Data::ST_DicePrefabData& a_diceData );
70
71     // returns the dice data to save
72     Brood::Application::Data::ST_DicePrefabData GetDataToSave();
73
74     // appends the the dice data to passed file
75     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
76
77     // appends the the dice data to passed file
78     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr );
79
80     // setter function to set the body size
81     void SetBodySize( sf::Vector2f a_size ) override;
82
83     // setter function to set the body size
84     void SetBodySize( float a_sizeX, float a_sizeY )override;
85
86     // setter function to set number of sides the dice has
87     void SetNumSides( unsigned a_numSides );
88
89     // sets texture
90     bool SetTexture( std::string a_texturePath );
91
92     // getter function
93     const unsigned GetNumSides();
94
95     // get a random num between 0 and m_numSides
96     unsigned RollDice();
97
98     // ====== private member variables ======
99 private:
100     unsigned m_numSides;
101 };
102
103 // ======
104 // ====== end of Dice class ======
105 // ======

```

10.34 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DiceEditor.cpp File Reference

This file is a source file for [DiceEditor](#) class.

```
#include "stdafx.h"
#include "DiceEditor.h"
```

10.34.1 Detailed Description

This file is a source file for [DiceEditor](#) class.

It contains all of the defination of the member funciton of [DiceEditor](#) class.

10.35 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/DiceEditor.h File Reference

This file is a header file for [DiceEditor](#) class.

```
#include "MenuBar.h"
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::DiceEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.35.1 Detailed Description

This file is a header file for [DiceEditor](#) class.

It contains all of the declaration of the member funciton of [DiceEditor](#) class.

10.36 DiceEditor.h

[Go to the documentation of this file.](#)

```
1 // ****
10 // ****
11
12 // =====
13 // ===== included files =====
14 // =====
15 #pragma once
16 #include "MenuBar.h" // includes all the other uiElements
17 #include "WorkSpace.h"
18
19 // =====
20 // ===== defining namespace =====
21 // =====
22 namespace Brood
23 {
24     namespace Application
25     {
26         class DiceEditor;
27     }
28 }
29 // =====
30 // ===== end of namespace defination =====
31 // =====
32
33 // =====
34 // ===== start of DiceEditor class =====
35 // =====
36
52 class Brood::Application::DiceEditor :
53     public Brood::Application::WorkSpace
```

```
54 {
55     // ====== public member function ======
56 public:
57     // default constructor
58     DiceEditor( Brood::Application::Components::GameDataManager* aGameData,
59                 sf::RectangleShape* a_panelPtr );
60
61     // default destructor
62     ~DiceEditor();
63     // ===== virtual funcitons ====
64
65     // initializes the component of a workspace
66     void InitializeWorkSpace();
67
68     // all the logic for the workspace is here
69     void Update();
70
71     // updates all the display elment
72     void UpdateAllDisplayElement();
73
74     // renders to the screen
75     void Draw( sf::RenderWindow& a_window );
76
77     // Calls Debugger for all UI elements in this class
78     void Debugger();
79
80     // ===== private member function =====
81 private:
82     // =====
83     // ===== update for Dice setting =====
84     // =====
85
86     // checks if the user interacted with the Dice
87     // side number panel Element
88     void UpdateDiceSideNum();
89
90     // checks if the user interacted with the Dice
91     // x size panel Element
92     void UpdateDiceSizeX();
93
94     // checks if the user interacted with the Dice
95     // y Pos panel Element
96     void UpdateDiceSizeY();
97
98     // checks if the user interacted with the Dice
99     // x size panel Element
100    void UpdateDicePosX();
101
102    // checks if the user interacted with the Dice
103    // y Pos panel Element
104    void UpdateDicePosY();
105
106    // checks if the user interacted with the load
107    // Dice texture panel Element
108    void UpdateDiceFileTexture();
109
110    // checks if the user interacted with the
111    // roll dice button
112    void UpdateRollDice();
113    // ===== private member variable =====
114 private:
115
116     sf::RectangleShape* m_panelBodyPtr;
117
118     Brood::Application::Components::Board* m_boardPtr;
119
120     Brood::Application::Components::Dice* m_dicePtr;
121
122     // =====
123     // ===== member variables for dice setting =====
124     // =====
125
126     // === UI elements to show setting title ===
127     Brood::BroodUI::TextBox* m_txtSettingTitle;
128
129     // === UI elements to change dice sides ===
130
131     Brood::BroodUI::TextBox* m_txtDiceSidePrompt;
132
133     Brood::BroodUI::Button* m_btnDiceDecSide;
134
135     Brood::BroodUI::TextBox* m_txtDiceSide;
136
137     Brood::BroodUI::Button* m_btnDiceIncSide;
138
139
140
141
142
143
144
145
146     // === UI elements to change x-size of dice ===
147
```

```

149     Brood::BroodUI::TextBox* m_txtDiceSizeXPrompt;
150     Brood::BroodUI::Button* m_btnDiceDecSizeX;
153
155     Brood::BroodUI::TextBox* m_txtDiceSizeX;
156
158     Brood::BroodUI::Button* m_btnDiceIncSizeX;
159
160 // ===== UI elements to change y-size of y-size Index =====
161
163     Brood::BroodUI::TextBox* m_txtDiceSizeYPrompt;
164
166     Brood::BroodUI::Button* m_btnDiceDecSizeY;
167
169     Brood::BroodUI::TextBox* m_txtDiceSizeY;
170
172     Brood::BroodUI::Button* m_btnDiceIncSizeY;
173
174 // ===== UI elements to change x-Pos of Dice Index =====
175
177     Brood::BroodUI::TextBox* m_txtDicePosXPrompt;
178
180     Brood::BroodUI::Button* m_btnDiceDecPosX;
181
183     Brood::BroodUI::TextBox* m_txtDicePosX;
184
186     Brood::BroodUI::Button* m_btnDiceIncPosX;
187
188 // ===== UI elements to change y-Pos of Dice Index =====
189
191     Brood::BroodUI::TextBox* m_txtDicePosYPrompt;
192
194     Brood::BroodUI::Button* m_btnDiceDecPosY;
195
197     Brood::BroodUI::TextBox* m_txtDicePosY;
198
200     Brood::BroodUI::Button* m_btnDiceIncPosY;
201
202 // ===== UI elements to enter a filename texture =====
203
205     Brood::BroodUI::TextBox* m_txtDiceFileNamePrompt;
206
208     Brood::BroodUI::TextBox* m_txtDiceFileInput;
209
211     Brood::BroodUI::Button* m_btnDiceOpenFile;
212
213 // ===== UI elements to change Dice sprite =====
214
216
218     Brood::BroodUI::Button* m_btnRollDice;
219
220
221 };
222
223 // =====
224 // ===== end of DiceEditor class =====
225 // =====

```

10.37 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DisplayCard.cpp File Reference

This file is a source file for DisplayCard class.

```
#include "stdafx.h"
#include "DisplayCard.h"
```

10.37.1 Detailed Description

This file is a source file for DisplayCard class.

It contains all of the defination of the member funciton of DisplayCard class.

10.38 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DisplayCard.h File Reference

```
#include "CardInfo.h"
#include "Button.h"
#include "TextBox.h"
#include "Data.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::DisplayCard](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.39 DisplayCard.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // ****
11 // ****
12
13 // =====
14 // ===== included files =====
15 // =====
16 #pragma once
17 #include "CardInfo.h"
18 #include "Button.h"
19 #include "TextBox.h"
20 #include "Data.h"
21 #include "FileAccess.h"
22
23
24 // =====
25 // ===== defining namespace =====
26 // =====
27 namespace Brood::Application
28 {
29     namespace Components
30     {
31         class DisplayCard;
32     }
33 }
34 // =====
35 // ===== end of namespace defination =====
36 // =====
37
38 // =====
39 // ===== start of DisplayCard class =====
40 // =====
41
48 class Brood::Application::Components::DisplayCard :
49     public BroodUI::Button
50 {
51     // ===== public member function =====
52 public:
53     // default constructor
54     DisplayCard();
```

```

56     // default destructor
57     ~DisplayCard();
58
59     // copy constructor
60     DisplayCard( const DisplayCard& a_other );
61
62     // assignment operator
63     DisplayCard& operator=( const DisplayCard& a_other );
64
65     // Initializes the ui elements
66     void InitializeDisplayCard();
67
68     // initialize the display Card
69     void InitializeDisplayCard( Brood::Application::Data::ST_DisplayCardData& a_displayCardData );
70
71     // returns the display Card data to save
72     Brood::Application::Data::ST_DisplayCardData GetDataToSave();
73
74     // appends the the displayCard data to passed file
75     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
76
77     // appends the the displayCard data to passed file
78     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr );
79
80     // ====== Getter funciton ======
81
82     // getter function to get the front bg filename
83     std::string GetFrontBgFileName();
84
85     // getter function to get the back bg filename
86     std::string GetBackBgFileName();
87
88     // displays time user have to complete the question
89     Brood::BroodUI::TextBox* GetFrontTimePromptPtr();
90     Brood::BroodUI::TextBox* GetFrontTimeValuePtr();
91
92     // displays number of steps that user goes forward
93     Brood::BroodUI::TextBox* GetUpPromptPtr();
94     Brood::BroodUI::TextBox* GetUpValuePtr();
95
96     // displays number of steps that user goes backward
97     Brood::BroodUI::TextBox* GetDownPromptPtr();
98     Brood::BroodUI::TextBox* GetDownValuePtr();
99
100    // button to turn the card of the card
101    Brood::BroodUI::Button* GetTurnCardPtr();
102
103    // ====== back of the card =====
104    // displays time user have to complete the question
105    Brood::BroodUI::TextBox* GetBackTimePromptPtr();
106    Brood::BroodUI::TextBox* GetBackTimeValuePtr();
107
108    // displays the question that user has to answer
109    Brood::BroodUI::TextBox* GetQuestionPromptPtr();
110    Brood::BroodUI::TextBox* GetQuestionValuePtr();
111
112    // displays the textbox for user to enter theiranswer
113    Brood::BroodUI::TextBox* GetAnswerPromptPtr();
114    Brood::BroodUI::TextBox* GetAnswerValuePtr();
115
116    // button to show back of the card
117    Brood::BroodUI::Button* GetSubmitPtr();
118
119    // gets the bottom right cordinate of the box that
120    // encloses all the component
121    sf::Vector2f GetFurtherstBottomRightCoordinate();
122
123    // gets the top left cordinate of the box that
124    // encloses all the component
125    sf::Vector2f GetNearestTopLeftCoordinate();
126
127    // ====== Setter funciton =====
128
129    // setter function to set the front bg filename
130    bool SetFrontBgFileName( std::string a_frontBgFilename );
131
132    // setter function to set the back bg filename
133    bool SetBackBgFileName( std::string a_backBgFilename );
134
135    // setter funciton to set the card info that will be
136    // used to display
137    void SetCardInfoToDisplay( Brood::Application::Components::CardInfo* a_cardInfoToDisplay );
138
139    // toggles the card to display front of the card
140    // or back of the card
141    void ToggleCardFace();
142

```

```
143     // sets m_isCurrFront
144     void SetCardFront( bool a_currFront );
145
146     // updates the displayed text
147     void UpdateDisplayedText();
148
149     // draw funciton
150     void Draw( sf::RenderWindow& a_window );
151
152     void Debugger(); // adds the element ID to the text
153
154 private:
155     // getter funciton to get the bottom right
156     // coridinate of the passed component
157     sf::Vector2f GetCompUpperPos( Brood::BroodUI::UIElement* a_comp );
158
159     // updates the passed largest upper pos by comparaing it with the
160     // upper position of the passed component
161     void UpdateLargestUpperPos( sf::Vector2f& a_largestUpperPos,
162                                 Brood::BroodUI::UIElement* a_comp );
163
164     // updates the passed smaller lower pos by comparaing it with the
165     // lower position of the passed component
166     void UpdateSmallestLowerPos( sf::Vector2f& a_smallestLowerPos,
167                                 Brood::BroodUI::UIElement* a_comp );
168
169
170     // ====== private member variables ======
171 private:
172
173     bool m_isCurrFront;
174
175     // holds the filename for front image
176     std::string m_frontBgFileName;
177
178     // holds the filename for back image
179     std::string m_backBgFileName;
180
181     // pointer contains the info to display
182     Brood::Application::Components::CardInfo* m_cardInfoToDisplay;
183
184     // ====== Font of the card ======
185
186     // displays time user have to complete the question
187     Brood::BroodUI::TextBox* m_TxtFrontTimePrompt;
188     Brood::BroodUI::TextBox* m_TxtFrontTimeValue;
189
190     // displays number of steps that user goes forward
191     Brood::BroodUI::TextBox* m_TxtUpPrompt;
192     Brood::BroodUI::TextBox* m_TxtUpValue;
193
194     // displays number of steps that user goes backward
195     Brood::BroodUI::TextBox* m_TxtDownPrompt;
196     Brood::BroodUI::TextBox* m_TxtDownValue;
197
198     // button to turn the card of the card
199     Brood::BroodUI::Button* m_BtnTurnCard;
200
201     // ====== back of the card ======
202     // displays time user have to complete the question
203     Brood::BroodUI::TextBox* m_TxtBackTimePrompt;
204     Brood::BroodUI::TextBox* m_TxtBackTimeValue;
205
206     // displays the question that user has to answer
207     Brood::BroodUI::TextBox* m_TxtQuestionPrompt;
208     Brood::BroodUI::TextBox* m_TxtQuestionValue;
209
210     // displays the textbox for user to enter theiranswer
211     Brood::BroodUI::TextBox* m_TxtUserAnswerPrompt;
212     Brood::BroodUI::TextBox* m_TxtUserAnswerValue;
213
214     // button to show back of the card
215     Brood::BroodUI::Button* m_BtnSubmit;
216 };
217
218 // ======
219 // ====== end of DisplayCard class ======
220 // ======
```

10.40 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DropDownInput.cpp File Reference

This file is a source file for [DropDownInput](#) class.

```
#include "stdafx.h"
#include "DropDownInput.h"
```

10.40.1 Detailed Description

This file is a source file for [DropDownInput](#) class.

It contains all of the definition of the member function of [DropDownInput](#) class.

10.41 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DropDownInput.h File Reference

This file is a header file for [DropDownInput](#) class.

```
#include "DropDownMenu.h"
```

Classes

- class [Brood::BroodUI::DropDownInput](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.41.1 Detailed Description

This file is a header file for [DropDownInput](#) class.

It contains all of the declaration of the member function of [DropDownInput](#) class.

10.42 DropDownInput.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // =====
11 // ===== included files =====
12 // =====
13 // ===== defining namespace =====
14 // =====
15 #pragma once
16 #include "DropDownMenu.h"
17
18 // =====
19 // ===== defining namespace =====
20 // =====
21 namespace Brood
22 {
23     namespace BroodUI
24     {
25         class DropDownInput;
26     }
27 }
28 // =====
29 // ===== end of namespace defination =====
30 // =====
31
32 // =====
33 // ===== start of DropDownInput class =====
34 // =====
35
127 class Brood::BroodUI::DropDownInput : public Brood::BroodUI::DropDownMenu
128 {
129     // ===== public member function =====
130 public:
131     // default constructor
132     DropDownInput( Brood::BroodUI::UIElement* a_parentPtr = nullptr,
133                     Brood::BroodUI::ENUM_UIType a_enumType =
134                     Brood::BroodUI::ENUM_UIType::UI_dropDownInput );
135     // default destructor
136     virtual ~DropDownInput();
137
138     // copy constructor
139     DropDownInput( const DropDownInput& a_otherElement );
140
141     // assignment operator
142     Brood::BroodUI::DropDownInput& operator=( const DropDownInput& a_otherButton );
143
144     // adds item to menu
145     virtual void AddItemToMenu( std::string a_menuName,
146                                sf::Color a_color =
147                                Brood::Application::StaticVariables::ST_ColourVariables::stm_White );
148
149     // adds the passed button to dropdown menu if a_createNew is true
150     // then create a new button using the passed item then adds
151     // the new button
152     virtual void AddItemToMenu( Brood::BroodUI::Button* a_buttonPtrToAdd,
153                                bool a_createNew = true );
152 };
153

```

10.43 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DropDownMenu.cpp File Reference

This file is a source file for [DropDownMenu](#) class.

```
#include "stdafx.h"
#include "DropDownMenu.h"
```

10.43.1 Detailed Description

This file is a source file for [DropDownMenu](#) class.

It contains all of the defination of the member funciton of [DropDownMenu](#) class.

10.44 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/DropDownMenu.h File Reference

This file is a header file for [DropDownMenu](#) class.

```
#include "Button.h"
```

Classes

- class [Brood::BroodUI::DropDownMenu](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.44.1 Detailed Description

This file is a header file for [DropDownMenu](#) class.

It contains all of the declaration of the member function of [DropDownMenu](#) class.

10.45 DropDownMenu.h

[Go to the documentation of this file.](#)

```
1 // ****
10 // **** included files ****
11 // ****
12 // =====
13 // ===== defining namespace =====
14 // =====
15 #pragma once
16 #include "Button.h"
17 // =====
18 // ===== defining namespace =====
19 // =====
20 // =====
21 namespace Brood
22 {
23     namespace BroodUI
24     {
25         class DropDownMenu;
26     }
27 }
28 // =====
29 // ===== end of namespace definition =====
30 // =====
31 // =====
32 // ===== start of DropDownMenu class =====
33 // =====
34 // =====
35
115 class Brood::BroodUI::DropDownMenu : public Brood::BroodUI::Button
116 {
117     // ===== public member function =====
118 public:
119     // default constructor
120     DropDownMenu( Brood::BroodUI::UIElement* a_parentPtr = nullptr,
121                 Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_dropDownMenu
122             );
123     // default destructor
```

```

123     virtual ~DropDownMenu();
124
125     // copy constructor
126     DropDownMenu( const DropDownMenu& a_otherElement );
127
128     // assignment operator
129     Brood::BroodUI::DropDownMenu& operator=( const DropDownMenu& a_otherElement );
130
131     // getter funciton
132     std::vector<Brood::BroodUI::Button*>& GetItemList();
133
134     // ====== setter function ======
135
136     // sets body size
137     void SetBodySize( sf::Vector2f a_eachItemSize ) override;
138
139     // sets body size
140     void SetBodySize( float a_itemSizeX, float a_itemSizeY ) override;
141
142     // sets body position
143     void SetBodyPosition( sf::Vector2f a_pos, bool a_relativeToParent = false ) override;
144
145     // sets body position
146     void SetBodyPosition( float aPosX, float aPosY, bool a_relativeToParent = false ) override;
147
148     // sets font for the menus
149     void SetFont( sf::Font* a_font );
150
151     // sets font color
152     virtual void SetFontColor( sf::Color a_color =
Brood::Application::StaticVariables::ST_ColourVariables::stm_White );
153
154     // sets fonts size
155     void SetFontSize( unsigned a_fontSize = 12 );
156
157     // sets text of the title menu
158     void SetText( std::string a_text = "" );
159
160     // sets element body color
161     virtual void SetBodyColor( sf::Color a_bodyColor );
162
163     // sets active overlay color
164     virtual void SetActiveOverlayColor( sf::Color a_color );
165
166     // sets hot overlay color
167     virtual void SetHotOverlayColor( sf::Color a_color );
168
169     // adds item to menu
170     virtual void AddItemToMenu( std::string a_menuName,
171                               sf::Color a_color =
Brood::Application::StaticVariables::ST_ColourVariables::stm_White );
172
173     // adds the passed button to dropdown menu if a_createNew is true
174     // then create a new button using the passed item then adds
175     // the new button
176     virtual void AddItemToMenu( Brood::BroodUI::Button* a_buttonPtrToAdd,
177                               bool a_createNew = true );
177
178     // overrided funciton
179
180     // checks to see if the logic for the element is to
181     // be executed or not
182     virtual bool DoElement() override;
183
184     // draw funciton
185     virtual void Draw( sf::RenderWindow& a_window ) override;
186
187     // adds the element ID to the text
188     virtual void Debugger();
189
190     // ====== protected member function ======
191 protected:
192
193     // ===== helper funcitons =====
194
195     // position menu item at given index correctly
196     void SetItemPos( int a_itemIndex );
197
198     // size the menu item at given index correctly
199     void SetItemSize( int a_itemIndex );
200
201     // size all the menu item correctly
202     void SetEachItemSize();
203
204     // position all the menu item correctly
205     void SetEachItemPos();
206
207

```

```

208     // ====== protected member variables ======
209 protected:
210
211     std::vector<Brood::BroodUI::Button*> m_items;
212
213     size_t m_maxItemLength;
214 };
215
216 // =====-
217 // ====== end of DropDownMenu class =====-
218 // =====-
219
220 // =====-

```

10.46 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/EditorWorkspace.cpp File Reference

This file is a source file for EditorWorkspace class.

```
#include "stdafx.h"
#include "EditorWorkspace.h"
```

10.46.1 Detailed Description

This file is a source file for EditorWorkspace class.

It also contains all of the declaration of the EditorWorkspace class's member function.

10.47 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/EditorWorkspace.h File Reference

This file is a header file for InitialWorkSpace class.

```
#include "WorkSpace.h"
#include "BoardEditor.h"
#include "GameEditor.h"
#include "TileEditor.h"
#include "PlayerEditor.h"
#include "DiceEditor.h"
#include "DeckEditor.h"
#include "CardEditor.h"
```

Classes

- class [Brood::Application::EditorWorkspace](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.47.1 Detailed Description

This file is a header file for InitialWorkSpace class.

It contains all of the declaration of the member funciton of InitialWorkSpace class.

10.48 EditorWorkspace.h

[Go to the documentation of this file.](#)

```
1 // ****
11 // ===== included files =====
15 // =====
16 #pragma once
17 #include "WorkSpace.h"
18 #include "BoardEditor.h"
19 #include "GameEditor.h"
20 #include "TileEditor.h"
21 #include "PlayerEditor.h"
22 #include "DiceEditor.h"
23 #include "DeckEditor.h"
24 #include "CardEditor.h"
25
26 // ===== defining namespace =====
27 // ===== end of namespace defination =====
28 // =====
29 namespace Brood
30 {
31     namespace Application
32     {
33         class EditorWorkspace;
34     }
35 }
36 // =====
37 // ===== end of namespace defination =====
38 // =====
39
40 // =====
41 // ===== start of EditorWorkspace class =====
42 // =====
43
56 class Brood::Application::EditorWorkspace :
57     public Brood::Application::WorkSpace
58 {
59     // ===== public member variables =====
60 public:
61     // default constructor
62     EditorWorkspace( Brood::Application::Components::GameDataManager* aGameData );
63
64     // default destructor
65     virtual ~EditorWorkspace();
66
67     // ===== virtual funcitons ====
68
70     void InitializeWorkSpace();
71
73     void Update();
74
75     // updates all the display elment
76     void UpdateAllDispalyElement();
77
79     void Draw( sf::RenderWindow& a_window );
80
82     void Debugger();
83
84     // setter function for game data
85     void SetGameManager( Brood::Application::Components::GameDataManager* aGameManagerPtr );
86
87     // ===== private member function =====
88 private:
89
91     void InitializeEditModeTabs();
92
94     void UpdateEditModeTabs();
95
96     // updates the edit Tab and selects the proper tab
```

```

97     void UpdateActiveEditorIdx( unsigned a_idx );
98     // ====== private member variables ======
100    private:
101
103    sf::RectangleShape m_sidePanel;
104
106    unsigned m_activeEditorIdx;
107
109    std::vector<Brood::BroodUI::Button*> m_editModesTabs;
110
112    std::vector<Brood::Application::WorkSpace*> m_editorWorkspaceList;
113 };
114
115 // =====-
116 // ===== end of EditorWorkspace class =====-
117 // =====-

```

10.49 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/ElementSelection.cpp File Reference

This file is a source file for [ElementSelection](#) class which is in BroodUI namespace.

```
#include "stdafx.h"
#include "ElementSelection.h"
```

10.49.1 Detailed Description

This file is a source file for [ElementSelection](#) class which is in BroodUI namespace.

It contains all of the declaration of the static data member of [ElementSelection](#) class.

10.50 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/ElementSelection.h File Reference

This file is a header file for [ElementSelection](#) class.

```
#include "UI_ID.h"
```

Classes

- class [Brood::BroodUI::ElementSelection](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.50.1 Detailed Description

This file is a header file for [ElementSelection](#) class.

This class store stores the currently active element, hot element, and element which on the verge of being selected

It contains all of the declaration of the member funciton of [ElementSelection](#) class.

10.51 ElementSelection.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ****
3 // =====
4 // ===== included files =====
5 // =====
6 // =====
7 // =====
8 // =====
9 #pragma once
10 #include "UI_ID.h"
11
12 // =====
13 // ===== defining namespace =====
14 // =====
15 // =====
16 // =====
17 // =====
18
19 namespace Brood
20 {
21     namespace BroodUI
22     {
23         class ElementSelection;
24         class Id;
25     }
26 }
27 // =====
28 // ===== end of namespace defination =====
29 // =====
30 // =====
31 // =====
32 // =====
33 // =====
34 // ===== start of ElemnetSelection class =====
35 // =====
36 // =====
37 // =====
38 // =====
39 // =====
40
41 class Brood::BroodUI::ElementSelection
42 {
43     // ===== public member function =====
44 public:
45     // ===== setter functions =====
46     // setter function to set the hot element Id pointer
47     static void SetHotElementIdPtr( Brood::BroodUI::Id* a_idPtr );
48
49     // setter function to set the almost active element Id pointer
50     static void SetAlmostActiveElementIdPtr( Brood::BroodUI::Id* a_idPtr );
51
52     // setter function to set the current active element Id pointer
53     static void SetCurrActiveElementIdPtr( Brood::BroodUI::Id* a_idPtr );
54
55     // setter function to set the last active element Id pointer
56     static void SetLastActiveElementIdPtr( Brood::BroodUI::Id* a_idPtr );
57
58     // setter function to set the hot element Id pointer Flag
59     static void SetHotElementIdFlag( bool a_found );
60
61     // ===== setter functions =====
62
63     // getter function to get the hot element Id pointer
64     static Brood::BroodUI::Id* GetHotElementIdPtr();
65
66     // getter function to get the almost active element Id pointer
67     static Brood::BroodUI::Id* GetAlmostActiveElementIdPtr();
68
69     // getter function to get the current active element Id pointer
70     static Brood::BroodUI::Id* GetCurrActiveElementIdPtr();
71
72     // getter function to get the last active element Id pointer
73     static Brood::BroodUI::Id* GetLastActiveElementIdPtr();
74
75     // getter function to get the hot element Id pointer Flag
76     static bool GetHotElementIdFlag();
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98

```

```

99
100 // debugging
101 static void Debugger();
102 static void DebugPrintSelectedElementID();
103
104 // ===== private member variables =====
105 private:
106
108     static Brood::BroodUI::Id* m_hotElementIdPtr;
109
111     static bool m_hotElementFlag;
112
114     static Brood::BroodUI::Id* m_almostActiveElementIdPtr;
115
119     static Brood::BroodUI::Id* m_currActiveElementIdPtr;
120
122     static Brood::BroodUI::Id* m_lastActiveElementIdPtr;
123 };
124
125 // =====
126 // ===== end of ElemenetSelection class =====
127 // =====

```

10.52 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/FileAccess.cpp File Reference

```
#include "stdafx.h"
#include " FileAccess.h"
```

10.53 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/FileAccess.h File Reference

This file is a header file for FileAccess class.

```
#include "GlobalVariables.h"
```

Classes

- class [Brood::Application::FileAccess](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.53.1 Detailed Description

This file is a header file for FileAccess class.

It contains all of the declaration of the member funciton of FileAccess class. It contains all of the inline funcitons defination of the member funciton of FileAccess class

10.54 FileAccess.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // **** included files ****
3 // ****
4 // =====
5 // ===== defining namespace =====
6 // =====
7
8 #pragma once
9 #include "GlobalVariables.h"
10
11 // =====
12 // ===== defining namespace =====
13 // =====
14 // =====
15 // ===== included files =====
16 // =====
17
18 // =====
19 // =====
20
21 // ===== defining namespace =====
22 // ===== defining namespace =====
23 // =====
24 namespace Brood
25 {
26     namespace Application
27     {
28         class FileAccess;
29     }
30 }
31 // =====
32 // ===== end of namespace definition =====
33 // =====
34
35 // =====
36 // ===== start of FileAccess class =====
37 // =====
38
39 class Brood::Application::FileAccess
40 {
41     // ===== public member function =====
42 public:
43     // constructor
44     FileAccess();
45
46     // destructor
47     ~FileAccess();
48
49     // opens the passed file
50     bool OpenFile( std::string a_fullfileName );
51
52     // creates a file and returns the file
53     bool CreateFile( std::string a_fullfileName );
54
55     // gets next line form the file
56     bool GetNextLine( std::string& a_line );
57
58     // goes back to the beginning of the file
59     void Rewind();
60
61     // removes all content form the file and goes to
62     // beginning of the file
63     void RemoveAllContent();
64
65     // appends a line to the end
66     void WriteOneLineToFile( const std::string& a_line );
67
68     // checks if the eof reached or not
69     bool CheckEOF();
70
71     // ===== private member functions =====
72 private:
73
74     std::fstream m_fileHandler;
75
76     std::filesystem::path m_filePath;
77 };
78
79 // =====
80 // ===== end of FileAccess class =====
81 // =====

```

10.55 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GameDataManager.cpp File Reference

This file is a source file for [GameDataManager](#) class.

```
#include "stdafx.h"
#include "GameDataManager.h"
```

10.55.1 Detailed Description

This file is a source file for [GameDataManager](#) class.

It also contains all of the declaration of the [GameDataManager](#) class's member function.

10.56 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GameDataManager.h File Reference

This file is a header file for [GameDataManager](#) class.

```
#include "Board.h"
#include "Dice.h"
#include "DeckManager.h"
#include "PlayerManager.h"
#include "DisplayCard.h"
```

Classes

- class [Brood::Application::Components::GameDataManager](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.56.1 Detailed Description

This file is a header file for [GameDataManager](#) class.

It contains all of the declaration of the member funciton of [GameDataManager](#) class.

10.57 GameDataManager.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // **** included files ****
11
12 // =====
13 // ===== defining namespace =====
14 // =====
15 #pragma once
16 #include "Board.h"
17 #include "Dice.h"
18 #include "DeckManager.h"
19 #include "PlayerManager.h"
20 #include "DisplayCard.h"
21
22 // =====
23 // ===== end of namespace definition =====
24 // =====
25 namespace Brood::Application
26 {
27     namespace Components
28     {
29         class GameDataManager;
30     }
31 }
32 // =====
33 // ===== end of namespace defintion =====
34 // =====
35
36 // =====
37 // ===== start of GameDataManager class =====
38 // =====
39
45 class Brood::Application::Components::GameDataManager
46 {
47     // ===== public member function =====
48 public:
49
50     // default constructor
51     GameDataManager();
52
53     // default destructor
54     ~GameDataManager();
55
56     // initializes the gameData
57     void InitializeGameDataManger();
58
59     void Draw( sf::RenderWindow& a_window );
60
61     void Debugger();
62
63     // initialize the game Data Manager
64     void InitializeGameManager( Brood::Application::Data::ST_GameData& a_gameDataManager );
65
66     // returns the game Data manager data to save
67     Brood::Application::Data::ST_GameData GetDataToSave();
68
69     // appends the the game data manager data to passed file
70     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
71
72     // appends the the game data manager data to passed file
73     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr );
74
75     // ===== setter functions =====
76
77     void SetGameTitle( std::string a_gameTitle );
78
79     // ===== getter functions =====
80
81     std::string GetGameTitle();
82
83     Brood::Application::Components::Board* GetBoardPtr();
84
85     Brood::Application::Components::Dice* GetDicePtr();
86
87     Brood::Application::Components::PlayerManager* GetPlayerManagerPtr();
88
89     Brood::Application::Components::DeckManager* GetDeckManagerPtr();
90
91     Brood::Application::Components::DisplayCard* GetDisplayCardPtr();
92
93
94     // ===== private member variables =====
95
96     private:
97
98
99
100
101
102
103
104

```

```

105     std::string m_gameTitle;
106     Brood::Application::Components::Board m_board;
107     Brood::Application::Components::Dice m_dice;
108     Brood::Application::Components::PlayerManager m_playerManager;
109     Brood::Application::Components::DeckManager m_deckManager;
110
111     // cardinfo
112     Brood::Application::Components::DisplayCard m_displayCard;
113 };
114
115 // =====
116 // ===== end of GameDataManager class =====
117 // =====

```

10.58 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GameEditor.cpp File Reference

This file is a source file for GameEditor class.

```
#include "stdafx.h"
#include "GameEditor.h"
```

10.58.1 Detailed Description

This file is a source file for GameEditor class.

It also contains all of the declaration of the GameEditor class's member function.

