

Classroom Inquisition

1.7.3

Generated by Doxygen 1.7.5

Wed May 2 2012 09:07:32

Contents

1 Namespace Index	1
1.1 Packages	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Namespace Documentation	7
4.1 Package SrP_ClassroomInq	7
4.2 Package SrP_ClassroomInq.Properties	7
5 Class Documentation	9
5.1 SrP_ClassroomInq::AboutBox_CI Class Reference	9
5.1.1 Detailed Description	10
5.1.2 Constructor & Destructor Documentation	10
5.1.2.1 AboutBox_CI	10
5.1.3 Member Function Documentation	11
5.1.3.1 AboutBox_CI_Load	11
5.1.3.2 Dispose	11
5.1.3.3 InitializeComponent	11
5.1.4 Member Data Documentation	12
5.1.4.1 components	12
5.1.4.2 labelCopyright	12
5.1.4.3 labelProductName	12
5.1.4.4 labelVersion	12

5.1.4.5	logoPictureBox	12
5.1.4.6	Motto	12
5.1.4.7	okButton	13
5.1.4.8	tableLayoutPanel	13
5.1.4.9	textBoxDescription	13
5.2	SrP_ClassroomInq::Form1 Class Reference	14
5.2.1	Detailed Description	15
5.2.2	Constructor & Destructor Documentation	16
5.2.2.1	Form1	16
5.2.3	Member Function Documentation	16
5.2.3.1	btnCLR_Click	16
5.2.3.2	btnCLS_Click	16
5.2.3.3	Dispose	16
5.2.3.4	Form1_Load	17
5.2.3.5	InitializeComponent	17
5.2.3.6	lbl_question_Click	18
5.2.3.7	quitToolStripMenuItem_Click	18
5.2.3.8	replyToolStripMenuItem_Click	18
5.2.3.9	textBox1_MouseClick	18
5.2.3.10	timer_Tick	18
5.2.3.11	trayICON_MouseDoubleClick	18
5.2.4	Member Data Documentation	18
5.2.4.1	broadcastToolStripMenuItem	18
5.2.4.2	btnCLR	18
5.2.4.3	btnCLS	19
5.2.4.4	btnCLS_WASclicked	19
5.2.4.5	btnRPL	19
5.2.4.6	btnSend	19
5.2.4.7	components	19
5.2.4.8	controlsToolStripMenuItem	19
5.2.4.9	directMessageToolStripMenuItem	19
5.2.4.10	grpbx_Reply	19
5.2.4.11	grpbxFeed	20
5.2.4.12	grpbxRPL_WASclicked	20

5.2.4.13	helpToolStripMenuItem	20
5.2.4.14	i	20
5.2.4.15	j	20
5.2.4.16	k	20
5.2.4.17	lbl_question	20
5.2.4.18	menuStrip1	20
5.2.4.19	quitToolStripMenuItem	20
5.2.4.20	replyToolStripMenuItem	21
5.2.4.21	sb_send_status	21
5.2.4.22	sendToolStripMenuItem	21
5.2.4.23	statusStrip1	21
5.2.4.24	textBox1	21
5.2.4.25	textbox1WASclicked	21
5.2.4.26	timer	21
5.2.4.27	trayICON	21
5.2.4.28	txtbx_Reply	21
5.2.4.29	viewToolStripMenuItem	22
5.2.4.30	vScrollBar1	22
5.3	SrP_ClassroomInq::frmClassromInq Class Reference	23
5.3.1	Detailed Description	33
5.3.2	Constructor & Destructor Documentation	33
5.3.2.1	frmClassromInq	33
5.3.3	Member Function Documentation	33
5.3.3.1	aboutToolStripMenuItem_Click	33
5.3.3.2	attendanceToolStripMenuItem_Click	33
5.3.3.3	broadcastToolStripMenuItem_Click	34
5.3.3.4	btnAddQM_Click	34
5.3.3.5	btnBkgrndColor_Click	34
5.3.3.6	btnCLR_Click	35
5.3.3.7	btnCLRStudents_Click	35
5.3.3.8	btnCLS_Click	35
5.3.3.9	btnCLS_Click_BrdCst	35
5.3.3.10	btnCLS_ConvView_Click	36
5.3.3.11	btnCLSStudents_Click	36

5.3.3.12	btnConvView_Click	36
5.3.3.13	btnCV_Print_Click	36
5.3.3.14	btnCV_Refresh_Click	37
5.3.3.15	btnCVReset_Click	37
5.3.3.16	btnDefaultTheme_Click	37
5.3.3.17	btnDM_Clr_Click	37
5.3.3.18	btnDM_Cls_Click	37
5.3.3.19	btnDM_Send_Click	37
5.3.3.20	btnDoneStudents_Click	38
5.3.3.21	btnEDTStudents_Click	38
5.3.3.22	btnExitAttendance_Click	38
5.3.3.23	btnExitClassVote_Click	38
5.3.3.24	btnExitQuiz_Click	39
5.3.3.25	btnFAQcls_Click	39
5.3.3.26	btnForeColor_Click	39
5.3.3.27	btnPrefs_Cls_Click	39
5.3.3.28	btnQMclr_Click	40
5.3.3.29	btnQMcls_Click	40
5.3.3.30	btnQMDel_MouseDown	40
5.3.3.31	btnQMEdit_Click	40
5.3.3.32	btnQuizAsk_Click	40
5.3.3.33	btnQuizMaker_Click	41
5.3.3.34	btnQuizMode_Click	41
5.3.3.35	btnRepl_Click	41
5.3.3.36	btnSend_Click	42
5.3.3.37	btnStuMgmt_Prefs_Click	42
5.3.3.38	CheckKeys	42
5.3.3.39	chkbxCtrlHide_CheckedChanged	43
5.3.3.40	ClassVoteHandler	43
5.3.3.41	classVoteToolStripMenuItem_Click	43
5.3.3.42	cmbbxAnimationSpeed_SelectedIndexChanged	44
5.3.3.43	cmbxDM_Click	44
5.3.3.44	cmbxDM_SelectedIndexChanged	44
5.3.3.45	cntxtMenu_arr_Popup	44

5.3.3.46	ConvViewPic_Click	44
5.3.3.47	DecryptFile	45
5.3.3.48	directMessageToolStripMenuItem_Click	45
5.3.3.49	Dispose	45
5.3.3.50	EncryptFile	45
5.3.3.51	FlowCtrl_Send	46
5.3.3.52	Form1_FormClosing	46
5.3.3.53	Form1_Load	46
5.3.3.54	generalFAQToolStripMenuItem_Click	46
5.3.3.55	GenerateKey	47
5.3.3.56	group_arr_MouseDown	47
5.3.3.57	grpbxFeed_MouseDown	47
5.3.3.58	HandleBCKSPC	47
5.3.3.59	InitializeComponent	48
5.3.3.60	lbl_question_MouseDown	48
5.3.3.61	lbl_question_MouseMove	48
5.3.3.62	lblNotify_MouseDown	49
5.3.3.63	InklblFAQ_LinkClicked	49
5.3.3.64	InklblIssues_LinkClicked	49
5.3.3.65	LogQandA	49
5.3.3.66	IstbxConvView_MouseLeave	50
5.3.3.67	MakeCtrls	50
5.3.3.68	MakeQuizQuestion	50
5.3.3.69	MoveCtrlsDown	51
5.3.3.70	multmAttendance_Click	51
5.3.3.71	multmBrdCst_Click	51
5.3.3.72	multmClassVote_Click	52
5.3.3.73	multmDelete_Click	52
5.3.3.74	multmDM_Click	52
5.3.3.75	multmFAQ_Click	52
5.3.3.76	multmName_Click	52
5.3.3.77	multmPrefs_Click	52
5.3.3.78	multmQuit_Click	52
5.3.3.79	multmQuiz_Click	53

5.3.3.80	multmQuizDelete_Click	53
5.3.3.81	multmUndo_Click	53
5.3.3.82	picbx_CV_MouseEnter	53
5.3.3.83	picbx_CV_MouseLeave	53
5.3.3.84	pictureBoxNotify_Click	54
5.3.3.85	pnlBrdCst_Leave	54
5.3.3.86	PrefsToolStripMenuItem_Click	54
5.3.3.87	Printer_PrintPage	54
5.3.3.88	quitToolStripMenuItem_Click	55
5.3.3.89	quizToolStripMenuItem_Click	55
5.3.3.90	RecordQuizData	55
5.3.3.91	SavePrefs	55
5.3.3.92	SaveStudentData	56
5.3.3.93	SendMsg	56
5.3.3.94	serialCOMcmbbx_Click	56
5.3.3.95	serialCOMcmbbx_SelectedIndexChanged	57
5.3.3.96	showNamesToolStripMenuItem_Click	57
5.3.3.97	ThemeApply	57
5.3.3.98	timer_ControlsCreate_Tick	58
5.3.3.99	timer_SerialRead_Tick	58
5.3.3.100	timer_Tick	58
5.3.3.101	trayICON_MouseDoubleClick	59
5.3.3.102	txtbxQM_Click	59
5.3.3.103	UnreadDecrement	59
5.3.3.104	ZeroMemory	59
5.3.4	Member Data Documentation	59
5.3.4.1	About	59
5.3.4.2	aboutToolStripMenuItem	59
5.3.4.3	aboutToolStripMenuItem1	59
5.3.4.4	AckRXd	60
5.3.4.5	addr	60
5.3.4.6	addr_tbl	60
5.3.4.7	attedanceToolStripMenuItem	60
5.3.4.8	AttendanceClicked	60

5.3.4.9	AttendanceShowing	60
5.3.4.10	AttendanceTimesClicked	60
5.3.4.11	BackColorTheme	60
5.3.4.12	brdcst_addr	61
5.3.4.13	BrdcstShowing	61
5.3.4.14	broadcastToolStripMenuItem	61
5.3.4.15	btnAddQM	61
5.3.4.16	btnBkgrndColor	61
5.3.4.17	btnCLRStudents	61
5.3.4.18	btnCLS_ConvView	61
5.3.4.19	btnCLS_WASclicked	61
5.3.4.20	btnCLSStudents	62
5.3.4.21	btnConvView	62
5.3.4.22	btnCV_Print	62
5.3.4.23	btnCV_Refresh	62
5.3.4.24	btnCVReset	62
5.3.4.25	btnDefaultTheme	62
5.3.4.26	btnDM_Clr	62
5.3.4.27	btnDM_Cls	62
5.3.4.28	btnDM_Send	62
5.3.4.29	btnDoneStudents	63
5.3.4.30	btnEDTStudents	63
5.3.4.31	btnExitAttendance	63
5.3.4.32	btnExitClassVote	63
5.3.4.33	btnExitQuiz	63
5.3.4.34	btnFAQcls	63
5.3.4.35	btnForeColor	63
5.3.4.36	btnPrefs_Cls	63
5.3.4.37	btnQMclr	63
5.3.4.38	btnQMcls	64
5.3.4.39	btnQMDel	64
5.3.4.40	btnQMEdit	64
5.3.4.41	btnQuizMaker	64
5.3.4.42	btnQuizMode	64

5.3.4.43	btnQuizQuestionSend_arr	64
5.3.4.44	btnQuizQuestionSend_arr_tmp	64
5.3.4.45	btnSend	64
5.3.4.46	btnStuMgmt_Prefs	65
5.3.4.47	chkbxCtrlHide	65
5.3.4.48	chkbxLameMode	65
5.3.4.49	chkbxNotify	65
5.3.4.50	chkbxRXSound	65
5.3.4.51	chkbxToolips	65
5.3.4.52	chkbxTXSound	65
5.3.4.53	classSize	65
5.3.4.54	ClassVoteClicked	65
5.3.4.55	ClassVoteShowing	66
5.3.4.56	ClassVoteTimesClicked	66
5.3.4.57	classVoteToolStripMenuItem	66
5.3.4.58	clear_arr	66
5.3.4.59	close_arr	66
5.3.4.60	cmbbxAnimationSpeed	66
5.3.4.61	cmbxDM	66
5.3.4.62	cntxtMenu	66
5.3.4.63	colorDlg	67
5.3.4.64	components	67
5.3.4.65	contentsToolStripMenuItem	67
5.3.4.66	ConvViewClicked	67
5.3.4.67	ConvViewShowing	67
5.3.4.68	ConvViewTimesClicked	67
5.3.4.69	copyToolStripMenuItem	67
5.3.4.70	customizeToolStripMenuItem	67
5.3.4.71	cutToolStripMenuItem	68
5.3.4.72	Del_ID	68
5.3.4.73	DeleteQuestion	68
5.3.4.74	DesireBrdcst	68
5.3.4.75	DesireDM	68
5.3.4.76	DesireID	68

5.3.4.77	DesireNotify	68
5.3.4.78	DesirePrefs	68
5.3.4.79	directMessageToolStripMenuItem	68
5.3.4.80	DirectMsgPanel	69
5.3.4.81	DMclicked	69
5.3.4.82	DMPanelShowing	69
5.3.4.83	DMtimesClicked	69
5.3.4.84	editToolStripMenuItem	69
5.3.4.85	EDTClicked	69
5.3.4.86	EncryptedData	69
5.3.4.87	exitToolStripMenuItem	69
5.3.4.88	FAQClicked	70
5.3.4.89	FAQShowing	70
5.3.4.90	FAQTimesClicked	70
5.3.4.91	fileToolStripMenuItem	70
5.3.4.92	ForeColorTheme	70
5.3.4.93	generalFAQToolStripMenuItem	70
5.3.4.94	group_arr	70
5.3.4.95	grpbxEDITStudents	70
5.3.4.96	grpbxFeed	71
5.3.4.97	grpbxRPL_WASclicked	71
5.3.4.98	grpbxTheme	71
5.3.4.99	grpbxUnread	71
5.3.4.100	helpToolStripMenuItem	71
5.3.4.101	helpToolStripMenuItem1	71
5.3.4.102	i	71
5.3.4.103	indexToolStripMenuItem	71
5.3.4.104	ix	72
5.3.4.105	j	72
5.3.4.106	JukeBox	72
5.3.4.107	jx	72
5.3.4.108	k	72
5.3.4.109	lbl_arr	72
5.3.4.110	lbl_arr_temp	72

5.3.4.111 lbl_ID	72
5.3.4.112 lbl_ID_2	72
5.3.4.113 lblAnimationPrefs	73
5.3.4.114 lblCVLeft	73
5.3.4.115 lblCVPipe	73
5.3.4.116 lblCVRRight	73
5.3.4.117 lblFAQ	73
5.3.4.118 lblNotify	73
5.3.4.119 lblNotifyTime	73
5.3.4.120 lblOption1	73
5.3.4.121 lblOption2	73
5.3.4.122 lblQuestionNum_arr	74
5.3.4.123 lblQuestionNum_arr_tmp	74
5.3.4.124 lblSerPt	74
5.3.4.125 lblStudentsName	74
5.3.4.126 lnklblIssues	74
5.3.4.127 logFiles	74
5.3.4.128 logHistoryLength	74
5.3.4.129 logTmpString	74
5.3.4.130 lstbxAttendance	75
5.3.4.131 lstbxConvView	75
5.3.4.132 lstbxQuizMaker	75
5.3.4.133 lstbxStudents	75
5.3.4.134 menuStrip1	75
5.3.4.135 modesToolStripMenuItem	75
5.3.4.136 multmAttendance	75
5.3.4.137 multmBrdCst	75
5.3.4.138 multmClassVote	76
5.3.4.139 multmDelete	76
5.3.4.140 multmDM	76
5.3.4.141 multmFAQ	76
5.3.4.142 multmName	76
5.3.4.143 multmPrefs	76
5.3.4.144 multmQuit	76

5.3.4.145 multmQuiz	76
5.3.4.146 multmQuizDelete	76
5.3.4.147 multmUndo	77
5.3.4.148 NEW_grpbx	77
5.3.4.149 new_lblID	77
5.3.4.150 newToolStripMenuItem	77
5.3.4.151 normalToolStripMenuItem	77
5.3.4.152 notif_I	77
5.3.4.153 NotifyShowing	77
5.3.4.154 NumQuestions	77
5.3.4.155 numUpDnNotify	77
5.3.4.156 old_lblID	78
5.3.4.157 openToolStripMenuItem	78
5.3.4.158 optionsToolStripMenuItem	78
5.3.4.159 origingrouparr	78
5.3.4.160 PanelAttendance	78
5.3.4.161 PanelClassVote	78
5.3.4.162 PanelConvView	78
5.3.4.163 PanelFAQ	78
5.3.4.164 PanelNotify	79
5.3.4.165 PanelPrefs	79
5.3.4.166 PanelQuestion_arr	79
5.3.4.167 PanelQuestion_arr_tmp	79
5.3.4.168 PanelQuizMaker	79
5.3.4.169 PanelQuizMode	79
5.3.4.170 PanelStudents	79
5.3.4.171 pasteToolStripMenuItem	79
5.3.4.172 picbx_arr	80
5.3.4.173 picbx_arr_tmp	80
5.3.4.174 picbx_ConvView_arr	80
5.3.4.175 picbx_ConvView_tmp	80
5.3.4.176 picbxPrefs	80
5.3.4.177 pictureBox1	80
5.3.4.178 pictureBox2	80

5.3.4.179 pictureBox3	80
5.3.4.180 pictureBox4	81
5.3.4.181 pictureBox5	81
5.3.4.182 pictureBox6	81
5.3.4.183 pictureBox7	81
5.3.4.184 pictureBox8	81
5.3.4.185 pictureBoxNotify	81
5.3.4.186 PlainData	81
5.3.4.187 pnIBrdCst	81
5.3.4.188 pnQuiz	81
5.3.4.189 portNames	82
5.3.4.190 PrefsClicked	82
5.3.4.191 PrefsShowing	82
5.3.4.192 PrefsTimesClicked	82
5.3.4.193 PrefsToolStripMenuItem	82
5.3.4.194 PresentStuNum	82
5.3.4.195 Printer	82
5.3.4.196 PrintFont	82
5.3.4.197 printPreviewToolStripMenuItem	83
5.3.4.198 printToolStripMenuItem	83
5.3.4.199 Q_NameShowing	83
5.3.4.200 Q_Replied	83
5.3.4.201 Q_sender	83
5.3.4.202 Q_status	83
5.3.4.203 Qs_to_create	83
5.3.4.204 Qs_to_Make	83
5.3.4.205 Question_Deleted	84
5.3.4.206 quitToolStripMenuItem	84
5.3.4.207 QuizDelID	84
5.3.4.208 QuizMakerClicked	84
5.3.4.209 QuizMakerShowing	84
5.3.4.210 QuizMakerTimesClicked	84
5.3.4.211 QuizModeClicked	84
5.3.4.212 QuizModeShowing	84

5.3.4.213 QuizModeTimesClicked	84
5.3.4.214 QuizQuestionDel	85
5.3.4.215 quizToolStripMenuItem	85
5.3.4.216 QzQuestionDel	85
5.3.4.217 rdbtnClick	85
5.3.4.218 rdbtnHover	85
5.3.4.219 redoToolStripMenuItem	85
5.3.4.220 reply_arr	85
5.3.4.221 Request_Undo	85
5.3.4.222 RX_Data	86
5.3.4.223 saveAsToolStripMenuItem	86
5.3.4.224 saveToolStripMenuItem	86
5.3.4.225 searchToolStripMenuItem	86
5.3.4.226 SecretKey	86
5.3.4.227 selectAllToolStripMenuItem	86
5.3.4.228 sendToolStripMenuItem	86
5.3.4.229 serialCOMcmbbx	86
5.3.4.230 SerialPort	87
5.3.4.231 showNamesToolStripMenuItem	87
5.3.4.232 StateMachine	87
5.3.4.233 Stream2Print	87
5.3.4.234 StudentPresent	87
5.3.4.235 Students_ID	87
5.3.4.236 Students_Name	87
5.3.4.237 StudentVoted	87
5.3.4.238 StuMgmtClicked	87
5.3.4.239 StuMgmtShowing	88
5.3.4.240 StuMgmtTimesClicked	88
5.3.4.241 sudo_kill	88
5.3.4.242 tempcleararr	88
5.3.4.243 tempclosearr	88
5.3.4.244 tempgrouparr	88
5.3.4.245 tempreplyarr	88
5.3.4.246 tempString	88

5.3.4.247 tempString2	88
5.3.4.248 textBox1	89
5.3.4.249 textBox1WASclicked	89
5.3.4.250 timeout_ack	89
5.3.4.251 timer	89
5.3.4.252 timer_ConditionsCreate	89
5.3.4.253 timer_SerialRead	89
5.3.4.254 timesClicked	89
5.3.4.255 timesThrough	89
5.3.4.256 toolsToolStripMenuItem	90
5.3.4.257 toolStripComboBox1	90
5.3.4.258 toolStripSeparator	90
5.3.4.259 toolStripSeparator1	90
5.3.4.260 toolStripSeparator2	90
5.3.4.261 toolStripSeparator3	90
5.3.4.262 toolStripSeparator4	90
5.3.4.263 toolStripSeparators5	90
5.3.4.264 trayICON	91
5.3.4.265 tt_picbxCV_arr	91
5.3.4.266 txtbx_reply_arr	91
5.3.4.267 txtbx_reply_temp	91
5.3.4.268 txtbxCVStats	91
5.3.4.269 txtbxDM	91
5.3.4.270 txtbxFAQ	91
5.3.4.271 txtbxQM	91
5.3.4.272 txtbxQuestion_arr	92
5.3.4.273 txtbxQuestion_arr_tmp	92
5.3.4.274 txtbxStudentsName	92
5.3.4.275 undoToolStripMenuItem	92
5.3.4.276 UnreadCount	92
5.3.4.277 viewToolStripMenuItem	92
5.3.4.278 WeGotData	92
5.3.4.279 x	92
5.4 SrP_ClassroomInq::Program Class Reference	93

5.4.1	Member Function Documentation	93
5.4.1.1	Main	93
5.5	SrP_ClassroomInq::Properties::Resources Class Reference	93
5.5.1	Detailed Description	94
5.5.2	Constructor & Destructor Documentation	94
5.5.2.1	Resources	94
5.5.3	Member Data Documentation	94
5.5.3.1	resourceCulture	94
5.5.3.2	resourceMan	95
5.5.4	Property Documentation	95
5.5.4.1	Background	95
5.5.4.2	bubbles2	95
5.5.4.3	comment_bubble	95
5.5.4.4	Culture	95
5.5.4.5	reply_arrow	95
5.5.4.6	ResourceManager	95
5.5.4.7	StuMngmntCtr	96
5.5.4.8	user_comment	96
5.6	SrP_ClassroomInq::Properties::Settings Class Reference	96
5.6.1	Member Data Documentation	97
5.6.1.1	defaultInstance	97
5.6.2	Property Documentation	97
5.6.2.1	Animations	97
5.6.2.2	AnimationSpeed	97
5.6.2.3	BackColorTheme	97
5.6.2.4	ClickRead	97
5.6.2.5	CtrlHide	97
5.6.2.6	Default	97
5.6.2.7	ForeColorTheme	98
5.6.2.8	HoverRead	98
5.6.2.9	key	98
5.6.2.10	Notify	98
5.6.2.11	NotifyTimeout	98
5.6.2.12	ShowNames	98

5.6.2.13	SoundRX	98
5.6.2.14	SoundTX	98
5.6.2.15	Tooltips	98
6	File Documentation	99
6.1	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AboutBox1.cs File Reference	99
6.2	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AboutBox1.cs	99
6.3	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AboutBox1.Designer.cs File Reference	100
6.4	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AboutBox1.Designer.cs	100
6.5	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AnimateTimer.cs File Reference	103
6.6	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/AnimateTimer.cs	103
6.7	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1-dyn-ctrls.cs File Reference	132
6.8	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1-dyn-ctrls.cs	132
6.9	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1-dyn-ctrls.Designer.cs File Reference	175
6.10	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1-dyn-ctrls.Designer.cs	175
6.11	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1.cs File Reference	207
6.12	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1.cs	208
6.13	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1.Designer.cs File Reference	210
6.14	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Form1.Designer.cs	211
6.15	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Program.cs File Reference	216
6.16	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Program.cs	216
6.17	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/AssemblyInfo.cs File Reference	216

6.18	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/AssemblyInfo.cs	216
6.19	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/Resources.Designer.cs File Reference . .	217
6.20	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/Resources.Designer.cs	217
6.21	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/Settings.Designer.cs File Reference . .	219
6.22	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/Properties/Settings.Designer.cs	219
6.23	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/themes.cs File Reference	222
6.24	/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/- SrP_ClassroomInq/themes.cs	222

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

SrP_ClassroomInq	7
SrP_ClassroomInq.Properties	7

Chapter 2

Class Index

2.1 Class List

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

SrP_ClassroomInq::AboutBox_CI	9
SrP_ClassroomInq::Form1	14
SrP_ClassroomInq::frmClassromInq	23
SrP_ClassroomInq::Program	93
SrP_ClassroomInq::Properties::Resources A strongly-typed resource class, for looking up localized strings, etc	93
SrP_ClassroomInq::Properties::Settings	96
SrP_ClassroomInq::Splash	??

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ AboutBox1.cs	99
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ AboutBox1.Designer.cs	100
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ AnimateTimer.cs	103
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Form1-dyn-ctrls.cs	132
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Form1-dyn-ctrls.Designer.cs	175
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Form1.cs	207
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Form1.Designer.cs	210
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Program.cs	216
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Splash.cs	??
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Splash.Designer.cs	??
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ themes.cs	222
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Properties/AssemblyInfo.cs	216
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Properties/Resources.Designer.cs	217
/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/Sr-	
P_ClassroomInq/ Properties/Settings.Designer.cs	219

Chapter 4

Namespace Documentation

4.1 Package SrP_ClassroomInq

Packages

- package [Properties](#)

Classes

- class [AboutBox_CI](#)
- class [frmClassroomInq](#)
- class [Form1](#)
- class [Program](#)
- class [Splash](#)

4.2 Package SrP_ClassroomInq.Properties

Classes

- class [Resources](#)
A strongly-typed resource class, for looking up localized strings, etc.
- class [Settings](#)

Chapter 5

Class Documentation

5.1 SrP_ClassroomInq::AboutBox_CI Class Reference

Public Member Functions

- [AboutBox_CI \(\)](#)

Private Member Functions

- void [AboutBox_CI_Load \(object sender, EventArgs e\)](#)

5.1.1 Constructor & Destructor Documentation

5.1.1.1 SrP_ClassroomInq::AboutBox_CI::AboutBox_CI ()

Definition at line 30 of file [AboutBox1.cs](#).

5.1.2 Member Function Documentation

5.1.2.1 void SrP_ClassroomInq::AboutBox_CI::AboutBox_CI_Load (object sender, EventArgs e) [private]

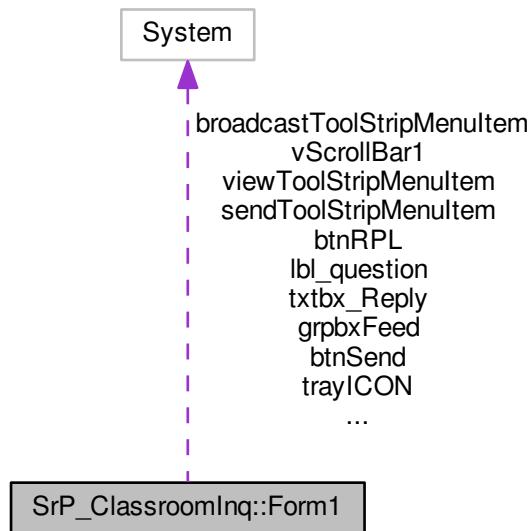
Definition at line 35 of file [AboutBox1.cs](#).

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

- [/home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_- ClassroomInq/AboutBox1.cs](#)

5.2 SrP_ClassroomInq::Form1 Class Reference

Collaboration diagram for SrP_ClassroomInq::Form1:



Public Member Functions

- [Form1 \(\)](#)

Protected Member Functions

- override void [Dispose \(bool disposing\)](#)

Clean up any resources being used.

Private Member Functions

- void [quitToolStripMenuItem_Click \(object sender, EventArgs e\)](#)
- void [textBox1_MouseClick \(object sender, MouseEventArgs e\)](#)
- void [trayICON_MouseDoubleClick \(object sender, MouseEventArgs e\)](#)
- void [timer_Tick \(object sender, EventArgs e\)](#)
- void [btnCLR_Click \(object sender, EventArgs e\)](#)
- void [lbl_question_Click \(object sender, EventArgs e\)](#)

- void [replyToolStripMenuItem_Click](#) (object sender, EventArgs e)
- void [btnCLS_Click](#) (object sender, EventArgs e)
- void [Form1_Load](#) (object sender, EventArgs e)
- void [InitializeComponent](#) ()

Required method for Designer support - do not modify the contents of this method with the code editor.

Private Attributes

- byte [i](#) = 0
 - byte [k](#) = 0
 - byte [j](#) = 0
 - bool [textBox1WASclicked](#) = new bool()
 - bool [grpbxRPL_WASclicked](#) = new bool()
 - bool [btnCLS_WASclicked](#) = new bool()
 - System.ComponentModel.IContainer [components](#) = null
- Required designer variable.*
- System.Windows.Forms.MenuStrip [menuStrip1](#)
 - System.Windows.Forms.ToolStripItem [sendToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [broadcastToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [directMessageToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [replyToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [viewToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [helpToolStripMenuItem](#)
 - System.Windows.Forms.TextBox [textBox1](#)
 - System.Windows.Forms.StatusStrip [statusStrip1](#)
 - System.Windows.Forms.ToolStripStatusLabel [sb_send_status](#)
 - System.Windows.Forms.ToolStripItem [controlsToolStripMenuItem](#)
 - System.Windows.Forms.ToolStripItem [quitToolStripMenuItem](#)
 - System.Windows.Forms.Timer [timer](#)
 - System.Windows.Forms.Button [btnSend](#)
 - System.Windows.Forms.NotifyIcon [trayICON](#)
 - System.Windows.Forms.GroupBox [grpbxFeed](#)
 - System.Windows.Forms.GroupBox [grpbx_Reply](#)
 - System.Windows.Forms.Button [btnCLR](#)
 - System.Windows.Forms.TextBox [txtbx_Reply](#)
 - System.Windows.Forms.Label [lbl_question](#)
 - System.Windows.Forms.Button [btnRPL](#)
 - System.Windows.Forms.Button [btnCLS](#)
 - System.Windows.Forms.VScrollBar [vScrollBar1](#)

5.2.1 Detailed Description

Definition at line 3 of file [Form1.Designer.cs](#).

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `SrP_ClassroomInq::Form1::Form1()`

Definition at line 14 of file [Form1.cs](#).

Here is the call graph for this function:



5.2.3 Member Function Documentation

5.2.3.1 `void SrP_ClassroomInq::Form1::btnCLR_Click (object sender, EventArgs e)`
[private]

Definition at line 142 of file [Form1.cs](#).

5.2.3.2 `void SrP_ClassroomInq::Form1::btnCLS_Click (object sender, EventArgs e)`
[private]

Definition at line 159 of file [Form1.cs](#).

5.2.3.3 `override void SrP_ClassroomInq::Form1::Dispose (bool disposing)`
[protected]

Clean up any resources being used.

Parameters

<code>disposing</code>	true if managed resources should be disposed; otherwise, false.
------------------------	-----------------------------------------------------------------

Definition at line 14 of file [Form1.Designer.cs](#).

5.2.3.4 `void SrP_ClassroomInq::Form1::Form1_Load (object sender, EventArgs e)`
[private]

Definition at line 165 of file [Form1.cs](#).

5.2.3.5 void SrP_ClassroomInq::Form1::InitializeComponent() [private]

Required method for Designer support - do not modify the contents of this method with the code editor.

Definition at line 29 of file [Form1.Designer.cs](#).

Here is the caller graph for this function:



5.2.3.6 void SrP_ClassroomInq::Form1::lbl_question_Click (object sender, EventArgs e) [private]

Definition at line 147 of file [Form1.cs](#).

5.2.3.7 void SrP_ClassroomInq::Form1::quitToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 27 of file [Form1.cs](#).

5.2.3.8 void SrP_ClassroomInq::Form1::replyToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 153 of file [Form1.cs](#).

5.2.3.9 void SrP_ClassroomInq::Form1::textBox1_MouseClick (object sender, MouseEventArgs e) [private]

Definition at line 43 of file [Form1.cs](#).

5.2.3.10 void SrP_ClassroomInq::Form1::timer_Tick (object sender, EventArgs e) [private]

Definition at line 68 of file [Form1.cs](#).

```
5.2.3.11 void SrP_ClassroomInq::Form1::trayICON_MouseDoubleClick ( object sender,
    MouseEventArgs e ) [private]
```

Definition at line [50](#) of file [Form1.cs](#).

5.2.4 Member Data Documentation

```
5.2.4.1 System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1-
    ::broadcastToolStripMenuItem [private]
```

Definition at line [300](#) of file [Form1.Designer.cs](#).

```
5.2.4.2 System.Windows.Forms.Button SrP_ClassroomInq::Form1::btnCLR
    [private]
```

Definition at line [315](#) of file [Form1.Designer.cs](#).

```
5.2.4.3 System.Windows.Forms.Button SrP_ClassroomInq::Form1::btnCLS
    [private]
```

Definition at line [319](#) of file [Form1.Designer.cs](#).

```
5.2.4.4 bool SrP_ClassroomInq::Form1::btnCLS_WASclicked = new bool()
    [private]
```

Definition at line [23](#) of file [Form1.cs](#).

```
5.2.4.5 System.Windows.Forms.Button SrP_ClassroomInq::Form1::btnRPL
    [private]
```

Definition at line [318](#) of file [Form1.Designer.cs](#).

```
5.2.4.6 System.Windows.Forms.Button SrP_ClassroomInq::Form1::btnSend
    [private]
```

Definition at line [311](#) of file [Form1.Designer.cs](#).

```
5.2.4.7 System.ComponentModel.IContainer SrP_ClassroomInq::Form1::components =
    null [private]
```

Required designer variable.

Definition at line [8](#) of file [Form1.Designer.cs](#).

5.2.4.8 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1-::controlsToolStripMenuItem** [private]

Definition at line 308 of file [Form1.Designer.cs](#).

5.2.4.9 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::direct-MessageToolStripMenuItem** [private]

Definition at line 301 of file [Form1.Designer.cs](#).

5.2.4.10 **System.Windows.Forms.GroupBox SrP_ClassroomInq::Form1::grpbx_Reply** [private]

Definition at line 314 of file [Form1.Designer.cs](#).

5.2.4.11 **System.Windows.Forms.GroupBox SrP_ClassroomInq::Form1::grpbxFeed** [private]

Definition at line 313 of file [Form1.Designer.cs](#).

5.2.4.12 **bool SrP_ClassroomInq::Form1::grpbxRPL_WASclicked = new bool()** [private]

Definition at line 22 of file [Form1.cs](#).

5.2.4.13 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::help-ToolStripMenuItem** [private]

Definition at line 304 of file [Form1.Designer.cs](#).

5.2.4.14 **byte SrP_ClassroomInq::Form1::i = 0** [private]

Definition at line 18 of file [Form1.cs](#).

5.2.4.15 **byte SrP_ClassroomInq::Form1::j = 0** [private]

Definition at line 20 of file [Form1.cs](#).

5.2.4.16 **byte SrP_ClassroomInq::Form1::k = 0** [private]

Definition at line 19 of file [Form1.cs](#).

5.2.4.17 **System.Windows.Forms.Label SrP_ClassroomInq::Form1::lbl_question**
[private]

Definition at line 317 of file [Form1.Designer.cs](#).

5.2.4.18 **System.Windows.Forms.MenuStrip SrP_ClassroomInq::Form1::menuStrip1**
[private]

Definition at line 298 of file [Form1.Designer.cs](#).

5.2.4.19 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::quit-ToolStripMenuItem** [private]

Definition at line 309 of file [Form1.Designer.cs](#).

5.2.4.20 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::reply-ToolStripMenuItem** [private]

Definition at line 302 of file [Form1.Designer.cs](#).

5.2.4.21 **System.Windows.Forms.ToolStripStatusLabel SrP_ClassroomInq::Form1::sb_send_status** [private]

Definition at line 307 of file [Form1.Designer.cs](#).

5.2.4.22 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::send-ToolStripMenuItem** [private]

Definition at line 299 of file [Form1.Designer.cs](#).

5.2.4.23 **System.Windows.Forms.StatusStrip SrP_ClassroomInq::Form1::statusStrip1**
[private]

Definition at line 306 of file [Form1.Designer.cs](#).

5.2.4.24 **System.Windows.Forms.TextBox SrP_ClassroomInq::Form1::textBox1**
[private]

Definition at line 305 of file [Form1.Designer.cs](#).

5.2.4.25 **bool SrP_ClassroomInq::Form1::textbox1WASclicked = new bool()**
[private]

Definition at line 21 of file [Form1.cs](#).

5.2.4.26 **System.Windows.Forms.Timer SrP_ClassroomInq::Form1::timer**
[private]

Definition at line 310 of file [Form1.Designer.cs](#).

5.2.4.27 **System.Windows.Forms.NotifyIcon SrP_ClassroomInq::Form1::trayICON**
[private]

Definition at line 312 of file [Form1.Designer.cs](#).

5.2.4.28 **System.Windows.Forms.TextBox SrP_ClassroomInq::Form1::txtbx_Reply**
[private]

Definition at line 316 of file [Form1.Designer.cs](#).

5.2.4.29 **System.Windows.Forms.ToolStripItem SrP_ClassroomInq::Form1::view-ToolStripMenuItem** [private]

Definition at line 303 of file [Form1.Designer.cs](#).

5.2.4.30 **System.Windows.Forms.VScrollBar SrP_ClassroomInq::Form1::vScrollBar1**
[private]

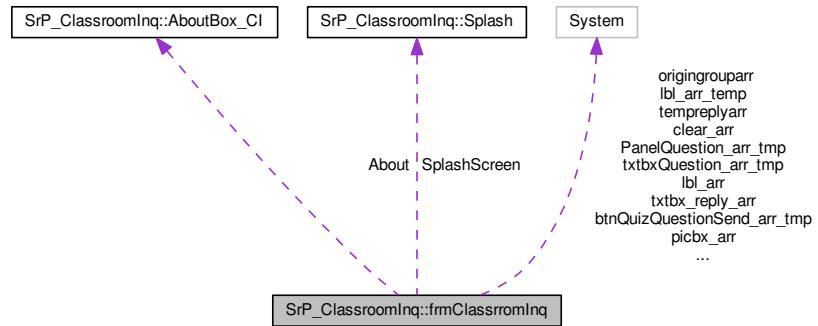
Definition at line 320 of file [Form1.Designer.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-ClassroomInq/[Form1.cs](#)
- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-ClassroomInq/[Form1.Designer.cs](#)

5.3 SrP_ClassroomInq::frmClassromInq Class Reference

Collaboration diagram for SrP_ClassroomInq::frmClassromInq:



Public Member Functions

- `frmClassromInq ()`
- `void MoveCtrlsDown ()`
- `void MakeCtrls (string question)`
- `string HandleBCKSPC (string RAW_DATA)`
- `void CheckKeys (object sender, System.Windows.Forms.KeyPressEventArgs e)`
- `void SaveStudentData ()`
- `static bool ZeroMemory (IntPtr Destination, int Length)`
- `bool SendMsg (string Message2Send, string Address2Send)`
- `void SavePrefs ()`
- `void UnreadDecrement (object sender)`
- `bool LogQandA (string dialog, bool teacher, string recipient)`
- `bool MakeQuizQuestion (string question, int number)`
- `void RecordQuizData (string question, string name_sender)`
- `void ThemeApply (Color Fore, Color Back)`

Public Attributes

- `const int classSize = 50`
- `const int MaxQuestions = 250`
- `StreamReader Stream2Print`
- `Font PrintFont`
- `Color ForeColorTheme = Color.DarkGoldenrod`
- `Color BackColorTheme = Color.Black`
- `AboutBox_CI About = new AboutBox_CI()`
- `Splash SplashScreen = new Splash()`

Private Member Functions

- void `timer_Tick` (object sender, EventArgs e)
- void `quitToolStripMenuItem_Click` (object sender, EventArgs e)
- void `trayICON_MouseDoubleClick` (object sender, MouseEventArgs e)
- void `btnCLR_Click` (object sender, EventArgs e)
- void `btnRepl_Click` (object sender, EventArgs e)
- void `lbl_question_MouseDown` (object sender, MouseEventArgs e)
- void `btnCLS_Click` (object sender, EventArgs e)
- void `Form1_Load` (object sender, EventArgs e)
- void `PrefsToolStripMenuItem_Click` (object sender, EventArgs e)
- void `serialCOMcmbbx_Click` (object sender, EventArgs e)
- void `serialCOMcmbbx_SelectedIndexChanged` (object sender, EventArgs e)
- void `Form1_FormClosing` (object sender, FormClosingEventArgs e)
- void `aboutToolStripMenuItem_Click` (object sender, EventArgs e)
- void `btnSend_Click` (object sender, EventArgs e)
- void `timer_SerialRead_Tick` (object sender, EventArgs e)
- void `lbl_question_MouseMove` (object sender, MouseEventArgs e)
- void `directMessageToolStripMenuItem_Click` (object sender, EventArgs e)
- void `btnDM_Cls_Click` (object sender, EventArgs e)
- void `broadcastToolStripMenuItem_Click` (object sender, EventArgs e)
- void `btnCLS_Click_BrdCst` (object sender, EventArgs e)
- void `btnPrefs_Cls_Click` (object sender, EventArgs e)
- void `btnDM_Clr_Click` (object sender, EventArgs e)
- void `btnDM_Send_Click` (object sender, EventArgs e)
- void `btnFAQcls_Click` (object sender, EventArgs e)
- void `generalFAQToolStripMenuItem_Click` (object sender, EventArgs e)
- void `InklblFAQ_LinkClicked` (object sender, LinkLabelLinkClickedEventArgs e)
- void `btnStuMgmt_Prefs_Click` (object sender, EventArgs e)
- void `btnCLSStudents_Click` (object sender, EventArgs e)
- void `btnEDTStudents_Click` (object sender, EventArgs e)
- void `btnCLRStudents_Click` (object sender, EventArgs e)
- void `btnDoneStudents_Click` (object sender, EventArgs e)
- void `cmbxDM_Click` (object sender, EventArgs e)
- void `cmbxDM_SelectedIndexChanged` (object sender, EventArgs e)
- void `FlowCtrl_Send` (string Address_Student, string desire)
- void `timer_ControlsCreate_Tick` (object sender, EventArgs e)
- void `InklblIssues_LinkClicked` (object sender, LinkLabelLinkClickedEventArgs e)
- void `btnCLS_ConvView_Click` (object sender, EventArgs e)
- void `btnConvView_Click` (object sender, EventArgs e)
- void `IstbxConvView_MouseLeave` (object sender, EventArgs e)
- void `ConvViewPic_Click` (object sender, EventArgs e)
- void `picbx_CV_MouseEnter` (object sender, EventArgs e)
- void `picbx_CV_MouseLeave` (object sender, EventArgs e)
- void `btnCV_Print_Click` (object sender, EventArgs e)
- void `btnCV_Refresh_Click` (object sender, EventArgs e)

- void `Printer_PrintPage` (object sender, System.Drawing.Printing.PrintPageEventArgs e)
- void `multmDelete_Click` (object sender, EventArgs e)
- void `cntxtMenu_arr_Popup` (object sender, EventArgs e)
- void `group_arr_MouseDown` (object sender, MouseEventArgs e)
- void `grpbxFeed_MouseDown` (object sender, MouseEventArgs e)
- void `multmPrefs_Click` (object sender, EventArgs e)
- void `multmDM_Click` (object sender, EventArgs e)
- void `multmFAQ_Click` (object sender, EventArgs e)
- void `multmQuit_Click` (object sender, EventArgs e)
- void `chkbxCtrlHide_CheckedChanged` (object sender, EventArgs e)
- void `multmUndo_Click` (object sender, EventArgs e)
- void `multmName_Click` (object sender, EventArgs e)
- void `showNamesToolStripMenuItem_Click` (object sender, EventArgs e)
- void `btnQuizMaker_Click` (object sender, EventArgs e)
- void `btnQMcls_Click` (object sender, EventArgs e)
- void `btnQMEdit_Click` (object sender, EventArgs e)
- void `btnQMDel_MouseDown` (object sender, MouseEventArgs e)
- void `btnQMclr_Click` (object sender, EventArgs e)
- void `btnAddQM_Click` (object sender, EventArgs e)
- void `txtbxQM_Click` (object sender, EventArgs e)
- void `btnExitQuiz_Click` (object sender, EventArgs e)
- void `btnQuizMode_Click` (object sender, EventArgs e)
- void `multmQuizDelete_Click` (object sender, EventArgs e)
- void `btnQuizAsk_Click` (object sender, EventArgs e)
- void `quizToolStripMenuItem_Click` (object sender, EventArgs e)
- void `attedanceToolStripMenuItem_Click` (object sender, EventArgs e)
- void `btnExitAttendance_Click` (object sender, EventArgs e)
- void `btnExitClassVote_Click` (object sender, EventArgs e)
- void `classVoteToolStripMenuItem_Click` (object sender, EventArgs e)
- void `multmAttendance_Click` (object sender, EventArgs e)
- void `multmClassVote_Click` (object sender, EventArgs e)
- void `multmQuiz_Click` (object sender, EventArgs e)
- void `multmBrdCst_Click` (object sender, EventArgs e)
- void `btnForeColor_Click` (object sender, EventArgs e)
- void `btnBkgrndColor_Click` (object sender, EventArgs e)
- void `btnDefaultTheme_Click` (object sender, EventArgs e)
- void `cmbbxAnimationSpeed_SelectedIndexChanged` (object sender, EventArgs e)
- void `pnlBrdCst_Leave` (object sender, EventArgs e)
- void `lblNotify_MouseDown` (object sender, MouseEventArgs e)
- void `pictureBoxNotify_Click` (object sender, EventArgs e)
- void `ClassVoteHandler` (string vote, int student)
- void `btnCVReset_Click` (object sender, EventArgs e)
- void `multmMarkAsRead_Click` (object sender, EventArgs e)
- void `multmMarkAsUnRead_Click` (object sender, EventArgs e)

Static Private Member Functions

- static string `GenerateKey ()`
- static void `EncryptFile (string sInputFilename, string sOutputFilename, string sKey)`
- static void `DecryptFile (string sInputFilename, string sOutputFilename, string sKey)`

Private Attributes

- string[] `Students_Name` = new string[`classSize`]
- string[] `Students_ID` = new string[`classSize`]
- string[] `Qs_to_create` = new string[`classSize`]
- bool `Qs_to_Make` = false
- string[] `addr_tbl`
- bool[] `Q_status` = new bool[`MaxQuestions`]
- bool[] `Q_Replied` = new bool[`MaxQuestions`]
- bool[] `Q_NameShowing` = new bool[`MaxQuestions`]
- bool[] `wasQunread` = new bool[`MaxQuestions`]
- bool[] `StudentPresent` = new bool[`classSize`]
- bool[] `StudentVoted` = new bool[`classSize`]
- byte `PresentStuNum` = 250
- string[] `Q_sender` = new string[`classSize`]
- string `brdcst_addr` = "\x01"
- string[] `logFiles` = new string[`classSize`]
- string `EncryptedData` = @".data\StudentInfo_Encrypted.txt"
- string `PlainData` = @".data\StudentInfo.txt"
- string[] `portNames` = new string[10]
- string `tempString` = ""
- string `tempString2` = ""
- string `RX_Data` = ""
- string `SecretKey`
- string `StateMachine` = "Normal"
- SoundPlayer `JukeBox` = new System.Media.SoundPlayer()
- bool `WeGotData` = false
- byte `i` = 0
- byte `k` = 0
- byte `j` = 0
- byte `x` = 0
- int `notif_I` = 0
- string[] `logTmpString` = new string[`logHistoryLength`]
- int `ix` = 0
- int `jx` = 0
- string `addr` = "\xFF"
- byte `NumQuestions` = 0
- byte `UnreadCount` = 0

- bool `EDTClicked` = false
- bool `BrdcstShowing` = false
- byte `timesClicked` = 0
- byte `DMtimesClicked` = 0
- byte `PrefsTimesClicked` = 0
- byte `FAQTimesClicked` = 0
- byte `StuMgmtTimesClicked` = 0
- bool `SplashScreenShowing` = new bool()
- bool `aPanellsMoving` = new bool()
- bool `DesireNotify` = new bool()
- bool `NotifyShowing` = new bool()
- bool `DesirePrefs` = new bool()
- bool `DesireDM` = new bool()
- bool `DesireBrdcst` = new bool()
- bool `DesireFAQ` = new bool()
- bool `DesireQzMkr` = new bool()
- bool `DesireQuizMd` = new bool()
- bool `DesireStuMgmt` = new bool()
- bool `DesireAttendance` = new bool()
- bool `DesireClassVote` = new bool()
- bool `DesireConvView` = new bool()
- bool `FAQClicked` = new bool()
- bool `FAQShowing` = new bool()
- bool `StuMgmtClicked` = new bool()
- bool `StuMgmtShowing` = new bool()
- bool `DMclicked` = new bool()
- bool `DMPanelShowing` = new bool()
- bool `PrefsShowing` = new bool()
- bool `PrefsClicked` = new bool()
- bool `ConvViewClicked` = false
- bool `ConvViewShowing` = false
- byte `ConvViewTimesClicked` = 0
- bool `QuizMakerShowing` = false
- bool `QuizMakerClicked` = false
- byte `QuizMakerTimesClicked` = 0
- bool `QuizModeShowing` = false
- bool `QuizModeClicked` = false
- byte `QuizModeTimesClicked` = 0
- bool `AttendanceShowing` = false
- bool `AttendanceClicked` = false
- byte `AttendanceTimesClicked` = 0
- bool `ClassVoteShowing` = false
- bool `ClassVoteClicked` = false
- byte `ClassVoteTimesClicked` = 0
- bool `textbox1WASclicked` = new bool()
- bool `grpbxRPL_WASclicked` = new bool()

- bool `btnCLS_WASclicked` = new bool()
- bool `NEW_grpbx` = new bool()
- bool `AckRXd` = false
- bool `timeout_ack` = false
- int `old_lblID` = 0
- bool `DesireID` = false
- bool `DeleteQuestion` = false
- bool `Question_Deleted` = false
- bool `QuizQuestionDel` = false
- int `QuizDelID` = 255
- string `QzQuestionDel` = ""
- bool `Request_Undo` = false
- bool `sudo_kill` = false
- byte `Del_ID` = 255
- byte `timesThrough` = 0
- int `new_lblID` = 0
- int `lbl_ID` = 0
- int `lbl_ID_2` = 0
- System.Windows.Forms.GroupBox[] `group_arr` = new System.Windows.Forms.-
GroupBox[`classSize`]
- System.Windows.Forms.Button[] `reply_arr` = new System.Windows.Forms.-
Button[`classSize`]
- System.Windows.Forms.Button[] `close_arr` = new System.Windows.Forms.-
Button[`classSize`]
- System.Windows.Forms.Button[] `clear_arr` = new System.Windows.Forms.-
Button[`classSize`]
- System.Windows.Forms.TextBox[] `txtbx_reply_arr` = new System.Windows.-
Forms.TextBox[`classSize`]
- System.Windows.Forms.Label[] `lbl_arr` = new System.Windows.Forms.-
Label[`classSize`]
- System.Windows.Forms.PictureBox[] `picbx_arr` = new System.Windows.Forms.-
PictureBox[`classSize`]
- System.Windows.Forms.PictureBox[] `picbx_ConvView_arr` = new System.-
Windows.Forms.PictureBox[`classSize`]
- System.Windows.Forms.ToolTip[] `tt_picbxCV_arr` = new System.Windows.-
Forms.ToolTip[`classSize`]
- System.Windows.Forms.Panel[] `PanelQuestion_arr` = new System.Windows.-
Forms.Panel[`classSize`]
- System.Windows.Forms.Button[] `btnQuizQuestionSend_arr` = new System.-
Windows.Forms.Button[`classSize`]
- System.Windows.Forms.Label[] `lblQuestionNum_arr` = new System.Windows.-
Forms.Label[`classSize`]
- System.Windows.Forms.TextBox[] `txtbxQuestion_arr` = new System.Windows.-
Forms.TextBox[`classSize`]
- System.Drawing.Point `origingrouparr` = new System.Drawing.Point(6, -13)
- System.Drawing.Point `tempgrouparr` = new System.Drawing.Point(6, 19)
- System.Drawing.Point `tempreplyarr` = new System.Drawing.Point(230, 98)

- System.Drawing.Point `tempclosearr` = new System.Drawing.Point(129, 98)
- System.Drawing.Point `tempcleararr` = new System.Drawing.Point(23, 98)
- System.Drawing.Point `txbx_reply_temp` = new System.Drawing.Point(23, 32)
- System.Drawing.Point `lbl_arr_temp` = new System.Drawing.Point(0, -13)
- System.Drawing.Point `picbx_arr_tmp` = new System.Drawing.Point(2, 9)
- System.Drawing.Point `picbx_ConvView_tmp` = new System.Drawing.Point(296, 7)
- System.Drawing.Point `lblQuestionNum_arr_tmp` = new System.Drawing.Point(5, 21)
- System.Drawing.Point `txbxQuestion_arr_tmp` = new System.Drawing.Point(32, 13)
- System.Drawing.Point `btnQuizQuestionSend_arr_tmp` = new System.Drawing.Point(288, 13)
- System.Drawing.Point `PanelQuestion_arr_tmp` = new System.Drawing.Point(7, 6)

Static Private Attributes

- static int `logHistoryLength` = 500

5.3.1 Detailed Description

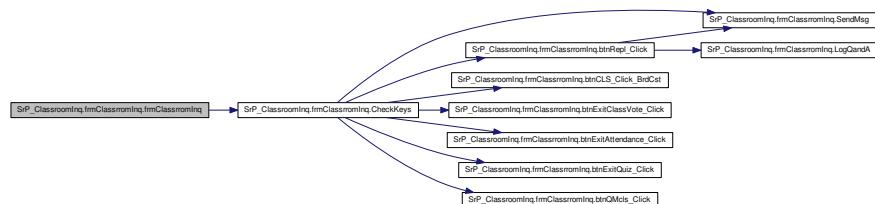
Definition at line 35 of file [Form1-dyn-ctrls.cs](#).

5.3.2 Constructor & Destructor Documentation

5.3.2.1 SrP_ClassroomInq::frmClassrromInq::frmClassrromInq ()

Definition at line 42 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3 Member Function Documentation

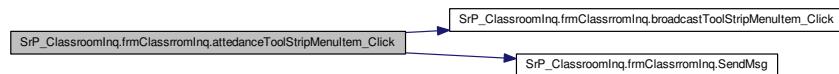
5.3.3.1 void SrP_ClassroomInq::frmClassrromInq::aboutToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 1005 of file [Form1-dyn-ctrls.cs](#).

5.3.3.2 void SrP_ClassroomInq::frmClassroomInq::attendanceToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 2661 of file Form1-dyn-ctrls.cs.

Here is the call graph for this function:



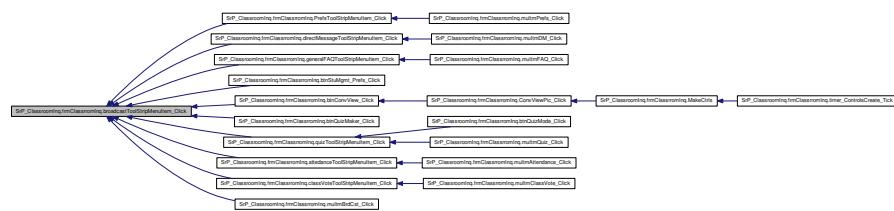
Here is the caller graph for this function:



5.3.3.3 void SrP_ClassroomInq::frmClassroomInq::broadcastToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 1147 of file Form1-dyn-ctrls.cs.

Here is the caller graph for this function:



5.3.3.4 void SrP_ClassroomInq::frmClassroomInq::btnAddQM_Click (object sender, EventArgs e) [private]

Definition at line 2427 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.5 void SrP_ClassroomInq::frmClassroomInq::btnBkgrndColor_Click (object sender, EventArgs e) [private]

Definition at line 2772 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.6 void SrP_ClassroomInq::frmClassroomInq::btnCLR_Click (object sender, EventArgs e) [private]

Definition at line 606 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.7 void SrP_ClassroomInq::frmClassroomInq::btnCLRStudents_Click (object sender, EventArgs e) [private]

Definition at line 1455 of file [Form1-dyn-ctrls.cs](#).