10.59 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GameEditor.h File Reference

This file is a header file for GameEditor class.

```
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::GameEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.59.1 Detailed Description

This file is a header file for GameEditor class.

It contains all of the declaration of the member function of GameEditor class.

10.60 GameEditor.h

[Go to the documentation of this file.](#)

```
1 // ****
10 // ****
11
12 // =====
13 // ===== included files =====
14 // =====
15 #pragma once
16 #include "WorkSpace.h"
17
18
19 // =====
20 // ===== defining namespace =====
21 // =====
22 namespace Brood
23 {
24     namespace Application
25     {
26         class GameEditor;
27     }
28 }
29 // =====
30 // ===== end of namespace defination =====
31 // =====
32
33 // =====
34 // ===== start of GameEditor class =====
35 // =====
36
51 class Brood::Application::GameEditor :
52     public Brood::Application::WorkSpace
53 {
54     // ===== public member function =====
55 public:
56     // default constructor
57     GameEditor( Brood::Application::Components::GameManager* aGameData,
58                 sf::RectangleShape* a_panelPtr );
59
60     // default destructor
61     ~GameEditor();
62     // ===== virtual funcitons ====
63
64     // initializes the component of a workspace
65     void InitializeWorkSpace();
66
67     // all the logic for the workspace is here
68     void Update();
69
70     // updates all the display elment
71     void UpdateAllDispalyElement();
72
73     // renders to the screen
74     void Draw( sf::RenderWindow& a_window );
75
76     // Calls Debugger for all UI elements in this class
77     void Debugger();
78
79     // ===== private member function =====
80 private:
81
82     // =====
83     // ===== Update funcitons for general game setting =====
84     // =====
85
86     // checks if the user interacted with the enter game
87     // name panel element
88     void UpdateGameTitleInput();
89
90     // checks if the user interacted with the movement
91     // type panel Element
```

```
92     void UpdateMovementTypePanelElement();
93
94     // checks if the user interacted with the incorrect
95     // penalty panel Element
96     void UpdateIncorrectPenaltyPanelElement();
97
98     // checks if the user interacted with the minumum
99     // player number panel Element
100    void UpdateMinimumPlayerNumber();
101
102    // checks if the user interacted with the maximum
103    // player number panel Element
104    void UpdateMaximumPlayerNumber();
105
106    // checks if the user interacted with the Deck
107    // number panel Element
108    void UpdateDeckNumber();
109
110    // ===== private member variable =====
111 private:
112
114     sf::RectangleShape* m_panelBodyPtr;
115
117     Brood::BroodUI::Button* m_titleScreenBtn;
118
119     // =====
120     // ===== member variables for general game setting =====
121     // =====
122
123     // === UI elements to show setting title ===
124     Brood::BroodUI::TextBox* m_txtSettingTitle;
125
126     // === UI elements to enter a game title name ===
127     Brood::BroodUI::TextBox* m_txtGameNamePrompt;
128
129     Brood::BroodUI::TextBox* m_txtGameInput;
130
131
132     // === UI elements to change the board movement rule ===
133
134     Brood::BroodUI::TextBox* m_txtMovementTypePromt;
135
136     Brood::BroodUI::DropDownInput* m_ddiMovementType;
137
138
139     // === UI elements to change the board Penalty rule ===
140
141     Brood::BroodUI::TextBox* m_txtIncorectPenaltyPromt;
142
143     Brood::BroodUI::DropDownInput* m_ddiIncorectPenalty;
144
145
146     // === UI elements to change the minimum player ===
147
148     Brood::BroodUI::TextBox* m_txtPlayerMinNumPrompt;
149
150     Brood::BroodUI::Button* m_btnPlayerDecMinNum;
151
152     Brood::BroodUI::TextBox* m_txtPlayerMinNum;
153
154     Brood::BroodUI::Button* m_btnPlayerIncMinNum;
155
156
157     // === UI elements to change the maximum player ===
158
159     Brood::BroodUI::TextBox* m_txtPlayerMaxNumPrompt;
160
161     Brood::BroodUI::Button* m_btnPlayerDecMaxnNum;
162
163     Brood::BroodUI::TextBox* m_txtPlayerMaxNum;
164
165     Brood::BroodUI::Button* m_btnPlayerIncMaxNum;
166
167
168     // === UI elements to change the number of decks ===
169
170     Brood::BroodUI::TextBox* m_txtDeckNumPrompt;
171
172     Brood::BroodUI::Button* m_btnDeckDecNum;
173
174     Brood::BroodUI::TextBox* m_txtDeckNum;
175
176     Brood::BroodUI::Button* m_btnDeckIncNum;
177
178
179
180     Brood::BroodUI::TextBox* m_txtDeckNum;
181
182     Brood::BroodUI::Button* m_btnDeckDecNum;
183
184     Brood::BroodUI::TextBox* m_txtDeckNum;
185
186     Brood::BroodUI::Button* m_btnDeckIncNum;
187
188
189
190
191
192
193 };
194
195 // =====
196 // ===== start of GameEditor class =====
197 // =====
```

10.61 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GlobalVariables.cpp File Reference

This file is a source file for GlobalVariables stuct and ColorVariable struct.

```
#include "stdafx.h"
#include "GlobalVariables.h"
```

10.61.1 Detailed Description

This file is a source file for GlobalVariables stuct and ColorVariable struct.

It contains all of the location declaration for the member the structs.

10.62 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/GlobalVariables.h File Reference

This file is a header file for color variables.

Classes

- struct [Brood::Application::StaticVariables::ST_ColorVariables](#)
- struct [Brood::Application::StaticVariables::ST_GlobalCoreVariables](#)
- struct [Brood::Application::StaticVariables::ST_Folders](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::StaticVariables](#)

10.62.1 Detailed Description

This file is a header file for color variables.

It contains all of the variables and defines for the color variables

10.63 GlobalVariables.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // **** included files ****
11
12 // =====
13 // ===== defining namespace =====
14 // =====
15
16 #pragma once
17
18 // =====
19 // ===== defining namespace =====
20 // =====
21 namespace Brood::Application
22 {
23     namespace StaticVariables
24     {
25         struct ST_ColorVariables;
26         struct ST_GlobalCoreVariables;
27         struct ST_Folders;
28     }
29 }
30
31 // =====
32 // ===== end of namespace defination =====
33 // =====
34 // ===== start of ST_ColorVariables struct =====
35 // =====
36
37 // =====
38 // ===== start of ST_GlobalCoreVariables struct =====
39 // =====
40
41 struct Brood::Application::StaticVariables::ST_ColorVariables
42 {
43     // ===== black and white color =====
44
45     static sf::Color stm_Black;
46
47     static sf::Color stm_White;
48
49     // ===== overlay colors =====
50
51     static sf::Color stm_HotOverlay;
52
53     static sf::Color stm_CurrActiveOverlay;
54
55     // ===== other colors =====
56
57     static sf::Color stm_MainMenu;
58
59     static sf::Color stm_AppPrimaryColor;
60
61     static sf::Color stm_AppSecondaryColor;
62
63     static sf::Color stm_ErrorColor;
64
65     static sf::Color GetRandomColor();
66
67 };
68
69 // =====
70 // ===== end of ST_ColorVariables struct =====
71 // =====
72
73 // =====
74 // ===== start of ST_GlobalCoreVariables struct =====
75 // =====
76
77 struct Brood::Application::StaticVariables::ST_GlobalCoreVariables
78 {
79     static uint32_t stm_window_height;
80
81     static uint32_t stm_window_width;
82
83     static sf::Font stm_font;
84
85     static bool stm_is_debug_mode;
86
87     static bool stm_exit;
88
89     static uint32_t stm_panelPercentage;
90
91     static unsigned stm_seed;
92
93 };
94
95 // =====

```

```
122 // ===== end of ST_GlobalCoreVariables struct =====
123 // =====
124
125
126
127 // =====
128 // ===== start of ST_Folders struct =====
129 // =====
130
136 struct Brood::Application::StaticVariables::ST_Folders
137 {
139     static std::filesystem::path stm_cwd;
140
142     static std::filesystem::path stm_assets;
143
145     static std::filesystem::path stm_fonts;
146
148     static std::filesystem::path stm_textures;
149
151     static std::filesystem::path stm_diceTextures;
152
154     static std::filesystem::path stm_playerTextures;
155
157     static std::filesystem::path stm_tileTextures;
158
160     static std::filesystem::path stm_deckTextures;
161
163     static std::filesystem::path stm_cardTextures;
164
166     static std::filesystem::path stm_data;
167 };
168
169 // =====
170 // ===== end of ST_Folders struct =====
171 // =====
172
173
```

10.64 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/HomeWorkSpace.cpp File Reference

```
#include "stdafx.h"
#include "HomeWorkSpace.h"
#include <filesystem>
```

10.65 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/HomeWorkSpace.h File Reference

This file is a header file for InitialWorkSpace class.

```
#include "EditorWorkspace.h"
#include " FileAccess.h"
```

Classes

- class [Brood::Application::HomeWorkSpace](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.65.1 Detailed Description

This file is a header file for InitialWorkSpace class.

It contains all of the declaration of the member funciton of InitialWorkSpace class.

10.66 HomeWorkSpace.h

[Go to the documentation of this file.](#)

```

1 // ****
11 // ===== included files =====
12 // =====
13 // ===== defining namespace =====
14 // ===== start of InitialWorkSpace class =====
15 // =====
16 #pragma once
17 #include "EditorWorkspace.h"
18 #include "FileAccess.h"
19
20 // =====
21 // ===== end of namespace defination =====
22 // =====
23 namespace Brood
24 {
25     namespace Application
26     {
27         // defining what component of this file will be inside namespace BroodUI
28         class HomeWorkSpace;
29     }
30 }
31 // =====
32 // ===== end of namespace defination =====
33 // =====
34
35 // =====
36 // ===== start of InitialWorkSpace class =====
37 // =====
38
65 class Brood::Application::HomeWorkSpace :
66     public Brood::Application::WorkSpace
67 {
68     // ===== public member variables =====
69 public:
70     // default constructor
71     HomeWorkSpace();
72
73     // default destructor
74     ~HomeWorkSpace();
75
76     // override functions
77     virtual void InitializeWorkSpace() override;
78
79     // all the logic for the workspace is here
80     virtual void Update();
81
82     // updates all the display elment
83     void UpdateAllDisplayElement();
84
85     // draw to the screen
86     virtual void Draw( sf::RenderWindow& a_window );
87
88     // debugger
89     virtual void Debugger();
90
91     // ===== private member function =====
92 private:
93
94     // Initialize the Ribbon Tabs
95     void InitializeRibbonTabs();
96
97     // checks if the ribbon tabs was pressed
98     void UpdateRibbonTabs();
99
100    // executes the menu item at the passed indexs
101    void ExecuteMenuItem( unsigned a_iIdx, unsigned a_jIdx );
102
103    // executes the logic for homeworkspace excluding the main menu
104    void UpdateHomeWorkspace();

```

```
105 // executes the logic for load game data
106 void UpdateFileNameFeildDialog();
107
108 // Swtichs to home workspace if it is not in it
109 void SwitchToHomeWorkspace();
110
111 // opening a dialog box so that user can enter file name to
112 // load previous edit
113 void ExecuteLoadPreviousEditDialogBox();
114
115 // opening a dialog box so that user can enter file name to
116 // create new edit
117 void ExecuteCreateNewEditDialogBox();
118
119 // resets the game data
120 void ResetGameData();
121
122 // saves the game data to the file
123 void SaveGameData();
124
125 // load the game data from the file
126 bool LoadGameData( std::filesystem::path a_gameOrEditorFile );
127
128 // ===== private member variables =====
129
130 private:
131
132 // save what to do when openfile button is pressed
133 // it currently has 2 states: open game file and open edit file
134 unsigned m_openFileBtnIdx;
135
136 Brood::BroodUI::MenuBar m_ribbonTabs;
137 sf::RectangleShape m_gameOrEditorForeground;
138
139 Brood::BroodUI::Button* m_gameOrEditorBtn;
140
141 Brood::BroodUI::Button* m_toggleGameInfoLeftBtn;
142
143 Brood::BroodUI::Button* m_toggleGameInfoRightBtn;
144
145 int m_currGameInfoIdx;
146
147 bool m_showImportDialogBox;
148
149 Brood::BroodUI::Button* m_importGameBtn;
150
151 Brood::BroodUI::Button* m_loadPreviousEditsBtn;
152
153 Brood::BroodUI::Button* m_createNewEditBtn;
154
155 Brood::BroodUI::Button* m_cancleBtn;
156
157 bool m_showEnterFilenameDialogBox;
158
159 Brood::BroodUI::TextBox* m_fileNameTxt;
160
161 Brood::BroodUI::TextBox* m_enterFileNameFeildTxt;
162
163 Brood::BroodUI::Button* m_openFileBtn;
164
165 std::vector<std::string> m_gameInfoFilePathList;
166
167 std::vector<sf::Texture> m_gameListTexture;
168
169 std::vector<Brood::Application::WorkSpace*> m_workSpacesList;
170
171 unsigned m_currWorkSpaceIdx;
172
173 bool m_isHomeWorkspace;
174
175 };
176
177 // ===== end of InitialWorkSpace class =====
178
179 // =====
```

10.67 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Main.cpp File Reference

This file is conatains the main function.

```
#include "stdafx.h"
#include "Application.h"
```

Functions

- int [main \(\)](#)

10.67.1 Detailed Description

This file is conatains the main function.

10.67.2 Function Documentation

10.67.2.1 [main\(\)](#)

```
int main ( )
```

10.68 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/MapIdToElement.cpp File Reference

This file is a source file for [MapIdToElement](#) class.

```
#include "stdafx.h"
#include "MapIdToElement.h"
```

10.68.1 Detailed Description

This file is a source file for [MapIdToElement](#) class.

It contains all of the defination of the member funciton of [MapIdToElement](#) class.

10.69 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/MapIdToElement.h File Reference

This file is a header file for [MapIdToElement](#) which are declared in BroodUI namespace.

```
#include "UI_ID.h"
#include "UIElement.h"
```

Classes

- class [Brood::BroodUI::MapIdToElement](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.69.1 Detailed Description

This file is a header file for [MapIdToElement](#) which are declared in BroodUI namespace.

[MapIdToElement](#) is a static class that maps the unique ui element id to its element.

It contains all of the declaration of the member funciton of [MapIdToElement](#) class.

10.70 MapIdToElement.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // **** included files ****
3 // ****
4 // **** defining namespace ****
5 // ****
6 // ****
7 // ****
8 // ****
9 // ****
10 // ****
11 // ****
12 // ****
13 // ****
14 // ****
15 // ****
16 // ****
17 // ****
18 // ****
19 // ****
20 #pragma once
21 #include "UI_ID.h"
22 #include "UIElement.h"
23 // ****
24 // ****
25 // ****
26 // ****
27 namespace Brood
28 {
29     namespace BroodUI
30     {
31         class MapIdToElement;
32         class UIElement;
33     }
34 }
35 // ****
36 // ****
37 // ****
38 // ****
39 // ****
40 // ****
41 // ****
42 // ****
43 class Brood::BroodUI::MapIdToElement
```

```

50 {
51     // ====== public member function ======
52 public:
53     // ===== Getter functions =====
54
55     // Gets a reference to the map
56     static std::map<const int, Brood::BroodUI::UIElement*>& GetMap();
57
58     // Gets a pointer to the UIElement from the map
59     // using the passed UI Id
60     static Brood::BroodUI::UIElement* GetElementPtrFromMap( int a_id );
61
62     // Adds a pointer to the UIElement to the map using
63     // the passed UI Id as the key
64     static bool AddToMap( int a_id, Brood::BroodUI::UIElement* a_elementPtr );
65
66     // Removes the a pointer to the UIElement from the map using
67     // the passed UI Id as the key
68     static bool RemoveFromMap( int a_id );
69
70     // ===== private member variables =====
71 private:
72     static std::map<const int, Brood::BroodUI::UIElement*> s_tm_mapper;
73 };
74
75 // =====
76 // ===== end of MapIdToElement class =====
77 // =====
78 // =====

```

10.71 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/MenuBar.cpp File Reference

This file is a source file for [MenuBar](#) class.

```
#include "stdafx.h"
#include "MenuBar.h"
```

10.71.1 Detailed Description

This file is a source file for [MenuBar](#) class.

It contains all of the definition of the member function of [MenuBar](#) class.

10.72 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/MenuBar.h File Reference

This file is a header file for [MenuBar](#) class.

```
#include "DropDownMenu.h"
```

Classes

- class [Brood::BroodUI::MenuBar](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

10.72.1 Detailed Description

This file is a header file for [MenuBar](#) class.

It contains all of the declaration of the member function of [MenuBar](#) class.

10.73 MenuBar.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ===== included files =====
3 // ===== defining namespace =====
4 // ===== start of MenuBar class =====
5 // =====
6
7 #pragma once
8
9 #include "DropDownMenu.h"
10
11
12
13 // =====
14 // ===== end of namespace definition =====
15 // =====
16
17 namespace Brood
18 {
19     namespace BroodUI
20     {
21         class MenuBar;
22     }
23 }
24
25 // =====
26 // ===== end of namespace defination =====
27 // =====
28
29 // =====
30 // ===== start of MenuBar class =====
31 // =====
32
33 // =====
34 // ===== start of MenuBar class =====
35 // =====
36
37 class Brood::BroodUI::MenuBar : public Brood::BroodUI::UIElement
38 {
39     // ===== public member function =====
40 public:
41     // default constructor
42     MenuBar( Brood::BroodUI::UIElement* a_parentPtr = nullptr );
43
44     // default destructor
45     virtual ~MenuBar();
46
47     // copy constructor
48     MenuBar( const MenuBar& a_otherElement );
49
50     // assignment operator
51     Brood::BroodUI::MenuBar& operator=( const MenuBar& a_otherElement );
52
53     // ===== getter funciton =====
54
55     // gets the menu list
56     std::vector<Brood::BroodUI::DropDownMenu*>& GetMenuList();
57
58     // ===== setter funciton =====
59
60     // sets the menu's body size
61     void SetBodySize( sf::Vector2f a_eachItemSize ) override;
62
63     // sets the menu's body size
64     void SetBodySize( float a_itemSizeX, float a_itemSizeY ) override;
65
66     // sets the menu's body position
67     void SetBodyPosition( sf::Vector2f a_pos, bool a_relativeToParent = false ) override;
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158

```

```

159 // // sets the menu's body position
160 void SetBodyPosition( float a_posX, float a_posY, bool a_relativeToParent = false ) override;
161
162 // sets the menu's font
163 void SetFont( sf::Font* a_font );
164
165 // sets font color
166 virtual void SetFontColor( sf::Color a_color =
167     Brood::Application::StaticVariables::ST_ColourVariables::stm_White );
168
169 // sets the font size
170 void SetFontSize( unsigned a_fontSize );
171
172 // sets element body color
173 virtual void SetBodyColor( sf::Color a_bodyColor );
174
175 // sets active overlay color
176 virtual void SetActiveOverlayColor( sf::Color a_color );
177
178 // sets hot overlay color
179 virtual void SetHotOverlayColor( sf::Color a_color );
180
181 // adds menus to the menu bar
182 void AddMenuToMenuBar( std::string a_menuName );
183
184 // adds the passed DropDownMenu at back of menubar
185 // if a_createNew is true then create a new DropDownMenu
186 // using the passed item then adds the new DropDownMenu
187 // to the menubar
188 void AddMenuToMenuBar( Brood::BroodUI::DropDownMenu* a_dropdownMenuPtrToAdd,
189     bool a_createNew = true );
190
191 // adds item to a menu at given index
192 void AddItemToMenu( unsigned a_index, std::string a_menuName );
193
194 // adds the passed button to dropdown menu at passed index
195 // if a_createNew is true then create a new button using
196 // the passed item then adds the new button to the DropDownMenu
197 void AddItemToMenu( unsigned a_index,
198     Brood::BroodUI::Button* a_buttonPtrToAdd,
199     bool a_createNew = true );
200
201 // draw funciton
202 virtual void Draw( sf::RenderWindow& a_window ) override;
203
204 // adds the element ID to the text
205 virtual void Debugger();
206
207 // ===== private member function =====
208 private:
209     // sets the menu's item size correctly
210     void SetMenuBodySize( int a_itemIndex );
211
212     // sets the menu's item position correctly
213     void SetMenuPos( int a_itemIndex );
214
215 private:
216     // ===== private member variables =====
217
218     std::vector<Brood::BroodUI::DropDownMenu*> m_menus;
219 };
220
221 // =====
222 // ===== end of MenuBar class =====
223 // =====
224 // =====

```

10.74 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/MouseHandler.cpp File Reference

This file is a source file for MouseHandler class.

```
#include "stdafx.h"
#include "MouseHandler.h"
#include "UtilityFunctions.h"
```

10.74.1 Detailed Description

This file is a source file for MouseHandler class.

It contains all of the definition of the member function of MouseHandler class.

10.75 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/MouseHandler.h File Reference

This file is a header file for MouseHandler class.

Classes

- class [Brood::MouseHandler](#)

Namespaces

- namespace [Brood](#)

10.75.1 Detailed Description

This file is a header file for MouseHandler class.

It contains all of the declaration of the member function of MouseHandler class. It contains all of the inline functions definition of the member function of MouseHandler class

10.76 MouseHandler.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // **** included files ****
3 // ****
4 // =====
5 // ===== defining namespace =====
6 // =====
7 #pragma once
8
9 // =====
10 // ===== end of namespace definition =====
11 // =====
12 // ===== start of MouseHandler class =====
13 // =====
14 class Brood::MouseHandler
15 {
16     // ===== public member function =====
17 public:
18     enum class ENUM_MouseCursorType
```

```

49     {
50         CURSOR_arrow,
51         CURSOR_wait,
52         CURSOR_text,
53         CURSOR_hand,
54         CURSOR_sizeHorizontal,
55         CURSOR_help,
56         CURSOR_notAllowed
57     };
58
59     // initializes the cursor
60     static void InitializeCursor( sf::RenderWindow& a_window );
61
62     // updates the mouse pos
63     static void UpdateMousePos( sf::RenderWindow& a_window );
64
65     // changes the cursor to the specified cursor
66     static void ChangeCursorTo( sf::RenderWindow& a_window,
67                               const Brood::MouseHandler::ENUM_MouseCursorType a_cursorType );
68
69     // ===== getter function =====
70
71     // gets the mouse position on last frame
72     static sf::Vector2i GetLastFrameMousePos();
73
74     // gets the mouse position on current frame
75     static sf::Vector2i GetCurrFrameMousePos();
76
77     // gets the left mouse button status on last frame
78     // i.e. is it was pressed or not
79     static bool GetlastFrameLMSStatus();
80
81     // gets the left mouse button status on current frame
82     // i.e. is it is pressed or not
83     static bool GetCurrFrameLMSStatus();
84
85     // gets the right mouse button status on last frame
86     // i.e. is if it was pressed or not
87     static bool GetlastFrameRMSStatus();
88
89     // gets the right mouse button status on current frame
90     // i.e. is if it is pressed or not
91     static bool GetCurrFrameRMSStatus();
92
93     //Updates mouse button status and saves the mouse
94     // button status for this frame and last frame
95     static void UpdateMouseButtonStatus();
96
97     // checkes to see if the left mouse button is held or not
98     static bool IsLeftButtonHold();
99
100    // checkes to see if the left mouse button is pressed or not
101    static bool IsLeftButtonPressed();
102
103    // checkes to see if the left mouse button is released or not
104    static bool IsLeftButtonReleased();
105
106    // checkes to see if the right mouse button is held or not
107    static bool IsRightButtonHold();
108
109    // checkes to see if the right mouse button is pressed or not
110    static bool IsRightButtonPressed();
111
112    // checkes to see if the right mouse button is released or not
113    static bool IsRightButtonReleased();
114
115    // debugger
116    static void Debugger();
117
118    // ===== private member variables =====
119 private:
120     // holds the cursor to display
121     static sf::Cursor m_cursor;
122
123     // stores the last frame mouse position
124     static sf::Vector2i m_lastFrameMousePos;
125
126     // stores the current frame mouse position
127     static sf::Vector2i m_currFrameMousePos;
128
129     // holds left mouse button state for last frame, true == was pressed
130     static bool m_lastFrameLeftMouseButtonPressed;
131
132     // holds right mouse button state for last frame, true == was pressed
133     static bool m_lastFrameRightMouseButtonPressed;
134
135     // holds left mouse button state for current frame, true == was pressed

```

```
136     static bool m_currFrameLeftMouseButtonPressed;
137
138     // holds right mouse button state for current frame, true == was pressed
139     static bool m_currFrameRightMouseButtonPressed;
140 };
141
142 // =====
143 // ===== end of MouseHandler class =====
144 // =====
```

10.77 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Path.cpp File Reference

```
#include "stdafx.h"
#include "Path.h"
```

Functions

- bool **operator==** (const Brood::Application::Components::Path &thisPath, const Brood::Application::Components::Path &otherPath)
overloading the == operator

10.77.1 Function Documentation

10.77.1.1 **operator==()**

```
bool operator== (
    const Brood::Application::Components::Path & thisPath,
    const Brood::Application::Components::Path & otherPath )
```

overloading the == operator

Parameters

<i>thisPath</i>	reference to st_path structure which is on left side of the == operator
<i>otherPath</i>	reference to st_path structure which is on right side of the == operator

Returns

10.78 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Path.h File Reference

```
#include "Tiles.h"
#include "Player.h"
#include "Deck.h"
```

Classes

- class [Brood::Application::Components::Path](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

Functions

- bool [operator==\(const Brood::Application::Components::Path &thisPath, const Brood::Application::Components::Path &otherPath\)](#)
overloading the == operator

10.78.1 Function Documentation

10.78.1.1 [operator==\(\)](#)

```
bool operator== (
    const Brood::Application::Components::Path & thisPath,
    const Brood::Application::Components::Path & otherPath )
```

overloading the == operator

Parameters

<code>thisPath</code>	reference to st_path structure which is on left side of the == operator
<code>otherPath</code>	reference to st_path structure which is on right side of the == operator

Returns

10.79 Path.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // **** included files ****
3 // ****
4 // =====
5 // ===== defining namespace =====
6 // =====
7
8 #pragma once
9 #include "Tiles.h"
10 #include "Player.h"
11 #include "Deck.h"
12
13
14 // =====
15 // ===== end of namespace defination =====
16 // =====
17
18
19
20
21
22
23 // =====
24 // ===== start of path class =====
25 // =====
26
27 namespace Brood::Application
28 {
29     namespace Components
30     {
31         class Path;
32         class Player;
33     }
34 }
35
36 // =====
37 // ===== end of namespace defination =====
38
39
40 // =====
41 // ===== start of path class =====
42
43
44
45
46
47
48
49
50
51 class Brood::Application::Components::Path
52 {
53     // ===== public member function =====
54 public:
55     //default constructor
56     Path( Brood::Application::Components::Tiles* a_tilePtr = nullptr,
57           Brood::Application::Components::Deck* a_deckPtr = nullptr,
58           Brood::Application::Components::ENUM_TileType a_tileType =
59           Brood::Application::Components::ENUM_TileType::TILE_blank );
60
61     // destructor
62     ~Path();
63
64     // copy constructor
65     Path( const Path& a_otherPath );
66
67     // assignment operator
68     Path& operator=( const Path& a_otherPath );
69
70     // initialize the path
71     void InitializePath( Brood::Application::Data::ST_PathPrefabData& a_pathData,
72                          Brood::Application::Components::Path* a_nextPathPtr,
73                          Brood::Application::Components::Path* a_bridgeEndPathPtr,
74                          Brood::Application::Components::Deck* a_deckPtr );
75
76     // returns the path data to save
77     Brood::Application::Data::ST_PathPrefabData GetDataToSave();
78
79     // ===== getter funciton =====
80
81     // getter funciton to get the tile type of the tile
82     Brood::Application::Components::ENUM_TileType GetTileType();
83
84     // Getter funciton to get the pointer to the tile
85     Brood::Application::Components::Tiles* GetTilePtr();
86
87     // Getter funciton to get the pointer to the tile
88     const Brood::Application::Components::Tiles* GetTilePtr() const;
89
90     // getter funciton to get the pointer to the next path
91     Brood::Application::Components::Path* GetNextPathPtr();
```

```

91 // getter funciton to get the pointer to the previous path
92 Brood::Application::Components::Path* GetPreviousPathPtr();
93
94 // getter funciton to get the pointer to the bridge end path
95 Brood::Application::Components::Path* GetBridgeEndPathPtr();
96
97 // getter function to get the player list pointer
98 const std::vector<Brood::Application::Components::Player*>& GetPlayerListPtr() const;
99
100 // getter function to get the deck pointer
101 const Brood::Application::Components::Deck* GetDeckPtr() const;
102
103 // getter funciton to get the index of the assinged deck
104 unsigned GetDeckIdx();
105
106 // Getter funciton to get the number of card to be drawn
107 unsigned GetNumCardDraw();
108
109 // getter funciton to get force dice roll
110 bool GetForceDiceRoll();
111
112 // gets the center point of the tile
113 sf::Vector2f GetTileCenter();
114
115 // gets the draw line var
116 bool GetDrawLine();
117
118 // ====== setter funciton ======
119
120 // Setter funciton to Set the tile type of the tile
121 void SetTileType( Brood::Application::Components::ENUM_TileType a_tileType );
122
123 // setter funciton to set the pointer to the tile
124 void SetTilePtr( Brood::Application::Components::Tiles* a_tilePtr );
125
126 // Setter funciton to Set the pointer to the next path
127 void SetNextPathPtr( Brood::Application::Components::Path* a_nextPathPtr );
128
129 // Setter funciton to Set the pointer to the previous path
130 void SetPreviousPathPtr( Brood::Application::Components::Path* a_previousPathPtr );
131
132 // Setter funciton to Set the pointer to the bridge end path
133 void SetBridgeEndPathPtr( Brood::Application::Components::Path* a_bridgeEndPathPtr );
134
135 // setter funciton to set the pointer to the deck
136 void SetDeckPtr( Brood::Application::Components::Deck* a_deckPtr );
137
138 // Setter funciton to Set the index of the assinged deck
139 void SetDeckIdx( unsigned a_deckIdx );
140
141 // Setter funciton to Set the number of card to be drawn
142 void SetNumCardDraw( unsigned a_numCardDraw );
143
144 // Setter funciton to Set force dice roll
145 void SetForceDiceRoll( bool a_diceRoll );
146
147 // toggles the the draw line var
148 void ToggleDrawLine();
149
150 // add player to its list
151 bool AddPlayerToList( Brood::Application::Components::Player* a_playerPtr );
152
153 // deletes the player from its player list
154 bool DeletePlayerFromList( Brood::Application::Components::Player* a_playerPtr );
155
156 // updates path line
157 void UpdatePathLines();
158
159 // draws the tile
160 void Draw( sf::RenderWindow& a_window );
161
162 // draws the tile
163 void DrawPath( sf::RenderWindow& a_window );
164
165 // ====== public member variable ======
166
167 private:
168
169     Brood::Application::Components::ENUM_TileType m_tileType;
170
171     Brood::Application::Components::Tiles* m_tilePtr;
172
173     Brood::Application::Components::Path* m_nextPathPtr;
174
175     Brood::Application::Components::Path* m_previousPathPtr;
176
177     Brood::Application::Components::Path* m_bridgeEndPathPtr;
178
179
180
181
182
183

```

```
184     std::vector<Brood::Application::Components::Player*> m_playerPtrList;
185
186     Brood::Application::Components::Deck* m_deckPtr;
187
188     unsigned m_deckIdx;
189
190     unsigned m_numCardDraw;
191
192     bool m_forceDiceRoll;
193
194     bool m_drawLine;
195
196     sf::VertexArray m_nextPathLine;
197
198     sf::VertexArray m_bridgePathLine;
199
200 };
201
202 // =====
203 // ===== end of path Class =====
204 // =====
205
206 // relation operators
207 // overloading == operator
208 bool operator==( const Brood::Application::Components::Path& thisPath,
209                     const Brood::Application::Components::Path& otherPath );
```

10.80 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Player.cpp File Reference

This file is a source file for [Player](#) class.

```
#include "stdafx.h"
#include "Player.h"
```

10.80.1 Detailed Description

This file is a source file for [Player](#) class.

It contains all of the defination of the member funciton of [Player](#) class.

10.81 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Player.h File Reference

This file is a header file for [Player](#) class.

```
#include "stdafx.h"
#include "Board.h"
#include "Path.h"
#include "Data.h"
```

Classes

- class [Brood::Application::Components::Player](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)
TODO initialize all of the data here to match the default values.
- namespace [Brood::Application::Components](#)

10.81.1 Detailed Description

This file is a header file for [Player](#) class.

It contains all of the declaration of the member function of [Player](#) class. It contains all of the inline functions definition of the member function of [Player](#) class

10.82 Player.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // =====
3 // ===== included files =====
4 // =====
5 #pragma once
6 #include "stdafx.h"
7
8 #include "Board.h"
9 #include "Path.h"
10 #include "Data.h"
11
12 // =====
13 // ===== defining namespace =====
14 // =====
15 namespace Brood::Application
16 {
17     namespace Components
18     {
19         class Player;
20         class Board;
21         class Path;
22     }
23 }
24
25 // =====
26 // ===== end of namespace defintion =====
27 // =====
28
29 // =====
30 // ===== start of Player class =====
31 // =====
32 class Brood::Application::Components::Player
33 {
34     // ===== public member function =====
35 public:
36     // Default Constructor when path is provided
37     Player( Brood::Application::Components::Path* a_pathPtr = nullptr, unsigned a_positionOffsetX = 0,
38             unsigned a_positionOffsetY = 0 );
39
40     // =====
41     // default destructor
42     ~Player();
43
44     // copy constructor
45     Player( const Player& a_otherPlayer );
46
47     // assignment operator
48     Player& operator=( const Player& a_otherPlayer );
49
50     // initialize the player
51     void InitializePlayer( Brood::Application::Data::ST_PlayerPrefabData& a_playerData,
52                           Brood::Application::Components::Path* a_playerCurrPath );
53
54     // returns the player data to save
55 }
```

```

80     Brood::Application::Data::ST_PlayerPrefabData GetDataToSave();
81
82     // appends the the player data to passed file
83     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
84
85     // appends the the player data to passed file
86     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr,
87                           Brood::Application::Components::Board* a_gameBoard );
88
89     // function to update the path iterator
90     void UpdatePathptr( Brood::Application::Components::Path* a_playerNewPathPtr );
91
92     // setter funciton to set the player's x-size
93     void SetPlayerSizeX( float a_sizeX );
94
95     // setter funciton to set the player's y-size
96     void SetPlayerSizeY( float a_sizeY );
97
98     // setter funciton to set the player's x-offset
99     void SetPositionOffsetX( int a_positionOffsetX );
100
101    // setter funciton to set the player's y-offset
102    void SetPositionOffsetY( int a_positionOffsetY );
103
104    // getter funciton to get the pointer to the tile
105    const Brood::Application::Components::Path* GetCurrPathPtr() const;
106
107    // getter funciton to get a reference to the player body
108    sf::RectangleShape& GetPlayerBody();
109
110    // getter funciton to get a reference to the player sprite body
111    Brood::SpriteHandler& GetPlayerSpriteBody();
112
113    // setter funciton to set the player's x-offset
114    int GetPositionOffsetX();
115
116    // setter funciton to set the player's y-offset
117    int GetPositionOffsetY();
118
119    // draw to screen
120    void Draw( sf::RenderWindow& a_window );
121
122    // ===== private member function =====
123 private:
124     // updates the player position with respect to tile
125     void UpdatePosition();
126
127     // ===== private member variable =====
128 private:
129
130     // player body
131     sf::RectangleShape m_playerBody;
132
133     Brood::SpriteHandler m_spriteBody;
134
135     // holds the x-offset relative to the tile's x position
136     int m_positionOffsetX;
137
138     // holds the x-offset relative to the tile's y position
139     int m_positionOffsetY;
140
141     // pointer that points to the current path node;
142     // it holds all tile that the player is on
143     Brood::Application::Components::Path* m_playerCurrPathPtr;
144 };
145
146 // =====
147 // ===== end of Player class =====
148 // =====

```

10.83 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/PlayerEditor.cpp File Reference

```
#include "stdafx.h"
#include "PlayerEditor.h"
#include "TileEditor.h"
```

10.84 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/PlayerEditor.h File Reference

This file is a header file for [PlayerEditor](#) class.

```
#include "MenuBar.h"
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::PlayerEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.84.1 Detailed Description

This file is a header file for [PlayerEditor](#) class.

It contains all of the declaration of the member funciton of [PlayerEditor](#) class.

10.85 PlayerEditor.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // =====
3 // ===== included files =====
4 // =====
5
6 #pragma once
7 #include "MenuBar.h" // includes all the other uiElements
8 #include "WorkSpace.h"
9
10
11
12 // =====
13 // ===== defining namespace =====
14 // =====
15
16
17
18
19
20
21
22 // =====
23 // ===== end of namespace defination =====
24 // =====
25 namespace Brood
26 {
27     namespace Application
28     {
29         class PlayerEditor;
30     }
31 }
32 // =====
33 // ===== start of PlayerEditor class =====
34 // =====
35
36
37 // =====
38 // =====
39
57 class Brood::Application::PlayerEditor :
```

```
58     public Brood::Application::WorkSpace
59 {
60     // ====== public member function ======
61 public:
62     // default constructor
63     PlayerEditor( Brood::Application::Components::GameDataManager* aGameData,
64                 sf::RectangleShape* a_panelPtr );
65
66     // default destructor
67     ~PlayerEditor();
68     // ===== virtual funcitons ====
69
70     // initializes the component of a workspace
71     void InitializeWorkSpace();
72
73     // all the logic for the workspace is here
74     void Update();
75
76     // updates all the display elment
77     void UpdateAllDispalyElement();
78
79     // reners to the screen
80     void Draw( sf::RenderWindow& a_window );
81
82     // Calls Debugger for all UI elements in this class
83     void Debugger();
84
85     // ===== private member function =====
86 private:
87
88     // =====
89     // ===== update for Player setting =====
90     // =====
91
92     // checks if the user interacted with the start row
93     // number panel Element
94     void UpdateStartRowNumber();
95
96     // checks if the user interacted with the start column
97     // number panel Element
98     void UpdateStartColNumber();
99
100    // checks if the user interacted with the current
101    // selcted player index panel Element
102    void UpdateCurrSelectedPlayerIdx();
103
104    // checks if the user interacted with the player
105    // x size panel Element
106    void UpdateCurrPlayerSizeX();
107
108    // checks if the user interacted with the player
109    // y Offset panel Element
110    void UpdateCurrPlayerSizeY();
111
112    // checks if the user interacted with the player
113    // x size panel Element
114    void UpdateCurrPlayerOffsetX();
115
116    // checks if the user interacted with the player
117    // y Offset panel Element
118    void UpdateCurrPlayerOffsetY();
119
120    // checks if the user interacted with the load
121    // player texture panel Element
122    void UpdateCurrPlayerFileTexture();
123
124    // ===== private member variable =====
125 private:
126
127     sf::RectangleShape* m_panelBodyPtr;
128
129     Brood::Application::Components::Board* m_boardPtr;
130
131     Brood::Application::Components::PlayerManager* m_playerManagerPtr;
132
133
134
135
136
137     // =====
138     // ===== member variables for player setting =====
139     // =====
140
141     // === UI elements to change the start row for player ===
142
143     // === UI elements to show setting title ===
144     Brood::BroodUI::TextBox* m_txtSettingTitle;
145
146     Brood::BroodUI::TextBox* m_txtPlayerStartRowPrompt;
```

```
150     Brood::BroodUI::Button* m_btnPlayerDecStartRowNum;
151     Brood::BroodUI::TextBox* m_txtPlayerStartRowNum;
152     Brood::BroodUI::Button* m_btnPlayerIncStartRowNum;
153     // ---- UI elements to change the start column for player ----
154     Brood::BroodUI::TextBox* m_txtPlayerStartColPrompt;
155     Brood::BroodUI::Button* m_btnPlayerDecStartColNum;
156     Brood::BroodUI::TextBox* m_txtPlayerStartColNum;
157     Brood::BroodUI::Button* m_btnPlayerIncStartColNum;
158     // ===== UI elements to change current selected player Index =====
159     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxPrompt;
160     Brood::BroodUI::Button* m_btnPlayerDecCurrIdx;
161     Brood::BroodUI::TextBox* m_txtPlayerCurrIdx;
162     Brood::BroodUI::Button* m_btnPlayerIncCurrIdx;
163     // ---- UI elements toggle draw player at current index ----
164     Brood::BroodUI::TextBox* m_txtDrawPlayerAtCurrIdxPromt;
165     Brood::BroodUI::DropDownInput* m_ddiDrawPlayerAtCurrIdx;
166     // ---- UI elements to change x-size of current selected player Index ----
167     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxSizeXPrompt;
168     Brood::BroodUI::Button* m_btnPlayerDecCurrIdxSizeX;
169     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxSizeX;
170     Brood::BroodUI::Button* m_btnPlayerIncCurrIdxSizeX;
171     // ---- UI elements to change y-size of current selected y-size Index ----
172     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxSizeYPrompt;
173     Brood::BroodUI::Button* m_btnPlayerDecCurrIdxSizeY;
174     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxSizeY;
175     Brood::BroodUI::Button* m_btnPlayerIncCurrIdxSizeY;
176     // ---- UI elements to change x-offset of current selected player Index ----
177     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxOffsetXPrompt;
178     Brood::BroodUI::Button* m_btnPlayerDecCurrIdxOffsetX;
179     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxOffsetX;
180     Brood::BroodUI::Button* m_btnPlayerIncCurrIdxOffsetX;
181     // ---- UI elements to change y-offset of current selected player Index ----
182     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxOffsetYPrompt;
183     Brood::BroodUI::Button* m_btnPlayerDecCurrIdxOffsetY;
184     Brood::BroodUI::TextBox* m_txtPlayerCurrIdxOffsetY;
185     Brood::BroodUI::Button* m_btnPlayerIncCurrIdxOffsetY;
186     // ---- UI elements to enter a filename texture ----
187     Brood::BroodUI::TextBox* m_txtplayerFileNamePrompt;
188     Brood::BroodUI::TextBox* m_txtPlayerFileInput;
189     Brood::BroodUI::Button* m_btnPlayerOpenFile;
190
191 };
192
193 // ===== end of PlayerEditor class =====
194
195 // =====
```

10.86 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/PlayerManager.cpp File Reference

This file is a source file for [PlayerManager](#) class.

```
#include "stdafx.h"
#include "PlayerManager.h"
```

10.86.1 Detailed Description

This file is a source file for [PlayerManager](#) class.

It contains all of the definition of the member function of [PlayerManager](#) class.

10.87 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/PlayerManager.h File Reference

This file is a header file for [PlayerManager](#) class.

```
#include "stdafx.h"
#include "Player.h"
#include "Board.h"
#include "Path.h"
#include "FileAccess.h"
```

Classes

- class [Brood::Application::Components::PlayerManager](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

- namespace [Brood::Application::Components](#)

10.87.1 Detailed Description

This file is a header file for [PlayerManager](#) class.

It contains all of the declaration of the member function of [PlayerManager](#) class.

10.88 PlayerManager.h

[Go to the documentation of this file.](#)

```

1 // ****
10 // =====
11 // ===== included files =====
12 // =====
13 // ===== defining namespace =====
14 // =====
15 #pragma once
16 #include "stdafx.h"
17 #include "Player.h"
18 #include "Board.h"
19 #include "Path.h"
20 #include "FileAccess.h"
21
22 // =====
23 // ===== end of namespace definition =====
24 // =====
25 namespace Brood::Application
26 {
27     namespace Components
28     {
29         class PlayerManager;
30     }
31 }
32 // =====
33 // ===== end of namespace defintion =====
34 // =====
35
36 // =====
37 // ===== start of PlayerManager class =====
38 // =====
39
40 class Brood::Application::Components::PlayerManager
41 {
42     // ===== public member function =====
43 public:
44
45     // default constructor
46     PlayerManager();
47
48     // destructor
49     ~PlayerManager();
50
51     // copy constructor
52     PlayerManager( const Brood::Application::Components::PlayerManager& a_otherPlayerManager );
53
54     // assignment operator
55     PlayerManager& operator=( const Brood::Application::Components::PlayerManager& a_otherPlayerManager );
56
57     // initializes the player Manger
58     void InitializePlayerManager( Brood::Application::Components::Path* a_pathPtr );
59
60
61     // initialize the player manager
62     void InitializePlayerManager( Brood::Application::Data::ST_PlayerManagerData& a_playerData,
63                                 Brood::Application::Components::Path* a_playerCurrPath );
64
65
66     // returns the PlayerManager data to save
67     Brood::Application::Data::ST_PlayerManagerData GetDataToSave();
68
69     // appends the the PlayerManager data to passed file
70     void SaveDataToFile( Brood::Application::FileAccess* a_fileAccessPtr );
71
72     // appends the the PlayerManager data to passed file
73     void LoadDataFromFile( Brood::Application::FileAccess* a_fileAccessPtr,
74                           Brood::Application::Components::Board* a_gameBoard );
75
76
77     // ===== getter function =====
78
79     // getter funciton to get the maximum player
80     const unsigned GetMaxPlayer() const;
81
82     // getter funciton to get the minimum player
83     const unsigned GetMinPlayer() const;
84
85     // Getter function to Get the curr Active Player
86     const unsigned GetCurrActivePlayerIdx() const;
87
88     // Getter funciton to Get the start path for new additional player
89     const Brood::Application::Components::Path* GetPlayerStartPath() const;
90
91     // getter funciton to get the Iterator to the start of allplayer vector
92
93
94
95

```

```

96     std::vector<Brood::Application::Components::Player*>::iterator GetAllPlayerBegin();
97
98     // getter funciton to get the Iterator to the end of allplayer vector
99     std::vector<Brood::Application::Components::Player*>::iterator GetAllPlayerEnd();
100
101    // ===== setter functions =====
102
103    // setter funciton to set the maximum player
104    void SetMaxPlayer( unsigned a_maxPlayer );
105
106    // setter funciton to set the minimum player
107    void SetMinPlayer( unsigned a_minPlayer );
108
109    // setter function to set the curr Active Player
110    void SetCurrActivePlayerIdx( unsigned a_currActivePlayerIdx );
111
112    // setter funciton to set the start path for new additional player
113    void SetPlayerStartPath( Brood::Application::Components::Path* a_startPathPtr );
114
115    // sets the path for the current active player
116    void SetPathForPlayerAtCurrlIdx( Brood::Application::Components::Path* a_pathPtr );
117
118    // Get the current active player
119    Brood::Application::Components::Player* GetPlayerAtCurrlIdx();
120
121    // Increases the currentPlayer index by 1 andreturns the player
122    // at that index
123    // if currentPlayer index points to the last player, then it loops
124    // back to the first element
125    Brood::Application::Components::Player* GetNextPlayer(); // returns next player in the list
126
127    // replaces the a new player at passed index with a copy of passed index
128    void ReplacaePlayerAt( unsigned a_index,
129                           Brood::Application::Components::Player* a_playerPtr = nullptr );
130
131    // draws the players
132    void Draw( sf::RenderWindow& a_window );
133
134    // ===== private member variables =====
135 private:
136
137    // contains all the player currently playing the game
138    std::vector<Brood::Application::Components::Player*> m_allPlayers;
139
140    // index of current playing player
141    unsigned m_currActivePlayerIdx;
142
143    // maximum number of player
144    unsigned m_maxPlayer;
145
146    // minimum number of player
147    unsigned m_minPlayer;
148
149    // stores the pointer to the path at which the players
150    // are going to start
151    // this is going to be used to create any new player
152    Brood::Application::Components::Path* m_startPathPtr;
153 };
154
155 // =====
156 // ===== end of PlayerManager class =====
157 // =====

```

10.89 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/SpriteHandler.cpp File Reference

This file is a source file for [SpriteHandler](#) class.

```
#include "stdafx.h"
#include "SpriteHandler.h"
```

10.89.1 Detailed Description

This file is a source file for [SpriteHandler](#) class.

It contains all of the defination of the member funciton of [SpriteHandler](#) class.

10.90 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/SpriteHandler.h File Reference

This file is a header file for [SpriteHandler](#) class which is in [Brood](#) namespace.

```
#include "UtilityFunctions.h"
#include "GlobalVariables.h"
```

Classes

- class [Brood::SpriteHandler](#)

Namespaces

- namespace [Brood](#)

10.90.1 Detailed Description

This file is a header file for [SpriteHandler](#) class which is in [Brood](#) namespace.

It contains all of the declaration of the member funciton of BroodSprite class.

10.91 SpriteHandler.h

[Go to the documentation of this file.](#)

```
1
2
3 // ****
4 // **** included files ****
5 // ****
6 // =====
7 // ===== included files =====
8 // =====
9 #pragma once
10 #include "UtilityFunctions.h"
11 #include "GlobalVariables.h"
12
13 // =====
14 // ===== defining namespace =====
15 // =====
16 // ===== end of namespace defination =====
17 // =====
18 // =====
19 // ===== start of BroodSprite class =====
20 // =====
21
22 // =====
23 // ===== defining namespace =====
24 // =====
25 namespace Brood
26 {
27     class SpriteHandler;
28 }
29 // =====
30 // ===== end of namespace defination =====
31 // =====
32
33 // =====
34 // ===== start of BroodSprite class =====
35 // =====
36
37 class Brood::SpriteHandler
38 {
39     // ===== public member function =====
40 public:
41     // default constructor
42     SpriteHandler( sf::RectangleShape* a_body );
43
44     // default destructor
45 }
```

```

73     virtual ~SpriteHandler();
74
75     // copy constructor
76     SpriteHandler( const SpriteHandler& a_otherSpriteHandler );
77
78     // assignment operator
79     SpriteHandler& operator=( const SpriteHandler& a_otherSpriteHandler );
80
81     // ===== setter function =====
82
83     // sets sprite length
84     void SetSpriteLength( float a_spriteLength );
85
86     // sets sprite height
87     void SetSpriteHeight( float a_spriteHeight );
88
89     // loads and sets the texture from provided path
90     bool SetTextureFromFilePath( std::string a_texturePath );
91
92     // loads and sets the texture from provided path
93     bool SetTextureFromFilePath( std::string a_textureDirectoryPath,
94                                 std::string a_textureFileName );
95
96     // loads and sets the texture from saved path
97     bool SetTextureFromSavedFilePath();
98
99     // sets sprite from texture
100    void SetSpriteFromTexture( unsigned a_num );
101
102    // ===== getter function =====
103
104    // getter function to get the texturefilename
105    const std::string GetTextureFileName();
106
107    // getter function to get the file path
108    const std::string GetTextureDirectoryPath();
109
110    // getter function to get the sprite length
111    const float GetSpriteLength();
112
113    // getter function to get the sprite height
114    const float GetSpriteHeight();
115
116    // getter function to get the current sprite index
117    const unsigned GetcurrSpriteIndex();
118
119    // removes texture from the body
120    void RemoveTexture();
121
122    // removes texture from the body for debugging purposes
123    void Debugger();
124
125    // ===== private member variables =====
126 private:
127
128     sf::RectangleShape* m_body;
129
130     std::string m_textureDirectoryPath;
131
132     std::string m_textureFileName;
133
134     sf::Texture m_texture;
135
136     float m_spriteLength;
137
138     float m_spriteHeight;
139
140     unsigned m_currSpriteIndex;
141
142 };
143
144 // ===== end of button class =====
145 // ===== end =====
146
147 // =====
148
149
150 // =====
151 // ===== end of button class =====
152 // =====

```

10.92 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/stdafx.cpp File Reference

This file is a source file that includes the standard includes.

```
#include "stdafx.h"
```

10.92.1 Detailed Description

This file is a source file that includes the standard includes.

Boardmaker.pch will be the pre - compiled header stdafx.obj will contain the pre-compiled type information

10.93 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/stdafx.h File Reference

This file is a include file for include files which are not changed infrequently.

```
#include <stdio.h>
#include <iostream>
#include <vector>
#include <list>
#include <algorithm>
#include <random>
#include <chrono>
#include <limits>
#include <string>
#include <sstream>
#include <filesystem>
#include <fstream>
#include <SFML/Graphics.hpp>
```

Macros

- #define WIN32_LEAN_AND_MEAN

10.93.1 Detailed Description

This file is a include file for include files which are not changed infrequently.

Note

visusal studio uses this as precompile header

10.93.2 Macro Definition Documentation

10.93.2.1 WIN32_LEAN_AND_MEAN

```
#define WIN32_LEAN_AND_MEAN
```

10.94 stdafx.h

[Go to the documentation of this file.](#)

```
1 // ****
10 // =====
11 // ===== included files =====
12 // =====
13 // ===== included files =====
14 // =====
15 #pragma once
16
17 #define WIN32_LEAN_AND_MEAN // Exclude rarely-used stuff from Windows headers
18
19 // standard include files
20 #include <stdio.h>
21 #include <iostream>
22 #include <vector>
23 #include <list>
24 #include <algorithm>
25 #include <random>
26 #include <chrono>
27 #include <limits>
28 #include <string>
29 #include <sstream>
30 #include <filesystem>
31 #include <fstream>
32
33 // sfml include files
34 #include <SFML/Graphics.hpp>
```

10.95 C:/data/OneDrive -

RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.cpp File Reference

This file is a source file for [TextBox](#) class.

```
#include "stdafx.h"
#include "TextBox.h"
```

10.95.1 Detailed Description

This file is a source file for [TextBox](#) class.

It contains all of the definition of the member function of [TextBox](#) class.

10.96 C:/data/OneDrive -

RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TextBox.h File Reference

This file is a header file for [TextBox](#) class which is in BroodUI namespace.

```
#include "UIElement.h"
```

Classes

- class [Brood::BroodUI::TextBox](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

Macros

- #define [DELETE_KEY](#) 8
- #define [ENTER_KEY](#) 13
- #define [ESCAPE_KEY](#) 27

10.96.1 Detailed Description

This file is a header file for [TextBox](#) class which is in BroodUI namespace.

It contains all of the declaration of the member function of [TextBox](#) class.

10.96.2 Macro Definition Documentation

10.96.2.1 [DELETE_KEY](#)

```
#define DELETE_KEY 8
```

10.96.2.2 [ENTER_KEY](#)

```
#define ENTER_KEY 13
```

10.96.2.3 [ESCAPE_KEY](#)

```
#define ESCAPE_KEY 27
```

10.97 TextBox.h

[Go to the documentation of this file.](#)

```

1 // ****
11 // ===== included files =====
12 // =====
13 // =====
14 // ===== defining namespace =====
15 // =====
16 #pragma once
17 #include "UIElement.h"
18
19 // =====
20 // ===== start of namespace =====
21 // =====
22 namespace Brood
23 {
24     namespace BroodUI
25     {
26         class TextBox;
27     }
28 }
29 // =====
30 // ===== end of namespace definition =====
31 // =====
32
33 // =====
34 // ===== start of TextBox class =====
35 // =====
36
155 class Brood::BroodUI::TextBox : public Brood::BroodUI::UIElement
156 {
157     // ===== private defines =====
158 private:
159     // defines for better code readability
160 #define DELETE_KEY 8
161 #define ENTER_KEY 13
162 #define ESCAPE_KEY 27
163
164 // ===== public member function =====
165 public:
166     // default constructor
167     TextBox( Brood::BroodUI::UIElement* a_parentPtr = nullptr,
168             Brood::BroodUI::ENUM_UIType a_enumType = Brood::BroodUI::ENUM_UIType::UI_textBox );
169
170     // default destructor
171     virtual ~TextBox();
172
173     // copy constructor
174     TextBox( const TextBox& a_otherTextBox );
175
176     // assignment operator
177     Brood::BroodUI::TextBox& operator=( const TextBox& a_otherTextBox );
178
179     // Dynamically create a textbox, initializes it, and adds
180     // it to m_unNameList
181     static Brood::BroodUI::TextBox* DyCreateTextBox( sf::Vector2f a_size,
182                                                 sf::Vector2f a_pos,
183                                                 std::string a_text = "",
184                                                 bool a_isEditable = false,
185                                                 std::string a_placeholderText = "",
186                                                 sf::Color a_color =
187                                                 Brood::Application::StaticVariables::ST_ColourVariables::stm_AppPrimaryColor );
188     static Brood::BroodUI::TextBox* DyCreateTextBox( float a_sizeX, float a_sizeY,
189                                                 float a_posX, float a_posY,
190                                                 std::string a_text = "",
191                                                 bool a_isEditable = false,
192                                                 std::string a_placeholderText = "",
193                                                 sf::Color a_color =
194                                                 Brood::Application::StaticVariables::ST_ColourVariables::stm_AppPrimaryColor );
195
196     // ===== getter funcitons =====
197     // checks to see if the text box is editable or not
198     const bool IsEditable() const;
199
200     // check to see if the text box is selected or not
201     const bool IsSelected() const;
202
203     // check to see enter was pressed or not
204     const bool IsEnterPressed() const;
205
206     // ===== setter funcitons =====
207

```

```

208     // sets text
209     void SetText( std::string a_text = "" ) override;
210
211     // sets the limit to number of character
212     void SetLimit( bool a_hasLimit = false, int a_limit = 0 );
213
214     // sets the state of the element.
215     void SetSelected( bool a_selected );
216
217     // sets if the textbox is editable or not
218     void SetEditable( bool a_isEditable );
219
220     // sets enterPressed to false
221     void SetEnterPressedFalse();
222
223     // sets a placeholder Text
224     void SetPlaceHolderText(std::string a_placeHolderText);
225
226     // called every time user enters a char
227     void TypeOn( sf::Event a_input );
228
229     // checks to see if the logic for the element is to
230     // be executed or not
231     virtual bool DoElement() override;
232
233     // draw funciton
234     virtual void Draw( sf::RenderWindow& a_window ) override;
235
236     // debugger
237     virtual void Debugger( ) override;
238
239     // ===== private member function =====
240 private:
241
242     // all of the logic; called on every key press
243     void InputLogic( int charTyped );
244
245     // delete last char
246     void DeleteLastChar();
247
248     // ===== private member variables =====
249 private:
250
251     std::ostringstream m_ossText;
252
253     std::string m_placeHolderText;
254
255     bool m_isEditable;
256
257     bool m_hasLimit;
258
259     int m_limit;
260
261     static bool m_enterPressed;
262 };
263
264 // =====
265 // ===== end of TextBox class =====
266 // =====

```

10.98 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/TileEditor.cpp File Reference

This file is a source file for [TileEditor](#) class.

```
#include "stdafx.h"
#include "TileEditor.h"
```

10.98.1 Detailed Description

This file is a source file for [TileEditor](#) class.

It contains all of the defination of the member funciton of [TileEditor](#) class.

10.99 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/TileEditor.h File Reference

This file is a header file for [TileEditor](#) class.

```
#include "WorkSpace.h"
```

Classes

- class [Brood::Application::TileEditor](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.99.1 Detailed Description

This file is a header file for [TileEditor](#) class.

It contains all of the declaration of the member function of [TileEditor](#) class.

10.100 TileEditor.h

[Go to the documentation of this file.](#)

```
1 // ****
2 // =====
3 // ===== included files =====
4 // =====
5 #pragma once
6 #include "WorkSpace.h"
7
8 // =====
9 // ===== defining namespace =====
10 // =====
11 namespace Brood
12 {
13     namespace Application
14     {
15         class TileEditor;
16     }
17 }
18 // =====
19 // ===== end of namespace definition =====
20 // =====
21
22 // =====
23 // ===== start of TileEditor class =====
24 // =====
25 class Brood::Application::TileEditor :
26     public Brood::Application::WorkSpace
27 {
28     // ===== public member function =====
29
30 public:
```

```

53     // default constructor
54     TileEditor( Brood::Application::Components::GameDataManager* aGameData,
55                 sf::RectangleShape* a_panelPtr );
56
57     // default destructor
58     ~TileEditor();
59     // ===== virtual funcitons =====
60
61     // initializes the component of a workspace
62     void InitializeWorkSpace();
63
64     // all the logic for the workspace is here
65     void Update();
66
67     // updates all the display elment
68     void UpdateAllDispalyElement();
69
70     // renders to the screen
71     void Draw( sf::RenderWindow& a_window );
72
73     // Calls Debugger for all UI elements in this class
74     void Debugger();
75
76     // ===== private member function =====
77 private:
78
79     // checks if the user interacted with the tile row
80     // number panel Element
81     void UpdateTileRowNumber();
82
83     // checks if the user interacted with the tile column
84     // number panel Element
85     void UpdateTileColNumber();
86
87     // checks if the user interacted with the tile type
88     // panel Element
89     void UpdateTileType();
90
91     // checks if the user interacted with the NextPath row
92     // number panel Element
93     void UpdateNextPathRowNumber();
94
95     // checks if the user interacted with the NextPath column
96     // number panel Element
97     void UpdateNextPathColNumber();
98
99     // checks if the user interacted with the EndBridge row
100    // number panel Element
101    void UpdateEndBridgeRowNumber();
102
103    // checks if the user interacted with the EndBridge column
104    // number panel Element
105    void UpdateEndBridgeColNumber();
106
107    // checks if the user interacted with the correct number
108    // of card panel Element
109    void UpdateCorrectCardNumber();
110
111    // checks if the user interacted with the assign deck idx
112    // panel Element
113    void UpdateAssinedDeckIdx();
114
115    // checks if the user interacted with the force dice roll
116    // panel Element
117    void UpdateForceDiceRoll();
118
119    // checks if the user interacted with the load
120    // tile texture panel Element
121    void UpdateTileFileTexture();
122
123    // ===== private member variable =====
124 private:
125
126     sf::RectangleShape* m_panelBodyPtr;
127
128     // =====
129     // ===== member variables for general game setting =====
130     // =====
131
132     // === UI elements to show setting title ===
133     Brood::BroodUI::TextBox* m_txtSettingTitle;
134
135     // === UI elements to change the current tile Row ===
136     Brood::BroodUI::TextBox* m_txtTileCurrRowPromt;
137
138     Brood::BroodUI::Button* m_btnTileDecCurrRow;
139
140
141
142

```

```
143     Brood::BroodUI::TextBox* m_txtTileCurrRow;
144
145     Brood::BroodUI::Button* m_btnTileIncCurrRow;
146
147     // ===== UI elements to change the current tile Col =====
148
149     Brood::BroodUI::TextBox* m_txtTileCurrColPromt;
150
151     Brood::BroodUI::Button* m_btnTileDecCurrCol;
152
153     Brood::BroodUI::TextBox* m_txtTileCurrCol;
154
155     Brood::BroodUI::Button* m_btnTileIncCurrCol;
156
157     // ===== UI elements to change the Tile type =====
158
159     Brood::BroodUI::TextBox* m_txtTileTypePromt;
160
161     Brood::BroodUI::DropDownInput* m_ddiTileType;
162
163     // ===== UI elements to change the NextPath tile Row =====
164
165     Brood::BroodUI::TextBox* m_txtTileNextPathRowPromt;
166
167     Brood::BroodUI::Button* m_btnTileDecNextPathRow;
168
169     Brood::BroodUI::TextBox* m_txtTileNextPathRow;
170
171     Brood::BroodUI::Button* m_btnTileIncNextPathRow;
172
173     // ===== UI elements to change the NextPath tile Col =====
174
175     Brood::BroodUI::TextBox* m_txtTileNextPathColPromt;
176
177     Brood::BroodUI::Button* m_btnTileDecNextPathCol;
178
179     Brood::BroodUI::TextBox* m_txtTileNextPathCol;
180
181     Brood::BroodUI::Button* m_btnTileIncNextPathCol;
182
183     // ===== UI elements to change the EndBridge tile Row =====
184
185     Brood::BroodUI::TextBox* m_txtTileEndBridgeRowPromt;
186
187     Brood::BroodUI::Button* m_btnTileDecEndBridgeRow;
188
189     Brood::BroodUI::TextBox* m_txtTileEndBridgeRow;
190
191     Brood::BroodUI::Button* m_btnTileIncEndBridgeRow;
192
193     // ===== UI elements to change the EndBridge tile Col =====
194
195     Brood::BroodUI::TextBox* m_txtTileEndBridgeColPromt;
196
197     Brood::BroodUI::Button* m_btnTileDecEndBridgeCol;
198
199     Brood::BroodUI::TextBox* m_txtTileEndBridgeCol;
200
201     Brood::BroodUI::Button* m_btnTileIncEndBridgeCol;
202
203     // ===== UI elements to change the tile's NumCorrectCard =====
204
205     Brood::BroodUI::TextBox* m_txtTileNumCorrectCardPromt;
206
207     Brood::BroodUI::Button* m_btnTileDecNumCorrectCard;
208
209     Brood::BroodUI::TextBox* m_txtTileNumCorrectCard;
210
211     Brood::BroodUI::Button* m_btnTileIncNumCorrectCard;
212
213     // ===== UI elements to change the AssignDeckNum tile =====
214
215     Brood::BroodUI::TextBox* m_txtTileAssignedDeckIdxPromt;
216
217     Brood::BroodUI::Button* m_btnTileDecAssignedDeckIdx;
218
219     Brood::BroodUI::TextBox* m_txtTileAssignedDeckIdx;
220
221     Brood::BroodUI::Button* m_btnTileIncAssignedDeckIdx;
222
223     // ===== UI elements to change tile force dice Roll rule =====
224
225     Brood::BroodUI::TextBox* m_txtTileForceDiceRollPromt;
226
227     Brood::BroodUI::DropDownInput* m_ddiTileForceDiceRoll;
```

```

265 // ===== UI elements to enter a Tile filename texture =====
266 Brood::BroodUI::TextBox* m_txtTileBgFileNamePrompt;
267 Brood::BroodUI::TextBox* m_txtTileBgInput;
268 Brood::BroodUI::Button* m_btnTileBgOpenFile;
269
270 };
271
272
273 // ===== end of TileEditor class =====
274 // =====
275
276
277
278
279
280
281

```

10.101 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Tiles.cpp File Reference

This file is a source file for [Tiles](#) class.

```
#include "stdafhx.h"
#include "Tiles.h"
```

10.101.1 Detailed Description

This file is a source file for [Tiles](#) class.

It contains the definition of the member functions of [Tiles](#) class.

10.102 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/Tiles.h File Reference

This file is a header file for [Tiles](#) class.

```
#include "Button.h"
```

Classes

- class [Brood::Application::Components::Tiles](#)

Namespaces

- namespace [Brood](#)
 - namespace [Brood::Application](#)
- TODO initialize all of the data here to match the default values.*
- namespace [Brood::Application::Components](#)

Enumerations

- enum class Brood::Application::Components::ENUM_TileType {
 Brood::Application::Components::TILE_blank , Brood::Application::Components::TILE_tile , Brood::Application::Components::TILE_start ,
 Brood::Application::Components::TILE_end ,
 Brood::Application::Components::TILE_bridge }

10.102.1 Detailed Description

This file is a header file for [Tiles](#) class.

It contains all of the declaration of the member function of [Tiles](#) class. It contains all of the inline functions definition of the member function of [Tiles](#) class

10.103 Tiles.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // =====
3 // ===== included files =====
4 // =====
5 // =====
6 // =====
7 #pragma once
8 #include "Button.h"
9
10 // =====
11 // ===== defining namespace =====
12 // =====
13
14 namespace Brood::Application
15 {
16     namespace Components
17     {
18         enum class ENUM_TileType;
19         class Tiles;
20     }
21 }
22
23
24 // =====
25 // ===== end of namespace defination =====
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44 enum class Brood::Application::Components::ENUM_TileType
45 {
46     TILE_blank,
47     TILE_tile,
48     TILE_start,
49     TILE_end,
50     TILE_bridge
51 };
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77 class Brood::Application::Components::Tiles :
78     public BroodUI::Button
79 {
80     // ====== public member functions ======
81 public:
82
83     // default constructor
84     Tiles();
85

```

```

86     // default destructors
87     ~Tiles();
88
89     // copy constructor
90     Tiles( const Tiles& a_otherTile );
91
92     // assignment constructor
93     Tiles& operator = ( const Tiles& a_other );
94
95
96
97     // getter funciton to set the row and column
98     const sf::Vector2u GetRowAndCol() const;
99
100    // getter funciton to set the row
101    const unsigned GetRow() const;
102
103    // getter funciton to set the column
104    const unsigned GetCol() const;
105
106    // setter funciton to set the row and column
107    void SetRowAndCol( sf::Vector2u a_rowCol );
108
109    // setter funciton to set the row
110    void SetRow( unsigned a_row );
111
112    // setter funciton to set the column
113    void SetCol( unsigned a_col );
114
115
116    // Populates the member variable and sets the position and size of the
117    // tile
118    void UpdateTile( unsigned a_tileRowNum = 0, unsigned a_tileColNum = 0,
119                      float a_tileSizeX = 0.f, float a_tileSizeY = 0.f,
120                      float a_boardPosX = 0.f, float a_boardPosY = 0.f );
121
122    // ===== private member variables =====
123 private:
124
125     unsigned m_tileRowNum;
126
127     unsigned m_tileColNum;
128 };
129
130 // =====
131 // ===== end of Tiles class =====
132 // =====

```

10.104 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UI_ID.cpp File Reference

This file is a source file for `Id` class which is in BroodUI namespace.

```
#include "stdafx.h"
#include "UI_ID.h"
```

10.104.1 Detailed Description

This file is a source file for `Id` class which is in BroodUI namespace.

It contains all of the defination of the member funciton of `Id` class.

10.105 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/UI_ID.h File Reference

This file is a header file for `Id` class which is in BroodUI namespace.

Classes

- class Brood::BroodUI::Id

Namespaces

- namespace Brood
- namespace Brood::BroodUI

10.105.1 Detailed Description

This file is a header file for `Id` class which is in BroodUI namespace.

It contains all of the declaration of the member function of `Id` class.