5.3.3.8 void SrP_ClassroomInq::frmClassroomInq::btnCLS_Click (object sender, EventArgs e) [private]

Definition at line 724 of file [Form1-dyn-ctrls.cs](#).

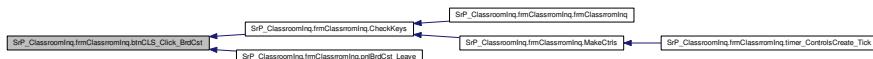
Here is the caller graph for this function:



5.3.3.9 void SrP_ClassroomInq::frmClassroomInq::btnCLS_Click_BrdCst (object sender, EventArgs e) [private]

Definition at line 1163 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.10 void SrP_ClassroomInq::frmClassroomInq::btnCLS_ConvView_Click (object sender, EventArgs e) [private]

Definition at line 1730 of file [Form1-dyn-ctrls.cs](#).

5.3.3.11 void SrP_ClassroomInq::frmClassroomInq::btnCLSSStudents_Click (object sender, EventArgs e) [private]

Definition at line 1318 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.12 void SrP_ClassroomInq::frmClassromInq::btnConvView_Click (object sender, EventArgs e) [private]

Definition at line 1738 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.13 void SrP_ClassroomInq::frmClassromInq::btnCV_Print_Click (object sender, EventArgs e) [private]

Definition at line 1867 of file [Form1-dyn-ctrls.cs](#).

5.3.3.14 void SrP_ClassroomInq::frmClassromInq::btnCV_Refresh_Click (object sender, EventArgs e) [private]

Definition at line 1900 of file [Form1-dyn-ctrls.cs](#).

5.3.3.15 void SrP_ClassroomInq::frmClassromInq::btnCVReset_Click (object sender, EventArgs e) [private]

Definition at line 2923 of file [Form1-dyn-ctrls.cs](#).

5.3.3.16 void SrP_ClassroomInq::frmClassromInq::btnDefaultTheme_Click (object sender, EventArgs e) [private]

Definition at line 2782 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.17 void SrP_ClassroomInq::frmClassroomInq::btnDM_Clr_Click (object sender, EventArgs e) [private]

Definition at line 1179 of file [Form1-dyn-ctrls.cs](#).

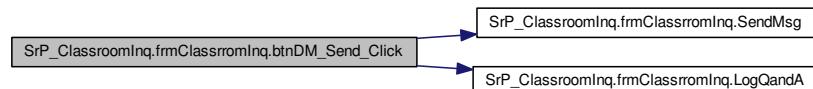
5.3.3.18 void SrP_ClassroomInq::frmClassroomInq::btnDM_Cls_Click (object sender, EventArgs e) [private]

Definition at line 1140 of file [Form1-dyn-ctrls.cs](#).

5.3.3.19 void SrP_ClassroomInq::frmClassroomInq::btnDM_Send_Click (object sender, EventArgs e) [private]

Definition at line 1185 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.20 void SrP_ClassroomInq::frmClassroomInq::btnDoneStudents_Click (object sender, EventArgs e) [private]

Definition at line 1473 of file [Form1-dyn-ctrls.cs](#).

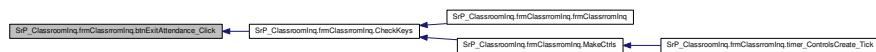
5.3.3.21 void SrP_ClassroomInq::frmClassroomInq::btnEDTStudents_Click (object sender, EventArgs e) [private]

Definition at line 1438 of file [Form1-dyn-ctrls.cs](#).

5.3.3.22 void SrP_ClassroomInq::frmClassromInq::btnExitAttendance_Click (object sender, EventArgs e) [private]

Definition at line 2688 of file [Form1-dyn-ctrls.cs](#).

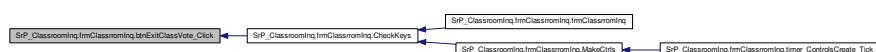
Here is the caller graph for this function:



5.3.3.23 void SrP_ClassroomInq::frmClassromInq::btnExitClassVote_Click (object sender, EventArgs e) [private]

Definition at line 2701 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.24 void SrP_ClassroomInq::frmClassromInq::btnExitQuiz_Click (object sender, EventArgs e) [private]

Definition at line 2537 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.25 void SrP_ClassroomInq::frmClassromInq::btnFAQcls_Click (object sender, EventArgs e) [private]

Definition at line 1217 of file [Form1-dyn-ctrls.cs](#).

5.3.3.26 void SrP_ClassroomInq::frmClassromInq::btnForeColor_Click (object sender, EventArgs e) [private]

Definition at line 2762 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.27 void SrP_ClassroomInq::frmClassromInq::btnPrefs_Cls_Click (object sender, EventArgs e) [private]

Definition at line 1171 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



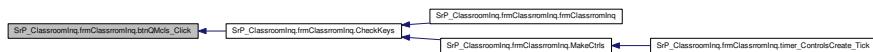
5.3.3.28 void SrP_ClassroomInq::frmClassromInq::btnQMclr_Click (object sender, EventArgs e) [private]

Definition at line 2420 of file [Form1-dyn-ctrls.cs](#).

5.3.3.29 void SrP_ClassroomInq::frmClassromInq::btnQMcls_Click (object sender, EventArgs e) [private]

Definition at line 2315 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.30 void SrP_ClassroomInq::frmClassromInq::btnQMdel_MouseDown (object sender, MouseEventArgs e) [private]

Definition at line 2381 of file [Form1-dyn-ctrls.cs](#).

5.3.3.31 void SrP_ClassroomInq::frmClassromInq::btnQMEdit_Click (object sender, EventArgs e) [private]

Definition at line [2341](#) of file [Form1-dyn-ctrls.cs](#).

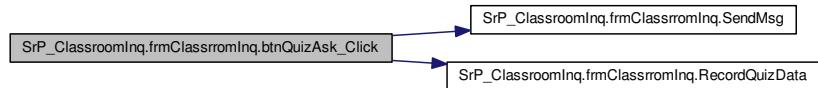
Here is the caller graph for this function:



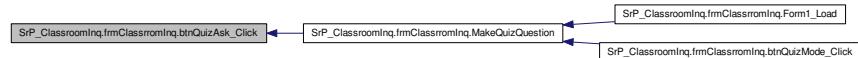
5.3.3.32 void SrP_ClassroomInq::frmClassromInq::btnQuizAsk_Click (object sender, EventArgs e) [private]

Definition at line [2574](#) of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.33 void SrP_ClassroomInq::frmClassromInq::btnQuizMaker_Click (object sender, EventArgs e) [private]

Definition at line [2257](#) of file [Form1-dyn-ctrls.cs](#).

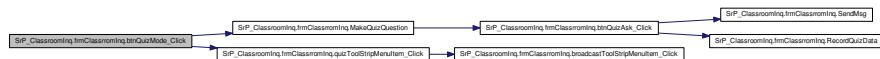
Here is the call graph for this function:



5.3.3.34 void SrP_ClassroomInq::frmClassromInq::btnQuizMode_Click (object sender, EventArgs e) [private]

Definition at line 2548 of file [Form1-dyn-ctrls.cs](#).

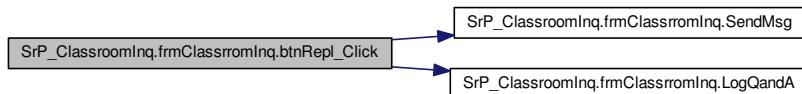
Here is the call graph for this function:



5.3.3.35 void SrP_ClassroomInq::frmClassromInq::btnRepl_Click (object sender, EventArgs e) [private]

Definition at line 624 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.36 void SrP_ClassroomInq::frmClassromInq::btnSend_Click (object sender, EventArgs e) [private]

Definition at line 1012 of file [Form1-dyn-ctrls.cs](#).

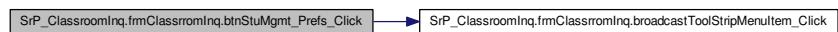
Here is the call graph for this function:



5.3.3.37 void SrP_ClassroomInq::frmClassromInq::btnStuMgmt_Prefs_Click (object sender, EventArgs e) [private]

Definition at line 1255 of file [Form1-dyn-ctrls.cs](#).

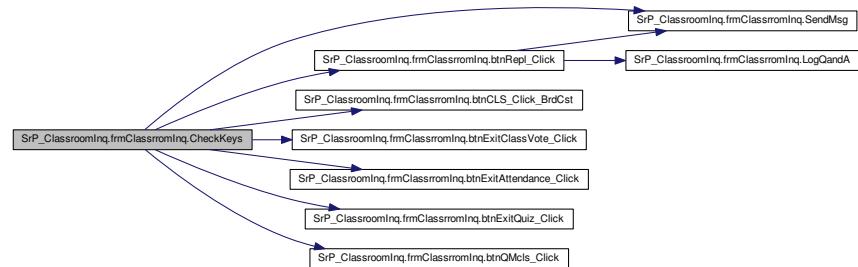
Here is the call graph for this function:



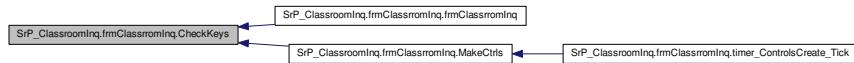
5.3.3.38 void SrP_ClassroomInq::frmClassromInq::CheckKeys (object sender, System.Windows.Forms.KeyPressEventArgs e)

Definition at line 502 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.39 void SrP_ClassroomInq::frmClassroomInq::chkbxCtrlHide_CheckedChanged (object sender, EventArgs e) [private]

Definition at line 2160 of file [Form1-dyn-ctrls.cs](#).

5.3.3.40 void SrP_ClassroomInq::frmClassroomInq::ClassVoteHandler (string vote, int student) [private]

Definition at line 2844 of file [Form1-dyn-ctrls.cs](#).

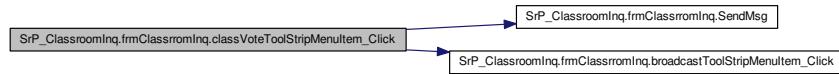
Here is the caller graph for this function:



5.3.3.41 void SrP_ClassroomInq::frmClassroomInq::classVoteToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 2714 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.42 void SrP_ClassroomInq::frmClassromInq::cmbbxAnimation-Speed_SelectedIndexChanged (object sender, EventArgs e)
[private]

Definition at line 2791 of file [Form1-dyn-ctrls.cs](#).

5.3.3.43 void SrP_ClassroomInq::frmClassromInq::cmbxDM_Click (object sender, EventArgs e) [private]

Definition at line 1487 of file [Form1-dyn-ctrls.cs](#).

5.3.3.44 void SrP_ClassroomInq::frmClassromInq::cmbxDM_SelectedIndexChanged (object sender, EventArgs e) [private]

Definition at line 1520 of file [Form1-dyn-ctrls.cs](#).

5.3.3.45 void SrP_ClassroomInq::frmClassromInq::ctxtMenu_arr_Popup (object sender, EventArgs e) [private]

Definition at line 2036 of file [Form1-dyn-ctrls.cs](#).

5.3.3.46 void SrP_ClassroomInq::frmClassromInq::ConvViewPic_Click (object sender, EventArgs e) [private]

Definition at line 1803 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.47 static void SrP_ClassroomInq::frmClassromInq::DecryptFile (string sInputFilename, string sOutputFilename, string sKey) [static, private]

Definition at line 1409 of file [Form1-dyn-ctrls.cs](#).

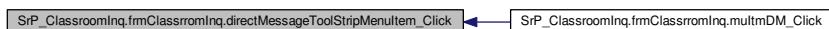
5.3.3.48 void SrP_ClassroomInq::frmClassromInq::directMessageToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 1106 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.49 static void SrP_ClassroomInq::frmClassromInq::EncryptFile (string sInputFilename, string sOutputFilename, string sKey) [static, private]

Definition at line 1384 of file [Form1-dyn-ctrls.cs](#).

5.3.3.50 void SrP_ClassroomInq::frmClassromInq::FlowCtrl_Send (string Address_Student, string desire) [private]

Definition at line 1530 of file [Form1-dyn-ctrls.cs](#).

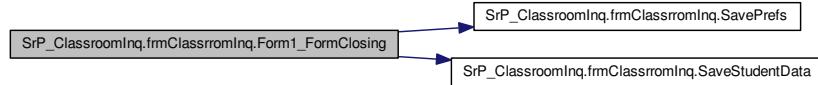
Here is the caller graph for this function:



5.3.3.51 void SrP_ClassroomInq::frmClassrromInq::Form1_FormClosing (object sender, FormClosingEventArgs e) [private]

Definition at line [932](#) of file [Form1-dyn-ctrls.cs](#).

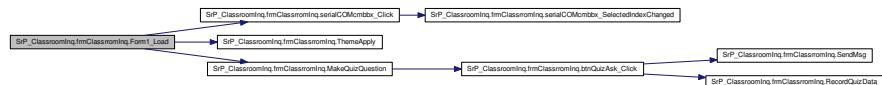
Here is the call graph for this function:



5.3.3.52 void SrP_ClassroomInq::frmClassrromInq::Form1_Load (object sender, EventArgs e) [private]

Definition at line [732](#) of file [Form1-dyn-ctrls.cs](#).

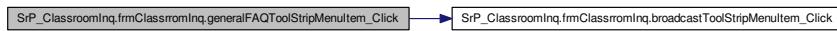
Here is the call graph for this function:



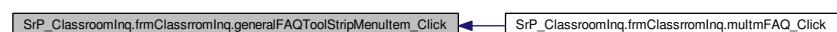
5.3.3.53 void SrP_ClassroomInq::frmClassrromInq::generalFAQToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line [1229](#) of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.54 static string SrP_ClassroomInq::frmClassroomInq::GenerateKey() [static, private]

Definition at line 1375 of file [Form1-dyn-ctrls.cs](#).

5.3.3.55 void SrP_ClassroomInq::frmClassroomInq::group_arr_MouseDown(object sender, MouseEventArgs e) [private]

Definition at line 2094 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.56 void SrP_ClassroomInq::frmClassroomInq::grpbxFeed_MouseDown(object sender, MouseEventArgs e) [private]

Definition at line 2122 of file [Form1-dyn-ctrls.cs](#).

5.3.3.57 string SrP_ClassroomInq::frmClassroomInq::HandleBCKSPC(string RAW_DATA)

Definition at line 405 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



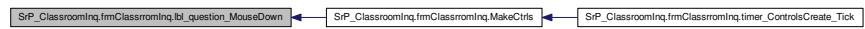
5.3.3.58 void SrP_ClassroomInq::frmClassromInq::lbl_question_MouseDown (object sender, MouseEventArgs e) [private]

Definition at line 674 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



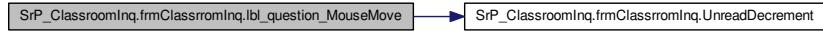
Here is the caller graph for this function:



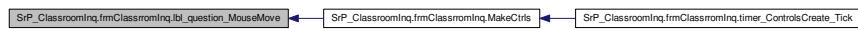
5.3.3.59 void SrP_ClassroomInq::frmClassromInq::lbl_question_MouseMove (object sender, MouseEventArgs e) [private]

Definition at line 1097 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



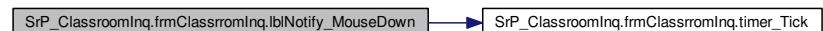
Here is the caller graph for this function:



5.3.3.60 void SrP_ClassroomInq::frmClassromInq::lblNotify_MouseDown (object sender, MouseEventArgs e) [private]

Definition at line 2830 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.61 void SrP_ClassroomInq::frmClassromInq::lnklblFAQ_LinkClicked (object sender, LinkLabelLinkClickedEventArgs e) [private]

Definition at line 1245 of file [Form1-dyn-ctrls.cs](#).

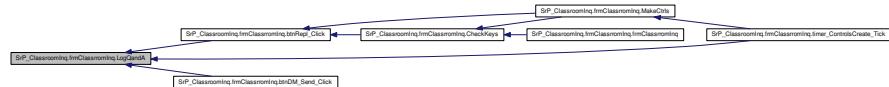
5.3.3.62 void SrP_ClassroomInq::frmClassromInq::lnklblIssues_LinkClicked (object sender, LinkLabelLinkClickedEventArgs e) [private]

Definition at line 1651 of file [Form1-dyn-ctrls.cs](#).

5.3.3.63 bool SrP_ClassroomInq::frmClassromInq::LogQandA (string dialog, bool teacher, string recipient)

Definition at line 1661 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



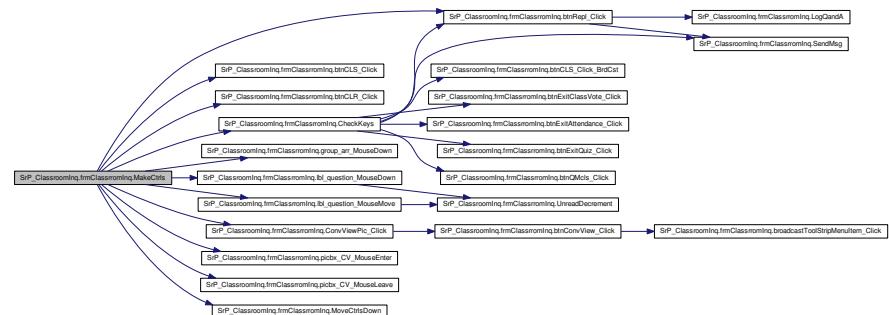
5.3.3.64 void SrP_ClassroomInq::frmClassromInq::lstbxConvView_MouseLeave (object sender, EventArgs e) [private]

Definition at line 1797 of file [Form1-dyn-ctrls.cs](#).

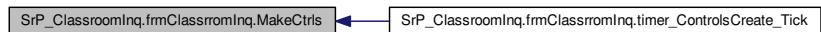
5.3.3.65 void SrP_ClassroomInq::frmClassromInq::MakeCtrls (string question)

Definition at line 221 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



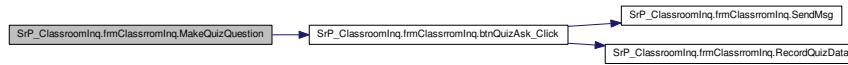
Here is the caller graph for this function:



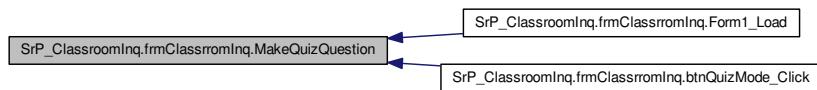
5.3.3.66 bool SrP_ClassroomInq::frmClassromInq::MakeQuizQuestion (string question, int number)

Definition at line 2466 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



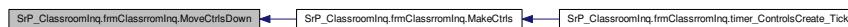
Here is the caller graph for this function:



5.3.3.67 void SrP_ClassroomInq::frmClassromInq::MoveCtrlsDown ()

Definition at line 213 of file [Form1-dyn-ctrls.cs](#).

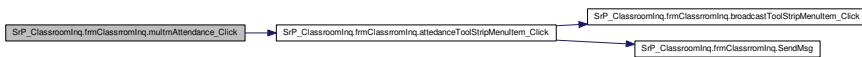
Here is the caller graph for this function:



5.3.3.68 void SrP_ClassroomInq::frmClassromInq::multmAttendance_Click (object sender, EventArgs e) [private]

Definition at line 2738 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.69 void SrP_ClassroomInq::frmClassromInq::multmBrdCst_Click (object sender, EventArgs e) [private]

Definition at line 2756 of file [Form1-dyn-ctrls.cs](#).

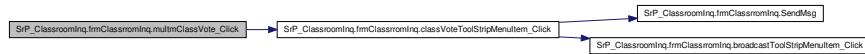
Here is the call graph for this function:



5.3.3.70 void SrP_ClassroomInq::frmClassromInq::multmClassVote_Click (object sender, EventArgs e) [private]

Definition at line 2744 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.71 void SrP_ClassroomInq::frmClassromInq::multmDelete_Click (object sender, EventArgs e) [private]

Definition at line 1984 of file [Form1-dyn-ctrls.cs](#).

5.3.3.72 void SrP_ClassroomInq::frmClassromInq::multmDM_Click (object sender, EventArgs e) [private]

Definition at line 2141 of file [Form1-dyn-ctrls.cs](#).

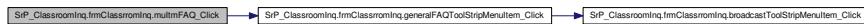
Here is the call graph for this function:



5.3.3.73 void SrP_ClassroomInq::frmClassroomInq::multmFAQ_Click (object sender, EventArgs e) [private]

Definition at line 2147 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.74 void SrP_ClassroomInq::frmClassroomInq::multmMarkAsRead_Click (object sender, EventArgs e) [private]

Definition at line 2944 of file [Form1-dyn-ctrls.cs](#).

5.3.3.75 void SrP_ClassroomInq::frmClassroomInq::multmMarkAsUnRead_Click (object sender, EventArgs e) [private]

Definition at line 2960 of file [Form1-dyn-ctrls.cs](#).

5.3.3.76 void SrP_ClassroomInq::frmClassroomInq::multmName_Click (object sender, EventArgs e) [private]

Definition at line 2183 of file [Form1-dyn-ctrls.cs](#).

5.3.3.77 void SrP_ClassroomInq::frmClassroomInq::multmPrefs_Click (object sender, EventArgs e) [private]

Definition at line 2135 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.78 void SrP_ClassroomInq::frmClassroomInq::multmQuit_Click (object sender, EventArgs e) [private]

Definition at line 2153 of file [Form1-dyn-ctrls.cs](#).

5.3.3.79 void SrP_ClassroomInq::frmClassromInq::multmQuiz_Click (object sender, EventArgs e) [private]

Definition at line 2750 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



5.3.3.80 void SrP_ClassroomInq::frmClassromInq::multmQuizDelete_Click (object sender, EventArgs e) [private]

Definition at line 2564 of file [Form1-dyn-ctrls.cs](#).

5.3.3.81 void SrP_ClassroomInq::frmClassromInq::multmUndo_Click (object sender, EventArgs e) [private]

Definition at line 2167 of file [Form1-dyn-ctrls.cs](#).

5.3.3.82 void SrP_ClassroomInq::frmClassromInq::picbx_CV_MouseEnter (object sender, EventArgs e) [private]

Definition at line 1828 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.83 void SrP_ClassroomInq::frmClassromInq::picbx_CV_MouseLeave (object sender, EventArgs e) [private]

Definition at line 1849 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



```
5.3.3.84 void SrP_ClassroomInq::frmClassroomInq::pictureBoxNotify_Click ( object sender,
EventArgs e ) [private]
```

Definition at line 2837 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



```
5.3.3.85 void SrP_ClassroomInq::frmClassroomInq::pnlBrdCst_Leave ( object sender, EventArgs
e ) [private]
```

Definition at line 2821 of file [Form1-dyn-ctrls.cs](#).

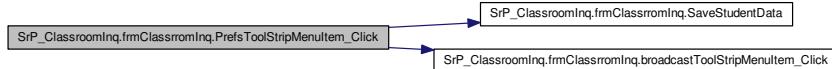
Here is the call graph for this function:



```
5.3.3.86 void SrP_ClassroomInq::frmClassroomInq::PrefsToolStripMenuItem_Click ( object
sender, EventArgs e ) [private]
```

Definition at line 859 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.87 void SrP_ClassroomInq::frmClassrromInq::Printer_PrintPage (object sender, System.Drawing.Printing.PrintPageEventArgs e) [private]

Definition at line 1942 of file [Form1-dyn-ctrls.cs](#).

5.3.3.88 void SrP_ClassroomInq::frmClassrromInq::quitToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 580 of file [Form1-dyn-ctrls.cs](#).

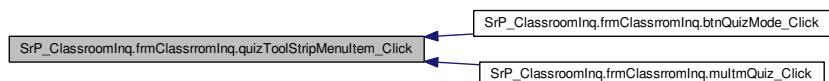
5.3.3.89 void SrP_ClassroomInq::frmClassrromInq::quizToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 2595 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



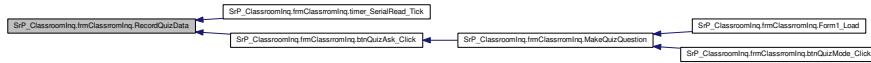
Here is the caller graph for this function:



5.3.3.90 void SrP_ClassroomInq::frmClassrromInq::RecordQuizData (string question, string name_sender)

Definition at line 2636 of file [Form1-dyn-ctrls.cs](#).

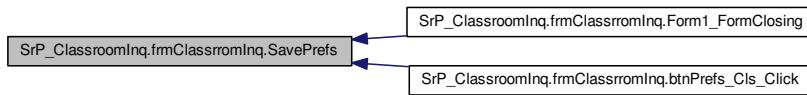
Here is the caller graph for this function:



5.3.3.91 void SrP_ClassroomInq::frmClassrromInq::SavePrefs ()

Definition at line 1601 of file [Form1-dyn-ctrls.cs](#).

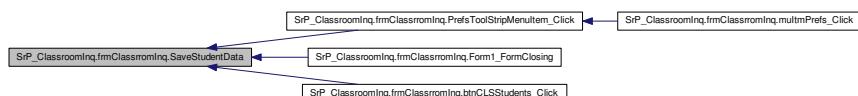
Here is the caller graph for this function:



5.3.3.92 void SrP_ClassroomInq::frmClassrromInq::SaveStudentData ()

Definition at line 1328 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.93 bool SrP_ClassroomInq::frmClassrromInq::SendMsg (string *Message2Send*, string *Address2Send*)

Definition at line 1577 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.94 void SrP_ClassroomInq::frmClassromInq::serialCOMcmbbx_Click (object sender, EventArgs e) [private]

Definition at line 893 of file [Form1-dyn-ctrls.cs](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.3.3.95 void SrP_ClassroomInq::frmClassrromInq::serialCOMcmbbx_SelectedIndexChanged (object sender, EventArgs e) [private]

Definition at line 911 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



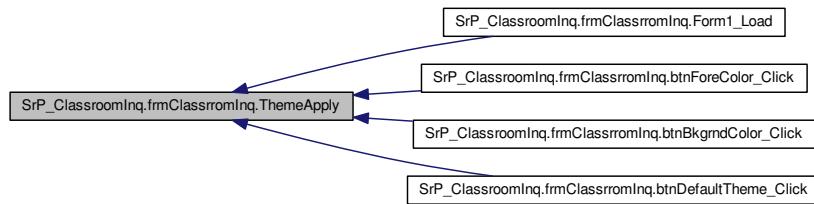
5.3.3.96 void SrP_ClassroomInq::frmClassroomInq::showNamesToolStripMenuItem_Click (object sender, EventArgs e) [private]

Definition at line 2210 of file [Form1-dyn-ctrls.cs](#).

5.3.3.97 void SrP_ClassroomInq::frmClassroomInq::ThemeApply (Color Fore, Color Back)

Definition at line 46 of file [themes.cs](#).

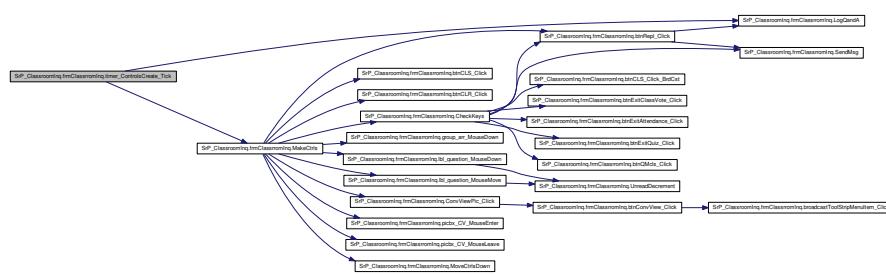
Here is the caller graph for this function:



5.3.3.98 void SrP_ClassroomInq::frmClassroomInq::timer_ControlsCreate_Tick (object sender, EventArgs e) [private]

Definition at line 1555 of file [Form1-dyn-ctrls.cs](#).

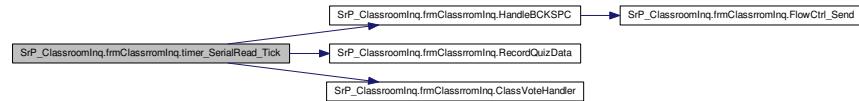
Here is the call graph for this function:



5.3.3.99 void SrP_ClassroomInq::frmClassroomInq::timer_SerialRead_Tick (object sender, EventArgs e) [private]

Definition at line 1031 of file [Form1-dyn-ctrls.cs](#).

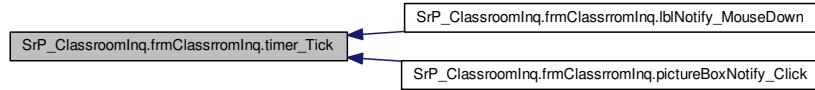
Here is the call graph for this function:



5.3.3.100 void SrP_ClassroomInq::frmClassrromInq::timer_Tick (object sender, EventArgs e) [private]

Definition at line 45 of file [AnimateTimer.cs](#).

Here is the caller graph for this function:



5.3.3.101 void SrP_ClassroomInq::frmClassrromInq::trayICON_MouseDoubleClick (object sender, MouseEventArgs e) [private]

Definition at line 586 of file [Form1-dyn-ctrls.cs](#).

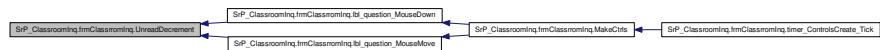
5.3.3.102 void SrP_ClassroomInq::frmClassrromInq::txtbxQM_Click (object sender, EventArgs e) [private]

Definition at line 2452 of file [Form1-dyn-ctrls.cs](#).

5.3.3.103 void SrP_ClassroomInq::frmClassrromInq::UnreadDecrement (object sender)

Definition at line 1621 of file [Form1-dyn-ctrls.cs](#).

Here is the caller graph for this function:



5.3.3.104 static bool SrP_ClassroomInq::frmClassrromInq::ZeroMemory (IntPtr *Destination*, int *Length*)

5.3.4 Member Data Documentation

5.3.4.1 AboutBox_CI SrP_ClassroomInq::frmClassrromInq::About = new
AboutBox_CI()

Definition at line 66 of file [Form1-dyn-ctrls.cs](#).

5.3.4.2 bool SrP_ClassroomInq::frmClassrromInq::AckRXd = false [private]

Definition at line 159 of file [Form1-dyn-ctrls.cs](#).

5.3.4.3 string SrP_ClassroomInq::frmClassrromInq::addr = "\xFF" [private]

Definition at line 108 of file [Form1-dyn-ctrls.cs](#).

5.3.4.4 string [] SrP_ClassroomInq::frmClassrromInq::addr_tbl [private]

Initial value:

```
{"\x20", "\x02", "\x03", "\x1F", "\x05", "\x06", "\x07", "\x22", "\x09", "\x21",
"\x0B", "\x0C", "\x0D", "\x0E", "\x0F", "\x10", "\x11", "\x
12", "\x13", "\x14",
"\x15", "\x16", "\x17", "\x18", "\x19", "\x1A", "\x1B", "\x
1C", "\x1D", "\x1E",
}
```

Definition at line 74 of file [Form1-dyn-ctrls.cs](#).

5.3.4.5 bool SrP_ClassroomInq::frmClassrromInq::aPanellsMoving = new bool()
[private]

Definition at line 119 of file [Form1-dyn-ctrls.cs](#).

5.3.4.6 bool SrP_ClassroomInq::frmClassrromInq::AttendanceClicked = false
[private]

Definition at line 150 of file [Form1-dyn-ctrls.cs](#).

5.3.4.7 bool SrP_ClassroomInq::frmClassrromInq::AttendanceShowing = false
[private]

Definition at line 149 of file [Form1-dyn-ctrls.cs](#).

5.3.4.8 byte SrP_ClassroomInq::frmClassrromInq::AttendanceTimesClicked = 0
[private]

Definition at line 151 of file [Form1-dyn-ctrls.cs](#).

5.3.4.9 Color SrP_ClassroomInq::frmClassrromInq::BackColorTheme = Color.Black

Definition at line 65 of file [Form1-dyn-ctrls.cs](#).

5.3.4.10 string SrP_ClassroomInq::frmClassrromInq::brdcst_addr = "\x01"
[private]

Definition at line 87 of file [Form1-dyn-ctrls.cs](#).

5.3.4.11 bool SrP_ClassroomInq::frmClassrromInq::BrdcstShowing = false
[private]

Definition at line 112 of file [Form1-dyn-ctrls.cs](#).

5.3.4.12 bool SrP_ClassroomInq::frmClassrromInq::btnCLS_WASclicked = new
bool() [private]

Definition at line 157 of file [Form1-dyn-ctrls.cs](#).

5.3.4.13 System.Windows.Forms.Button [] SrP_ClassroomInq::frmClassrromInq::btn-
QuizQuestionSend_arr = new System.Windows.Forms.Button[classSize]
[private]

Definition at line 189 of file [Form1-dyn-ctrls.cs](#).

5.3.4.14 System.Drawing.Point SrP_ClassroomInq::frmClassrromInq::btn-
QuizQuestionSend_arr_tmp = new System.Drawing.Point(288, 13)
[private]

Definition at line 205 of file [Form1-dyn-ctrls.cs](#).

5.3.4.15 const int SrP_ClassroomInq::frmClassrromInq::classSize = 50

Definition at line 60 of file [Form1-dyn-ctrls.cs](#).

5.3.4.16 bool SrP_ClassroomInq::frmClassrromInq::ClassVoteClicked = false
[private]

Definition at line 153 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.17 bool SrP_ClassroomInq::frmClassrromInq::ClassVoteShowing = false  
[private]
```

Definition at line 152 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.18 byte SrP_ClassroomInq::frmClassrromInq::ClassVoteTimesClicked = 0  
[private]
```

Definition at line 154 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.19 System.Windows.Forms.Button [] SrP_ClassroomInq::frmClassrrom-  
Inq::clear_arr = new System.Windows.Forms.Button[classSize]  
[private]
```

Definition at line 181 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.20 System.Windows.Forms.Button [] SrP_ClassroomInq::frmClassrrom-  
Inq::close_arr = new System.Windows.Forms.Button[classSize]  
[private]
```

Definition at line 180 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.21 bool SrP_ClassroomInq::frmClassrromInq::ConvViewClicked = false  
[private]
```

Definition at line 140 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.22 bool SrP_ClassroomInq::frmClassrromInq::ConvViewShowing = false  
[private]
```

Definition at line 141 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.23 byte SrP_ClassroomInq::frmClassrromInq::ConvViewTimesClicked = 0  
[private]
```

Definition at line 142 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.24 byte SrP_ClassroomInq::frmClassrromInq::Del_ID = 255 [private]
```

Definition at line 170 of file [Form1-dyn-ctrls.cs](#).

5.3.4.25 **bool SrP_ClassroomInq::frmClassrromInq::DeleteQuestion = false**
[private]

Definition at line 163 of file [Form1-dyn-ctrls.cs](#).

5.3.4.26 **bool SrP_ClassroomInq::frmClassrromInq::DesireAttendance = new bool()**
[private]

Definition at line 129 of file [Form1-dyn-ctrls.cs](#).

5.3.4.27 **bool SrP_ClassroomInq::frmClassrromInq::DesireBrdcst = new bool()**
[private]

Definition at line 124 of file [Form1-dyn-ctrls.cs](#).

5.3.4.28 **bool SrP_ClassroomInq::frmClassrromInq::DesireClassVote = new bool()**
[private]

Definition at line 130 of file [Form1-dyn-ctrls.cs](#).

5.3.4.29 **bool SrP_ClassroomInq::frmClassrromInq::DesireConvView = new bool()**
[private]

Definition at line 131 of file [Form1-dyn-ctrls.cs](#).

5.3.4.30 **bool SrP_ClassroomInq::frmClassrromInq::DesireDM = new bool()**
[private]

Definition at line 123 of file [Form1-dyn-ctrls.cs](#).

5.3.4.31 **bool SrP_ClassroomInq::frmClassrromInq::DesireFAQ = new bool()**
[private]

Definition at line 125 of file [Form1-dyn-ctrls.cs](#).

5.3.4.32 **bool SrP_ClassroomInq::frmClassrromInq::DesireID = false** [private]

Definition at line 162 of file [Form1-dyn-ctrls.cs](#).

5.3.4.33 **bool SrP_ClassroomInq::frmClassrromInq::DesireNotify = new bool()**
[private]

Definition at line 120 of file [Form1-dyn-ctrls.cs](#).

5.3.4.34 **bool SrP_ClassroomInq::frmClassrromInq::DesirePrefs = new bool()**
[private]

Definition at line 122 of file [Form1-dyn-ctrls.cs](#).

5.3.4.35 **bool SrP_ClassroomInq::frmClassrromInq::DesireQuizMd = new bool()**
[private]

Definition at line 127 of file [Form1-dyn-ctrls.cs](#).

5.3.4.36 **bool SrP_ClassroomInq::frmClassrromInq::DesireQzMkr = new bool()**
[private]

Definition at line 126 of file [Form1-dyn-ctrls.cs](#).

5.3.4.37 **bool SrP_ClassroomInq::frmClassrromInq::DesireStuMgmt = new bool()**
[private]

Definition at line 128 of file [Form1-dyn-ctrls.cs](#).

5.3.4.38 **bool SrP_ClassroomInq::frmClassrromInq::DMclicked = new bool()**
[private]

Definition at line 136 of file [Form1-dyn-ctrls.cs](#).

5.3.4.39 **bool SrP_ClassroomInq::frmClassrromInq::DMPanelShowing = new bool()**
[private]

Definition at line 137 of file [Form1-dyn-ctrls.cs](#).

5.3.4.40 **byte SrP_ClassroomInq::frmClassrromInq::DMtimesClicked = 0**
[private]

Definition at line 114 of file [Form1-dyn-ctrls.cs](#).

5.3.4.41 **bool SrP_ClassroomInq::frmClassrromInq::EDTClicked = false**
[private]

Definition at line 111 of file [Form1-dyn-ctrls.cs](#).

5.3.4.42 **string SrP_ClassroomInq::frmClassrromInq::EncryptedData =
@".data\StudentInfo.Encrypted.txt"** [private]

Definition at line 89 of file [Form1-dyn-ctrls.cs](#).

5.3.4.43 **bool SrP_ClassroomInq::frmClassrromInq::FAQClicked = new bool()**
[private]

Definition at line 132 of file [Form1-dyn-ctrls.cs](#).

5.3.4.44 **bool SrP_ClassroomInq::frmClassrromInq::FAQShowing = new bool()**
[private]

Definition at line 133 of file [Form1-dyn-ctrls.cs](#).

5.3.4.45 **byte SrP_ClassroomInq::frmClassrromInq::FAQTimesClicked = 0**
[private]

Definition at line 116 of file [Form1-dyn-ctrls.cs](#).

5.3.4.46 **Color SrP_ClassroomInq::frmClassrromInq::ForeColorTheme = Color.DarkGoldenrod**

Definition at line 64 of file [Form1-dyn-ctrls.cs](#).

5.3.4.47 **System.Windows.Forms.GroupBox [] SrP_ClassroomInq::frmClassrromInq::group_arr = new System.Windows.Forms.GroupBox[classSize]**
[private]

Definition at line 178 of file [Form1-dyn-ctrls.cs](#).

5.3.4.48 **bool SrP_ClassroomInq::frmClassrromInq::grpbxRPL_WASclicked = new bool()** [private]

Definition at line 156 of file [Form1-dyn-ctrls.cs](#).

5.3.4.49 **byte SrP_ClassroomInq::frmClassrromInq::i = 0** [private]

Definition at line 99 of file [Form1-dyn-ctrls.cs](#).

5.3.4.50 **int SrP_ClassroomInq::frmClassrromInq::ix = 0** [private]

Definition at line 106 of file [Form1-dyn-ctrls.cs](#).

5.3.4.51 **byte SrP_ClassroomInq::frmClassrromInq::j = 0** [private]

Definition at line 101 of file [Form1-dyn-ctrls.cs](#).

5.3.4.52 SoundPlayer SrP_ClassroomInq::frmClassroomInq::JukeBox = new
System.Media.SoundPlayer() [private]

Definition at line 97 of file [Form1-dyn-ctrls.cs](#).

5.3.4.53 int SrP_ClassroomInq::frmClassroomInq::jx = 0 [private]

Definition at line 107 of file [Form1-dyn-ctrls.cs](#).

5.3.4.54 byte SrP_ClassroomInq::frmClassroomInq::k = 0 [private]

Definition at line 100 of file [Form1-dyn-ctrls.cs](#).

5.3.4.55 System.Windows.Forms.Label [] SrP_ClassroomInq::frmClassroomInq::lbl_arr
= new System.Windows.Forms.Label[classSize] [private]

Definition at line 183 of file [Form1-dyn-ctrls.cs](#).

5.3.4.56 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq::lbl_arr_temp =
new System.Drawing.Point(0, -13) [private]

Definition at line 199 of file [Form1-dyn-ctrls.cs](#).

5.3.4.57 int SrP_ClassroomInq::frmClassroomInq::lbl_ID = 0 [private]

Definition at line 173 of file [Form1-dyn-ctrls.cs](#).

5.3.4.58 int SrP_ClassroomInq::frmClassroomInq::lbl_ID_2 = 0 [private]

Definition at line 174 of file [Form1-dyn-ctrls.cs](#).

5.3.4.59 System.Windows.Forms.Label [] SrP_ClassroomInq::frmClassroomInq-
::lblQuestionNum_arr = new System.Windows.Forms.Label[classSize]
[private]

Definition at line 190 of file [Form1-dyn-ctrls.cs](#).

5.3.4.60 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq-
::lblQuestionNum_arr_tmp = new System.Drawing.Point(5, 21)
[private]

Definition at line 203 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.61 string [] SrP_ClassroomInq::frmClassrromInq::logFiles = new  
    string[classSize] [private]
```

Definition at line [88](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.62 int SrP_ClassroomInq::frmClassrromInq::logHistoryLength = 500  
    [static, private]
```

Definition at line [104](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.63 string [] SrP_ClassroomInq::frmClassrromInq::logTmpString = new  
    string[logHistoryLength] [private]
```

Definition at line [105](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.64 const int SrP_ClassroomInq::frmClassrromInq::MaxQuestions = 250
```

Definition at line [61](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.65 bool SrP_ClassroomInq::frmClassrromInq::NEW_grpbx = new bool()  
    [private]
```

Definition at line [158](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.66 int SrP_ClassroomInq::frmClassrromInq::new_IbIID = 0 [private]
```

Definition at line [172](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.67 int SrP_ClassroomInq::frmClassrromInq::notif_I = 0 [private]
```

Definition at line [103](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.68 bool SrP_ClassroomInq::frmClassrromInq::NotifyShowing = new bool()  
    [private]
```

Definition at line [121](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.69 byte SrP_ClassroomInq::frmClassrromInq::NumQuestions = 0  
    [private]
```

Definition at line [109](#) of file [Form1-dyn-ctrls.cs](#).

5.3.4.70 int SrP_ClassroomInq::frmClassroomInq::old_lbIID = 0 [private]

Definition at line 161 of file [Form1-dyn-ctrls.cs](#).

5.3.4.71 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq::origingrouparr
= new System.Drawing.Point(6, -13) [private]

Definition at line 193 of file [Form1-dyn-ctrls.cs](#).

5.3.4.72 System.Windows.Forms.Panel [] SrP_ClassroomInq::frmClassroomInq-
::PanelQuestion_arr = new System.Windows.Forms.Panel[classSize]
[private]

Definition at line 188 of file [Form1-dyn-ctrls.cs](#).

5.3.4.73 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq-
::PanelQuestion_arr_tmp = new System.Drawing.Point(7, 6)
[private]

Definition at line 206 of file [Form1-dyn-ctrls.cs](#).

5.3.4.74 System.Windows.Forms.PictureBox [] SrP_ClassroomInq::frmClassroom-
Inq::picbx_arr = new System.Windows.Forms.PictureBox[classSize]
[private]

Definition at line 184 of file [Form1-dyn-ctrls.cs](#).

5.3.4.75 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq::picbx_arr_tmp
= new System.Drawing.Point(2, 9) [private]

Definition at line 200 of file [Form1-dyn-ctrls.cs](#).

5.3.4.76 System.Windows.Forms.PictureBox [] SrP_ClassroomInq::frmClassroomInq-
::picbx_ConvView_arr = new System.Windows.Forms.PictureBox[classSize]
[private]

Definition at line 185 of file [Form1-dyn-ctrls.cs](#).

5.3.4.77 System.Drawing.Point SrP_ClassroomInq::frmClassroom-
Inq::picbx_ConvView_tmp = new System.Drawing.Point(296, 7)
[private]

Definition at line 201 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.78 string SrP_ClassroomInq::frmClassrromInq::PlainData =
    @".data\StudentInfo.txt" [private]
```

Definition at line [90](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.79 string [] SrP_ClassroomInq::frmClassrromInq::portNames = new string[10]
    [private]
```

Definition at line [91](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.80 bool SrP_ClassroomInq::frmClassrromInq::PrefsClicked = new bool()
    [private]
```

Definition at line [139](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.81 bool SrP_ClassroomInq::frmClassrromInq::PrefsShowing = new bool()
    [private]
```

Definition at line [138](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.82 byte SrP_ClassroomInq::frmClassrromInq::PrefsTimesClicked = 0
    [private]
```

Definition at line [115](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.83 byte SrP_ClassroomInq::frmClassrromInq::PresentStuNum = 250
    [private]
```

Definition at line [85](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.84 Font SrP_ClassroomInq::frmClassrromInq::PrintFont
```

Definition at line [63](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.85 bool [] SrP_ClassroomInq::frmClassrromInq::Q_NameShowing = new
    bool[MaxQuestions] [private]
```

Definition at line [80](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.86 bool [] SrP_ClassroomInq::frmClassrromInq::Q_Replied = new
    bool[MaxQuestions] [private]
```

Definition at line [79](#) of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.87 string [] SrP_ClassroomInq::frmClassrromInq::Q_sender = new  
string[classSize] [private]
```

Definition at line 86 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.88 bool [] SrP_ClassroomInq::frmClassrromInq::Q_status = new  
bool[MaxQuestions] [private]
```

Definition at line 78 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.89 string [] SrP_ClassroomInq::frmClassrromInq::Qs_to_create = new  
string[classSize] [private]
```

Definition at line 72 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.90 bool SrP_ClassroomInq::frmClassrromInq::Qs_to_Make = false  
[private]
```

Definition at line 73 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.91 bool SrP_ClassroomInq::frmClassrromInq::Question_Deleted = false  
[private]
```

Definition at line 164 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.92 int SrP_ClassroomInq::frmClassrromInq::QuizDelID = 255 [private]
```

Definition at line 166 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.93 bool SrP_ClassroomInq::frmClassrromInq::QuizMakerClicked = false  
[private]
```

Definition at line 144 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.94 bool SrP_ClassroomInq::frmClassrromInq::QuizMakerShowing = false  
[private]
```

Definition at line 143 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.95 byte SrP_ClassroomInq::frmClassrromInq::QuizMakerTimesClicked = 0  
[private]
```

Definition at line 145 of file [Form1-dyn-ctrls.cs](#).

5.3.4.96 **bool SrP_ClassroomInq::frmClassrromInq::QuizModeClicked = false**
[private]

Definition at line 147 of file [Form1-dyn-ctrls.cs](#).

5.3.4.97 **bool SrP_ClassroomInq::frmClassrromInq::QuizModeShowing = false**
[private]

Definition at line 146 of file [Form1-dyn-ctrls.cs](#).

5.3.4.98 **byte SrP_ClassroomInq::frmClassrromInq::QuizModeTimesClicked = 0**
[private]

Definition at line 148 of file [Form1-dyn-ctrls.cs](#).

5.3.4.99 **bool SrP_ClassroomInq::frmClassrromInq::QuizQuestionDel = false**
[private]

Definition at line 165 of file [Form1-dyn-ctrls.cs](#).

5.3.4.100 **string SrP_ClassroomInq::frmClassrromInq::QzQuestionDel = ""**
[private]

Definition at line 167 of file [Form1-dyn-ctrls.cs](#).

5.3.4.101 **System.Windows.Forms.Button [] SrP_ClassroomInq::frmClassrrom-
Inq::reply_arr = new System.Windows.Forms.Button[classSize]**
[private]

Definition at line 179 of file [Form1-dyn-ctrls.cs](#).

5.3.4.102 **bool SrP_ClassroomInq::frmClassrromInq::Request_Undo = false**
[private]

Definition at line 168 of file [Form1-dyn-ctrls.cs](#).

5.3.4.103 **string SrP_ClassroomInq::frmClassrromInq::RX_Data = ""** [private]

Definition at line 94 of file [Form1-dyn-ctrls.cs](#).

5.3.4.104 **string SrP_ClassroomInq::frmClassrromInq::SecretKey** [private]

Definition at line 95 of file [Form1-dyn-ctrls.cs](#).

5.3.4.105 **Splash** **SrP_ClassroomInq::frmClassroomInq::SplashScreen** = new
Splash()

Definition at line 67 of file [Form1-dyn-ctrls.cs](#).

5.3.4.106 **bool SrP_ClassroomInq::frmClassroomInq::SplashScreenShowing** = new
bool() [private]

Definition at line 118 of file [Form1-dyn-ctrls.cs](#).

5.3.4.107 **string SrP_ClassroomInq::frmClassroomInq::StateMachine** = "Normal"
[private]

Definition at line 96 of file [Form1-dyn-ctrls.cs](#).

5.3.4.108 **StreamReader SrP_ClassroomInq::frmClassroomInq::Stream2Print**

Definition at line 62 of file [Form1-dyn-ctrls.cs](#).

5.3.4.109 **bool [] SrP_ClassroomInq::frmClassroomInq::StudentPresent** = new
bool[classSize] [private]

Definition at line 83 of file [Form1-dyn-ctrls.cs](#).

5.3.4.110 **string [] SrP_ClassroomInq::frmClassroomInq::Students_ID** = new
string[classSize] [private]

Definition at line 71 of file [Form1-dyn-ctrls.cs](#).

5.3.4.111 **string [] SrP_ClassroomInq::frmClassroomInq::Students_Name** = new
string[classSize] [private]

Definition at line 70 of file [Form1-dyn-ctrls.cs](#).

5.3.4.112 **bool [] SrP_ClassroomInq::frmClassroomInq::StudentVoted** = new
bool[classSize] [private]

Definition at line 84 of file [Form1-dyn-ctrls.cs](#).

5.3.4.113 **bool SrP_ClassroomInq::frmClassroomInq::StuMgmtClicked** = new bool()
[private]

Definition at line 134 of file [Form1-dyn-ctrls.cs](#).

5.3.4.114 **bool SrP_ClassroomInq::frmClassrromInq::StuMgmtShowing = new bool()**
[private]

Definition at line 135 of file [Form1-dyn-ctrls.cs](#).

5.3.4.115 **byte SrP_ClassroomInq::frmClassrromInq::StuMgmtTimesClicked = 0**
[private]

Definition at line 117 of file [Form1-dyn-ctrls.cs](#).

5.3.4.116 **bool SrP_ClassroomInq::frmClassrromInq::sudo_kill = false**
[private]

Definition at line 169 of file [Form1-dyn-ctrls.cs](#).

5.3.4.117 **System.Drawing.Point SrP_ClassroomInq::frmClassrromInq::tempcleararr = new System.Drawing.Point(23, 98)** [private]

Definition at line 197 of file [Form1-dyn-ctrls.cs](#).

5.3.4.118 **System.Drawing.Point SrP_ClassroomInq::frmClassrromInq::tempclosearr = new System.Drawing.Point(129, 98)** [private]

Definition at line 196 of file [Form1-dyn-ctrls.cs](#).

5.3.4.119 **System.Drawing.Point SrP_ClassroomInq::frmClassrromInq::tempgrouparr = new System.Drawing.Point(6, 19)** [private]

Definition at line 194 of file [Form1-dyn-ctrls.cs](#).

5.3.4.120 **System.Drawing.Point SrP_ClassroomInq::frmClassrromInq::tempreplyarr = new System.Drawing.Point(230, 98)** [private]

Definition at line 195 of file [Form1-dyn-ctrls.cs](#).

5.3.4.121 **string SrP_ClassroomInq::frmClassrromInq::tempString = ""**
[private]

Definition at line 92 of file [Form1-dyn-ctrls.cs](#).

5.3.4.122 **string SrP_ClassroomInq::frmClassrromInq::tempString2 = ""**
[private]

Definition at line 93 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.123 bool SrP_ClassroomInq::frmClassrromInq::textbox1WASclicked = new  
bool() [private]
```

Definition at line 155 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.124 bool SrP_ClassroomInq::frmClassrromInq::timeout_ack = false  
[private]
```

Definition at line 160 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.125 byte SrP_ClassroomInq::frmClassrromInq::timesClicked = 0  
[private]
```

Definition at line 113 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.126 byte SrP_ClassroomInq::frmClassrromInq::timesThrough = 0  
[private]
```

Definition at line 171 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.127 System.Windows.Forms.ToolTip [] SrP_ClassroomInq::frmClassrrom-  
Inq::tt_picbxCV_arr = new System.Windows.Forms.ToolTip[classSize]  
[private]
```

Definition at line 186 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.128 System.Windows.Forms.TextBox [] SrP_ClassroomInq::frmClassrrom-  
Inq::txtbx_reply_arr = new System.Windows.Forms.TextBox[classSize]  
[private]
```

Definition at line 182 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.129 System.Drawing.Point SrP_ClassroomInq::frmClassrrom-  
Inq::txtbx_reply_temp = new System.Drawing.Point(23, 32)  
[private]
```

Definition at line 198 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.130 System.Windows.Forms.TextBox [] SrP_ClassroomInq::frmClassrromInq-  
::txtbxQuestion_arr = new System.Windows.Forms.TextBox[classSize]  
[private]
```

Definition at line 191 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.131 System.Drawing.Point SrP_ClassroomInq::frmClassroomInq-
    ::txtbxQuestion_arr_tmp = new System.Drawing.Point(32, 13)
    [private]
```

Definition at line 204 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.132 byte SrP_ClassroomInq::frmClassroomInq::UnreadCount = 0
    [private]
```

Definition at line 110 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.133 bool [] SrP_ClassroomInq::frmClassroomInq::wasQunread = new
    bool[MaxQuestions] [private]
```

Definition at line 81 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.134 bool SrP_ClassroomInq::frmClassroomInq::WeGotData = false
    [private]
```

Definition at line 98 of file [Form1-dyn-ctrls.cs](#).

```
5.3.4.135 byte SrP_ClassroomInq::frmClassroomInq::x = 0 [private]
```

Definition at line 102 of file [Form1-dyn-ctrls.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-
ClassroomInq/[AnimateTimer.cs](#)
- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-
ClassroomInq/[Form1-dyn-ctrls.cs](#)
- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-
ClassroomInq/[themes.cs](#)

5.4 SrP_ClassroomInq::Program Class Reference

Static Private Member Functions

- static void [Main \(\)](#)

The main entry point for the application.

5.4.1 Member Function Documentation

5.4.1.1 static void SrP_ClassroomInq::Program::Main() [static, private]

The main entry point for the application.

Definition at line 31 of file [Program.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-ClassroomInq/[Program.cs](#)

5.5 SrP_ClassroomInq::Properties::Resources Class Reference

A strongly-typed resource class, for looking up localized strings, etc.

Collaboration diagram for SrP_ClassroomInq::Properties::Resources:



Properties

- static internal global::System.Resources.ResourceManager [ResourceManager](#) [get]
Returns the cached ResourceManager instance used by this class.
- static internal global::System.Globalization.CultureInfo [Culture](#) [get, set]
Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.
- static internal System.Drawing.Bitmap [Background](#) [get]
- static internal System.Drawing.Bitmap [bubbles2](#) [get]
- static internal System.Drawing.Bitmap [comment_bubble](#) [get]
- static internal System.Drawing.Bitmap [reply_arrow](#) [get]
- static internal System.Drawing.Bitmap [Splash](#) [get]
- static internal System.Drawing.Bitmap [StuMngmntCtr](#) [get]
- static internal System.Drawing.Bitmap [user_comment](#) [get]

Private Member Functions

- internal Resources ()

Static Private Attributes

- static global::System.Resources.ResourceManager resourceMan
- static global::System.Globalization.CultureInfo resourceCulture

5.5.1 Detailed Description

A strongly-typed resource class, for looking up localized strings, etc.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 internal SrP_ClassroomInq::Properties::Resources::Resources() [private]

Definition at line 32 of file [Resources.Designer.cs](#).

5.5.3 Member Data Documentation

5.5.3.1 global::System.Globalization.CultureInfo SrP_ClassroomInq::Properties::Resources::resourceCulture [static, private]

Definition at line 29 of file [Resources.Designer.cs](#).

5.5.3.2 global::System.Resources.ResourceManager SrP_ClassroomInq::Properties::Resources::resourceMan [static, private]

Definition at line 27 of file [Resources.Designer.cs](#).

5.5.4 Property Documentation

5.5.4.1 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::Background [static, get, private]

Definition at line 63 of file [Resources.Designer.cs](#).

5.5.4.2 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::bubbles2 [static, get, private]

Definition at line 70 of file [Resources.Designer.cs](#).

```
5.5.4.3 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::comment_-  
bubble [static, get, private]
```

Definition at line 77 of file [Resources.Designer.cs](#).

```
5.5.4.4 internal global::System.Globalization.CultureInfo SrP_Classroom-  
Inq::Properties::Resources::Culture [static, get, set,  
private]
```

Overrides the current thread's CurrentUICulture property for all resource lookups using
this strongly typed resource class.

Definition at line 54 of file [Resources.Designer.cs](#).

```
5.5.4.5 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::reply_arrow  
[static, get, private]
```

Definition at line 84 of file [Resources.Designer.cs](#).

```
5.5.4.6 internal global::System.Resources.ResourceManager SrP_Classroom-  
Inq::Properties::Resources::ResourceManager [static, get,  
private]
```

Returns the cached ResourceManager instance used by this class.

Definition at line 39 of file [Resources.Designer.cs](#).

```
5.5.4.7 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::Splash  
[static, get, private]
```

Definition at line 91 of file [Resources.Designer.cs](#).

```
5.5.4.8 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::StuMngmnt-  
Ctr [static, get, private]
```

Definition at line 98 of file [Resources.Designer.cs](#).

```
5.5.4.9 internal System.Drawing.Bitmap SrP_ClassroomInq::Properties::Resources::user_-  
comment [static, get, private]
```

Definition at line 105 of file [Resources.Designer.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-
ClassroomInq/Properties/[Resources.Designer.cs](#)

5.6 SrP_ClassroomInq::Properties::Settings Class Reference

Collaboration diagram for SrP_ClassroomInq::Properties::Settings:

```

graph LR
    A["SrP_ClassroomInq::Properties::Settings"] --- B["defaultInstance"]

```

Properties

- static `Settings Default [get]`
- string `key [get, set]`
- bool `Animations [get, set]`
- bool `SoundRX [get, set]`
- bool `SoundTX [get, set]`
- bool `HoverRead [get, set]`
- bool `ClickRead [get, set]`
- bool `Tooltips [get, set]`
- bool `ShowNames [get, set]`
- bool `Notify [get, set]`
- bool `CtrlHide [get, set]`
- global::System.Drawing.Color `ForeColorTheme [get, set]`
- global::System.Drawing.Color `BackColorTheme [get, set]`
- byte `AnimationSpeed [get, set]`
- decimal `NotifyTimeout [get, set]`
- bool `Splash [get, set]`

Static Private Attributes

- static `Settings defaultInstance = ((Settings)(global::System.Configuration.- ApplicationSettingsBase.Synchronized(new Settings()))))`

5.6.1 Member Data Documentation

5.6.1.1 `Settings SrP_ClassroomInq::Properties::Settings::defaultInstance = ((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new Settings())))) [static, private]`

Definition at line 18 of file `Settings.Designer.cs`.

5.6.2 Property Documentation

5.6.2.1 `bool SrP_ClassroomInq::Properties::Settings::Animations [get, set]`

Definition at line 41 of file [Settings.Designer.cs](#).

5.6.2.2 `byte SrP_ClassroomInq::Properties::Settings::AnimationSpeed [get, set]`

Definition at line 173 of file [Settings.Designer.cs](#).

5.6.2.3 `global::System.Drawing.Color SrP_ClassroomInq::Properties::Settings::BackColorTheme [get, set]`

Definition at line 161 of file [Settings.Designer.cs](#).

5.6.2.4 `bool SrP_ClassroomInq::Properties::Settings::ClickRead [get, set]`

Definition at line 89 of file [Settings.Designer.cs](#).

5.6.2.5 `bool SrP_ClassroomInq::Properties::Settings::CtrlHide [get, set]`

Definition at line 137 of file [Settings.Designer.cs](#).

5.6.2.6 `Settings SrP_ClassroomInq::Properties::Settings::Default [static, get]`

Definition at line 20 of file [Settings.Designer.cs](#).

5.6.2.7 `global::System.Drawing.Color SrP_ClassroomInq::Properties::Settings::ForeColorTheme [get, set]`

Definition at line 149 of file [Settings.Designer.cs](#).

5.6.2.8 `bool SrP_ClassroomInq::Properties::Settings::HoverRead [get, set]`

Definition at line 77 of file [Settings.Designer.cs](#).

5.6.2.9 `string SrP_ClassroomInq::Properties::Settings::key [get, set]`

Definition at line 29 of file [Settings.Designer.cs](#).

5.6.2.10 `bool SrP_ClassroomInq::Properties::Settings::Notify [get, set]`

Definition at line 125 of file [Settings.Designer.cs](#).

5.6.2.11 decimal SrP_ClassroomInq::Properties::Settings::NotifyTimeout [get, set]

Definition at line 185 of file [Settings.Designer.cs](#).

5.6.2.12 bool SrP_ClassroomInq::Properties::Settings::ShowNames [get, set]

Definition at line 113 of file [Settings.Designer.cs](#).

5.6.2.13 bool SrP_ClassroomInq::Properties::Settings::SoundRX [get, set]

Definition at line 53 of file [Settings.Designer.cs](#).

5.6.2.14 bool SrP_ClassroomInq::Properties::Settings::SoundTX [get, set]

Definition at line 65 of file [Settings.Designer.cs](#).

5.6.2.15 bool SrP_ClassroomInq::Properties::Settings::Splash [get, set]

Definition at line 197 of file [Settings.Designer.cs](#).

5.6.2.16 bool SrP_ClassroomInq::Properties::Settings::Tooltips [get, set]

Definition at line 101 of file [Settings.Designer.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-ClassroomInq/Properties/[Settings.Designer.cs](#)

5.7 SrP_ClassroomInq::Splash Class Reference

Public Member Functions

- [Splash \(\)](#)

Private Member Functions

- void [FatherTime_Tick](#) (object sender, EventArgs e)

Private Attributes

- byte [i](#) = 0

5.7.1 Constructor & Destructor Documentation

5.7.1.1 `SrP_ClassroomInq::Splash::Splash()`

Definition at line 31 of file [Splash.cs](#).

5.7.2 Member Function Documentation

5.7.2.1 `void SrP_ClassroomInq::Splash::FatherTime_Tick (object sender, EventArgs e)`
[private]

Definition at line 38 of file [Splash.cs](#).

5.7.3 Member Data Documentation

5.7.3.1 `byte SrP_ClassroomInq::Splash::i = 0` [private]

Definition at line 35 of file [Splash.cs](#).

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

- /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_-ClassroomInq/[Splash.cs](#)

Chapter 6

File Documentation

6.1 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AboutBox1.cs File Reference

Classes

- class [SrP_ClassroomInq::AboutBox_CI](#)

Packages

- package [SrP_ClassroomInq](#)

Variables

- using [System](#)

6.1.1 Variable Documentation

6.1.1.1 using System

Definition at line 18 of file [AboutBox1.cs](#).

6.2 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AboutBox1.cs

```
00001 /*****  
00002 **  
00002 * Copyright (C) 2012 David V. Christensen  
00003 *  
00004 * This program is free software; you can redistribute it and/or
```

```

00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 *****/
00019 using System;
00020 using System.Collections.Generic;
00021 using System.ComponentModel;
00022 using System.Drawing;
00023 using System.Linq;
00024 using System.Reflection;
00025 using System.Windows.Forms;
00026 namespace SrP_ClassroomInq
00027 {
00028     public partial class AboutBox_CI : Form
00029     {
00030         public AboutBox_CI()
00031         {
00032             InitializeComponent();
00033         }
00034
00035         private void AboutBox_CI_Load(object sender, EventArgs e)
00036         {
00037             //Version version =
00038             //Assembly.GetExecutingAssembly().GetName().Version;
00039             //labelVersion.Text = version.ToString();
00040             labelVersion.BackColor = this.BackColor;
00041             labelVersion.ForeColor = this.ForeColor;
00042         }
00043
00044     }
00045 }
00046 }
```

6.3 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AboutBox1.Designer.cs File - Reference

Variables

- namespace [SrP_ClassroomInq](#)

6.3.1 Variable Documentation

6.3.1.1 namespace [SrP_ClassroomInq](#)

Definition at line 19 of file [AboutBox1.Designer.cs](#).

6.4 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AboutBox1.Designer.cs

```
00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 *****/
00019 namespace SrP_ClassroomInq
00020 {
00021     partial class AboutBox_CI
00022     {
00023         private System.ComponentModel.IContainer components = null;
00024
00025         protected override void Dispose(bool disposing)
00026         {
00027             if (disposing && (components != null))
00028             {
00029                 components.Dispose();
00030             }
00031             base.Dispose(disposing);
00032         }
00033
00034         #region Windows Form Designer generated code
00035
00036         private void InitializeComponent()
00037         {
00038             System.ComponentModel.ComponentResourceManager resources = new
00039             System.ComponentModel.ComponentResourceManager(typeof(AboutBox_CI));
00040             this.tableLayoutPanel = new System.Windows.Forms.TableLayoutPanel();
00041 ;
00042             this.logoPictureBox = new System.Windows.Forms.PictureBox();
00043             this.labelProductName = new System.Windows.Forms.Label();
00044             this.labelVersion = new System.Windows.Forms.Label();
00045             this.labelCopyright = new System.Windows.Forms.Label();
00046             this.textBoxDescription = new System.Windows.Forms.TextBox();
00047             this.okButton = new System.Windows.Forms.Button();
00048             this.Motto = new System.Windows.Forms.Label();
00049             this.tableLayoutPanel.SuspendLayout();
00050             ((System.ComponentModel.ISupportInitialize)(this.logoPictureBox)).BeginInit();
00051             BeginInit();
00052             this.SuspendLayout();
00053             // 
00054             // tableLayoutPanel
00055             //
00056             this.tableLayoutPanel.ColumnCount = 2;
00057             this.tableLayoutPanel.ColumnStyles.Add(new System.Windows.Forms.ColumnStyle(
00058                 System.Windows.Forms.SizeType.Percent, 33F));
00059             this.tableLayoutPanel.ColumnStyles.Add(new System.Windows.Forms.ColumnStyle(
00060                 System.Windows.Forms.SizeType.Percent, 67F));
00061             this.tableLayoutPanel.ColumnStyles[0].SizeType = System.Windows.Forms.SizeType.Percent;
00062             this.tableLayoutPanel.ColumnStyles[1].SizeType = System.Windows.Forms.SizeType.Percent;
00063             this.tableLayoutPanel.ColumnStyles[0].Size = 33F;
00064             this.tableLayoutPanel.ColumnStyles[1].Size = 67F;
00065             this.tableLayoutPanel.Controls.Add(this.logoPictureBox, 0, 0);
00066             this.tableLayoutPanel.Controls.Add(this.labelProductName, 1, 0);
00067             this.tableLayoutPanel.Controls.Add(this.labelVersion, 1, 1);
00068             this.tableLayoutPanel.Controls.Add(this.labelCopyright, 1, 2);
00069             this.tableLayoutPanel.Controls.Add(this.textBoxDescription, 1, 4);
00070             this.tableLayoutPanel.Controls.Add(this.okButton, 1, 5);
00071             this.tableLayoutPanel.Controls.Add(this.Motto, 1, 3);
```

```

00072         this.tableLayoutPanel.Dock = System.Windows.Forms.DockStyle.Fill;
00073         this.tableLayoutPanel.Location = new System.Drawing.Point(9, 9);
00074         this.tableLayoutPanel.Name = "tableLayoutPanel";
00075         this.tableLayoutPanel.RowCount = 6;
00076         this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00077             RowStyle(System.Windows.Forms.SizeType.Percent, 10F));
00078             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00079                 RowStyle(System.Windows.Forms.SizeType.Percent, 10F));
00080             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00081                 RowStyle(System.Windows.Forms.SizeType.Percent, 8.668731F));
00082             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00083                 RowStyle(System.Windows.Forms.SizeType.Percent, 11.14551F));
00084             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00085                 RowStyle(System.Windows.Forms.SizeType.Percent, 50F));
00086             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00087                 RowStyle(System.Windows.Forms.SizeType.Percent, 10F));
00088             this.tableLayoutPanel.RowStyles.Add(new System.Windows.Forms.
00089                 RowStyle(System.Windows.Forms.SizeType.Absolute, 20F));
00090             this.tableLayoutPanel.Size = new System.Drawing.Size(445, 323);
00091             this.tableLayoutPanel.TabIndex = 0;
00092             // 
00093             // logoPictureBox
00094             // 
00095             this.logoPictureBox.Dock = System.Windows.Forms.DockStyle.Fill;
00096             this.logoPictureBox.Image = ((System.Drawing.Image)(resources.
00097                 GetObject("logoPictureBox.Image")));
00098             this.logoPictureBox.Location = new System.Drawing.Point(3, 3);
00099             this.logoPictureBox.Name = "logoPictureBox";
00100             this.tableLayoutPanel.SetRowSpan(this.logoPictureBox, 6);
00101             this.logoPictureBox.Size = new System.Drawing.Size(140, 317);
00102             this.logoPictureBox.SizeMode = System.Windows.Forms.
00103                 PictureBoxSizeMode.StretchImage;
00104             this.logoPictureBox.TabIndex = 12;
00105             this.logoPictureBox.TabStop = false;
00106             // 
00107             // labelProductName
00108             // 
00109             this.labelProductName.Dock = System.Windows.Forms.DockStyle.Fill;
00110             this.labelProductName.Location = new System.Drawing.Point(152, 0);
00111             this.labelProductName.Margin = new System.Windows.Forms.Padding(6,
0, 3, 0);
00112             this.labelProductName.MaximumSize = new System.Drawing.Size(0, 17);
00113             this.labelProductName.Name = "labelProductName";
00114             this.labelProductName.Size = new System.Drawing.Size(290, 17);
00115             this.labelProductName.TabIndex = 19;
00116             this.labelProductName.Text = "Classroom Inquisition";
00117             this.labelProductName.TextAlign = System.Drawing.ContentAlignment.
00118                 MiddleLeft;
00119             // 
00120             // labelVersion
00121             // 
00122             this.labelVersion.BackColor = System.Drawing.Color.Black;
00123             this.labelVersion.Dock = System.Windows.Forms.DockStyle.Fill;
00124             this.labelVersion.ForeColor = System.Drawing.Color.DarkGoldenrod;
00125             this.labelVersion.Location = new System.Drawing.Point(152, 32);
00126             this.labelVersion.Margin = new System.Windows.Forms.Padding(6, 0,
3, 0);
00127             this.labelVersion.MaximumSize = new System.Drawing.Size(0, 17);
00128             this.labelVersion.Name = "labelVersion";
00129             this.labelVersion.Size = new System.Drawing.Size(290, 17);
00130             this.labelVersion.Text = "Version: 1.7.3";
00131             this.labelVersion.TextAlign = System.Drawing.ContentAlignment.

```

```
00132         this.labelCopyright.TabIndex = 21;
00133         this.labelCopyright.Text = "Licensed under GPL Version 2";
00134         this.labelCopyright.TextAlign = System.Drawing.ContentAlignment.
00135             MiddleLeft;
00136         // 
00137         // textBoxDescription
00138         this.textBoxDescription.Dock = System.Windows.Forms.DockStyle.Fill;
00139         this.textBoxDescription.Location = new System.Drawing.Point(152, 13
00140             1);
00140         this.textBoxDescription.Margin = new System.Windows.Forms.Padding(6
00141             , 3, 3, 3);
00142         this.textBoxDescription.Multiline = true;
00143         this.textBoxDescription.Name = "textBoxDescription";
00144         this.textBoxDescription.ReadOnly = true;
00145         this.textBoxDescription.Size = new System.Drawing.Size(290, 155);
00146         this.textBoxDescription.TabIndex = 23;
00147         this.textBoxDescription.TabStop = false;
00148         this.textBoxDescription.Text = resources.GetString("textBoxDescription.
00149             Text");
00149         // 
00150         // okButton
00151         this.okButton.Anchor = ((System.Windows.Forms.AnchorStyles)((System.
00151             Windows.Forms.AnchorStyles.Bottom | System.Windows.Forms.AnchorStyles.Right)));
00152         this.okButton.DialogResult = System.Windows.Forms.DialogResult.
00153             Cancel;
00153         this.okButton.Location = new System.Drawing.Point(367, 297);
00154         this.okButton.Name = "okButton";
00155         this.okButton.Size = new System.Drawing.Size(75, 23);
00156         this.okButton.TabIndex = 24;
00157         this.okButton.Text = "&OK";
00158         // 
00159         // Motto
00160         // 
00161         this.Motto.Anchor = ((System.Windows.Forms.AnchorStyles)((System.
00161             Windows.Forms.AnchorStyles.Left | System.Windows.Forms.AnchorStyles.Right)));
00162         this.Motto.AutoSize = true;
00163         this.Motto.FlatStyle = System.Windows.Forms.FlatStyle.System;
00164         this.Motto.Location = new System.Drawing.Point(149, 103);
00165         this.Motto.Name = "Motto";
00166         this.Motto.Size = new System.Drawing.Size(293, 13);
00167         this.Motto.TabIndex = 25;
00168         this.Motto.Text = "\\"Raising Hands is a Thing of the Past.\\"";
00169         this.Motto.TextAlign = System.Drawing.ContentAlignment.MiddleCenter
00170         ;
00170         // 
00171         // AboutBox_CI
00172         // 
00173         this.AcceptButton = this.okButton;
00174         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
00175         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
00176         this.BackColor = System.Drawing.Color.Black;
00177         this.ClientSize = new System.Drawing.Size(463, 341);
00178         this.Controls.Add(this.tableLayoutPanel);
00179         this.ForeColor = System.Drawing.Color.DarkGoldenrod;
00180         this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.
00180             FixedDialog;
00181         this.MaximizeBox = false;
00182         this.MinimizeBox = false;
00183         this.Name = "AboutBox_CI";
00184         this.Padding = new System.Windows.Forms.Padding(9);
00185         this.ShowIcon = false;
00186         this.ShowInTaskbar = false;
00187         this.StartPosition = System.Windows.Forms.FormStartPosition.
00187             CenterParent;
00188         this.Text = "Classroom Inquisition | About";
00189         this.Load += new System.EventHandler(this.AboutBox_CI_Load);
00190         this.tableLayoutPanel.ResumeLayout(false);
00191         this.tableLayoutPanel.PerformLayout();
00192         ((System.ComponentModel.ISupportInitialize)(this.logoPictureBox)).EndInit();
00193         this.ResumeLayout(false);
00194         this.PerformLayout(false);
```

```

00195     }
00196     #endregion
00197
00198     private System.Windows.Forms.TableLayoutPanel tableLayoutPanel;
00199     private System.Windows.Forms.PictureBox logoPictureBox;
00200     private System.Windows.Forms.Label labelProductName;
00201     private System.Windows.Forms.Label labelVersion;
00202     private System.Windows.Forms.Label labelCopyright;
00203     private System.Windows.Forms.TextBox textBoxDescription;
00204     private System.Windows.Forms.Button okButton;
00205     private System.Windows.Forms.Label Motto;
00206
00207 }
00208

```

6.5 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AnimateTimer.cs File Reference

Classes

- class [SrP_ClassroomInq::frmClassromInq](#)

Packages

- package [SrP_ClassroomInq](#)

Variables

- using [System](#)

6.5.1 Variable Documentation

6.5.1.1 using System

Definition at line 18 of file [AnimateTimer.cs](#).

6.6 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/AnimateTimer.cs

```

00001 ****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

```

```

00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301,
00017 * USA.
00017 ****
00018 using System;
00019 using System.Collections.Generic;
00020 using System.ComponentModel;
00021 using System.Data;
00022 using System.Drawing;
00023 using System.Drawing.Printing;
00024 using System.Linq;
00025 using System.Media;
00026 using System.Text;
00027 using System.Windows.Forms;
00028 using System.IO;
00029 using System.Security;
00030 using System.Security.Cryptography;
00031 using System.Runtime.InteropServices;
00032
00033 namespace SrP_ClassroomInq
00034 {
00035     ****
00036     *
00037     * Classroom Inquisition Teacher Application
00038     * Programmer: David Christensen
00039     * Bug Tracking: https://github.com/dchriste/Classroom-Inquisition/issues
00040     *
00041     ****
00042     */
00042     public partial class frmClassrromInq : Form
00043     {
00044         /*This method does all the animation for the application*/
00045         private void timer_Tick(object sender, EventArgs e)
00046         {
00047             #region BroadCast Panel
00048
00049             if ((!BrdcstShowing) && (textbox1WASclicked == true) && (pn1BrdCst.
00050                 Location.Y < 41))// if not visible
00051                 {
00052                     if (!chkbxLameMode.Checked) //animations
00053                     {
00054                         if (i < 10)
00055                         {
00056                             if (i == 0)
00057                             {
00058                                 pn1BrdCst.Hide();
00059                                 pn1BrdCst.BringToFront();
00060                                 menuStrip1.BringToFront();
00061                                 pn1BrdCst.Show();
00062                                 aPanelIsMoving = true;
00063                             }
00064                             pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00065                                         Location.Y + 5,
00066                                         pn1BrdCst.Size.Width, pn1BrdCst.
00067                                         Size.Height);
00068                         i++;
00069                     }
00070                     else if (i < 20)
00071                     {
00072                         pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00073                                         Location.Y + 2,
00074                                         pn1BrdCst.Size.Width, pn1BrdCst.
00075                                         Size.Height);
00076                         i++;
00077                     }
00078                     /* */ /*uncomment this for bounce effect
00079                     else if (i < 25)
00080                     {
00081                         pn1BrdCst.SetBounds(pn1BrdCst.Location.X,
00082                                         pn1BrdCst.Location.Y + 3,
00083                                         pn1BrdCst.Size.Width, pn1BrdCst.
00084                                         Size.Height);
00085                     }
00086                 }
00087             }
00088         }
00089     }
00090 }
```

```

00076                               pn1BrdCst.Size.Width,
00077                               i++;
00078                           }
00079                           else if (i < 30)
00080                           {
00081                               pn1BrdCst.SetBounds(pn1BrdCst.Location.X,
00082                               pn1BrdCst.Location.Y - 3,
00083                               pn1BrdCst.Size.Width,
00084                               i++);
00085                           }*/
00086                           else
00087                           {
00088                               i = 0;
00089                               if (!NotifyShowing && !DesireNotify)
00090                               {
00091                                   timer.Enabled = false;
00092                                   BrdcstShowing = true;
00093                                   aPanelIsMoving = false;
00094                                   textBox1.Clear();
00095                                   this.Text = " Classroom Inquisition | Broadcast" + (
00096                                       UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00097                               }
00098                           else //no animations
00099                           {
00100                               //jump straight to whatever position..
00101                               pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00102                               Location.Y + 70,
00103                               pn1BrdCst.Size.Width, pn1BrdCst.Size.
00104                               Height);
00105                               if (!NotifyShowing && !DesireNotify)
00106                               {
00107                                   timer.Enabled = false;
00108                                   BrdcstShowing = true;
00109                                   aPanelIsMoving = false;
00110                                   textBox1.Clear();
00111                                   textBox1WASClicked = false;
00112                                   this.Text = " Classroom Inquisition | Broadcast";
00113                               }
00114                           }
00115                           if ((BrdcstShowing) && (textBox1WASClicked == true) && (pn1BrdCst.
00116                               Location.Y > -46))// if visible, destroy
00117                           {
00118                               if (!chkbxLameMode.Checked)
00119                               {
00120                                   if (i < 10)
00121                                   {
00122                                       if (i == 0)
00123                                       {
00124                                           aPanelIsMoving = true;
00125                                       pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00126                               Location.Y - 2,
00127                               pn1BrdCst.Size.Width, pn1BrdCst.
00128                               Size.Height);
00129                               i++;
00130                           }
00131                           else if (i < 20)
00132                           {
00133                               pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00134                               Location.Y - 5,
00135                               pn1BrdCst.Size.Width, pn1BrdCst.
00136                               Size.Height);
00137                               i++;
00138                           }
00139                           else
00140                           {
00141                               i = 0;

```

```

00139
00140
00141
00142
00143
00144
00145
00146
00147
00148
00149
00150
00151
00152
00153
00154
00155
00156
00157
00158
00159
00160
00161
00162
00163
00164
00165
00166
00167
00168
00169
00170
00171
00172
00173
00174
00175
00176
00177
00178
00179
00180
00181
00182
00183
00184
00185
00186
    Mode now
00187
00188
00189
00190
00191
00192
00193
00194
    now
00195
00196
00197
00198
00199
00200
00201
00202
00203
00204
00205
00206
00207
00208
00209

        if (DesirePrefs)
        {
            if (!PrefsShowing && !DMPPanelShowing)
            {
                PrefsClicked = true; //show prefs now
                DesirePrefs = false;
            }
            else if (DMPPanelShowing && !PrefsShowing)
            {
                DMClicked = true; //hide
                //leave desire prefs on..
            }
            else
            {
                DesirePrefs = false;
            }
        }

        else if (DesireDM)
        {
            if (!DMPPanelShowing && !PrefsShowing)
            {
                DMClicked = true; //show dm now
                DesireDM = false;
            }
            else if (PrefsShowing && !DMPPanelShowing)
            {
                PrefsClicked = true; //hide this
                //leaves desire DM on...
            }
            else
            {
                DesireDM = false;
            }
        }

        else if (DesireFAQ)
        {
            if (!FAQShowing)
            {
                FAQClicked = true; //show FAQ now
            }
            DesireFAQ = false;
        }

        else if (DesireAttendance)
        {
            if (!AttendanceShowing)
            {
                AttendanceClicked = true; //show Attendance
            }
            DesireAttendance = false;
        }

        else if (DesireClassVote)
        {
            if (!ClassVoteShowing)
            {
                ClassVoteClicked = true; //show ClassVote Mode
            }
            DesireClassVote = false;
        }

        else if (DesireQuizMd)
        {
            if (!QuizModeShowing)
            {
                QuizModeClicked = true; //show Quiz Mode now
            }
            DesireQuizMd = false;
        }

        else if (DesireQzMkr)
        {
            if (!QuizMakerShowing)
            {

```

```

00210                         QuizMakerClicked = true; //show QuizMaker now
00211
00212                     }
00213             DesireQzMkr = false;
00214         }
00215     else if (DesireStuMgmt)
00216     {
00217         if (!StuMgmtShowing)
00218         {
00219             StuMgmtClicked = true; //show Student
00220             Management now
00221         }
00222         DesireStuMgmt = false;
00223     }
00224     else if (DesireConvView)
00225     {
00226         if (!ConvViewShowing)
00227         {
00228             ConvViewClicked = true; //show Conversation
00229             Viewer now
00230         }
00231         DesireConvView = false;
00232     }
00233     else
00234     {
00235         if (!NotifyShowing && !DesireNotify)
00236         {
00237             timer.Enabled = false;
00238         }
00239         aPanelIsMoving = false;
00240         textbox1WASClicked = false;
00241         textbox1WASClicked = false;
00242         BrdstShowing = false;
00243         if (FAQShowing)
00244         {
00245             this.Text = " Classroom Inquisition | FAQ" + (
00246                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00247             PanelFAQ.Focus();
00248         }
00249         else if (ConvViewShowing)
00250         {
00251             this.Text = " Classroom Inquisition |
00252             Conversation Viewer" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00253             PanelConvView.Focus();
00254         }
00255         else if (StuMgmtShowing)
00256         {
00257             this.Text = " Classroom Inquisition | Student
00258             Managment" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00259             PanelStudents.Focus();
00260         }
00261         else if (PrefsShowing)
00262         {
00263             this.Text = " Classroom Inquisition | Prefs" + (
00264                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00265             PanelPrefs.Focus();
00266         }
00267         else if (DMPPanelShowing)
00268         {
00269             this.Text = " Classroom Inquisition | Direct Msg"
00270             + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00271             txtbxDM.Focus();
00272         }
00273     else //no animations
00274     {

```

```

00275             pn1BrdCst.SetBounds(pn1BrdCst.Location.X, pn1BrdCst.
00276                                         Location.Y - 70,
00277                                         pn1BrdCst.Size.Width, pn1BrdCst.Size.
00278                                         Height);
00279
00280         if (DesirePrefs)
00281     {
00282             if (!PrefsShowing && !DMPanelShowing)
00283         {
00284                 PrefsClicked = true; //show prefs now
00285                 DesirePrefs = false;
00286             }
00287             else if (DMPanelShowing && !PrefsShowing)
00288             {
00289                 DMClicked = true; //hide
00290                 //leave desire prefs on..
00291             }
00292             else
00293             {
00294                 DesirePrefs = false;
00295             }
00296         }
00297         else if (DesireDM)
00298     {
00299             if (!DMPanelShowing && !PrefsShowing)
00300         {
00301                 DMClicked = true; //show dm now
00302                 DesireDM = false;
00303             }
00304             else if (PrefsShowing && !DMPanelShowing)
00305         {
00306                 PrefsClicked = true; //hide this
00307                 //leaves desire DM on...
00308             }
00309             else
00310             {
00311                 DesireDM = false;
00312             }
00313         }
00314         else if (DesireFAQ)
00315     {
00316             if (!FAQShowing)
00317         {
00318                 FAQClicked = true; //show FAQ now
00319             }
00320             DesireFAQ = false;
00321         }
00322         else if (DesireAttendance)
00323     {
00324             if (!AttendanceShowing)
00325         {
00326                 AttendanceClicked = true; //show Attendance Mode
00327                 now
00328             }
00329             DesireAttendance = false;
00330         }
00331         else if (DesireClassVote)
00332     {
00333             if (!ClassVoteShowing)
00334         {
00335                 ClassVoteClicked = true; //show ClassVote Mode now
00336             }
00337             DesireClassVote = false;
00338         }
00339         else if (DesireQuizMd)
00340     {
00341             if (!QuizModeShowing)
00342         {
00343                 QuizModeClicked = true; //show Quiz Mode now
00344             }
00345             DesireQuizMd = false;
00346         }
00347     }

```

```

00345        else if (DesireQzMkr)
00346        {
00347            if (!QuizMakerShowing)
00348            {
00349                QuizMakerClicked = true; //show QuizMaker now
00350            }
00351            DesireQzMkr = false;
00352        }
00353        else if (DesireStuMgmt)
00354        {
00355            if (!StuMgmtShowing)
00356            {
00357                StuMgmtClicked = true; //show Student Management
00358            }
00359            DesireStuMgmt = false;
00360        }
00361        else if (DesireConvView)
00362        {
00363            if (!ConvViewShowing)
00364            {
00365                ConvViewClicked = true; //show Conversation Viewer
00366            }
00367            DesireConvView = false;
00368        }
00369        else
00370        {
00371            if (!NotifyShowing && !DesireNotify)
00372            {
00373                timer.Enabled = false;
00374            }
00375            aPanelIsMoving = false;
00376            textBox1WASClicked = false;
00377            textBox1WASClicked = false;
00378            BrdcstShowing = false;
00379            if (FAQShowing)
00380            {
00381                this.Text = " Classroom Inquisition | FAQ" + (
00382                    UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00383                PanelFAQ.Focus();
00384            }
00385            else if (ConvViewShowing)
00386            {
00387                this.Text = " Classroom Inquisition | Conversation
00388                    Viewer" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00389                PanelConvView.Focus();
00390            }
00391            else if (StuMgmtShowing)
00392            {
00393                this.Text = " Classroom Inquisition | Student
00394                    Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00395                PanelStudents.Focus();
00396            }
00397            else if (PrefsShowing)
00398            {
00399                this.Text = " Classroom Inquisition | Prefs" + (
00400                    UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00401                PanelPrefs.Focus();
00402            }
00403            else if (DMPPanelShowing)
00404            {
00405                this.Text = " Classroom Inquisition | Direct Msg" + (
00406                    UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00407                txtbxDM.Focus();
00408            }
00409            else
00410            {
00411                this.Text = " Classroom Inquisition | Home" + (
00412                    UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00413                this.Focus();
00414            }
00415        }

```

```

00411         }
00412
00413     #endregion
00414
00415     #region GrpBxRPL_Clicked
00416     if (grpbxFeed.Controls.Count > 0) //allows textbox click with no
00417     questions
00418     {
00419         if ((group_arr[	lbl_ID].Height < 141) && (grpbxRPL_WASclicked ==
00420             true))// if not visible
00421         {
00422             Point tmp = new Point();
00423             if (!chkbxLameMode.Checked)
00424             {
00425                 if (k < 3) //use groupboxID..
00426                 {
00427                     if (k == 0)
00428                     {
00429                         aPanelIsMoving = true;
00430                         group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].
00431                             Location.X, group_arr[lbl_ID].Location.Y,
00432                             group_arr[lbl_ID].Size.
00433                             Width, group_arr[lbl_ID].Size.Height + 2);
00434                         if (lbl_ID != 0)
00435                         {
00436                             for (int i = lbl_ID - 1; i >= 0; i--) // to
00437                             achieve LIFO
00438                             {
00439                                 tmp.X = group_arr[i].Location.X;
00440                                 tmp.Y = group_arr[i].Location.Y + 2;
00441                                 group_arr[i].Location = tmp;
00442                             }
00443                         }
00444                         else if ((k > 2) && (k < 36))
00445                         {
00446                             group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].
00447                             Location.X, group_arr[lbl_ID].Location.Y,
00448                             group_arr[lbl_ID].Size.
00449                             Width, group_arr[lbl_ID].Size.Height + 3);
00450                             //fire ctrl resize event
00451                             if (lbl_ID != 0)
00452                             {
00453                                 for (int i = lbl_ID - 1; i >= 0; i--) // to
00454                                 achieve LIFO
00455                                 {
00456                                     tmp.X = group_arr[i].Location.X;
00457                                     tmp.Y = group_arr[i].Location.Y + 3;
00458                                     group_arr[i].Location = tmp;
00459                                 }
00460                             }
00461                             k = 0;
00462                             if (!NotifyShowing && !DesireNotify)
00463                             {
00464                                 timer.Enabled = false;
00465                             }
00466                             aPanelIsMoving = false;
00467                             grpbxRPL_WASclicked = false;
00468                             txtbx_reply_arr[lbl_ID].Focus(); //places cursor in
00469                             question reply box on open
00470                             }
00471                         }
00472                         else //no animations
00473                         {
00474                             group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].Location.
X, group_arr[lbl_ID].Location.Y,
group_arr[lbl_ID].Size.

```

```

        Width, group_arr[lbl_ID].Size.Height + 105);
00475            if (lbl_ID != 0)
00476            {
00477                for (int i = lbl_ID - 1; i >= 0; i--) // to achieve
00478                    LIFO
00479                {
00480                    tmp.X = group_arr[i].Location.X;
00481                    tmp.Y = group_arr[i].Location.Y + 105;
00482                    group_arr[i].Location = tmp;
00483                }
00484            }
00485            if (!NotifyShowing && !DesireNotify)
00486            {
00487                timer.Enabled = false;
00488            }
00489            aPanelIsMoving = false;
00490            grpbxRPL_WASClicked = false;
00491            txtbx_reply_arr[lbl_ID].Focus(); //places cursor in
00492            question reply box on open
00493        }
00494    }
00495    #endregion
00496
00497    #region Destroy Reply
00498    if (grpbxFeed.Controls.Count > 0) //allows textbox click with no
00499    questions
00500    {
00501        if ((group_arr[lbl_ID].Height > 31) && (btnCLS_WASClicked ==
00502            true))
00503        {
00504            Point tmp = new Point(); // for dynamic moving of all
00505            controls below the clicked one
00506            if (!chkbxLameMode.Checked)
00507            {
00508                if (j < 3)
00509                {
00510                    if (j == 0)
00511                    {
00512                        aPanelIsMoving = true;
00513                        group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].
00514                            Location.X, group_arr[lbl_ID].Location.Y,
00515                                group_arr[lbl_ID].Size.
00516                                Width, group_arr[lbl_ID].Size.Height - 2);
00517                        if (lbl_ID != 0)
00518                        {
00519                            for (int x = lbl_ID - 1; x >= 0; x--)
00520                            {
00521                                tmp.X = group_arr[x].Location.X;
00522                                tmp.Y = group_arr[x].Location.Y - 2;
00523                                group_arr[x].Location = tmp;
00524                            }
00525                        }
00526                    else if ((j > 2) && (j < 36)) //was 19 with 3,6 pixel
00527                        increments
00528                    {
00529                        group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].
00530                            Location.X, group_arr[lbl_ID].Location.Y,
00531                                group_arr[lbl_ID].Size.
00532                                Width, group_arr[lbl_ID].Size.Height - 3);
00533                        if (lbl_ID != 0)
00534                        {
00535                            for (int x = lbl_ID - 1; x >= 0; x--)
00536                            {
00537                                tmp.X = group_arr[x].Location.X;
00538                                tmp.Y = group_arr[x].Location.Y - 3;
00539                                group_arr[x].Location = tmp;
00540                            }
00541                        }
00542                    }
00543                }
00544            }
00545        }
00546    }
00547}

```

```

00538         j++;
00539     }
00540     else
00541     {
00542         j = 0;
00543         btnCLS_WASclicked = false;
00544         aPanelIsMoving = false;
00545         if (!DesireID)
00546         {
00547             if (!NotifyShowing && !DesireNotify)
00548             {
00549                 timer.Enabled = false;
00550             }
00551             timesClicked = 0; //reset
00552         }
00553     else
00554     {
00555         lbl_ID = new_lblID; //open other question
00556         grpbxRPL_WASclicked = true;
00557         DesireID = false;
00558     }
00559 }
00560 }
00561 else //no animations :(
00562 {
00563     group_arr[lbl_ID].SetBounds(group_arr[lbl_ID].Location.
X, group_arr[lbl_ID].Location.Y,
00564                                     group_arr[lbl_ID].Size.
Width, group_arr[lbl_ID].Size.Height - 105);
00565     if (lbl_ID != 0)
00566     {
00567         for (int x = lbl_ID - 1; x >= 0; x--)
00568         {
00569             tmp.X = group_arr[x].Location.X;
00570             tmp.Y = group_arr[x].Location.Y - 105;
00571             group_arr[x].Location = tmp;
00572         }
00573     }
00574     j = 0;
00575     btnCLS_WASclicked = false;
00576     aPanelIsMoving = false;
00577     if (!DesireID)
00578     {
00579         if (!NotifyShowing && !DesireNotify)
00580         {
00581             timer.Enabled = false;
00582         }
00583         timesClicked = 0; //reset
00584     }
00585     else
00586     {
00587         lbl_ID = new_lblID;
00588         grpbxRPL_WASclicked = true;
00589         DesireID = false;
00590     }
00591 }
00592 }
00593 }
00594 #endregion
00595
00596 #region Delete Question
00597 //keeps things from being upset when Del_ID is initialized to 255
00598 // (so as not to delete the first question that comes in...)
00599 if (DeleteQuestion)
00600 {
00601     if ((group_arr[Del_ID].Height > 0) && (DeleteQuestion == true))
00602     // if open and clicked
00603     {
00604         Point tmp = new Point(); // for dynamic moving of all
00605         controls below the clicked one
00606         if (!chkbxLameMode.Checked)
00607         {
00608             aPanelIsMoving = true;
00609             group_arr[Del_ID].SetBounds(group_arr[Del_ID].Location.

```

```

        X, group_arr[Del_ID].Location.Y,
00608     Width, group_arr[Del_ID].Size.Height - 5);
00609             if (Del_ID != 0)
00610             {
00611                 for (int x = Del_ID - 1; x >= 0; x--)
00612                 {
00613                     tmp.X = group_arr[x].Location.X;
00614                     tmp.Y = group_arr[x].Location.Y - 5;
00615                     group_arr[x].Location = tmp;
00616                 }
00617             }
00618         }
00619     else
00620     {
00621         for (int x = Del_ID - 1; x >= 0; x--)
00622         {
00623             tmp.X = group_arr[x].Location.X;
00624             tmp.Y = group_arr[x].Location.Y - group_arr[Del_ID]
00625             .Size.Height;
00626             group_arr[x].Location = tmp;
00627         }
00628         group_arr[Del_ID].SetBounds(group_arr[Del_ID].Location.
00629             X, group_arr[Del_ID].Location.Y,
00630             group_arr[Del_ID].Size.
00631             Width, 0);
00632     }
00633 }
00634 else if (DeleteQuestion)
00635 {
00636     btnCLS_WASClicked = false; //needed if you delete an open
00637     question
00638     timesClicked = 0; //reset
00639     x = 0;
00640     DeleteQuestion = false;
00641     Question_Deleted = true;
00642     aPanelIsMoving = false;
00643     if (!NotifyShowing && !DesireNotify)
00644     {
00645         timer.Enabled = false;
00646         group_arr[Del_ID].Hide(); //real deleting would mess up a
00647         lot of numbers in status arrays etc..
00648         trayICON.BalloonTipTitle = "Undo Delete by";
00649         trayICON.BalloonTipText = "Right click Feed (grey) & choose
00650         Undo Delete";
00651         trayICON.ShowBalloonTip(1200);
00652     }
00653 }
00654 #endregion
00655
00656 #region Undo Delete Question
00657 //keeps things from being upset when Del_ID is initialized to 255
00658 //((so as not to delete the first question that comes in...))
00659 if (Request_Undo && Question_Deleted)
00660 {
00661     if ((group_arr[Del_ID].Height < 32) && (Request_Undo == true))
00662     // if open and clicked
00663     {
00664         Point tmp = new Point(); // for dynamic moving of all
00665         controls below the clicked one
00666         if (!chkbxLameMode.Checked)
00667         {
00668             aPanelIsMoving = true;
00669             group_arr[Del_ID].SetBounds(group_arr[Del_ID].Location.
00670                 X, group_arr[Del_ID].Location.Y,
00671                 group_arr[Del_ID].Size.
00672                 Width, group_arr[Del_ID].Size.Height + 4);
00673             if (Del_ID != 0)
00674             {
00675                 if (group_arr[Del_ID].Height < 32)
00676                 {
00677                     for (int x = Del_ID - 1; x >= 0; x--)
00678                     {
00679

```

```

00670                         tmp.X = group_arr[x].Location.X;
00671                         tmp.Y = group_arr[x].Location.Y + 4;
00672                         group_arr[x].Location = tmp;
00673                     }
00674                 }
00675             }
00676         {
00677             for (int x = Del_ID - 1; x >= 0; x--)
00678             {
00679                 tmp.X = group_arr[x].Location.X;
00680                 tmp.Y = group_arr[x].Location.Y + 6; // looks better with 6 than 4 for last run
00681                 group_arr[x].Location = tmp;
00682             }
00683         }
00684     }
00685     else
00686     {
00687         if (Del_ID != 0)
00688         {
00689             group_arr[Del_ID].SetBounds(group_arr[Del_ID].
00690             Location.X, group_arr[Del_ID].Location.Y,
00691             group_arr[Del_ID].Size.
00692             Width, 32);
00693             for (int x = Del_ID - 1; x >= 0; x--)
00694             {
00695                 tmp.X = group_arr[x].Location.X;
00696                 tmp.Y = group_arr[x].Location.Y + 32;
00697                 group_arr[x].Location = tmp;
00698             }
00699         }
00700     }
00701     else if (Request_Undo)
00702     {
00703         Question_Deleted = false;
00704         Request_Undo = false;
00705         aPanelIsMoving = false;
00706         x = 0;
00707         if (!NotifyShowing && !DesireNotify)
00708         {
00709             timer.Enabled = false;
00710         }
00711     }
00712 }
00713 #endregion
00714
00715 #region MoveCtrlsDown
00716 if ((NEW_grpbx == true))
00717 {
00718     Point TEMP = new Point();
00719     if (k < 3) //use groupboxID..
00720     {
00721         if (k == 0)
00722         {
00723             aPanelIsMoving = true;
00724         }
00725         for (int i = NumQuestions - 1; i >= 0; i--) // to achieve
00726             LIFO
00727         {
00728             TEMP.X = group_arr[i].Location.X;
00729             TEMP.Y = group_arr[i].Location.Y + 2;
00730             group_arr[i].Location = TEMP;
00731         }
00732         k++;
00733     }
00734     else if ((k > 2) && (k < 12)) //was 19 with 3,6 pixel
00735     increments
00736     {
00737         LIFO
00738         {

```

```

00738             TEMP.X = group_arr[i].Location.X;
00739             TEMP.Y = group_arr[i].Location.Y + 3;
00740             group_arr[i].Location = TEMP;
00741         }
00742
00743
00744         k++;
00745     }
00746     else
00747     {
00748         k = 0;
00749         aPanelIsMoving = false;
00750         if (!NotifyShowing && !DesireNotify)
00751         {
00752             timer.Enabled = false;
00753         }
00754         NEW_grpbx = false;
00755     }
00756 }
00757
00758 #endregion
00759
00760 #region DMPanel Animate
00761 if ((DirectMsgPanel.Location.X < 16) && (DMclicked == true)
00762     && (DMtimesClicked == 0) && (DMPanelShowing == false))
00763 {
00764     if (!chkbxLameMode.Checked)
00765     {
00766         if (x < 10)
00767         {
00768             if (x == 0)
00769             {
00770                 aPanelIsMoving = true;
00771             }
00772             DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X + 16
00773 , DirectMsgPanel.Location.Y,
00774             DirectMsgPanel.Size.Width,
00775             DirectMsgPanel.Size.Height);
00776             grpbxFeed.SetBounds(grpbxFed.Location.X + 16,
00777             grpbxFeed.Location.Y,
00778             grpbxFeed.Size.Width, grpbxFeed.
00779             Size.Height);
00780             x++;
00781         }
00782         else if (x < 20)
00783         {
00784             DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X + 15
00785 , DirectMsgPanel.Location.Y,
00786             DirectMsgPanel.Size.Width,
00787             DirectMsgPanel.Size.Height);
00788             grpbxFeed.SetBounds(grpbxFed.Location.X + 15,
00789             grpbxFeed.Location.Y,
00790             grpbxFeed.Size.Width, grpbxFeed.
00791             Size.Height);
00792             x++;
00793         }
00794     else
00795     {
00796         x = 0;
00797         DMPanelShowing = true;
00798         DMclicked = false;
00799         aPanelIsMoving = false;

```

```
00800     DMtimesClicked = 1;
00801     if (!NotifyShowing && !DesireNotify)
00802     {
00803         timer.Enabled = false;
00804     }
00805     DirectMsgPanel.Focus();
00806     this.Text = " Classroom Inquisition | Direct Msg" + (
00807     UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00808     }
00809     else //no animations, boo!
00810     {
00811         DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X + 410,
00812         DirectMsgPanel.Location.Y,
00813             DirectMsgPanel.Size.Width,
00814             DirectMsgPanel.Size.Height);
00815         grpbxFeed.SetBounds(grpbxFed.Location.X + 410, grpbxFeed.
00816         Location.Y,
00817             grpbxFeed.Size.Width, grpbxFeed.Size.
00818         Height);
00819         DMPanelShowing = true;
00820         DMclicked = false;
00821         aPanelIsMoving = false;
00822         DMtimesClicked = 1;
00823         if (!NotifyShowing && !DesireNotify)
00824         {
00825             timer.Enabled = false;
00826         }
00827         else if ((DirectMsgPanel.Location.X > -396) && (DMclicked == true)
00828                 && (DMtimesClicked == 1) && (DMPanelShowing == true)
00829         )
00830         {
00831             if (!chkbxLameMode.Checked)
00832             {
00833                 if (x < 10)
00834                 {
00835                     if (x == 0)
00836                     {
00837                         aPanelIsMoving = true;
00838                     }
00839                     DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X - 10
00840 , DirectMsgPanel.Location.Y,
00841             DirectMsgPanel.Size.Width,
00842             DirectMsgPanel.Size.Height);
00843             grpbxFeed.SetBounds(grpbxFed.Location.X - 10,
00844             grpbxFeed.Location.Y,
00845             grpbxFeed.Size.Width, grpbxFeed.
00846             Size.Height);
00847             x++;
00848             }
00849             else if (x < 20)
00850             {
00851                 DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X - 15
00852 , DirectMsgPanel.Location.Y,
00853             DirectMsgPanel.Size.Width,
00854             DirectMsgPanel.Size.Height);
00855             grpbxFeed.SetBounds(grpbxFed.Location.X - 15,
00856             grpbxFeed.Location.Y,
00857             grpbxFeed.Size.Width, grpbxFeed.
00858             Size.Height);
00859             x++;
00860             }
00861             else if (x < 30)
00862             {
00863                 DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X - 16
00864 , DirectMsgPanel.Location.Y,
00865             DirectMsgPanel.Size.Width,
00866             grpbxFeed.SetBounds(grpbxFed.Location.X - 16,
```

```

        grpbxFeed.Location.Y,
00857                         grpbxFeed.Size.Width, grpbxFeed.
00858             Size.Height);
00859             x++;
00860         }
00861     {
00862         x = 0;
00863         DMPanelShowing = false;
00864         DMclicked = false;
00865         aPanelIsMoving = false;
00866         DMtimesClicked = 0;
00867         if (DesirePrefs)
00868     {
00869             if (!PrefsShowing)
00870             {
00871                 PrefsClicked = true; //show prefs now
00872             }
00873             DesirePrefs = false;
00874         }
00875     else
00876     {
00877         if (!NotifyShowing && !DesireNotify)
00878         {
00879             timer.Enabled = false;
00880         }
00881         this.Text = " Classroom Inquisition | Home" + (
00882             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00883         }
00884     }
00885     else //no animations, boo!
00886     {
00887         DirectMsgPanel.SetBounds(DirectMsgPanel.Location.X - 410,
00888             DirectMsgPanel.Location.Y,
00889             DirectMsgPanel.Size.Width,
00890             DirectMsgPanel.Size.Height);
00891         grpbxFeed.SetBounds(grpbxFed.Location.X - 410, grpbxFeed.
00892             Location.Y,
00893             grpbxFeed.Size.Width, grpbxFeed.Size.
00894             Height);
00895         DMPanelShowing = false;
00896         DMclicked = false;
00897         aPanelIsMoving = false;
00898         DMtimesClicked = 0;
00899         if (DesirePrefs)
00900     {
00901             if (!PrefsShowing)
00902             {
00903                 PrefsClicked = true; //show prefs now
00904             }
00905             DesirePrefs = false;
00906         }
00907     else
00908     {
00909         if (!NotifyShowing && !DesireNotify)
00910         {
00911             timer.Enabled = false;
00912         }
00913         this.Text = " Classroom Inquisition | Home" + (
00914             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00915     }
00916 #endregion
00917 #region Prefs Animate
00918 if ((PanelPrefs.Location.X > 14) && (PrefsClicked == true)
00919     && (PrefsTimesClicked == 0) && (PrefsShowing == false))
00920 {
00921     if (!chkbxLameMode.Checked)

```

```

00923             {
00924                 if (x < 10)
00925                 {
00926                     if (x == 0)
00927                     {
00928                         aPanelIsMoving = true;
00929                     }
00930                     PanelPrefs.SetBounds(PanelPrefs.Location.X - 16,
00931                                         PanelPrefs.Location.Y,
00932                                         PanelPrefs.Size.Width, PanelPrefs.
00933                                         Size.Height);
00932                     grpbxFeed.SetBounds(grpbxFed.Location.X - 16,
00933                                         grpbxFeed.Location.Y,
00934                                         grpbxFeed.Size.Width, grpbxFeed.
00935                                         Size.Height);
00934                     x++;
00935                 }
00936                 else if (x < 20)
00937                 {
00938                     PanelPrefs.SetBounds(PanelPrefs.Location.X - 12,
00939                                         PanelPrefs.Location.Y,
00940                                         PanelPrefs.Size.Width, PanelPrefs.
00941                                         Size.Height);
00940                     grpbxFeed.SetBounds(grpbxFed.Location.X - 12,
00942                                         grpbxFeed.Location.Y,
00943                                         grpbxFeed.Size.Width, grpbxFeed.
00944                                         Size.Height);
00942                     x++;
00943                 }
00944                 else if (x < 30)
00945                 {
00946                     PanelPrefs.SetBounds(PanelPrefs.Location.X - 10,
00947                                         PanelPrefs.Location.Y,
00948                                         PanelPrefs.Size.Width, PanelPrefs.
00949                                         Size.Height);
00949                     grpbxFeed.SetBounds(grpbxFed.Location.X - 10,
00950                                         grpbxFeed.Location.Y,
00951                                         grpbxFeed.Size.Width, grpbxFeed.
00952                                         Size.Height);
00950                     x++;
00951                 }
00952             else
00953             {
00954                 x = 0;
00955                 PrefsShowing = true;
00956                 PrefsClicked = false;
00957                 aPanelIsMoving = false;
00958                 PrefsTimesClicked = 1;
00959                 if (!NotifyShowing && !DesireNotify)
00960                 {
00961                     timer.Enabled = false;
00962                 }
00963                 else
00964                 {
00965                     PanelNotify.BringToFront();
00966                 }
00967                 PanelPrefs.Focus();
00968                 this.Text = " Classroom Inquisition | Prefs" + (
00969                     UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00970                 }
00971             else //no animations
00972             {
00973                 PanelPrefs.SetBounds(PanelPrefs.Location.X - 380,
00974                                         PanelPrefs.Location.Y,
00975                                         PanelPrefs.Size.Width, PanelPrefs.Size.
00976                                         .Height);
00975                 grpbxFeed.SetBounds(grpbxFed.Location.X - 380, grpbxFeed.
00977                                         Location.Y,
00978                                         grpbxFeed.Size.Width, grpbxFeed.Size.
00979                                         Height);
00976                 PrefsShowing = true;
00977                 PrefsClicked = false;
00978                 aPanelIsMoving = false;

```

```

00980     PrefsTimesClicked = 1;
00981     if (!NotifyShowing && !DesireNotify)
00982     {
00983         timer.Enabled = false;
00984     }
00985     else
00986     {
00987         PanelNotify.BringToFront();
00988     }
00989     PanelPrefs.Focus();
00990     this.Text = "Classroom Inquisition | Prefs" + (
00991         UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
00992     }
00993     else if ((PanelPrefs.Location.X < 396) && (PrefsClicked == true)
00994             && (PrefsTimesClicked == 1) && (PrefsShowing == true))
00995     {
00996         if (!chkbxLameMode.Checked)
00997         {
00998             if (x < 10)
00999             {
01000                 if (x == 0)
01001                 {
01002                     aPanelIsMoving = true;
01003                 }
01004                 PanelPrefs.SetBounds(PanelPrefs.Location.X + 10,
01005                                     PanelPrefs.Size.Width, PanelPrefs.
01006                                     Size.Height);
01006                 grpbxFeed.SetBounds(grpbxFed.Location.X + 10,
01007                                     grpbxFeed.Location.Y,
01007                                     grpbxFeed.Size.Width, grpbxFeed.
01008                                     Size.Height);
01008                 x++;
01009             }
01010             else if (x < 20)
01011             {
01012                 PanelPrefs.SetBounds(PanelPrefs.Location.X + 12,
01013                                     PanelPrefs.Location.Y,
01013                                     PanelPrefs.Size.Width, PanelPrefs.
01014                                     Size.Height);
01014                 grpbxFeed.SetBounds(grpbxFed.Location.X + 12,
01015                                     grpbxFeed.Location.Y,
01015                                     grpbxFeed.Size.Width, grpbxFeed.
01016                                     Size.Height);
01016                 x++;
01017             }
01018             else if (x < 30)
01019             {
01020                 PanelPrefs.SetBounds(PanelPrefs.Location.X + 16,
01021                                     PanelPrefs.Location.Y,
01021                                     PanelPrefs.Size.Width, PanelPrefs.
01022                                     Size.Height);
01022                 grpbxFeed.SetBounds(grpbxFed.Location.X + 16,
01023                                     grpbxFeed.Location.Y,
01023                                     grpbxFeed.Size.Width, grpbxFeed.
01024                                     Size.Height);
01024                 x++;
01025             }
01026         else
01027         {
01028             x = 0;
01029             PrefsShowing = false;
01030             PrefsClicked = false;
01031             PrefsTimesClicked = 0;
01032             aPanelIsMoving = false;
01033             if (DesireDM)
01034             {
01035                 if (!DMPanelShowing)
01036                 {
01037                     DMclicked = true; //show DM now
01038                 }
01039             DesireDM = false;
01040         }

```