10.106 UI_ID.h

[Go to the documentation of this file.](#)

```

1 // ****
11 // **** included files ****
12
13 // =====
14 // ===== defining namespace =====
15 // =====
16 #pragma once
17
18 // =====
19 // ===== end of namespace defination =====
20 // =====
21 namespace Brood
22 {
27     namespace BroodUI
28     {
29         class Id;
30     }
31 }
32 // =====
33 // ===== start of ID class =====
34 // =====
35
36 // =====
37 // ===== public member function =====
38 // =====
39
51 class Brood::BroodUI::Id
52 {
53     // ===== public member function =====
54 public:
55     // default constructor
56     Id( Id* a_parentIDPtr = nullptr );
57
58     // default destructor
59     ~Id( );
60
62     // copy constructor
63     Id( const Id& a_otherId );
64
65     // assignment operator
66     Id& operator=( const Id& a_otherId );
67
68     // getter functions
69
70     // returns its parent id;
71     const int GetParentID() const;
72
73     // returns Parent Id Ptr
74     const Brood::BroodUI::Id* GetParentIDPtr() const;
75
76     // returns Parent Id Ptr

```

```

77     Brood::BroodUI::Id* GetParentIDPtr();
78
79     // returns its id
80     const int GetElementID() const;
81
82     // returns the child index in the list
83     const int GetChildIdx( const Brood::BroodUI::Id* a_childIDPtr ) const;
84
85     // returns total number of its child
86     const int GetTotalChildNum() const;
87
88     // returns child at the passed index
89     Id* GetChildIdAtIndex( const int a_index ) const;
90
91     // returns true if it has a child
92     bool HasChild() const;
93
94     // returns true if it has a parent
95     bool HasParent() const;
96
97     // setter functions
98
99     // sets parent
100    void SetParent( Id* a_parentID = nullptr );
101
102    // adds child to its child list
103    void AddChild( Brood::BroodUI::Id* a_childIDPtr );
104
105    // removes the child at passed index
106    void DeleteChildAtIndex( const int a_index );
107
108    // ===== private member variables =====
109 private:
110     static unsigned GLOBAL_ID_NUM;
111
112     Brood::BroodUI::Id* m_parentIDPtr;
113
114     int m_parentID;
115
116     int m_elementID;
117
118     bool m_hasChilds;
119
120     std::vector<Brood::BroodUI::Id*> m_allChildPtrs;
121
122 };
123
124 // ===== end of ID class =====
125 // =====
126
127 // =====
128
129 // =====
130 // ===== end of ID class =====
131 // =====

```

10.107 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/UIElement.cpp File Reference

This file is a source file for [UIElement](#) class and struct [MapIdToElement](#) which is in [BroodUI](#) namespace.

```
#include "stdafx.h"
#include "UIElement.h"
```

10.107.1 Detailed Description

This file is a source file for [UIElement](#) class and struct [MapIdToElement](#) which is in [BroodUI](#) namespace.

It also contains all of the declaration of the [UIElement](#) class's member function. It also contains all of the declaration of the static data member of [MapIdToElement](#) struct.

10.108 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/UIElement.h File Reference

This file is a header file for [UIElement](#) class and it also contains a enum ENUM_UIType, struct ST_UIElement←
CtorParam, and struct [MapIdToElement](#) which are declared in BroodUI namespace.

```
#include "UI_ID.h"
#include "MapIdToElement.h"
#include "ElementSelection.h"
#include "MouseHandler.h"
#include "GlobalVariables.h"
```

Classes

- class [Brood::BroodUI::UIElement](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::BroodUI](#)

Enumerations

- enum class [Brood::BroodUI::ENUM_UIType](#) {
 [Brood::BroodUI::UI_textBox](#) , [Brood::BroodUI::UI_button](#) , [Brood::BroodUI::UI_dropDownMenu](#) , [Brood::BroodUI::UI_dropDown](#) ,
 [Brood::BroodUI::UI_menuBar](#) , [Brood::BroodUI::UI_scrollBar](#) }

10.108.1 Detailed Description

This file is a header file for [UIElement](#) class and it also contains a enum ENUM_UIType, struct ST_UIElement←
CtorParam, and struct [MapIdToElement](#) which are declared in BroodUI namespace.

[UIElement](#) class is a abstract base class that is parent to all the different UI elements.

ENUM_UIType contains the types of UI elements that are derived from [UIElement](#) class

[MapIdToElement](#) is a struct that maps the unique ui element id to its element.

It contains all of the declaration of the member function of [UIElement](#) class, ST_UIElementCtorParam struct, and
[MapIdToElement](#) struct.

10.109 UIElement.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ===== included files =====
3 // =====
4 #pragma once
5 #include "UI_ID.h"
6 #include "MapIdToElement.h"
7 #include "ElementSelection.h"
8 #include "MouseHandler.h"
9 #include "GlobalVariables.h"
10
11 // =====
12 // ===== defining namespace =====
13 // =====
14
15 namespace Brood
16 {
17     namespace BroodUI
18     {
19         enum class ENUM_UIType;
20         class UIElement;
21         class MapIdToElement;
22     }
23
24 // =====
25 // ===== end of namespace defination =====
26 // =====
27
28 // =====
29 // ===== start of ENUM_UIType Enum class defination =====
30 // =====
31
32 enum class Brood::BroodUI::ENUM_UIType
33 {
34     UI_textBox,
35     UI_button,
36     UI_dropDownMenu,
37     UI_dropDownInput,
38     UI_menuBar,
39     UI_scrollBar
40 };
41
42 // =====
43 // ===== end of ENUM_UIType Enum class defination =====
44 // =====
45
46 // =====
47 // ===== start of UIElement class =====
48 // =====
49
50 // =====
51 // ===== start of UIElement class =====
52 // =====
53
54 class Brood::BroodUI::UIElement
55 {
56     // ===== public member function =====
57 public:
58     // default constructor
59     UIElement( Brood::BroodUI::ENUM_UIType a_elementType,
60                 Brood::BroodUI::UIElement* a_parentPtr = nullptr );
61
62     // virtual default destructor
63     virtual ~UIElement();
64
65     // copy constructor
66     UIElement( const UIElement& a_otherElement );
67
68     // assignment operator
69     Brood::BroodUI::UIElement& operator=( const UIElement& a_otherElement );
70
71     // ===== getter functions =====
72
73     // gets element type
74     const Brood::BroodUI::ENUM_UIType GetElementType() const;
75
76     // Get a reference to element body
77     const sf::RectangleShape& GetBody() const;
78
79     // get element body position
80     const sf::Vector2f GetBodyPosition() const;
81
82     // get element body size
83     const sf::Vector2f GetBodySize() const;
84
85     // get element body color
86 }
```

```
135     const sf::Color GetBodyColor();
136
137     // get element body active overlay color
138     const sf::Color GetActiveOverlayColor();
139
140     // get element body hot overlay color
141     const sf::Color GetHotOverlayColor();
142
143     // get element's Id pointer
144     Brood::BroodUI::Id* GetElementIdPtr();
145
146     // return the value of m_isSelected
147     const bool IsSelected() const;
148
149     // returns the text that is stored
150     std::string GetText() const;
151
152     // gets font size
153     unsigned int GetFontSize() const;
154
155     // ====== setter functions ======
156
157     // sets element body size
158     virtual void SetBodySize( sf::Vector2f a_size );
159
160     // sets element body size
161     virtual void SetBodySize( float a_sizeX, float a_sizeY );
162
163     // sets element body position
164     virtual void SetBodyPosition( sf::Vector2f a_pos, bool a_relativeToParent = false );
165
166     // sets element body position
167     virtual void SetBodyPosition( float aPosX, float aPosY, bool a_relativeToParent = false );
168
169     // sets font
170     virtual void SetFont( sf::Font& a_font );
171
172     // sets font color
173     virtual void SetFontColor( sf::Color a_color =
Brood::Application::StaticVariables::ST_ColourVariables::stm_White );
174
175     // sets font size
176     virtual void SetFontSize( int a_fontSize = 12 );
177
178     // sets text
179     virtual void SetText( std::string a_text = "" );
180
181     // sets the state of the element.
182     virtual void SetSelected( bool a_selected );
183
184     // sets element body color
185     virtual void SetBodyColor( sf::Color a_bodyColor );
186
187     // sets active overlay color
188     virtual void SetActiveOverlayColor( sf::Color a_color );
189
190     // sets hot overlay color
191     virtual void SetHotOverlayColor( sf::Color a_color );
192
193     // funciton to check if the mouse is over element
194     bool IsMouseOverElement();
195
196     // funciton to check if element set as almost active element
197     bool IsAlmostActiveElement();
198
199     // funciton to check if element set as active element
200     bool IsCurrActiveElement();
201
202     // funciton to check if element is hot active element
203     bool IsHotElement();
204
205     // checks to see if the logic for the element is to be excecuted or not
206     virtual bool DoElement();
207
208     // draw funciton
209     virtual void Draw( sf::RenderWindow& a_window );
210
211     virtual void Debugger() = 0; // adds the element ID to the text
212
213     // ====== protected member function ======
214 protected:
215
216     // to set the position of the text
217     void SetTextPosition();
218
219     // logic for if overlay is to be drawn or not
220     void SetDrawOverlay();
```

```

221
222     // ====== protected member variables ======
223 protected:
224     Brood::BroodUI::ENUM_UIType m_elementType;
225     Brood::BroodUI::Id m_elementId;
226     sf::RectangleShape m_body;
227     sf::RectangleShape m_bodyOverLay;
228     sf::Color m_hotOverlayColor;
229     sf::Color m_activeOverlayColor;
230     bool m_drawOverlay;
231     bool m_isSelected;
232     sf::Text m_text;
233     std::string m_textContent;
234     std::string m_debugTextSave;
235     sf::Font* m_font;
236     unsigned m_fontSize;
237     bool m_drawText;
238 };
239 // ====== end of UIElement class ======
240 // ======
241

```

10.110 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/UtilityFunctions.cpp File Reference

```
#include "stdafhx.h"
#include "UtilityFunctions.h"
```

10.111 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/UtilityFunctions.h File Reference

Namespaces

- namespace [Brood](#)
- namespace [Brood::UtilityFuncs](#)

It contains all the utility functions for which making a stand alone class would not make sense.

Functions

- bool [Brood::UtilityFuncs::LoadTextureFromFile](#) (sf::Texture &a_texture, const std::string &a_cwd, std::string a_filePath)

It opens a texture form a file and loads it into the texture.

- bool [Brood::UtilityFuncs::LoadTextureFromFile](#) (sf::Texture &a_texture, std::string a_filePath)

It opens a texture form a file and loads it into the texture.

10.112 UtilityFunctions.h

[Go to the documentation of this file.](#)

```

1 #pragma once
2 // #include "windows.h"
3 // #include "commdlg.h"
4
5 namespace Brood
6 {
7     namespace UtilityFuncs
8     {
9         /*#define HEAP_ALLOC(bytes) HeapAlloc(GetProcessHeap(), HEAP_ZERO_MEMORY, bytes);
10 #define HEAP_FREE(ptr) HeapFree(GetProcessHeap(), 0, ptr);/** */
11
12         // a function to load the texture from a file
13         bool LoadTextureFromFile( sf::Texture& a_texture, const std::string& a_cwd, std::string
14             a_filePath );
15         bool LoadTextureFromFile(sf::Texture& a_texture, std::string a_filePath);
16         //static char* showOpenFileDialog( char* descriptionOfFile, char* fileExtension );
17     }
18 }
```

10.113 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/WorkSpace.cpp File Reference

This file is a source file for [WorkSpace](#) class.

```
#include "stdafx.h"
#include "WorkSpace.h"
```

10.113.1 Detailed Description

This file is a source file for [WorkSpace](#) class.

It also contains all of the declaration of the [WorkSpace](#) class's member function.

10.114 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/← BroodMaker/WorkSpace.h File Reference

This file is a header file for [WorkSpace](#) class.

```
#include "MenuBar.h"
#include "DropDownInput.h"
#include "TextBox.h"
#include "GameDataManager.h"
```

Classes

- class [Brood::Application::WorkSpace](#)

Namespaces

- namespace [Brood](#)
- namespace [Brood::Application](#)

TODO initialize all of the data here to match the default values.

10.114.1 Detailed Description

This file is a header file for [WorkSpace](#) class.

It contains all of the declaration of the member funciton of [WorkSpace](#) class.

10.115 WorkSpace.h

[Go to the documentation of this file.](#)

```

1 // ****
2 // ===== included files =====
3 // =====
4 // =====
5 #pragma once
6 #include "MenuBar.h"
7 #include "DropDownInput.h"
8 #include "TextBox.h"
9 #include "GameDataManager.h"
10 // =====
11 // ===== defining namespace =====
12 // =====
13 // ===== end of namespace defination =====
14 // =====
15 namespace Brood
16 {
17     namespace Application
18     {
19         class WorkSpace;
20     }
21 // =====
22 // ===== start of WorkSpace class =====
23 // =====
24 class Brood::Application::WorkSpace
25 {
26     // ===== public member function =====
27     public:
28         // default constructor
29         WorkSpace();
30         // default destructor
31         virtual ~WorkSpace();
32     // =====
33     // ===== setter function
34     void SetFont( sf::Font* a_font );
35     // =====
36     // ===== pure virtual funciton
37     // =====
38     // initializes the component of a workspace
39     virtual void InitializeWorkSpace() = 0;
40     // all the logic for the workspace is here
41     virtual void Update() = 0;
42     // updates all the display elment
43     virtual void UpdateAllDispalyElement() = 0;
44     // draw to the screen
45     virtual void Draw( sf::RenderWindow& a_window ) = 0;
46     // debugger
47     virtual void Debugger() = 0;
48 }
```

```
73 // setter function for game data
74 virtual void SetGameDataManager( Brood::Application::Components::GameDataManager*
75 a_gameDataManagerPtr );
76
77 // ====== protected member function ======
78
79 // ===== helper funciton to dynamicallycreate UI elements =====
80
81 // ===== dynamically create a button, initializes it, and adds
82 // it to m_unNameList
83 // =====
84
85 // Dynamically create a button, initializes it, and adds
86 // it to m_unNameList
87 Brood::BroodUI::Button* DyCreateButton( sf::Vector2f a_size,
88                                         sf::Vector2f a_pos,
89                                         std::string a_text = "",
90                                         sf::Color a_color =
91                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
92     Brood::BroodUI::Button* DyCreateButton( float a_sizeX, float a_sizeY,
93                                         float a_posX, float a_posY,
94                                         std::string a_text = "",
95                                         sf::Color a_color =
96                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
97
98 // Dynamically create a textbox, initializes it, and adds
99 // it to m_unNameList
100 Brood::BroodUI::TextBox* DyCreateTextBox( sf::Vector2f a_size,
101                                         sf::Vector2f a_pos,
102                                         std::string a_text = "",
103                                         bool a_isEditable = false,
104                                         std::string a_placeholderText = "",
105                                         sf::Color a_color =
106                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
107     Brood::BroodUI::TextBox* DyCreateTextBox( float a_sizeX, float a_sizeY,
108                                         float a_posX, float a_posY,
109                                         std::string a_text = "",
110                                         bool a_isEditable = false,
111                                         std::string a_placeholderText = "",
112                                         sf::Color a_color =
113                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
114
115 // Dynamically create a Drop down Input, initializes it,
116 // and adds it to m_unNameList
117 Brood::BroodUI::DropDownInput* DyCreateDropDownInput( sf::Vector2f a_size,
118                                         sf::Vector2f a_pos,
119                                         sf::Color a_color =
120                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
121     Brood::BroodUI::DropDownInput* DyCreateDropDownInput( float a_sizeX, float a_sizeY,
122                                         float a_posX, float a_posY,
123                                         sf::Color a_color =
124                                         Brood::Application::StaticVariables::ST_ColorVariables::stm_AppPrimaryColor );
125
126 // Dynamically createss a dropdown input Panel Element.
127 // It has 2 UI element and its layout is
128 // <textBox prompt> <dropdown input>
129 void DyCreateDropdownInputElement( sf::RectangleShape* a_panelBodyPtr,
130                                         Brood::BroodUI::TextBox** a_txtPromptPtr,
131                                         Brood::BroodUI::DropDownInput** a_ddiPtr,
132                                         std::string a_promptToDisplay,
133                                         std::vector<std::string> a_itemsToAdd,
134                                         unsigned a_firstItemLength = 0,
135                                         bool a_createFromTop = false );
136
137 // Dynamically createss a Panel Element.
138 // It has 4 UI element and its layout is
139 // <textBox prompt> <button '<'> <textbox value> <button '>'>
140 void DyCreateDecIncPannelElement( sf::RectangleShape* a_panelBodyPtr,
141                                         Brood::BroodUI::TextBox** a_txtPromptPtr,
142                                         Brood::BroodUI::Button** a_btnDecPtr,
143                                         Brood::BroodUI::TextBox** a_txtValuePtr,
144                                         Brood::BroodUI::Button** a_btnIncPtr,
145                                         std::string a_promptToDisplay,
146                                         std::string a_valueToDisplay,
147                                         bool a_createFromTop = false );
148
149 // Dynamically createss a file input Panel Element.
150 // It has 3 UI element and its layout is
151 // <textBox prompt>
152 // <textbox>
153 // <open butotn>
154 void DyCreateFileInputPannelElement( sf::RectangleShape* a_panelBodyPtr,
155                                         Brood::BroodUI::TextBox** a_txtPromptPtr,
156                                         Brood::BroodUI::TextBox** a_txtFileInputPtr,
157                                         Brood::BroodUI::Button** a_btnOpenFilePtr,
158                                         std::string a_promptToDisplay,
159                                         std::string a_valueToDisplay,
```