```

01041         else if (DesireBrdcst)
01042             {
01043                 textbox1WASclicked = true; // show broadcast box
01044                 DesireBrdcst = false;
01045             }
01046         else
01047             {
01048                 if (!NotifyShowing && !DesireNotify)
01049                 {
01050                     timer.Enabled = false;//all done
01051                 }
01052                 this.Text = " Classroom Inquisition | Home" + (
01053                     UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01054                 }
01055             }
01056         else //no animations, boo!
01057             {
01058                 PanelPrefs.SetBounds(PanelPrefs.Location.X + 380,
01059                     PanelPrefs.Location.Y,
01060                         PanelPrefs.Size.Width, PanelPrefs.Size
01061                     .Height);
01060                 grpbxFeed.SetBounds(grpbxFed.Location.X + 380, grpbxFeed.
01061                     Location.Y,
01062                         grpbxFeed.Size.Width, grpbxFeed.Size.
01062                     Height);
01063                 PrefsShowing = false;
01064                 PrefsClicked = false;
01065                 aPanelIsMoving = false;
01066                 PrefsTimesClicked = 0;
01067                 //SavePrefs(); //when hiding the panel save the preferences
01068                 this.Text = " Classroom Inquisition | Home" + (UnreadCount
01067 > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01069                 if (DesireDM)
01070                     {
01071                         if (!DMPanelShowing)
01072                         {
01073                             DMclicked = true; //show DM now
01074                         }
01075                         DesiredDM = false;
01076                     }
01077                 else if (DesireBrdcst)
01078                     {
01079                         textbox1WASclicked = true; // show broadcast box
01080                         DesireBrdcst = false;
01081                     }
01082                 else
01083                     {
01084                         if (!NotifyShowing && !DesireNotify)
01085                         {
01086                             timer.Enabled = false;//all done
01087                         }
01088                     }
01089             }
01090
01091
01092 #endregion
01093
01094 #region FAQ Animate
01095
01096     if ((PanelFAQ.Location.Y < 26) && (FAQClicked == true)
01097         && (FAQTimesClicked == 0) && (FAQShowing == false))
01098     {
01099         if (!chkbxLameMode.Checked)
01100         {
01101             if (x < 10)
01102                 {
01103                     if (x == 0)
01104                     {
01105                         aPanelIsMoving = true;
01106                         PanelFAQ.BringToFront();
01107                         menuStrip1.BringToFront();
01108                     }
01109                 }
01110             }
01111         }
01112     }
01113
01114     if (x == 10)
01115     {
01116         aPanelIsMoving = false;
01117         PanelFAQ.BringToFront();
01118         menuStrip1.BringToFront();
01119     }
01120
01121     if (x == 20)
01122     {
01123         aPanelIsMoving = false;
01124         PanelFAQ.BringToFront();
01125         menuStrip1.BringToFront();
01126     }
01127
01128     if (x == 30)
01129     {
01130         aPanelIsMoving = false;
01131         PanelFAQ.BringToFront();
01132         menuStrip1.BringToFront();
01133     }
01134
01135     if (x == 40)
01136     {
01137         aPanelIsMoving = false;
01138         PanelFAQ.BringToFront();
01139         menuStrip1.BringToFront();
01140     }
01141
01142     if (x == 50)
01143     {
01144         aPanelIsMoving = false;
01145         PanelFAQ.BringToFront();
01146         menuStrip1.BringToFront();
01147     }
01148
01149     if (x == 60)
01150     {
01151         aPanelIsMoving = false;
01152         PanelFAQ.BringToFront();
01153         menuStrip1.BringToFront();
01154     }
01155
01156     if (x == 70)
01157     {
01158         aPanelIsMoving = false;
01159         PanelFAQ.BringToFront();
01160         menuStrip1.BringToFront();
01161     }
01162
01163     if (x == 80)
01164     {
01165         aPanelIsMoving = false;
01166         PanelFAQ.BringToFront();
01167         menuStrip1.BringToFront();
01168     }
01169
01170     if (x == 90)
01171     {
01172         aPanelIsMoving = false;
01173         PanelFAQ.BringToFront();
01174         menuStrip1.BringToFront();
01175     }
01176
01177     if (x == 100)
01178     {
01179         aPanelIsMoving = false;
01180         PanelFAQ.BringToFront();
01181         menuStrip1.BringToFront();
01182     }
01183
01184     if (x == 110)
01185     {
01186         aPanelIsMoving = false;
01187         PanelFAQ.BringToFront();
01188         menuStrip1.BringToFront();
01189     }
01190
01191     if (x == 120)
01192     {
01193         aPanelIsMoving = false;
01194         PanelFAQ.BringToFront();
01195         menuStrip1.BringToFront();
01196     }
01197
01198     if (x == 130)
01199     {
01200         aPanelIsMoving = false;
01201         PanelFAQ.BringToFront();
01202         menuStrip1.BringToFront();
01203     }
01204
01205     if (x == 140)
01206     {
01207         aPanelIsMoving = false;
01208         PanelFAQ.BringToFront();
01209         menuStrip1.BringToFront();
01210     }
01211
01212     if (x == 150)
01213     {
01214         aPanelIsMoving = false;
01215         PanelFAQ.BringToFront();
01216         menuStrip1.BringToFront();
01217     }
01218
01219     if (x == 160)
01220     {
01221         aPanelIsMoving = false;
01222         PanelFAQ.BringToFront();
01223         menuStrip1.BringToFront();
01224     }
01225
01226     if (x == 170)
01227     {
01228         aPanelIsMoving = false;
01229         PanelFAQ.BringToFront();
01230         menuStrip1.BringToFront();
01231     }
01232
01233     if (x == 180)
01234     {
01235         aPanelIsMoving = false;
01236         PanelFAQ.BringToFront();
01237         menuStrip1.BringToFront();
01238     }
01239
01240     if (x == 190)
01241     {
01242         aPanelIsMoving = false;
01243         PanelFAQ.BringToFront();
01244         menuStrip1.BringToFront();
01245     }
01246
01247     if (x == 200)
01248     {
01249         aPanelIsMoving = false;
01250         PanelFAQ.BringToFront();
01251         menuStrip1.BringToFront();
01252     }
01253
01254     if (x == 210)
01255     {
01256         aPanelIsMoving = false;
01257         PanelFAQ.BringToFront();
01258         menuStrip1.BringToFront();
01259     }
01260
01261     if (x == 220)
01262     {
01263         aPanelIsMoving = false;
01264         PanelFAQ.BringToFront();
01265         menuStrip1.BringToFront();
01266     }
01267
01268     if (x == 230)
01269     {
01270         aPanelIsMoving = false;
01271         PanelFAQ.BringToFront();
01272         menuStrip1.BringToFront();
01273     }
01274
01275     if (x == 240)
01276     {
01277         aPanelIsMoving = false;
01278         PanelFAQ.BringToFront();
01279         menuStrip1.BringToFront();
01280     }
01281
01282     if (x == 250)
01283     {
01284         aPanelIsMoving = false;
01285         PanelFAQ.BringToFront();
01286         menuStrip1.BringToFront();
01287     }
01288
01289     if (x == 260)
01290     {
01291         aPanelIsMoving = false;
01292         PanelFAQ.BringToFront();
01293         menuStrip1.BringToFront();
01294     }
01295
01296     if (x == 270)
01297     {
01298         aPanelIsMoving = false;
01299         PanelFAQ.BringToFront();
01300         menuStrip1.BringToFront();
01301     }
01302
01303     if (x == 280)
01304     {
01305         aPanelIsMoving = false;
01306         PanelFAQ.BringToFront();
01307         menuStrip1.BringToFront();
01308     }
01309
01310     if (x == 290)
01311     {
01312         aPanelIsMoving = false;
01313         PanelFAQ.BringToFront();
01314         menuStrip1.BringToFront();
01315     }
01316
01317     if (x == 300)
01318     {
01319         aPanelIsMoving = false;
01320         PanelFAQ.BringToFront();
01321         menuStrip1.BringToFront();
01322     }
01323
01324     if (x == 310)
01325     {
01326         aPanelIsMoving = false;
01327         PanelFAQ.BringToFront();
01328         menuStrip1.BringToFront();
01329     }
01330
01331     if (x == 320)
01332     {
01333         aPanelIsMoving = false;
01334         PanelFAQ.BringToFront();
01335         menuStrip1.BringToFront();
01336     }
01337
01338     if (x == 330)
01339     {
01340         aPanelIsMoving = false;
01341         PanelFAQ.BringToFront();
01342         menuStrip1.BringToFront();
01343     }
01344
01345     if (x == 340)
01346     {
01347         aPanelIsMoving = false;
01348         PanelFAQ.BringToFront();
01349         menuStrip1.BringToFront();
01350     }
01351
01352     if (x == 350)
01353     {
01354         aPanelIsMoving = false;
01355         PanelFAQ.BringToFront();
01356         menuStrip1.BringToFront();
01357     }
01358
01359     if (x == 360)
01360     {
01361         aPanelIsMoving = false;
01362         PanelFAQ.BringToFront();
01363         menuStrip1.BringToFront();
01364     }
01365
01366     if (x == 370)
01367     {
01368         aPanelIsMoving = false;
01369         PanelFAQ.BringToFront();
01370         menuStrip1.BringToFront();
01371     }
01372
01373     if (x == 380)
01374     {
01375         aPanelIsMoving = false;
01376         PanelFAQ.BringToFront();
01377         menuStrip1.BringToFront();
01378     }
01379
01380     if (x == 390)
01381     {
01382         aPanelIsMoving = false;
01383         PanelFAQ.BringToFront();
01384         menuStrip1.BringToFront();
01385     }
01386
01387     if (x == 400)
01388     {
01389         aPanelIsMoving = false;
01390         PanelFAQ.BringToFront();
01391         menuStrip1.BringToFront();
01392     }
01393
01394     if (x == 410)
01395     {
01396         aPanelIsMoving = false;
01397         PanelFAQ.BringToFront();
01398         menuStrip1.BringToFront();
01399     }
01400
01401     if (x == 420)
01402     {
01403         aPanelIsMoving = false;
01404         PanelFAQ.BringToFront();
01405         menuStrip1.BringToFront();
01406     }
01407
01408     if (x == 430)
01409     {
01410         aPanelIsMoving = false;
01411         PanelFAQ.BringToFront();
01412         menuStrip1.BringToFront();
01413     }
01414
01415     if (x == 440)
01416     {
01417         aPanelIsMoving = false;
01418         PanelFAQ.BringToFront();
01419         menuStrip1.BringToFront();
01420     }
01421
01422     if (x == 450)
01423     {
01424         aPanelIsMoving = false;
01425         PanelFAQ.BringToFront();
01426         menuStrip1.BringToFront();
01427     }
01428
01429     if (x == 460)
01430     {
01431         aPanelIsMoving = false;
01432         PanelFAQ.BringToFront();
01433         menuStrip1.BringToFront();
01434     }
01435
01436     if (x == 470)
01437     {
01438         aPanelIsMoving = false;
01439         PanelFAQ.BringToFront();
01440         menuStrip1.BringToFront();
01441     }
01442
01443     if (x == 480)
01444     {
01445         aPanelIsMoving = false;
01446         PanelFAQ.BringToFront();
01447         menuStrip1.BringToFront();
01448     }
01449
01450     if (x == 490)
01451     {
01452         aPanelIsMoving = false;
01453         PanelFAQ.BringToFront();
01454         menuStrip1.BringToFront();
01455     }
01456
01457     if (x == 500)
01458     {
01459         aPanelIsMoving = false;
01460         PanelFAQ.BringToFront();
01461         menuStrip1.BringToFront();
01462     }
01463
01464     if (x == 510)
01465     {
01466         aPanelIsMoving = false;
01467         PanelFAQ.BringToFront();
01468         menuStrip1.BringToFront();
01469     }
01470
01471     if (x == 520)
01472     {
01473         aPanelIsMoving = false;
01474         PanelFAQ.BringToFront();
01475         menuStrip1.BringToFront();
01476     }
01477
01478     if (x == 530)
01479     {
01480         aPanelIsMoving = false;
01481         PanelFAQ.BringToFront();
01482         menuStrip1.BringToFront();
01483     }
01484
01485     if (x == 540)
01486     {
01487         aPanelIsMoving = false;
01488         PanelFAQ.BringToFront();
01489         menuStrip1.BringToFront();
01490     }
01491
01492     if (x == 550)
01493     {
01494         aPanelIsMoving = false;
01495         PanelFAQ.BringToFront();
01496         menuStrip1.BringToFront();
01497     }
01498
01499     if (x == 560)
01500     {
01501         aPanelIsMoving = false;
01502         PanelFAQ.BringToFront();
01503         menuStrip1.BringToFront();
01504     }
01505
01506     if (x == 570)
01507     {
01508         aPanelIsMoving = false;
01509         PanelFAQ.BringToFront();
01510         menuStrip1.BringToFront();
01511     }
01512
01513     if (x == 580)
01514     {
01515         aPanelIsMoving = false;
01516         PanelFAQ.BringToFront();
01517         menuStrip1.BringToFront();
01518     }
01519
01520     if (x == 590)
01521     {
01522         aPanelIsMoving = false;
01523         PanelFAQ.BringToFront();
01524         menuStrip1.BringToFront();
01525     }
01526
01527     if (x == 600)
01528     {
01529         aPanelIsMoving = false;
01530         PanelFAQ.BringToFront();
01531         menuStrip1.BringToFront();
01532     }
01533
01534     if (x == 610)
01535     {
01536         aPanelIsMoving = false;
01537         PanelFAQ.BringToFront();
01538         menuStrip1.BringToFront();
01539     }
01540
01541     if (x == 620)
01542     {
01543         aPanelIsMoving = false;
01544         PanelFAQ.BringToFront();
01545         menuStrip1.BringToFront();
01546     }
01547
01548     if (x == 630)
01549     {
01550         aPanelIsMoving = false;
01551         PanelFAQ.BringToFront();
01552         menuStrip1.BringToFront();
01553     }
01554
01555     if (x == 640)
01556     {
01557         aPanelIsMoving = false;
01558         PanelFAQ.BringToFront();
01559         menuStrip1.BringToFront();
01560     }
01561
01562     if (x == 650)
01563     {
01564         aPanelIsMoving = false;
01565         PanelFAQ.BringToFront();
01566         menuStrip1.BringToFront();
01567     }
01568
01569     if (x == 660)
01570     {
01571         aPanelIsMoving = false;
01572         PanelFAQ.BringToFront();
01573         menuStrip1.BringToFront();
01574     }
01575
01576     if (x == 670)
01577     {
01578         aPanelIsMoving = false;
01579         PanelFAQ.BringToFront();
01580         menuStrip1.BringToFront();
01581     }
01582
01583     if (x == 680)
01584     {
01585         aPanelIsMoving = false;
01586         PanelFAQ.BringToFront();
01587         menuStrip1.BringToFront();
01588     }
01589
01590     if (x == 690)
01591     {
01592         aPanelIsMoving = false;
01593         PanelFAQ.BringToFront();
01594         menuStrip1.BringToFront();
01595     }
01596
01597     if (x == 700)
01598     {
01599         aPanelIsMoving = false;
01600         PanelFAQ.BringToFront();
01601         menuStrip1.BringToFront();
01602     }
01603
01604     if (x == 710)
01605     {
01606         aPanelIsMoving = false;
01607         PanelFAQ.BringToFront();
01608         menuStrip1.BringToFront();
01609     }
01610
01611     if (x == 720)
01612     {
01613         aPanelIsMoving = false;
01614         PanelFAQ.BringToFront();
01615         menuStrip1.BringToFront();
01616     }
01617
01618     if (x == 730)
01619     {
01620         aPanelIsMoving = false;
01621         PanelFAQ.BringToFront();
01622         menuStrip1.BringToFront();
01623     }
01624
01625     if (x == 740)
01626     {
01627         aPanelIsMoving = false;
01628         PanelFAQ.BringToFront();
01629         menuStrip1.BringToFront();
01630     }
01631
01632     if (x == 750)
01633     {
01634         aPanelIsMoving = false;
01635         PanelFAQ.BringToFront();
01636         menuStrip1.BringToFront();
01637     }
01638
01639     if (x == 760)
01640     {
01641         aPanelIsMoving = false;
01642         PanelFAQ.BringToFront();
01643         menuStrip1.BringToFront();
01644     }
01645
01646     if (x == 770)
01647     {
01648         aPanelIsMoving = false;
01649         PanelFAQ.BringToFront();
01650         menuStrip1.BringToFront();
01651     }
01652
01653     if (x == 780)
01654     {
01655         aPanelIsMoving = false;
01656         PanelFAQ.BringToFront();
01657         menuStrip1.BringToFront();
01658     }
01659
01660     if (x == 790)
01661     {
01662         aPanelIsMoving = false;
01663         PanelFAQ.BringToFront();
01664         menuStrip1.BringToFront();
01665     }
01666
01667     if (x == 800)
01668     {
01669         aPanelIsMoving = false;
01670         PanelFAQ.BringToFront();
01671         menuStrip1.BringToFront();
01672     }
01673
01674     if (x == 810)
01675     {
01676         aPanelIsMoving = false;
01677         PanelFAQ.BringToFront();
01678         menuStrip1.BringToFront();
01679     }
01680
01681     if (x == 820)
01682     {
01683         aPanelIsMoving = false;
01684         PanelFAQ.BringToFront();
01685         menuStrip1.BringToFront();
01686     }
01687
01688     if (x == 830)
01689     {
01690         aPanelIsMoving = false;
01691         PanelFAQ.BringToFront();
01692         menuStrip1.BringToFront();
01693     }
01694
01695     if (x == 840)
01696     {
01697         aPanelIsMoving = false;
01698         PanelFAQ.BringToFront();
01699         menuStrip1.BringToFront();
01700     }
01701
01702     if (x == 850)
01703     {
01704         aPanelIsMoving = false;
01705         PanelFAQ.BringToFront();
01706         menuStrip1.BringToFront();
01707     }
01708
01709     if (x == 860)
01710     {
01711         aPanelIsMoving = false;
01712         PanelFAQ.BringToFront();
01713         menuStrip1.BringToFront();
01714     }
01715
01716     if (x == 870)
01717     {
01718         aPanelIsMoving = false;
01719         PanelFAQ.BringToFront();
01720         menuStrip1.BringToFront();
01721     }
01722
01723     if (x == 880)
01724     {
01725         aPanelIsMoving = false;
01726         PanelFAQ.BringToFront();
01727         menuStrip1.BringToFront();
01728     }
01729
01730     if (x == 890)
01731     {
01732         aPanelIsMoving = false;
01733         PanelFAQ.BringToFront();
01734         menuStrip1.BringToFront();
01735     }
01736
01737     if (x == 900)
01738     {
01739         aPanelIsMoving = false;
01740         PanelFAQ.BringToFront();
01741         menuStrip1.BringToFront();
01742     }
01743
01744     if (x == 910)
01745     {
01746         aPanelIsMoving = false;
01747         PanelFAQ.BringToFront();
01748         menuStrip1.BringToFront();
01749     }
01750
01751     if (x == 920)
01752     {
01753         aPanelIsMoving = false;
01754         PanelFAQ.BringToFront();
01755         menuStrip1.BringToFront();
01756     }
01757
01758     if (x == 930)
01759     {
01760         aPanelIsMoving = false;
01761         PanelFAQ.BringToFront();
01762         menuStrip1.BringToFront();
01763     }
01764
01765     if (x == 940)
01766     {
01767         aPanelIsMoving = false;
01768         PanelFAQ.BringToFront();
01769         menuStrip1.BringToFront();
01770     }
01771
01772     if (x == 950)
01773     {
01774         aPanelIsMoving = false;
01775         PanelFAQ.BringToFront();
01776         menuStrip1.BringToFront();
01777     }
01778
01779     if (x == 960)
01780     {
01781         aPanelIsMoving = false;
01782         PanelFAQ.BringToFront();
01783         menuStrip1.BringToFront();
01784     }
01785
01786     if (x == 970)
01787     {
01788         aPanelIsMoving = false;
01789         PanelFAQ.BringToFront();
01790         menuStrip1.BringToFront();
01791     }
01792
01793     if (x == 980)
01794     {
01795         aPanelIsMoving = false;
01796         PanelFAQ.BringToFront();
01797         menuStrip1.BringToFront();
01798     }
01799
01800     if (x == 990)
01801     {
01802         aPanelIsMoving = false;
01803         PanelFAQ.BringToFront();
01804         menuStrip1.BringToFront();
01805     }
01806
01807     if (x == 1000)
01808     {
01809         aPanelIsMoving = false;
01810         PanelFAQ.BringToFront();
01811         menuStrip1.BringToFront();
01812     }
01813
01814     if (x == 1010)
01815     {
01816         aPanelIsMoving = false;
01817         PanelFAQ.BringToFront();
01818         menuStrip1.BringToFront();
01819     }
01820
01821     if (x == 1020)
01822     {
01823         aPanelIsMoving = false;
01824         PanelFAQ.BringToFront();
01825         menuStrip1.BringToFront();
01826     }
01827
01828     if (x == 1030)
01829     {
01830         aPanelIsMoving = false;
01831         PanelFAQ.BringToFront();
01832         menuStrip1.BringToFront();
01833     }
01834
01835     if (x == 1040)
01836     {
01837         aPanelIsMoving = false;
01838         PanelFAQ.BringToFront();
01839         menuStrip1.BringToFront();
01840     }
01841
01842     if (x == 1050)
01843     {
01844         aPanelIsMoving = false;
01845         PanelFAQ.BringToFront();
01846         menuStrip1.BringToFront();
01847     }
01848
01849     if (x == 1060)
01850     {
01851         aPanelIsMoving = false;
01852         PanelFAQ.BringToFront();
01853         menuStrip1.BringToFront();
01854     }
01855
01856     if (x == 1070)
01857     {
01858         aPanelIsMoving = false;
01859         PanelFAQ.BringToFront();
01860         menuStrip1.BringToFront();
01861     }
01862
01863     if (x == 1080)
01864     {
01865         aPanelIsMoving = false;
01866         PanelFAQ.BringToFront();
01867         menuStrip1.BringToFront();
01868     }
01869
01870     if (x == 1090)
01871     {
01872         aPanelIsMoving = false;
01873         PanelFAQ.BringToFront();
01874         menuStrip1.BringToFront();
01875     }
01876
01877     if (x == 1100)
01878     {
01879         aPanelIsMoving = false;
01880         PanelFAQ.BringToFront();
01881         menuStrip1.BringToFront();
01882     }
01883
01884     if (x == 1110)
01885     {
01886         aPanelIsMoving = false;
01887         PanelFAQ.BringToFront();
01888         menuStrip1.BringToFront();
01889     }
01890
01891     if (x == 1120)
01892     {
01893         aPanelIsMoving = false;
01894         PanelFAQ.BringToFront();
01895         menuStrip1.BringToFront();
01896     }
01897
01898     if (x == 1130)
01899     {
01900         aPanelIsMoving = false;
01901         PanelFAQ.BringToFront();
01902         menuStrip1.BringToFront();
01903     }
01904
01905     if (x == 1140)
01906     {
01907         aPanelIsMoving = false;
01908         PanelFAQ.BringToFront();
01909         menuStrip1.BringToFront();
01910     }
01911
01912     if (x == 1150)
01913     {
01914         aPanelIsMoving = false;
01915         PanelFAQ.BringToFront();
01916         menuStrip1.BringToFront();
01917     }
01918
01919     if (x == 1160)
01920     {
01921         aPanelIsMoving = false;
01922         PanelFAQ.BringToFront();
01923         menuStrip1.BringToFront();
01924     }
01925
01926     if (x == 1170)
01927     {
01928         aPanelIsMoving = false;
01929         PanelFAQ.BringToFront();
01930         menuStrip1.BringToFront();
01931     }
01932
01933     if (x == 1180)
01934     {
01935         aPanelIsMoving = false;
01936         PanelFAQ.BringToFront();
01937         menuStrip1.BringToFront();
01938     }
01939
01940     if (x == 1190)
01941     {
01942         aPanelIsMoving = false;
01943         PanelFAQ.BringToFront();
01944         menuStrip1.BringToFront();
01945     }
01946
01947     if (x == 1200)
01948     {
01949         aPanelIsMoving = false;
01950         PanelFAQ.BringToFront();
01951         menuStrip1.BringToFront();
01952     }
01953
01954     if (x == 1210)
01955     {
01956         aPanelIsMoving = false;
01957         PanelFAQ.BringToFront();
01958         menuStrip1.BringToFront();
01959     }
01960
01961     if (x == 1220)
01962     {
01963         aPanelIsMoving = false;
01964         PanelFAQ.BringToFront();
01965         menuStrip1.BringToFront();
01966     }
01967
01968     if (x == 1230)
01969     {
01970         aPanelIsMoving = false;
01971         PanelFAQ.BringToFront();
01972         menuStrip1.BringToFront();
01973     }
01974
01975     if (x == 1240)
01976     {
01977         aPanelIsMoving = false;
01978         PanelFAQ.BringToFront();
01979         menuStrip1.BringToFront();
01980     }
01981
01982     if (x == 1250)
01983     {
01984         aPanelIsMoving = false;
01985         PanelFAQ.BringToFront();
01986         menuStrip1.BringToFront();
01987     }
01988
01989     if (x == 1260)
01990     {
01991         aPanelIsMoving = false;
01992         PanelFAQ.BringToFront();
01993         menuStrip1.BringToFront();
01994     }
01995
01996     if (x == 1270)
01997     {
01998         aPanelIsMoving = false;
01999         PanelFAQ.BringToFront();
02000         menuStrip1.BringToFront();
02001     }
02002
02003     if (x == 1280)
02004     {
02005         aPanelIsMoving = false;
02006         PanelFAQ.BringToFront();
02007         menuStrip1.BringToFront();
02008     }
02009
02010     if (x == 1290)
02011     {
02012         aPanelIsMoving = false;
02013         PanelFAQ.BringToFront();
02014         menuStrip1.BringToFront();
02015     }
02016
02017     if (x == 1300)
02018     {
02019         aPanelIsMoving = false;
02020         PanelFAQ.BringToFront();
02021         menuStrip1.BringToFront();
02022     }
02023
02024     if (x == 1310)
02025     {
02026         aPanelIsMoving = false;
02027         PanelFAQ.BringToFront();
02028         menuStrip1.BringToFront();
02029     }
02030
02031     if (x == 1320)
02032     {
02033         aPanelIsMoving = false;
02034         PanelFAQ.BringToFront();
02035         menuStrip1.BringToFront();
02036     }
02037
02038     if (x == 1330)
02039     {
02040         aPanelIsMoving = false;
02041         PanelFAQ.BringToFront();
02042         menuStrip1.BringToFront();
02043     }
02044
02045     if (x == 1340)
02046     {
02047         aPanelIsMoving = false;
02048         PanelFAQ.BringToFront();
02049         menuStrip1.BringToFront();
02050     }
02051
02052     if (x == 1350)
02053     {
02054         aPanelIsMoving = false;
02055         PanelFAQ.BringToFront();
02056         menuStrip1.BringToFront();
02057     }
02058
02059     if (x == 1360)
02060     {
02061         aPanelIsMoving = false;
02062         PanelFAQ.BringToFront();
02063         menuStrip1.BringToFront();
02064     }
02065
02066     if (x == 1370)
02067     {
02068         aPanelIsMoving = false;
02069         PanelFAQ.BringToFront();
02070         menuStrip1.BringToFront();
02071     }
02072
02073     if (x == 1380)
02074     {
02075         aPanelIsMoving = false;
02076         PanelFAQ.BringToFront();
02077         menuStrip1.BringToFront();
02078     }
02079
02080     if (x == 1390)
02081     {
02082         aPanelIsMoving = false;

```

```

01109             PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01110                                         Location.Y + 25,
01111                                         PanelFAQ.Size.Width, PanelFAQ.Size.
01112                                         Height);
01113             x++;
01114         }
01115         else if (x < 20)
01116         {
01117             PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01118                                         Location.Y + 15,
01119                                         PanelFAQ.Size.Width, PanelFAQ.Size.
01120                                         Height);
01121             x++;
01122         }
01123         else if (x < 30)
01124         {
01125             PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01126                                         Location.Y + 12,
01127                                         PanelFAQ.Size.Width, PanelFAQ.Size.
01128                                         Height);
01129             x++;
01130         }
01131     }
01132     else
01133     {
01134         x = 0;
01135         FAQShowing = true;
01136         FAQClicked = false;
01137         aPanelIsMoving = false;
01138         FAQTimesClicked = 1;
01139         if (!NotifyShowing && !DesireNotify)
01140         {
01141             timer.Enabled = false;
01142             PanelFAQ.Focus();
01143             this.Text = " Classroom Inquisition | FAQ" +
01144                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "";
01145         }
01146     }
01147     else //no animations
01148     {
01149         PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.Location.Y
01150             + 520,
01151             PanelFAQ.Size.Width, PanelFAQ.Size.
01152             Height);
01153             FAQShowing = true;
01154             FAQClicked = false;
01155             aPanelIsMoving = false;
01156             FAQTimesClicked = 1;
01157             if (!NotifyShowing && !DesireNotify)
01158             {
01159                 timer.Enabled = false;
01160             }
01161             PanelFAQ.Focus();
01162             this.Text = " Classroom Inquisition | FAQ" +
01163                 (UnreadCount
01164                 > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01165             }
01166             else if ((PanelFAQ.Location.Y > -496) && (FAQClicked == true)
01167                     && (FAQTimesClicked == 1) && (FAQShowing == true))
01168             {
01169                 if (!chkbxLameMode.Checked)
01170                 {
01171                     if (x < 10)
01172                     {
01173                         if (x == 0)
01174                         {
01175                             aPanelIsMoving = true;
01176                         }
01177                         PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01178                                         Location.Y - 12,
01179                                         PanelFAQ.Size.Width, PanelFAQ.Size.
01180                                         Height);
01181                         x++;
01182                     }
01183                 }
01184             }
01185         }
01186     }
01187 }
```

```

01171             else if (x < 20)
01172             {
01173                 PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01174                     Location.Y - 15,
01175                     PanelFAQ.Size.Width, PanelFAQ.Size.
01176                     Height);
01177                     x++;
01178             }
01179             else if (x < 30)
01180             {
01181                 PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.
01182                     Location.Y - 25,
01183                     PanelFAQ.Size.Width, PanelFAQ.Size.
01184                     Height);
01185                     x++;
01186             }
01187             else
01188             {
01189                 x = 0;
01190                 FAQShowing = false;
01191                 FAQClicked = false;
01192                 aPanelIsMoving = false;
01193                 FAQTimesClicked = 0;
01194                 if (DesireBrdcst)
01195                 {
01196                     if (!btnSend.Visible)
01197                     {
01198                         textbox1WASclicked = true; //show broadcast now
01199                     }
01200                     DesireBrdcst = false;
01201                 }
01202                 else if (DesireDM)
01203                 {
01204                     if (!DMPanelShowing)
01205                     {
01206                         DMclicked = true; //show DM now
01207                     }
01208                     else
01209                     {
01210                         DirectMsgPanel.Focus();
01211                         this.Text = " Classroom Inquisition | Direct
01212                         Msg" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01213                     }
01214                     DesireDM = false;
01215                 }
01216                 else if (DesirePrefs)
01217                 {
01218                     if (!PrefsShowing)
01219                     {
01220                         PrefsClicked = true; //show prefs now
01221                     }
01222                     else
01223                     {
01224                         PanelPrefs.Focus();
01225                         this.Text = " Classroom Inquisition | Prefs"
01226                         + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01227                     }
01228                     DesirePrefs = false;
01229                 }
01230                 else
01231                 {
01232                     if (!NotifyShowing && !DesireNotify)
01233                     {
01234                         timer.Enabled = false;//all done
01235                     }
01236                     if (StuMgmtShowing)
01237                         this.Text = " Classroom Inquisition | Student
01238                         Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01239                     else if (DMPanelShowing)
01240                         this.Text = " Classroom Inquisition | Direct
01241                         Msg" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01242                     else if (PrefsShowing)
01243                         this.Text = " Classroom Inquisition | Prefs"
01244                         + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");

```

```

01236                     else
01237                         this.Text = " Classroom Inquisition | Home" +
01238                             (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01239                         }
01240                     }
01241                 else //no animations
01242                 {
01243                     PanelFAQ.SetBounds(PanelFAQ.Location.X, PanelFAQ.Location.Y
01244                         - 520,
01245                     Height);
01246                     FAQShowing = false;
01247                     FAQClicked = false;
01248                     aPanelIsMoving = false;
01249                     FAQTimesClicked = 0;
01250                     if (DesireBrdcst)
01251                     {
01252                         if (!btnSend.Visible)
01253                         {
01254                             textbox1WASclicked = true; //show broadcast now
01255                         }
01256                         DesireBrdcst = false;
01257                     }
01258                     else if (DesireDM)
01259                     {
01260                         if (!DMPanelShowing)
01261                         {
01262                             DMclicked = true; //show DM now
01263                         }
01264                         else
01265                             DirectMsgPanel.Focus();
01266                         this.Text = " Classroom Inquisition | Direct Msg"
01267                             + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01268                         DesireDM = false;
01269                     }
01270                     else if (DesirePrefs)
01271                     {
01272                         if (!PrefsShowing)
01273                         {
01274                             PrefsClicked = true; //show prefs now
01275                         }
01276                         else
01277                         {
01278                             PanelPrefs.Focus();
01279                         this.Text = " Classroom Inquisition | Prefs" + (
01280                             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01281                         DesirePrefs = false;
01282                     }
01283                     else
01284                     {
01285                         if (!NotifyShowing && !DesireNotify)
01286                         {
01287                             timer.Enabled = false; //all done
01288                         }
01289                         if (StuMgmtShowing)
01290                             this.Text = " Classroom Inquisition | Student
01291                             Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01292                             else if (DMPanelShowing)
01293                             this.Text = " Classroom Inquisition | Direct Msg"
01294                             + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01295                             else if (PrefsShowing)
01296                             this.Text = " Classroom Inquisition | Prefs" + (
01297                             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01298                         else
01299                             this.Text = " Classroom Inquisition | Home" + (
01300                             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01301                     }
01302                 }

```

```

01299         }
01300
01301
01302     #endregion
01303
01304     #region Student Management Animate
01305
01306     if ((PanelStudents.Location.Y > 24) && (StuMgmtClicked == true)
01307         && (StuMgmtTimesClicked == 0) && (StuMgmtShowing == false))
01308     {
01309         if (!chkbxLameMode.Checked)
01310         {
01311             if (x < 10)
01312             {
01313                 if (x == 0)
01314                 {
01315                     aPanelIsMoving = true;
01316                     PanelStudents.BringToFront();
01317                 }
01318                 PanelStudents.SetBounds(PanelStudents.Location.X,
01319                     PanelStudents.Location.Y - 25,
01320                     PanelStudents.Size.Width,
01321                     PanelStudents.Size.Height);
01320                     x++;
01321                 }
01322             else if (x < 20)
01323             {
01324                 PanelStudents.SetBounds(PanelStudents.Location.X,
01325                     PanelStudents.Location.Y - 15,
01326                     PanelStudents.Size.Width,
01327                     PanelStudents.Size.Height);
01326                     x++;
01327                 }
01328             else if (x < 30)
01329             {
01330                 PanelStudents.SetBounds(PanelStudents.Location.X,
01331                     PanelStudents.Location.Y - 12,
01332                     PanelStudents.Size.Width,
01333                     PanelStudents.Size.Height);
01332                     x++;
01334                 }
01335             else
01336             {
01337                 x = 0;
01338                 StuMgmtShowing = true;
01339                 StuMgmtClicked = false;
01340                 aPanelIsMoving = false;
01341                 StuMgmtTimesClicked = 1;
01341                 if (!NotifyShowing && !DesireNotify)
01342                 {
01343                     timer.Enabled = false;
01344                 }
01345                 PanelStudents.Focus();
01346                 this.Text = " Classroom Inquisition | Student
Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01347             }
01348         }
01349     else //no animations, boo!
01350     {
01351         aPanelIsMoving = true;
01352         PanelStudents.BringToFront();
01353         PanelStudents.SetBounds(PanelStudents.Location.X,
01354             PanelStudents.Location.Y - 520,
01355             PanelStudents.Size.Width,
01356             PanelStudents.Size.Height);
01355             StuMgmtShowing = true;
01356             StuMgmtClicked = false;
01357             aPanelIsMoving = false;
01358             StuMgmtTimesClicked = 1;
01359             if (!NotifyShowing && !DesireNotify)
01360             {
01361                 timer.Enabled = false;
01362             }
01363             PanelStudents.Focus();

```

```

01364             this.Text = " Classroom Inquisition | Student Management"
01365         + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01366     }
01367     else if ((PanelStudents.Location.Y < 546) && (StuMgmtClicked ==
01368 true)
01369     {
01370         if (!chkbxLameMode.Checked)
01371         {
01372             if (x < 10)
01373             {
01374                 if (x == 0)
01375                 {
01376                     aPanelIsMoving = true;
01377                 }
01378                 PanelStudents.SetBounds(PanelStudents.Location.X,
01379                                         PanelStudents.Location.Y + 12,
01380                                         PanelStudents.Size.Width,
01381                                         PanelStudents.Size.Height);
01382             }
01383             else if (x < 20)
01384             {
01385                 PanelStudents.SetBounds(PanelStudents.Location.X,
01386                                         PanelStudents.Location.Y + 15,
01387                                         PanelStudents.Size.Width,
01388                                         PanelStudents.Size.Height);
01389             }
01390             else if (x < 30)
01391             {
01392                 PanelStudents.SetBounds(PanelStudents.Location.X,
01393                                         PanelStudents.Location.Y + 25,
01394                                         PanelStudents.Size.Width,
01395                                         PanelStudents.Size.Height);
01396             }
01397             else
01398             {
01399                 x = 0;
01400                 StuMgmtShowing = false;
01401                 StuMgmtClicked = false;
01402                 aPanelIsMoving = false;
01403                 StuMgmtTimesClicked = 0;
01404                 if (DesirePrefs)
01405                 {
01406                     if (PrefsShowing)
01407                     {
01408                         PrefsClicked = true; //hide prefs now, only for
01409                         StuMgmt
01410                     }
01411                     DesirePrefs = false;
01412                 }
01413                 else
01414                 {
01415                     if (!NotifyShowing && !DesireNotify)
01416                     {
01417                         timer.Enabled = false;
01418                     }
01419                     if (PrefsShowing)
01420                     this.Text = " Classroom Inquisition | Prefs"
01421                     + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01422                     else
01423                     this.Text = " Classroom Inquisition | Home" +
01424                     (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01425                 }
01426             }
01427             else //no animations
01428             {
01429                 PanelStudents.SetBounds(PanelStudents.Location.X,
01430                                         PanelStudents.Location.Y + 520,

```

```
01425                                     PanelStudents.Size.Width,
01426             StuMgmtShowing = false;
01427             StuMgmtClicked = false;
01428             aPanelIsMoving = false;
01429             StuMgmtTimesClicked = 0;
01430             if (DesirePrefs)
01431             {
01432                 if (PrefsShowing)
01433                 {
01434                     PrefsClicked = true; //hide prefs now, only for
01435                     StuMgmt
01436                 }
01437             }
01438         }
01439     }
01440     if (!NotifyShowing && !DesireNotify)
01441     {
01442         timer.Enabled = false;
01443     }
01444     if (PrefsShowing)
01445         this.Text = " Classroom Inquisition | Prefs" + (
01446         UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01447     else
01448         this.Text = " Classroom Inquisition | Home" + (
01449         UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01450     }
01451 #endregion
01452
01453 #region Conversation Viewer Animation
01454 if ((PanelConvView.Location.Y < 1) && (ConvViewClicked == true)
01455     && (ConvViewTimesClicked == 0) && (ConvViewShowing == false))
01456 {
01457     if (!chkbxLameMode.Checked)
01458     {
01459         if (x < 10)
01460         {
01461             if (x == 0)
01462             {
01463                 aPanelIsMoving = true;
01464                 PanelConvView.BringToFront();
01465             }
01466             PanelConvView.SetBounds(PanelConvView.Location.X,
01467             PanelConvView.Location.Y + 25,
01468             PanelConvView.Size.Width,
01469             PanelConvView.Size.Height);
01470         }
01471     }
01472     PanelConvView.SetBounds(PanelConvView.Location.X,
01473     PanelConvView.Location.Y + 15,
01474     PanelConvView.Size.Width,
01475     PanelConvView.Size.Height);
01476     x++;
01477 }
01478 else if (x < 20)
01479 {
01480     PanelConvView.SetBounds(PanelConvView.Location.X,
01481     PanelConvView.Location.Y + 13,
01482     PanelConvView.Size.Width,
01483     PanelConvView.Size.Height);
01484     x++;
01485 }
01486 else
01487 {
01488     x = 0;
01489     ConvViewShowing = true;
01490     ConvViewClicked = false;
01491     aPanelIsMoving = false;
01492     ConvViewTimesClicked = 1;
```

```

01489
01490             if (!NotifyShowing && !DesireNotify)
01491             {
01492                 timer.Enabled = false;
01493             }
01494
01495                 PanelConvView.Focus();
01496                 this.Text = " Classroom Inquisition | Conversation
01497                 Viewer" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01498             }
01499         else //no animations
01500         {
01501             aPanelIsMoving = true;
01502             PanelConvView.BringToFront();
01503             PanelConvView.SetBounds(PanelConvView.Location.X,
01504             PanelConvView.Location.Y + 530,
01505             PanelConvView.Size.Width,
01506             PanelConvView.Size.Height);
01507             ConvViewShowing = true;
01508             ConvViewClicked = false;
01509             aPanelIsMoving = false;
01510             ConvViewTimesClicked = 1;
01511
01512             if (!NotifyShowing && !DesireNotify)
01513             {
01514                 timer.Enabled = false;
01515             }
01516
01517             PanelConvView.Focus();
01518             this.Text = " Classroom Inquisition | Conversation Viewer
01519             " + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01520         }
01521     else if ((PanelConvView.Location.Y > -531) && (ConvViewClicked ==
01522             true)
01523             && (ConvViewTimesClicked == 1) && (ConvViewShowing ==
01524             true))
01525     {
01526         if (!chkbxLameMode.Checked)
01527         {
01528             if (x < 10)
01529             {
01530                 aPanelIsMoving = true;
01531                 PanelConvView.SetBounds(PanelConvView.Location.X,
01532                 PanelConvView.Location.Y - 13,
01533                 PanelConvView.Size.Width,
01534                 PanelConvView.Size.Height);
01535             }
01536             PanelConvView.SetBounds(PanelConvView.Location.X,
01537             PanelConvView.Location.Y - 15,
01538             PanelConvView.Size.Width,
01539             PanelConvView.Size.Height);
01540             x++;
01541         }
01542         else if (x < 20)
01543         {
01544             PanelConvView.SetBounds(PanelConvView.Location.X,
01545             PanelConvView.Location.Y - 25,
01546             PanelConvView.Size.Width,
01547             PanelConvView.Size.Height);
01548             x++;
01549         }
01550     }

```

```

01551     aPanelIsMoving = false;
01552     menuStrip1.BringToFront();
01553     ConvViewTimesClicked = 0;
01554
01555     if (!NotifyShowing && !DesireNotify)
01556     {
01557         timer.Enabled = false;
01558     }
01559
01560     if (StuMgmtShowing)
01561         this.Text = " Classroom Inquisition | Student
01562         Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01563     else if (PrefsShowing)
01564         this.Text = " Classroom Inquisition | Prefs" + (
01565             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01566     else
01567         this.Text = " Classroom Inquisition | Home" + (
01568             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01569     }
01570
01571     PanelConvView.SetBounds(PanelConvView.Location.X,
01572                             PanelConvView.Location.Y - 530,
01573                             PanelConvView.Size.Width,
01574                             PanelConvView.Size.Height);
01575     ConvViewShowing = false;
01576     ConvViewClicked = false;
01577     aPanelIsMoving = false;
01578     ConvViewTimesClicked = 0;
01579     menuStrip1.BringToFront();
01580
01581     if (!NotifyShowing && !DesireNotify)
01582     {
01583         timer.Enabled = false;
01584     }
01585
01586     if (StuMgmtShowing)
01587         this.Text = " Classroom Inquisition | Student
01588         Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01589     else if (PrefsShowing)
01590         this.Text = " Classroom Inquisition | Prefs" + (
01591             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01592     }
01593
01594 #endregion
01595
01596 #region Quiz Maker Animate
01597
01598 if ((PanelQuizMaker.Location.X > -1) && (QuizMakerClicked == true)
01599     && (QuizMakerTimesClicked == 0) && (QuizMakerShowing == false))
01600 {
01601     if (!chkbxLameMode.Checked)
01602     {
01603         if (x < 10)
01604         {
01605             aPanelIsMoving = true;
01606             PanelQuizMaker.BringToFront();
01607         }
01608         PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X - 20
01609             , PanelQuizMaker.Location.Y,
01610             PanelQuizMaker.Size.Width,
01611             PanelQuizMaker.Size.Height);
01612         x++;
01613     }
01614     else if (x < 20)
01615     {
01616         PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X - 12

```

```
    , PanelQuizMaker.Location.Y,
01615                                         PanelQuizMaker.Size.Width,
01616     PanelQuizMaker.Size.Height);
01617             x++;
01618         }
01619     {
01620         PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X - 8,
01621     PanelQuizMaker.Location.Y,
01622                                         PanelQuizMaker.Size.Width,
01623             x++);
01624         }
01625     {
01626         x = 0;
01627         QuizMakerShowing = true;
01628         QuizMakerClicked = false;
01629         aPanelIsMoving = false;
01630         QuizMakerTimesClicked = 1;
01631
01632         if (!NotifyShowing && !DesireNotify)
01633         {
01634             timer.Enabled = false;
01635         }
01636
01637         PanelQuizMaker.Focus();
01638         this.Text = " Classroom Inquisition | Quiz Maker" + (
01639             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01640         }
01641     else //no animations
01642     {
01643         aPanelIsMoving = true;
01644         PanelQuizMaker.BringToFront();
01645         PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X - 400,
01646     PanelQuizMaker.Location.Y,
01647                                         PanelQuizMaker.Size.Width,
01648             PanelQuizMaker.Size.Height);
01649         QuizMakerShowing = true;
01650         QuizMakerClicked = false;
01651         aPanelIsMoving = false;
01652         QuizMakerTimesClicked = 1;
01653
01654         if (!NotifyShowing && !DesireNotify)
01655         {
01656             timer.Enabled = false;
01657
01658             PanelQuizMaker.Focus();
01659             this.Text = " Classroom Inquisition | Quiz Maker" + (
01660                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01661             }
01662         else if ((PanelQuizMaker.Location.X < 401) && (QuizMakerClicked ==
01663             true)
01664             && (QuizMakerTimesClicked == 1) && (QuizMakerShowing ==
01665             true))
01666         {
01667             if (!chkbxLameMode.Checked)
01668             {
01669                 if (x < 10)
01670                 {
01671                     if (x == 0)
01672                     {
01673                         aPanelIsMoving = true;
01674                     }
01675                     PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X + 8,
01676     PanelQuizMaker.Location.Y,
01677                                         PanelQuizMaker.Size.Width,
01678             PanelQuizMaker.Size.Height);
01679             x++;
01680         }
01681     else if (x < 20)
```

```
01677             {
01678                 PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X + 12
01679                 , PanelQuizMaker.Location.Y,
01680                 PanelQuizMaker.Size.Width,
01681                 PanelQuizMaker.Size.Height);
01682                 x++;
01683             }
01684             else if (x < 30)
01685             {
01686                 PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X + 20
01687                 , PanelQuizMaker.Location.Y,
01688                 PanelQuizMaker.Size.Width,
01689                 PanelQuizMaker.Size.Height);
01690                 x++;
01691             }
01692         else
01693         {
01694             x = 0;
01695             QuizMakerShowing = false;
01696             QuizMakerClicked = false;
01697             aPanelIsMoving = false;
01698             QuizMakerTimesClicked = 0;
01699
01700             if (!NotifyShowing && !DesireNotify)
01701             {
01702                 timer.Enabled = false;
01703             }
01704         else //no animations, boo!
01705         {
01706             PanelQuizMaker.SetBounds(PanelQuizMaker.Location.X + 400,
01707             PanelQuizMaker.Location.Y,
01708             PanelQuizMaker.Size.Width,
01709             PanelQuizMaker.Size.Height);
01710             QuizMakerShowing = false;
01711             QuizMakerClicked = false;
01712             aPanelIsMoving = false;
01713             QuizMakerTimesClicked = 0;
01714             this.Text = " Classroom Inquisition | Home" + (UnreadCount
01715             > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01716         }
01717     }
01718 }
01719 }
01720
01721
01722 #endregion
01723
01724 #region Quiz Mode Animate
01725 if ((PanelQuizMode.Location.X < 1) && (QuizModeClicked == true)
01726     && (QuizModeTimesClicked == 0) && (QuizModeShowing == false))
01727 {
01728     if (!chkbxLameMode.Checked)
01729     {
01730         if (x < 10)
01731         {
01732             if (x == 0)
01733             {
01734                 aPanelIsMoving = true;
01735                 PanelQuizMode.BringToFront();
01736             }
01737             PanelQuizMode.SetBounds(PanelQuizMode.Location.X + 20,
01738             PanelQuizMode.Location.Y,
01739             PanelQuizMode.Size.Width,
01740             PanelQuizMode.Size.Height);
01741             x++;
01742     }
```

```

01741             else if (x < 20)
01742             {
01743                 PanelQuizMode.SetBounds(PanelQuizMode.Location.X + 12,
01744                                         PanelQuizMode.Location.Y,
01745                                         PanelQuizMode.Size.Width,
01746                                         PanelQuizMode.Size.Height);
01747             }
01748             else if (x < 30)
01749             {
01750                 PanelQuizMode.SetBounds(PanelQuizMode.Location.X + 8,
01751                                         PanelQuizMode.Location.Y,
01752                                         PanelQuizMode.Size.Width,
01753                                         PanelQuizMode.Size.Height);
01754             }
01755             x = 0;
01756             QuizModeShowing = true;
01757             QuizModeClicked = false;
01758             aPanelIsMoving = false;
01759             QuizModeTimesClicked = 1;
01760
01761             if (!NotifyShowing && !DesireNotify)
01762             {
01763                 timer.Enabled = false;
01764             }
01765
01766             PanelQuizMode.Focus();
01767             this.Text = " Classroom Inquisition | Quiz Mode" + (
01768             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01769         }
01770         else //no animations
01771         {
01772             aPanelIsMoving = true;
01773             PanelQuizMode.BringToFront();
01774             PanelQuizMode.SetBounds(PanelQuizMode.Location.X + 400,
01775                                         PanelQuizMode.Location.Y,
01776                                         PanelQuizMode.Size.Width,
01777                                         PanelQuizMode.Size.Height);
01778             QuizModeShowing = true;
01779             QuizModeClicked = false;
01780             aPanelIsMoving = false;
01781             QuizModeTimesClicked = 1;
01782             if (!NotifyShowing && !DesireNotify)
01783             {
01784                 timer.Enabled = false;
01785             }
01786             PanelQuizMode.Focus();
01787             this.Text = " Classroom Inquisition | Quiz Mode" + (
01788             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01789         }
01790         else if ((PanelQuizMode.Location.X > -401) && (QuizModeClicked ==
01791             true)
01792             && (QuizModeTimesClicked == 1) && (QuizModeShowing ==
01793             true))
01794         {
01795             if (!chkbxLameMode.Checked)
01796             {
01797                 if (x < 10)
01798                 {
01799                     if (x == 0)
01800                     {
01801                         aPanelIsMoving = true;
01802                         PanelQuizMode.SetBounds(PanelQuizMode.Location.X - 8,
01803                                         PanelQuizMode.Location.Y,
01804                                         PanelQuizMode.Size.Width,
01805                                         PanelQuizMode.Size.Height);

```

```

01803             x++;
01804         }
01805         else if (x < 20)
01806     {
01807         PanelQuizMode.SetBounds(PanelQuizMode.Location.X - 12,
01808         PanelQuizMode.Location.Y,
01809         PanelQuizMode.Size.Width,
01810         PanelQuizMode.Size.Height);
01811         x++;
01812         else if (x < 30)
01813     {
01814         PanelQuizMode.SetBounds(PanelQuizMode.Location.X - 20,
01815         PanelQuizMode.Location.Y,
01816         PanelQuizMode.Size.Width,
01817         PanelQuizMode.Size.Height);
01818         x++;
01819         else
01820     {
01821         x = 0;
01822         QuizModeShowing = false;
01823         QuizModeClicked = false;
01824         aPanelIsMoving = false;
01825         QuizModeTimesClicked = 0;
01826         if (!NotifyShowing && !DesireNotify)
01827     {
01828         timer.Enabled = false;
01829     }
01830         this.Text = " Classroom Inquisition | Home" + (
01831         UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01832     }
01833     else //no animations, boo!
01834     {
01835         PanelQuizMode.SetBounds(PanelQuizMode.Location.X - 400,
01836         PanelQuizMode.Location.Y,
01837         PanelQuizMode.Size.Width,
01838         PanelQuizMode.Size.Height);
01839         QuizModeShowing = false;
01840         QuizModeClicked = false;
01841         aPanelIsMoving = false;
01842         QuizModeTimesClicked = 0;
01843         this.Text = " Classroom Inquisition | Home" + (UnreadCount
01844         > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01845         if (!NotifyShowing && !DesireNotify)
01846     {
01847         timer.Enabled = false;
01848     }
01849 }
01850 #endregion
01851
01852 #region Attendance Mode Animation (android)
01853 if ((PanelAttendance.Location.Y <= 265) && (AttendanceClicked ==
01854 true)
01855     && (AttendanceTimesClicked == 0) && (AttendanceShowing ==
01856 false))
01857 {
01858     if (!chkbxLameMode.Checked)
01859     {
01860         if (x < 10)
01861         {
01862             if (x == 0)
01863             {
01864                 aPanelIsMoving = true;
01865                 PanelAttendance.Show();
01866                 PanelAttendance.SetBounds(190, PanelAttendance.
01867                 Location.Y - 5,

```

```

01866                               PanelAttendance.Size;
01867                               }
01868                               PanelAttendance.SetBounds(PanelAttendance.Location.X -
01869                               19, PanelAttendance.Location.Y,
01870                               PanelAttendance.Size.Height + 2);
01871                               }
01872                               else if (x < 20)
01873                               {
01874                               PanelAttendance.SetBounds(PanelAttendance.Location.X,
01875                               PanelAttendance.Location.Y - 10,
01876                               PanelAttendance.Size.Width,
01877                               PanelAttendance.Size.Height + 23);
01878                               }
01879                               else if (x < 30)
01880                               {
01881                               PanelAttendance.SetBounds(PanelAttendance.Location.X,
01882                               PanelAttendance.Location.Y - 16,
01883                               PanelAttendance.Size.Width,
01884                               PanelAttendance.Size.Height + 32);
01885                               }
01886                               else
01887                               {
01888                               x = 0;
01889                               AttendanceShowing = true;
01890                               AttendanceClicked = false;
01891                               aPanelIsMoving = false;
01892                               AttendanceTimesClicked = 1;
01893                               if (!NotifyShowing && !DesireNotify)
01894                               {
01895                                   timer.Enabled = false;
01896                               }
01897                               else
01898                               {
01899                                   PanelNotify.BringToFront();
01900                                   this.Text = " Classroom Inquisition | Attendance" +
01901                                   UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "";
01902                               }
01903                               else //no animations
01904                               {
01905                                   PanelAttendance.SetBounds(0, 0, 384, 550);
01906                                   PanelAttendance.Show();
01907                                   AttendanceShowing = true;
01908                                   AttendanceClicked = false;
01909                                   aPanelIsMoving = false;
01910                                   AttendanceTimesClicked = 1;
01911                                   if (!NotifyShowing && !DesireNotify)
01912                                   {
01913                                       timer.Enabled = false;
01914                                   }
01915                                   else
01916                                   {
01917                                       PanelNotify.BringToFront();
01918                                   }
01919                                   PanelAttendance.Focus();
01920                                   this.Text = " Classroom Inquisition | Attendance" +
01921                                   UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "";
01922                               }
01923                               else if ((PanelAttendance.Location.Y >= 0) && (AttendanceClicked ==
01924                               true)
01925                               && (AttendanceTimesClicked == 1) && (AttendanceShowing ==
01926                               true))
01927                               {
01928                               if (!chkbxLameMode.Checked)
01929                               {
01930                                   if (x < 10)

```

```

01929         {
01930             if (x==0)
01931             {
01932                 aPanelIsMoving = true;
01933             }
01934             PanelAttendance.SetBounds(PanelAttendance.Location.X,
01935                                         PanelAttendance.Location.Y + 16,
01936                                         PanelAttendance.Size.Width,
01937                                         PanelAttendance.Size.Height - 32);
01938             x++;
01939         }
01940         else if (x < 20)
01941         {
01942             PanelAttendance.SetBounds(PanelAttendance.Location.X,
01943                                         PanelAttendance.Location.Y + 10,
01944                                         PanelAttendance.Size.Width,
01945                                         PanelAttendance.Size.Height - 23);
01946             x++;
01947         }
01948         else if (x < 30)
01949         {
01950             PanelAttendance.SetBounds(PanelAttendance.Location.X +
01951                                         19, PanelAttendance.Location.Y,
01952                                         PanelAttendance.Size.Width -
01953                                         38, PanelAttendance.Size.Height);
01954             if (x == 29)
01955             {
01956                 PanelAttendance.SetBounds(PanelAttendance.Location.X,
01957                                         PanelAttendance.Location.Y + 5,
01958                                         PanelAttendance.Size.Width -
01959                                         4, PanelAttendance.Size.Height - 2);
01960             }
01961             x++;
01962         }
01963     }
01964     else
01965     {
01966         x = 0;
01967         AttendanceShowing = false;
01968         AttendanceClicked = false;
01969         aPanelIsMoving = false;
01970         AttendanceTimesClicked = 0;
01971         PanelAttendance.Hide();
01972         if (!NotifyShowing && !DesireNotify)
01973         {
01974             timer.Enabled = false;
01975         }
01976         if (StuMgmtShowing)
01977             this.Text = " Classroom Inquisition | Student
Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01978         else if (PrefsShowing)
01979             this.Text = " Classroom Inquisition | Prefs" + (
01980             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01981         else
01982             this.Text = " Classroom Inquisition | Home" + (
01983             UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01984         PanelAttendance.SetBounds(190, 265, 0, 0);
01985         PanelAttendance.Hide();
01986         AttendanceShowing = false;
01987         AttendanceClicked = false;
01988         aPanelIsMoving = false;
01989         AttendanceTimesClicked = 0;
01990         if (!NotifyShowing && !DesireNotify)
01991         {
01992             timer.Enabled = false;
01993         }
01994         if (StuMgmtShowing)

```

```

01992             this.Text = " Classroom Inquisition | Student
01993             Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01994             else if (PrefsShowing)
01995                 this.Text = " Classroom Inquisition | Prefs" + (
01996                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
01997             else
01998                 this.Text = " Classroom Inquisition | Home" + (
01999                 UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02000         }
02001     #endregion
02002     #region Class Vote Mode Animation (inverse android)
02003     if ((PanelClassVote.Location.Y <= 266) && (ClassVoteClicked == true
02004         )
02005             && (ClassVoteTimesClicked == 0) && (ClassVoteShowing == false)
02006         )
02007         {
02008             if (!chkbxLameMode.Checked)
02009                 {
02010                     if (x < 10)
02011                         {
02012                             if (x == 0)
02013                                 {
02014                                     aPanelIsMoving = true;
02015                                     PanelClassVote.BringToFront();
02016                                     PanelNotify.BringToFront();
02017                                     PanelClassVote.Show();
02018                                     PanelClassVote.SetBounds(190, PanelClassVote.
02019                                         Location.Y - 5,
02020                                         PanelClassVote.Size.Width +
02021                                         4, PanelClassVote.Size.Height+2);
02022                                     }
02023                                     PanelClassVote.SetBounds(PanelClassVote.Location.X,
02024                                         PanelClassVote.Location.Y - 26,
02025                                         PanelClassVote.Size.Width,
02026                                         PanelClassVote.Size.Height + 55);
02027                                     x++;
02028                                     }
02029                                     else if (x < 20)
02030                                         {
02031                                             PanelClassVote.SetBounds(PanelClassVote.Location.X - 12
02032                                         , PanelClassVote.Location.Y,
02033                                         PanelClassVote.Size.Width + 24
02034                                         , PanelClassVote.Size.Height);
02035                                         x++;
02036                                         }
02037                                         else
02038                                             {
02039                                                 x = 0;
02040                                                 ClassVoteShowing = true;
02041                                                 ClassVoteClicked = false;
02042                                                 aPanelIsMoving = false;
02043                                                 ClassVoteTimesClicked = 1;
02044                                                 if (!NotifyShowing && !DesireNotify)
02045                                                     {
02046                                                         timer.Enabled = false;
02047                                                         }
02048                                                         pictureBox8.Show();
02049                                                         lblCVLeft.Show();
02050                                                         lblCVRight.Show();
02051                                                         lblOption1.Show();
02052                                                         lblOption2.Show();
02053                                                         lblCVPipe.Show();

```

```

02053                     txtbxCVStats.Show();
02054                     btnCVReset.Show();
02055                     btnExitClassVote.Show();
02056                     PanelClassVote.Focus();
02057                     this.Text = " Classroom Inquisition | ClassVote" + (
02058                         UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02059                     }
02060             }
02061         else //no animations
02062             {
02063                 PanelClassVote.SetBounds(0, 0, 384, 550);
02064                 aPanelIsMoving = true;
02065                 PanelClassVote.BringToFront();
02066                 PanelNotify.BringToFront();
02067                 PanelClassVote.Show();
02068                 ClassVoteShowing = true;
02069                 ClassVoteClicked = false;
02070                 aPanelIsMoving = false;
02071                 ClassVoteTimesClicked = 1;
02072                 if (!NotifyShowing && !DesireNotify)
02073                 {
02074                     timer.Enabled = false;
02075                 }
02076                 pictureBox8.Show();
02077                 lblCVLeft.Show();
02078                 lblCVRight.Show();
02079                 lblOption1.Show();
02080                 lblOption2.Show();
02081                 lblCVPipe.Show();
02082                 txtbxCVStats.Show();
02083                 btnCVReset.Show();
02084                 btnExitClassVote.Show();
02085                 PanelClassVote.Focus();
02086                 this.Text = " Classroom Inquisition | ClassVote" + (
02087                     UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02088             }
02089         else if ((PanelClassVote.Location.Y >= 0) && (ClassVoteClicked ==
02090             true)
02091             && (ClassVoteTimesClicked == 1) && (ClassVoteShowing ==
02092             true))
02093             {
02094                 if (!chkbxLameMode.Checked)
02095                 {
02096                     if (x < 10)
02097                     {
02098                         if (x == 0)
02099                         {
02100                             aPanelIsMoving = true;
02101                             pictureBox8.Hide();
02102                             lblCVLeft.Hide();
02103                             lblCVRight.Hide();
02104                             lblOption1.Hide();
02105                             lblOption2.Hide();
02106                             lblCVPipe.Hide();
02107                             txtbxCVStats.Hide();
02108                             btnCVReset.Hide();
02109                             btnExitClassVote.Hide();
02110                         }
02111                         PanelClassVote.SetBounds(PanelClassVote.Location.X + 7,
02112                                         PanelClassVote.Location.Y,
02113                                         PanelClassVote.Size.Width - 14
02114                                         , PanelClassVote.Size.Height);
02115                         x++;
02116                     }
02117                     else if (x < 20)
02118                     {
02119                         PanelClassVote.SetBounds(PanelClassVote.Location.X + 12
02120                                         , PanelClassVote.Location.Y,
02121                                         PanelClassVote.Size.Width - 24
02122                                         , PanelClassVote.Size.Height);
02123                         x++;
02124                     }
02125                 else if (x < 30)

```

```

02119             {
02120                 PanelClassVote.SetBounds(PanelClassVote.Location.X,
02121                     PanelClassVote.Location.Y + 26,
02122                     PanelClassVote.Size.Width,
02123                     PanelClassVote.Size.Height - 55);
02124                 if (x == 29)
02125                 {
02126                     PanelClassVote.SetBounds(PanelClassVote.Location.X,
02127                         PanelClassVote.Location.Y + 5,
02128                         PanelClassVote.Size.Width
02129                         - 4, PanelClassVote.Size.Height);
02130                 }
02131                 x++;
02132             }
02133             else
02134             {
02135                 x = 0;
02136                 ClassVoteShowing = false;
02137                 ClassVoteClicked = false;
02138                 aPanelIsMoving = false;
02139                 ClassVoteTimesClicked = 0;
02140                 PanelClassVote.Hide();
02141
02142                 if (!NotifyShowing && !DesireNotify)
02143                 {
02144                     timer.Enabled = false;
02145
02146                     if (StuMgmtShowing)
02147                         this.Text = " Classroom Inquisition | Student
Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02148                     else if (PrefsShowing)
02149                         this.Text = " Classroom Inquisition | Prefs" + (
UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02150                     else
02151                         this.Text = " Classroom Inquisition | Home" + (
UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02152                 }
02153                 else //no animations
02154                 {
02155                     PanelClassVote.SetBounds(190, 265, 0, 0);
02156                     pictureBox8.Hide();
02157                     lblCVLeft.Hide();
02158                     lblCVRight.Hide();
02159                     lblOption1.Hide();
02160                     lblOption2.Hide();
02161                     lblCVPipe.Hide();
02162                     txtbxCVStats.Hide();
02163                     btnCVReset.Hide();
02164                     btnExitClassVote.Hide();
02165                     PanelClassVote.Hide();
02166                     ClassVoteShowing = false;
02167                     ClassVoteClicked = false;
02168                     aPanelIsMoving = false;
02169                     ClassVoteTimesClicked = 0;
02170
02171                     if (!NotifyShowing && !DesireNotify)
02172                     {
02173                         timer.Enabled = false;
02174
02175                         if (StuMgmtShowing)
02176                             this.Text = " Classroom Inquisition | Student
Management" + (UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02177                         else if (PrefsShowing)
02178                             this.Text = " Classroom Inquisition | Prefs" + (
UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02179                         else
02180                             this.Text = " Classroom Inquisition | Home" + (
UnreadCount > 0 ? "(" + UnreadCount.ToString() + ")" : "");
02181                     }
02182             }

```

```
02183         #endregion
02184
02185         #region Notify Panel
02186         if (!NotifyShowing) && (DesireNotify == true) && (PanelNotify.
02187             Location.Y > 514)) // if not visible
02188         {
02189             if (!chkbxLameMode.Checked) //animations
02190             {
02191                 if (notif_I < 5)
02192                 {
02193                     if (notif_I == 0)
02194                     {
02195                         PanelNotify.BringToFront();
02196                     }
02197                     PanelNotify.SetBounds(PanelNotify.Location.X,
02198                         PanelNotify.Location.Y - 3,
02199                         PanelNotify.Size.Width,
02200                         PanelNotify.Size.Height);
02201                     notif_I++;
02202                 }
02203                 else if (notif_I < 15)
02204                 {
02205                     PanelNotify.SetBounds(PanelNotify.Location.X,
02206                         PanelNotify.Location.Y - 2,
02207                         PanelNotify.Size.Width,
02208                         PanelNotify.Size.Height);
02209                     notif_I++;
02210                 }
02211                 else
02212                 {
02213                     //timer stays on to hide the panel soon..
02214                     DesireNotify = false;
02215                     NotifyShowing = true;
02216                 }
02217             }
02218             else //no animations
02219             {
02220                 //jump straight to whatever position..
02221                 PanelNotify.SetBounds(PanelNotify.Location.X, PanelNotify.
02222                     Location.Y - 35,
02223                     PanelNotify.Size.Width, PanelNotify.
02224                     Size.Height);
02225                     NotifyShowing = true;
02226                     DesireNotify = false;
02227             }
02228         }
02229         if ((NotifyShowing) && (!DesireNotify) && (PanelNotify.Location.Y <
02230             551)) // if visible, destroy
02231         {
02232             if (!chkbxLameMode.Checked)
02233             {
02234                 if (notif_I < ((double)numUpDnNotify.Value*100))
02235                 {
02236                     notif_I++; //wait until timeout
02237                 }
02238                 else if (notif_I < ((double)numUpDnNotify.Value * 100)+5)
02239                 {
02240                     PanelNotify.SetBounds(PanelNotify.Location.X,
02241                         PanelNotify.Location.Y + 7,
02242                         PanelNotify.Size.Width,
02243                         PanelNotify.Size.Height);
02244                     notif_I++;
02245                 }
02246             }
02247             else
02248             {
02249                 notif_I = 0;
02250                 if (!aPanelIsMoving)
02251                 {
02252                     timer.Enabled = false;
02253                 }
02254                 DesireNotify = false;
02255                 NotifyShowing = false;
02256             }
02257         }
```

```

02247         }
02248         else //no animations
02249     {
02250         if (notif_I < ((double)numUpDnNotify.Value * 100))
02251         {
02252             notif_I++; //wait until timeout
02253         }
02254         else
02255         {
02256             PanelNotify.SetBounds(PanelNotify.Location.X,
02257             PanelNotify.Location.Y + 35,
02258             PanelNotify.Size.Width,
02259             PanelNotify.Size.Height);
02260             if (!aPanelIsMoving)
02261             {
02262                 timer.Enabled = false;
02263             }
02264             notif_I = 0;
02265             DesireNotify = false;
02266             NotifyShowing = false;
02267         }
02268     }
02269 #endregion
02270
02271 } //end of timer_Tick
02272
02273 }

```

6.7 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1-dyn-ctrls.cs File Reference

Classes

- class [SrP_ClassroomInq::frmClassromInq](#)

Packages

- package [SrP_ClassroomInq](#)

Variables

- using [System](#)

6.7.1 Variable Documentation

6.7.1.1 using System

Definition at line 18 of file [Form1-dyn-ctrls.cs](#).

6.8 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1-dyn-ctrls.cs

```
00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 ****/
00019 using System;
00020 using System.Collections.Generic;
00021 using System.ComponentModel;
00022 using System.Data;
00023 using System.Drawing;
00024 using System.Drawing.Printing;
00025 using System.Linq;
00026 using System.Media;
00027 using System.Text;
00028 using System.Windows.Forms;
00029 using System.IO;
00030 using System.Security.Cryptography;
00031 using System.Runtime.InteropServices;
00032
00033 namespace SrP_ClassroomInq
00034 {
00035     public partial class frmClassrromInq : Form
00036     {
00037         *****
00038         * Classroom Inquisition Teacher Application
00039         * Programmer: David Christensen
00040         * Bug Tracking: https://github.com/dchriste/Classroom-Inquisition
00041         issues
00042         *****/
00043         public frmClassrromInq()
00044         {
00045             InitializeComponent();
00046             textBox1.KeyPress += new System.Windows.Forms.KeyPressEventHandler(
00047                 CheckKeys); // so you can send with enter in the main txtbx
00048             PanelStudents.KeyPress += new System.Windows.Forms.
00049             KeyPressEventHandler(CheckKeys);
00050             PanelFAQ.KeyPress += new System.Windows.Forms.KeyPressEventHandler(
00051                 CheckKeys);
00052             PanelPrefs.KeyPress += new System.Windows.Forms.
00053             KeyPressEventHandler(CheckKeys);
00054             DirectMsgPanel.KeyPress += new System.Windows.Forms.
00055             KeyPressEventHandler(CheckKeys);
00056             PanelConvView.KeyPress += new System.Windows.Forms.
00057             KeyPressEventHandler(CheckKeys);
00058             PanelAttendance.KeyPress += new System.Windows.Forms.
00059             KeyPressEventHandler(CheckKeys);
00060             PanelQuizMaker.KeyPress += new System.Windows.Forms.
00061             KeyPressEventHandler(CheckKeys);
00062             PanelQuizMode.KeyPress += new System.Windows.Forms.
00063             KeyPressEventHandler(CheckKeys);
00064         }
00065     }
00066 }
```

```

00055             PanelClassVote.KeyPress += new System.Windows.Forms.
00056                 KeyPressEventHandler(CheckKeys);
00057                     pnlBrdCst.KeyPress += new System.Windows.Forms.KeyPressEventHandler
00058                         (CheckKeys);
00059                     }
00060             #region Initialization
00061         public const int classSize = 50; //will be max questions handled at
00062             once
00063             public const int MaxQuestions = 250;
00064             public StreamReader Stream2Print;
00065             public Font PrintFont;
00066             public Color ForeColorTheme = Color.DarkGoldenrod;
00067             public Color BackColorTheme = Color.Black;
00068             public AboutBox_CI About = new AboutBox_CI();
00069             public Splash SplashScreen = new Splash();
00070             #region Variables
00071             string[] Students_Name = new string[classSize];
00072             string[] Students_ID = new string[classSize];
00073             string[] Os_to_create = new string[classSize];
00074             bool Qs_to_Make = false;//flag to create controls
00075             string[] addr_tbl = {"\x20", "\x02", "\x03", "\x1F", "\x05", "\x06", "\x07",
00076             "\x22", "\x09", "\x21",
00077             "\x0B", "\x0C", "\x0D", "\x0E", "\x0F", "\x10", "\x11", "\x
00078             12", "\x13", "\x14",
00079             "\x15", "\x16", "\x17", "\x18", "\x19", "\x1A", "\x1B", "\x
00080             1C", "\x1D", "\x1E",
00081             };
00082             bool[] Q_status = new bool[MaxQuestions]; //records the read status of
00083             a question
00084             bool[] Q_Replied = new bool[MaxQuestions]; //records if the question
00085             has been replied to
00086             bool[] Q_NameShowing = new bool[MaxQuestions];
00087             bool[] wasQunread = new bool[MaxQuestions];
00088             bool[] StudentPresent = new bool[classSize]; //for attendance mode
00089             bool[] StudentVoted = new bool[classSize];
00090             byte PresentStuNum = 250;
00091             string[] Q_sender = new string[classSize]; // records the addr of the
00092             sender of each question
00093             string brdcst_addr = "\x01";
00094             string[] logFiles = new string[classSize];
00095             string EncryptedData = @"data\StudentInfo_Encrypted.txt";
00096             string PlainData = @"data\StudentInfo.txt";
00097             string[] portNames = new string[10];
00098             string tempString = "";
00099             string tempString2 = "";
00100             string RX_Data = "";
00101             string SecretKey;
00102             string StateMachine = "Normal"; //initial state
00103             SoundPlayer JukeBox = new System.Media.SoundPlayer(); //for alert tone
00104             bool WeGotData = false;
00105             byte i = 0;
00106             byte k = 0;
00107             byte j = 0;
00108             byte x = 0;
00109             int notif_I = 0; //incrementer for notify panel only
00110             static int logHistoryLength = 500; //this could be a preference, it
00111             determines the amount shown in conversation viewer
00112             string[] logTmpString = new string[logHistoryLength];
00113             int ix = 0; //only for serial tick timer...
00114             int jx = 0; //only for the make controls timer
00115             string addr = "\xFF";
00116             byte NumQuestions = 0;
00117             byte UnreadCount = 0;
00118             bool EDTClicked = false;
00119             bool BrdcstShowing = false;
00120             byte timesClicked = 0; //allows for double click of question
00121             byte DMtimesClicked = 0;
00122             byte PrefstTimesClicked = 0;
00123             byte FAQTimesClicked = 0;
00124             byte StuMgmtTimesClicked = 0;
00125             bool SplashScreenShowing = new bool();

```

```

00119     bool aPanelIsMoving = new bool();
00120     bool DesireNotify = new bool();
00121     bool NotifyShowing = new bool();
00122     bool DesirePrefs = new bool();
00123     bool DesireDM = new bool();
00124     bool DesireBrdcst = new bool();
00125     bool DesireFAQ = new bool();
00126     bool DesireQzMkr = new bool();
00127     bool DesireQuizMd = new bool();
00128     bool DesireStuMgmt = new bool();
00129     bool DesireAttendance = new bool();
00130     bool DesireClassVote = new bool();
00131     bool DesireConvView = new bool();
00132     bool FAQClicked = new bool();
00133     bool FAQShowing = new bool();
00134     bool StuMgmtClicked = new bool();
00135     bool StuMgmtShowing = new bool();
00136     bool DMclicked = new bool();
00137     bool DMPanelShowing = new bool();
00138     bool PrefsShowing = new bool();
00139     bool PrefClicked = new bool();
00140     bool ConvViewClicked = false;
00141     bool ConvViewShowing = false;
00142     byte ConvViewTimesClicked = 0;
00143     bool QuizMakerShowing = false;
00144     bool QuizMakerClicked = false;
00145     byte QuizMakerTimesClicked = 0;
00146     bool QuizModeShowing = false;
00147     bool QuizModeClicked = false;
00148     byte QuizModeTimesClicked = 0;
00149     bool AttendanceShowing = false;
00150     bool AttendanceClicked = false;
00151     byte AttendanceTimesClicked = 0;
00152     bool ClassVoteShowing = false;
00153     bool ClassVoteClicked = false;
00154     byte ClassVoteTimesClicked = 0;
00155         bool textbox1WASclicked = new bool();
00156         bool grpbxRPL_WASclicked = new bool();
00157         bool btnCLS_WASclicked = new bool();
00158     bool NEW_grpbx = new bool();
00159     bool AckRXd = false;
00160     bool timeout_ack = false;
00161     int old_lblID = 0;
00162     bool DesireID = false;
00163     bool DeleteQuestion = false;
00164     bool Question_Deleted = false;
00165     bool QuizQuestionDel = false;
00166     int QuizDelID = 255;
00167     string QzQuestionDel = "";
00168     bool Request_Undo = false;
00169     bool sudo_kill = false;
00170     byte Del_ID = 255;
00171     byte timesThrough = 0;
00172     int new_lblID = 0;
00173         int lbl_ID = 0;
00174     int lbl_ID_2 = 0; // for unread counter
00175 #endregion
00176
00177     #region Dyn-Ctrls-Init_DoNotChange
00178     System.Windows.Forms.GroupBox[] group_arr = new System.Windows.Forms.
00179     GroupBox[classSize]; //add const for max_lns
00180         System.Windows.Forms.Button[] reply_arr = new System.Windows.
00181     Forms.Button[classSize]; //add const for max_lns
00182         System.Windows.Forms.Button[] close_arr = new System.Windows.
00183     Forms.Button[classSize]; //add const for max_lns
00184         System.Windows.Forms.Button[] clear_arr = new System.Windows.
00185     Forms.Button[classSize]; //add const for max_lns
00186         System.Windows.Forms.TextBox[] txtbx_reply_arr = new System.
00187     Windows.Forms.TextBox[classSize]; //add const for max_lns
00188         System.Windows.Forms.Label[] lbl_arr = new System.Windows.Forms.
00189     .Label[classSize]; //add const for max_lns
00190         System.Windows.Forms.PictureBox[] picbx_arr = new System.Windows.Forms.
00191     PictureBox[classSize];
00192         System.Windows.Forms.PictureBox[] picbx_ConvView_arr = new System.

```

```

    Windows.Forms.PictureBox[classSize];
00186     System.Windows.Forms.ToolTip[] tt_picbxCV_arr = new System.Windows.
Forms.ToolTip[classSize];
00187     /*This is for the quiz mode*/
00188     System.Windows.Forms.Panel[] PanelQuestion_arr = new System.Windows.
Forms.Panel[classSize]; //limits number of quiz questions
00189     System.Windows.Forms.Button[] btnQuizQuestionSend_arr = new System.
Windows.Forms.Button[classSize];// ^
00190     System.Windows.Forms.Label[] lblQuestionNum_arr = new System.Windows.
Forms.Label[classSize];
00191     System.Windows.Forms.TextBox[] txtbxQuestion_arr = new System.Windows.
Forms.TextBox[classSize];
00192     /***** */
00193     System.Drawing.Point origingrouparr = new System.Drawing.Point(6, -13);
//originally 6,-13
00194     System.Drawing.Point tempgrouparr = new System.Drawing.Point(6, 19);
00195     System.Drawing.Point tempreplyarr = new System.Drawing.Point(230, 98);
00196     System.Drawing.Point tempclosearr = new System.Drawing.Point(129, 98);
00197     System.Drawing.Point tempcleararr = new System.Drawing.Point(23, 98);
00198     System.Drawing.Point txtbx_reply_temp = new System.Drawing.Point(23, 32
);
00199     System.Drawing.Point lbl_arr_temp = new System.Drawing.Point(0, -13);
00200     System.Drawing.Point picbx_arr_tmp = new System.Drawing.Point(2, 9);
00201     System.Drawing.Point picbx_ConvView_tmp = new System.Drawing.Point(296,
7);
/*This is for quiz mode*/
00202     System.Drawing.Point lblQuestionNum_arr_tmp = new System.Drawing.Point(
5, 21);
00203     System.Drawing.Point txtbxQuestion_arr_tmp = new System.Drawing.Point(3
2, 13);
00204     System.Drawing.Point btnQuizQuestionSend_arr_tmp = new System.Drawing.
Point(288, 13);
00205     System.Drawing.Point PanelQuestion_arr_tmp = new System.Drawing.Point(7
, 6);
     **** */
00206 #endregion
00207 #endregion
00208
00209
00210
00211
00212     /*This method allows for the moving down of controls when a new control
is added*/
00213     public void MoveCtrlsDown()
00214     {
00215         //move grpbs down if new question received to implement LIFO
00216         NEW_grpbx = true;
00217         timer.Enabled = true;
00218     }
00219
00220     /*This method makes the controls for the new question dynamically upon
receiving the quesiton*/
00221     public void MakeCtrls(string question){
00222         string oldTitleText = this.Text;
00223         char tmpTitle = oldTitleText[27]; //first letter of variable title
00224
00225         reply_arr[NumQuestions] = new System.Windows.Forms.Button();
00226         reply_arr[NumQuestions].BackColor = BackColorTheme;
00227         reply_arr[NumQuestions].ForeColor = ForeColorTheme;
00228         reply_arr[NumQuestions].FlatStyle = System.Windows.Forms.FlatStyle.
Standard;
00229         reply_arr[NumQuestions].Location = tempreplyarr;
00230         reply_arr[NumQuestions].Name = "reply_arr_" + NumQuestions.ToString
();
00231         reply_arr[NumQuestions].Size = new System.Drawing.Size(67, 25);
00232         reply_arr[NumQuestions].TabIndex = 2;
00233         reply_arr[NumQuestions].Text = "Repl&y";
00234         reply_arr[NumQuestions].UseVisualStyleBackColor = true;
00235         reply_arr[NumQuestions].Click += new System.EventHandler(this.
btnRepl_Click);
00236
00237         close_arr[NumQuestions] = new Button();
00238         close_arr[NumQuestions].BackColor = BackColorTheme;
00239         close_arr[NumQuestions].ForeColor = ForeColorTheme;
00240         close_arr[NumQuestions].FlatStyle = System.Windows.Forms.FlatStyle.
Standard;

```

```

00241         close_arr[NumQuestions].Location = tempclosearr;
00242         close_arr[NumQuestions].Name = "close_arr_" + NumQuestions.ToString
00243         ();
00244         close_arr[NumQuestions].Size = new System.Drawing.Size(67, 25);
00245         close_arr[NumQuestions].TabIndex = 3;
00246         close_arr[NumQuestions].Text = "Close";
00247         close_arr[NumQuestions].UseVisualStyleBackColor = true;
00248         close_arr[NumQuestions].Click += new System.EventHandler(this.
00249             btnCLS_Click);
00250         clear_arr[NumQuestions] = new Button();
00251         clear_arr[NumQuestions].BackColor = BackColorTheme;
00252         clear_arr[NumQuestions].ForeColor = ForeColorTheme;
00253         clear_arr[NumQuestions].FlatStyle = System.Windows.Forms.FlatStyle.
00254             Standard;
00255         clear_arr[NumQuestions].Location = tempcleararr;
00256         clear_arr[NumQuestions].Name = "clear_arr_" + NumQuestions.ToString
00257         ();
00258         clear_arr[NumQuestions].Size = new System.Drawing.Size(67, 25);
00259         clear_arr[NumQuestions].TabIndex = 4;
00260         clear_arr[NumQuestions].Text = "Clear";
00261         clear_arr[NumQuestions].UseVisualStyleBackColor = true;
00262         clear_arr[NumQuestions].Click += new System.EventHandler(this.
00263             btnCLR_Click);
00264         ;
00265         txtbx_reply_arr[NumQuestions] = new TextBox();
00266         txtbx_reply_arr[NumQuestions].AcceptsReturn = false;
00267         txtbx_reply_arr[NumQuestions].Location = txtbx_reply_temp;
00268         txtbx_reply_arr[NumQuestions].BorderStyle = BorderStyle.FixedSingle
00269         ;
00270         txtbx_reply_arr[NumQuestions].Multiline = true;
00271         txtbx_reply_arr[NumQuestions].Name = "txtbx_reply_arr_" +
00272             NumQuestions.ToString();
00273         txtbx_reply_arr[NumQuestions].Size = new System.Drawing.Size(274, 5
00274             );
00275         txtbx_reply_arr[NumQuestions].TabIndex = 1;
00276         txtbx_reply_arr[NumQuestions].KeyPress += new System.Windows.Forms.
00277             KeyPressEventHandler(CheckKeys);
00278         group_arr[NumQuestions] = new System.Windows.Forms.GroupBox(); ////
00279         Instantiate next index of groupbox array
00280         group_arr[NumQuestions].BackColor = BackColorTheme;
00281         group_arr[NumQuestions].ForeColor = ForeColorTheme;
00282         group_arr[NumQuestions].Location = origingrouparr; //insert new
00283         question at the top
00284         group_arr[NumQuestions].Name = "group_arr_" + NumQuestions.ToString
00285         ();
00286         group_arr[NumQuestions].Size = new System.Drawing.Size(328, 32);
00287         group_arr[NumQuestions].TabIndex = 2;
00288         group_arr[NumQuestions].TabStop = false;
00289         group_arr[NumQuestions].MouseDown += new MouseEventHandler(
00290             group_arr_MouseDown);
00291         ;
00292         lbl_arr[NumQuestions] = new Label();
00293         lbl_arr[NumQuestions].AutoSize = false;
00294         lbl_arr[NumQuestions].AutoEllipsis = true;
00295         lbl_arr[NumQuestions]. TextAlign = System.Drawing.ContentAlignment.
00296             TopCenter;
00297         lbl_arr[NumQuestions].BackColor = BackColorTheme;
00298         lbl_arr[NumQuestions].ForeColor = ForeColorTheme;
00299         lbl_arr[NumQuestions].FlatStyle = System.Windows.Forms.FlatStyle.
00300             Standard;
00301         lbl_arr[NumQuestions].Dock = DockStyle.Top;
00302         lbl_arr[NumQuestions].Location = lbl_arr_temp;
00303         lbl_arr[NumQuestions].Name = "lbl_arr_" + NumQuestions.ToString();
00304         lbl_arr[NumQuestions].Size = new System.Drawing.Size(330, 15);
00305         lbl_arr[NumQuestions].TabIndex = 0;
00306         if (showNamesToolStripMenuItem.Checked)
00307         {
00308             lbl_arr[NumQuestions].Text = "****" + " " + question +
00309             " -" + tempString2;
00310             Q_NameShowing[NumQuestions] = true; //flag for name visibility
00311         }
00312     }
00313 
```

```

00299         {
00300             lbl_arr[NumQuestions].Text = "***" + " " + question;
00301             Q_NameShowing[NumQuestions] = false;
00302         }
00303         lbl_arr[NumQuestions].MouseDown += new System.Windows.Forms.
00304             MouseEventHandler(this.lbl_question_MouseDown);
00305         lbl_arr[NumQuestions].MouseMove += new System.Windows.Forms.
00306             MouseEventHandler(this.lbl_question_MouseMove);
00307         picbx_arr[NumQuestions] = new PictureBox();
00308         picbx_arr[NumQuestions].BackColor = BackColorTheme;
00309         picbx_arr[NumQuestions].Image = global::SrP_ClassroomInq.Properties
00310             .Resources.reply_arrow;
00311         picbx_arr[NumQuestions].Location = picbx_arr_tmp;
00312         picbx_arr[NumQuestions].Name = "picbx_arr_" + NumQuestions.ToString
00313             ();
00314         picbx_arr[NumQuestions].Size = new System.Drawing.Size(24, 20);
00315         picbx_arr[NumQuestions].SizeMode = System.Windows.Forms.
00316             PictureBoxSizeMode.StretchImage;
00317         picbx_arr[NumQuestions].TabIndex = 0;
00318         picbx_arr[NumQuestions].TabStop = false;
00319         picbx_arr[NumQuestions].Visible = false; //set to visible after
00320             marked read
00321         picbx_ConvView_arr[NumQuestions] = new PictureBox();
00322         picbx_ConvView_arr[NumQuestions].BackColor = BackColorTheme;
00323         picbx_ConvView_arr[NumQuestions].Image = global::SrP_ClassroomInq.
00324             Properties.Resources.bubbles2;
00325         picbx_ConvView_arr[NumQuestions].Location = picbx_ConvView_tmp;
00326         picbx_ConvView_arr[NumQuestions].Name = "picbx_ConvView_arr_" +
00327             NumQuestions.ToString();
00328         picbx_ConvView_arr[NumQuestions].Size = new System.Drawing.Size(30,
00329             24);
00330         picbx_ConvView_arr[NumQuestions].SizeMode = System.Windows.Forms.
00331             PictureBoxSizeMode.StretchImage;
00332         picbx_ConvView_arr[NumQuestions].TabIndex = 0;
00333         picbx_ConvView_arr[NumQuestions].TabStop = false;
00334         picbx_ConvView_arr[NumQuestions].Visible = true; //set to visible
00335             after marked read
00336         picbx_ConvView_arr[NumQuestions].Click += new System.EventHandler(
00337             this.ConvViewPic_Click);
00338         picbx_ConvView_arr[NumQuestions].MouseEnter += new System.
00339             EventHandler(this.picbx_CV_MouseEnter);
00340         picbx_ConvView_arr[NumQuestions].MouseLeave += new System.
00341             EventHandler(this.picbx_CV_MouseLeave);
00342         tt_picbxCV_arr[NumQuestions] = new ToolTip();
00343         tt_picbxCV_arr[NumQuestions].BackColor = BackColorTheme;
00344         tt_picbxCV_arr[NumQuestions].ForeColor = ForeColorTheme;
00345         tt_picbxCV_arr[NumQuestions].IsBalloon = true;
00346         tt_picbxCV_arr[NumQuestions].ToolTipTitle = "Click the icon!";
00347         tt_picbxCV_arr[NumQuestions].AutoPopDelay = 2500;
00348         tt_picbxCV_arr[NumQuestions].InitialDelay = 300;
00349         tt_picbxCV_arr[NumQuestions].ReshowDelay = 1000;
00350         tt_picbxCV_arr[NumQuestions].UseFading = true;
00351         tt_picbxCV_arr[NumQuestions].UseAnimation = true;
00352         group_arr[NumQuestions].Controls.Add(reply_arr[NumQuestions]);
00353         group_arr[NumQuestions].Controls.Add(close_arr[NumQuestions]);
00354         group_arr[NumQuestions].Controls.Add(clear_arr[NumQuestions]);
00355         group_arr[NumQuestions].Controls.Add(txtbx_reply_arr[NumQuestions])
00356         ;
00357         group_arr[NumQuestions].Controls.Add(lbl_arr[NumQuestions]);
00358         group_arr[NumQuestions].Controls.Add(picbx_arr[NumQuestions]);
00359         group_arr[NumQuestions].Controls.Add(picbx_ConvView_arr[
00360             NumQuestions]);
00361         MoveCtrlsDown(); //move ctrls for lifo operation
00362         grpbxFeed.Controls.Add(group_arr[NumQuestions]);
00363         if (chkbxRXSound.Checked) //eventually have preference here
00364         {
00365             JukeBox.SoundLocation = @"blip.wav"; // this is in
00366             [projdir]/bin/debug/blip.wav
00367             JukeBox.Play();

```

```

00356         }
00357
00358     if (UnreadCount > 0)
00359     {
00360         switch (tmpTitle)
00361         {
00362             //this switch avoids the title being changed while the user
00363             is in a panel
00364                 //other than the home screen...
00365             case('H'):
00366                 oldTitleText = "Home";
00367                 break;
00368             case('P'):
00369                 oldTitleText = "Prefs";
00370                 break;
00371             case('S'):
00372                 oldTitleText = "Student Management";
00373                 break;
00374             case('C'): //Class Vote can't happen due to state machine
00375                 oldTitleText = "Conversation Viewer";
00376                 break;
00377             case('D'):
00378                 oldTitleText = "Direct Message";
00379                 break;
00380             case('Q'): //Quiz mode can't happen because of state
00381                 oldTitleText = "Quiz Maker";
00382                 break;
00383             case('F'):
00384                 oldTitleText = "FAQ";
00385                 break;
00386             default:
00387                 oldTitleText = "Home"; //it's the best guess
00388                 break;
00389         }
00390         this.Text = " Classroom Inquisition | " + oldTitleText + " (" +
00391             + UnreadCount.ToString() + ")";
00392     }
00393
00394     if (chkbxNotify.Checked)
00395     {
00396         trayICON.BalloonTipTitle = tempString2 + " asked:";
00397         trayICON.BalloonTipText = question;
00398         trayICON.ShowBalloonTip(1250); //show for 1.25 seconds
00399     }
00400     lblNotify.Text = "New Message from: " + tempString2;
00401     DesireNotify = true; //show notify panel
00402
00403     NumQuestions++; //increment number of ctrls
00404 }
00405
00406     /*This method handles the backspace character, formats questions, and
00407     fills arrays for later use*/
00408     public string HandleBCKSPC(string RAW_DATA)
00409     {
00410         string tmpString = RAW_DATA;
00411         string finalString = "";
00412         string[] strARRAY = new string[RAW_DATA.Length];
00413         char tmp=' ';
00414         char tmp_response=' ';
00415         strARRAY = tmpString.Split('\b');
00416         if (RAW_DATA.Length >= 4)//contains at least stx,addr,msg,etx,LF
00417         {
00418             foreach (string str in strARRAY)
00419             {
00420                 if (finalString.Length > 1)
00421                 {
00422                     finalString = finalString.Substring(0, (finalString.
Length - 1)) + str; //remove char to delete and concat strings
00423                 }
00424                 else if (finalString == "")
00425                 {
00426                     finalString = str; // initial value
00427                 }
00428             }
00429         }
00430     }

```

```

00425             else
00426             {
00427                 finalString += str.Substring(0, (str.Length - 1)); ///
00428                 if you fudge and backspace the second letter you type
00429             }
00430
00431             tmp = finalString[1]; //this should be the address
00432             tmp_response = finalString[2]; //if this is a flowcontrol
00433             string otherwise this would be the first char of the message
00434             finalString = finalString.TrimStart('\x02'); //clean up rx'd
00435             string start of transmit char
00436             finalString = finalString.TrimEnd('\x03'); //trim end of
00437             transmit
00438             if ((tmp_response != 0x05) && (tmp_response != 0x06)) //if not
00439             eng or ack
00440             {
00441                 switch(StateMachine){
00442                     case("ClassVote"):
00443                     case("Attendance"):
00444                         finalString = finalString.TrimStart(tmp); //trim
00445                         off the address
00446                         for (int i = 0; i < Students_ID.Length; i++)
00447                             {
00448                                 if (String.CompareOrdinal(addr_tbl[i], tmp.
00449                                     ToString().TrimStart('\x27').TrimEnd('\x27')) == 0) //CompareOrdinal Compares
00450                                     Numerical value
00451                                     tempString2 = Students_Name[i];
00452                                     PresentStuNum = (byte)i; //save for later
00453                                     break;
00454                         case("Quiz"):
00455                         finalString = finalString.TrimStart(tmp); //trim
00456                         off the address
00457                         for (int i = 0; i < Students_ID.Length; i++)
00458                             {
00459                                 if (String.CompareOrdinal(addr_tbl[i], tmp.
00460                                     ToString().TrimStart('\x27').TrimEnd('\x27')) == 0) //CompareOrdinal Compares
00461                                     Numerical value
00462                                     tempString2 = Students_Name[i];
00463                                     break;
00464
00465                         case("Normal"):
00466                         default:
00467                             Q_sender[NumQuestions] = tmp.ToString().TrimStart('
00468                             \x27').TrimEnd('\x27'); //the trims are to fix the mess ToString makes of the
00469                             unprintable char
00470                             finalString = finalString.TrimStart(tmp); //trim
00471                             off the address
00472                             for (int i = 0; i < Students_ID.Length; i++)
00473                                 {
00474                                     if (String.CompareOrdinal(addr_tbl[i], Q_sender
00475                                     [NumQuestions]) == 0) //CompareOrdinal Compares Numerical value
00476                                     tempString2 = Students_Name[i];
00477                                     break;
00478                                 }
00479
00480                             return finalString;
00481                         }
00482                         else // the string received is part of the flow control
00483                         {

```

```
00484             if (tmp_response == 0x05) //enq has been received
00485             {
00486                 FlowCtrl_Send(tmp.ToString().TrimStart('\x27').TrimEnd(
00487                     '\x27'), "ack"); //ack the enq
00488             }
00489             else //ack is received
00490             {
00491                 AckRXd = true; //proceed with the sending
00492             }
00493             return "";
00494         }
00495     }
00496     {
00497         return "";
00498     }
00499 }
00500
00501     /*This method allows for key presses to be detected and used even by
00502     controls which don't have keypress events*/
00503     public void CheckKeys(object sender, System.Windows.Forms.
00504     KeyPressEventArgs e)
00505     {
00506         if (e.KeyChar == (char)13) //enter key in windows
00507         {
00508             if (textBox1.Focused) //then it's the sender
00509             {
00510                 if ((textBox1.Text != "") && (textBox1.Text != Environment.
00511                     NewLine))
00512                 {
00513                     if (SendMsg(textBox1.Text, brdcst_addr))
00514                     {
00515                         textBox1.Clear();
00516                         grpbxFeed.Focus(); //get cursor out of text box
00517                         textbox1WASClicked = true;
00518                         timer.Enabled = true; // hide the send button
00519                     }
00520                 }
00521             }
00522         }
00523     }
00524     else
00525     {
00526         btnRepl_Click(sender, e);
00527     }
00528 }
00529 else if (e.KeyChar == (char)27) //escape key
00530 {
00531     if (BrdcstShowing)
00532     {
00533         btnCLS_Click_BrdCst(sender, e);
00534     }
00535     else if (ClassVoteShowing)
00536     {
00537         btnExitClassVote_Click(sender, e);
00538     }
00539     else if (AttendanceShowing)
00540     {
00541         btnExitAttendance_Click(sender, e);
00542     }
00543     else if (QuizModeShowing)
00544     {
00545         btnExitQuiz_Click(sender, e);
00546     }
00547     else if (QuizMakerShowing)
00548     {
00549         btnQMcls_Click(sender, e);
00550     }
00551     else if (ConvViewShowing)
00552     {
00553         ConvViewClicked = true;
```

```

00554             timer.Enabled = true;
00555         }
00556         else if (StuMgmtShowing)
00557     {
00558             StuMgmtClicked = true;
00559             timer.Enabled = true;
00560         }
00561         else if (FAQShowing)
00562     {
00563             FAQClicked = true;
00564             timer.Enabled = true;
00565         }
00566         else if (PrefsShowing)
00567     {
00568             PrefsClicked = true;
00569             timer.Enabled = true;
00570         }
00571         else if (DMPanelShowing)
00572     {
00573             DMClicked = true;
00574             timer.Enabled = true;
00575         }
00576     }
00577 }
00578
00579 /*This is the method which quits the app from the menu bar*/
00580 private void quitToolStripMenuItem_Click(object sender,
EventArgs e)
00581 {
00582     this.Close(); // there is a prompt in form closing
00583 }
00584
00585 /*This method allows for minimize to tray functionality*/
00586 private void trayICON_MouseDoubleClick(object sender,
MouseEventArgs e)
00587 {
00588     trayICON.BalloonTipTitle ="Classroom Inquistion";
00589     trayICON.BalloonTipText = "Raising Hands is a thing of the past";
00590
00591         if (FormWindowState.Minimized == this.WindowState)
00592     {
00593         this.Show();
00594         this.WindowState = FormWindowState.Normal;
00595     }
00596     else if (FormWindowState.Normal == this.WindowState)
00597     {
00598         this.Hide();
00599         trayICON.ShowBalloonTip(500);
00600         this.WindowState = FormWindowState.Minimized;
00601     }
00602 }
00603
00604
00605 /*This clears the question repl box of the specifically clicked
question*/
00606 private void btnCLR_Click(object sender, EventArgs e)
00607 {
00608     if ((NumQuestions - 1) > 0)
00609     {
00610         for (int i = 0; i < NumQuestions - 1; i++)
00611     {
00612
00613             txtbx_reply_arr[i].ResetText(); //clear text
00614
00615     }
00616
00617     else
00618     {
00619         txtbx_reply_arr[0].ResetText();
00620     }
00621
00622 }
00623 /*This handles the replies for every question in the feed box*/
00624 private void btnRepl_Click(object sender, EventArgs e)

```

```

00625      {
00626          for (int i = 0; i < lbl_arr.Length - 1; i++) //loop through to find
00627              the clicked one
00628          {
00629              if (sender.Equals(lbl_arr[i]))
00630              {
00631                  lbl_ID = i;
00632              }
00633          }
00634          addr = Q_sender[lbl_ID]; // find out who sent the question clicked
00635
00636          if ((txtbx_reply_arr[lbl_ID].Text!=Environment.NewLine) && (
00637              txtbx_reply_arr[lbl_ID].Text!="")) // we have communications and data to send
00638          {
00639              //FlowCtrl_Send(addr, "enq");
00640              // timeout_ack = false; //so that the following statement will
00641              work
00642              // while (!AckRXd && !timeout_ack) ;
00643              // if (AckRXd)
00644              // {
00645                  if (SendMsg(txtbx_reply_arr[lbl_ID].Text.TrimEnd('\x0A'),
00646                      addr)) //SendMsg returns true for a successful send
00647                  {
00648                      LogQandA(txtbx_reply_arr[lbl_ID].Text.TrimEnd('\x0A'),
00649                          true, addr); //log message
00650                      txtbx_reply_arr[lbl_ID].Clear();
00651                      btnCLS_WASclicked = true; //close question
00652                      timer.Enabled = true; // hide the send button
00653
00654                      if (!Q_Replied[lbl_ID])
00655                      {
00656                          Q_Replied[lbl_ID] = true;
00657                          picbx_arr[lbl_ID].BringToFront(); //to display on
00658                          the label
00659                          picbx_arr[lbl_ID].Visible = true; //come out from
00660                          hiding
00661                      }
00662                      //AckRXd = false; //reset
00663
00664                  }
00665                  // else
00666                  //
00667                  MessageBox.Show("Student System Not Responding." +
00668                      Environment.NewLine + "Message not sent.");
00669
00670
00671      }
00672
00673      /*This allows the click to open/close questions as well as mark read
00674      and show context menus*/
00675      private void lbl_question_MouseDown(object sender,
00676          MouseEventArgs e)
00677      {
00678          bool rightClick = (e.Button == System.Windows.Forms.MouseButtons.
00679              Right);
00680          bool leftClick = (e.Button == System.Windows.Forms.MouseButtons.
00681              Left);
00682
00683          for (int i = 0; i < lbl_arr.Length - 1; i++) //loop through to find
00684              the clicked one
00685          {
00686              if (sender.Equals(lbl_arr[i]))
00687              {
00688                  old_lblID = lbl_ID; //save in case a separate question
00689                  is clicked
00690                  new_lblID = i; //store new to use later

```

```

00685             }
00686         }
00687
00688         if (rightClick)
00689     {
00690         //show context menu at mouse click location aligned right
00691         ctxtMenu.Show(lbl_arr[new_lblID], e.Location,
00692             LeftRightAlignment.Right);
00692     }
00693     else if (leftClick)
00694     {
00695         if (rdbtnClick.Checked)
00696         {
00697             UnreadDecrement(sender);
00698         }
00699         if (timesClicked == 0)
00700         {
00701             grpbxRPL_WASclicked = true;
00702             timesClicked = 1;
00703             lbl_ID = new_lblID; //there isn't possibly one open yet
00704         }
00705     else
00706     {
00707         if (old_lblID == new_lblID)
00708         {
00709             btnCLS_WASclicked = true; //second click closes
00710             timesClicked = 0;
00711         }
00712     else
00713     {
00714         DesireID = true; //queue dynamic animations
00715         btnCLS_WASclicked = true; //second click closes
00716         question.
00717             //the animation will use lbl_ID which is the old one
00718         still
00719     }
00720     timer.Enabled = true;
00721 }
00722
00723 /*Closes the question which it relates to*/
00724     private void btnCLS_Click(object sender, EventArgs e)
00725     {
00726         timer.Enabled = true;
00727         btnCLS_WASclicked = true;
00728         timesClicked = 0;
00729     }
00730
00731 /*This method loads up prefs and other things needed to be initialized
at startup*/
00732     private void Form1_Load(object sender, EventArgs e)
00733     {
00734         this.Hide();
00735         serialCOMcmbbx_Click(sender, e); //pre-load the combobox
00736         //auto fill feed box or something... for testing if needed
00737
00738         Properties.Settings.Default.Reload();
00739
00740         /*Check the state of previous settings and reset them*/
00741         SecretKey = Properties.Settings.Default.key;
00742         chkbxLameMode.Checked = Properties.Settings.Default.Animations;
00743         chkbxRXSound.Checked = Properties.Settings.Default.SoundRX;
00744         chkbxTXSound.Checked = Properties.Settings.Default.SoundTX;
00745         rdbtnClick.Checked = Properties.Settings.Default.ClickRead;
00746         rdbtnHover.Checked = Properties.Settings.Default.HoverRead;
00747         chkbxToolTips.Checked = Properties.Settings.Default.Tooltips;
00748         showNamesToolStripMenuItem.Checked = Properties.Settings.Default.
00749             ShowNames;
00750         chkbxNotify.Checked = Properties.Settings.Default.Notify;
00751         chkbxCtrlHide.Checked = Properties.Settings.Default.CtrlHide;
00752         ForeColorTheme = Properties.Settings.Default.ForeColorTheme;
00753         BackColorTheme = Properties.Settings.Default.BackColorTheme;

```

```

00753     timer.Interval = Properties.Settings.Default.AnimationSpeed;
00754     numUpDnNotify.Value = Properties.Settings.Default.NotifyTimeout;
00755     chkbxSplash.Checked = Properties.Settings.Default.Splash;
00756     /*********************************************************************
00757
00758     ThemeApply(ForeColorTheme, BackColorTheme); //set from saved theme
00759     if (!chkbxSplash.Checked)
00760     {
00761         this.Show();
00762     }
00763
00764     string[] tmpstring = new string[classSize];
00765     try
00766     {
00767         try
00768         {
00769             using (StreamReader sr = File.OpenText(PlainData))
00770             {
00771                 string tempstr = "";
00772                 while ((tempstr = sr.ReadLine()) != null)
00773                 {
00774                     tempstr = tempstr.TrimEnd('\x0A');
00775                     tmpstring = tempstr.Split(',');
00776                     string should always be name
00777                     Students_Name[i] = tmpstring[0]; // store student
00778                     names as they are read from file
00779                     Students_ID[i++] = tmpstring[1]; //store ID's
00780                     }
00781                     sr.Close();
00782             }
00783             catch (DirectoryNotFoundException dirEx)
00784             {
00785                 Directory.CreateDirectory(@".data"); //in case the
00786                 directory didn't exist
00787             }
00788             catch (FileNotFoundException fileEx)
00789             {
00790                 File.Create(PlainData); //recreate the file
00791             }
00792             lstbxStudents.Items.Clear();
00793             for (j = 0; j < i; j++)
00794             {
00795                 lstbxStudents.Items.Add(Students_Name[j]); //intialize the
00796                 lstbx
00797             }
00798             i = j = 0;
00799         }
00800         catch
00801         {
00802             MessageBox.Show("Stuff's not right... You need permissions to
00803             the Program Files Directory.");
00804         }
00805         try
00806         {
00807             try
00808             {
00809                 i = 0;
00810                 using (StreamReader sr = File.OpenText(@"
00811                 .data\Quiz\QuizMaker.txt"))
00812                 {
00813                     string tempstr = "";
00814                     while ((tempstr = sr.ReadLine()) != null)
00815                     {
00816                         tmpstring = tempstr.Split('\x0A'); //divvy it up by
00817                         newline
00818                         logTmpString[i] = tmpstring[0]; //weird error where
00819                         tmpstring is only using index 0...this fixes it
00820                         i++;
00821                     }
00822                     sr.Close();
00823                     for (j = 0; j <= (i - 1); j++)
00824                 }
00825             }
00826         }
00827     }
00828 
```

```

00819                     {
00820                         MakeQuizQuestion(logTmpString[j], j); //populate
00821                         quiz from data file
00822                         }
00823                         i = j = 0;
00824                     }
00825                     catch (DirectoryNotFoundException dirEx)
00826                     {
00827                         Directory.CreateDirectory(@".\data\Quiz"); // recreate the
00828                         directory
00829                     }
00830                     catch (FileNotFoundException fileEX)
00831                     {
00832                         File.Create(@".\data\Quiz\QuizMaker.txt"); // recreate the
00833                         quiz maker file
00834                     }
00835                     catch
00836                     {
00837                         MessageBox.Show("Oh No's.... You need permissions to the
00838                         Program Files Directory.");
00839                     }
00840                     /*Splash Screen Code*/
00841                     if (chkbxSplash.Checked)
00842                     {
00843                         timer.Enabled = false;
00844                         timer_ControlsCreate.Enabled = false;
00845                         timer_SerialRead.Enabled = false;
00846                         while (DialogResult.OK != SplashScreen.ShowDialog()); //wait
00847                         here
00848                     }
00849                     while (frmClassrromInq.ActiveForm == SplashScreen);
00850                     ****
00851                     timer_SerialRead.Enabled = true;
00852                     this.Show();
00853                     this.Focus();
00854                     lblNotify.Text = "Ready to Rock!";
00855                     DesireNotify = true;
00856                     timer.Enabled = true;
00857                     }
00858                     /*This method shows the PrefsPanel*/
00859                     private void PrefsToolStripMenuItem_Click(object sender, EventArgs e)
00860                     {
00861                         if ((StuMgmtShowing == true) && (FAQShowing == false) &&
00862                             (DMPanelShowing == false))
00863                         {
00864                             SaveStudentData(); //if the panel is hiding save data
00865                             StuMgmtClicked = true;
00866                             timer.Enabled = true;
00867                             PanelPrefs.Focus(); //for use of esc key
00868                         }
00869                         else if ((FAQShowing == false) && (DMPanelShowing == false) && (!
00870                             BrdcstShowing))
00871                         {
00872                             PrefsClicked = true;
00873                             timer.Enabled = true;
00874                         }
00875                         else if ((FAQShowing == true) && (DMPanelShowing == false) && (!
00876                             BrdcstShowing))
00877                         {
00878                             DesirePrefs = true; //show me after hiding FAQ
00879                             FAQClicked = true; //hide
00880                             timer.Enabled = true;
00881                         }
00882                         else if ((FAQShowing == false) && (DMPanelShowing == true) && (!
00883                             BrdcstShowing))
00884                         {
00885                             DesirePrefs = true; //show me after hiding DM
00886                             DMClicked = true; //hide
00887                             timer.Enabled = true;
00888                         }
00889                     }
00890                     }

```