```

154                                     bool a_createFromTop = false );
155
156 // Dynamically creates a Test Input Panel Element.
157 //     It has 2 UI element and its layout is
158 //     <textBox prompt> <textbox>
159 void DyCreateTextInputPannelElement( sf::RectangleShape* a_panelBodyPtr,
160                                     Brood::BroodUI::TextBox** a_txtPromptPtr,
161                                     Brood::BroodUI::TextBox** a_txtTextInputPtr,
162                                     std::string a_promptToDisplay,
163                                     std::string a_valueToDisplay,
164                                     std::string a_placeHolderText,
165                                     bool a_createFromTop = false );
166
167 // =====
168 // ===== generalized update functions =====
169 // =====
170
171 // TODO may be generalize for other editor too
172 // TODO maybe could be made like a initializer function
173 // TODO make this a signify if the the element was
174 //     increased, decreased, or none
175
176 // checks if the user interacted with the
177 // x size panel Element
178 // This is a generalized update SizeX
179 void UpdateDecIncSizeX( Brood::BroodUI::Button* a_btnDecSizeX,
180                         Brood::BroodUI::TextBox* a_txtSizeX,
181                         Brood::BroodUI::Button* a_btnIncSizeX,
182                         Brood::BroodUI::UIElement* a_elemnetToChangeSizeX,
183                         unsigned a_unit,
184                         unsigned a_lowerLimit,
185                         unsigned a_upperLimit );
186
187 // checks if the user interacted with the
188 // x size panel Element
189 // This is a generalized update SizeY
190 void UpdateDecIncSizeY( Brood::BroodUI::Button* a_btnDecSizeY,
191                         Brood::BroodUI::TextBox* a_txtSizeY,
192                         Brood::BroodUI::Button* a_btnIncSizeY,
193                         Brood::BroodUI::UIElement* a_elemnetToChangeSizeY,
194                         unsigned a_unit,
195                         unsigned a_lowerLimit,
196                         unsigned a_upperLimit );
197
198 // checks if the user interacted with the Card
199 // x Pos panel Element
200 void UpdateDecIncPosX( Brood::BroodUI::Button* a_btnDecPosX,
201                         Brood::BroodUI::TextBox* a_txtPosX,
202                         Brood::BroodUI::Button* a_btnIncPosX,
203                         Brood::BroodUI::UIElement* a_elemnetToChangePosX,
204                         unsigned a_unit,
205                         unsigned a_lowerLimit,
206                         unsigned a_upperLimit );
207
208 // checks if the user interacted with the Card
209 // x Pos panel Element
210 void UpdateDecIncPosY( Brood::BroodUI::Button* a_btnDecPosY,
211                         Brood::BroodUI::TextBox* a_txtPosY,
212                         Brood::BroodUI::Button* a_btnIncPosY,
213                         Brood::BroodUI::UIElement* a_elemnetToChangePosY,
214                         unsigned a_unit,
215                         unsigned a_lowerLimit,
216                         unsigned a_upperLimit );
217
218 // ===== protected data member =====
219 protected:
220
221     sf::Font* m_font;
222
223     static Brood::Application::Components::GameDataManager* m_gameData;
224
225     std::vector<Brood::BroodUI::UIElement*> m_unNamedUIList;
226
227     bool m_isDebugger;
228 };
229
230
231 // =====
232 // ===== end of WorkSpace class =====
233 // =====

```

Index

~Application
 Brood::Application::Application, 28

~Board
 Brood::Application::Components::Board, 33

~BoardEditor
 Brood::Application::BoardEditor, 46

~Button
 Brood::BroodUI::Button, 55

~CardEditor
 Brood::Application::CardEditor, 72

~CardInfo
 Brood::Application::Components::CardInfo, 129

~Deck
 Brood::Application::Components::Deck, 135

~DeckEditor
 Brood::Application::DeckEditor, 144

~DeckManager
 Brood::Application::Components::DeckManager, 155

~Dice
 Brood::Application::Components::Dice, 163

~DiceEditor
 Brood::Application::DiceEditor, 170

~DisplayCard
 Brood::Application::Components::DisplayCard, 182

~DropDownInput
 Brood::BroodUI::DropDownInput, 195

~DropDownMenu
 Brood::BroodUI::DropDownMenu, 200

~EditorWorkSpace
 Brood::Application::EditorWorkspace, 210

~FileAccess
 Brood::Application::FileAccess, 220

~GameDataManager
 Brood::Application::Components::GameDataManager, 223

~GameEditor
 Brood::Application::GameEditor, 231

~HomeWorkSpace
 Brood::Application::HomeWorkSpace, 239

~Id
 Brood::BroodUI::Id, 249

~MenuBar
 Brood::BroodUI::MenuBar, 260

~Path
 Brood::Application::Components::Path, 279

~Player
 Brood::Application::Components::Player, 292

~PlayerEditor
 Brood::Application::PlayerEditor, 301

~PlayerManager
 Brood::Application::Components::PlayerManager, 314

~SpriteHandler
 Brood::SpriteHandler, 323

~TextBox
 Brood::BroodUI::TextBox, 377

~TileEditor
 Brood::Application::TileEditor, 389

~Tiles
 Brood::Application::Components::Tiles, 402

~UIElement
 Brood::BroodUI::UIElement, 408

~WorkSpace
 Brood::Application::WorkSpace, 424

AddCardInfoToDeck
 Brood::Application::Components::Deck, 136

AddCardToDeckAtCurrlIdx
 Brood::Application::Components::DeckManager, 156

AddChild
 Brood::BroodUI::Id, 249

AddItemToMenu
 Brood::BroodUI::DropDownInput, 195, 196
 Brood::BroodUI::DropDownMenu, 201
 Brood::BroodUI::MenuBar, 261

AddMenuBar
 Brood::BroodUI::MenuBar, 261, 262

AddPlayerToList
 Brood::Application::Components::Path, 279

AddToMap
 Brood::BroodUI::MapIdToElement, 256

Application, 17
 Brood::Application::Application, 28

Board, 31
 Brood::Application::Components::Board, 33

BoardEditor, 43
 Brood::Application::BoardEditor, 46

Brood, 17, 21

Brood::Application, 21

Brood::Application::Application, 27
 ~Application, 28
 Application, 28
 CreateWorkSpace, 28
 Debugger, 28
 Draw, 29
 m_events, 30

m_mainWorkspace, 30
 m_window, 30
 myBoard, 30
 myDice, 30
 PollEvents, 29
 RunApplicaiton, 29
 Brood::Application::BoardEditor, 44
 ~BoardEditor, 46
 BoardEditor, 46
 Debugger, 46
 Draw, 46
 InitializeWorkSpace, 47
 m_btnBoardDecCol, 49
 m_btnBoardDecRow, 49
 m_btnBoardIncCol, 49
 m_btnBoardIncRow, 50
 m_btnBoardPosDecX, 50
 m_btnBoardPosDecY, 50
 m_btnBoardPosIncX, 50
 m_btnBoardPosIncY, 50
 m_btnBoardSizeDecX, 50
 m_btnBoardSizeDecY, 51
 m_btnBoardSizeIncX, 51
 m_btnBoardSizeIncY, 51
 m_panelBodyPtr, 51
 m_txtBoardCol, 51
 m_txtBoardColPromt, 51
 m_txtBoardPosPromtX, 52
 m_txtBoardPosPromtY, 52
 m_txtBoardPosX, 52
 m_txtBoardPosY, 52
 m_txtBoardRow, 52
 m_txtBoardRowPromt, 52
 m_txtBoardSizePromtX, 53
 m_txtBoardSizePromtY, 53
 m_txtBoardSizeX, 53
 m_txtBoardSizeY, 53
 m_txtSettingTitle, 53
 Update, 47
 UpdateAllDisplayElement, 47
 UpdateBoardColPanelElement, 47
 UpdateBoardRowPanelElement, 48
 UpdateBoardXPosPanelElement, 48
 UpdateBoardXSizePanelElement, 48
 UpdateBoardYPosPanelElement, 48
 UpdateBoardYSizePanelElement, 49
 Brood::Application::CardEditor, 60
 ~CardEditor, 72
 CardEditor, 72
 Debugger, 73
 Draw, 73
 DrawCardDisplayBcompPanel, 73
 DrawCardDisplayFcompPanel, 74
 DrawCardDisplayPanel, 74
 DrawCardInfoPanel, 74
 InitializeCardDisplayBcompPanel, 76
 InitializeCardDisplayFcompPanel, 76
 InitializeCardDisplayPanel, 76
 InitializeCardInfoPanel, 76
 InitializeSettingSelectionDDI, 76
 InitializeWorkSpace, 77
 m_btnCardAnswerPromptDecPosX, 82
 m_btnCardAnswerPromptDecPosY, 82
 m_btnCardAnswerPromptDecSizeX, 82
 m_btnCardAnswerPromptDecSizeY, 82
 m_btnCardAnswerPromptIncPosX, 82
 m_btnCardAnswerPromptIncPosY, 82
 m_btnCardAnswerPromptIncSizeX, 83
 m_btnCardAnswerPromptIncSizeY, 83
 m_btnCardAnswerValueDecPosX, 83
 m_btnCardAnswerValueDecPosY, 83
 m_btnCardAnswerValueDecSizeX, 83
 m_btnCardAnswerValueDecSizeY, 83
 m_btnCardAnswerValueIncPosX, 84
 m_btnCardAnswerValueIncPosY, 84
 m_btnCardAnswerValueIncSizeX, 84
 m_btnCardAnswerValueIncSizeY, 84
 m_btnCardBackBgOpenFile, 84
 m_btnCardBackTimePromptDecPosX, 84
 m_btnCardBackTimePromptDecPosY, 85
 m_btnCardBackTimePromptDecSizeX, 85
 m_btnCardBackTimePromptDecSizeY, 85
 m_btnCardBackTimePromptIncPosX, 85
 m_btnCardBackTimePromptIncPosY, 85
 m_btnCardBackTimePromptIncSizeX, 85
 m_btnCardBackTimePromptIncSizeY, 86
 m_btnCardBackTimeValueDecPosX, 86
 m_btnCardBackTimeValueDecPosY, 86
 m_btnCardBackTimeValueDecSizeX, 86
 m_btnCardBackTimeValueDecSizeY, 86
 m_btnCardBackTimeValueIncPosX, 86
 m_btnCardBackTimeValueIncPosY, 87
 m_btnCardBackTimeValueIncSizeX, 87
 m_btnCardBackTimeValueIncSizeY, 87
 m_btnCardDecPosX, 87
 m_btnCardDecPosY, 87
 m_btnCardDecSizeX, 87
 m_btnCardDecSizeY, 88
 m_btnCardDownPromptDecPosX, 88
 m_btnCardDownPromptDecPosY, 88
 m_btnCardDownPromptDecSizeX, 88
 m_btnCardDownPromptDecSizeY, 88
 m_btnCardDownPromptIncPosX, 88
 m_btnCardDownPromptIncPosY, 89
 m_btnCardDownPromptIncSizeX, 89
 m_btnCardDownPromptIncSizeY, 89
 m_btnCardDownValueDecPosX, 89
 m_btnCardDownValueDecPosY, 89
 m_btnCardDownValueDecSizeX, 89
 m_btnCardDownValueDecSizeY, 90
 m_btnCardDownValueIncPosX, 90
 m_btnCardDownValueIncPosY, 90
 m_btnCardDownValueIncSizeX, 90
 m_btnCardDownValueIncSizeY, 90
 m_btnCardFrontBgOpenFile, 90
 m_btnCardFrontTimePromptDecPosX, 91

m_btnCardFrontTimePromptDecPosY, 91
m_btnCardFrontTimePromptDecSizeX, 91
m_btnCardFrontTimePromptDecSizeY, 91
m_btnCardFrontTimePromptIncPosX, 91
m_btnCardFrontTimePromptIncPosY, 91
m_btnCardFrontTimePromptIncSizeX, 92
m_btnCardFrontTimePromptIncSizeY, 92
m_btnCardFrontTimeValueDecPosX, 92
m_btnCardFrontTimeValueDecPosY, 92
m_btnCardFrontTimeValueDecSizeX, 92
m_btnCardFrontTimeValueDecSizeY, 92
m_btnCardFrontTimeValueIncPosX, 93
m_btnCardFrontTimeValueIncPosY, 93
m_btnCardFrontTimeValueIncSizeX, 93
m_btnCardFrontTimeValueIncSizeY, 93
m_btnCardIncPosX, 93
m_btnCardIncPosY, 93
m_btnCardIncSizeX, 94
m_btnCardIncSizeY, 94
m_btnCardQuestionPromptDecPosX, 94
m_btnCardQuestionPromptDecPosY, 94
m_btnCardQuestionPromptDecSizeX, 94
m_btnCardQuestionPromptDecSizeY, 94
m_btnCardQuestionPromptIncPosX, 95
m_btnCardQuestionPromptIncPosY, 95
m_btnCardQuestionPromptIncSizeX, 95
m_btnCardQuestionPromptIncSizeY, 95
m_btnCardQuestionValueDecPosX, 95
m_btnCardQuestionValueDecPosY, 95
m_btnCardQuestionValueDecSizeX, 96
m_btnCardQuestionValueDecSizeY, 96
m_btnCardQuestionValueIncPosX, 96
m_btnCardQuestionValueIncPosY, 96
m_btnCardQuestionValueIncSizeX, 96
m_btnCardQuestionValueIncSizeY, 96
m_btnCardSubmitDecPosX, 97
m_btnCardSubmitDecPosY, 97
m_btnCardSubmitDecSizeX, 97
m_btnCardSubmitDecSizeY, 97
m_btnCardSubmitIncPosX, 97
m_btnCardSubmitIncPosY, 97
m_btnCardSubmitIncSizeX, 98
m_btnCardSubmitIncSizeY, 98
m_btnCardTurnCardDecPosX, 98
m_btnCardTurnCardDecPosY, 98
m_btnCardTurnCardDecSizeX, 98
m_btnCardTurnCardDecSizeY, 98
m_btnCardTurnCardIncPosX, 99
m_btnCardTurnCardIncPosY, 99
m_btnCardTurnCardIncSizeX, 99
m_btnCardTurnCardIncSizeY, 99
m_btnCardUpPromptDecPosX, 99
m_btnCardUpPromptDecPosY, 99
m_btnCardUpPromptDecSizeX, 100
m_btnCardUpPromptDecSizeY, 100
m_btnCardUpPromptIncPosX, 100
m_btnCardUpPromptIncPosY, 100
m_btnCardUpPromptIncSizeX, 100
m_btnCardUpPromptIncSizeY, 100
m_btnCardUpPromptIncSizeX, 100
m_btnCardUpPromptIncSizeY, 100
m_btnCardUpValueDecPosX, 101
m_btnCardUpValueDecPosY, 101
m_btnCardUpValueDecSizeX, 101
m_btnCardUpValueDecSizeY, 101
m_btnCardUpValueIncPosX, 101
m_btnCardUpValueIncPosY, 101
m_btnCardUpValueIncSizeX, 102
m_btnCardUpValueIncSizeY, 102
m_btnDecCurrCardDown, 102
m_btnDecCurrCardNum, 102
m_btnDecCurrCardTime, 102
m_btnDecCurrCardUp, 102
m_btnDecDeckNum, 103
m_btnIncCurrCardDown, 103
m_btnIncCurrCardNum, 103
m_btnIncCurrCardTime, 103
m_btnIncCurrCardUp, 103
m_btnIncDeckNum, 103
m_btnTurnCardAround, 104
m_ddiSettingSelection, 104
m_panelBodyPtr, 104
m_selectedSettingIdx, 104
m_txtCardAnswerPromptPosX, 104
m_txtCardAnswerPromptPosXPrompt, 104
m_txtCardAnswerPromptPosY, 104
m_txtCardAnswerPromptPosYPrompt, 105
m_txtCardAnswerPromptSizeX, 105
m_txtCardAnswerPromptSizeXPrompt, 105
m_txtCardAnswerPromptSizeY, 105
m_txtCardAnswerPromptSizeYPrompt, 105
m_txtCardAnswerValuePosX, 105
m_txtCardAnswerValuePosXPrompt, 106
m_txtCardAnswerValuePosY, 106
m_txtCardAnswerValuePosYPrompt, 106
m_txtCardAnswerValueSizeX, 106
m_txtCardAnswerValueSizeXPrompt, 106
m_txtCardAnswerValueSizeY, 106
m_txtCardAnswerValueSizeYPrompt, 107
m_txtCardBackBgFileInput, 107
m_txtCardBackBgFileNamePrompt, 107
m_txtCardBackTimePromptPosX, 107
m_txtCardBackTimePromptPosXPrompt, 107
m_txtCardBackTimePromptPosY, 107
m_txtCardBackTimePromptPosYPrompt, 108
m_txtCardBackTimePromptSizeX, 108
m_txtCardBackTimePromptSizeXPrompt, 108
m_txtCardBackTimePromptSizeY, 108
m_txtCardBackTimePromptSizeYPrompt, 108
m_txtCardBackTimeValuePosX, 108
m_txtCardBackTimeValuePosXPrompt, 109
m_txtCardBackTimeValuePosY, 109
m_txtCardBackTimeValuePosYPrompt, 109
m_txtCardBackTimeValueSizeX, 109
m_txtCardBackTimeValueSizeXPrompt, 109
m_txtCardBackTimeValueSizeY, 109
m_txtCardBackTimeValueSizeYPrompt, 110
m_txtCardDownPromptPosX, 110

m_txtCardDownPromptPosXPrompt, 110
m_txtCardDownPromptPosY, 110
m_txtCardDownPromptPosYPrompt, 110
m_txtCardDownPromptSizeX, 110
m_txtCardDownPromptSizeXPrompt, 111
m_txtCardDownPromptSizeY, 111
m_txtCardDownPromptSizeYPrompt, 111
m_txtCardDownValuePosX, 111
m_txtCardDownValuePosXPrompt, 111
m_txtCardDownValuePosY, 111
m_txtCardDownValuePosYPrompt, 112
m_txtCardDownValueSizeX, 112
m_txtCardDownValueSizeXPrompt, 112
m_txtCardDownValueSizeY, 112
m_txtCardDownValueSizeYPrompt, 112
m_txtCardFrontBgFileInput, 112
m_txtCardFrontBgFileNamePrompt, 113
m_txtCardFrontTimePromptPosX, 113
m_txtCardFrontTimePromptPosXPrompt, 113
m_txtCardFrontTimePromptPosY, 113
m_txtCardFrontTimePromptPosYPrompt, 113
m_txtCardFrontTimePromptSizeX, 113
m_txtCardFrontTimePromptSizeXPrompt, 114
m_txtCardFrontTimePromptSizeY, 114
m_txtCardFrontTimePromptSizeYPrompt, 114
m_txtCardFrontTimeValuePosX, 114
m_txtCardFrontTimeValuePosXPrompt, 114
m_txtCardFrontTimeValuePosY, 114
m_txtCardFrontTimeValuePosYPrompt, 115
m_txtCardFrontTimeValueSizeX, 115
m_txtCardFrontTimeValueSizeXPrompt, 115
m_txtCardFrontTimeValueSizeY, 115
m_txtCardFrontTimeValueSizeYPrompt, 115
m_txtCardPosX, 115
m_txtCardPosXPrompt, 116
m_txtCardPosY, 116
m_txtCardPosYPrompt, 116
m_txtCardQuestionPromptPosX, 116
m_txtCardQuestionPromptPosXPrompt, 116
m_txtCardQuestionPromptPosY, 116
m_txtCardQuestionPromptPosYPrompt, 117
m_txtCardQuestionPromptSizeX, 117
m_txtCardQuestionPromptSizeXPrompt, 117
m_txtCardQuestionPromptSizeY, 117
m_txtCardQuestionPromptSizeYPrompt, 117
m_txtCardQuestionValuePosX, 117
m_txtCardQuestionValuePosXPrompt, 118
m_txtCardQuestionValuePosY, 118
m_txtCardQuestionValuePosYPrompt, 118
m_txtCardQuestionValueSizeX, 118
m_txtCardQuestionValueSizeXPrompt, 118
m_txtCardQuestionValueSizeY, 118
m_txtCardQuestionValueSizeYPrompt, 119
m_txtCardSizeX, 119
m_txtCardSizeXPrompt, 119
m_txtCardSizeY, 119
m_txtCardSizeYPrompt, 119
m_txtCardSubmitPosX, 119
m_txtCardSubmitPosXPrompt, 120
m_txtCardSubmitPosY, 120
m_txtCardSubmitPosYPrompt, 120
m_txtCardSubmitSizeX, 120
m_txtCardSubmitSizeXPrompt, 120
m_txtCardSubmitSizeY, 120
m_txtCardSubmitSizeYPrompt, 121
m_txtCardTurnCardPosX, 121
m_txtCardTurnCardPosXPrompt, 121
m_txtCardTurnCardPosY, 121
m_txtCardTurnCardPosYPrompt, 121
m_txtCardTurnCardSizeX, 121
m_txtCardTurnCardSizeXPrompt, 122
m_txtCardTurnCardSizeY, 122
m_txtCardTurnCardSizeYPrompt, 122
m_txtCardUpPromptPosX, 122
m_txtCardUpPromptPosXPrompt, 122
m_txtCardUpPromptPosY, 122
m_txtCardUpPromptPosYPrompt, 123
m_txtCardUpPromptSizeX, 123
m_txtCardUpPromptSizeXPrompt, 123
m_txtCardUpPromptSizeY, 123
m_txtCardUpPromptSizeYPrompt, 123
m_txtCardUpValuePosX, 123
m_txtCardUpValuePosXPrompt, 124
m_txtCardUpValuePosY, 124
m_txtCardUpValuePosYPrompt, 124
m_txtCardUpValueSizeX, 124
m_txtCardUpValueSizeXPrompt, 124
m_txtCardUpValueSizeY, 124
m_txtCardUpValueSizeYPrompt, 125
m_txtCurrCardAnswerInput, 125
m_txtCurrCardAnswerPrompt, 125
m_txtCurrCardDown, 125
m_txtCurrCardDownPrompt, 125
m_txtCurrCardNum, 125
m_txtCurrCardNumPrompt, 126
m_txtCurrCardQuestionInput, 126
m_txtCurrCardQuestionPrompt, 126
m_txtCurrCardTime, 126
m_txtCurrCardTimePrompt, 126
m_txtCurrCardUp, 126
m_txtCurrCardUpPrompt, 127
m_txtDeckNum, 127
m_txtDeckNumPrompt, 127
Update, 77
UpdateAllDispalyElement, 77
UpdateCardBackFileTexture, 77
UpdateCardDisplayBcompPanel, 78
UpdateCardDisplayFcompPanel, 78
UpdateCardDisplayPanel, 78
UpdateCardFrontFileTexture, 79
UpdateCardInfoPanel, 79
UpdateCurrCardAnswerInput, 79
UpdateCurrCardInfoDown, 79
UpdateCurrCardInfoTime, 80
UpdateCurrCardInfoUp, 80
UpdateCurrCardQuestionInput, 80

UpdateCurrSelectedCardIdx, 80
UpdateCurrSelectedDeckIdx, 81
UpdateSettingSelectionDDI, 81
UpdateTurnCard, 81
Brood::Application::Components, 22
 ENUM_MovementType, 22
 ENUM_TileType, 23
 MOVEMENT_cardOnly, 23
 MOVEMENT_diceOnly, 23
 MOVEMENT_diceThenCard, 23
 TILE_blank, 23
 TILE_bridge, 23
 TILE_end, 23
 TILE_start, 23
 TILE_tile, 23
Brood::Application::Components::Board, 31
 ~Board, 33
 Board, 33
 Debugger, 34
 DecreaseNumCol, 34
 DecreaseNumRow, 34
 Draw, 34
 GetBoardPathList, 36
 GetBoardPos, 36
 GetBoardSize, 36
 GetCurrentActivePath, 36
 GetDataToSave, 37
 GetNumCol, 37
 GetNumRow, 37
 IncreaseNumCol, 37
 IncreaseNumRow, 38
 InitializeBoard, 38
 LoadDataFromFile, 39
 m_boardBody, 42
 m_boardPaths, 42
 m_currActivePathPtr, 42
 m_numCols, 43
 m_numRows, 43
 SaveDataToFile, 39
 SetBoardPos, 39, 40
 SetBoardSize, 40
 SetCurrentActiveTilePtr, 41
 SetNumCol, 41
 SetNumRow, 41
 ToggleDrawLine, 41
 UpdateBoardPath, 42
Brood::Application::Components::CardInfo, 128
 ~CardInfo, 129
 CardInfo, 129
 GetCorrectAnswer, 129
 GetCorrectNumSteps, 129
 GetDataToSave, 130
 GetIncorrectNumSteps, 130
 GetQuestion, 130
 GetTime, 130
 InitializeCard, 131
 m_correctAnswer, 133
 m_correctNumSteps, 133
 m_incorrectNumSteps, 133
 m_question, 133
 m_time, 133
 operator=, 131
 SetCorrectAnswer, 131
 SetCorrectNumSteps, 132
 SetIncorrectNumSteps, 132
 SetQuestion, 132
 SetTime, 132
Brood::Application::Components::Deck, 134
 ~Deck, 135
 AddCardInfoToDeck, 136
 DealCard, 136
 Deck, 135
 GetCardInitFileName, 136
 GetCardInitFilePath, 136
 GetCardList, 137
 GetCardPtrAtIdx, 137
 GetCurrActiveCardIdx, 137
 GetCurrActiveCardPtr, 138
 GetDataToSave, 138
 InitializeDeck, 138
 LoadCardFromInitFile, 138
 LoadDataFromFile, 139
 m_cardInitFileName, 141
 m_cardInitFilePath, 141
 m_cardList, 141
 m_currActiveCardIdx, 141
 m_currUndealtCardIdx, 141
 operator=, 139
 ResetDeck, 139
 SaveCardToInitFile, 139
 SaveDataToFile, 140
 SetCardListSize, 140
 SetCurrActiveCardIdx, 140
 Shuffel, 140
Brood::Application::Components::DeckManager, 154
 ~DeckManager, 155
 AddCardToDeckAtCurrIdx, 156
 Debugger, 156
 DeckManager, 155, 156
 Draw, 156
 GetCurrActiveDeck, 157
 GetCurrActiveDeckIdx, 157
 GetDataToSave, 157
 GetDeckAtIdx, 157
 GetDeckList, 158
 GetIncorrectPenalty, 158
 GetMovementType, 158
 InitializeDeckManager, 158
 LoadDataFromFile, 159
 m_currDeckIdx, 161
 m_deckList, 161
 m_incorrectPenalty, 161
 m_movementType, 161
 operator=, 159
 SaveDataToFile, 159
 SetCurrActiveDeckIdx, 160

SetDeckSize, 160
 SetIncorrectPenalty, 160
 SetMovementType, 160
 Brood::Application::Components::Dice, 161
 ~Dice, 163
 Dice, 162, 163
 GetDataToSave, 164
 GetNumSides, 164
 InitializeDice, 164
 LoadDataFromFile, 164
 m_numSides, 167
 operator=, 165
 RollDice, 165
 SaveDataToFile, 165
 SetBodySize, 166
 SetNumSides, 166
 SetTexture, 167
 Brood::Application::Components::DisplayCard, 179
 ~DisplayCard, 182
 Debugger, 182
 DisplayCard, 182
 Draw, 182
 GetAnswerPromptPtr, 183
 GetAnswerValuePtr, 183
 GetBackBgFileName, 183
 GetBackTimePromptPtr, 183
 GetBackTimeValuePtr, 184
 GetCompUpperPos, 184
 GetDataToSave, 184
 GetDownPromptPtr, 185
 GetDownValuePtr, 185
 GetFrontBgFileName, 185
 GetFrontTimePromptPtr, 185
 GetFrontTimeValuePtr, 185
 GetFurtherstBottomRightCoordinate, 186
 GetNearestTopLeftCoordinate, 186
 GetQuestionPromptPtr, 186
 GetQuestionValuePtr, 186
 GetSubmitPtr, 187
 GetTurnCardPtr, 187
 GetUpPromptPtr, 187
 GetUpValuePtr, 187
 InializeDisplayCard, 188
 InitializeDisplayCard, 188
 LoadDataFromFile, 188
 m_backBgFileName, 191
 m_BtnSubmit, 191
 m_BtnTurnCard, 192
 m_cardInfoToDisplay, 192
 m_frontBgFileName, 192
 m_isCurrFront, 192
 m_TxtBackTimePromt, 192
 m_TxtBackTimeValue, 192
 m_TxtDownPrompt, 192
 m_TxtDownValue, 193
 m_TxtFrontTimePromt, 193
 m_TxtFrontTimeValue, 193
 m_TxtQuestionPrompt, 193
 m_TxtQuestionValue, 193
 m_TxtUpPromt, 193
 m_TxtUpValue, 193
 m_TxtUserAnswerPrompt, 194
 m_TxtUserAnswerValue, 194
 operator=, 188
 SaveDataToFile, 189
 SetBackBgFileName, 189
 SetCardFront, 189
 SetCardInfoToDisplay, 190
 SetFrontBgFileName, 190
 ToggleCardFace, 190
 UpdateDisplayedText, 190
 UpdateLargestUpperPos, 191
 UpdateSmallestLowerPos, 191
 Brood::Application::Components::GameDataManager, 222
 ~GameDataManager, 223
 Debugger, 224
 Draw, 224
 GameDataManager, 223
 GetBoardPtr, 224
 GetDataToSave, 224
 GetDeckManagerPtr, 224
 GetDicePtr, 225
 GetDisplayCardPtr, 225
 GetGameTitle, 225
 GetPlayerManagerPtr, 225
 InitializeGameDataManager, 226
 InitializeGameDataManger, 226
 LoadDataFromFile, 226
 m_board, 227
 m_deckManager, 227
 m_dice, 227
 m_displayCard, 228
 m_gameTitle, 228
 m_playerManager, 228
 SaveDataToFile, 226
 SetGameTitle, 227
 Brood::Application::Components::Path, 277
 ~Path, 279
 AddPlayerToList, 279
 DeletePlayerFromList, 280
 Draw, 280
 DrawPath, 280
 GetBridgeEndPathPtr, 281
 GetDataToSave, 281
 GetDeckIdx, 281
 GetDeckPtr, 281
 GetDrawLine, 281
 GetForceDiceRoll, 282
 GetNextPathPtr, 282
 GetNumCardDraw, 282
 GetPlayerListPtr, 282
 GetPrevioiuPathPtr, 283
 GetTileCenter, 283
 GetTilePtr, 283
 GetTileType, 284

InitializePath, 284
m_bridgeEndPathPtr, 288
m_bridgePathLine, 288
m_deckIdx, 288
m_deckPtr, 288
m_drawLine, 288
m_forceDiceRoll, 288
m_nextPathLine, 289
m_nextPathPtr, 289
m_numCardDraw, 289
m_playerPtrList, 289
m_previousPathPtr, 289
m_tilePtr, 289
m_tileType, 290
operator=, 284
Path, 279
SetBridgeEndPathPtr, 285
SetDeckIdx, 285
SetDeckPtr, 285
SetForceDiceRoll, 285
SetNextPathPtr, 286
SetNumCardDraw, 286
SetPreviousPathPtr, 286
SetTilePtr, 287
SetTileType, 287
ToggleDrawLine, 287
UpdatePathLines, 287
Brood::Application::Components::Player, 290
 ~Player, 292
 Draw, 293
 GetCurrPathPtr, 293
 GetDataToSave, 293
 GetPlayerBody, 293
 GetPlayerSpriteBody, 293
 GetPositionOffsetX, 294
 GetPositionOffsetY, 294
 InitializePlayer, 294
 LoadDataFromFile, 295
 m_playerBody, 297
 m_playerCurrPathPtr, 297
 m_positionOffsetX, 297
 m_positionOffsetY, 298
 m_spriteBody, 298
 operator=, 295
 Player, 292
 SaveDataToFile, 295
 SetPlayerSizeX, 295
 SetPlayerSizeY, 296
 SetPositionOffsetX, 296
 SetPositionOffsetY, 296
 UpdatePathptr, 297
 UpdatePosition, 297
Brood::Application::Components::PlayerManager, 312
 ~PlayerManager, 314
 Draw, 314
 GetAllPlayerBegin, 315
 GetAllPlayerEnd, 315
 GetCurrActivePlayerIdx, 315
 GetDataToSave, 315
 GetMaxPlayer, 316
 GetMinPlayer, 316
 GetNextPlayer, 316
 GetPlayerAtCurrlIdx, 316
 GetPlayerStartPath, 317
 InitializePlayerManager, 317
 LoadDataFromFile, 318
 m_allPlayers, 320
 m_currActivePlayerIdx, 320
 m_maxPlayer, 320
 m_minPlayer, 321
 m_startPathPtr, 321
 operator=, 318
 PlayerManager, 314
 ReplacePlayerAt, 318
 SaveDataToFile, 319
 SetCurrActivePlayerIdx, 319
 SetMaxPlayer, 319
 SetMinPlayer, 319
 SetPathForPlayerAtCurrlIdx, 320
 SetPlayerStartPath, 320
Brood::Application::Components::Tiles, 401
 ~Tiles, 402
 GetCol, 402
 GetRow, 402
 GetRowAndCol, 403
 m_tileColNum, 405
 m_tileRowNum, 405
 operator=, 403
 SetCol, 403
 SetRow, 403
 SetRowAndCol, 404
 Tiles, 402
 UpdateTile, 404
Brood::Application::Data, 23
Brood::Application::Data::ST_BoardData, 330
 GetString, 330
 PopulateFromString, 331
 stm_boardPosX, 331
 stm_boardPosY, 331
 stm_boardSizeX, 331
 stm_boardSizeY, 331
 stm_currActiveNumCol, 332
 stm_currActiveNumRow, 332
 stm_numCol, 332
 stm_numRow, 332
Brood::Application::Data::ST_CardInfoPrefabData, 333
 GetString, 334
 PopulateFromString, 334
 stm_correctAnswer, 334
 stm_correctNumSteps, 334
 stm_incorrectNumSteps, 334
 stm_question, 335
 stm_time, 335
Brood::Application::Data::ST_DeckManagerData, 338
 GetString, 339
 PopulateFromString, 339

stm_currDecksIdx, 339
 stm_incorrectPenalty, 339
 stm_movementType, 339
 stm_numDecks, 340
Brood::Application::Data::ST_DeckPrefabData, 340
 GetString, 341
 PopulateFromString, 341
 stm_cardInitFilename, 342
 stm_currActiveCardIdx, 342
 stm_deckPosX, 342
 stm_deckPosY, 342
 stm_deckSizeX, 342
 stm_deckSizeY, 342
 stm_numTotalCard, 343
 stm_textureFileName, 343
 stm_undealtCardIdx, 343
Brood::Application::Data::ST_DicePrefabData, 344
 GetString, 344
 PopulateFromString, 344
 stm_dicePosX, 345
 stm_dicePosY, 345
 stm_diceSizeX, 345
 stm_diceSizeY, 345
 stm_numSides, 345
 stm_textureFileName, 345
Brood::Application::Data::ST_DisplayCardData, 346
 GetString, 349
 PopulateFromString, 349
 stm_answerPromptPositionX, 350
 stm_answerPromptPositionY, 350
 stm_answerPromptSizeX, 350
 stm_answerPromptSizeY, 350
 stm_answerValuePositionX, 350
 stm_answerValuePositionY, 350
 stm_answerValueSizeX, 351
 stm_answerValueSizeY, 351
 stm_backTimePromptPositionX, 351
 stm_backTimePromptPositionY, 351
 stm_backTimePromptSizeX, 351
 stm_backTimePromptSizeY, 351
 stm_backTimeValuePositionX, 352
 stm_backTimeValuePositionY, 352
 stm_backTimeValueSizeX, 352
 stm_backTimeValueSizeY, 352
 stm_displayCardBackTextureFilename, 352
 stm_displayCardFrontTextureFilename, 352
 stm_displayCardPositionX, 353
 stm_displayCardPositionY, 353
 stm_displayCardSizeX, 353
 stm_displayCardSizeY, 353
 stm_downPromptPositionX, 353
 stm_downPromptPositionY, 353
 stm_downPromptSizeX, 354
 stm_downPromptSizeY, 354
 stm_downValuePositionX, 354
 stm_downValuePositionY, 354
 stm_downValueSizeX, 354
 stm_downValueSizeY, 354
 stm_frontTimePromptPositionX, 355
 stm_frontTimePromptPositionY, 355
 stm_frontTimePromptSizeX, 355
 stm_frontTimePromptSizeY, 355
 stm_frontTimeValuePositionX, 355
 stm_frontTimeValuePositionY, 355
 stm_frontTimeValueSizeX, 356
 stm_frontTimeValueSizeY, 356
 stm_questionPromptPositionX, 356
 stm_questionPromptPositionY, 356
 stm_questionPromptSizeX, 356
 stm_questionPromptSizeY, 356
 stm_questionValuePositionX, 357
 stm_questionValuePositionY, 357
 stm_questionValueSizeX, 357
 stm_questionValueSizeY, 357
 stm_submitButtonValuePositionX, 357
 stm_submitButtonValuePositionY, 357
 stm_submitButtonValueSizeX, 358
 stm_submitButtonValueSizeY, 358
 stm_turnCardPromptPositionX, 358
 stm_turnCardPromptPositionY, 358
 stm_turnCardPromptSizeX, 358
 stm_turnCardPromptSizeY, 358
 stm_upPromptPositionX, 359
 stm_upPromptPositionY, 359
 stm_upPromptSizeX, 359
 stm_upPromptSizeY, 359
 stm_upValuePositionX, 359
 stm_upValuePositionY, 359
 stm_upValueSizeX, 360
 stm_upValueSizeY, 360
Brood::Application::Data::ST_GameData, 363
 GetString, 364
 PopulateFromString, 364
 stm_gameTitle, 364
Brood::Application::Data::ST_PathPrefabData, 366
 GetString, 367
 PopulateFromString, 367
 stm_assignedDeckId, 368
 stm_ColNum, 368
 stm_endBridgeTileColNum, 368
 stm_endBridgeTileRowNum, 368
 stm_forceDiceRoll, 368
 stm_nextTileColNum, 368
 stm_nextTileRowNum, 368
 stm_numberCardDraw, 369
 stm_RowNum, 369
 stm_textureFileName, 369
 stm_tileType, 369
Brood::Application::Data::ST_PlayerManagerData, 369
 GetString, 370
 PopulateFromString, 370
 stm_currPlayerIdx, 370
 stm_maxPlayer, 371
 stm_minPlayer, 371
Brood::Application::Data::ST_PlayerPrefabData, 371
 GetString, 372

PopulateFromString, 372
stm_currCol, 373
stm_currRow, 373
stm_playerOffsetX, 373
stm_playerOffsetY, 373
stm_playerSizeX, 373
stm_playerSizeY, 373
stm_textureFileName, 374
Brood::Application::DeckEditor, 142
 ~DeckEditor, 144
 Debugger, 144
 DeckEditor, 144
 Draw, 145
 InitializeWorkSpace, 145
 m_btnCardInitOpenFile, 149
 m_btnDecCardNum, 149
 m_btnDeckBgOpenFile, 149
 m_btnDeckDecCurrIdx, 149
 m_btnDeckDecPosX, 149
 m_btnDeckDecPosY, 149
 m_btnDeckDecSizeX, 150
 m_btnDeckDecSizeY, 150
 m_btnDeckIncCurrIdx, 150
 m_btnDeckIncPosX, 150
 m_btnDeckIncPosY, 150
 m_btnDeckIncSizeX, 150
 m_btnDeckIncSizeY, 151
 m_btnIncCardNum, 151
 m_panelBodyPtr, 151
 m_txtCardInitFileInput, 151
 m_txtCardInitFileNamePrompt, 151
 m_txtCardNum, 151
 m_txtCardNumPrompt, 152
 m_txtDeckBgFileInput, 152
 m_txtDeckBgFileNamePrompt, 152
 m_txtDeckCurrIdx, 152
 m_txtDeckCurrIdxPrompt, 152
 m_txtDeckPosX, 152
 m_txtDeckPosXPrompt, 153
 m_txtDeckPosY, 153
 m_txtDeckPosYPrompt, 153
 m_txtDeckSizeX, 153
 m_txtDeckSizeXPrompt, 153
 m_txtDeckSizeY, 153
 m_txtDeckSizeYPrompt, 154
 m_txtSettingTitle, 154
 Update, 145
 UpdateAllDisplayElement, 146
 UpdateCardInitFile, 146
 UpdateCardNumber, 146
 UpdateCurrSelectedDeckIdx, 147
 UpdateDeckFileTexture, 147
 UpdateDeckPosX, 147
 UpdateDeckPosY, 148
 UpdateDeckSizeX, 148
 UpdateDeckSizeY, 148
Brood::Application::DiceEditor, 168
 ~DiceEditor, 170
Debugger, 170
DiceEditor, 170
Draw, 170
InitializeWorkSpace, 171
m_boardPtr, 174
m_btnDiceDecPosX, 174
m_btnDiceDecPosY, 174
m_btnDiceDecSide, 174
m_btnDiceDecSizeX, 175
m_btnDiceDecSizeY, 175
m_btnDiceIncPosX, 175
m_btnDiceIncPosY, 175
m_btnDiceIncSide, 175
m_btnDiceIncSizeX, 175
m_btnDiceIncSizeY, 176
m_btnDiceOpenFile, 176
m_btnRollDice, 176
m_dicePtr, 176
m_panelBodyPtr, 176
m_txtDiceFileInput, 176
m_txtDiceFileNamePrompt, 177
m_txtDicePosX, 177
m_txtDicePosXPrompt, 177
m_txtDicePosY, 177
m_txtDicePosYPrompt, 177
m_txtDiceSide, 177
m_txtDiceSidePrompt, 178
m_txtDiceSizeX, 178
m_txtDiceSizeXPrompt, 178
m_txtDiceSizeY, 178
m_txtDiceSizeYPrompt, 178
m_txtSettingTitle, 178
Update, 171
UpdateAllDisplayElement, 171
UpdateDiceFileTexture, 172
UpdateDicePosX, 172
UpdateDicePosY, 172
UpdateDiceSideNum, 173
UpdateDiceSizeX, 173
UpdateDiceSizeY, 173
UpdateRollDice, 173
Brood::Application::EditorWorkspace, 209
 ~EditorWorkspace, 210
 Debugger, 210
 Draw, 210
 EditorWorkspace, 210
 InitializeEditModeTabs, 211
 InitializeWorkSpace, 211
 m_activeEditorIdx, 213
 m_editModesTabs, 213
 m_editorWorkspaceList, 213
 m_sidePanel, 213
 SetGameDataManager, 211
 Update, 212
 UpdateActiveEditorIdx, 212
 UpdateAllDisplayElement, 212
 UpdateEditModeTabs, 212
Brood::Application:: FileAccess, 219

~FileAccess, 220
 CheckEOF, 220
 CreateFile, 220
 FileAccess, 220
 GetNextLine, 220
 m_fileHandler, 222
 m.filePath, 222
 OpenFile, 221
 RemoveAllContent, 221
 Rewind, 221
 WriteOneLineToFile, 221
Brood::Application::GameEditor, 229
 ~GameEditor, 231
 Debugger, 231
 Draw, 231
 GameEditor, 230
 InitializeWorkSpace, 231
 m_btnDeckDecNum, 234
 m_btnDeckIncNum, 234
 m_btnPlayerDecMaxnNum, 234
 m_btnPlayerDecMinNum, 234
 m_btnPlayerIncMaxNum, 234
 m_btnPlayerIncMinNum, 235
 m_ddiIncorectPenalty, 235
 m_ddiMovementType, 235
 m_panelBodyPtr, 235
 m_titleScreenBtn, 235
 m_txtDeckNum, 235
 m_txtDeckNumPrompt, 236
 m_txtGameInput, 236
 m_txtGameNamePrompt, 236
 m_txtncorectPenaltyPromt, 236
 m_txtnovementTypePromt, 236
 m_txtPlayerMaxNum, 236
 m_txtPlayerMaxNumPrompt, 237
 m_txtPlayerMinNum, 237
 m_txtPlayerMinNumPrompt, 237
 m_txtSettingTitle, 237
 Update, 232
 UpdateAllDisplayElement, 232
 UpdateDeckNumber, 232
 UpdateGameTitleInput, 232
 UpdateIncorrectPenaltyPanelElement, 233
 UpdateMaximumPlayerNumber, 233
 UpdateMinimumPlayerNumber, 233
 UpdateMovementTypePanelElement, 233
Brood::Application::HomeWorkSpace, 237
 ~HomeWorkSpace, 239
 Debugger, 240
 Draw, 240
 ExecuteCreateNewEditDialogBox, 240
 ExecuteLoadPreviousEditDialogBox, 240
 ExecuteMenuItem, 240
 HomeWorkSpace, 239
 InitializeRibbonTabs, 241
 InitializeWorkSpace, 241
 LoadGameData, 241
 m_cancleBtn, 243
 m_createNewEditBtn, 243
 m_currGameInfoIdx, 243
 m_currWorkSpaceIdx, 244
 m_enterFileNameFeildTxt, 244
 m_fileNameTxt, 244
 m_gameInfoFilePathList, 244
 m_gameListTexture, 244
 m_gameOrEditorBtn, 244
 m_gameOrEditorForeground, 245
 m_importGameBtn, 245
 m_isHomeWorkspace, 245
 m_loadPreviousEditsBtn, 245
 m_openFileBtn, 245
 m_openFileBtnIdx, 245
 m_ribbonTabs, 245
 m_showEnterFilenameDialogBox, 246
 m_showImportDialogBox, 246
 m_toggleGameInfoLeftBtn, 246
 m_toggleGameInfoRightBtn, 246
 m_workSpacesList, 246
 ResetGameData, 241
 SaveGameData, 242
 SwitchToHomeWorkspace, 242
 Update, 242
 UpdateAllDisplayElement, 242
 UpdateFileNameFeildDialog, 242
 UpdateHomeWorkspace, 243
 UpdateRibbonTabs, 243
Brood::Application::PlayerEditor, 299
 ~PlayerEditor, 301
 Debugger, 302
 Draw, 302
 InitializeWorkSpace, 302
 m_boardPtr, 305
 m_btnPlayerDecCurrIdx, 306
 m_btnPlayerDecCurrIdxOffsetX, 306
 m_btnPlayerDecCurrIdxOffsetY, 306
 m_btnPlayerDecCurrIdxSizeX, 306
 m_btnPlayerDecCurrIdxSizeY, 306
 m_btnPlayerDecStartColNum, 306
 m_btnPlayerDecStartRowNum, 307
 m_btnPlayerIncCurrIdx, 307
 m_btnPlayerIncCurrIdxOffsetX, 307
 m_btnPlayerIncCurrIdxOffsetY, 307
 m_btnPlayerIncCurrIdxSizeX, 307
 m_btnPlayerIncCurrIdxSizeY, 307
 m_btnPlayerIncStartColNum, 308
 m_btnPlayerIncStartRowNum, 308
 m_btnPlayerOpenFile, 308
 m_ddiDrawPlayerAtCurrIdx, 308
 m_panelBodyPtr, 308
 m_playerManagerPtr, 308
 m_txtdrawPlayerAtCurrIdxPromt, 309
 m_txtpPlayerCurrIdx, 309
 m_txtpPlayerCurrIdxOffsetX, 309
 m_txtpPlayerCurrIdxOffsetXPrompt, 309
 m_txtpPlayerCurrIdxOffsetY, 309
 m_txtpPlayerCurrIdxOffsetYPrompt, 309

m_txtPlayerCurrlIdxPrompt, 310
m_txtPlayerCurrlIdxSizeX, 310
m_txtPlayerCurrlIdxSizeXPrompt, 310
m_txtPlayerCurrlIdxSizeY, 310
m_txtPlayerCurrlIdxSizeYPrompt, 310
m_txtPlayerFileInput, 310
m_txtpPlayerFileNamePrompt, 311
m_txtPlayerStartColNum, 311
m_txtPlayerStartColPrompt, 311
m_txtPlayerStartRowNum, 311
m_txtPlayerStartRowPrompt, 311
m_txtSettingTitle, 311
PlayerEditor, 301
Update, 302
UpdateAllDisplayElement, 303
UpdateCurrPlayerFileTexture, 303
UpdateCurrPlayerOffsetX, 303
UpdateCurrPlayerOffsetY, 304
UpdateCurrPlayerSizeX, 304
UpdateCurrPlayerSizeY, 304
UpdateCurrSelectedPlayerIdx, 304
UpdateStartColNumber, 305
UpdateStartRowNumber, 305
Brood::Application::StaticVariables, 23
Brood::Application::StaticVariables::ST_ColorVariables,
 335
 GetRandomColor, 336
 stm_AppPrimaryColor, 336
 stm_AppSecondaryColor, 336
 stm_Black, 337
 stm_CurrActiveOverlay, 337
 stm_ErrorColor, 337
 stm_HotOverlay, 337
 stm_MainMenu, 337
 stm_White, 337
Brood::Application::StaticVariables::ST_Folders, 360
 stm_assets, 361
 stm_cardTextures, 361
 stm_cwd, 361
 stm_data, 362
 stm_deckTextures, 362
 stm_diceTextures, 362
 stm_fonts, 362
 stm_playerTextures, 362
 stm_textures, 363
 stm_tileTextures, 363
Brood::Application::StaticVariables::ST_GlobalCoreVariables,
 364
 stm_exit, 365
 stm_font, 365
 stm_is_debug_mode, 365
 stm_panelPercentage, 365
 stm_seed, 365
 stm_window_height, 366
 stm_window_width, 366
Brood::Application::TileEditor, 386
 ~TileEditor, 389
 Debugger, 389
Draw, 389
InitializeWorkSpace, 390
m_btnTileBgOpenFile, 393
m_btnTileDecAssignedDeckIdx, 393
m_btnTileDecCurrCol, 393
m_btnTileDecCurrRow, 394
m_btnTileDecEndBridgeCol, 394
m_btnTileDecEndBridgeRow, 394
m_btnTileDecNextPathCol, 394
m_btnTileDecNextPathRow, 394
m_btnTileDecNumCorrectCard, 394
m_btnTileIncAssignedDeckIdx, 395
m_btnTileIncCurrCol, 395
m_btnTileIncCurrRow, 395
m_btnTileIncEndBridgeCol, 395
m_btnTileIncEndBridgeRow, 395
m_btnTileIncNextPathCol, 395
m_btnTileIncNextPathRow, 396
m_btnTileIncNumCorrectCard, 396
m_ddiTileForceDiceRoll, 396
m_ddiTileType, 396
m_panelBodyPtr, 396
m_txtSettingTitle, 396
m_txtTileAssignedDeckIdx, 397
m_txtTileAssignedDeckIdxPromt, 397
m_txtTileBgFileInput, 397
m_txtTileBgFileNamePrompt, 397
m_txtTileCurrCol, 397
m_txtTileCurrColPromt, 397
m_txtTileCurrRow, 398
m_txtTileCurrRowPromt, 398
m_txtTileEndBridgeCol, 398
m_txtTileEndBridgeColPromt, 398
m_txtTileEndBridgeRow, 398
m_txtTileEndBridgeRowPromt, 398
m_txtTileForceDiceRollPromt, 399
m_txtTileNextPathCol, 399
m_txtTileNextPathColPromt, 399
m_txtTileNextPathRow, 399
m_txtTileNextPathRowPromt, 399
m_txtTileNumCorrectCard, 399
m_txtTileNumCorrectCardPromt, 400
m_txtTileTypePromt, 400
TileEditor, 388
Update, 390
UpdateAllDisplayElement, 390
UpdateAssinedDeckIdx, 390
UpdateCorrectCardNumber, 390
UpdateEndBridgeColNumber, 391
UpdateEndBridgeRowNumber, 391
UpdateForceDiceRoll, 391
UpdateNextPathColNumber, 391
UpdateNextPathRowNumber, 392
UpdateTileColNumber, 392
UpdateTileFileTexture, 392
UpdateTileRowNumber, 392
UpdateTileType, 393
Brood::Application::WorkSpace, 422

~WorkSpace, 424
 Debugger, 425
 Draw, 425
 DyCreateButton, 425, 426
 DyCreateDeclIncPannelElement, 426
 DyCreateDropDownInput, 427, 428
 DyCreateDropdownInputPannelElement, 428
 DyCreateFileInputPannelElement, 429
 DyCreateTextBox, 430, 431
 DyCreateTextInputPannelElement, 431
 InitializeWorkSpace, 432
 m_font, 435
 mGameData, 435
 m_isDebugger, 435
 m_unNamedUIList, 436
 SetFont, 432
 SetGameDataManager, 432
 Update, 432
 UpdateAllDisplayElement, 433
 UpdateDecIncPosX, 433
 UpdateDecIncPosY, 434
 UpdateDecIncSizeX, 434
 UpdateDecIncSizeY, 435
 WorkSpace, 424
Brood::BroodUI, 24
 ENUM_UIType, 24
 UI_button, 24
 UI_dropDownInput, 24
 UI_dropDownMenu, 24
 UI_menuBar, 24
 UI_scrollBar, 24
 UI_textBox, 24
Brood::BroodUI::Button, 54
 ~Button, 55
 Button, 55
 Debugger, 56
 DoElement, 56
 DyCreateButton, 56, 57
 GetSpriteBody, 57
 m_bodySprite, 58
 operator=, 58
 SetSelected, 58
Brood::BroodUI::DropDownInput, 194
 ~DropDownInput, 195
 AddItemToMenu, 195, 196
 DropDownInput, 195
 operator=, 196
Brood::BroodUI::DropDownMenu, 198
 ~DropDownMenu, 200
 AddItemToMenu, 201
 Debugger, 201
 DoElement, 202
 Draw, 202
 DropDownMenu, 200
 GetItemList, 202
 m_items, 207
 m_maxItemLength, 207
 operator=, 202
 SetActiveOverlayColor, 203
 SetBodyColor, 203
 SetBodyPosition, 203, 204
 SetBodySize, 204, 205
 SetEachItemPos, 205
 SetEachItemSize, 205
 SetFont, 205
 SetFontSize, 206
 SetHotOverlayColor, 206
 SetItemPos, 206
 SetItemSize, 207
 SetText, 207
Brood::BroodUI::ElementSelection, 214
 Debugger, 214
 DebugPrintSelectedElementID, 215
 GetAlmostActiveElementIdPtr, 215
 GetCurrActiveElementIdPtr, 215
 GetHotElementIdFlag, 215
 GetHotElementIdPtr, 215
 GetLastActiveElementIdPtr, 216
 m_almostActiveElementIdPtr, 217
 m_currActiveElementIdPtr, 218
 m_hotElementFlag, 218
 m_hotElementIdPtr, 218
 m_lastActiveElementIdPtr, 218
 SetAlmostActiveElementIdPtr, 216
 SetCurrActiveElementIdPtr, 216
 SetHotElementIdPtr, 217
 SetHotElementIdPtrFlag, 217
 SetLastActiveElementIdPtr, 217
Brood::BroodUI::Id, 247
 ~Id, 249
 AddChild, 249
 DeleteChildIdAtIdx, 249
 GetChildIdAtIdx, 251
 GetChildIdx, 251
 GetElementID, 251
 GetParentID, 252
 GetParentIDPtr, 252
 GetTotalChildNum, 252
 GLOBAL_ID_NUM, 254
 HasChild, 253
 HasParent, 253
 Id, 248, 249
 m_allChildPtrs, 254
 m_elementID, 254
 m_hasChilds, 254
 m_parentID, 255
 m_parentIDPtr, 255
 operator=, 253
 SetParent, 254
Brood::BroodUI::MapIdToElement, 255
 AddToMap, 256
 GetElementPtrFromMap, 256
 GetMap, 257
 RemoveFromMap, 257
 stm_mapper, 257

Brood::BroodUI::MenuBar, 258
~MenuBar, 260
AddItemToMenu, 261
AddMenuToMenuBar, 261, 262
Debugger, 263
Draw, 263
GetMenuList, 263
m_menus, 268
MenuBar, 260
operator=, 263
SetActiveOverlayColor, 264
SetBodyColor, 264
SetBodyPosition, 264, 265
SetBodySize, 265, 266
SetFont, 266
SetFontColor, 267
SetFontSize, 267
SetHotOverlayColor, 267
SetMenuBodySize, 268
SetMenuPos, 268
Brood::BroodUI::TextBox, 375
~TextBox, 377
Debugger, 377
DeleteLastChar, 377
DoElement, 377
Draw, 378
DyCreateTextBox, 378, 379
InputLogic, 379
IsEditable, 380
IsEnterPressed, 380
IsSelected, 380
m_enterPressed, 383
m_hasLimit, 383
m_isEditable, 383
m_limit, 383
m_ossText, 383
m_placeHolderText, 384
operator=, 380
SetEditable, 381
SetEnterPressedFalse, 381
SetLimit, 381
SetPlaceHolderText, 382
SetSelected, 382
SetText, 382
TextBox, 376, 377
TypeOn, 382
Brood::BroodUI::UIElement, 406
~UIElement, 408
Debugger, 409
DoElement, 409
Draw, 409
GetActiveOverlayColor, 411
GetBody, 411
GetBodyColor, 411
GetBodyPosition, 411
GetBodySize, 411
GetElementIdPtr, 412
GetElementType, 412
GetFontSize, 412
GetHotOverlayColor, 412
GetText, 413
IsAlmostActiveElement, 413
IsCurrActiveElement, 413
IsHotElement, 413
IsMouseOverElement, 414
IsSelected, 414
m_activeOverlayColor, 419
m_body, 419
m_bodyOverLay, 419
m_debugTextSave, 419
m_drawOverlay, 420
m_drawText, 420
m_elementId, 420
m_elementType, 420
m_font, 420
m_fontSize, 420
m_hotOverlayColor, 421
m_isSelected, 421
m_text, 421
m_textContent, 421
operator=, 414
SetActiveOverlayColor, 415
SetBodyColor, 415
SetBodyPosition, 415, 416
SetBodySize, 416, 417
SetDrawOverlay, 417
SetFont, 417
SetFontColor, 417
SetFontSize, 417
SetHotOverlayColor, 418
SetSelected, 418
SetText, 418
SetTextPosition, 419
UIElement, 408, 409
Brood::MouseHandler, 270
ChangeCursorTo, 271
CURSOR_arrow, 271
CURSOR_hand, 271
CURSOR_help, 271
CURSOR_notAllowed, 271
CURSOR_sizeHorizontal, 271
CURSOR_text, 271
CURSOR_wait, 271
Debugger, 272
ENUM_MouseCursorType, 271
GetCurrFrameLMSStatus, 272
GetCurrFrameMousePos, 272
GetCurrFrameRMSStatus, 272
GetlastFrameLMSStatus, 273
GetLastFrameMousePos, 273
GetlastFrameRMSStatus, 273
InitializeCursor, 273
IsLeftButtonHold, 274
IsLeftButtonPressed, 274
IsLeftButtonReleased, 274
IsRightButtonHold, 274

IsRightButtonPressed, 274
 IsRightButtonReleased, 275
 m_currFrameLeftMouseButtonPressed, 276
 m_currFrameMousePos, 276
 m_currFrameRightMouseButtonPressed, 276
 m_cursor, 276
 m_lastFrameLeftMouseButtonPressed, 276
 m_lastFrameMousePos, 276
 m_lastFrameRightMouseButtonPressed, 276
 UpdateMouseButtonStatus, 275
 UpdateMousePos, 275
 Brood::SpriteHandler, 321
 ~SpriteHandler, 323
 Debugger, 323
 GetcurrSpriteIndex, 324
 GetSpritHeight, 324
 GetSpritLenght, 324
 GetTextureDirectoryPath, 324
 GetTextureFileName, 324
 m_body, 327
 m_currSpriteIndex, 327
 m_spriteHeight, 328
 m_spriteLength, 328
 m_texture, 328
 m_textureDirectoryPath, 328
 m_textureFileName, 328
 operator=, 325
 RemoveTexture, 325
 SetSpriteFromTexture, 325
 SetSpriteHeight, 326
 SetSpriteLength, 326
 SetTextureFromFilePath, 326, 327
 SetTextureFromSavedFilePath, 327
 SpriteHandler, 323
 Brood::UtilityFuncs, 25
 LoadTextureFromFile, 25
 BrodUI, 19
 Button, 59
 Brood::BrodUI::Button, 55

 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/437
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/437, 438
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/439
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/439, 440
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/442
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/442, 443
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/445
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/445, 446
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/447

 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/447, 448
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/458
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/458, 459
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/460
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/460, 461
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/467
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/467, 468
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/469
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/470
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/472
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/473, 474
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/475
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/475, 476
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/477
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/478
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/480
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/481
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/484
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/484, 485
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/485
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/486
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/488
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/488, 489
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/489
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/490
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/490, 491
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/492
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/492, 493
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/493
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/493, 494
 C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BroodMaker/494, 495

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
534

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
534, 535

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
536

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
536, 537

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
538

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
539, 540

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
542

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
542, 543

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
543

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
543, 544

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
546, 507

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardEditor.h
Brood::Application::CardEditor, 72

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.cpp,
508

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h,
509

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::MouseHandler, 271

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
511

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CreateFile

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CreateWorkSpace

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_arrow

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_hand

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_help

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_notAllowed

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_sizeHorizontal

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_text

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
CURSOR_wait

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::MouseHandler, 271

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
525

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Data, 18

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
DealCard

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::Components::Deck, 136

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Debugger

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::Application, 28

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::BoardEditor, 46

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::CardEditor, 73

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::Components::Board, 34

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::Components::DeckManager,
156

C:/data/OneDrive - RCNJ/data/Programming/vs/BroodMaker/BoardHandler.h
Brood::Application::Components::DisplayCard, 182

Brood::Application::Components::GameDataManager, 224
 Brood::Application::DeckEditor, 144
 Brood::Application::DiceEditor, 170
 Brood::Application::EditorWorkspace, 210
 Brood::Application::GameEditor, 231
 Brood::Application::HomeWorkSpace, 240
 Brood::Application::PlayerEditor, 302
 Brood::Application::TileEditor, 389
 Brood::Application::WorkSpace, 425
 Brood::BroodUI::Button, 56
 Brood::BroodUI::DropDownMenu, 201
 Brood::BroodUI::ElementSelection, 214
 Brood::BroodUI::MenuBar, 263
 Brood::BroodUI::TextBox, 377
 Brood::BroodUI::UIElement, 409
 Brood::MouseHandler, 272
 Brood::SpriteHandler, 323
 DebugPrintSelectedElementID
 Brood::BroodUI::ElementSelection, 215
 Deck
 Brood::Application::Components::Deck, 135
 DeckEditor
 Brood::Application::DeckEditor, 144
 DeckManager
 Brood::Application::Components::DeckManager, 155, 156
 DecreaseNumCol
 Brood::Application::Components::Board, 34
 DecreaseNumRow
 Brood::Application::Components::Board, 34
 DELETE_KEY
 TextBox.h, 528
 DeleteChildIdAtIdx
 Brood::BroodUI::Id, 249
 DeleteLastChar
 Brood::BroodUI::TextBox, 377
 DeletePlayerFromList
 Brood::Application::Components::Path, 280
 Dice, 167
 Brood::Application::Components::Dice, 162, 163
 DiceEditor, 179
 Brood::Application::DiceEditor, 170
 DisplayCard
 Brood::Application::Components::DisplayCard, 182
 DoElement
 Brood::BroodUI::Button, 56
 Brood::BroodUI::DropDownMenu, 202
 Brood::BroodUI::TextBox, 377
 Brood::BroodUI::UIElement, 409
 Draw
 Brood::Application::Application, 29
 Brood::Application::BoardEditor, 46
 Brood::Application::CardEditor, 73
 Brood::Application::Components::Board, 34
 Brood::Application::Components::DeckManager, 156
 Brood::Application::Components::DisplayCard, 182
 Brood::Application::Components::GameDataManager, 224
 Brood::Application::Components::Path, 280
 Brood::Application::Components::Player, 293
 Brood::Application::Components::PlayerManager, 314
 Brood::Application::DeckEditor, 145
 Brood::Application::DiceEditor, 170
 Brood::Application::EditorWorkspace, 210
 Brood::Application::GameEditor, 231
 Brood::Application::HomeWorkSpace, 240
 Brood::Application::PlayerEditor, 302
 Brood::Application::TileEditor, 389
 Brood::Application::WorkSpace, 425
 Brood::BroodUI::DropDownMenu, 202
 Brood::BroodUI::MenuBar, 263
 Brood::BroodUI::TextBox, 378
 Brood::BroodUI::UIElement, 409
 DrawCardDisplayBcompPanel
 Brood::Application::CardEditor, 73
 DrawCardDisplayFcompPanel
 Brood::Application::CardEditor, 74
 DrawCardDisplayPanel
 Brood::Application::CardEditor, 74
 DrawCardInfoPanel
 Brood::Application::CardEditor, 74
 DrawPath
 Brood::Application::Components::Path, 280
 DropDownInput, 197
 Brood::BroodUI::DropDownInput, 195
 DropDownMenu, 208
 Brood::BroodUI::DropDownMenu, 200
 DyCreateButton
 Brood::Application::WorkSpace, 425, 426
 Brood::BroodUI::Button, 56, 57
 DyCreateDeclIncPannelElement
 Brood::Application::WorkSpace, 426
 DyCreateDropDownInput
 Brood::Application::WorkSpace, 427, 428
 DyCreateDropdownInputPannelElement
 Brood::Application::WorkSpace, 428
 DyCreateFileInputPannelElement
 Brood::Application::WorkSpace, 429
 DyCreateTextBox
 Brood::Application::WorkSpace, 430, 431
 Brood::BroodUI::TextBox, 378, 379
 DyCreateTextInputPannelElement
 Brood::Application::WorkSpace, 431
 EditorWorkspace
 Brood::Application::EditorWorkspace, 210
 ElementSelection, 218
 ENTER_KEY
 TextBox.h, 528
 ENUM_MouseCursorType
 Brood::MouseHandler, 271
 ENUM_MovementType
 Brood::Application::Components, 22
 ENUM_TileType

Brood::Application::Components, 23
ENUM_UIType
 Brood::BroodUI, 24
ESCAPE_KEY
 TextBox.h, 528
ExecuteCreateNewEditDialogBox
 Brood::Application::HomeWorkSpace, 240
ExecuteLoadPreviousEditDialogBox
 Brood::Application::HomeWorkSpace, 240
ExecuteMenuItem
 Brood::Application::HomeWorkSpace, 240

FileAccess
 Brood::Application::FileAccess, 220

GameDataManager, 228
 Brood::Application::Components::GameDataManager, 223
GameEditor
 Brood::Application::GameEditor, 230
GetActiveOverlayColor
 Brood::BroodUI::UIElement, 411
GetAllPlayerBegin
 Brood::Application::Components::PlayerManager, 315
GetAllPlayerEnd
 Brood::Application::Components::PlayerManager, 315
GetAlmostActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 215
GetAnswerPromptPtr
 Brood::Application::Components::DisplayCard, 183
GetAnswerValuePtr
 Brood::Application::Components::DisplayCard, 183
GetBackBgFileName
 Brood::Application::Components::DisplayCard, 183
GetBackTimePromptPtr
 Brood::Application::Components::DisplayCard, 183
GetBackTimeValuePtr
 Brood::Application::Components::DisplayCard, 184
GetBoardPathList
 Brood::Application::Components::Board, 36
GetBoardPos
 Brood::Application::Components::Board, 36
GetBoardPtr
 Brood::Application::Components::GameDataManager, 224
GetBoardSize
 Brood::Application::Components::Board, 36
GetBody
 Brood::BroodUI::UIElement, 411
GetBodyColor
 Brood::BroodUI::UIElement, 411
GetBodyPosition
 Brood::BroodUI::UIElement, 411
GetBodySize
 Brood::BroodUI::UIElement, 411
GetBridgeEndPathPtr
 Brood::Application::Components::Path, 281

 GetCardInitFileName
 Brood::Application::Components::Deck, 136
 GetCardInitFilePath
 Brood::Application::Components::Deck, 136
 GetCardList
 Brood::Application::Components::Deck, 137
 GetCardPtrAtIdx
 Brood::Application::Components::Deck, 137
 GetChildIdAtIdx
 Brood::BroodUI::Id, 251
 GetChildIdx
 Brood::BroodUI::Id, 251
 GetCol
 Brood::Application::Components::Tiles, 402
 GetCompUpperPos
 Brood::Application::Components::DisplayCard, 184
 GetCorrectAnswer
 Brood::Application::Components::CardInfo, 129
 GetCorrectNumSteps
 Brood::Application::Components::CardInfo, 129
 GetCurrActiveCardIdx
 Brood::Application::Components::Deck, 137
 GetCurrActiveCardPtr
 Brood::Application::Components::Deck, 138
 GetCurrActiveDeck
 Brood::Application::Components::DeckManager, 157
 GetCurrActiveDeckIdx
 Brood::Application::Components::DeckManager, 157
 GetCurrActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 215
 GetCurrActivePlayerIdx
 Brood::Application::Components::PlayerManager, 315
 GetCurrentActivePath
 Brood::Application::Components::Board, 36
 GetCurrFrameLMSStatus
 Brood::MouseHandler, 272
 GetCurrFrameMousePos
 Brood::MouseHandler, 272
 GetCurrFrameRMSStatus
 Brood::MouseHandler, 272
 GetCurrPathPtr
 Brood::Application::Components::Player, 293
 GetcurrSpriteIndex
 Brood::SpriteHandler, 324
 GetDataToSave
 Brood::Application::Components::Board, 37
 Brood::Application::Components::CardInfo, 130
 Brood::Application::Components::Deck, 138
 Brood::Application::Components::DeckManager, 157
 Brood::Application::Components::Dice, 164
 Brood::Application::Components::DisplayCard, 184
 Brood::Application::Components::GameDataManager, 224
 Brood::Application::Components::Path, 281

Brood::Application::Components::Player, 293
 Brood::Application::Components::PlayerManager, 315
 GetDeckAtIdx
 Brood::Application::Components::DeckManager, 157
 GetDeckIdx
 Brood::Application::Components::Path, 281
 GetDeckList
 Brood::Application::Components::DeckManager, 158
 GetDeckManagerPtr
 Brood::Application::Components::GameDataManager, 224
 GetDeckPtr
 Brood::Application::Components::Path, 281
 GetDicePtr
 Brood::Application::Components::GameDataManager, 225
 GetDisplayCardPtr
 Brood::Application::Components::GameDataManager, 225
 GetDownPromptPtr
 Brood::Application::Components::DisplayCard, 185
 GetDownValuePtr
 Brood::Application::Components::DisplayCard, 185
 GetDrawLine
 Brood::Application::Components::Path, 281
 GetElementID
 Brood::BroodUI::Id, 251
 GetElementIdPtr
 Brood::BroodUI::UIElement, 412
 GetElementPtrFromMap
 Brood::BroodUI::MapIdToElement, 256
 GetElementType
 Brood::BroodUI::UIElement, 412
 GetFontSize
 Brood::BroodUI::UIElement, 412
 GetForceDiceRoll
 Brood::Application::Components::Path, 282
 GetFrontBgFileName
 Brood::Application::Components::DisplayCard, 185
 GetFrontTimePromptPtr
 Brood::Application::Components::DisplayCard, 185
 GetFrontTimeValuePtr
 Brood::Application::Components::DisplayCard, 185
 GetFurtherstBottomRightCordinate
 Brood::Application::Components::DisplayCard, 186
 GetGameTitle
 Brood::Application::Components::GameDataManager, 225
 GetHotElementIdFlag
 Brood::BroodUI::ElementSelection, 215
 GetHotElementIdPtr
 Brood::BroodUI::ElementSelection, 215
 GetHotOverlayColor
 Brood::BroodUI::UIElement, 412
 GetIncorrectNumSteps
 Brood::Application::Components::CardInfo, 130
 GetIncorrectPenalty
 Brood::Application::Components::DeckManager, 158
 GetItemList
 Brood::BroodUI::DropDownMenu, 202
 GetLastActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 216
 GetlastFrameLMSStatus
 Brood::MouseHandler, 273
 GetLastFrameMousePos
 Brood::MouseHandler, 273
 GetlastFrameRMSStatus
 Brood::MouseHandler, 273
 GetMap
 Brood::BroodUI::MapIdToElement, 257
 GetMaxPlayer
 Brood::Application::Components::PlayerManager, 316
 GetMenuList
 Brood::BroodUI::MenuBar, 263
 GetMinPlayer
 Brood::Application::Components::PlayerManager, 316
 GetMovementType
 Brood::Application::Components::DeckManager, 158
 GetNearestTopLeftCordinate
 Brood::Application::Components::DisplayCard, 186
 GetNextLine
 Brood::Application::FileAccess, 220
 GetNextPathPtr
 Brood::Application::Components::Path, 282
 GetNextPlayer
 Brood::Application::Components::PlayerManager, 316
 GetNumCardDraw
 Brood::Application::Components::Path, 282
 GetNumCol
 Brood::Application::Components::Board, 37
 GetNumRow
 Brood::Application::Components::Board, 37
 GetNumSides
 Brood::Application::Components::Dice, 164
 GetParentID
 Brood::BroodUI::Id, 252
 GetParentIDPtr
 Brood::BroodUI::Id, 252
 GetPlayerAtCurrlIdx
 Brood::Application::Components::PlayerManager, 316
 GetPlayerBody
 Brood::Application::Components::Player, 293
 GetPlayerListPtr
 Brood::Application::Components::Path, 282
 GetPlayerManagerPtr
 Brood::Application::Components::GameDataManager, 225

GetPlayerSpriteBody
 Brood::Application::Components::Player, 293

GetPlayerStartPath
 Brood::Application::Components::PlayerManager,
 317

GetPositionOffsetX
 Brood::Application::Components::Player, 294

GetPositionOffsetY
 Brood::Application::Components::Player, 294

GetPreviousPathPtr
 Brood::Application::Components::Path, 283

GetQuestion
 Brood::Application::Components::CardInfo, 130

GetQuestionPromptPtr
 Brood::Application::Components::DisplayCard, 186

GetQuestionValuePtr
 Brood::Application::Components::DisplayCard, 186

GetRandomColor
 Brood::Application::StaticVariables::ST_ColorVariables,
 336

GetRow
 Brood::Application::Components::Tiles, 402

GetRowAndCol
 Brood::Application::Components::Tiles, 403

GetSpriteBody
 Brood::BroodUI::Button, 57

GetSpriteHeight
 Brood::SpriteHandler, 324

GetSpriteLength
 Brood::SpriteHandler, 324

GetString
 Brood::Application::Data::ST_BoardData, 330
 Brood::Application::Data::ST_CardInfoPrefabData,
 334
 Brood::Application::Data::ST_DeckManagerData,
 339
 Brood::Application::Data::ST_DeckPrefabData,
 341
 Brood::Application::Data::ST_DicePrefabData, 344
 Brood::Application::Data::ST_DisplayCardData,
 349
 Brood::Application::Data::ST_GameData, 364
 Brood::Application::Data::ST_PathPrefabData, 367
 Brood::Application::Data::ST_PlayerManagerData,
 370
 Brood::Application::Data::ST_PlayerPrefabData,
 372

GetSubmitPtr
 Brood::Application::Components::DisplayCard, 187

GetText
 Brood::BroodUI::UIElement, 413

GetTextureDirectoryPath
 Brood::SpriteHandler, 324

GetTextureFileName
 Brood::SpriteHandler, 324

GetTileCenter
 Brood::Application::Components::Path, 283

GetTilePtr
 Brood::Application::Components::Path, 283

 Brood::Application::Components::Path, 283

GetTileType
 Brood::Application::Components::Path, 284

GetTime
 Brood::Application::Components::CardInfo, 130

GetTotalChildNum
 Brood::BroodUI::Id, 252

GetTurnCardPtr
 Brood::Application::Components::DisplayCard, 187

GetUpPromptPtr
 Brood::Application::Components::DisplayCard, 187

GetUpValuePtr
 Brood::Application::Components::DisplayCard, 187

GLOBAL_ID_NUM
 Brood::BroodUI::Id, 254

HasChild
 Brood::BroodUI::Id, 253

HasParent
 Brood::BroodUI::Id, 253

HomeWorkSpace, 247
 Brood::Application::HomeWorkSpace, 239

Id, 255
 Brood::BroodUI::Id, 248, 249

IncreaseNumCol
 Brood::Application::Components::Board, 37

IncreaseNumRow
 Brood::Application::Components::Board, 38

InitializeCardDisplayBcompPanel
 Brood::Application::CardEditor, 76

InitializeCardDisplayFcompPanel
 Brood::Application::CardEditor, 76

InitializeCardDisplayPanel
 Brood::Application::CardEditor, 76

InitializeCardInfoPanel
 Brood::Application::CardEditor, 76

InitializeDisplayCard
 Brood::Application::Components::DisplayCard, 188

InitializeSettingSelectionDDI
 Brood::Application::CardEditor, 76

InitializeBoard
 Brood::Application::Components::Board, 38

InitializeCard
 Brood::Application::Components::CardInfo, 131

InitializeCursor
 Brood::MouseHandler, 273

InitializeDeck
 Brood::Application::Components::Deck, 138

InitializeDeckManager
 Brood::Application::Components::DeckManager,
 158

InitializeDice
 Brood::Application::Components::Dice, 164

InitializeDisplayCard
 Brood::Application::Components::DisplayCard, 188

InitializeEditModeTabs
 Brood::Application::EditorWorkspace, 211

InitializeGameDataManager

Brood::Application::Components::GameDataManager, 226
InitializeGameDataManger
 Brood::Application::Components::GameDataManager, 226
InitializePath
 Brood::Application::Components::Path, 284
InitializePlayer
 Brood::Application::Components::Player, 294
InitializePlayerManager
 Brood::Application::Components::PlayerManager, 317
InitializeRibbonTabs
 Brood::Application::HomeWorkSpace, 241
InitializeWorkSpace
 Brood::Application::BoardEditor, 47
 Brood::Application::CardEditor, 77
 Brood::Application::DeckEditor, 145
 Brood::Application::DiceEditor, 171
 Brood::Application::EditorWorkspace, 211
 Brood::Application::GameEditor, 231
 Brood::Application::HomeWorkSpace, 241
 Brood::Application::PlayerEditor, 302
 Brood::Application::TileEditor, 390
 Brood::Application::WorkSpace, 432
InputLogic
 Brood::BroodUI::TextBox, 379
IsAlmostActiveElement
 Brood::BroodUI::UIElement, 413
IsCurrActiveElement
 Brood::BroodUI::UIElement, 413
IsEditable
 Brood::BroodUI::TextBox, 380
IsEnterPressed
 Brood::BroodUI::TextBox, 380
IsHotElement
 Brood::BroodUI::UIElement, 413
IsLeftButtonHold
 Brood::MouseHandler, 274
IsLeftButtonPressed
 Brood::MouseHandler, 274
IsLeftButtonReleased
 Brood::MouseHandler, 274
IsMouseOverElement
 Brood::BroodUI::UIElement, 414
IsRightButtonHold
 Brood::MouseHandler, 274
IsRightButtonPressed
 Brood::MouseHandler, 274
IsRightButtonReleased
 Brood::MouseHandler, 275
IsSelected
 Brood::BroodUI::TextBox, 380
 Brood::BroodUI::UIElement, 414
LoadCardFromInitFile
 Brood::Application::Components::Deck, 138
LoadDataFromFile
 Brood::Application::Components::Board, 39
 Brood::Application::Components::Deck, 139
 Brood::Application::Components::DeckManager, 159
 Brood::Application::Components::Dice, 164
 Brood::Application::Components::DisplayCard, 188
 Brood::Application::Components::GameDataManager, 226
 Brood::Application::Components::Player, 295
 Brood::Application::Components::PlayerManager, 318
LoadGameData
 Brood::Application::HomeWorkSpace, 241
LoadTextureFromFile
 Brood::UtilityFuncs, 25
m_activeEditorIdx
 Brood::Application::EditorWorkspace, 213