```

00884         }
00885         else if ((BrdcstShowing))
00886         {
00887             DesirePrefs = true;
00888             broadcastToolStripMenuItem_Click(sender, e);
00889         }
00890     }
00891 
00892     /*This method fills the serial port combo box and auto selects a port
00893     if there's only one*/
00893     private void serialCOMcmbbx_Click(object sender, EventArgs e)
00894     {
00895         byte tmp = 0;
00896         serialCOMcmbbx.Items.Clear();
00897         portNames = System.IO.Ports.SerialPort.GetPortNames();
00898         foreach (string name in portNames)
00899         {
00900             serialCOMcmbbx.Items.Add(name);
00901             tmp++;
00902         }
00903         if (tmp == 1) //there's only 1 serial port, let's connect
00904         {
00905             serialCOMcmbbx.SelectedIndex = 0; //choose the first one
00906             serialCOMcmbbx_SelectedIndexChanged(sender, e);
00907         }
00908     }
00909 
00910     /*Attempts to open the selected port from the combobox*/
00911     private void serialCOMcmbbx_SelectedIndexChanged(object sender,
EventArgs e)
00912     {
00913         if (SerialPort.IsOpen == true) //make sure we don't try to open a
port that is already
00914             SerialPort.Close();
00915 
00916         SerialPort.PortName = portNames[serialCOMcmbbx.SelectedIndex];
00917 
00918         try
00919         {
00920             SerialPort.Open();
00921         }
00922         catch (Exception E)
00923         {
00924             MessageBox.Show("Serial Port selected could not be opened..."); 
00925             SerialPort.Close();
00926         }
00927 
00928     }
00929 
00930 
00931     /*This method runs while the form is still running but close has been
issued, allows cancel of close*/
00932     private void Form1_FormClosing(object sender, FormClosingEventArgs e)
00933     {
00934         if (!sudo_kill)
00935         {
00936             string message = "Leaving so soon?";
00937 
00938             string caption = "Are you Sure you want to do that...?";
00939 
00940             MessageBoxButtons buttons = MessageBoxButtons.YesNo;
00941             DialogResult result;
00942             result = MessageBox.Show(this, message, caption, buttons,
MessageBoxIcon.Question);
00943 
00944             if (result == DialogResult.Yes)
00945             {
00946                 SerialPort.Close(); //tie up loose ends..
00947                 SavePrefs();
00948 
00949                 if (StuMgmtShowing)
00950                 {
00951                     SaveStudentData(); //if the panel is showing, on close,
save data

```

```

00952             }
00953             if (QuizMakerShowing) // you'll want to save whatever
00954             changes they've made
00955             {
00956             try
00957             {
00958                 using (StreamWriter sw = new StreamWriter(@".
00959                 .data\Quiz\QuizMaker.txt"))
00960                 {
00961                     //build and write the strings for the data file
00962                     for (int i = 0; i < lstbxQuizMaker.Items.Count;
00963                         i++)
00964                         {
00965                             sw.WriteLine(lstbxQuizMaker.Items[i]);
00966                         }
00967                     }
00968                     catch {}
00969                 }
00970             }
00971             else
00972             {
00973                 e.Cancel = true;
00974             }
00975         }
00976         else
00977         {
00978             SerialPort.Close(); //tie up loose ends..
00979             SavePrefs();
00980             if (StuMgmtShowing)
00981             {
00982                 SaveStudentData(); //if the panel is showing, on close,
00983                 save data
00984                 if (QuizMakerShowing) // you'll want to save whatever changes
00985                 they've made
00986                 {
00987                     try
00988                     {
00989                         using (StreamWriter sw = new StreamWriter(@".
00990                         .data\Quiz\QuizMaker.txt"))
00991                         {
00992                             //build and write the strings for the data file
00993                             for (int i = 0; i < lstbxQuizMaker.Items.Count; i+
00994                             )
00995                             {
00996                                 sw.WriteLine(lstbxQuizMaker.Items[i]);
00997                             }
00998                         }
00999                         catch {}
01000                     }
01001                 }
01002             }
01003             /*This shows the about information in a separate dialog*/
01004             private void aboutToolStripMenuItem_Click(object sender, EventArgs e)
01005             {
01006                 //Show About Information
01007                 About.ShowDialog();
01008             }
01009             /*This sends a broadcast message from the broadcast box*/
01010             private void btnSend_Click(object sender, EventArgs e)
01011             {
01012                 if ((textBox1.Text != "") && (textBox1.Text != Environment.NewLine)
01013                 )
01014                 {
01015                     if (SendMsg(textBox1.Text, brdcst_addr)) //Sending Function,
01016                     returns true if send succeeds

```

```

01017             {
01018                 textBox1.Clear();
01019                 grpbxFeed.Focus(); //get cursor out of textbox
01020                 textBox1WasClicked = true;
01021                 timer.Enabled = true; // hide the send button
01022             }
01023         }
01024     else
01025     {
01026         MessageBox.Show("You Haven't typed a message yet..");
01027     }
01028 }
01029
01030 /*This is a timer which polls the serial port to see if there is data*/
01031 private void timer_SerialRead_Tick(object sender, EventArgs e) //my
very own SerialDataRX'd event
01032 {
01033     string tmp_str = "";
01034     if (SerialPort.IsOpen == true)
01035     {
01036         SerialPort.ReadTimeout = 10; //in milliseconds
01037         tempString = null;
01038         try
01039         {
01040             tempString = SerialPort.ReadLine();
01041         }
01042         catch {}
01043
01044         if (tempString != null)
01045         {
01046             //MessageBox.Show(tempString); //testing
01047             RX_Data = tempString;
01048             WeGotData = true;
01049             tempString = null;
01050         }
01051
01052         if (WeGotData != false)
01053         {
01054             //we just had data, call some event
01055             tmp_str = HandleBCKSPC(RX_Data);
01056             if (tmp_str == "")// handle backspace (0x08) before
printing questions
01057             {
01058                 //if "" is true then the string received was a flwctrl
string, and it's taken care of
01059             }
01060             else // we can decide what to do based on program state
01061             {
01062                 switch (StateMachine)
01063                 {
01064                     case("Quiz"):
01065                         RecordQuizData(tmp_str, tempString2); //
question, student's name
01066                         break;
01067
01068                     case("Attendance"):
01069                         //we're in attendance mode
01070                         if (StudentPresent[PresentStuNum] == false){
01071                             lstbxAttendance.Items.Add(tempString2 + "
is here!");
01072                             StudentPresent[PresentStuNum] = true; //
prevents student's saying there here more than once
01073                         }
01074                         break;
01075
01076                     case("ClassVote"):
01077                         //in class vote mode
01078                         ClassVoteHandler(tmp_str, PresentStuNum);
01079                         break;
01080
01081                     case("Normal"): //normal is the default
01082                     default:
01083                         UnreadCount++;
01084                         Qs_to_create[ix++] = tmp_str; //store and

```

```

increment pointer
01085     Qs_to_Make = true; //set the flag
01086     timer_ControlsCreate.Enabled = true;
01087     break;
01088   }
01089 }
01090   WeGotData = false;
01091 }
01092 }
01093 }
01094 }
01095 /*This allows for unread decrement by mouse hover*/
01096 private void lbl_question_MouseMove(object sender, MouseEventArgs e)
01097 {
01098   if (rdbtnHover.Checked)
01099   {
01100     UnreadDecrement(sender);
01101   }
01102 }
01103 }
01104 /*This opens the direct message panel*/
01105 private void directMessageToolStripMenuItem_Click(object sender,
EventArgs e)
01106 {
01107   if ((StuMgmtShowing == true) && (PrefsShowing == true) && (!
BrdcstShowing))
01108   {
01109     DesireDM = true; //show after hiding stumgmt
01110     StuMgmtClicked = true; //hide
01111     DesirePrefs = true; //tells stumgmt to hide prefs
01112     timer.Enabled = true;
01113   }
01114   else if ((FAQShowing == true) && (PrefsShowing == false) && (!
BrdcstShowing))
01115   {
01116     DesireDM = true; //show me after hiding FAQ
01117     FAQClicked = true; //hide me
01118     timer.Enabled = true;
01119   }
01120   else if ((FAQShowing == false) && (PrefsShowing == true) && (!
BrdcstShowing))
01121   {
01122     DesireDM = true; //show me after hiding Prefs
01123     PrefsClicked = true; //hide me
01124     timer.Enabled = true;
01125   }
01126   else if ((BrdcstShowing))
01127   {
01128     DesireDM = true;
01129     broadcastToolStripMenuItem_Click(sender, e);
01130   }
01131   else
01132   {
01133     DMClicked = true;
01134     timer.Enabled = true;
01135   }
01136 }
01137 }
01138 /*This closes the direct message panel*/
01139 private void btnDM_Cls_Click(object sender, EventArgs e)
01140 {
01141   DMClicked = true;
01142   timer.Enabled = true;
01143 }
01144
01145 /*This opens the broadcast box*/
01146 private void broadcastToolStripMenuItem_Click(object sender, EventArgs
e)
01147 {
01148   //pn1BrdCst.BringToFront();
01149   if ((StateMachine == "Normal") && !FAQShowing && !StuMgmtShowing &&
!ConvViewShowing && !QuizMakerShowing)
01150   {
01151

```

```

01152         //only shows broadcast if at DM, Home, or prefs panel
01153         timer.Enabled = true;
01154         textBox1WASclicked = true;
01155         textBox1.Clear();
01156         textBox1.Focus();
01157     }
01158
01159     //don't show if we're not in Normal mode...
01160 }
01161
01162 /*This closes the broadcast box*/
01163 private void btnCLS_Click_BrdCst(object sender, EventArgs e)
01164 {
01165     textBox1WASclicked = true;
01166     timer.Enabled = true;
01167     this.Focus();
01168 }
01169
01170 /*This closes preferences*/
01171 private void btnPrefs_Cls_Click(object sender, EventArgs e)
01172 {
01173     SavePrefs();
01174     PrefsClicked = true;
01175     timer.Enabled = true;
01176 }
01177
01178 /*Clears the Direct message box*/
01179 private void btnDM_Clr_Click(object sender, EventArgs e)
01180 {
01181     txtbxDM.ResetText();
01182 }
01183
01184 /*This sends a direct message to the student selected in the DMcombobox
 */
01185 private void btnDM_Send_Click(object sender, EventArgs e)
01186 {
01187     //send stuff obviously
01188     if ((txtbxDM.Text != "") && (txtbxDM.Text != Environment.NewLine))
01189     {
01190         //FlowCtrl_Send(addr, "eng");
01191         //timeout_ack = false; //so that the following statement will
01192         work
01193         //while (!timeout_ack);
01194         //if (AckRXd)
01195         //{
01196             if (SendMsg(txtbxDM.Text, addr)) //Sending Function,
01197             returns true if send succeeds
01198             {
01199                 LogQandA(txtbxDM.Text, true, addr);
01200                 txtbxDM.ResetText(); //clear the textbox after sending
01201                 DMclicked = true; //to close the direct message panel
01202                 timer.Enabled = true;
01203             }
01204             // AckRXd = false;//reset
01205             //}
01206             //else
01207             //{
01208                 //    MessageBox.Show("Student System Not Responding." +
01209                 Environment.NewLine + "Message not sent.");
01210             //}
01211             else
01212             {
01213                 MessageBox.Show("You Haven't typed a message yet..");
01214             }
01215     }
01216 /*Closes the FAQ Panel*/
01217 private void btnFAQcls_Click(object sender, EventArgs e)
01218 {
01219     FAQClicked = true;
01220     timer.Enabled = true;
01221

```

```

01222         if (StuMgmtShowing)
01223         {
01224             PanelStudents.Focus();
01225         }
01226     }
01227
01228     /*Opens the FAQ Panel, button in the Help menu*/
01229     private void generalFAQToolStripMenuItem_Click(object sender, EventArgs
e)
01230     {
01231         if (BrdcstShowing)
01232         {
01233             DesireFAQ = true;
01234             broadcastToolStripMenuItem_Click(sender, e);
01235         }
01236         else
01237         {
01238             FAQClicked = true;
01239             timer.Enabled = true;
01240             PanelFAQ.Focus();
01241         }
01242     }
01243
01244     /*This allows the emailing of the developer*/
01245     private void lnklblFAQ_LinkClicked(object sender,
LinkLabelLinkClickedEventArgs e)
01246     {
01247         try
01248         {
01249             System.Diagnostics.Process.Start("
mailto:bondslaveofJesus@gmail.com");
01250         }
01251         catch { }
01252     }
01253
01254     /*This shows the Student Management Center and propagates the listbox
from the data file*/
01255     private void btnStuMgmt_Prefs_Click(object sender, EventArgs e)
01256     {
01257         string[] tmpstring = new string[classSize];
01258         int i = 0;
01259         //open StuMgmtPanel
01260         if (StuMgmtTimesClicked == 0)
01261         {
01262             //try
01263             //{
01264                 //    if (File.Exists(EncryptedData))
01265                 //{
01266                     //        DecryptFile(EncryptedData, PlainData, SecretKey);
01267                     //        File.Delete(EncryptedData); //remove old encrypted
01268                     data, on panel open
01269                     //}
01270                     //}
01271                     //catch (Exception E)
01272                     //{
01273                         //    MessageBox.Show("Decrypt failed..." + Environment.NewLine
+ E);
01274                     //}
01275                     try
01276                     {
01277                         using (StreamReader sr = File.OpenText(PlainData))
01278                         {
01279                             string tempstr = "";
01280                             while ((tempstr = sr.ReadLine()) != null)
01281                             {
01282                                 tempstr = tempstr.TrimEnd('\x0A');
01283                                 tmpstring = tempstr.Split(',');
01284                                 string should always be name
01285                                 Students_Name[i] = tmpstring[0]; // store student
01286                                 names as they are read from file
01287                                 Students_ID[i++] = tmpstring[1]; //store ID's
01288                             }
01289                         sr.Close();
01290                     }
01291
01292
01293
01294
01295
01296
01297
01298
01299
01300
01301
01302
01303
01304
01305
01306
01307
01308
01309
01310
01311
01312
01313
01314
01315
01316
01317
01318
01319
01320
01321
01322
01323
01324
01325
01326
01327
01328
01329
01330
01331
01332
01333
01334
01335
01336
01337
01338
01339
01340
01341
01342
01343
01344
01345
01346
01347
01348
01349
01350
01351
01352
01353
01354
01355
01356
01357
01358
01359
01360
01361
01362
01363
01364
01365
01366
01367
01368
01369
01370
01371
01372
01373
01374
01375
01376
01377
01378
01379
01380
01381
01382
01383
01384
01385
01386
01387
01388
01389
01390
01391
01392
01393
01394
01395
01396
01397
01398
01399
01400
01401
01402
01403
01404
01405
01406
01407
01408
01409
01410
01411
01412
01413
01414
01415
01416
01417
01418
01419
01420
01421
01422
01423
01424
01425
01426
01427
01428
01429
01430
01431
01432
01433
01434
01435
01436
01437
01438
01439
01440
01441
01442
01443
01444
01445
01446
01447
01448
01449
01450
01451
01452
01453
01454
01455
01456
01457
01458
01459
01460
01461
01462
01463
01464
01465
01466
01467
01468
01469
01470
01471
01472
01473
01474
01475
01476
01477
01478
01479
01480
01481
01482
01483
01484
01485
01486
01487
01488
01489
01490
01491
01492
01493
01494
01495
01496
01497
01498
01499
01500
01501
01502
01503
01504
01505
01506
01507
01508
01509
01510
01511
01512
01513
01514
01515
01516
01517
01518
01519
01520
01521
01522
01523
01524
01525
01526
01527
01528
01529
01530
01531
01532
01533
01534
01535
01536
01537
01538
01539
01540
01541
01542
01543
01544
01545
01546
01547
01548
01549
01550
01551
01552
01553
01554
01555
01556
01557
01558
01559
01560
01561
01562
01563
01564
01565
01566
01567
01568
01569
01570
01571
01572
01573
01574
01575
01576
01577
01578
01579
01580
01581
01582
01583
01584
01585
01586
01587
01588
01589
01590
01591
01592
01593
01594
01595
01596
01597
01598
01599
01600
01601
01602
01603
01604
01605
01606
01607
01608
01609
01610
01611
01612
01613
01614
01615
01616
01617
01618
01619
01620
01621
01622
01623
01624
01625
01626
01627
01628
01629
01630
01631
01632
01633
01634
01635
01636
01637
01638
01639
01640
01641
01642
01643
01644
01645
01646
01647
01648
01649
01650
01651
01652
01653
01654
01655
01656
01657
01658
01659
01660
01661
01662
01663
01664
01665
01666
01667
01668
01669
01670
01671
01672
01673
01674
01675
01676
01677
01678
01679
01680
01681
01682
01683
01684
01685
01686
01687
01688
01689
01690
01691
01692
01693
01694
01695
01696
01697
01698
01699
01700
01701
01702
01703
01704
01705
01706
01707
01708
01709
01710
01711
01712
01713
01714
01715
01716
01717
01718
01719
01720
01721
01722
01723
01724
01725
01726
01727
01728
01729
01730
01731
01732
01733
01734
01735
01736
01737
01738
01739
01740
01741
01742
01743
01744
01745
01746
01747
01748
01749
01750
01751
01752
01753
01754
01755
01756
01757
01758
01759
01760
01761
01762
01763
01764
01765
01766
01767
01768
01769
01770
01771
01772
01773
01774
01775
01776
01777
01778
01779
01780
01781
01782
01783
01784
01785
01786
01787
01788
01789
01790
01791
01792
01793
01794
01795
01796
01797
01798
01799
01800
01801
01802
01803
01804
01805
01806
01807
01808
01809
01810
01811
01812
01813
01814
01815
01816
01817
01818
01819
01820
01821
01822
01823
01824
01825
01826
01827
01828
01829
01830
01831
01832
01833
01834
01835
01836
01837
01838
01839
01840
01841
01842
01843
01844
01845
01846
01847
01848
01849
01850
01851
01852
01853
01854
01855
01856
01857
01858
01859
01860
01861
01862
01863
01864
01865
01866
01867
01868
01869
01870
01871
01872
01873
01874
01875
01876
01877
01878
01879
01880
01881
01882
01883
01884
01885
01886
01887
01888
01889
01890
01891
01892
01893
01894
01895
01896
01897
01898
01899
01900
01901
01902
01903
01904
01905
01906
01907
01908
01909
01910
01911
01912
01913
01914
01915
01916
01917
01918
01919
01920
01921
01922
01923
01924
01925
01926
01927
01928
01929
01930
01931
01932
01933
01934
01935
01936
01937
01938
01939
01940
01941
01942
01943
01944
01945
01946
01947
01948
01949
01950
01951
01952
01953
01954
01955
01956
01957
01958
01959
01960
01961
01962
01963
01964
01965
01966
01967
01968
01969
01970
01971
01972
01973
01974
01975
01976
01977
01978
01979
01980
01981
01982
01983
01984
01985
01986
01987
01988
01989
01990
01991
01992
01993
01994
01995
01996
01997
01998
01999
01999
02000
02001
02002
02003
02004
02005
02006
02007
02008
02009
02009
02010
02011
02012
02013
02014
02015
02016
02017
02018
02019
02019
02020
02021
02022
02023
02024
02025
02026
02027
02028
02029
02029
02030
02031
02032
02033
02034
02035
02036
02037
02038
02039
02039
02040
02041
02042
02043
02044
02045
02046
02047
02048
02049
02049
02050
02051
02052
02053
02054
02055
02056
02057
02058
02059
02059
02060
02061
02062
02063
02064
02065
02066
02067
02068
02069
02069
02070
02071
02072
02073
02074
02075
02076
02077
02078
02079
02079
02080
02081
02082
02083
02084
02085
02086
02087
02088
02089
02089
02090
02091
02092
02093
02094
02095
02096
02097
02098
02099
02099
02100
02101
02102
02103
02104
02105
02106
02107
02108
02109
02109
02110
02111
02112
02113
02114
02115
02116
02117
02118
02119
02119
02120
02121
02122
02123
02124
02125
02126
02127
02128
02129
02129
02130
02131
02132
02133
02134
02135
02136
02137
02138
02139
02139
02140
02141
02142
02143
02144
02145
02146
02147
02148
02149
02149
02150
02151
02152
02153
02154
02155
02156
02157
02158
02159
02159
02160
02161
02162
02163
02164
02165
02166
02167
02168
02169
02169
02170
02171
02172
02173
02174
02175
02176
02177
02178
02179
02179
02180
02181
02182
02183
02184
02185
02186
02187
02188
02189
02189
02190
02191
02192
02193
02194
02195
02196
02197
02198
02199
02199
02200
02201
02202
02203
02204
02205
02206
02207
02208
02209
02209
02210
02211
02212
02213
02214
02215
02216
02217
02218
02219
02219
02220
02221
02222
02223
02224
02225
02226
02227
02228
02229
02229
02230
02231
02232
02233
02234
02235
02236
02237
02238
02239
02239
02240
02241
02242
02243
02244
02245
02246
02247
02248
02249
02249
02250
02251
02252
02253
02254
02255
02256
02257
02258
02259
02259
02260
02261
02262
02263
02264
02265
02266
02267
02268
02269
02269
02270
02271
02272
02273
02274
02275
02276
02277
02278
02279
02279
02280
02281
02282
02283
02284
02285
02286
02287
02288
02289
02289
02290
02291
02292
02293
02294
02295
02296
02297
02298
02299
02299
02300
02301
02302
02303
02304
02305
02306
02307
02308
02309
02309
02310
02311
02312
02313
02314
02315
02316
02317
02318
02319
02319
02320
02321
02322
02323
02324
02325
02326
02327
02328
02329
02329
02330
02331
02332
02333
02334
02335
02336
02337
02338
02339
02339
02340
02341
02342
02343
02344
02345
02346
02347
02348
02349
02349
02350
02351
02352
02353
02354
02355
02356
02357
02358
02359
02359
02360
02361
02362
02363
02364
02365
02366
02367
02368
02369
02369
02370
02371
02372
02373
02374
02375
02376
02377
02378
02379
02379
02380
02381
02382
02383
02384
02385
02386
02387
02388
02389
02389
02390
02391
02392
02393
02394
02395
02396
02397
02398
02399
02399
02400
02401
02402
02403
02404
02405
02406
02407
02408
02409
02409
02410
02411
02412
02413
02414
02415
02416
02417
02418
02419
02419
02420
02421
02422
02423
02424
02425
02426
02427
02428
02429
02429
02430
02431
02432
02433
02434
02435
02436
02437
02438
02439
02439
02440
02441
02442
02443
02444
02445
02446
02447
02448
02449
02449
02450
02451
02452
02453
02454
02455
02456
02457
02458
02459
02459
02460
02461
02462
02463
02464
02465
02466
02467
02468
02469
02469
02470
02471
02472
02473
02474
02475
02476
02477
02478
02479
02479
02480
02481
02482
02483
02484
02485
02486
02487
02488
02489
02489
02490
02491
02492
02493
02494
02495
02496
02497
02498
02499
02499
02500
02501
02502
02503
02504
02505
02506
02507
02508
02509
02509
02510
02511
02512
02513
02514
02515
02516
02517
02518
02519
02519
02520
02521
02522
02523
02524
02525
02526
02527
02528
02529
02529
02530
02531
02532
02533
02534
02535
02536
02537
02538
02539
02539
02540
02541
02542
02543
02544
02545
02546
02547
02548
02549
02549
02550
02551
02552
02553
02554
02555
02556
02557
02558
02559
02559
02560
02561
02562
02563
02564
02565
02566
02567
02568
02569
02569
02570
02571
02572
02573
02574
02575
02576
02577
02578
02579
02579
02580
02581
02582
02583
02584
02585
02586
02587
02588
02589
02589
02590
02591
02592
02593
02594
02595
02596
02597
02598
02599
02599
02600
02601
02602
02603
02604
02605
02606
02607
02608
02609
02609
02610
02611
02612
02613
02614
02615
02616
02617
02618
02619
02619
02620
02621
02622
02623
02624
02625
02626
02627
02628
02629
02629
02630
02631
02632
02633
02634
02635
02636
02637
02638
02639
02639
02640
02641
02642
02643
02644
02645
02646
02647
02648
02649
02649
02650
02651
02652
02653
02654
02655
02656
02657
02658
02659
02659
02660
02661
02662
02663
02664
02665
02666
02667
02668
02669
02669
02670
02671
02672
02673
02674
02675
02676
02677
02678
02679
02679
02680
02681
02682
02683
02684
02685
02686
02687
02688
02689
02689
02690
02691
02692
02693
02694
02695
02696
02697
02698
02699
02699
02700
02701
02702
02703
02704
02705
02706
02707
02708
02709
02709
02710
02711
02712
02713
02714
02715
02716
02717
02718
02719
02719
02720
02721
02722
02723
02724
02725
02726
02727
02728
02729
02729
02730
02731
02732
02733
02734
02735
02736
02737
02738
02739
02739
02740
02741
02742
02743
02744
02745
02746
02747
02748
02749
02749
02750
02751
02752
02753
02754
02755
02756
02757
02758
02759
02759
02760
02761
02762
02763
02764
02765
02766
02767
02768
02769
02769
02770
02771
02772
02773
02774
02775
02776
02777
02778
02779
02779
02780
02781
02782
02783
02784
02785
02786
02787
02788
02789
02789
02790
02791
02792
02793
02794
02795
02796
02797
02798
02799
02799
02800
02801
02802
02803
02804
02805
02806
02807
02808
02809
02809
02810
02811
02812
02813
02814
02815
02816
02817
02818
02819
02819
02820
02821
02822
02823
02824
02825
02826
02827
02828
02829
02829
02830
02831
02832
02833
02834
02835
02836
02837
02838
02839
02839
02840
02841
02842
02843
02844
02845
02846
02847
02848
02849
02849
02850
02851
02852
02853
02854
02855
02856
02857
02858
02859
02859
02860
02861
02862
02863
02864
02865
02866
02867
02868
02869
02869
02870
02871
02872
02873
02874
02875
02876
02877
02878
02879
02879
02880
02881
02882
02883
02884
02885
02886
02887
02888
02889
02889
02890
02891
02892
02893
02894
02895
02896
02897
02898
02899
02899
02900
02901
02902
02903
02904
02905
02906
02907
02908
02909
02909
02910
02911
02912
02913

```

```
01288         lstbxStudents.Items.Clear();
01289         for (j = 0; j < i; j++)
01290         {
01291             lstbxStudents.Items.Add(Students_Name[j]); //repopulate
01292             the lstbx
01293         }
01294         //File.Delete(PlainData); //otherwise while in the panel
01295         decrypted sensitive data would be vulnerable..
01296         i = j = 0;
01297     }
01298     catch (Exception E)
01299     {
01300         MessageBox.Show("File open for filling Listbox Failed..." +
01301         Environment.NewLine + E);
01302     }
01303 }
01304 if (BrdcstShowing)
01305 {
01306     DesireStuMgmt = true;
01307     broadcastToolStripMenuItem_Click(sender, e);
01308 }
01309 else
01310 {
01311     StuMgmtClicked = true;
01312     timer.Enabled = true;
01313 }
01314
01315 }
01316
01317 /*This closes the Student Management Center*/
01318 private void btnCLSStudents_Click(object sender, EventArgs e)
01319 {
01320     StuMgmtClicked = true;
01321     timer.Enabled = true;
01322
01323     SaveStudentData();
01324     PanelPrefs.Focus();
01325 }
01326
01327 /*This saves the information which may have been changed about the
01328 students in StuMgmt*/
01329 public void SaveStudentData()
01330 {
01331     if (File.Exists(PlainData) == false)
01332     {
01333         File.Create(PlainData);
01334     }
01335
01336     try
01337     {
01338         using (StreamWriter sw = new StreamWriter(PlainData))
01339         {
01340             //build and write the strings for the data file
01341             for (int i = 0; i < lstbxStudents.Items.Count; i++)
01342             {
01343                 sw.WriteLine(Students_Name[i] + ',' + Students_ID[i]);
01344             }
01345             sw.Flush();
01346             sw.Close();
01347         }
01348     }
01349     catch
01350     {
01351         MessageBox.Show("You need permissions to the Program Files \n
01352         directory for Classroom Inquisition");
01353     }
01354     //SecretKey = GenerateKey();
01355     //GCHandle gch = GCHandle.Alloc(SecretKey, GCHandleType.Pinned);
```

```

01359
01361      //EncryptFile(PlainData, EncryptedData, SecretKey);
01362
01363      //Properties.Settings.Default.key = SecretKey;
01364      //Properties.Settings.Default.Save();
01365
01366      //ZeroMemory(gch.AddrOfPinnedObject(), SecretKey.Length);
01368      //gch.Free();
01369
01370      //File.Delete(PlainData); //get rid of the unencrypted data
01371  }
01372
01373  //courtesy of msdn: http://support.microsoft.com/kb/307010
01374  #region Crypto from Msdn
01375  static string GenerateKey()
01376  {
01377      //create an instance of Symetric Algorithm. Key and IV are
01378      //generated automatically.
01379      DESCryptoServiceProvider desCrypto = (DESCryptoServiceProvider)
01380      DESCryptoServiceProvider.Create();
01381
01382      //Use the Automatically generated key for encryption..
01383      return ASCIIEncoding.ASCII.GetString(desCrypto.Key);
01384
01385  static void EncryptFile(string sInputFilename, string sOutputFilename,
01386  string sKey)
01387  {
01388      //input filestream
01389      FileStream fsInput = new FileStream(sInputFilename, FileMode.Open,
01390      FileAccess.Read);
01391      //output filestream
01392      FileStream fsEncrypted = new FileStream(sOutputFilename, FileMode.
01393      Create, FileAccess.Write);
01394
01395      //give the cryptographic provider the secret key and init vector
01396      //otherwise it will make them up...
01397      DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey);
01398      DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey);
01399
01400      //create a cryptostream class using the previous cryptographic
01401      //provider and existing filestream obj
01402      ICryptoTransform desencrypt = DES.CreateEncryptor();
01403      CryptoStream cryptoStream = new CryptoStream(fsEncrypted,
01404      desencrypt, CryptoStreamMode.Write);
01405
01406      //read in the input and write the encrypted output file by passing
01407      //it to cryptostream
01408      byte[] bytearrayinput = new byte[fsInput.Length];
01409      fsInput.Read(bytearrayinput, 0, bytearrayinput.Length);
01410      cryptoStream.Write(bytearrayinput, 0, bytearrayinput.Length);
01411
01412
01413  static void DecryptFile(string sInputFilename, string sOutputFilename,
01414  string sKey)
01415  {
01416      DESCryptoServiceProvider DES = new DESCryptoServiceProvider();
01417      DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey);
01418      DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey);
01419
01420      //create file stream to read the encrypted file back
01421      FileStream fsread = new FileStream(sInputFilename, FileMode.Open,
01422      FileAccess.Read);
01423
01424      //create aes decryptor from the aes instance
01425      ICryptoTransform desdecrypt = DES.CreateDecryptor();
01426
01427      //create crypto stream set to read and do a aes decryption
01428      //transform on incoming bytes
01429      CryptoStream cryptostreamDecr = new CryptoStream(fsread, desdecrypt

```

```
    , CryptoStreamMode.Read);
01423
01424     //prints the contents of the decrypted file
01425     Stream fsDecrypted = new StreamWriter(sOutputFilename);
01426     fsDecrypted.Write(new StreamReader(cryptostreamDecr).ReadToEnd());
01427     fsDecrypted.Flush();
01428     fsDecrypted.Close();
01429 }
01430
01431     // Call this function to remove the key from memory after use for
01432     // security.
01433     [System.Runtime.InteropServices.DllImport("KERNEL32.DLL", EntryPoint =
01434     "RtlZeroMemory")]
01435     public static extern bool ZeroMemory(IntPtr Destination, int Length);
01436
01437     /*This allows the editing of student names*/
01438     private void btnEDTStudents_Click(object sender, EventArgs e)
01439     {
01440         //edit the currently selected student in Listbox, messagebox to
01441         //select one if they haven't yet
01442         if (lstbxStudents.SelectedIndex == -1)
01443         {
01444             MessageBox.Show("You haven't selected a Student to edit yet.");
01445         }
01446         else
01447         {
01448             txtbxStudentsName.Text = Students_Name[lstbxStudents.
01449             SelectedIndex];
01450             txtbxStudentsName.Focus();
01451             grpbxEDITStudents.Text = "Editing Student " + (lstbxStudents.
01452             SelectedIndex + 1).ToString();
01453         }
01454
01455     /*This allows for the resetting of the name to default*/
01456     private void btnCLRStudents_Click(object sender, EventArgs e)
01457     {
01458         //reset the student name element to the default StudentNum...
01459         if (lstbxStudents.SelectedIndex == -1)
01460         {
01461             MessageBox.Show("You haven't selected a Student to edit yet.");
01462         }
01463         else
01464         {
01465             Students_Name[lstbxStudents.SelectedIndex] = "Student " + (
01466             lstbxStudents.SelectedIndex + 1).ToString();
01467             lstbxStudents.Items.Insert(lstbxStudents.SelectedIndex,
01468             Students_Name[lstbxStudents.SelectedIndex]);
01469             lstbxStudents.Items.RemoveAt(lstbxStudents.SelectedIndex);
01470             lstbxStudents.Refresh();
01471         }
01472         PanelStudents.Focus(); //for esc to work
01473     }
01474
01475     /*This completes the rename*/
01476     private void btnDoneStudents_Click(object sender, EventArgs e)
01477     {
01478         //save the changes made in the textbox to the name back into the
01479         //database
01480         Students_Name[lstbxStudents.SelectedIndex] = txtbxStudentsName.Text
01481         ;
01482         lstbxStudents.Items.Insert(lstbxStudents.SelectedIndex,
01483         txtbxStudentsName.Text);
01484         lstbxStudents.Items.RemoveAt(lstbxStudents.SelectedIndex);
01485         txtbxStudentsName.Text = "<----- Select a Student";
01486         lstbxStudents.Refresh();
01487         grpbxEDITStudents.Text = "Student #";
01488
01489         PanelStudents.Focus();
01490     }
01491 }
```

```

01486     /*Propagates the combobox from the student data file*/
01487     private void cmbxDM_Click(object sender, EventArgs e)
01488     {
01489         string[] tmpstring = new string[classSize];
01490         try
01491         {
01492             int i = 0;
01493             using (StreamReader sr = File.OpenText(PlainData))
01494             {
01495                 string tempstr = "";
01496                 while ((tempstr = sr.ReadLine()) != null)
01497                 {
01498                     tempstr = tempstr.TrimEnd('\x0A');
01499                     tmpstring = tempstr.Split(',');
01500                     string should always be name
01501                     Students_Name[i] = tmpstring[0]; // store student names
01502                     as they are read from file
01503                     Students_ID[i++] = tmpstring[1]; //store ID's
01504                 }
01505                 cmbxDM.Items.Clear();
01506                 for (j = 0; j < i; j++)
01507                 {
01508                     cmbxDM.Items.Add(Students_Name[j]); //repopulate the lstbx
01509                 }
01510                 i = j = 0;
01511             }
01512             catch (Exception E)
01513             {
01514                 MessageBox.Show("File open for filling Combobox Failed..." +
Environment.NewLine + E);
01515             }
01516         }
01517     }
01518
01519     /*This selects to student to be messaged and set the address variable
appropriately*/
01520     private void cmbxDM_SelectedIndexChanged(object sender, EventArgs e)
01521     {
01522         if (cmbxDM.SelectedIndex != -1)
01523         {
01524             addr = addr_tbl[cmbxDM.SelectedIndex];
01525         }
01526         DirectMsgPanel.Focus(); //so esc key will work
01527     }
01528
01529     /*This is intended to provide a software flow control (but isn't used
yet...)*/
01530     private void FlowCtrl_Send(string Address_Student, string desire)
01531     {
01532         //0x05 = Enquiry 0x06= Acknowledgement
01533         string flwctrl_send = "";
01534         if (desire == "enq")
01535         {
01536             flwctrl_send = "\x05"; //enquire
01537         }
01538         else
01539         {
01540             flwctrl_send = "\x06"; //reply with Ack
01541         }
01542
01543         if (SerialPort.IsOpen == true)
01544         {
01545             SerialPort.WriteLine('\x02' + Address_Student + flwctrl_send +
01546             '\x03'); //Enquire if it's clear to send
01547         }
01548         else
01549         {
01550             MessageBox.Show("You must select a COM port before sending
communications.");
01551         }
01552     }

```

```

01553     /*This creates the controls in a timely manner so that they don't get
01554      created on top of one another */
01555      private void timer_ControlsCreate_Tick(object sender, EventArgs e)
01556      { //this event goes off when timer enabled, every 1 seconds
01557          if (Qs_to_Make)
01558          {
01559              if (ix > jx) // we haven't made all the controls yet
01560              {
01561                  LogQandA(Qs_to_create[jx], false, Q_sender[jx]);
01562                  MakeCtrls(Qs_to_create[jx++]); //make controls and
01563                  increment creation counter
01564                  picbx_ConvView_arr[jx-1].BringToFront(); //shows the "show
01565                  conv view pic"
01566              }
01567              else
01568              {
01569                  Qs_to_Make = false; //stop trying to make things
01570              }
01571          }
01572          else //No more changes to make!
01573          {
01574              timer_ControlsCreate.Enabled = false;
01575          }
01576
01577     /*The method is responsible for sending all the serial messages*/
01578     public bool SendMsg(string Message2Send, string Address2Send)
01579     {
01580         bool SuccessOfSending = new bool();
01581         if (SerialPort.IsOpen == true)
01582         {
01583             SerialPort.WriteLine('\x02' + Address2Send + Message2Send + '\x
01584             03'); //framing with stx addr message etx
01585             SuccessOfSending = true;
01586             if (chkbxTXSound.Checked)
01587             {
01588                 JukeBox.SoundLocation = @"SentMsg.wav"; // this is in
01589                 [projdir]/bin/debug/blip.wav
01590                 JukeBox.Play();
01591             }
01592             else
01593             {
01594                 MessageBox.Show("You must select a COM port before sending
01595                 communications.");
01596                 SuccessOfSending = false;
01597             }
01598             lblNotify.Text = SuccessOfSending ? "Message Sent" : "Message
01599             Sending Failed!";
01600             DesireNotify = true;
01601             return SuccessOfSending; //lets sender know if sending succeeded
01602         }
01603
01604     /*Save the preferences whenever called*/
01605     public void SavePrefs()
01606     {
01607         Properties.Settings.Default.Animations = chkbxLameMode.Checked;
01608         Properties.Settings.Default.SoundRX = chkbxRXSound.Checked;
01609         Properties.Settings.Default.SoundTX = chkbxTXSound.Checked;
01610         Properties.Settings.Default.ClickRead = rdbtnClick.Checked;
01611         Properties.Settings.Default.HoverRead = rdbtnHover.Checked;
01612         Properties.Settings.Default.Tooltips = chkbxToolTips.Checked;
01613         Properties.Settings.Default.ShowNames = showNamesToolStripMenuItem.
01614             Checked;
01615         Properties.Settings.Default.Notify = chkbxNotify.Checked;
01616         Properties.Settings.Default.CtrlHide = chkbxCtrlHide.Checked;
01617         Properties.Settings.Default.ForeColorTheme = ForeColorTheme;
01618         Properties.Settings.Default.BackColorTheme = BackColorTheme;
01619         Properties.Settings.Default.AnimationSpeed = (byte)timer.Interval;
01620         Properties.Settings.Default.NotifyTimeout = (decimal)numUpDnNotify.
01621             Value;
01622         Properties.Settings.Default.Splash = chkbxSplash.Checked;
01623         Properties.Settings.Default.Save();
01624     }

```

```

01618     }
01619
01620     /*This method decrements the unread count smartly*/
01621     public void UnreadDecrement(object sender)
01622     {
01623         char tmp = (char)UnreadCount; //contains old value
01624
01625         for (int i = 0; i < lbl_arr.Length - 1; i++) //loop through to find
01626         the clicked one
01627         {
01628             if (sender.Equals(lbl_arr[i]))
01629             {
01630                 lbl_ID_2 = i;
01631             }
01632
01633             if (Q_status[lbl_ID_2] != true) //if status is true then it has
01634             already been read
01635             {
01636                 Q_status[lbl_ID_2] = true; // so unread won't get decremented
01637             if opened again..
01638                 UnreadCount--;
01639                 lbl_arr[lbl_ID_2].Text = lbl_arr[lbl_ID_2].Text.TrimStart('*').
01640                 TrimStart(' '); //removes unread indicator in the label
01641
01642                 if (UnreadCount > 0)
01643                 {
01644                     this.Text = " Classroom Inquisition | Home (" +
01645                     UnreadCount.ToString() + ")";
01646                 }
01647             }
01648         }
01649
01650     /*This allows the user to submit issues from the FAQ*/
01651     private void lnklblIssues_LinkClicked(object sender,
01652     LinkLabelLinkClickedEventArgs e)
01653     {
01654         try
01655         {
01656             System.Diagnostics.Process.Start("
01657             https://github.com/dchriste/Classroom-Inquisition/issues");
01658         }
01659     }
01660
01661     /*This method logs all questions and answers*/
01662     public bool LogQandA(string dialog, bool teacher, string recipient)
01663     {
01664         bool success = false;
01665         byte student = 0;
01666         string name2write = "";
01667         try
01668         {
01669             for (int i = 0; i < Students_ID.Length; i++) //loop to find
01670             student name by comparing addresses
01671             {
01672                 if (String.CompareOrdinal(addr_tbl[i], recipient) == 0) ///
01673                 CompareOrdinal Compares Numerical value
01674                 {
01675                     name2write = Students_Name[i]; //student name
01676                     student = (byte)i; //for use as a index below
01677                     break;
01678                 }
01679
01680                 //make the generic filename
01681                 logFiles[student] = @"logs\Student" + student.ToString() + "
01682                 -log.txt"; //Student#-log.txt in hidden folder .logs
01683                 if (Directory.Exists(@"logs"))
01684                 {

```

```

01682             if (File.Exists(logFiles[student]) == false)
01683             {
01684                 using (FileStream fs = File.Create(logFiles[student]))
01685                 {
01686                     fs.Close();
01687                 }
01688             }
01689         }
01690     else
01691     {
01692         Directory.CreateDirectory(@".logs");
01693         using (FileStream fs = File.Create(logFiles[student])) //if
01694             folder is gone so was the file
01695             {
01696                 fs.Close();
01697             }
01698         if (!teacher)
01699         {
01700             //then I have the name already from the loop above.
01701         }
01702         else
01703         {
01704             name2write = "Teacher";
01705         }
01706
01707         using (StreamWriter sw = File.AppendText(logFiles[student]))
01708         {
01709             sw.NewLine = "\x0A"; //sets the writeline terminator
01710             sw.WriteLine(name2write + " (" + DateTime.Now.ToString("HH:mm:ss") + " | " + DateTime.Today.ToString("d") + " )" + ":" + "\x0A" +
01711             dialog + "\x0A"); //this will leave one blank line between each message
01712             /*Example:
01713             * Teacher (13:10:12 | 4/1/2012):
01714             * It's okay johnny, you're dad will be here soon.. April
01715             * ...
01716             */
01717             sw.Flush();
01718             sw.Close();
01719         }
01720         success = true;
01721     }
01722     catch
01723     {
01724         success = false;
01725         MessageBox.Show("Log creation // writing failed!!!\nYou need
permissions to the ClassroomInq Program Files directory.");
01726     }
01727     return success;
01728 }
01729 /*This closes the conversation view*/
01730 private void btnCLS_ConvView_Click(object sender, EventArgs e)
01731 {
01732     //hide the conversation view...
01733     ConvViewClicked = true;
01734     timer.Enabled = true;
01735 }
01736
01737 /*This opens and fills the conversation view*/
01738 private void btnConvView_Click(object sender, EventArgs e)
01739 {
01740     //show the conversation viewer for the selected student.....
01741     string[] tmpstring = new string[logHistoryLength];
01742     if (lstbxStudents.SelectedIndex == -1)
01743     {
01744         MessageBox.Show("You haven't selected a Student to read yet.");
01745     }
01746     else
01747     {
01748         try
01749         {
01750             if (File.Exists(@".logs\Student" + lstbxStudents.

```

```

        SelectedIndex.ToString() + "-log.txt"))
01751            {
01752                i = 0;
01753                using (StreamReader sr = File.OpenText(@".\logs\Student"
+ lstbxStudents.SelectedIndex.ToString() + "-log.txt")) //dynamically pick the
logfile
01754            {
01755                string tempstr = "";
01756                while ((tempstr = sr.ReadLine()) != null)
01757                {
01758                    tmpstring = tempstr.Split('\x0A'); //divvy it
up by newline
01759                    logTmpString[i] = tmpstring[0]; //weird error
where tmpstring is only using index 0...this fixes it
01760                    i++;
01761                }
01762                sr.Close();
01763                lstbxConvView.Items.Clear(); //clean up last view
01764                for (j = 0; j < (i - 1); j++)
01765                {
01766                    lstbxConvView.Items.Add(logTmpString[j]); //
repopulate the lstbx
01767                }
01768                i = j = 0;
01769            }
01770        }
01771        else
01772        {
01773            lstbxConvView.Items.Clear();
01774            lstbxConvView.Items.Add("There isn't a conversation to
view with this student.");
01775        }
01776    }
01777    catch (Exception E)
01778    {
01779        MessageBox.Show("This is embarrassing..there were Errors..." +
Environment.NewLine + E);
01780    }
01781    if (BrdcstShowing)
01782    {
01783        DesireConvView = true;
01784        broadcastToolStripMenuItem_Click(sender, e);
01785    }
01786    else
01787    {
01788        ConvViewClicked = true;
01789        timer.Enabled = true;
01790    }
01791}
01792}
01793}
01794}
01795}
01796 /*Allows for esc key to be able to exit the panel normally, see
checkkeys()*/
01797 private void lstbxConvView_MouseLeave(object sender, EventArgs e)
01798 {
01799     PanelConvView.Focus();
01800 }
01801
/*This method allows for conversation view to be opened dynamically per
question*/
01802 private void ConvViewPic_Click(object sender, EventArgs e)
01803 {
01804     /*Figure out who sent the message and select their student in mgmt
for even call to work*/
01805     byte student = 0;
01806     for (int i = 0; i < picbx_ConvView_arr.Length - 1; i++) //loop
through to find the clicked one
01807     {
01808         if (sender.Equals(picbx_ConvView_arr[i]))
01809         {
01810             student = (byte) i;
01811             break;
01812

```

```

01813             }
01814         }
01815         for (int i = 0; i < Students_ID.Length; i++) //loop to find student
01816             name by comparing addresses
01817             {
01818                 if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
01819                     //CompareOrdinal Compares Numerical value
01820                     {
01821                         lstbxStudents.SetSelected(i, true); //selects the correct
01822                         item so the next event call succeeds
01823                         break;
01824                     }
01825             }
01826             ****
01827             btnConvView_Click(sender, e);
01828
01829     /*This method dynamically shows tooltips for questions when the mouse
01830 enters the cv pic*/
01831     private void picbx_CV_MouseEnter(object sender, EventArgs e)
01832     {
01833         if (chkbxTooltips.Checked)
01834         {
01835             byte student = 0;
01836             for (int i = 0; i < picbx_ConvView_arr.Length - 1; i++) //loop
01837             through to find the clicked one
01838             {
01839                 if (sender.Equals(picbx_ConvView_arr[i]))
01840                 {
01841                     student = (byte)i;
01842                     break;
01843                 }
01844             }
01845             /*The below is a workaround for a tooltip bug which doesn't set
01846             the relative location until after showing/
01847             tt_picbxCV_arr[student].Show("To open the Conversation Viewer",
01848             picbx_ConvView_arr[student], 29, 20, 1);
01849             tt_picbxCV_arr[student].Show("To open the Conversation Viewer",
01850             picbx_ConvView_arr[student], 29, 20, 5000);
01851             ****
01852         }
01853     }
01854     /*This method dynamically hides tooltips for questions when the mouse
01855 leaves the cv pic*/
01856     private void picbx_CV_MouseLeave(object sender, EventArgs e)
01857     {
01858         if (chkbxTooltips.Checked)
01859         {
01860             byte student = 0;
01861             for (int i = 0; i < picbx_ConvView_arr.Length - 1; i++) //loop
01862             through to find the clicked one
01863             {
01864                 if (sender.Equals(picbx_ConvView_arr[i]))
01865                 {
01866                     student = (byte)i;
01867                     break;
01868                 }
01869             }
01870             tt_picbxCV_arr[student].Hide(picbx_ConvView_arr[student]); //
01871             make it hide if your not hovering
01872         }
01873     }
01874     /*This method preps the printstream for the print page method use later
01875 */
01876     private void btnCV_Print_Click(object sender, EventArgs e)
01877     {
01878         //helped by:
01879         http://msdn.microsoft.com/en-us/library/system.drawing.printing.printdocument(v=vs.100).aspx
01880         if (File.Exists(@".\logs\Student" + lstbxStudents.SelectedIndex.
01881 ToString() + "-log.txt"))
01882         {

```

```

01872             try
01873             {
01874                 Stream2Print = new StreamReader(@".logs\Student" +
01875                 lstbxStudents.SelectedIndex.ToString() + "-log.txt"); //dynamically pick the logfile
01876                 //print the currently showing log file
01877                 PrintFont = new Font("Arial", 12);
01878                 Printer.DocumentName = "ClassroomInq_Conversation";
01879                 Printer.OriginAtMargins = true;
01880                 Printer.Print();
01881             }
01882             catch (Exception E)
01883             {
01884                 MessageBox.Show("This is embarrassing..there were Errors..." +
01885                     + Environment.NewLine + E);
01886             }
01887             finally
01888             {
01889                 Stream2Print.Close();
01890             }
01891         else
01892         {
01893             lstbxConvView.Items.Clear();
01894             lstbxConvView.Items.Add("There isn't a conversation to print.");
01895         }
01896     }
01897 }
01898
/*This method allows the refresh of the conversation view*/
01900 private void btnCV_Refresh_Click(object sender, EventArgs e)
01901 {
    //re-read the log file showing to see if there were changes and
01902 update
01903     string[] tmpstring = new string[logHistoryLength];
01904
01905     try
01906     {
01907         if (File.Exists(@".logs\Student" + lstbxStudents.SelectedIndex.
01908             ToString() + "-log.txt"))
01909         {
01910             i = 0;
01911             using (StreamReader sr = File.OpenText(@".logs\Student" +
01912                 lstbxStudents.SelectedIndex.ToString() + "-log.txt")) //dynamically pick the
01913                 logfile
01914             {
01915                 string tempstr = "";
01916                 while ((tempstr = sr.ReadLine()) != null)
01917                 {
01918                     tmpstring = tempstr.Split('\x0A'); //divvy it up by
01919                     newline
01920                     logTmpString[i] = tmpstring[0]; //weird error where
01921                     tmpstring is only using index 0...this fixes it
01922                     i++;
01923                 }
01924             }
01925             sr.Close();
01926             lstbxConvView.Items.Clear(); //clean up last view
01927             for (j = 0; j < (i - 1); j++)
01928             {
01929                 lstbxConvView.Items.Add(logTmpString[j]); //
01930             }
01931             repopulate the lstbx
01932             }
01933             else
01934             {
01935                 lstbxConvView.Items.Clear();
01936                 lstbxConvView.Items.Add("There isn't a conversation to view
01937                 with this student.");
01938             }
01939         }
01940     }

```

```
01935         catch (Exception E)
01936         {
01937             MessageBox.Show("This is embarrassing..there were Errors..." +
01938                 Environment.NewLine + E);
01939         }
01940
01941     /*This method allows the teacher to print a conversation log file if he
01942     /she so chooses*/
01943     private void Printer_PrintPage(object sender, System.Drawing.Printing.
01944     PrintPageEventArgs e)
01945     {
01946         //adapted from:
01947         //http://msdn.microsoft.com/en-us/library/system.drawing.printing.printdocument(v=vs.100).aspx
01948         float linesPerPage = 0;
01949         float yPosition = 0;
01950         int count = 0;
01951         float leftMargin = e.MarginBounds.Left;
01952         float topMargin = e.MarginBounds.Top / 3;
01953
01954         try
01955         {
01956             // Calculate the number of lines per page.
01957             linesPerPage = e.MarginBounds.Height / PrintFont.GetHeight(
01958                 e.Graphics);
01959
01960             // Print each line of the file.
01961             while ((count < linesPerPage) && ((line = Stream2Print.
01962             ReadLine()) != null))
01963             {
01964                 yPosition = topMargin + (count * PrintFont.GetHeight(e.
01965                     Graphics));
01966                 e.Graphics.DrawString(line, PrintFont, Brushes.Black,
01967                     leftMargin, yPosition, new StringFormat());
01968                 count++;
01969             }
01970             // If more lines exist, print another page.
01971             if (line != null)
01972             {
01973                 e.HasMorePages = true;
01974             }
01975             else
01976             {
01977                 e.HasMorePages = false;
01978             }
01979         }
01980     }
01981
01982
01983     /*This method allows the user to delete a question from the feed*/
01984     private void muItmDelete_Click(object sender, EventArgs e)
01985     {
01986         byte student = (byte)new_lblID;
01987         string caption = "";
01988
01989         for (i = 0; i < Students_Name.Length - 1; i++)
01990         {
01991             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
01992             {
01993                 student = (byte) i;//lock in the name by looking up address
01994                 break; //otherwise the loop runs too far
01995             }
01996         }
01997
01998         string message = "Are you sure you want to Delete this?";
```

```

02000
02001      caption = "About to Delete Question from " + Students_Name[student]
02002      ;
02003      MessageBoxButtons buttons = MessageBoxButtons.YesNo;
02004      DialogResult result;
02005      result = MessageBox.Show(this, message, caption, buttons,
02006      MessageBoxIcon.Question);
02007      if (result == DialogResult.Yes)
02008      {
02009          //delete the question
02010          DeleteQuestion = true;
02011          Del_ID = (byte)new_lblID;
02012          timer.Enabled = true;
02013
02014          if (lbl_arr[new_lblID].Text.Contains("***"))
02015          {
02016              //this means it is unread
02017              wasQunread[new_lblID] = true;
02018              UnreadCount--; //decrement unread count
02019              if (UnreadCount > 0)
02020              {
02021                  this.Text = " Classroom Inquisition | Home (" +
02022                  UnreadCount.ToString() + ")";
02023              }
02024              else
02025              {
02026                  this.Text = " Classroom Inquisition | Home";
02027              }
02028          }
02029          else
02030          {
02031              //question still exists
02032          }
02033      }
02034
02035      /*This method builds the context menu based on which control is
desiring a context menu*/
02036      private void cntxtMenu_arr_Popup(object sender, EventArgs e)
02037      {
02038          cntxtMenu.MenuItems.Clear(); //make clean for new context menu
02039
02040          if (cntxtMenu.SourceControl == lbl_arr[new_lblID]) //if a question
label is context clicked
02041          {
02042              cntxtMenu.MenuItems.Add(multiName); //add the toggle name
02043              cntxtMenu.MenuItems.Add("-");
02044              if (!Q_status[new_lblID])
02045              {
02046                  //it is unread show mark as read
02047                  cntxtMenu.MenuItems.Add(multiMarkAsRead);
02048              }
02049              else
02050              {
02051                  cntxtMenu.MenuItems.Add(multiMarkAsUnRead);
02052              }
02053              cntxtMenu.MenuItems.Add("-");
02054              cntxtMenu.MenuItems.Add(multiDelete); //add context menu for
delete
02055
02056          }
02057          else if (cntxtMenu.SourceControl == group_arr[new_lblID])
02058          {
02059              cntxtMenu.MenuItems.Add(multiName); //add the toggle name
02060              cntxtMenu.MenuItems.Add("-");
02061              cntxtMenu.MenuItems.Add(multiDelete); //add context menu for
delete
02062
02063          }
02064          else if (cntxtMenu.SourceControl == grpbxFeed)
02065          {
02066              if (Question_Deleted)
02067              {

```

```
02067             cntxtMenu.MenuItems.Add(muItmUndo);
02068             cntxtMenu.MenuItems.Add("-");
02069         }
02070         cntxtMenu.MenuItems.Add(muItmBrdCst);
02071         cntxtMenu.MenuItems.Add(muItmDM);
02072         cntxtMenu.MenuItems.Add("-");
02073         cntxtMenu.MenuItems.Add(muItmFAQ);
02074         cntxtMenu.MenuItems.Add(muItmPrefs);
02075         cntxtMenu.MenuItems.Add("-");
02076         cntxtMenu.MenuItems.Add(muItmQuiz);
02077         cntxtMenu.MenuItems.Add(muItmAttendance);
02078         cntxtMenu.MenuItems.Add(muItmClassVote);
02079         if (ModifierKeys == System.Windows.Forms.Keys.Shift)
02080     {
02081         cntxtMenu.MenuItems.Add("-");
02082         cntxtMenu.MenuItems.Add(muItmQuit); //hidden quit option ;
02083     }
02084 }
02085 }
02086 else if (cntxtMenu.SourceControl == btnQMDel)
02087 {
02088     cntxtMenu.MenuItems.Add(muItmQuizDelete);
02089 }
02090 }
02091 }
02092 /*This method is intended to be an extension of the mousedown for
02093 lbl_arr incase there is a mis-click*/
02094 private void group_arr_MouseDown(object sender, MouseEventArgs e)
02095 {
02096     bool rightClick = (e.Button == System.Windows.Forms.MouseButtons.
Right);
02097     bool leftClick = (e.Button == System.Windows.Forms.MouseButtons.
Left);
02098     byte student = 0;
02099
02100     for (int i = 0; i < lbl_arr.Length - 1; i++) //loop through to find
the clicked one
02101     {
02102         if (sender.Equals(lbl_arr[i]))
02103         {
02104             student = (byte) i;
02105             break;
02106         }
02107     }
02108
02109     if (rightClick)
02110     {
02111         //show context menu at mouse click location aligned right
02112         try
02113         {
02114             cntxtMenu.Show(group_arr[student], e.Location,
LeftRightAlignment.Right);
02115         }
02116         catch { }
02117     }
02118 }
02119 }
02120
02121 /*This method is detecting mouse clicks and show the context menus*/
02122 private void grpbxFeed_MouseDown(object sender, MouseEventArgs e)
02123 {
02124     bool rightClick = (e.Button == System.Windows.Forms.MouseButtons.
Right);
02125     bool leftClick = (e.Button == System.Windows.Forms.MouseButtons.
Left);
02126
02127     if (rightClick)
02128     {
02129         //show context menu at mouse click location aligned right
02130         cntxtMenu.Show(grpbxFed, e.Location, LeftRightAlignment.Right)
02131     }
02132 }
```