m_activeOverlayColor
 Brood::BroodUI::UIElement, 419
m_allChildPtrs
 Brood::BroodUI::Id, 254
m_allPlayers
 Brood::Application::Components::PlayerManager, 320
m_almostActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 217
m_backBgFileName
 Brood::Application::Components::DisplayCard, 191
m_board
 Brood::Application::Components::GameDataManager, 227
m_boardBody
 Brood::Application::Components::Board, 42
m_boardPaths
 Brood::Application::Components::Board, 42
m_boardPtr
 Brood::Application::DiceEditor, 174
 Brood::Application::PlayerEditor, 305
m_body
 Brood::BroodUI::UIElement, 419
 Brood::SpriteHandler, 327
m_bodyOverLay
 Brood::BroodUI::UIElement, 419
m_bodySprite
 Brood::BroodUI::Button, 58
m_bridgeEndPathPtr
 Brood::Application::Components::Path, 288
m_bridgePathLine
 Brood::Application::Components::Path, 288
m_btnBoardDecCol
 Brood::Application::BoardEditor, 49
m_btnBoardDecRow
 Brood::Application::BoardEditor, 49
m_btnBoardIncCol
 Brood::Application::BoardEditor, 49
m_btnBoardIncRow
 Brood::Application::BoardEditor, 50
m_btnBoardPosDecX
 Brood::Application::BoardEditor, 50

m_btnBoardPosDecY
 Brood::Application::BoardEditor, 50

m_btnBoardPosIncX
 Brood::Application::BoardEditor, 50

m_btnBoardPosIncY
 Brood::Application::BoardEditor, 50

m_btnBoardSizeDecX
 Brood::Application::BoardEditor, 50

m_btnBoardSizeDecY
 Brood::Application::BoardEditor, 51

m_btnBoardSizeIncX
 Brood::Application::BoardEditor, 51

m_btnBoardSizeIncY
 Brood::Application::BoardEditor, 51

m_btnCardAnswerPromptDecPosX
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptDecPosY
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptDecSizeX
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptDecSizeY
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptIncPosX
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptIncPosY
 Brood::Application::CardEditor, 82

m_btnCardAnswerPromptIncSizeX
 Brood::Application::CardEditor, 83

m_btnCardAnswerPromptIncSizeY
 Brood::Application::CardEditor, 83

m_btnCardAnswerValueDecPosX
 Brood::Application::CardEditor, 83

m_btnCardAnswerValueDecPosY
 Brood::Application::CardEditor, 83

m_btnCardAnswerValueDecSizeX
 Brood::Application::CardEditor, 83

m_btnCardAnswerValueDecSizeY
 Brood::Application::CardEditor, 83

m_btnCardAnswerValueIncPosX
 Brood::Application::CardEditor, 84

m_btnCardAnswerValueIncPosY
 Brood::Application::CardEditor, 84

m_btnCardAnswerValueIncSizeX
 Brood::Application::CardEditor, 84

m_btnCardAnswerValueIncSizeY
 Brood::Application::CardEditor, 84

m_btnCardBackBgOpenFile
 Brood::Application::CardEditor, 84

m_btnCardBackTimePromptDecPosX
 Brood::Application::CardEditor, 84

m_btnCardBackTimePromptDecPosY
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptDecSizeX
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptDecSizeY
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncPosX
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncPosY
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncSizeX
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncSizeY
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncPosY
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncSizeX
 Brood::Application::CardEditor, 85

m_btnCardBackTimePromptIncSizeY
 Brood::Application::CardEditor, 85

m_btnCardBackTimeValueDecPosX
 Brood::Application::CardEditor, 86

m_btnCardBackTimeValueDecPosY
 Brood::Application::CardEditor, 86

m_btnCardBackTimeValueDecSizeX
 Brood::Application::CardEditor, 86

m_btnCardBackTimeValueDecSizeY
 Brood::Application::CardEditor, 86

m_btnCardBackTimeValueIncPosX
 Brood::Application::CardEditor, 86

m_btnCardBackTimeValueIncPosY
 Brood::Application::CardEditor, 87

m_btnCardBackTimeValueIncSizeX
 Brood::Application::CardEditor, 87

m_btnCardBackTimeValueIncSizeY
 Brood::Application::CardEditor, 87

m_btnCardDecPosX
 Brood::Application::CardEditor, 87

m_btnCardDecPosY
 Brood::Application::CardEditor, 87

m_btnCardDecSizeX
 Brood::Application::CardEditor, 87

m_btnCardDecSizeY
 Brood::Application::CardEditor, 88

m_btnCardDownPromptDecPosX
 Brood::Application::CardEditor, 88

m_btnCardDownPromptDecPosY
 Brood::Application::CardEditor, 88

m_btnCardDownPromptDecSizeX
 Brood::Application::CardEditor, 88

m_btnCardDownPromptDecSizeY
 Brood::Application::CardEditor, 88

m_btnCardDownPromptIncPosX
 Brood::Application::CardEditor, 88

m_btnCardDownPromptIncPosY
 Brood::Application::CardEditor, 89

m_btnCardDownPromptIncSizeX
 Brood::Application::CardEditor, 89

m_btnCardDownPromptIncSizeY
 Brood::Application::CardEditor, 89

m_btnCardDownValueDecPosX
 Brood::Application::CardEditor, 89

m_btnCardDownValueDecPosY
 Brood::Application::CardEditor, 89

m_btnCardDownValueDecSizeX
 Brood::Application::CardEditor, 89

m_btnCardDownValueDecSizeY
 Brood::Application::CardEditor, 90

m_btnCardDownValueIncPosX
 Brood::Application::CardEditor, 90

m_btnCardDownValueIncPosY
 Brood::Application::CardEditor, 90

m_btnCardDownValueIncSizeX
 Brood::Application::CardEditor, 90
m_btnCardDownValueIncSizeY
 Brood::Application::CardEditor, 90
m_btnCardFrontBgOpenFile
 Brood::Application::CardEditor, 90
m_btnCardFrontTimePromptDecPosX
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptDecPosY
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptDecSizeX
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptDecSizeY
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptIncPosX
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptIncPosY
 Brood::Application::CardEditor, 91
m_btnCardFrontTimePromptIncSizeX
 Brood::Application::CardEditor, 92
m_btnCardFrontTimePromptIncSizeY
 Brood::Application::CardEditor, 92
m_btnCardFrontTimeValueDecPosX
 Brood::Application::CardEditor, 92
m_btnCardFrontTimeValueDecPosY
 Brood::Application::CardEditor, 92
m_btnCardFrontTimeValueDecSizeX
 Brood::Application::CardEditor, 92
m_btnCardFrontTimeValueDecSizeY
 Brood::Application::CardEditor, 92
m_btnCardFrontTimeValueIncPosX
 Brood::Application::CardEditor, 93
m_btnCardFrontTimeValueIncPosY
 Brood::Application::CardEditor, 93
m_btnCardFrontTimeValueIncSizeX
 Brood::Application::CardEditor, 93
m_btnCardFrontTimeValueIncSizeY
 Brood::Application::CardEditor, 93
m_btnCardIncPosX
 Brood::Application::CardEditor, 93
m_btnCardIncPosY
 Brood::Application::CardEditor, 93
m_btnCardIncSizeX
 Brood::Application::CardEditor, 94
m_btnCardIncSizeY
 Brood::Application::CardEditor, 94
m_btnCardInitOpenFile
 Brood::Application::DeckEditor, 149
m_btnCardQuestionPromptDecPosX
 Brood::Application::CardEditor, 94
m_btnCardQuestionPromptDecPosY
 Brood::Application::CardEditor, 94
m_btnCardQuestionPromptDecSizeX
 Brood::Application::CardEditor, 94
m_btnCardQuestionPromptDecSizeY
 Brood::Application::CardEditor, 94
m_btnCardQuestionPromptIncPosX
 Brood::Application::CardEditor, 95
m_btnCardQuestionPromptIncPosY
 Brood::Application::CardEditor, 95
m_btnCardQuestionPromptIncSizeX
 Brood::Application::CardEditor, 95
m_btnCardQuestionPromptIncSizeY
 Brood::Application::CardEditor, 95
m_btnCardQuestionValueDecPosX
 Brood::Application::CardEditor, 95
m_btnCardQuestionValueDecPosY
 Brood::Application::CardEditor, 95
m_btnCardQuestionValueDecSizeX
 Brood::Application::CardEditor, 96
m_btnCardQuestionValueDecSizeY
 Brood::Application::CardEditor, 96
m_btnCardQuestionValueIncPosX
 Brood::Application::CardEditor, 96
m_btnCardQuestionValueIncPosY
 Brood::Application::CardEditor, 96
m_btnCardQuestionValueIncSizeX
 Brood::Application::CardEditor, 96
m_btnCardQuestionValueIncSizeY
 Brood::Application::CardEditor, 96
m_btnCardSubmitDecPosX
 Brood::Application::CardEditor, 97
m_btnCardSubmitDecPosY
 Brood::Application::CardEditor, 97
m_btnCardSubmitDecSizeX
 Brood::Application::CardEditor, 97
m_btnCardSubmitDecSizeY
 Brood::Application::CardEditor, 97
m_btnCardSubmitIncPosX
 Brood::Application::CardEditor, 97
m_btnCardSubmitIncPosY
 Brood::Application::CardEditor, 97
m_btnCardSubmitIncSizeX
 Brood::Application::CardEditor, 98
m_btnCardSubmitIncSizeY
 Brood::Application::CardEditor, 98
m_btnCardTurnCardDecPosX
 Brood::Application::CardEditor, 98
m_btnCardTurnCardDecPosY
 Brood::Application::CardEditor, 98
m_btnCardTurnCardDecSizeX
 Brood::Application::CardEditor, 98
m_btnCardTurnCardDecSizeY
 Brood::Application::CardEditor, 98
m_btnCardTurnCardIncPosX
 Brood::Application::CardEditor, 99
m_btnCardTurnCardIncPosY
 Brood::Application::CardEditor, 99
m_btnCardTurnCardIncSizeX
 Brood::Application::CardEditor, 99
m_btnCardTurnCardIncSizeY
 Brood::Application::CardEditor, 99
m_btnCardUpPromptDecPosX
 Brood::Application::CardEditor, 99
m_btnCardUpPromptDecPosY
 Brood::Application::CardEditor, 99

m_btnCardUpPromptDecSizeX
 Brood::Application::CardEditor, 100
m_btnCardUpPromptDecSizeY
 Brood::Application::CardEditor, 100
m_btnCardUpPromptIncPosX
 Brood::Application::CardEditor, 100
m_btnCardUpPromptIncPosY
 Brood::Application::CardEditor, 100
m_btnCardUpPromptIncSizeX
 Brood::Application::CardEditor, 100
m_btnCardUpPromptIncSizeY
 Brood::Application::CardEditor, 100
m_btnCardUpValueDecPosX
 Brood::Application::CardEditor, 101
m_btnCardUpValueDecPosY
 Brood::Application::CardEditor, 101
m_btnCardUpValueDecSizeX
 Brood::Application::CardEditor, 101
m_btnCardUpValueDecSizeY
 Brood::Application::CardEditor, 101
m_btnCardUpValueIncPosX
 Brood::Application::CardEditor, 101
m_btnCardUpValueIncPosY
 Brood::Application::CardEditor, 101
m_btnCardUpValueIncSizeX
 Brood::Application::CardEditor, 102
m_btnCardUpValueIncSizeY
 Brood::Application::CardEditor, 102
m_btnDecCardNum
 Brood::Application::DeckEditor, 149
m_btnDecCurrCardDown
 Brood::Application::CardEditor, 102
m_btnDecCurrCardNum
 Brood::Application::CardEditor, 102
m_btnDecCurrCardTime
 Brood::Application::CardEditor, 102
m_btnDecCurrCardUp
 Brood::Application::CardEditor, 102
m_btnDecDeckNum
 Brood::Application::CardEditor, 103
m_btnDeckBgOpenFile
 Brood::Application::DeckEditor, 149
m_btnDeckDecCurrlIdx
 Brood::Application::DeckEditor, 149
m_btnDeckDecNum
 Brood::Application::GameEditor, 234
m_btnDeckDecPosX
 Brood::Application::DeckEditor, 149
m_btnDeckDecPosY
 Brood::Application::DeckEditor, 149
m_btnDeckDecSizeX
 Brood::Application::DeckEditor, 150
m_btnDeckDecSizeY
 Brood::Application::DeckEditor, 150
m_btnDeckInccurrIdx
 Brood::Application::DeckEditor, 150
m_btnDeckIncNum
 Brood::Application::GameEditor, 234
m_btnDeckIncPosX
 Brood::Application::DeckEditor, 150
m_btnDeckIncPosY
 Brood::Application::DeckEditor, 150
m_btnDeckIncSizeX
 Brood::Application::DeckEditor, 150
m_btnDeckIncSizeY
 Brood::Application::DeckEditor, 151
m_btnDiceDecPosX
 Brood::Application::DiceEditor, 174
m_btnDiceDecPosY
 Brood::Application::DiceEditor, 174
m_btnDiceDecSide
 Brood::Application::DiceEditor, 174
m_btnDiceDecSizeX
 Brood::Application::DiceEditor, 175
m_btnDiceDecSizeY
 Brood::Application::DiceEditor, 175
m_btnDiceIncPosX
 Brood::Application::DiceEditor, 175
m_btnDiceIncPosY
 Brood::Application::DiceEditor, 175
m_btnDiceIncSide
 Brood::Application::DiceEditor, 175
m_btnDiceIncSizeX
 Brood::Application::DiceEditor, 175
m_btnDiceIncSizeY
 Brood::Application::DiceEditor, 176
m_btnDiceOpenFile
 Brood::Application::DiceEditor, 176
m_btnInccurrCardNum
 Brood::Application::DeckEditor, 151
m_btnInccurrCardDown
 Brood::Application::CardEditor, 103
m_btnInccurrCardNum
 Brood::Application::CardEditor, 103
m_btnInccurrCardTime
 Brood::Application::CardEditor, 103
m_btnInccurrCardUp
 Brood::Application::CardEditor, 103
m_btnIncDeckNum
 Brood::Application::CardEditor, 103
m_btnPlayerDecCurrlIdx
 Brood::Application::PlayerEditor, 306
m_btnPlayerDecCurrlIdxOffsetX
 Brood::Application::PlayerEditor, 306
m_btnPlayerDecCurrlIdxOffsetY
 Brood::Application::PlayerEditor, 306
m_btnPlayerDecCurrlIdxSizeX
 Brood::Application::PlayerEditor, 306
m_btnPlayerDecCurrlIdxSizeY
 Brood::Application::PlayerEditor, 306
m_btnPlayerDecMaxnNum
 Brood::Application::GameEditor, 234
m_btnPlayerDecMinNum
 Brood::Application::GameEditor, 234
m_btnPlayerDecStartColNum
 Brood::Application::PlayerEditor, 306

m_btnPlayerDecStartRowNum
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdx
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxOffsetX
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxOffsetY
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxSizeX
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxSizeY
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxSizeX
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncCurrIdxSizeY
 Brood::Application::PlayerEditor, 307

m_btnPlayerIncMaxNum
 Brood::Application::GameEditor, 234

m_btnPlayerIncMinNum
 Brood::Application::GameEditor, 235

m_btnPlayerIncStartColNum
 Brood::Application::PlayerEditor, 308

m_btnPlayerIncStartRowNum
 Brood::Application::PlayerEditor, 308

m_btnPlayerOpenFile
 Brood::Application::PlayerEditor, 308

m_btnRollDice
 Brood::Application::DiceEditor, 176

m_BtnSubmit
 Brood::Application::Components::DisplayCard, 191

m_btnTileBgOpenFile
 Brood::Application::TileEditor, 393

m_btnTileDecAssignedDeckIdx
 Brood::Application::TileEditor, 393

m_btnTileDecCurrCol
 Brood::Application::TileEditor, 393

m_btnTileDecCurrRow
 Brood::Application::TileEditor, 394

m_btnTileDecEndBridgeCol
 Brood::Application::TileEditor, 394

m_btnTileDecEndBridgeRow
 Brood::Application::TileEditor, 394

m_btnTileDecNextPathCol
 Brood::Application::TileEditor, 394

m_btnTileDecNextPathRow
 Brood::Application::TileEditor, 394

m_btnTileDecNumCorrectCard
 Brood::Application::TileEditor, 394

m_btnTileIncAssignedDeckIdx
 Brood::Application::TileEditor, 395

m_btnTileIncCurrCol
 Brood::Application::TileEditor, 395

m_btnTileIncCurrRow
 Brood::Application::TileEditor, 395

m_btnTileIncEndBridgeCol
 Brood::Application::TileEditor, 395

m_btnTileIncEndBridgeRow
 Brood::Application::TileEditor, 395

m_btnTileIncNextPathCol
 Brood::Application::TileEditor, 395

m_btnTileIncNextPathRow
 Brood::Application::TileEditor, 396

m_btnTileIncNumCorrectCard
 Brood::Application::TileEditor, 396

m_BtnTurnCard
 Brood::Application::Components::DisplayCard, 192

m_btnTurnCardAround
 Brood::Application::CardEditor, 104

m_cancleBtn
 Brood::Application::HomeWorkSpace, 243

m_cardInfoToDisplay
 Brood::Application::Components::DisplayCard, 192

m_cardInitFileName
 Brood::Application::Components::Deck, 141

m_cardInitFilePath
 Brood::Application::Components::Deck, 141

m_cardList
 Brood::Application::Components::Deck, 141

m_correctAnswer
 Brood::Application::Components::CardInfo, 133

m_correctNumSteps
 Brood::Application::Components::CardInfo, 133

m_createNewEditBtn
 Brood::Application::HomeWorkSpace, 243

m_currActiveCardIdx
 Brood::Application::Components::Deck, 141

m_currActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 218

m_currActivePathPtr
 Brood::Application::Components::Board, 42

m_currActivePlayerIdx
 Brood::Application::Components::PlayerManager,
 320

m_currDeckIdx
 Brood::Application::Components::DeckManager,
 161

m_currFrameLeftMouseButtonPressed
 Brood::MouseHandler, 276

m_currFrameMousePos
 Brood::MouseHandler, 276

m_currFrameRightMouseButtonPressed
 Brood::MouseHandler, 276

m_currGameInfoIdx
 Brood::Application::HomeWorkSpace, 243

m_currSpriteIndex
 Brood::SpriteHandler, 327

m_currUndealtCardIdx
 Brood::Application::Components::Deck, 141

m_currWorkSpaceIdx
 Brood::Application::HomeWorkSpace, 244

m_cursor
 Brood::MouseHandler, 276

m_ddiDrawPlayerAtCurrIdx
 Brood::Application::PlayerEditor, 308

m_ddiIncorectPenalty
 Brood::Application::GameEditor, 235

m_ddiMovementType
 Brood::Application::GameEditor, 235

m_ddiSettingSelection
 Brood::Application::CardEditor, 104

m_ddiTlForceDiceRoll
 Brood::Application::TileEditor, 396

m_ddiTlType
 Brood::Application::TileEditor, 396

m_debugTextSave
 Brood::BroodUI::UIElement, 419

m_deckIdx
 Brood::Application::Components::Path, 288

m_deckList
 Brood::Application::Components::DeckManager,
 161

m_deckManager
 Brood::Application::Components::GameDataManager,
 227

m_deckPtr
 Brood::Application::Components::Path, 288

m_dice
 Brood::Application::Components::GameDataManager,
 227

m_dicePtr
 Brood::Application::DiceEditor, 176

m_displayCard
 Brood::Application::Components::GameDataManager
 228

m_drawLine
 Brood::Application::Components::Path, 288

m_drawOverlay
 Brood::BroodUI::UIElement, 420

m_drawText
 Brood::BroodUI::UIElement, 420

m_editModesTabs
 Brood::Application::EditorWorkspace, 213

m_editorWorkspaceList
 Brood::Application::EditorWorkspace, 213

m_elementID
 Brood::BroodUI::Id, 254

m_elementId
 Brood::BroodUI::UIElement, 420

m_elementType
 Brood::BroodUI::UIElement, 420

m_enterFileNameFeildTxt
 Brood::Application::HomeWorkSpace, 244

m_enterPressed
 Brood::BroodUI::TextBox, 383

m_events
 Brood::Application::Application, 30

m_fileHandler
 Brood::Application::FileAccess, 222

m_fileNameTxt
 Brood::Application::HomeWorkSpace, 244

m_filePath
 Brood::Application::FileAccess, 222

m_font
 Brood::Application::WorkSpace, 435

m_fontSize
 Brood::BroodUI::UIElement, 420

m_forceDiceRoll

 Brood::Application::Components::Path, 288

m_frontBgFileName
 Brood::Application::Components::DisplayCard, 192

m_gameData
 Brood::Application::WorkSpace, 435

m_gameInfoFilePathList
 Brood::Application::HomeWorkSpace, 244

m_gameListTexture
 Brood::Application::HomeWorkSpace, 244

m_gameOrEditorBn
 Brood::Application::HomeWorkSpace, 244

m_gameOrEditorForeground
 Brood::Application::HomeWorkSpace, 245

m_gameTitle
 Brood::Application::Components::GameDataManager,
 228

m_hasChilds
 Brood::BroodUI::Id, 254

m_hasLimit
 Brood::BroodUI::TextBox, 383

m_hotElementFlag
 Brood::BroodUI::ElementSelection, 218

m_hotElementIdPtr
 Brood::BroodUI::ElementSelection, 218

m_hotOverlayColor
 Brood::BroodUI::UIElement, 421

m_importGameBn
 Brood::Application::HomeWorkSpace, 245

m_incorrectNumSteps
 Brood::Application::Components::CardInfo, 133

m_incorrectPenalty
 Brood::Application::Components::DeckManager,
 161

m_isCurrFront
 Brood::Application::Components::DisplayCard, 192

m_isDebugger
 Brood::Application::WorkSpace, 435

m_isEditable
 Brood::BroodUI::TextBox, 383

m_isHomeWorkspace
 Brood::Application::HomeWorkSpace, 245

m_isSelected
 Brood::BroodUI::UIElement, 421

m_items
 Brood::BroodUI::DropDownMenu, 207

m_lastActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 218

m_lastFrameLeftMouseButtonPressed
 Brood::MouseHandler, 276

m_lastFrameMousePos
 Brood::MouseHandler, 276

m_lastFrameRightMouseButtonPressed
 Brood::MouseHandler, 276

m_limit
 Brood::BroodUI::TextBox, 383

m_loadPreviousEditsBn
 Brood::Application::HomeWorkSpace, 245

m_mainWorkspace

Brood::Application::Application, 30
 m_maxItemLength
 Brood::BroodUI::DropDownMenu, 207
 m_maxPlayer
 Brood::Application::Components::PlayerManager,
 320
 m_menus
 Brood::BroodUI::MenuBar, 268
 m_minPlayer
 Brood::Application::Components::PlayerManager,
 321
 m_movementType
 Brood::Application::Components::DeckManager,
 161
 m_nextPathLine
 Brood::Application::Components::Path, 289
 m_nextPathPtr
 Brood::Application::Components::Path, 289
 m_numCardDraw
 Brood::Application::Components::Path, 289
 m_numCols
 Brood::Application::Components::Board, 43
 m_numRows
 Brood::Application::Components::Board, 43
 m_numSides
 Brood::Application::Components::Dice, 167
 m_openFileBtn
 Brood::Application::HomeWorkSpace, 245
 m_openFileBtnIdx
 Brood::Application::HomeWorkSpace, 245
 m_ossText
 Brood::BroodUI::TextBox, 383
 m_panelBodyPtr
 Brood::Application::BoardEditor, 51
 Brood::Application::CardEditor, 104
 Brood::Application::DeckEditor, 151
 Brood::Application::DiceEditor, 176
 Brood::Application::GameEditor, 235
 Brood::Application::PlayerEditor, 308
 Brood::Application::TileEditor, 396
 m_parentID
 Brood::BroodUI::Id, 255
 m_parentIDPtr
 Brood::BroodUI::Id, 255
 m_placeHolderText
 Brood::BroodUI::TextBox, 384
 m_playerBody
 Brood::Application::Components::Player, 297
 m_playerCurrPathPtr
 Brood::Application::Components::Player, 297
 m_playerManager
 Brood::Application::Components::GameDataManager,
 228
 m_playerManagerPtr
 Brood::Application::PlayerEditor, 308
 m_playerPtrList
 Brood::Application::Components::Path, 289
 m_positionOffsetX
 Brood::Application::Components::Player, 297
 m_positionOffsetY
 Brood::Application::Components::Player, 298
 m_previousPathPtr
 Brood::Application::Components::Path, 289
 m_question
 Brood::Application::Components::CardInfo, 133
 m_ribbonTabs
 Brood::Application::HomeWorkSpace, 245
 m_selectedSettingIdx
 Brood::Application::CardEditor, 104
 m_showEnterFilenameDialogBox
 Brood::Application::HomeWorkSpace, 246
 m_showImportDialogBox
 Brood::Application::HomeWorkSpace, 246
 m_sidePanel
 Brood::Application::EditorWorkspace, 213
 m_spriteBody
 Brood::Application::Components::Player, 298
 m_spriteHeight
 Brood::SpriteHandler, 328
 m_spriteLength
 Brood::SpriteHandler, 328
 m_startPathPtr
 Brood::Application::Components::PlayerManager,
 321
 m_text
 Brood::BroodUI::UIElement, 421
 m_textContent
 Brood::BroodUI::UIElement, 421
 m_texture
 Brood::SpriteHandler, 328
 m_textureDirectoryPath
 Brood::SpriteHandler, 328
 m_textureFileName
 Brood::SpriteHandler, 328
 m_tileColNum
 Brood::Application::Components::Tiles, 405
 m_tilePtr
 Brood::Application::Components::Path, 289
 m_tileRowNum
 Brood::Application::Components::Tiles, 405
 m_tileType
 Brood::Application::Components::Path, 290
 m_time
 Brood::Application::Components::CardInfo, 133
 m_titleScreenBtn
 Brood::Application::GameEditor, 235
 m_toggleGameInfoLeftBtn
 Brood::Application::HomeWorkSpace, 246
 m_toggleGameInfoRightBtn
 Brood::Application::HomeWorkSpace, 246
 m_TxtBackTimePromt
 Brood::Application::Components::DisplayCard, 192
 m_TxtBackTimeValue
 Brood::Application::Components::DisplayCard, 192
 m_txtBoardCol
 Brood::Application::BoardEditor, 51

m_txtBoardColPromt
 Brood::Application::BoardEditor, 51

m_txtBoardPosPromtX
 Brood::Application::BoardEditor, 52

m_txtBoardPosPromtY
 Brood::Application::BoardEditor, 52

m_txtBoardPosX
 Brood::Application::BoardEditor, 52

m_txtBoardPosY
 Brood::Application::BoardEditor, 52

m_txtBoardRow
 Brood::Application::BoardEditor, 52

m_txtBoardRowPromt
 Brood::Application::BoardEditor, 52

m_txtBoardSizePromtX
 Brood::Application::BoardEditor, 53

m_txtBoardSizePromtY
 Brood::Application::BoardEditor, 53

m_txtBoardSizeX
 Brood::Application::BoardEditor, 53

m_txtBoardSizeY
 Brood::Application::BoardEditor, 53

m_txtCardAnswerPromptPosX
 Brood::Application::CardEditor, 104

m_txtCardAnswerPromptPosXPrompt
 Brood::Application::CardEditor, 104

m_txtCardAnswerPromptPosY
 Brood::Application::CardEditor, 104

m_txtCardAnswerPromptPosYPrompt
 Brood::Application::CardEditor, 105

m_txtCardAnswerPromptSizeX
 Brood::Application::CardEditor, 105

m_txtCardAnswerPromptSizeXPrompt
 Brood::Application::CardEditor, 105

m_txtCardAnswerPromptSizeY
 Brood::Application::CardEditor, 105

m_txtCardAnswerPromptSizeYPrompt
 Brood::Application::CardEditor, 105

m_txtCardAnswerValuePosX
 Brood::Application::CardEditor, 105

m_txtCardAnswerValuePosXPrompt
 Brood::Application::CardEditor, 106

m_txtCardAnswerValuePosY
 Brood::Application::CardEditor, 106

m_txtCardAnswerValuePosYPrompt
 Brood::Application::CardEditor, 106

m_txtCardAnswerValueSizeX
 Brood::Application::CardEditor, 106

m_txtCardAnswerValueSizeXPrompt
 Brood::Application::CardEditor, 106

m_txtCardAnswerValueSizeY
 Brood::Application::CardEditor, 106

m_txtCardAnswerValueSizeYPrompt
 Brood::Application::CardEditor, 107

m_txtCardBackBgFileInput
 Brood::Application::CardEditor, 107

m_txtCardBackBgFileNamePrompt
 Brood::Application::CardEditor, 107

m_txtCardBackTimePromptPosX
 Brood::Application::CardEditor, 107

m_txtCardBackTimePromptPosXPrompt
 Brood::Application::CardEditor, 107

m_txtCardBackTimePromptPosY
 Brood::Application::CardEditor, 107

m_txtCardBackTimePromptPosYPrompt
 Brood::Application::CardEditor, 108

m_txtCardBackTimePromptSizeX
 Brood::Application::CardEditor, 108

m_txtCardBackTimePromptSizeXPrompt
 Brood::Application::CardEditor, 108

m_txtCardBackTimePromptSizeY
 Brood::Application::CardEditor, 108

m_txtCardBackTimePromptSizeYPrompt
 Brood::Application::CardEditor, 108

m_txtCardBackTimeValuePosX
 Brood::Application::CardEditor, 108

m_txtCardBackTimeValuePosXPrompt
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValuePosY
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValuePosYPrompt
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValueSizeX
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValueSizeXPrompt
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValueSizeY
 Brood::Application::CardEditor, 109

m_txtCardBackTimeValueSizeYPrompt
 Brood::Application::CardEditor, 110

m_txtCardDownPromptPosX
 Brood::Application::CardEditor, 110

m_txtCardDownPromptPosXPrompt
 Brood::Application::CardEditor, 110

m_txtCardDownPromptPosY
 Brood::Application::CardEditor, 110

m_txtCardDownPromptPosYPrompt
 Brood::Application::CardEditor, 110

m_txtCardDownPromptSizeX
 Brood::Application::CardEditor, 110

m_txtCardDownPromptSizeXPrompt
 Brood::Application::CardEditor, 111

m_txtCardDownPromptSizeY
 Brood::Application::CardEditor, 111

m_txtCardDownPromptSizeYPrompt
 Brood::Application::CardEditor, 111

m_txtCardDownValuePosX
 Brood::Application::CardEditor, 111

m_txtCardDownValuePosXPrompt
 Brood::Application::CardEditor, 111

m_txtCardDownValuePosY
 Brood::Application::CardEditor, 111

m_txtCardDownValuePosYPrompt
 Brood::Application::CardEditor, 112

m_txtCardDownValueSizeX
 Brood::Application::CardEditor, 112

m_txtCardDownValueSizeXPrompt
 Brood::Application::CardEditor, 112
m_txtCardDownValueSizeY
 Brood::Application::CardEditor, 112
m_txtCardDownValueSizeYPrompt
 Brood::Application::CardEditor, 112
m_txtCardFrontBgFileInput
 Brood::Application::CardEditor, 112
m_txtCardFrontBgFileNamePrompt
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptPosX
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptPosXPrompt
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptPosY
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptPosYPrompt
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptSizeX
 Brood::Application::CardEditor, 113
m_txtCardFrontTimePromptSizeXPrompt
 Brood::Application::CardEditor, 114
m_txtCardFrontTimePromptSizeY
 Brood::Application::CardEditor, 114
m_txtCardFrontTimePromptSizeYPrompt
 Brood::Application::CardEditor, 114
m_txtCardFrontTimeValuePosX
 Brood::Application::CardEditor, 114
m_txtCardFrontTimeValuePosXPrompt
 Brood::Application::CardEditor, 114
m_txtCardFrontTimeValuePosY
 Brood::Application::CardEditor, 114
m_txtCardFrontTimeValuePosYPrompt
 Brood::Application::CardEditor, 115
m_txtCardFrontTimeValueSizeX
 Brood::Application::CardEditor, 115
m_txtCardFrontTimeValueSizeXPrompt
 Brood::Application::CardEditor, 115
m_txtCardFrontTimeValueSizeY
 Brood::Application::CardEditor, 115
m_txtCardFrontTimeValueSizeYPrompt
 Brood::Application::CardEditor, 115
m_txtCardInitFileInput
 Brood::Application::DeckEditor, 151
m_txtCardInitFileNamePrompt
 Brood::Application::DeckEditor, 151
m_txtCardNum
 Brood::Application::DeckEditor, 151
m_txtCardNumPrompt
 Brood::Application::DeckEditor, 152
m_txtCardPosX
 Brood::Application::CardEditor, 115
m_txtCardPosXPrompt
 Brood::Application::CardEditor, 116
m_txtCardPosY
 Brood::Application::CardEditor, 116
m_txtCardPosYPrompt
 Brood::Application::CardEditor, 116

m_txtCardQuestionPromptPosX
 Brood::Application::CardEditor, 116
m_txtCardQuestionPromptPosXPrompt
 Brood::Application::CardEditor, 116
m_txtCardQuestionPromptPosY
 Brood::Application::CardEditor, 116
m_txtCardQuestionPromptPosYPrompt
 Brood::Application::CardEditor, 117
m_txtCardQuestionPromptSizeX
 Brood::Application::CardEditor, 117
m_txtCardQuestionPromptSizeXPrompt
 Brood::Application::CardEditor, 117
m_txtCardQuestionPromptSizeY
 Brood::Application::CardEditor, 117
m_txtCardQuestionValuePosX
 Brood::Application::CardEditor, 117
m_txtCardQuestionValuePosXPrompt
 Brood::Application::CardEditor, 118
m_txtCardQuestionValuePosY
 Brood::Application::CardEditor, 118
m_txtCardQuestionValuePosYPrompt
 Brood::Application::CardEditor, 118
m_txtCardQuestionValueSizeX
 Brood::Application::CardEditor, 118
m_txtCardQuestionValueSizeXPrompt
 Brood::Application::CardEditor, 118
m_txtCardQuestionValueSizeY
 Brood::Application::CardEditor, 118
m_txtCardQuestionValueSizeYPrompt
 Brood::Application::CardEditor, 119
m_txtCardSizeX
 Brood::Application::CardEditor, 119
m_txtCardSizeXPrompt
 Brood::Application::CardEditor, 119
m_txtCardSizeY
 Brood::Application::CardEditor, 119
m_txtCardSizeYPrompt
 Brood::Application::CardEditor, 119
m_txtCardSubmitPosX
 Brood::Application::CardEditor, 119
m_txtCardSubmitPosXPrompt
 Brood::Application::CardEditor, 120
m_txtCardSubmitPosY
 Brood::Application::CardEditor, 120
m_txtCardSubmitPosYPrompt
 Brood::Application::CardEditor, 120
m_txtCardSubmitSizeX
 Brood::Application::CardEditor, 120
m_txtCardSubmitSizeXPrompt
 Brood::Application::CardEditor, 120
m_txtCardSubmitSizeY
 Brood::Application::CardEditor, 120
m_txtCardSubmitSizeYPrompt
 Brood::Application::CardEditor, 121
m_txtCardTurnCardPosX
 Brood::Application::CardEditor, 121

m_txtCardTurnCardPosXPrompt
 Brood::Application::CardEditor, 121

m_txtCardTurnCardPosY
 Brood::Application::CardEditor, 121

m_txtCardTurnCardPosYPrompt
 Brood::Application::CardEditor, 121

m_txtCardTurnCardSizeX
 Brood::Application::CardEditor, 121

m_txtCardTurnCardSizeXPrompt
 Brood::Application::CardEditor, 122

m_txtCardTurnCardSizeY
 Brood::Application::CardEditor, 122

m_txtCardTurnCardSizeYPrompt
 Brood::Application::CardEditor, 122

m_txtCardUpPromptPosX
 Brood::Application::CardEditor, 122

m_txtCardUpPromptPosXPrompt
 Brood::Application::CardEditor, 122

m_txtCardUpPromptPosY
 Brood::Application::CardEditor, 122

m_txtCardUpPromptPosYPrompt
 Brood::Application::CardEditor, 123

m_txtCardUpPromptSizeX
 Brood::Application::CardEditor, 123

m_txtCardUpPromptSizeXPrompt
 Brood::Application::CardEditor, 123

m_txtCardUpPromptSizeY
 Brood::Application::CardEditor, 123

m_txtCardUpValuePosX
 Brood::Application::CardEditor, 123

m_txtCardUpValuePosXPrompt
 Brood::Application::CardEditor, 124

m_txtCardUpValuePosY
 Brood::Application::CardEditor, 124

m_txtCardUpValuePosYPrompt
 Brood::Application::CardEditor, 124

m_txtCardUpValueSizeX
 Brood::Application::CardEditor, 124

m_txtCardUpValueSizeXPrompt
 Brood::Application::CardEditor, 124

m_txtCardUpValueSizeY
 Brood::Application::CardEditor, 124

m_txtCardUpValueSizeYPrompt
 Brood::Application::CardEditor, 125

m_txtCurrCardAnswerInput
 Brood::Application::CardEditor, 125

m_txtCurrCardAnswerPrompt
 Brood::Application::CardEditor, 125

m_txtCurrCardDown
 Brood::Application::CardEditor, 125

m_txtCurrCardDownPrompt
 Brood::Application::CardEditor, 125

m_txtCurrCardNum
 Brood::Application::CardEditor, 125

m_txtCurrCardNumPrompt
 Brood::Application::CardEditor, 126

m_txtCurrCardQuestionInput
 Brood::Application::CardEditor, 126