```

02133
02134     /*This method allows for the showing of PanelPrefs from the home
02135 context menu*/
02136     private void muItmPrefs_Click(object sender, EventArgs e)
02137     {
02138         PrefsToolStripMenuItem_Click(sender, e);
02139     }
02140
02141     /*This method allows for the showing of PanelDM from the home context
02142 menu*/
02143     private void muItmDM_Click(object sender, EventArgs e)
02144     {
02145         directMessageToolStripMenuItem_Click(sender, e);
02146     }
02147     /*This method allows for the showing of PanelFAQ from the home context
02148 menu*/
02149     private void muItmFAQ_Click(object sender, EventArgs e)
02150     {
02151         generalFAQToolStripMenuItem_Click(sender, e);
02152
02153         /*This method allows for a secret quit option available only to those
02154 who know where tis*/
02155         private void muItmQuit_Click(object sender, EventArgs e)
02156         {
02157             sudo_kill = true; //bypass are you sure...
02158             this.Close(); //will be caught and confirmed on Form_closing
02159         }
02160
02161         /*This method allows for the hiding of the window close control and,
02162 inseparably, the icon*/
02163         private void chkbxCtrlHide_CheckedChanged(object sender, EventArgs e)
02164     {
02165         this.ControlBox = !chkbxCtrlHide.Checked; //toggles control
02166         this.Refresh();
02167     }
02168
02169     /*Method to call the Undo Delete Animation/Action*/
02170     private void muItmUndo_Click(object sender, EventArgs e)
02171     {
02172         //Del_ID is still the ID of the most recently deleted Item
02173         Request_Undo = true;
02174         timer.Enabled = true;
02175         group_arr[Del_ID].Show();
02176
02177         if (wasQunread[Del_ID])
02178         {
02179             //then I need to alter the unread count since this question is
02180             returning
02181             UnreadCount++;
02182             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02183             ToString() + ")";
02184         }
02185
02186         /*Method for individual name toggling on click of context menu item*/
02187         private void muItmName_Click(object sender, EventArgs e)
02188     {
02189         byte student = (byte)new_lblID;
02190         for (i = 0; i < Students_Name.Length - 1; i++)
02191         {
02192             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02193             {
02194                 student = (byte)i;//lock in the name by looking up address
02195                 break; //otherwise the loop runs too far
02196             }
02197
02198             if (Q_NameShowing[new_lblID]) //toggle name off
02199             {
02200                 //replace method returns the new string
02201                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02202 + Students_Name[student]), "");
02203
02204             }
02205         }
02206     }
02207
02208     /*Method to call the Redo Animation/Action*/
02209     private void muItmRedo_Click(object sender, EventArgs e)
02210     {
02211         //Redo is the same as Undo
02212         Request_Undo = false;
02213         timer.Enabled = false;
02214         group_arr[Del_ID].Show();
02215
02216         if (wasQunread[Del_ID])
02217         {
02218             //then I need to alter the unread count since this question is
02219             returning
02220             UnreadCount++;
02221             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02222             ToString() + ")";
02223         }
02224
02225         /*Method for individual name toggling on click of context menu item*/
02226         private void muItmName_Click(object sender, EventArgs e)
02227     {
02228         byte student = (byte)new_lblID;
02229         for (i = 0; i < Students_Name.Length - 1; i++)
02230         {
02231             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02232             {
02233                 student = (byte)i;//lock in the name by looking up address
02234                 break; //otherwise the loop runs too far
02235             }
02236
02237             if (Q_NameShowing[new_lblID]) //toggle name off
02238             {
02239                 //replace method returns the new string
02240                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02241 + Students_Name[student]), "");
02242
02243             }
02244         }
02245     }
02246
02247     /*Method to call the Cut Animation/Action*/
02248     private void muItmCut_Click(object sender, EventArgs e)
02249     {
02250         //Cut is the same as Undo
02251         Request_Undo = true;
02252         timer.Enabled = true;
02253         group_arr[Del_ID].Show();
02254
02255         if (wasQunread[Del_ID])
02256         {
02257             //then I need to alter the unread count since this question is
02258             returning
02259             UnreadCount++;
02260             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02261             ToString() + ")";
02262         }
02263
02264         /*Method for individual name toggling on click of context menu item*/
02265         private void muItmName_Click(object sender, EventArgs e)
02266     {
02267         byte student = (byte)new_lblID;
02268         for (i = 0; i < Students_Name.Length - 1; i++)
02269         {
02270             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02271             {
02272                 student = (byte)i;//lock in the name by looking up address
02273                 break; //otherwise the loop runs too far
02274             }
02275
02276             if (Q_NameShowing[new_lblID]) //toggle name off
02277             {
02278                 //replace method returns the new string
02279                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02280 + Students_Name[student]), "");
02281
02282             }
02283         }
02284     }
02285
02286     /*Method to call the Copy Animation/Action*/
02287     private void muItmCopy_Click(object sender, EventArgs e)
02288     {
02289         //Copy is the same as Undo
02290         Request_Undo = true;
02291         timer.Enabled = true;
02292         group_arr[Del_ID].Show();
02293
02294         if (wasQunread[Del_ID])
02295         {
02296             //then I need to alter the unread count since this question is
02297             returning
02298             UnreadCount++;
02299             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02300             ToString() + ")";
02301         }
02302
02303         /*Method for individual name toggling on click of context menu item*/
02304         private void muItmName_Click(object sender, EventArgs e)
02305     {
02306         byte student = (byte)new_lblID;
02307         for (i = 0; i < Students_Name.Length - 1; i++)
02308         {
02309             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02310             {
02311                 student = (byte)i;//lock in the name by looking up address
02312                 break; //otherwise the loop runs too far
02313             }
02314
02315             if (Q_NameShowing[new_lblID]) //toggle name off
02316             {
02317                 //replace method returns the new string
02318                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02319 + Students_Name[student]), "");
02320
02321             }
02322         }
02323     }
02324
02325     /*Method to call the Paste Animation/Action*/
02326     private void muItmPaste_Click(object sender, EventArgs e)
02327     {
02328         //Paste is the same as Undo
02329         Request_Undo = true;
02330         timer.Enabled = true;
02331         group_arr[Del_ID].Show();
02332
02333         if (wasQunread[Del_ID])
02334         {
02335             //then I need to alter the unread count since this question is
02336             returning
02337             UnreadCount++;
02338             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02339             ToString() + ")";
02340         }
02341
02342         /*Method for individual name toggling on click of context menu item*/
02343         private void muItmName_Click(object sender, EventArgs e)
02344     {
02345         byte student = (byte)new_lblID;
02346         for (i = 0; i < Students_Name.Length - 1; i++)
02347         {
02348             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02349             {
02350                 student = (byte)i;//lock in the name by looking up address
02351                 break; //otherwise the loop runs too far
02352             }
02353
02354             if (Q_NameShowing[new_lblID]) //toggle name off
02355             {
02356                 //replace method returns the new string
02357                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02358 + Students_Name[student]), "");
02359
02360             }
02361         }
02362     }
02363
02364     /*Method to call the Select All Animation/Action*/
02365     private void muItmSelectAll_Click(object sender, EventArgs e)
02366     {
02367         //Select All is the same as Undo
02368         Request_Undo = true;
02369         timer.Enabled = true;
02370         group_arr[Del_ID].Show();
02371
02372         if (wasQunread[Del_ID])
02373         {
02374             //then I need to alter the unread count since this question is
02375             returning
02376             UnreadCount++;
02377             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02378             ToString() + ")";
02379         }
02380
02381         /*Method for individual name toggling on click of context menu item*/
02382         private void muItmName_Click(object sender, EventArgs e)
02383     {
02384         byte student = (byte)new_lblID;
02385         for (i = 0; i < Students_Name.Length - 1; i++)
02386         {
02387             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02388             {
02389                 student = (byte)i;//lock in the name by looking up address
02390                 break; //otherwise the loop runs too far
02391             }
02392
02393             if (Q_NameShowing[new_lblID]) //toggle name off
02394             {
02395                 //replace method returns the new string
02396                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02397 + Students_Name[student]), "");
02398
02399             }
02400         }
02401     }
02402
02403     /*Method to call the Find Animation/Action*/
02404     private void muItmFind_Click(object sender, EventArgs e)
02405     {
02406         //Find is the same as Undo
02407         Request_Undo = true;
02408         timer.Enabled = true;
02409         group_arr[Del_ID].Show();
02410
02411         if (wasQunread[Del_ID])
02412         {
02413             //then I need to alter the unread count since this question is
02414             returning
02415             UnreadCount++;
02416             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02417             ToString() + ")";
02418         }
02419
02420         /*Method for individual name toggling on click of context menu item*/
02421         private void muItmName_Click(object sender, EventArgs e)
02422     {
02423         byte student = (byte)new_lblID;
02424         for (i = 0; i < Students_Name.Length - 1; i++)
02425         {
02426             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02427             {
02428                 student = (byte)i;//lock in the name by looking up address
02429                 break; //otherwise the loop runs too far
02430             }
02431
02432             if (Q_NameShowing[new_lblID]) //toggle name off
02433             {
02434                 //replace method returns the new string
02435                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02436 + Students_Name[student]), "");
02437
02438             }
02439         }
02440     }
02441
02442     /*Method to call the Replace Animation/Action*/
02443     private void muItmReplace_Click(object sender, EventArgs e)
02444     {
02445         //Replace is the same as Undo
02446         Request_Undo = true;
02447         timer.Enabled = true;
02448         group_arr[Del_ID].Show();
02449
02450         if (wasQunread[Del_ID])
02451         {
02452             //then I need to alter the unread count since this question is
02453             returning
02454             UnreadCount++;
02455             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02456             ToString() + ")";
02457         }
02458
02459         /*Method for individual name toggling on click of context menu item*/
02460         private void muItmName_Click(object sender, EventArgs e)
02461     {
02462         byte student = (byte)new_lblID;
02463         for (i = 0; i < Students_Name.Length - 1; i++)
02464         {
02465             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02466             {
02467                 student = (byte)i;//lock in the name by looking up address
02468                 break; //otherwise the loop runs too far
02469             }
02470
02471             if (Q_NameShowing[new_lblID]) //toggle name off
02472             {
02473                 //replace method returns the new string
02474                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02475 + Students_Name[student]), "");
02476
02477             }
02478         }
02479     }
02480
02481     /*Method to call the Cut Animation/Action*/
02482     private void muItmCut_Click(object sender, EventArgs e)
02483     {
02484         //Cut is the same as Undo
02485         Request_Undo = true;
02486         timer.Enabled = true;
02487         group_arr[Del_ID].Show();
02488
02489         if (wasQunread[Del_ID])
02490         {
02491             //then I need to alter the unread count since this question is
02492             returning
02493             UnreadCount++;
02494             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02495             ToString() + ")";
02496         }
02497
02498         /*Method for individual name toggling on click of context menu item*/
02499         private void muItmName_Click(object sender, EventArgs e)
02500     {
02501         byte student = (byte)new_lblID;
02502         for (i = 0; i < Students_Name.Length - 1; i++)
02503         {
02504             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02505             {
02506                 student = (byte)i;//lock in the name by looking up address
02507                 break; //otherwise the loop runs too far
02508             }
02509
02510             if (Q_NameShowing[new_lblID]) //toggle name off
02511             {
02512                 //replace method returns the new string
02513                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02514 + Students_Name[student]), "");
02515
02516             }
02517         }
02518     }
02519
02520     /*Method to call the Copy Animation/Action*/
02521     private void muItmCopy_Click(object sender, EventArgs e)
02522     {
02523         //Copy is the same as Undo
02524         Request_Undo = true;
02525         timer.Enabled = true;
02526         group_arr[Del_ID].Show();
02527
02528         if (wasQunread[Del_ID])
02529         {
02530             //then I need to alter the unread count since this question is
02531             returning
02532             UnreadCount++;
02533             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02534             ToString() + ")";
02535         }
02536
02537         /*Method for individual name toggling on click of context menu item*/
02538         private void muItmName_Click(object sender, EventArgs e)
02539     {
02540         byte student = (byte)new_lblID;
02541         for (i = 0; i < Students_Name.Length - 1; i++)
02542         {
02543             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02544             {
02545                 student = (byte)i;//lock in the name by looking up address
02546                 break; //otherwise the loop runs too far
02547             }
02548
02549             if (Q_NameShowing[new_lblID]) //toggle name off
02550             {
02551                 //replace method returns the new string
02552                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02553 + Students_Name[student]), "");
02554
02555             }
02556         }
02557     }
02558
02559     /*Method to call the Paste Animation/Action*/
02560     private void muItmPaste_Click(object sender, EventArgs e)
02561     {
02562         //Paste is the same as Undo
02563         Request_Undo = true;
02564         timer.Enabled = true;
02565         group_arr[Del_ID].Show();
02566
02567         if (wasQunread[Del_ID])
02568         {
02569             //then I need to alter the unread count since this question is
02570             returning
02571             UnreadCount++;
02572             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02573             ToString() + ")";
02574         }
02575
02576         /*Method for individual name toggling on click of context menu item*/
02577         private void muItmName_Click(object sender, EventArgs e)
02578     {
02579         byte student = (byte)new_lblID;
02580         for (i = 0; i < Students_Name.Length - 1; i++)
02581         {
02582             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02583             {
02584                 student = (byte)i;//lock in the name by looking up address
02585                 break; //otherwise the loop runs too far
02586             }
02587
02588             if (Q_NameShowing[new_lblID]) //toggle name off
02589             {
02590                 //replace method returns the new string
02591                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02592 + Students_Name[student]), "");
02593
02594             }
02595         }
02596     }
02597
02598     /*Method to call the Select All Animation/Action*/
02599     private void muItmSelectAll_Click(object sender, EventArgs e)
02600     {
02601         //Select All is the same as Undo
02602         Request_Undo = true;
02603         timer.Enabled = true;
02604         group_arr[Del_ID].Show();
02605
02606         if (wasQunread[Del_ID])
02607         {
02608             //then I need to alter the unread count since this question is
02609             returning
02610             UnreadCount++;
02611             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02612             ToString() + ")";
02613         }
02614
02615         /*Method for individual name toggling on click of context menu item*/
02616         private void muItmName_Click(object sender, EventArgs e)
02617     {
02618         byte student = (byte)new_lblID;
02619         for (i = 0; i < Students_Name.Length - 1; i++)
02620         {
02621             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02622             {
02623                 student = (byte)i;//lock in the name by looking up address
02624                 break; //otherwise the loop runs too far
02625             }
02626
02627             if (Q_NameShowing[new_lblID]) //toggle name off
02628             {
02629                 //replace method returns the new string
02630                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02631 + Students_Name[student]), "");
02632
02633             }
02634         }
02635     }
02636
02637     /*Method to call the Find Animation/Action*/
02638     private void muItmFind_Click(object sender, EventArgs e)
02639     {
02640         //Find is the same as Undo
02641         Request_Undo = true;
02642         timer.Enabled = true;
02643         group_arr[Del_ID].Show();
02644
02645         if (wasQunread[Del_ID])
02646         {
02647             //then I need to alter the unread count since this question is
02648             returning
02649             UnreadCount++;
02650             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02651             ToString() + ")";
02652         }
02653
02654         /*Method for individual name toggling on click of context menu item*/
02655         private void muItmName_Click(object sender, EventArgs e)
02656     {
02657         byte student = (byte)new_lblID;
02658         for (i = 0; i < Students_Name.Length - 1; i++)
02659         {
02660             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02661             {
02662                 student = (byte)i;//lock in the name by looking up address
02663                 break; //otherwise the loop runs too far
02664             }
02665
02666             if (Q_NameShowing[new_lblID]) //toggle name off
02667             {
02668                 //replace method returns the new string
02669                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02670 + Students_Name[student]), "");
02671
02672             }
02673         }
02674     }
02675
02676     /*Method to call the Replace Animation/Action*/
02677     private void muItmReplace_Click(object sender, EventArgs e)
02678     {
02679         //Replace is the same as Undo
02680         Request_Undo = true;
02681         timer.Enabled = true;
02682         group_arr[Del_ID].Show();
02683
02684         if (wasQunread[Del_ID])
02685         {
02686             //then I need to alter the unread count since this question is
02687             returning
02688             UnreadCount++;
02689             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02690             ToString() + ")";
02691         }
02692
02693         /*Method for individual name toggling on click of context menu item*/
02694         private void muItmName_Click(object sender, EventArgs e)
02695     {
02696         byte student = (byte)new_lblID;
02697         for (i = 0; i < Students_Name.Length - 1; i++)
02698         {
02699             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02700             {
02701                 student = (byte)i;//lock in the name by looking up address
02702                 break; //otherwise the loop runs too far
02703             }
02704
02705             if (Q_NameShowing[new_lblID]) //toggle name off
02706             {
02707                 //replace method returns the new string
02708                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02709 + Students_Name[student]), "");
02710
02711             }
02712         }
02713     }
02714
02715     /*Method to call the Cut Animation/Action*/
02716     private void muItmCut_Click(object sender, EventArgs e)
02717     {
02718         //Cut is the same as Undo
02719         Request_Undo = true;
02720         timer.Enabled = true;
02721         group_arr[Del_ID].Show();
02722
02723         if (wasQunread[Del_ID])
02724         {
02725             //then I need to alter the unread count since this question is
02726             returning
02727             UnreadCount++;
02728             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02729             ToString() + ")";
02730         }
02731
02732         /*Method for individual name toggling on click of context menu item*/
02733         private void muItmName_Click(object sender, EventArgs e)
02734     {
02735         byte student = (byte)new_lblID;
02736         for (i = 0; i < Students_Name.Length - 1; i++)
02737         {
02738             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02739             {
02740                 student = (byte)i;//lock in the name by looking up address
02741                 break; //otherwise the loop runs too far
02742             }
02743
02744             if (Q_NameShowing[new_lblID]) //toggle name off
02745             {
02746                 //replace method returns the new string
02747                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02748 + Students_Name[student]), "");
02749
02750             }
02751         }
02752     }
02753
02754     /*Method to call the Copy Animation/Action*/
02755     private void muItmCopy_Click(object sender, EventArgs e)
02756     {
02757         //Copy is the same as Undo
02758         Request_Undo = true;
02759         timer.Enabled = true;
02760         group_arr[Del_ID].Show();
02761
02762         if (wasQunread[Del_ID])
02763         {
02764             //then I need to alter the unread count since this question is
02765             returning
02766             UnreadCount++;
02767             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02768             ToString() + ")";
02769         }
02770
02771         /*Method for individual name toggling on click of context menu item*/
02772         private void muItmName_Click(object sender, EventArgs e)
02773     {
02774         byte student = (byte)new_lblID;
02775         for (i = 0; i < Students_Name.Length - 1; i++)
02776         {
02777             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02778             {
02779                 student = (byte)i;//lock in the name by looking up address
02780                 break; //otherwise the loop runs too far
02781             }
02782
02783             if (Q_NameShowing[new_lblID]) //toggle name off
02784             {
02785                 //replace method returns the new string
02786                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02787 + Students_Name[student]), "");
02788
02789             }
02790         }
02791     }
02792
02793     /*Method to call the Paste Animation/Action*/
02794     private void muItmPaste_Click(object sender, EventArgs e)
02795     {
02796         //Paste is the same as Undo
02797         Request_Undo = true;
02798         timer.Enabled = true;
02799         group_arr[Del_ID].Show();
02800
02801         if (wasQunread[Del_ID])
02802         {
02803             //then I need to alter the unread count since this question is
02804             returning
02805             UnreadCount++;
02806             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02807             ToString() + ")";
02808         }
02809
02810         /*Method for individual name toggling on click of context menu item*/
02811         private void muItmName_Click(object sender, EventArgs e)
02812     {
02813         byte student = (byte)new_lblID;
02814         for (i = 0; i < Students_Name.Length - 1; i++)
02815         {
02816             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02817             {
02818                 student = (byte)i;//lock in the name by looking up address
02819                 break; //otherwise the loop runs too far
02820             }
02821
02822             if (Q_NameShowing[new_lblID]) //toggle name off
02823             {
02824                 //replace method returns the new string
02825                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02826 + Students_Name[student]), "");
02827
02828             }
02829         }
02830     }
02831
02832     /*Method to call the Select All Animation/Action*/
02833     private void muItmSelectAll_Click(object sender, EventArgs e)
02834     {
02835         //Select All is the same as Undo
02836         Request_Undo = true;
02837         timer.Enabled = true;
02838         group_arr[Del_ID].Show();
02839
02840         if (wasQunread[Del_ID])
02841         {
02842             //then I need to alter the unread count since this question is
02843             returning
02844             UnreadCount++;
02845             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02846             ToString() + ")";
02847         }
02848
02849         /*Method for individual name toggling on click of context menu item*/
02850         private void muItmName_Click(object sender, EventArgs e)
02851     {
02852         byte student = (byte)new_lblID;
02853         for (i = 0; i < Students_Name.Length - 1; i++)
02854         {
02855             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02856             {
02857                 student = (byte)i;//lock in the name by looking up address
02858                 break; //otherwise the loop runs too far
02859             }
02860
02861             if (Q_NameShowing[new_lblID]) //toggle name off
02862             {
02863                 //replace method returns the new string
02864                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02865 + Students_Name[student]), "");
02866
02867             }
02868         }
02869     }
02870
02871     /*Method to call the Find Animation/Action*/
02872     private void muItmFind_Click(object sender, EventArgs e)
02873     {
02874         //Find is the same as Undo
02875         Request_Undo = true;
02876         timer.Enabled = true;
02877         group_arr[Del_ID].Show();
02878
02879         if (wasQunread[Del_ID])
02880         {
02881             //then I need to alter the unread count since this question is
02882             returning
02883             UnreadCount++;
02884             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02885             ToString() + ")";
02886         }
02887
02888         /*Method for individual name toggling on click of context menu item*/
02889         private void muItmName_Click(object sender, EventArgs e)
02890     {
02891         byte student = (byte)new_lblID;
02892         for (i = 0; i < Students_Name.Length - 1; i++)
02893         {
02894             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02895             {
02896                 student = (byte)i;//lock in the name by looking up address
02897                 break; //otherwise the loop runs too far
02898             }
02899
02900             if (Q_NameShowing[new_lblID]) //toggle name off
02901             {
02902                 //replace method returns the new string
02903                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02904 + Students_Name[student]), "");
02905
02906             }
02907         }
02908     }
02909
02910     /*Method to call the Replace Animation/Action*/
02911     private void muItmReplace_Click(object sender, EventArgs e)
02912     {
02913         //Replace is the same as Undo
02914         Request_Undo = true;
02915         timer.Enabled = true;
02916         group_arr[Del_ID].Show();
02917
02918         if (wasQunread[Del_ID])
02919         {
02920             //then I need to alter the unread count since this question is
02921             returning
02922             UnreadCount++;
02923             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02924             ToString() + ")";
02925         }
02926
02927         /*Method for individual name toggling on click of context menu item*/
02928         private void muItmName_Click(object sender, EventArgs e)
02929     {
02930         byte student = (byte)new_lblID;
02931         for (i = 0; i < Students_Name.Length - 1; i++)
02932         {
02933             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02934             {
02935                 student = (byte)i;//lock in the name by looking up address
02936                 break; //otherwise the loop runs too far
02937             }
02938
02939             if (Q_NameShowing[new_lblID]) //toggle name off
02940             {
02941                 //replace method returns the new string
02942                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02943 + Students_Name[student]), "");
02944
02945             }
02946         }
02947     }
02948
02949     /*Method to call the Cut Animation/Action*/
02950     private void muItmCut_Click(object sender, EventArgs e)
02951     {
02952         //Cut is the same as Undo
02953         Request_Undo = true;
02954         timer.Enabled = true;
02955         group_arr[Del_ID].Show();
02956
02957         if (wasQunread[Del_ID])
02958         {
02959             //then I need to alter the unread count since this question is
02960             returning
02961             UnreadCount++;
02962             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02963             ToString() + ")";
02964         }
02965
02966         /*Method for individual name toggling on click of context menu item*/
02967         private void muItmName_Click(object sender, EventArgs e)
02968     {
02969         byte student = (byte)new_lblID;
02970         for (i = 0; i < Students_Name.Length - 1; i++)
02971         {
02972             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
02973             {
02974                 student = (byte)i;//lock in the name by looking up address
02975                 break; //otherwise the loop runs too far
02976             }
02977
02978             if (Q_NameShowing[new_lblID]) //toggle name off
02979             {
02980                 //replace method returns the new string
02981                 lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.Replace((" -"
02982 + Students_Name[student]), "");
02983
02984             }
02985         }
02986     }
02987
02988     /*Method to call the Copy Animation/Action*/
02989     private void muItmCopy_Click(object sender, EventArgs e)
02990     {
02991         //Copy is the same as Undo
02992         Request_Undo = true;
02993         timer.Enabled = true;
02994         group_arr[Del_ID].Show();
02995
02996         if (wasQunread[Del_ID])
02997         {
02998             //then I need to alter the unread count since this question is
02999             returning
03000             UnreadCount++;
03001             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
03002             ToString() + ")";
03003         }
03004
03005         /*Method for individual name toggling on click of context menu item*/
03006         private void muItmName_Click(object sender, EventArgs e)
03007     {
03008         byte student = (byte)new_lblID;
03009         for (i = 0; i < Students_Name.Length - 1; i++)
03010         {
03011             if (String.CompareOrdinal(addr_tbl[i], Q_sender[student]) == 0)
03012             {
03013                 student = (byte)i;//lock in the name by looking up address
```

```

02199         Q_NameShowing[new_lblID] = false;
02200     }
02201     else
02202     {
02203         lbl_arr[new_lblID].Text += " -" + Students_Name[student]; //
02204         append name
02205     }
02206
02207 }
02208
02209 /*Method for the global toggling of name by view>ShowNames menu item*/
02210 private void showNamesToolStripMenuItem_Click(object sender, EventArgs
e)
02211 {
02212     //toggle all name values to true or false and change those who need
02213     changed
02214     byte student = 0;
02215     showNamesToolStripMenuItem.Checked = !showNamesToolStripMenuItem.
Checked; //toggle the value
02216     if (showNamesToolStripMenuItem.Checked)
02217     {
02218         for (int i = 0; i < NumQuestions; i++) //go through all the
questions
02219         {
02220             if (Q_NameShowing[i] == false) //toggle the false ones
02221             {
02222                 for (j = 0; j < Students_Name.Length - 1; j++)
02223                 {
02224                     if (String.CompareOrdinal(addr_tbl[j], Q_sender[i])
== 0)
02225                     {
02226                         up address
02227                         student = (byte)j;//lock in the name by looking
02228                         break; //otherwise the loop runs too far
02229                     }
02230                     lbl_arr[i].Text += " -" + Students_Name[student]; //
02231                     append name
02232                     Q_NameShowing[i] = true;
02233                 }
02234             }
02235             else
02236             {
02237                 for (int i = 0; i < NumQuestions; i++) //go through all the
questions
02238                 {
02239                     if (Q_NameShowing[i] == true) //toggle the true ones
02240                     {
02241                         for (j = 0; j < Students_Name.Length - 1; j++)
02242                         {
02243                             if (String.CompareOrdinal(addr_tbl[j], Q_sender[i])
== 0)
02244                             {
02245                                 up address
02246                                 student = (byte)j;//lock in the name by looking
02247                                 break; //otherwise the loop runs too far
02248                             }
02249                             lbl_arr[i].Text = lbl_arr[i].Text.Replace((" -" +
Students_Name[student]), "");
02250                             Q_NameShowing[i] = false;
02251                         }
02252                     }
02253                 }
02254             }
02255
02256 /*This method opens the quiz maker (via animation)*/
02257 private void btnQuizMaker_Click(object sender, EventArgs e)
02258 {
02259     //this should bring in QuizMaker
02260     string[] tmpstring = new string[logHistoryLength];

```

```

02261     try
02262     {
02263         if (File.Exists(@".data\Quiz\QuizMaker.txt"))
02264     {
02265         i = 0;
02266         using (StreamReader sr = File.OpenText(@"
02267             .data\Quiz\QuizMaker.txt"))
02268         {
02269             string tempstr = "";
02270             while ((tempstr = sr.ReadLine()) != null)
02271             {
02272                 tmpstring = tempstr.Split('\x0A'); //divvy it up by
02273                 newline
02274                 logTmpString[i] = tmpstring[0]; //weird error where
02275                 tmpstring is only using index 0...this fixes it
02276                 i++;
02277             }
02278             sr.Close();
02279             lstbxQuizMaker.Items.Clear();
02280             for (j = 0; j <= (i - 1); j++)
02281             {
02282                 lstbxQuizMaker.Items.Add(logTmpString[j]); //
02283             }
02284             i = j = 0;
02285         }
02286         //file was deleted for some reason...
02287         using (FileStream fs = File.Create(@"
02288             .data\Quiz\QuizMaker.txt")) //make a file please
02289         {
02290             fs.Close();
02291         }
02292         lstbxQuizMaker.Items.Clear();
02293         //maybe tell the teacher the quiz file was gone?
02294     }
02295 }
02296 catch (Exception E)
02297 {
02298     MessageBox.Show("This is embarrassing..there were Errors..." +
Environment.NewLine + E);
02299 }
02300
02301     if (BrdcstShowing)
02302     {
02303         DesireQzMkr = true;
02304         broadcastToolStripMenuItem_Click(sender, e);
02305     }
02306     else
02307     {
02308         QuizMakerClicked = true;
02309         timer.Enabled = true;
02310     }
02311 }
02312 }
02313 /*This method closes the quiz maker (via animation)*/
02314 private void btnQMcls_Click(object sender, EventArgs e)
02315 {
02316     //hide quiz maker
02317     try
02318     {
02319         using (StreamWriter sw = new StreamWriter(@"
02320             .data\Quiz\QuizMaker.txt"))
02321         {
02322             //build and write the strings for the data file
02323             for (int i = 0; i < lstbxQuizMaker.Items.Count; i++)
02324             {
02325                 sw.WriteLine(lstbxQuizMaker.Items[i]);
02326             }
02327             sw.Flush();

```

```
02328             sw.Close();
02329         }
02330     }
02331     catch
02332     {
02333         MessageBox.Show("He's dead Jim! You need permissions to the
02334         Program Files Directory.");
02335     }
02336     QuizMakerClicked = true;
02337     timer.Enabled = true;
02338 }
02339
02340 /*This Allows for the editing of existing questions*/
02341 private void btnQMEdit_Click(object sender, EventArgs e)
02342 {
02343     //edit the currently selected question in Listbox, messagebox to
02344     select one if they haven't yet
02345     if (!EDTClicked)
02346     {
02347         if (lstbxQuizMaker.SelectedIndex == -1)
02348         {
02349             MessageBox.Show("You haven't selected a Question to edit
02350             yet.");
02351         }
02352         txtbxQM.Text = lstbxQuizMaker.Items[lstbxQuizMaker.
02353         SelectedIndex].ToString();
02354         txtbxQM.TextAlign = HorizontalAlignment.Left;
02355         txtbxQM.Focus();
02356         btnQMEdit.Text = "Done"; //multipurpose the button
02357         EDTClicked = true;
02358     }
02359     else
02360     {
02361         //second time clicked, they're done..
02362         if ((txtbxQM.Text.Trim() != "") && (txtbxQM.Text != "^^^^
02363         Select a question to edit. ^^^^"))
02364         {
02365             lstbxQuizMaker.Items.Insert(lstbxQuizMaker.SelectedIndex,
02366             txtbxQM.Text);
02367             lstbxQuizMaker.Items.RemoveAt(lstbxQuizMaker.SelectedIndex)
02368         ;
02369         }
02370         lstbxQuizMaker.Refresh();
02371     }
02372
02373     txtbxQM.Text = "^^^^ Select a question to edit. ^^^^";
02374     txtbxQM.TextAlign = HorizontalAlignment.Center;
02375     btnQMEdit.Text = "Edit";
02376     EDTClicked = false;
02377 }
02378 }
02379
02380 /*Allows for the deleting of a question*/
02381 private void btnQMDel_MouseDown(object sender, MouseEventArgs e)
02382 {
02383     bool rightClick = (e.Button == System.Windows.Forms.MouseButtons.
02384     Right);
02385     bool leftClick = (e.Button == System.Windows.Forms.MouseButtons.
02386     Left);
02387
02388     if (rightClick)
02389     {
02390         //show context menu for undo delete, if something has been
02391         deleted
02392         if (QuizQuestionDel)
02393             ctxtMenu.Show(btnQMDel, e.Location, LeftRightAlignment.
```

```

        Right);
02391     }
02392     else
02393     {
02394         if (lstbxQuizMaker.SelectedIndex != -1) //something is actually
selected
02395         {
02396             QuizDelID = lstbxQuizMaker.SelectedIndex; //to save the ID
but also to simplify the code
02397             QzQuestionDel = lstbxQuizMaker.Items[QuizDelID].ToString();
// save question
02398             lstbxQuizMaker.Items.RemoveAt(QuizDelID);
02399             if (EDTClicked) //we're in edit mode
{
                //resets textbox in case you were editing the question
you deleted
02400                 txtbxQM.Text = "^^^^ Select a question to edit. ^^^^";
02401                 txtbxQM.TextAlign = HorizontalAlignment.Center;
02402                 btnQMEedit.Text = "Edit";
02403                 EDTClicked = false; //reset flag
02404             }
02405             QuizQuestionDel = true;
02406             trayICON.BalloonTipTitle = "Undo Delete by";
02407             trayICON.BalloonTipText = "Right click Delete Button";
02408             trayICON.ShowBalloonTip(1200);
02409         }
02410     }
02411     else
02412     {
02413         MessageBox.Show("Select a question to Delete!");
02414     }
02415 }
02416 }
02417 }
02418 /*Clears the rename textbox*/
02419 private void btnQMclr_Click(object sender, EventArgs e)
02420 {
02421     txtbxQM.Clear();
02422     txtbxQM.TextAlign = HorizontalAlignment.Left;
02423 }
02424
02425 /*Allows you to add a new question*/
02426 private void btnAddQM_Click(object sender, EventArgs e)
02427 {
02428     if (!EDTClicked)
02429     {
02430         if ((txtbxQM.Text.Trim() != "") && (txtbxQM.Text != "^^^^
Select a question to edit. ^^^^"))
02431         {
02432             lstbxQuizMaker.Items.Add(txtbxQM.Text);
02433             lstbxQuizMaker.Refresh();
02434             txtbxQM.Text = "^^^^ Select a question to edit. ^^^^";
02435             txtbxQM.TextAlign = HorizontalAlignment.Center;
02436         }
02437     }
02438     else
02439     {
02440         MessageBox.Show("You must type a question!");
02441     }
02442 }
02443 else
02444 {
02445     //add was accidentally clicked while in edit mode, perform edit
action instead
02446     btnQMEedit_Click(sender, e);
02447 }
02448 }
02449 }
02450
02451 /*This method clears the textbox if it's clicked with nothing or
default inside*/
02452 private void txtbxQM_Click(object sender, EventArgs e)
02453 {
02454     if ((txtbxQM.Text.Trim() != "") && (txtbxQM.Text != "^^^^ Select a
question to edit. ^^^^"))
02455     {

```

```
02456         //there's something there
02457     }
02458     else
02459     {
02460         txtbxQM.Clear();
02461         txtbxQM.TextAlign = HorizontalAlignment.Left;
02462     }
02463 }
02464
02465 /*Method to create questions on the quiz*/
02466 public bool MakeQuizQuestion(string question, int number)
02467 {
02468     bool success = false;
02469     int index = number;
02470     try
02471     {
02472         //
02473         // lblQuestionNum_*
02474         //
02475         lblQuestionNum_arr[index] = new Label();
02476         lblQuestionNum_arr[index].AutoSize = true;
02477         lblQuestionNum_arr[index].Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
02478         lblQuestionNum_arr[index].Location = lblQuestionNum_arr_tmp;
02479         lblQuestionNum_arr[index].Name = "lblQuestionNum_arr_" + number
02480         .ToString();
02481         lblQuestionNum_arr[index].Size = new System.Drawing.Size(23, 20
02482 );
02483         lblQuestionNum_arr[index].TabIndex = 0;
02484         lblQuestionNum_arr[index].Text = (number + 1).ToString() + ")";
02485         //
02486         // txtbxQuestion_*
02487         //
02488         txtbxQuestion_arr[index] = new TextBox();
02489         txtbxQuestion_arr[index].BackColor = System.Drawing.
02490         SystemColors.ControlDarkDark;
02491         txtbxQuestion_arr[index].BorderStyle = System.Windows.Forms.
02492         BorderStyle.FixedSingle;
02493         txtbxQuestion_arr[index].ForeColor = ForeColorTheme;
02494         txtbxQuestion_arr[index].Location = txtbxQuestion_arr_tmp;
02495         txtbxQuestion_arr[index].Multiline = true;
02496         txtbxQuestion_arr[index].Name = "txtbxQuestion_arr_" + number.
02497         ToString();
02498         txtbxQuestion_arr[index].ReadOnly = true;
02499         txtbxQuestion_arr[index].Size = new System.Drawing.Size(252, 37
02500 );
02501         txtbxQuestion_arr[index].TabIndex = 1;
02502         txtbxQuestion_arr[index].Text = question;
02503         txtbxQuestion_arr[index]. TextAlign = System.Windows.Forms.
02504         HorizontalAlignment.Center;
02505         //
02506         // btnQuizQuestionSend_*
02507         //
02508         btnQuizQuestionSend_arr[index] = new Button();
02509         btnQuizQuestionSend_arr[index].Location =
02510         btnQuizQuestionSend_arr_tmp;
02511         btnQuizQuestionSend_arr[index].Name = "btnQuizQuestionSend_arr_"
02512         " + number.ToString();
02513         btnQuizQuestionSend_arr[index].Size = new System.Drawing.Size(3
02514         6, 37);
02515         btnQuizQuestionSend_arr[index].TabIndex = 2;
02516         btnQuizQuestionSend_arr[index].Text = "Ask";
02517         btnQuizQuestionSend_arr[index].UseVisualStyleBackColor = true;
02518         btnQuizQuestionSend_arr[index].MouseClick += new
02519         MouseEventHandler(btnQuizAsk_Click);
02520         //
02521         // PanelQuestion_arr_*
02522         //
02523         PanelQuestion_arr[index] = new Panel();
02524         PanelQuestion_arr[index].BackColor = BackColorTheme;
02525         PanelQuestion_arr[index].Controls.Add(btnQuizQuestionSend_arr[
02526         number]);
02527         PanelQuestion_arr[index].Controls.Add(txtbxQuestion_arr[number]
```

```

);
02516     PanelQuestion_arr[index].Controls.Add(lblQuestionNum_arr[number
]);
02517     PanelQuestion_arr[index].ForeColor = ForeColorTheme;
02518     PanelQuestion_arr[index].Location = PanelQuestion_arr_tmp;
02519     PanelQuestion_arr[index].Name = "PanelQuestion_arr_" + number.
02520     ToString();
02521     PanelQuestion_arr[index].Size = new System.Drawing.Size(330, 63
);
02522     PanelQuestion_arr[index].TabIndex = 0;
02523     pnlQuiz.Controls.Add(PanelQuestion_arr[index]); //add new
question to container
02524     PanelQuestion_arr_tmp.Y += 63; //move down for next question
02525     success = true;
02526 }
02527 catch
02528 {
02529     success = false;
02530 }
02531
02532 return success;
02533 }
02534
02535 /*Closes Quiz mode*/
02536 private void btnExitQuiz_Click(object sender, EventArgs e)
{
02537     //exit quiz mode
02538     QuizModeClicked = true;
02539     timer.Enabled = true;
02540     quitToolStripMenuItem.Checked = false; //it won't uncheck itself
02541     StateMachine = "Normal";
02542     normalToolStripMenuItem.Checked = true;
02543 }
02544
02545 /*Opens quiz mode*/
02546 private void btnQuizMode_Click(object sender, EventArgs e)
{
02547     pnlQuiz.Controls.Clear(); //get rid of old questions
02548     PanelQuestion_arr_tmp.X = 7;//default coordinates
02549     PanelQuestion_arr_tmp.Y = 6;// ^
02550
02551     for (int i = 0; i < lstbxQuizMaker.Items.Count; i++)
02552     {
02553         MakeQuizQuestion(lstbxQuizMaker.Items[i].ToString(), i); //make
02554         new questions
02555     }
02556
02557     //enter into quiz mode
02558     quizToolStripMenuItem_Click(sender, e);
02559 }
02560
02561 /*Allows for Undoing the delete of a question*/
02562 private void muItmQuizDelete_Click(object sender, EventArgs e)
{
02563     //Undo the delete of a question from quiz
02564     lstbxQuizMaker.Items.Insert(QuizDelID, OzQuestionDel);
02565     lstbxQuizMaker.SelectedIndex = QuizDelID;
02566
02567     QuizQuestionDel = false;
02568 }
02569
02570 /*Method for the sending of questions via broadcast*/
02571 private void btnQuizAsk_Click(object sender, EventArgs e)
{
02572     byte question = 0;
02573
02574     for (int i = 0; i < btnQuizQuestionSend_arr.Length - 1; i++) //loop through to find the clicked one
02575     {
02576         if (sender.Equals(btnQuizQuestionSend_arr[i]))
02577         {
02578             question = (byte) i;
02579
02580
02581
02582
02583
02584
02585
02586
02587
02588
02589
02590
02591
02592
02593
02594
02595
02596
02597
02598
02599
02600
02601
02602
02603
02604
02605
02606
02607
02608
02609
02610
02611
02612
02613
02614
02615
02616
02617
02618
02619
02620
02621
02622
02623
02624
02625
02626
02627
02628
02629
02630
02631
02632
02633
02634
02635
02636
02637
02638
02639
02640
02641
02642
02643
02644
02645
02646
02647
02648
02649
02650
02651
02652
02653
02654
02655
02656
02657
02658
02659
02660
02661
02662
02663
02664
02665
02666
02667
02668
02669
02670
02671
02672
02673
02674
02675
02676
02677
02678
02679
02680
02681
02682
02683
02684
02685
02686
02687
02688
02689
02690
02691
02692
02693
02694
02695
02696
02697
02698
02699
02700
02701
02702
02703
02704
02705
02706
02707
02708
02709
02710
02711
02712
02713
02714
02715
02716
02717
02718
02719
02720
02721
02722
02723
02724
02725
02726
02727
02728
02729
02730
02731
02732
02733
02734
02735
02736
02737
02738
02739
02740
02741
02742
02743
02744
02745
02746
02747
02748
02749
02750
02751
02752
02753
02754
02755
02756
02757
02758
02759
02760
02761
02762
02763
02764
02765
02766
02767
02768
02769
02770
02771
02772
02773
02774
02775
02776
02777
02778
02779
02780
02781
02782
02783
02784
02785
02786
02787
02788
02789
02790
02791
02792
02793
02794
02795
02796
02797
02798
02799
02800
02801
02802
02803
02804
02805
02806
02807
02808
02809
02810
02811
02812
02813
02814
02815
02816
02817
02818
02819
02820
02821
02822
02823
02824
02825
02826
02827
02828
02829
02830
02831
02832
02833
02834
02835
02836
02837
02838
02839
02840
02841
02842
02843
02844
02845
02846
02847
02848
02849
02850
02851
02852
02853
02854
02855
02856
02857
02858
02859
02860
02861
02862
02863
02864
02865
02866
02867
02868
02869
02870
02871
02872
02873
02874
02875
02876
02877
02878
02879
02880
02881
02882
02883
02884
02885
02886
02887
02888
02889
02890
02891
02892
02893
02894
02895
02896
02897
02898
02899
02900
02901
02902
02903
02904
02905
02906
02907
02908
02909
02910
02911
02912
02913
02914
02915
02916
02917
02918
02919
02920
02921
02922
02923
02924
02925
02926
02927
02928
02929
02930
02931
02932
02933
02934
02935
02936
02937
02938
02939
02940
02941
02942
02943
02944
02945
02946
02947
02948
02949
02950
02951
02952
02953
02954
02955
02956
02957
02958
02959
02960
02961
02962
02963
02964
02965
02966
02967
02968
02969
02970
02971
02972
02973
02974
02975
02976
02977
02978
02979
02980
02981
02982
02983
02984
02985
02986
02987
02988
02989
02990
02991
02992
02993
02994
02995
02996
02997
02998
02999
02999
03000
03001
03002
03003
03004
03005
03006
03007
03008
03009
030010
030011
030012
030013
030014
030015
030016
030017
030018
030019
030020
030021
030022
030023
030024
030025
030026
030027
030028
030029
030030
030031
030032
030033
030034
030035
030036
030037
030038
030039
030040
030041
030042
030043
030044
030045
030046
030047
030048
030049
030050
030051
030052
030053
030054
030055
030056
030057
030058
030059
030060
030061
030062
030063
030064
030065
030066
030067
030068
030069
030070
030071
030072
030073
030074
030075
030076
030077
030078
030079
030080
030081
030082
030083
030084
030085
030086
030087
030088
030089
030090
030091
030092
030093
030094
030095
030096
030097
030098
030099
030099
030100
030101
030102
030103
030104
030105
030106
030107
030108
030109
030110
030111
030112
030113
030114
030115
030116
030117
030118
030119
030120
030121
030122
030123
030124
030125
030126
030127
030128
030129
030130
030131
030132
030133
030134
030135
030136
030137
030138
030139
030140
030141
030142
030143
030144
030145
030146
030147
030148
030149
030150
030151
030152
030153
030154
030155
030156
030157
030158
030159
030160
030161
030162
030163
030164
030165
030166
030167
030168
030169
030170
030171
030172
030173
030174
030175
030176
030177
030178
030179
030180
030181
030182
030183
030184
030185
030186
030187
030188
030189
030190
030191
030192
030193
030194
030195
030196
030197
030198
030199
030199
030200
030201
030202
030203
030204
030205
030206
030207
030208
030209
030210
030211
030212
030213
030214
030215
030216
030217
030218
030219
030220
030221
030222
030223
030224
030225
030226
030227
030228
030229
030229
030230
030231
030232
030233
030234
030235
030236
030237
030238
030239
030239
030240
030241
030242
030243
030244
030245
030246
030247
030248
030249
030249
030250
030251
030252
030253
030254
030255
030256
030257
030258
030259
030259
030260
030261
030262
030263
030264
030265
030266
030267
030268
030269
030269
030270
030271
030272
030273
030274
030275
030276
030277
030278
030279
030279
030280
030281
030282
030283
030284
030285
030286
030287
030288
030289
030289
030290
030291
030292
030293
030294
030295
030296
030297
030298
030299
030299
030300
030301
030302
030303
030304
030305
030306
030307
030308
030309
030309
030310
030311
030312
030313
030314
030315
030316
030317
030318
030319
030319
030320
030321
030322
030323
030324
030325
030326
030327
030328
030329
030329
030330
030331
030332
030333
030334
030335
030336
030337
030338
030339
030339
030340
030341
030342
030343
030344
030345
030346
030347
030348
030349
030349
030350
030351
030352
030353
030354
030355
030356
030357
030358
030359
030359
030360
030361
030362
030363
030364
030365
030366
030367
030368
030369
030369
030370
030371
030372
030373
030374
030375
030376
030377
030378
030379
030379
030380
030381
030382
030383
030384
030385
030386
030387
030388
030389
030389
030390
030391
030392
030393
030394
030395
030396
030397
030398
030399
030399
030400
030401
030402
030403
030404
030405
030406
030407
030408
030409
030409
030410
030411
030412
030413
030414
030415
030416
030417
030418
030419
030419
030420
030421
030422
030423
030424
030425
030426
030427
030428
030429
030429
030430
030431
030432
030433
030434
030435
030436
030437
030438
030439
030439
030440
030441
030442
030443
030444
030445
030446
030447
030448
030449
030449
030450
030451
030452
030453
030454
030455
030456
030457
030458
030459
030459
030460
030461
030462
030463
030464
030465
030466
030467
030468
030469
030469
030470
030471
030472
030473
030474
030475
030476
030477
030478
030479
030479
030480
030481
030482
030483
030484
030485
030486
030487
030488
030489
030489
030490
030491
030492
030493
030494
030495
030496
030497
030498
030499
030499
030500
030501
030502
030503
030504
030505
030506
030507
030508
030509
030509
030510
030511
030512
030513
030514
030515
030516
030517
030518
030519
030519
030520
030521
030522
030523
030524
030525
030526
030527
030528
030529
030529
030530
030531
030532
030533
030534
030535
030536
030537
030538
030539
030539
030540
030541
030542
030543
030544
030545
030546
030547
030548
030549
030549
030550
030551
030552
030553
030554
030555
030556
030557
030558
030559
030559
030560
030561
030562
030563
030564
030565
030566
030567
030568
030569
030569
030570
030571
030572
030573
030574
030575
030576
030577
030578
030579
030579
030580
030581
030582
030583
030584
030585
030586
030587
030588
030589
030589
030590
030591
030592
030593
030594
030595
030596
030597
030598
030599
030599
030600
030601
030602
030603
030604
030605
030606
030607
030608
030609
030609
030610
030611
030612
030613
030614
030615
030616
030617
030618
030619
030619
030620
030621
030622
030623
030624
030625
030626
030627
030628
030629
030629
030630
030631
030632
030633
030634
030635
030636
030637
030638
030639
030639
030640
030641
030642
030643
030644
030645
030646
030647
030648
030649
030649
030650
030651
030652
030653
030654
030655
030656
030657
030658
030659
030659
030660
030661
030662
030663
030664
030665
030666
030667
030668
030669
030669
030670
030671
030672
030673
030674
030675
030676
030677
030678
030679
030679
030680
030681
030682
030683
030684
030685
030686
030687
030688
030689
030689
030690
030691
030692
030693
030694
030695
030696
030697
030698
030699
030699
030700
030701
030702
030703
030704
030705
030706
030707
030708
030709
030709
030710
030711
030712
030713
030714
030715
030716
030717
030718
030719
030719
030720
030721
030722
030723
030724
030725
030726
030727
030728
030729
030729
030730
030731
030732
030733
030734
030735
030736
030737
030738
030739
030739
030740
030741
030742
030743
030744
030745
030746
030747
030748
030749
030749
030750
030751
030752
030753
030754
030755
030756
030757
030758
030759
030759
030760
030761
030762
030763
030764
030765
030766
030767
030768
030769
030769
030770
030771
030772
030773
030774
030775
030776
030777
030778
030779
030779
030780
030781
030782
030783
030784
030785
030786
030787
030788
030789
030789
030790
030791
030792
030793
030794
030795
030796
030797
030798
030799
030799
030800
030801
030802
030803
030804
030805
030806
030807
030808
030809
030809
030810
030811
030812
030813
030814
030815
030816
030817
030818
030819
030819
030820
030821
030822
030823
030824
030825
030826
030827
030828
030829
030829
030830
030831
030832
030833
030834
030835
030836
030837
030838
030839
030839
030840
030841
030842
030843
030844
030845
030846
030847
030848
030849
030849
030850
030851
030852
030853
030854
030855
030856
030857
030858
030859
030859
030860
030861
030862
030863
030864
030865
030866
030867
030868
030869
030869
030870
030871
030872
030873
030874
030875
030876
030877
030878
030879
030879
030880
030881
030882
030883
030884
030885
030886
030887
030888
030889
030889
030890
030891
030892
030893
030894
030895
030896
030897
030898
030899
030899
030900
030901
030902
030903
030904
030905
030906
030907
030908
030909
030909
030910
030911
030912
030913
030914
030915
030916
030917
030918
030919
030919
030920
030921
030922
030923
030924
030925
030926
030927
030928
030929
030929
030930
030931
030932
030933
030934
030935
030936
030937
030938
030939
030939
030940
030941
030942
030943
030944
030945
030946
030947
030948
030949
030949
030950
030951
030952
030953
030954
030955
030956
030957
030958
030959
030959
030960
030961
030962
030963
030964
030965
030966
030967
030968
030969
030969
030970
030971
030972
030973
030974
030975
030976
030977
030978
030979
030979
030980
030981
030982
030983
030984
030985
030986
030987
030988
030989
030989
030990
030991
030992
030993
030994
030995
030996
030997
030998
030999
030999
031000
031001
031002
031003
031004
031005
031006
031007
031008
031009
031009
031010
031011
031012
031013
031014
031015
031016
031017
031018
031019
031019
031020
031021
031022
031023

```

```
02583         break;
02584     }
02585 }
02586
02587 if (txbxQuestion_arr[question].Text.Trim() != "")
02588 {
02589     SendMsg(txbxQuestion_arr[question].Text, brdcst_addr);
02590     RecordQuizData(txbxQuestion_arr[question].Text, "Teacher");
02591 }
02592 }
02593
02594 /*This allows the toggling of state in menubar, also shows the quiz*/
02595 private void quizToolStripMenuItem_Click(object sender, EventArgs e)
02596 {
02597     string file2write = @"data/Quiz/QuizData.txt";
02598
02599     if (File.Exists(file2write) == true)
02600     {
02601         /*If Only File.Move was used the 2nd or 3rd time this ran it
02602         would barf, File.Replace Fixes that*/
02603         if (File.Exists(@"data/Quiz/QuizData-oldQuiz.txt"))
02604         {
02605             File.Replace(file2write, @"data/Quiz/QuizData-oldQuiz.txt"
02606             , @"data/Quiz/QuizData-bak"); // replace the file
02607         }
02608         else
02609         {
02610             File.Move(file2write, @"data/Quiz/QuizData-oldQuiz.txt");
02611             //move to a backup
02612         }
02613     }
02614     using (FileStream fs = File.Create(file2write))
02615     {
02616         fs.Close();
02617     }
02618     quitToolStripMenuItem.Checked = true;
02619     normalToolStripMenuItem.Checked = false;
02620     classVoteToolStripMenuItem.Checked = false;
02621     attendanceToolStripMenuItem.Checked = false;
02622     StateMachine = "Quiz"; //program state
02623
02624     if (BrdcstShowing)
02625     {
02626         DesireQuizMd = true;
02627         broadcastToolStripMenuItem_Click(sender, e);
02628     }
02629     else
02630     {
02631         QuizModeClicked = true;
02632         timer.Enabled = true;
02633     }
02634 }
02635 /*This method allows for the recording of questions and answers for the
02636 quiz*/
02637 public void RecordQuizData(string question, string name_sender)
02638 {
02639     string file2write = @"data/Quiz/QuizData.txt";
02640     try
02641     {
02642         using (StreamWriter sw = File.AppendText(file2write))
02643         {
02644             sw.NewLine = "\x0A"; //sets the writeline terminator
02645             sw.WriteLine(name_sender + " (" + DateTime.Now.ToString("HH:mm:ss") + " | " + DateTime.Today.ToString("d") + ")" + ":" + "\x0A" +
02646             question + "\x0A"); //this will leave one blank line between each message
02647             /*Example:
02648             * Teacher (13:10:12 | 4/1/2012):
02649             * What is the factorial of 100?
02650             * ...
02651             */
02652         }
02653     }
02654 }
```