m_txtCurrCardQuestionPrompt
 Brood::Application::CardEditor, 126

m_txtCurrCardTime
 Brood::Application::CardEditor, 126

m_txtCurrCardTimePrompt
 Brood::Application::CardEditor, 126

m_txtCurrCardUp
 Brood::Application::CardEditor, 126

m_txtCurrCardUpPrompt
 Brood::Application::CardEditor, 127

m_txtDeckBgFileInput
 Brood::Application::DeckEditor, 152

m_txtDeckBgFileNamePrompt
 Brood::Application::DeckEditor, 152

m_txtDeckCurrIdx
 Brood::Application::DeckEditor, 152

m_txtDeckCurrIdxPrompt
 Brood::Application::DeckEditor, 152

m_txtDeckNum
 Brood::Application::CardEditor, 127

m_txtDeckNumPrompt
 Brood::Application::GameEditor, 235

m_txtDeckPosX
 Brood::Application::DeckEditor, 152

m_txtDeckPosXPrompt
 Brood::Application::DeckEditor, 153

m_txtDeckPosY
 Brood::Application::DeckEditor, 153

m_txtDeckPosYPrompt
 Brood::Application::DeckEditor, 153

m_txtDeckSizeX
 Brood::Application::DeckEditor, 153

m_txtDeckSizeXPrompt
 Brood::Application::DeckEditor, 153

m_txtDeckSizeY
 Brood::Application::DeckEditor, 153

m_txtDeckSizeYPrompt
 Brood::Application::DeckEditor, 154

m_txtDiceFileInput
 Brood::Application::DiceEditor, 176

m_txtDiceFileNamePrompt
 Brood::Application::DiceEditor, 177

m_txtDicePosX
 Brood::Application::DiceEditor, 177

m_txtDicePosXPrompt
 Brood::Application::DiceEditor, 177

m_txtDicePosY
 Brood::Application::DiceEditor, 177

m_txtDicePosYPrompt
 Brood::Application::DiceEditor, 177

m_txtDiceSide
 Brood::Application::DiceEditor, 177

m_txtDiceSidePrompt
 Brood::Application::DiceEditor, 178

m_txtDiceSizeX
 Brood::Application::DiceEditor, 178

m_txtDiceSizeXPrompt
 Brood::Application::DiceEditor, 178

m_txtDiceSizeY
 Brood::Application::DiceEditor, 178

m_txtDiceSizeYPrompt
 Brood::Application::DiceEditor, 178

m_TxtDownPrompt
 Brood::Application::Components::DisplayCard, 192

m_TxtDownValue
 Brood::Application::Components::DisplayCard, 193

m_txtDrawPlayerAtCurrlIdxPromt
 Brood::Application::PlayerEditor, 309

m_TxtFrontTimePromt
 Brood::Application::Components::DisplayCard, 193

m_TxtFrontTimeValue
 Brood::Application::Components::DisplayCard, 193

m_txtGameInput
 Brood::Application::GameEditor, 236

m_txtGameNamePrompt
 Brood::Application::GameEditor, 236

m_txtIncorectPenaltyPromt
 Brood::Application::GameEditor, 236

m_txtMovementTypePromt
 Brood::Application::GameEditor, 236

m_txtPlayerCurrlIdx
 Brood::Application::PlayerEditor, 309

m_txtPlayerCurrlIdxOffsetX
 Brood::Application::PlayerEditor, 309

m_txtPlayerCurrlIdxOffsetXPrompt
 Brood::Application::PlayerEditor, 309

m_txtPlayerCurrlIdxOffsetY
 Brood::Application::PlayerEditor, 309

m_txtPlayerCurrlIdxOffsetYPrompt
 Brood::Application::PlayerEditor, 309

m_txtPlayerCurrlIdxPrompt
 Brood::Application::PlayerEditor, 310

m_txtPlayerCurrlIdxSizeX
 Brood::Application::PlayerEditor, 310

m_txtPlayerCurrlIdxSizeXPrompt
 Brood::Application::PlayerEditor, 310

m_txtPlayerCurrlIdxSizeY
 Brood::Application::PlayerEditor, 310

m_txtPlayerCurrlIdxSizeYPrompt
 Brood::Application::PlayerEditor, 310

m_txtPlayerFileInput
 Brood::Application::PlayerEditor, 310

m_txtplayerFileNamePrompt
 Brood::Application::PlayerEditor, 311

m_txtPlayerMaxNum
 Brood::Application::GameEditor, 236

m_txtPlayerMaxNumPrompt
 Brood::Application::GameEditor, 237

m_txtPlayerMinNum
 Brood::Application::GameEditor, 237

m_txtPlayerMinNumPrompt
 Brood::Application::GameEditor, 237

m_txtPlayerStartColNum
 Brood::Application::PlayerEditor, 311

m_txtPlayerStartColPrompt
 Brood::Application::PlayerEditor, 311

m_txtPlayerStartRowNum
 Brood::Application::PlayerEditor, 311

m_txtPlayerStartRowPrompt
 Brood::Application::PlayerEditor, 311

m_TxtQuestionPrompt
 Brood::Application::Components::DisplayCard, 193

m_TxtQuestionValue
 Brood::Application::Components::DisplayCard, 193

m_txtSettingTitle
 Brood::Application::BoardEditor, 53

m_txtTileAssignedDeckIdx
 Brood::Application::TileEditor, 397

m_txtTileAssignedDeckIdxPromt
 Brood::Application::TileEditor, 397

m_txtTileBgFileInput
 Brood::Application::TileEditor, 397

m_txtTileBgFileNamePrompt
 Brood::Application::TileEditor, 397

m_txtTileCurrCol
 Brood::Application::TileEditor, 397

m_txtTileCurrColPromt
 Brood::Application::TileEditor, 397

m_txtTileCurrRow
 Brood::Application::TileEditor, 398

m_txtTileCurrRowPromt
 Brood::Application::TileEditor, 398

m_txtTileEndBridgeCol
 Brood::Application::TileEditor, 398

m_txtTileEndBridgeColPromt
 Brood::Application::TileEditor, 398

m_txtTileEndBridgeRow
 Brood::Application::TileEditor, 398

m_txtTileEndBridgeRowPromt
 Brood::Application::TileEditor, 398

m_txtTileForceDiceRollPromt
 Brood::Application::TileEditor, 399

m_txtTileNextPathCol
 Brood::Application::TileEditor, 399

m_txtTileNextPathColPromt
 Brood::Application::TileEditor, 399

m_txtTileNextPathRow
 Brood::Application::TileEditor, 399

m_txtTileNextPathRowPromt
 Brood::Application::TileEditor, 399

m_txtTileNumCorrectCard
 Brood::Application::TileEditor, 399

m_txtTileNumCorrectCardPromt
 Brood::Application::TileEditor, 400

m_txtTileTypePromt

Brood::Application::TileEditor, 400
m_TxtUpPrompt
 Brood::Application::Components::DisplayCard, 193
m_TxtUpValue
 Brood::Application::Components::DisplayCard, 193
m_TxtUserAnswerPrompt
 Brood::Application::Components::DisplayCard, 194
m_TxtUserAnswerValue
 Brood::Application::Components::DisplayCard, 194
m_unNamedUIList
 Brood::Application::WorkSpace, 436
m_window
 Brood::Application::Application, 30
m_workSpacesList
 Brood::Application::HomeWorkSpace, 246
main
 Main.cpp, 504
Main.cpp
 main, 504
MapIdToElement, 258
MenuBar, 268
 Brood::BroodUI::MenuBar, 260
MOVEMENT_cardOnly
 Brood::Application::Components, 23
MOVEMENT_diceOnly
 Brood::Application::Components, 23
MOVEMENT_diceThenCard
 Brood::Application::Components, 23
myBoard
 Brood::Application::Application, 30
myDice
 Brood::Application::Application, 30

OpenFile
 Brood::Application:: FileAccess, 221
operator=
 Brood::Application::Components::CardInfo, 131
 Brood::Application::Components::Deck, 139
 Brood::Application::Components::DeckManager,
 159
 Brood::Application::Components::Dice, 165
 Brood::Application::Components::DisplayCard, 188
 Brood::Application::Components::Path, 284
 Brood::Application::Components::Player, 295
 Brood::Application::Components::PlayerManager,
 318
 Brood::Application::Components::Tiles, 403
 Brood::BroodUI::Button, 58
 Brood::BroodUI::DropDownInput, 196
 Brood::BroodUI::DropDownMenu, 202
 Brood::BroodUI::Id, 253
 Brood::BroodUI::MenuBar, 263
 Brood::BroodUI::TextBox, 380
 Brood::BroodUI::UIElement, 414
 Brood::SpriteHandler, 325
operator==
 Path.cpp, 511
 Path.h, 512

Path, 290
 Brood::Application::Components::Path, 279
Path.cpp
 operator==, 511
Path.h
 operator==, 512
Player, 298
 Brood::Application::Components::Player, 292
PlayerEditor, 312
 Brood::Application::PlayerEditor, 301
PlayerManager, 321
 Brood::Application::Components::PlayerManager,
 314
PollEvents
 Brood::Application::Application, 29
PopulateFromString
 Brood::Application::Data::ST_BoardData, 331
 Brood::Application::Data::ST_CardInfoPrefabData,
 334
 Brood::Application::Data::ST_DeckManagerData,
 339
 Brood::Application::Data::ST_DeckPrefabData,
 341
 Brood::Application::Data::ST_DicePrefabData, 344
 Brood::Application::Data::ST_DisplayCardData,
 349
 Brood::Application::Data::ST_GameData, 364
 Brood::Application::Data::ST_PathPrefabData, 367
 Brood::Application::Data::ST_PlayerManagerData,
 370
 Brood::Application::Data::ST_PlayerPrefabData,
 372

RemoveAllContent
 Brood::Application:: FileAccess, 221
RemoveTexture
 Brood::SpriteHandler, 325
ReomveFromMap
 Brood::BroodUI::MapIdToElement, 257
ReplacePlayerAt
 Brood::Application::Components::PlayerManager,
 318
ResetDeck
 Brood::Application::Components::Deck, 139
ResetGameData
 Brood::Application::HomeWorkSpace, 241
Rewind
 Brood::Application:: FileAccess, 221
RollDice
 Brood::Application::Components::Dice, 165
RunApplicaiton
 Brood::Application::Application, 29

SaveCardToInitFile
 Brood::Application::Components::Deck, 139
SaveDataToFile
 Brood::Application::Components::Board, 39
 Brood::Application::Components::Deck, 140

Brood::Application::Components::DeckManager, 159
 Brood::Application::Components::Dice, 165
 Brood::Application::Components::DisplayCard, 189
 Brood::Application::Components::GameDataManagerSetDeckIdx, 226
 Brood::Application::Components::Player, 295
 Brood::Application::Components::PlayerManager, 319
SaveGameData
 Brood::Application::HomeWorkSpace, 242
SetActiveOverlayColor
 Brood::BroodUI::DropDownMenu, 203
 Brood::BroodUI::MenuBar, 264
 Brood::BroodUI::UIElement, 415
SetAlmostActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 216
SetBackBgFileName
 Brood::Application::Components::DisplayCard, 189
SetBoardPos
 Brood::Application::Components::Board, 39, 40
SetBoardSize
 Brood::Application::Components::Board, 40
SetBodyColor
 Brood::BroodUI::DropDownMenu, 203
 Brood::BroodUI::MenuBar, 264
 Brood::BroodUI::UIElement, 415
SetBodyPosition
 Brood::BroodUI::DropDownMenu, 203, 204
 Brood::BroodUI::MenuBar, 264, 265
 Brood::BroodUI::UIElement, 415, 416
SetBodySize
 Brood::Application::Components::Dice, 166
 Brood::BroodUI::DropDownMenu, 204, 205
 Brood::BroodUI::MenuBar, 265, 266
 Brood::BroodUI::UIElement, 416, 417
SetBridgeEndPathPtr
 Brood::Application::Components::Path, 285
SetCardFront
 Brood::Application::Components::DisplayCard, 189
SetCardInfoToDisplay
 Brood::Application::Components::DisplayCard, 190
SetCardListSize
 Brood::Application::Components::Deck, 140
SetCol
 Brood::Application::Components::Tiles, 403
SetCorrectAnswer
 Brood::Application::Components::CardInfo, 131
SetCorrectNumSteps
 Brood::Application::Components::CardInfo, 132
SetCurrActiveCardIdx
 Brood::Application::Components::Deck, 140
SetCurrActiveDeckIdx
 Brood::Application::Components::DeckManager, 160
SetCurrActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 216
SetCurrActivePlayerIdx

Brood::Application::Components::PlayerManager, 319
SetCurrentActiveTilePtr
 Brood::Application::Components::Board, 41
SetDeckIdx
 Brood::Application::Components::Path, 285
SetDeckPtr
 Brood::Application::Components::Path, 285
SetDeckSize
 Brood::Application::Components::DeckManager, 160
SetDrawOverlay
 Brood::BroodUI::UIElement, 417
SetEachItemPos
 Brood::BroodUI::DropDownMenu, 205
SetEachItemSize
 Brood::BroodUI::DropDownMenu, 205
SetEditable
 Brood::BroodUI::TextBox, 381
SetEnterPressedFalse
 Brood::BroodUI::TextBox, 381
SetFont
 Brood::Application::WorkSpace, 432
 Brood::BroodUI::DropDownMenu, 205
 Brood::BroodUI::MenuBar, 266
 Brood::BroodUI::UIElement, 417
SetFontColor
 Brood::BroodUI::DropDownMenu, 206
 Brood::BroodUI::MenuBar, 267
 Brood::BroodUI::UIElement, 417
SetFontSize
 Brood::BroodUI::DropDownMenu, 206
 Brood::BroodUI::MenuBar, 267
 Brood::BroodUI::UIElement, 417
SetForceDiceRoll
 Brood::Application::Components::Path, 285
SetFrontBgFileName
 Brood::Application::Components::DisplayCard, 190
SetGameDataManager
 Brood::Application::EditorWorkspace, 211
 Brood::Application::WorkSpace, 432
SetTitle
 Brood::Application::Components::GameDataManager, 227
SetHotElementIdPtr
 Brood::BroodUI::ElementSelection, 217
SetHotElementIdPtrFlag
 Brood::BroodUI::ElementSelection, 217
SetHotOverlayColor
 Brood::BroodUI::DropDownMenu, 206
 Brood::BroodUI::MenuBar, 267
 Brood::BroodUI::UIElement, 418
SetIncorrectNumSteps
 Brood::Application::Components::CardInfo, 132
SetIncorrectPenalty
 Brood::Application::Components::DeckManager, 160
SetItemPos

Brood::BroodUI::DropDownMenu, 206
SetItemSize
 Brood::BroodUI::DropDownMenu, 207
SetLastActiveElementIdPtr
 Brood::BroodUI::ElementSelection, 217
SetLimit
 Brood::BroodUI::TextBox, 381
SetMaxPlayer
 Brood::Application::Components::PlayerManager,
 319
SetMenuBodySize
 Brood::BroodUI::MenuBar, 268
SetMenuPos
 Brood::BroodUI::MenuBar, 268
SetMinPlayer
 Brood::Application::Components::PlayerManager,
 319
SetMovementType
 Brood::Application::Components::DeckManager,
 160
SetNextPathPtr
 Brood::Application::Components::Path, 286
SetNumCardDraw
 Brood::Application::Components::Path, 286
SetNumCol
 Brood::Application::Components::Board, 41
SetNumRow
 Brood::Application::Components::Board, 41
SetNumSides
 Brood::Application::Components::Dice, 166
SetParent
 Brood::BroodUI::Id, 254
SetPathForPlayerAtCurrlIdx
 Brood::Application::Components::PlayerManager,
 320
SetPlaceHolderText
 Brood::BroodUI::TextBox, 382
SetPlayerSizeX
 Brood::Application::Components::Player, 295
SetPlayerSizeY
 Brood::Application::Components::Player, 296
SetPlayerStartPath
 Brood::Application::Components::PlayerManager,
 320
SetPositionOffsetX
 Brood::Application::Components::Player, 296
SetPositionOffsetY
 Brood::Application::Components::Player, 296
SetPreviouPathPtr
 Brood::Application::Components::Path, 286
SetQuestion
 Brood::Application::Components::CardInfo, 132
SetRow
 Brood::Application::Components::Tiles, 403
SetRowAndCol
 Brood::Application::Components::Tiles, 404
SetSelected
 Brood::BroodUI::Button, 58

 Brood::BroodUI::TextBox, 382
 Brood::BroodUI::UIElement, 418
SetSpriteFromTexture
 Brood::SpriteHandler, 325
SetSpriteHeight
 Brood::SpriteHandler, 326
SetSpriteLength
 Brood::SpriteHandler, 326
SetText
 Brood::BroodUI::DropDownMenu, 207
 Brood::BroodUI::TextBox, 382
 Brood::BroodUI::UIElement, 418
SetTextPosition
 Brood::BroodUI::UIElement, 419
SetTexture
 Brood::Application::Components::Dice, 167
SetTextureFromFilePath
 Brood::SpriteHandler, 326, 327
SetTextureFromSavedFilePath
 Brood::SpriteHandler, 327
SetTilePtr
 Brood::Application::Components::Path, 287
SetTileType
 Brood::Application::Components::Path, 287
SetTime
 Brood::Application::Components::CardInfo, 132
Shuffel
 Brood::Application::Components::Deck, 140
SpriteHandler, 329
 Brood::SpriteHandler, 323
ST_ApplicationData, 329
ST_BoardData, 332
St_CardInfo, 333
ST_CardPrefabData, 335
ST_ColorVariables, 338
ST_DeckManagerData, 340
ST_DeckPrefabData, 343
ST_DicePrefabData, 346
ST_DisplayCardData, 360
ST_PlayerManagerData, 371
ST_PlayerPrefabData, 374
ST_TilePrefabData, 374
stdafx.h
 WIN32_LEAN_AND_MEAN, 526
stm_answerPromptPositionX
 Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerPromptPositonY
 Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerPromptSizeX
 Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerPromptSizeY
 Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerValuePositionX

Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerValuePositonY
 Brood::Application::Data::ST_DisplayCardData,
 350
stm_answerValueSizeX
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_answerValueSizeY
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_AppPrimaryColor
 Brood::Application::StaticVariables::ST_ColorVariables,
 336
stm_AppSecondaryColor
 Brood::Application::StaticVariables::ST_ColorVariables,
 336
stm_assets
 Brood::Application::StaticVariables::ST_Folders,
 361
stm_assignedDeckId
 Brood::Application::Data::ST_PathPrefabData, 368
stm_backTimePromptPositionX
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_backTimePromptPositonY
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_backTimePromptSizeX
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_backTimePromptSizeY
 Brood::Application::Data::ST_DisplayCardData,
 351
stm_backTimeValuePositionX
 Brood::Application::Data::ST_DisplayCardData,
 352
stm_backTimeValuePositonY
 Brood::Application::Data::ST_DisplayCardData,
 352
stm_backTimeValueSizeX
 Brood::Application::Data::ST_DisplayCardData,
 352
stm_backTimeValueSizeY
 Brood::Application::Data::ST_DisplayCardData,
 352
stm_Black
 Brood::Application::StaticVariables::ST_ColorVariables
 337
stm_boardPosX
 Brood::Application::Data::ST_BoardData, 331
stm_boardPosY
 Brood::Application::Data::ST_BoardData, 331
stm_boardSizeX
 Brood::Application::Data::ST_BoardData, 331
stm_boardSizeY
 Brood::Application::Data::ST_BoardData, 331
stm_cardInitFilename
 Brood::Application::Data::ST_DicePrefabData, 345

Brood::Application::Data::ST_DeckPrefabData,
 342
stm_cardTextures
 Brood::Application::StaticVariables::ST_Folders,
 361
stm_ColNum
 Brood::Application::Data::ST_PathPrefabData, 368
stm_correctAnswer
 Brood::Application::Data::ST_CardInfoPrefabData,
 334
stm_correctNumSteps
 Brood::Application::Data::ST_CardInfoPrefabData,
 334
stm_currActiveCardIdx
 Brood::Application::Data::ST_DeckPrefabData,
 342
stm_currActiveNumCol
 Brood::Application::Data::ST_BoardData, 332
stm_currActiveNumRow
 Brood::Application::Data::ST_BoardData, 332
stm_CurrActiveOverlay
 Brood::Application::StaticVariables::ST_ColorVariables,
 337
stm_currCol
 Brood::Application::Data::ST_PlayerPrefabData,
 373
stm_currDecksIdx
 Brood::Application::Data::ST_DeckManagerData,
 339
stm_currPlayerIdx
 Brood::Application::Data::ST_PlayerManagerData,
 370
stm_currRow
 Brood::Application::Data::ST_PlayerPrefabData,
 373
stm_cwd
 Brood::Application::StaticVariables::ST_Folders,
 361
stm_data
 Brood::Application::StaticVariables::ST_Folders,
 362
stm_deckPosX
 Brood::Application::Data::ST_DeckPrefabData,
 342
stm_deckPosY
 Brood::Application::Data::ST_DeckPrefabData,
 342
stm_deckSizeX
 Brood::Application::Data::ST_DeckPrefabData,
 342
stm_deckSizeY
 Brood::Application::Data::ST_DeckPrefabData,
 342
stm_deckTextures
 Brood::Application::StaticVariables::ST_Folders,
 362
stm_dicePosX
 Brood::Application::Data::ST_DicePrefabData, 345

stm_dicePosY
 Brood::Application::Data::ST_DicePrefabData, 345

stm_diceSizeX
 Brood::Application::Data::ST_DicePrefabData, 345

stm_diceSizeY
 Brood::Application::Data::ST_DicePrefabData, 345

stm_diceTextures
 Brood::Application::StaticVariables::ST_Folders, 362

stm_displayCardBackTextureFilename
 Brood::Application::Data::ST_DisplayCardData, 352

stm_displayCardFrontTextureFilename
 Brood::Application::Data::ST_DisplayCardData, 352

stm_displayCardPositionX
 Brood::Application::Data::ST_DisplayCardData, 353

stm_displayCardPositionY
 Brood::Application::Data::ST_DisplayCardData, 353

stm_displayCardSizeX
 Brood::Application::Data::ST_DisplayCardData, 353

stm_displayCardSizeY
 Brood::Application::Data::ST_DisplayCardData, 353

stm_downPromptPositionX
 Brood::Application::Data::ST_DisplayCardData, 353

stm_downPromptPositionY
 Brood::Application::Data::ST_DisplayCardData, 353

stm_downPromptSizeX
 Brood::Application::Data::ST_DisplayCardData, 354

stm_downPromptSizeY
 Brood::Application::Data::ST_DisplayCardData, 354

stm_downValuePositionX
 Brood::Application::Data::ST_DisplayCardData, 354

stm_downValuePositionY
 Brood::Application::Data::ST_DisplayCardData, 354

stm_downValueSizeX
 Brood::Application::Data::ST_DisplayCardData, 354

stm_downValueSizeY
 Brood::Application::Data::ST_DisplayCardData, 354

stm_endBridgeTileColNum
 Brood::Application::Data::ST_PathPrefabData, 368

stm_endBridgeTileRowNum
 Brood::Application::Data::ST_PathPrefabData, 368

stm_ErrorColor
 Brood::Application::StaticVariables::ST_ColorVariables, 337

stm_exit
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 365

stm_font
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 365

stm_fonts
 Brood::Application::StaticVariables::ST_Folders, 362

stm_forceDiceRoll
 Brood::Application::Data::ST_PathPrefabData, 368

stm_frontTimePromptPositionX
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimePromptPositionY
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimePromptSizeX
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimePromptSizeY
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimeValuePositionX
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimeValuePositionY
 Brood::Application::Data::ST_DisplayCardData, 355

stm_frontTimeValueSizeX
 Brood::Application::Data::ST_DisplayCardData, 356

stm_frontTimeValueSizeY
 Brood::Application::Data::ST_DisplayCardData, 356

stm_gameTitle
 Brood::Application::Data::ST_GameData, 364

stm_HotOverlay
 Brood::Application::StaticVariables::ST_ColorVariables, 337

stm_incorrectNumSteps
 Brood::Application::Data::ST_CardInfoPrefabData, 334

stm_incorrectPenalty
 Brood::Application::Data::ST_DeckManagerData, 339

stm_is_debug_mode
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 365

stm_MainMenu
 Brood::Application::StaticVariables::ST_ColorVariables, 337

stm_mapper
 Brood::BroodUI::MapIdToElement, 257

stm_maxPlayer
 Brood::Application::Data::ST_PlayerManagerData, 371

stm_minPlayer

Brood::Application::Data::ST_PlayerManagerData, 371
 stm_movementType
 Brood::Application::Data::ST_DeckManagerData, 339
 stm_nextTileColNum
 Brood::Application::Data::ST_PathPrefabData, 368
 stm_nextTileRowNum
 Brood::Application::Data::ST_PathPrefabData, 368
 stm_numberCardDraw
 Brood::Application::Data::ST_PathPrefabData, 369
 stm_numCol
 Brood::Application::Data::ST_BoardData, 332
 stm_numDecks
 Brood::Application::Data::ST_DeckManagerData, 340
 stm_numRow
 Brood::Application::Data::ST_BoardData, 332
 stm_numSides
 Brood::Application::Data::ST_DicePrefabData, 345
 stm_numTotalCard
 Brood::Application::Data::ST_DeckPrefabData, 343
 stm_panelPercentage
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 365
 stm_playerOffsetX
 Brood::Application::Data::ST_PlayerPrefabData, 373
 stm_playerOffsetY
 Brood::Application::Data::ST_PlayerPrefabData, 373
 stm_playerSizeX
 Brood::Application::Data::ST_PlayerPrefabData, 373
 stm_playerSizeY
 Brood::Application::Data::ST_PlayerPrefabData, 373
 stm_playerTextures
 Brood::Application::StaticVariables::ST_Folders, 362
 stm_question
 Brood::Application::Data::ST_CardInfoPrefabData, 335
 stm_questionPromptPositionX
 Brood::Application::Data::ST_DisplayCardData, 356
 stm_questionPromptPositionY
 Brood::Application::Data::ST_DisplayCardData, 356
 stm_questionPromptSizeX
 Brood::Application::Data::ST_DisplayCardData, 356
 stm_questionPromptSizeY
 Brood::Application::Data::ST_DisplayCardData, 356
 stm_questionValuePositionX
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_questionValuePositionY
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_questionValueSizeX
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_questionValueSizeY
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_RowNum
 Brood::Application::Data::ST_PathPrefabData, 369
 stm_seed
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 365
 stm_submitButtonValuePositionX
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_submitButtonValuePositionY
 Brood::Application::Data::ST_DisplayCardData, 357
 stm_submitButtonValueSizeX
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_submitButtonValueSizeY
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_textureFileName
 Brood::Application::Data::ST_DeckPrefabData, 343
 Brood::Application::Data::ST_DicePrefabData, 345
 Brood::Application::Data::ST_PathPrefabData, 369
 Brood::Application::Data::ST_PlayerPrefabData, 374
 stm_textures
 Brood::Application::StaticVariables::ST_Folders, 363
 stm_tileTextures
 Brood::Application::StaticVariables::ST_Folders, 363
 stm_tileType
 Brood::Application::Data::ST_PathPrefabData, 369
 stm_time
 Brood::Application::Data::ST_CardInfoPrefabData, 335
 stm_turnCardPromptPositionX
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_turnCardPromptPositionY
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_turnCardPromptSizeX
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_turnCardPromptSizeY
 Brood::Application::Data::ST_DisplayCardData, 358
 stm_undealtCardIdx

Brood::Application::Data::ST_DeckPrefabData, 343
stm_upPromptPositionX
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upPromptPositonY
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upPromptSizeX
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upPromptSizeY
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upValuePositionX
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upValuePositonY
 Brood::Application::Data::ST_DisplayCardData, 359
stm_upValueSizeX
 Brood::Application::Data::ST_DisplayCardData, 360
stm_upValueSizeY
 Brood::Application::Data::ST_DisplayCardData, 360
stm_White
 Brood::Application::StaticVariables::ST_ColorVariables, 337
stm_window_height
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 366
stm_window_width
 Brood::Application::StaticVariables::ST_GlobalCoreVariables, 366
SwitchToHomeWorkspace
 Brood::Application::HomeWorkSpace, 242
TextBox, 384
 Brood::BroodUI::TextBox, 376, 377
TextBox.h
 DELETE_KEY, 528
 ENTER_KEY, 528
 ESCAPE_KEY, 528
TILE_blank
 Brood::Application::Components, 23
TILE_bridge
 Brood::Application::Components, 23
TILE_end
 Brood::Application::Components, 23
TILE_start
 Brood::Application::Components, 23
TILE_tile
 Brood::Application::Components, 23
TileEditor, 400
 Brood::Application::TileEditor, 388
Tiles, 405
 Brood::Application::Components::Tiles, 402
ToggleCardFace
 Brood::Application::Components::DisplayCard, 190
ToggleDrawLine
 Brood::Application::Components::Board, 41
 Brood::Application::Components::Path, 287
TypeOn
 Brood::BroodUI::TextBox, 382
UI_button
 Brood::BroodUI, 24
UI_dropDownInput
 Brood::BroodUI, 24
UI_dropDownMenu
 Brood::BroodUI, 24
UI_menuBar
 Brood::BroodUI, 24
UI_scrollBar
 Brood::BroodUI, 24
UI_textBox
 Brood::BroodUI, 24
UIElement, 421
 Brood::BroodUI::UIElement, 408, 409
Update
 Brood::Application::BoardEditor, 47
 Brood::Application::CardEditor, 77
 Brood::Application::DeckEditor, 145
 Brood::Application::DiceEditor, 171
 Brood::Application::EditorWorkspace, 212
 Brood::Application::GameEditor, 232
 Brood::Application::HomeWorkSpace, 242
 Brood::Application::PlayerEditor, 302
 Brood::Application::TileEditor, 390
 Brood::Application::WorkSpace, 432
UpdateActiveEditorIdx
 Brood::Application::EditorWorkspace, 212
UpdateAllDispalyElement
 Brood::Application::BoardEditor, 47
 Brood::Application::CardEditor, 77
 Brood::Application::DeckEditor, 146
 Brood::Application::DiceEditor, 171
 Brood::Application::EditorWorkspace, 212
 Brood::Application::GameEditor, 232
 Brood::Application::HomeWorkSpace, 242
 Brood::Application::PlayerEditor, 303
 Brood::Application::TileEditor, 390
 Brood::Application::WorkSpace, 433
UpdateAssinedDeckIdx
 Brood::Application::TileEditor, 390
UpdateBoardColPanelElement
 Brood::Application::BoardEditor, 47
UpdateBoardPath
 Brood::Application::Components::Board, 42
UpdateBoardRowPanelElement
 Brood::Application::BoardEditor, 48
UpdateBoardXPosPanelElement
 Brood::Application::BoardEditor, 48
UpdateBoardXSizePanelElement
 Brood::Application::BoardEditor, 48
UpdateBoardYPosPanelElement
 Brood::Application::BoardEditor, 48

UpdateBoardYSizePanelElement
 Brood::Application::BoardEditor, 49

UpdateCardBackFileTexture
 Brood::Application::CardEditor, 77

UpdateCardDisplayBcompPanel
 Brood::Application::CardEditor, 78

UpdateCardDisplayFcompPanel
 Brood::Application::CardEditor, 78

UpdateCardDisplayPanel
 Brood::Application::CardEditor, 78

UpdateCardFrontFileTexture
 Brood::Application::CardEditor, 79

UpdateCardInfoPanel
 Brood::Application::CardEditor, 79

UpdateCardInitFile
 Brood::Application::DeckEditor, 146

UpdateCardNumber
 Brood::Application::DeckEditor, 146

UpdateCorrectCardNumber
 Brood::Application::TileEditor, 390

UpdateCurrCardAnswerInput
 Brood::Application::CardEditor, 79

UpdateCurrCardInfoDown
 Brood::Application::CardEditor, 79

UpdateCurrCardInfoTime
 Brood::Application::CardEditor, 80

UpdateCurrCardInfoUp
 Brood::Application::CardEditor, 80

UpdateCurrCardQuestionInput
 Brood::Application::CardEditor, 80

UpdateCurrPlayerFileDialog
 Brood::Application::PlayerEditor, 303

UpdateCurrPlayerOffsetX
 Brood::Application::PlayerEditor, 303

UpdateCurrPlayerOffsetY
 Brood::Application::PlayerEditor, 304

UpdateCurrPlayerSizeX
 Brood::Application::PlayerEditor, 304

UpdateCurrPlayerSizeY
 Brood::Application::PlayerEditor, 304

UpdateCurrSelectedCardIdx
 Brood::Application::CardEditor, 80

UpdateCurrSelectedDeckIdx
 Brood::Application::CardEditor, 81
 Brood::Application::DeckEditor, 147

UpdateCurrSelectedPlayerIdx
 Brood::Application::PlayerEditor, 304

UpdateDeclnPosX
 Brood::Application::WorkSpace, 433

UpdateDeclnPosY
 Brood::Application::WorkSpace, 434

UpdateDeclnSizeX
 Brood::Application::WorkSpace, 434

UpdateDeclnSizeY
 Brood::Application::WorkSpace, 435

UpdateDeckFileDialog
 Brood::Application::DeckEditor, 147

UpdateDeckNumber
 Brood::Application::GameEditor, 232

UpdateDeckPosX
 Brood::Application::DeckEditor, 147

UpdateDeckPosY
 Brood::Application::DeckEditor, 148

UpdateDeckSizeX
 Brood::Application::DeckEditor, 148

UpdateDeckSizeY
 Brood::Application::DeckEditor, 148

UpdateDiceFileDialog
 Brood::Application::DiceEditor, 172

UpdateDicePosX
 Brood::Application::DiceEditor, 172

UpdateDicePosY
 Brood::Application::DiceEditor, 172

UpdateDiceSideNum
 Brood::Application::DiceEditor, 173

UpdateDiceSizeX
 Brood::Application::DiceEditor, 173

UpdateDiceSizeY
 Brood::Application::DiceEditor, 173

UpdateDisplayedText
 Brood::Application::Components::DisplayCard, 190

UpdateEditModeTabs
 Brood::Application::EditorWorkspace, 212

UpdateEndBridgeColNumber
 Brood::Application::TileEditor, 391

UpdateEndBridgeRowNumber
 Brood::Application::TileEditor, 391

UpdateFileNameFeildDialog
 Brood::Application::HomeWorkSpace, 242

UpdateForceDiceRoll
 Brood::Application::TileEditor, 391

UpdateGameTitleInput
 Brood::Application::GameEditor, 232

UpdateHomeWorkspace
 Brood::Application::HomeWorkSpace, 243

UpdateIncorrectPenaltyPanelElement
 Brood::Application::GameEditor, 233

UpdateLargestUpperPos
 Brood::Application::Components::DisplayCard, 191

UpdateMaximumPlayerNumber
 Brood::Application::GameEditor, 233

UpdateMinimumPlayerNumber
 Brood::Application::GameEditor, 233

UpdateMouseButtonStatus
 Brood::MouseHandler, 275

UpdateMousePos
 Brood::MouseHandler, 275

UpdateMovementTypePanelElement
 Brood::Application::GameEditor, 233

UpdateNextPathColNumber
 Brood::Application::TileEditor, 391

UpdateNextPathRowNumber
 Brood::Application::TileEditor, 392

UpdatePathLines
 Brood::Application::Components::Path, 287

UpdatePathptr

Brood::Application::Components::Player, [297](#)
UpdatePosition
 Brood::Application::Components::Player, [297](#)
UpdateRibbionTabs
 Brood::Application::HomeWorkSpace, [243](#)
UpdateRollDice
 Brood::Application::DiceEditor, [173](#)
UpdateSettingSelectionDDI
 Brood::Application::CardEditor, [81](#)
UpdateSmallestLowerPos
 Brood::Application::Components::DisplayCard, [191](#)
UpdateStartColNumber
 Brood::Application::PlayerEditor, [305](#)
UpdateStartRowNumber
 Brood::Application::PlayerEditor, [305](#)
UpdateTile
 Brood::Application::Components::Tiles, [404](#)
UpdateTileColNumber
 Brood::Application::TileEditor, [392](#)
UpdateTileFileTexture
 Brood::Application::TileEditor, [392](#)
UpdateTileRowNumber
 Brood::Application::TileEditor, [392](#)
UpdateTileType
 Brood::Application::TileEditor, [393](#)
UpdateTurnCard
 Brood::Application::CardEditor, [81](#)

WIN32_LEAN_AND_MEAN
 stdafx.h, [526](#)
WorkSpace, [436](#)
 Brood::Application::WorkSpace, [424](#)
WriteOneLineToFile
 Brood::Application:: FileAccess, [221](#)