```

02651         sw.Close();
02652     }
02653 }
02654 catch
02655 {
02656     MessageBox.Show("Quiz Log Creation // Writing Failed!!!!");
02657 }
02658 }
02659
02660 /*Mode for the Attendance Keeping of the class*/
02661 private void attendanceToolStripMenuItem_Click(object sender, EventArgs
02662 e)
02663 {
02664     //show attendance panel
02665     attendanceToolStripMenuItem.Checked = true;
02666
02667     normalToolStripMenuItem.Checked = false;
02668     classVoteToolStripMenuItem.Checked = false;
02669     quizToolStripMenuItem.Checked = false;
02670
02671     if (BrdcstShowing)
02672     {
02673         DesireAttendance = true;
02674         broadcastToolStripMenuItem_Click(sender, e); //must call before
02675         state change
02676     }
02677     else
02678     {
02679         AttendanceClicked = true;
02680         timer.Enabled = true;
02681
02682         StateMachine = "Attendance";
02683         PanelAttendance.BringToFront();
02684         SendMsg("Reply if you're here..", brdcst_addr); //let's the
02685         students know to check-in
02686     }
02687
02688 /*Closes the attendance mode*/
02689 private void btnExitAttendance_Click(object sender, EventArgs e)
02690 {
02691     attendanceToolStripMenuItem.Checked = false;
02692     normalToolStripMenuItem.Checked = true;
02693     classVoteToolStripMenuItem.Checked = false;
02694     quizToolStripMenuItem.Checked = false;
02695
02696     StateMachine = "Normal";
02697     AttendanceClicked = true;
02698     timer.Enabled = true;
02699 }
02700
02701 /*Closes the Class Vote Mode*/
02702 private void btnExitClassVote_Click(object sender, EventArgs e)
02703 {
02704     classVoteToolStripMenuItem.Checked = false;
02705     normalToolStripMenuItem.Checked = true;
02706     attendanceToolStripMenuItem.Checked = false;
02707     quizToolStripMenuItem.Checked = false;
02708
02709     StateMachine = "Normal";
02710     ClassVoteClicked = true;
02711     timer.Enabled = true;
02712 }
02713
02714 /*Mode for hosting Class Votes*/
02715 private void classVoteToolStripMenuItem_Click(object sender, EventArgs
e)
02716 {
02717     attendanceToolStripMenuItem.Checked = false;
02718     normalToolStripMenuItem.Checked = false;
02719     classVoteToolStripMenuItem.Checked = true;
02720     quizToolStripMenuItem.Checked = false;

```

```
02721     SendMsg("Vote for option 1 or 2, please!",brdcst_addr);
02722
02723     if (BrdcstShowing)
02724     {
02725         DesireClassVote = true;
02726         broadcastToolStripMenuItem_Click(sender, e);
02727     }
02728     else
02729     {
02730         ClassVoteClicked = true;
02731         timer.Enabled = true;
02732     }
02733
02734     StateMachine = "ClassVote";
02735 }
02736
02737 /*Allows opening of Attendance mode from context menu at home screen*/
02738 private void muItmAttendance_Click(object sender, EventArgs e)
02739 {
02740     attendanceToolStripMenuItem_Click(sender, e);
02741 }
02742
02743 /*Allows the entering of Class Vote mode from home screen context menu
 */
02744 private void muItmClassVote_Click(object sender, EventArgs e)
02745 {
02746     classVoteToolStripMenuItem_Click(sender, e);
02747 }
02748
02749 /*Allows entering Quiz mode from home screen context menu*/
02750 private void muItmQuiz_Click(object sender, EventArgs e)
02751 {
02752     quizToolStripMenuItem_Click(sender, e);
02753 }
02754
02755 /*Allows for quick broadcast message from the home screen*/
02756 private void muItmBrdCst_Click(object sender, EventArgs e)
02757 {
02758     broadcastToolStripMenuItem_Click(sender, e);
02759 }
02760
02761 /*This method allows the choosing of a custom color for theming*/
02762 private void btnForeColor_Click(object sender, EventArgs e)
02763 {
02764     colorDlg.ShowDialog();
02765     ForeColorTheme = colorDlg.Color;
02766
02767     ThemeApply(ForeColorTheme, BackColorTheme);
02768     this.Show();
02769 }
02770
02771 /*This method allows the choosing of a custom color for theming*/
02772 private void btnBkgrndColor_Click(object sender, EventArgs e)
02773 {
02774     colorDlg.ShowDialog();
02775     BackColorTheme = colorDlg.Color;
02776
02777     ThemeApply(ForeColorTheme, BackColorTheme);
02778     this.Show();
02779 }
02780
02781 /*This method allows for Resetting the theme to default Purdue Colors*/
02782 private void btnDefaultTheme_Click(object sender, EventArgs e)
02783 {
02784     ForeColorTheme = Color.DarkGoldenrod;
02785     BackColorTheme = Color.Black;
02786     ThemeApply(Color.DarkGoldenrod, Color.Black); //default theme
02787     this.Show();
02788 }
02789
02790 /*Allows for the speed of animation preference*/
02791 private void cmbbxAnimationSpeed_SelectedIndexChanged(object sender,
EventArgs e)
02792 {
```

```

02793     switch (cmbbxAnimationSpeed.SelectedIndex)
02794     {
02795         case (-1): //the user selected the text which says speed...
02796             break;
02797
02798         case (0): //slow
02799             timer.Interval = 48; //twice as long as normal
02800             break;
02801
02802         case (1): //normal
02803             timer.Interval = 24; //default setting
02804             break;
02805
02806         case (2): //fast
02807             timer.Interval = 12; //twice as fast as normal
02808             break;
02809
02810         case (3): //blazing
02811             timer.Interval = 6; //4 times faster than normal
02812             break;
02813
02814         default:
02815             timer.Interval = 25;
02816             break;
02817     }
02818 }
02819
02820 /*Allows for the hiding of the broadcast panel when it loses focus*/
02821 private void pnlBrdCst_Leave(object sender, EventArgs e)
02822 {
02823     if (BrdcstShowing)
02824     {
02825         btnCLS_Click_BrdCst(sender, e);
02826     }
02827 }
02828
02829 /*Allows dismissing the notify panel by clicking it*/
02830 private void lblNotify_MouseDown(object sender, MouseEventArgs e)
02831 {
02832     notif_I = (int)numUpDnNotify.Value * 100; //next tick the panel
02833     will close
02834     timer_Tick(sender, e);
02835 }
02836
02837 /*Allows dismissing the notify panel by clicking it*/
02838 private void pictureBoxNotify_Click(object sender, EventArgs e)
02839 {
02840     notif_I = (int)numUpDnNotify.Value * 100; //next tick the panel
02841     will close
02842     timer_Tick(sender, e);
02843 }
02844
02845 /*This handles the votes of the kiddies*/
02846 private void ClassVoteHandler(string vote, int student)
02847 {
02848     int tmp = 0;
02849     int numLeft = Convert.ToInt16(lblCVLeft.Text);
02850     int numRight = Convert.ToInt16(lblCVRight.Text);
02851     decimal percentMajority = 0;
02852
02853     if (StudentVoted[student] == false)
02854     {
02855         if (int.TryParse(vote, out tmp))
02856         {
02857             if (tmp > 0 && tmp < 3)
02858             {
02859                 StudentVoted[student] = true; //prevents double voting!
02860
02861                 if (tmp == 1)
02862                 {
02863                     numLeft++;
02864                 }
02865                 else if (tmp == 2)
02866                 {
02867
02868
02869
02870
02871
02872
02873
02874
02875
02876
02877
02878
02879
02880
02881
02882
02883
02884
02885
02886
02887
02888
02889
02890
02891
02892
02893
02894
02895
02896
02897
02898
02899
02900
02901
02902
02903
02904
02905
02906
02907
02908
02909
02910
02911
02912
02913
02914
02915
02916
02917
02918
02919
02920
02921
02922
02923
02924
02925
02926
02927
02928
02929
02930
02931
02932
02933
02934
02935
02936
02937
02938
02939
02940
02941
02942
02943
02944
02945
02946
02947
02948
02949
02950
02951
02952
02953
02954
02955
02956
02957
02958
02959
02960
02961
02962
02963
02964
02965
02966
02967
02968
02969
02970
02971
02972
02973
02974
02975
02976
02977
02978
02979
02980
02981
02982
02983
02984
02985
02986
02987
02988
02989
02990
02991
02992
02993
02994
02995
02996
02997
02998
02999
03000
03001
03002
03003
03004
03005
03006
03007
03008
03009
03010
03011
03012
03013
03014
03015
03016
03017
03018
03019
03020
03021
03022
03023
03024
03025
03026
03027
03028
03029
03030
03031
03032
03033
03034
03035
03036
03037
03038
03039
03040
03041
03042
03043
03044
03045
03046
03047
03048
03049
03050
03051
03052
03053
03054
03055
03056
03057
03058
03059
03060
03061
03062
03063
03064
03065
03066
03067
03068
03069
03070
03071
03072
03073
03074
03075
03076
03077
03078
03079
03080
03081
03082
03083
03084
03085
03086
03087
03088
03089
03090
03091
03092
03093
03094
03095
03096
03097
03098
03099
03100
03101
03102
03103
03104
03105
03106
03107
03108
03109
03110
03111
03112
03113
03114
03115
03116
03117
03118
03119
03120
03121
03122
03123
03124
03125
03126
03127
03128
03129
03130
03131
03132
03133
03134
03135
03136
03137
03138
03139
03140
03141
03142
03143
03144
03145
03146
03147
03148
03149
03150
03151
03152
03153
03154
03155
03156
03157
03158
03159
03160
03161
03162
03163
03164
03165
03166
03167
03168
03169
03170
03171
03172
03173
03174
03175
03176
03177
03178
03179
03180
03181
03182
03183
03184
03185
03186
03187
03188
03189
03190
03191
03192
03193
03194
03195
03196
03197
03198
03199
03200
03201
03202
03203
03204
03205
03206
03207
03208
03209
03210
03211
03212
03213
03214
03215
03216
03217
03218
03219
03220
03221
03222
03223
03224
03225
03226
03227
03228
03229
03230
03231
03232
03233
03234
03235
03236
03237
03238
03239
03240
03241
03242
03243
03244
03245
03246
03247
03248
03249
03250
03251
03252
03253
03254
03255
03256
03257
03258
03259
03260
03261
03262
03263
03264
03265
03266
03267
03268
03269
03270
03271
03272
03273
03274
03275
03276
03277
03278
03279
03280
03281
03282
03283
03284
03285
03286
03287
03288
03289
03290
03291
03292
03293
03294
03295
03296
03297
03298
03299
03300
03301
03302
03303
03304
03305
03306
03307
03308
03309
03310
03311
03312
03313
03314
03315
03316
03317
03318
03319
03320
03321
03322
03323
03324
03325
03326
03327
03328
03329
03330
03331
03332
03333
03334
03335
03336
03337
03338
03339
03340
03341
03342
03343
03344
03345
03346
03347
03348
03349
03350
03351
03352
03353
03354
03355
03356
03357
03358
03359
03360
03361
03362
03363
03364
03365
03366
03367
03368
03369
03370
03371
03372
03373
03374
03375
03376
03377
03378
03379
03380
03381
03382
03383
03384
03385
03386
03387
03388
03389
03390
03391
03392
03393
03394
03395
03396
03397
03398
03399
03400
03401
03402
03403
03404
03405
03406
03407
03408
03409
03410
03411
03412
03413
03414
03415
03416
03417
03418
03419
03420
03421
03422
03423
03424
03425
03426
03427
03428
03429
03430
03431
03432
03433
03434
03435
03436
03437
03438
03439
03440
03441
03442
03443
03444
03445
03446
03447
03448
03449
03450
03451
03452
03453
03454
03455
03456
03457
03458
03459
03460
03461
03462
03463
03464
03465
03466
03467
03468
03469
03470
03471
03472
03473
03474
03475
03476
03477
03478
03479
03480
03481
03482
03483
03484
03485
03486
03487
03488
03489
03490
03491
03492
03493
03494
03495
03496
03497
03498
03499
03500
03501
03502
03503
03504
03505
03506
03507
03508
03509
03510
03511
03512
03513
03514
03515
03516
03517
03518
03519
03520
03521
03522
03523
03524
03525
03526
03527
03528
03529
03530
03531
03532
03533
03534
03535
03536
03537
03538
03539
03540
03541
03542
03543
03544
03545
03546
03547
03548
03549
03550
03551
03552
03553
03554
03555
03556
03557
03558
03559
03560
03561
03562
03563
03564
03565
03566
03567
03568
03569
03570
03571
03572
03573
03574
03575
03576
03577
03578
03579
03580
03581
03582
03583
03584
03585
03586
03587
03588
03589
03590
03591
03592
03593
03594
03595
03596
03597
03598
03599
03600
03601
03602
03603
03604
03605
03606
03607
03608
03609
03610
03611
03612
03613
03614
03615
03616
03617
03618
03619
03620
03621
03622
03623
03624
03625
03626
03627
03628
03629
03630
03631
03632
03633
03634
03635
03636
03637
03638
03639
03640
03641
03642
03643
03644
03645
03646
03647
03648
03649
03650
03651
03652
03653
03654
03655
03656
03657
03658
03659
03660
03661
03662
03663
03664
03665
03666
03667
03668
03669
03670
03671
03672
03673
03674
03675
03676
03677
03678
03679
03680
03681
03682
03683
03684
03685
03686
03687
03688
03689
03690
03691
03692
03693
03694
03695
03696
03697
03698
03699
03700
03701
03702
03703
03704
03705
03706
03707
03708
03709
03710
03711
03712
03713
03714
03715
03716
03717
03718
03719
03720
03721
03722
03723
03724
03725
03726
03727
03728
03729
03730
03731
03732
03733
03734
03735
03736
03737
03738
03739
03740
03741
03742
03743
03744
03745
03746
03747
03748
03749
03750
03751
03752
03753
03754
03755
03756
03757
03758
03759
03760
03761
03762
03763
03764
03765
03766
03767
03768
03769
03770
03771
03772
03773
03774
03775
03776
03777
03778
03779
03780
03781
03782
03783
03784
03785
03786
03787
03788
03789
03790
03791
03792
03793
03794
03795
03796
03797
03798
03799
03800
03801
03802
03803
03804
03805
03806
03807
03808
03809
03810
03811
03812
03813
03814
03815
03816
03817
03818
03819
03820
03821
03822
03823
03824
03825
03826
03827
03828
03829
03830
03831
03832
03833
03834
03835
03836
03837
03838
03839
03840
03841
03842
03843
03844
03845
03846
03847
03848
03849
03850
03851
03852
03853
03854
03855
03856
03857
03858
03859
03860
03861
03862
03863
03864
03865
03866
03867
03868
03869
03870
03871
03872
03873
03874
03875
03876
03877
03878
03879
03880
03881
03882
03883
03884
03885
03886
03887
03888
03889
03890
03891
03892
03893
03894
03895
03896
03897
03898
03899
03900
03901
03902
03903
03904
03905
03906
03907
03908
03909
03910
03911
03912
03913
03914
03915
03916
03917
03918
03919
03920
03921
03922
03923
03924
03925
03926
03927
03928
03929
03930
03931
03932
03933
03934
03935
03936
03937
03938
03939
03940
03941
03942
03943
03944
03945
03946
03947
03948
03949
03950
03951
03952
03953
03954
03955
03956
03957
03958
03959
03960
03961
03962
03963
03964
03965
03966
03967
03968
03969
03970
03971
03972
03973
03974
03975
03976
03977
03978
03979
03980
03981
03982
03983
03984
03985
03986
03987
03988
03989
03990
03991
03992
03993
03994
03995
03996
03997
03998
03999
04000
04001
04002
04003
04004
04005
04006
04007
04008
04009
04010
04011
04012
04013
04014
04015
04016
04017
04018
04019
04020
04021
04022
04023
04024
04025
04026
04027
04028
04029
04030
04031
04032
04033
04034
04035
04036
04037
04038
04039
04040
04041
04042
04043
04044
04045
04046
04047
04048
04049
04050
04051
04052
04053
04054
04055
04056
04057
04058
04059
04060
04061
04062
04063
04064
04065
04066
04067
04068
04069
04070
04071
04072
04073
04074
04075
04076
04077
04078
04079
04080
04081
04082
04083
04084
04085
04086
04087
04088
04089
04090
04091
04092
04093
04094
04095
04096
04097
04098
04099
04100
04101
04102
04103
04104
04105
04106
04107
04108
04109
04110
04111
04112
04113
04114
04115
04116
04117
04118
04119
04120
04121
04122
04123
04124
04125
04126
04127
04128
04129
04130
04131
04132
04133
04134
04135
04136
04137
04138
04139
04140
04141
04142
04143
04144
04145
04146
04147
04148
04149
04150
04151
04152
04153
04154
04155
04156
04157
04158
04159
04160
04161
04162
04163
04164
04165
04166
04167
04168
04169
04170
04171
04172
04173
04174
04175
04176
04177
04178
04179
04180
04181
04182
04183
04184
04185
04186
04187
04188
04189
04190
04191
04192
04193
04194
04195
04196
04197
04198
04199
04200
04201
04202
04203
04204
04205
04206
04207
04208
04209
04210
04211
04212
04213
04214
04215
04216
04217
04218
04219
04220
04221
04222
04223
04224
04225
04226
04227
04228
04229
04230
04231
04232
04233
04234
04235
04236
04237
04238
04239
04240
04241
04242
04243
04244
04245
04246
04247
04248
04249
04250
04251
04252
04253
04254
04255
04256
04257
04258
04259
04260
04261
04262
04263
04264
04265
04266
04267
04268
04269
04270
04271
04272
04273
04274
04275
04276
04277
04278
04279
04280
04281
04282
04283
04284
04285
04286
04287
04288
04289
04290
04291
04292
04293
04294
04295
04296
04297
04298
04299
04300
04301
04302
04303
04304
04305
04306
04307
04308
04309
04310
04311
04312
04313
04314
04315
04316
04317
04318
04319
04320
04321
04322
04323
04324
04325
04326
04327
04328
04329
04330
04331
04332
04333
04334
04335
04336
04337
04338
04339
04340
04341
04342
04343
04344
04345
04346
04347
04348
04349
04350
04351
04352
04353
04354
04355
04356
04357
04358
04359
04360
04361
04362
04363
04364
04365
04366
04367
04368
04369
04370
04371
04372
04373
04374
04375
04376
04377
04378
04379
04380
04381
04382
04383
04384
04385
04386
04387
04388
04389
04390
04391
04392
04393
04394
04395
04396
04397
04398
04399
04400
04401
04402
04403
04404
04405
04406
04407
04408
04409
04410
04411
04412
04413
04414
04415
04416
04417
04418
04419
04420
04421
04422
04423
04424
04425
04426
04427
04428
04429
04430
04431
04432
04433
04434
04435
04436
04437
04438
04439
04440
04441
04442
04443
04444
04445
04446
04447
04448
04449
04450
04451
04452
04453
04454
04455
04456
04457
04458
04459
04460
04461
04462
04463
04464
04465
04466
04467
04468
04469
04470
04471
04472
04473
04474
04475
04476
04477
04478
04479
04480
04481
04482
04483
04484
04485
04486
04487
04488
04489
04490
04491
04492
04493
04494
04495
04496
04497
04498
04499
04500
04501
04502
04503
04504
04505
04506
04507
04508
04509
04510
04511
04512
04513
04514
04515
04516
04517
04518
04519
04520
04521
04522
04523
04524
04525
04526
04527
04528
04529
04530
04531
04532
04533
04534
04535
04536
04537
04538
04539
04540
04541
04542
04543
04544
04545
04546
04547
04548
04549
04550
04551
04552
04553
04554
04555
04556
04557
04558
04559
04560
04561
04562
04563
04564
04565
04566
04567
04568
04569
0457
```

```
02865             numRight++;
02866         }
02867
02868         if (numLeft != numRight)
02869     {
02870         if (numLeft > numRight)
02871     {
02872             lblCVLeft.Font = new Font(lblCVLeft.Font,
02873             FontStyle.Bold);
02874             lblCVLeft.Font = new Font(lblCVLeft.Font,
02875             FontStyle.Underline);
02876
02877             lblCVRight.Font = new Font(lblCVRight.Font,
02878             FontStyle.Regular);
02879             lblCVLeft.Text = numLeft.ToString(); //update
02880             number
02881             percentMajority = ((decimal)numLeft / (numLeft
02882             + numRight));
02883         }
02884         else
02885     {
02886         lblCVRight.Font = new Font(lblCVRight.Font,
02887         FontStyle.Bold);
02888         lblCVRight.Font = new Font(lblCVRight.Font,
02889         FontStyle.Underline);
02890
02891         lblCVLeft.Font = new Font(lblCVLeft.Font,
02892         FontStyle.Regular);
02893         lblCVRight.Text = numRight.ToString(); //update
02894         number
02895         percentMajority = ((decimal)numRight / (numLeft
02896             + numRight));
02897     }
02898
02899         txtbxCVStats.Text = "Option #"
02900             + (numLeft >
02901             numRight ? "1" : "2") + " is winning with "
02902             + percentMajority.ToString("P") + " of the
02903             vote!";
02904     }
02905     else
02906     {
02907         lblCVRight.Font = new Font(lblCVRight.Font,
02908         FontStyle.Italic);
02909         lblCVLeft.Font = new Font(lblCVLeft.Font, FontStyle
02910             .Italic);
02911
02912         lblCVLeft.Text = numLeft.ToString(); //update number
02913         lblCVRight.Text = numRight.ToString(); //update
02914         number
02915
02916         txtbxCVStats.Text = "After "
02917             + (numLeft + numRight)
02918             .ToString() + " votes, we're all tied up!";
02919     }
02920
02921     /*This allows for the reset of voting so that there may be more than
02922     */
```

```

    one per app run*/
02923     private void btnCVReset_Click(object sender, EventArgs e)
02924     {
02925         for (int i = 0; i < classSize; i++)
02926         {
02927             StudentVoted[i] = false;
02928         }
02929
02930         lblCVLeft.Text = "0";
02931         lblCVRight.Text = "0";
02932
02933         lblCVRight.Font = new Font(lblCVRight.Font, FontStyle.Regular);
02934         lblCVLeft.Font = new Font(lblCVLeft.Font, FontStyle.Regular);
02935
02936         txtbxCVStats.Text = "...Stats of Vote...";
02937
02938         //let'em know it's done..
02939         lblNotify.Text = "Class Vote reset!";
02940         DesireNotify = true;
02941         timer.Enabled = true;
02942     }
02943
02944     private void muItmMarkAsRead_Click(object sender, EventArgs e)
02945     {
02946         Q_status[new_lblID] = true; // so unread won't get decremented if
02947         opened again..
02948         UnreadCount--;
02949         lbl_arr[new_lblID].Text = lbl_arr[new_lblID].Text.TrimStart('*');
02950         TrimStart(' '); //removes unread indicator in the label
02951
02952         if (UnreadCount > 0)
02953         {
02954             this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02955             ToString() + ")";
02956         }
02957         else
02958         {
02959             this.Text = " Classroom Inquisition | Home";
02960         }
02961
02962     private void muItmMarkAsUnRead_Click(object sender, EventArgs e)
02963     {
02964         Q_status[new_lblID] = false;
02965         UnreadCount++;
02966         this.Text = " Classroom Inquisition | Home (" + UnreadCount.
02967         ToString() + ")";
02968         lbl_arr[new_lblID].Text = "***" + " " + lbl_arr[new_lblID].
02969         .Text; //prepend the stars
02970     } //end of partial class
02971 } //end of namespace

```

6.9 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1-dyn-ctrls.Designer.cs File Reference

Variables

- namespace [SrP_ClassroomInq](#)

6.9.1 Variable Documentation

6.9.1.1 namespace SrP_ClassroomInq

Definition at line 19 of file [Form1-dyn-ctrls.Designer.cs](#).

6.10 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1-dyn-ctrls.Designer.cs

```
00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 ****/
00019 namespace SrP_ClassroomInq
00020 {
00021     partial class frmClassrromInq
00022     {
00023         private System.ComponentModel.IContainer components = null;
00024
00025         protected override void Dispose(bool disposing)
00026         {
00027             if (disposing && (components != null))
00028             {
00029                 components.Dispose();
00030             }
00031             base.Dispose(disposing);
00032         }
00033
00034         #region Windows Form Designer generated code
00035
00036         private void InitializeComponent()
00037         {
00038             this.components = new System.ComponentModel.Container();
00039             System.ComponentModel.ComponentResourceManager resources = new
00040             System.ComponentModel.ComponentResourceManager(typeof(frmClassrromInq));
00041             this.menuStrip1 = new System.Windows.Forms.MenuStrip();
00042             this.sendToolStripMenuItem = new System.Windows.Forms.
00043             ToolStripMenuItem();
00044             this.broadcastToolStripMenuItem = new System.Windows.Forms.
00045             ToolStripMenuItem();
00046             this.directMessageToolStripMenuItem = new System.Windows.Forms.
00047             ToolStripMenuItem();
00048             this.attendanceToolStripMenuItem = new System.Windows.Forms.
00049             ToolStripMenuItem();
00050             this.classVoteToolStripMenuItem = new System.Windows.Forms.
00051             ToolStripMenuItem();
00052             this.normalToolStripMenuItem = new System.Windows.Forms.
00053             ToolStripMenuItem();
00054             this.quizToolStripMenuItem = new System.Windows.Forms.
00055             ToolStripMenuItem();
00056         }
00057     }
00058 }
```

```
00060      this.showNamesToolStripMenuItem = new System.Windows.Forms.
00061      ToolStripMenuItem();
00062      this.PrefsToolStripMenuItem = new System.Windows.Forms.
00063      ToolStripMenuItem();
00064      this.helpToolStripMenuItem = new System.Windows.Forms.
00065      ToolStripMenuItem();
00066      this.generalFAQToolStripMenuItem = new System.Windows.Forms.
00067      ToolStripMenuItem();
00068      this.aboutToolStripMenuItem = new System.Windows.Forms.
00069      ToolStripMenuItem();
00070      this.quitToolStripMenuItem = new System.Windows.Forms.
00071      ToolStripMenuItem();
00072      this.toolStripButton1 = new System.Windows.Forms.
00073      ToolStripComboBox();
00074      this.fileToolStripMenuItem = new System.Windows.Forms.
00075      ToolStripMenuItem();
00076      this.newToolStripMenuItem = new System.Windows.Forms.
00077      ToolStripMenuItem();
00078      this.openToolStripMenuItem = new System.Windows.Forms.
00079      ToolStripMenuItem();
00080      this.saveToolStripMenuItem = new System.Windows.Forms.
00081      ToolStripSeparator();
00082      this.toolStripSeparator1 = new System.Windows.Forms.
00083      ToolStripSeparator();
00084      this.printToolStripMenuItem = new System.Windows.Forms.
00085      ToolStripMenuItem();
00086      this.printPreviewToolStripMenuItem = new System.Windows.Forms.
00087      ToolStripMenuItem();
00088      this.toolStripSeparator2 = new System.Windows.Forms.
00089      ToolStripSeparator();
00090      this.exitToolStripMenuItem = new System.Windows.Forms.
00091      ToolStripMenuItem();
00092      this.editToolStripMenuItem = new System.Windows.Forms.
00093      ToolStripMenuItem();
00094      this.undoToolStripMenuItem = new System.Windows.Forms.
00095      ToolStripMenuItem();
00096      this.redoToolStripMenuItem = new System.Windows.Forms.
00097      ToolStripMenuItem();
00098      this.toolStripSeparator3 = new System.Windows.Forms.
00099      ToolStripSeparator();
00100      this.toolsToolStripMenuItem = new System.Windows.Forms.
00101      ToolStripMenuItem();
00102      this.selectAllToolStripMenuItem = new System.Windows.Forms.
00103      ToolStripMenuItem();
00104      this.toolsToolStripMenuItem = new System.Windows.Forms.
00105      ToolStripMenuItem();
00106      this.customizeToolStripMenuItem = new System.Windows.Forms.
00107      ToolStripMenuItem();
00108      this.toolStripSeparator4 = new System.Windows.Forms.
00109      ToolStripSeparator();
00110      this.optionsToolStripMenuItem = new System.Windows.Forms.
00111      ToolStripMenuItem();
00112      this.toolsToolStripMenuItem1 = new System.Windows.Forms.
00113      ToolStripMenuItem();
00114      this.contentsToolStripMenuItem = new System.Windows.Forms.
00115      ToolStripMenuItem();
00116      this.indexToolStripMenuItem = new System.Windows.Forms.
00117      ToolStripMenuItem();
00118      this.searchToolStripMenuItem = new System.Windows.Forms.
00119      ToolStripMenuItem();
00120      this.toolStripSeparator5 = new System.Windows.Forms.
00121      ToolStripSeparator();
00122      this.aboutToolStripMenuItem1 = new System.Windows.Forms.
00123      ToolStripMenuItem();
00124      this.textBox1 = new System.Windows.Forms.TextBox();
00125      this.timer = new System.Windows.Forms.Timer(this.components);
```

```
00098     this.trayICON = new System.Windows.Forms.NotifyIcon(this.components
00099 );
00100     this.grpbxFeed = new System.Windows.Forms.Panel();
00101     this.SerialPort = new System.IO.Ports.SerialPort(this.components);
00102     this.timer_SerialRead = new System.Windows.Forms.Timer(this.
00103     components);
00104     this.PanelPrefs = new System.Windows.Forms.Panel();
00105     this.lblNotifyTime = new System.Windows.Forms.Label();
00106     this.numUpDnNotify = new System.Windows.Forms.NumericUpDown();
00107     this.lblSerPt = new System.Windows.Forms.Label();
00108     this.serialCOMcmbbx = new System.Windows.Forms.ComboBox();
00109     this.lblAnimationPrefs = new System.Windows.Forms.Label();
00110     this.cmbbxAnimationSpeed = new System.Windows.Forms.ComboBox();
00111     this.grpbxTheme = new System.Windows.Forms.GroupBox();
00112     this.btnDefaultTheme = new System.Windows.Forms.Button();
00113     this.btnForeColor = new System.Windows.Forms.Button();
00114     this.btnBkgrndColor = new System.Windows.Forms.Button();
00115     this.btnAddQuizMaker = new System.Windows.Forms.Button();
00116     this.chkbxCtrlHide = new System.Windows.Forms.CheckBox();
00117     this.chkbxNotify = new System.Windows.Forms.CheckBox();
00118     this.chkbxToolips = new System.Windows.Forms.CheckBox();
00119     this.grpbxUnread = new System.Windows.Forms.GroupBox();
00120     this.rdbtnClick = new System.Windows.Forms.RadioButton();
00121     this.rdbtnHover = new System.Windows.Forms.RadioButton();
00122     this.chkbxLameMode = new System.Windows.Forms.CheckBox();
00123     this.btnStuMgmt_Prefs = new System.Windows.Forms.Button();
00124     this.chkbxTXSound = new System.Windows.Forms.CheckBox();
00125     this.chkbxRXSound = new System.Windows.Forms.CheckBox();
00126     this.btnPrefs_Cls = new System.Windows.Forms.Button();
00127     this.picbxPrefs = new System.Windows.Forms.PictureBox();
00128     this.DirectMsgPanel = new System.Windows.Forms.Panel();
00129     this.pictureBox1 = new System.Windows.Forms.PictureBox();
00130     this.btnDM_Send = new System.Windows.Forms.Button();
00131     this.btnDM_Cls = new System.Windows.Forms.Button();
00132     this.btnDM_Clr = new System.Windows.Forms.Button();
00133     this.cmbbxDM = new System.Windows.Forms.ComboBox();
00134     this.txtbxDM = new System.Windows.Forms.TextBox();
00135     this.PanelFAQ = new System.Windows.Forms.Panel();
00136     this.lnklblIssues = new System.Windows.Forms.LinkLabel();
00137     this.lblFAQ = new System.Windows.Forms.Label();
00138     this.txtbxFAQ = new System.Windows.Forms.TextBox();
00139     this.btnAddFAQcls = new System.Windows.Forms.Button();
00140     this.pictureBox2 = new System.Windows.Forms.PictureBox();
00141     this.PanelStudents = new System.Windows.Forms.Panel();
00142     this.btnConvView = new System.Windows.Forms.Button();
00143     this.grpbxEDITStudents = new System.Windows.Forms.GroupBox();
00144     this.btnAddStudents = new System.Windows.Forms.Button();
00145     this.lblStudentsName = new System.Windows.Forms.Label();
00146     this.txtbxStudentsName = new System.Windows.Forms.TextBox();
00147     this.btnAddCLRStudents = new System.Windows.Forms.Button();
00148     this.btnAddEDTStudents = new System.Windows.Forms.Button();
00149     this.lstbxStudents = new System.Windows.Forms.ListBox();
00150     this.btnAddCLSStudents = new System.Windows.Forms.Button();
00151     this.pictureBox3 = new System.Windows.Forms.PictureBox();
00152     this.timer_ControlsCreate = new System.Windows.Forms.Timer(this.
00153     components);
00154     this.PanelConvView = new System.Windows.Forms.Panel();
00155     this.btnCV_Refresh = new System.Windows.Forms.Button();
00156     this.btnCV_Print = new System.Windows.Forms.Button();
00157     this.lstbxConvView = new System.Windows.Forms.ListBox();
00158     this.btnCLS_ConvView = new System.Windows.Forms.Button();
00159     this.pictureBox4 = new System.Windows.Forms.PictureBox();
00160     this.Printer = new System.Drawing.Printing.PrintDocument();
00161     this.cntxtMenu = new System.Windows.Forms.ContextMenu();
00162     this.muiitmDelete = new System.Windows.Forms.MenuItem();
00163     this.muiitmQuizDelete = new System.Windows.Forms.MenuItem();
00164     this.muiitmPrefs = new System.Windows.Forms.MenuItem();
00165     this.muiitmDM = new System.Windows.Forms.MenuItem();
00166     this.muiitmFAQ = new System.Windows.Forms.MenuItem();
00167     this.muiitmQuit = new System.Windows.Forms.MenuItem();
00168     this.muiitmUndo = new System.Windows.Forms.MenuItem();
00169     this.muiitmName = new System.Windows.Forms.MenuItem();
00170     this.muiitmAttendance = new System.Windows.Forms.MenuItem();
00171     this.muiitmClassVote = new System.Windows.Forms.MenuItem();
```

```

00169     this.muItmQuiz = new System.Windows.Forms.MenuItem();
00170     this.muItmBrdCst = new System.Windows.Forms.MenuItem();
00171     this.muItmMarkAsRead = new System.Windows.Forms.MenuItem();
00172     this.muItmMarkAsUnRead = new System.Windows.Forms.MenuItem();
00173     this.PanelQuizMaker = new System.Windows.Forms.Panel();
00174     this.btnAddQM = new System.Windows.Forms.Button();
00175     this.btnQMDel = new System.Windows.Forms.Button();
00176     this.btnQMClr = new System.Windows.Forms.Button();
00177     this.txtbxQM = new System.Windows.Forms.TextBox();
00178     this.btnQMcls = new System.Windows.Forms.Button();
00179     this.btnQMEdit = new System.Windows.Forms.Button();
00180     this.lstbxQuizMaker = new System.Windows.Forms.ListBox();
00181     this.pictureBox5 = new System.Windows.Forms.PictureBox();
00182     this.btnQuizMode = new System.Windows.Forms.Button();
00183     this.PanelQuizMode = new System.Windows.Forms.Panel();
00184     this.pnlQuiz = new System.Windows.Forms.Panel();
00185     this.btnExitQuiz = new System.Windows.Forms.Button();
00186     this.pictureBox6 = new System.Windows.Forms.PictureBox();
00187     this.PanelAttendance = new System.Windows.Forms.Panel();
00188     this.lstbxAttendance = new System.Windows.Forms.ListBox();
00189     this.btnExitAttendance = new System.Windows.Forms.Button();
00190     this.pictureBox7 = new System.Windows.Forms.PictureBox();
00191     this.PanelClassVote = new System.Windows.Forms.Panel();
00192     this.btnCVReset = new System.Windows.Forms.Button();
00193     this.txtbxCVStats = new System.Windows.Forms.TextBox();
00194     this.lblOption2 = new System.Windows.Forms.Label();
00195     this.lblOption1 = new System.Windows.Forms.Label();
00196     this.lblCVPipe = new System.Windows.Forms.Label();
00197     this.lblCVRight = new System.Windows.Forms.Label();
00198     this.lblCVLeft = new System.Windows.Forms.Label();
00199     this.btnExitClassVote = new System.Windows.Forms.Button();
00200     this.pictureBox8 = new System.Windows.Forms.PictureBox();
00201     this.colorDlg = new System.Windows.Forms.ColorDialog();
00202     this.pnlBrdCst = new System.Windows.Forms.Panel();
00203     this.btnSend = new System.Windows.Forms.Button();
00204     this.PanelNotify = new System.Windows.Forms.Panel();
00205     this.lblNotify = new System.Windows.Forms.Label();
00206     this.pictureBoxNotify = new System.Windows.Forms.PictureBox();
00207     this.chkbxSplash = new System.Windows.Forms.CheckBox();
00208     this.menuStrip1.SuspendLayout();
00209     this.PanelPrefs.SuspendLayout();
00210     ((System.ComponentModel.ISupportInitialize)(this.numUpDnNotify)).BeginInit();
00211     this.grpbxTheme.SuspendLayout();
00212     this.grpbxUnread.SuspendLayout();
00213     ((System.ComponentModel.ISupportInitialize)(this.picbxPrefs)).BeginInit();
00214     this.DirectMsgPanel.SuspendLayout();
00215     ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).BeginInit();
00216     this.PanelFAQ.SuspendLayout();
00217     ((System.ComponentModel.ISupportInitialize)(this.pictureBox2)).BeginInit();
00218     this.PanelStudents.SuspendLayout();
00219     this.grpbxEDITStudents.SuspendLayout();
00220     ((System.ComponentModel.ISupportInitialize)(this.pictureBox3)).BeginInit();
00221     this.PanelConvView.SuspendLayout();
00222     ((System.ComponentModel.ISupportInitialize)(this.pictureBox4)).BeginInit();
00223     this.PanelQuizMaker.SuspendLayout();
00224     ((System.ComponentModel.ISupportInitialize)(this.pictureBox5)).BeginInit();
00225     this.PanelQuizMode.SuspendLayout();
00226     ((System.ComponentModel.ISupportInitialize)(this.pictureBox6)).BeginInit();
00227     this.PanelAttendance.SuspendLayout();
00228     ((System.ComponentModel.ISupportInitialize)(this.pictureBox7)).BeginInit();
00229     this.PanelClassVote.SuspendLayout();
00230     ((System.ComponentModel.ISupportInitialize)(this.pictureBox8)).BeginInit();
00231     this.pnlBrdCst.SuspendLayout();
00232     this.PanelNotify.SuspendLayout();

```

```
00233     ((System.ComponentModel.ISupportInitialize)(this.pictureBoxNotify))
00234     .BeginInit();
00235     // 
00236     // menuStrip1
00237     // 
00238     this.menuStrip1.BackColor = System.Drawing.Color.DarkGoldenrod;
00239     this.menuStrip1.Items.AddRange(new System.Windows.Forms.ToolStripItem[] {
00240         this.sendToolStripMenuItem,
00241         this.viewToolStripMenuItem,
00242         this.PrefsToolStripMenuItem,
00243         this.helpToolStripMenuItem,
00244         this.quitToolStripMenuItem,
00245         this.toolStripComboBox1,
00246         this.fileToolStripMenuItem,
00247         this.editToolStripMenuItem,
00248         this.toolsToolStripMenuItem,
00249         this.helpToolStripMenuItem1});
00250     this.menuStrip1.Location = new System.Drawing.Point(0, 0);
00251     this.menuStrip1.Name = "menuStrip1";
00252     this.menuStrip1.RenderMode = System.Windows.Forms.ToolStripRenderMode.Professional;
00253     this.menuStrip1.ShowItemToolTips = true;
00254     this.menuStrip1.Size = new System.Drawing.Size(384, 25);
00255     this.menuStrip1.TabIndex = 6;
00256     this.menuStrip1.Text = "menuStrip1";
00257     // 
00258     // sendToolStripMenuItem
00259     // 
00260     this.sendToolStripMenuItem.DropDownItems.AddRange(new System.Windows.Forms.ToolStripItem[] {
00261         this.broadcastToolStripMenuItem,
00262         this.directMessageToolStripMenuItem});
00263     this.sendToolStripMenuItem.Name = "sendToolStripMenuItem";
00264     this.sendToolStripMenuItem.Size = new System.Drawing.Size(63, 21);
00265     this.sendToolStripMenuItem.Text = "Compose";
00266     this.sendToolStripMenuItem.ToolTipText = "Choose Direct or
00267     Broadcast";
00268     // 
00269     // broadcastToolStripMenuItem
00270     // 
00271     this.broadcastToolStripMenuItem.BackColor = System.Drawing.Color.
00272     DarkGoldenrod;
00273     this.broadcastToolStripMenuItem.ForeColor = System.Drawing.Color.
00274     Black;
00275     this.broadcastToolStripMenuItem.Name = "broadcastToolStripMenuItem"
00276     ;
00277     this.broadcastToolStripMenuItem.ShortcutKeys = ((System.Windows.
00278     Forms.Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.B)));
00279     this.broadcastToolStripMenuItem.Size = new System.Drawing.Size(197,
00280     22);
00281     this.broadcastToolStripMenuItem.Text = "Broadcast";
00282     this.broadcastToolStripMenuItem.Click += new System.EventHandler(
00283     this.broadcastToolStripMenuItem_Click);
00284     // 
00285     // directMessageToolStripMenuItem
00286     // 
00287     this.directMessageToolStripMenuItem.BackColor = System.Drawing.
00288     Color.DarkGoldenrod;
00289     this.directMessageToolStripMenuItem.ForeColor = System.Drawing.
00290     Color.Black;
00291     this.directMessageToolStripMenuItem.Name = "
00292     directMessageToolStripMenuItem";
00293     this.directMessageToolStripMenuItem.ShortcutKeys = ((System.Windows.
00294     Forms.Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.D)))
00295     ;
00296     this.directMessageToolStripMenuItem.Size = new System.Drawing.Size(
00297     197, 22);
00298     this.directMessageToolStripMenuItem.Text = "Direct Message";
00299     this.directMessageToolStripMenuItem.Click += new System.
00300     EventHandler(this.directMessageToolStripMenuItem_Click);
00301     // 
00302     // viewToolStripMenuItem
00303 
```

```

00289      // 
00290      this.viewToolStripMenuItem.BackColor = System.Drawing.Color.
00291      DarkGoldenrod;
00291      this.viewToolStripMenuItem.DropDownItems.AddRange(new System.
00292      Windows.Forms.ToolStripItem[] {
00292          this.modesToolStripMenuItem,
00293          this.showNamesToolStripMenuItem});
00294      this.viewToolStripMenuItem.Name = "viewToolStripMenuItem";
00295      this.viewToolStripMenuItem.Size = new System.Drawing.Size(41, 21);
00296      this.viewToolStripMenuItem.Text = "View";
00297      //
00298      // modesToolStripMenuItem
00299      //
00300      this.modesToolStripMenuItem.BackColor = System.Drawing.Color.
00301      DarkGoldenrod;
00301      this.modesToolStripMenuItem.DropDownItems.AddRange(new System.
00302      Windows.Forms.ToolStripItem[] {
00302          this.attendanceToolStripMenuItem,
00303          this.classVoteToolStripMenuItem,
00304          this.normalToolStripMenuItem,
00305          this.quizToolStripMenuItem});
00306      this.modesToolStripMenuItem.ForeColor = System.Drawing.Color.Black;
00307      this.modesToolStripMenuItem.Name = "modesToolStripMenuItem";
00308      this.modesToolStripMenuItem.Size = new System.Drawing.Size(185, 22)
00308      ;
00309      this.modesToolStripMenuItem.Text = "Modes";
00310      //
00311      // attendanceToolStripMenuItem
00312      //
00313      this.attendanceToolStripMenuItem.BackColor = System.Drawing.Color.
00314      DarkGoldenrod;
00314      this.attendanceToolStripMenuItem.ForeColor = System.Drawing.Color.
00315      Black;
00315      this.attendanceToolStripMenuItem.Name = "attendanceToolStripMenuItem"
00315      ;
00316      this.attendanceToolStripMenuItem.Size = new System.Drawing.Size(135,
00316      22);
00317      this.attendanceToolStripMenuItem.Text = "Attendance";
00318      this.attendanceToolStripMenuItem.Click += new System.EventHandler(
00318      this.attendanceToolStripMenuItem_Click);
00319      //
00320      // classVoteToolStripMenuItem
00321      //
00322      this.classVoteToolStripMenuItem.BackColor = System.Drawing.Color.
00323      DarkGoldenrod;
00323      this.classVoteToolStripMenuItem.ForeColor = System.Drawing.Color.
00324      Black;
00324      this.classVoteToolStripMenuItem.Name = "classVoteToolStripMenuItem"
00324      ;
00325      this.classVoteToolStripMenuItem.Size = new System.Drawing.Size(135,
00325      22);
00326      this.classVoteToolStripMenuItem.Text = "Class Vote";
00327      this.classVoteToolStripMenuItem.Click += new System.EventHandler(
00327      this.classVoteToolStripMenuItem_Click);
00328      //
00329      // normalToolStripMenuItem
00330      //
00331      this.normalToolStripMenuItem.BackColor = System.Drawing.Color.
00332      DarkGoldenrod;
00332      this.normalToolStripMenuItem.Checked = true;
00333      this.normalToolStripMenuItem.CheckState = System.Windows.Forms.
00334      CheckState.Checked;
00334      this.normalToolStripMenuItem.ForeColor = System.Drawing.Color.Black
00334      ;
00335      this.normalToolStripMenuItem.Name = "normalToolStripMenuItem";
00336      this.normalToolStripMenuItem.Size = new System.Drawing.Size(135, 22
00336      );
00337      this.normalToolStripMenuItem.Text = "Normal";
00338      //
00339      // quizToolStripMenuItem
00340      //
00341      this.quizToolStripMenuItem.BackColor = System.Drawing.Color.
00342      DarkGoldenrod;
00342      this.quizToolStripMenuItem.ForeColor = System.Drawing.Color.Black;

```

```
00343         this.quizToolStripMenuItem.Name = "quizToolStripMenuItem";
00344         this.quizToolStripMenuItem.Size = new System.Drawing.Size(135, 22);
00345         this.quizToolStripMenuItem.Text = "Quiz";
00346         this.quizToolStripMenuItem.Click += new System.EventHandler(this.
    quizToolStripMenuItem_Click);
00347         //
00348         // showNamesToolStripMenuItem
00349         //
00350         this.showNamesToolStripMenuItem.AutoToolTip = true;
00351         this.showNamesToolStripMenuItem.BackColor = System.Drawing.Color.
    DarkGoldenrod;
00352         this.showNamesToolStripMenuItem.Checked = true;
00353         this.showNamesToolStripMenuItem.CheckState = System.Windows.Forms.
    CheckState.Checked;
00354         this.showNamesToolStripMenuItem.ForeColor = System.Drawing.Color.
    Black;
00355         this.showNamesToolStripMenuItem.Name = "showNamesToolStripMenuItem"
    ;
00356         this.showNamesToolStripMenuItem.ShortcutKeys = ((System.Windows.
    Forms.Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.N)));
00357         this.showNamesToolStripMenuItem.Size = new System.Drawing.Size(185,
    22);
00358         this.showNamesToolStripMenuItem.Text = "Show Names";
00359         this.showNamesToolStripMenuItem.ToolTipText = "Toggles Name
    Appending on Message Receive\r\n                                (This Applies Globall" +
00360             "y.)";
00361         this.showNamesToolStripMenuItem.Click += new System.EventHandler(
    this.showNamesToolStripMenuItem_Click);
00362         //
00363         // PrefsToolStripMenuItem
00364         //
00365         this.PrefsToolStripMenuItem.Name = "PrefsToolStripMenuItem";
00366         this.PrefsToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
    Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.P)));
00367         this.PrefsToolStripMenuItem.Size = new System.Drawing.Size(77, 21);
00368         this.PrefsToolStripMenuItem.Text = "&Preferences";
00369         this.PrefsToolStripMenuItem.ToolTipText = "Click to Edit/View";
00370         this.PrefsToolStripMenuItem.Click += new System.EventHandler(this.
    PrefsToolStripMenuItem_Click);
00371         //
00372         // helpToolStripMenuItem
00373         //
00374         this.helpToolStripMenuItem.DropDownItems.AddRange(new System.
    Windows.Forms.ToolStripItem[]
    {
00375             this.generalFAQToolStripMenuItem,
00376             this.aboutToolStripMenuItem});
00377         this.helpToolStripMenuItem.Name = "helpToolStripMenuItem";
00378         this.helpToolStripMenuItem.Size = new System.Drawing.Size(40, 21);
00379         this.helpToolStripMenuItem.Text = "Help";
00380         //
00381         // generalFAQToolStripMenuItem
00382         //
00383         this.generalFAQToolStripMenuItem.BackColor = System.Drawing.Color.
    DarkGoldenrod;
00384         this.generalFAQToolStripMenuItem.ForeColor = System.Drawing.Color.
    Black;
00385         this.generalFAQToolStripMenuItem.Name = "
    generalFAQToolStripMenuItem";
00386         this.generalFAQToolStripMenuItem.Size = new System.Drawing.Size(146
    , 22);
00387         this.generalFAQToolStripMenuItem.Text = "General FAQ";
00388         this.generalFAQToolStripMenuItem.Click += new System.EventHandler(
    this.generalFAQToolStripMenuItem_Click);
00389         //
00390         // aboutToolStripMenuItem
00391         //
00392         this.aboutToolStripMenuItem.BackColor = System.Drawing.Color.
    DarkGoldenrod;
00393         this.aboutToolStripMenuItem.ForeColor = System.Drawing.Color.Black;
00394         this.aboutToolStripMenuItem.Name = "aboutToolStripMenuItem";
00395         this.aboutToolStripMenuItem.Size = new System.Drawing.Size(146, 22)
    ;
00396         this.aboutToolStripMenuItem.Text = "About";
00397         this.aboutToolStripMenuItem.Click += new System.EventHandler(this.
```

```

        aboutToolStripMenuItem_Click);
00398         //quitToolStripMenuItem
00399         //quitToolStripMenuItem
00400         //
00401         this.quitToolStripMenuItem.Alignment = System.Windows.Forms.
    ToolStripItemAlignment.Right;
00402         this.quitToolStripMenuItem.Name = "quitToolStripMenuItem";
00403         this.quitToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
    Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.Q)));
00404         this.quitToolStripMenuItem.Size = new System.Drawing.Size(39, 21);
00405         this.quitToolStripMenuItem.Text = "&Quit";
00406         this.quitToolStripMenuItem.ToolTipText = "To Quit use \"Ctrl + Q\""
;
00407         this.quitToolStripMenuItem.Click += new System.EventHandler(this.
    quitToolStripMenuItem_Click);
00408         //
00409         //toolStripComboBox1
00410         //
00411         this.toolStripComboBox1.Name = "toolStripComboBox1";
00412         this.toolStripComboBox1.Size = new System.Drawing.Size(121, 21);
00413         //
00414         //fileToolStripMenuItem
00415         //
00416         this.fileToolStripMenuItem.DropDownItems.AddRange(new System.
    Windows.Forms.ToolStripItem[] {
00417             this.newToolStripMenuItem,
00418             this.openToolStripMenuItem,
00419             this.toolStripSeparator,
00420             this.saveToolStripMenuItem,
00421             this.saveAsToolStripMenuItem,
00422             this.toolStripSeparator1,
00423             this.printToolStripMenuItem,
00424             this.printPreviewToolStripMenuItem,
00425             this.toolStripSeparator2,
00426             this.exitToolStripMenuItem});
00427         this.fileToolStripMenuItem.Name = "fileToolStripMenuItem";
00428         this.fileToolStripMenuItem.Size = new System.Drawing.Size(35, 21);
00429         this.fileToolStripMenuItem.Text = "&File";
00430         //
00431         //newToolStripMenuItem
00432         //
00433         this.newToolStripMenuItem.Image = ((System.Drawing.Image)(resources.
    GetObject("newToolStripMenuItem.Image")));
00434         this.newToolStripMenuItem.ImageTransparentColor = System.Drawing.
    Color.Magenta;
00435         this.newToolStripMenuItem.Name = "newToolStripMenuItem";
00436         this.newToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
    Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.N)));
00437         this.newToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00438         this.newToolStripMenuItem.Text = "&New";
00439         //
00440         //openToolStripMenuItem
00441         //
00442         this.openToolStripMenuItem.Image = ((System.Drawing.Image)(
    resources.GetObject("openToolStripMenuItem.Image")));
00443         this.openToolStripMenuItem.ImageTransparentColor = System.Drawing.
    Color.Magenta;
00444         this.openToolStripMenuItem.Name = "openToolStripMenuItem";
00445         this.openToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
    Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.O)));
00446         this.openToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00447         this.openToolStripMenuItem.Text = "&Open";
00448         //
00449         //toolStripSeparator
00450         //
00451         this.toolStripSeparator.Name = "toolStripSeparator";
00452         this.toolStripSeparator.Size = new System.Drawing.Size(148, 6);
00453         //
00454         //saveToolStripMenuItem
00455         //
00456         this.saveToolStripMenuItem.Image = ((System.Drawing.Image)(
    resources.GetObject("saveToolStripMenuItem.Image")));
00457         this.saveToolStripMenuItem.ImageTransparentColor = System.Drawing.
    Color.Magenta;

```

```
00458         this.saveToolStripMenuItem.Name = "saveToolStripMenuItem";
00459         this.saveToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00460             Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.S)));
00460         this.saveToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00461         this.saveToolStripMenuItem.Text = "&Save";
00462         //
00463         // saveAsToolStripMenuItem
00464         //
00465         this.saveAsToolStripMenuItem.Name = "saveAsToolStripMenuItem";
00466         this.saveAsToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00467         ;
00468         this.saveAsToolStripMenuItem.Text = "Save &As";
00469         //
00470         // toolStripSeparator1
00471         //
00472         this.toolStripMenuItem1.Name = "toolStripSeparator1";
00472         this.toolStripMenuItem1.Size = new System.Drawing.Size(148, 6);
00473         //
00474         // printToolStripMenuItem
00475         //
00476         this.printToolStripMenuItem.Image = ((System.Drawing.Image) (
00477             resources.GetObject("printToolStripMenuItem.Image")));
00477         this.printToolStripMenuItem.ImageTransparentColor = System.Drawing.
00478             Color.Magenta;
00478         this.printToolStripMenuItem.Name = "printToolStripMenuItem";
00479         this.printToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00480             Keys)((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.P)));
00480         this.printToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00481         ;
00482         this.printToolStripMenuItem.Text = "&Print";
00483         //
00484         // printPreviewToolStripMenuItem
00485         //
00485         this.printPreviewToolStripMenuItem.Image = ((System.Drawing.Image) (
00486             resources.GetObject("printPreviewToolStripMenuItem.Image")));
00486         this.printPreviewToolStripMenuItem.ImageTransparentColor = System.
00487             Drawing.Color.Magenta;
00487         this.printPreviewToolStripMenuItem.Name = "
00488         printPreviewToolStripMenuItem";
00488         this.printPreviewToolStripMenuItem.Size = new System.Drawing.Size(1
00489             51, 22);
00489         this.printPreviewToolStripMenuItem.Text = "Print Pre&view";
00490         //
00491         // toolStripSeparator2
00492         //
00493         this.toolStripMenuItem2.Name = "toolStripSeparator2";
00494         this.toolStripMenuItem2.Size = new System.Drawing.Size(148, 6);
00495         //
00496         // exitToolStripMenuItem
00497         //
00498         this.exitToolStripMenuItem.Name = "exitToolStripMenuItem";
00499         this.exitToolStripMenuItem.Size = new System.Drawing.Size(151, 22);
00500         this.exitToolStripMenuItem.Text = "E&xit";
00501         //
00502         // editToolStripMenuItem
00503         //
00504         this.editToolStripMenuItem.DropDownItems.AddRange(new System.
00504             Windows.Forms.ToolStripItem[] {
00505             this.undoToolStripMenuItem,
00506             this.redoToolStripMenuItem,
00507             this.toolStripMenuItem3,
00508             this.cutToolStripMenuItem,
00509             this.copyToolStripMenuItem,
00510             this.pasteToolStripMenuItem,
00511             this.toolStripSeparator4,
00512             this.selectAllToolStripMenuItem});
00513         this.editToolStripMenuItem.Name = "editToolStripMenuItem";
00514         this.editToolStripMenuItem.Size = new System.Drawing.Size(37, 21);
00515         this.editToolStripMenuItem.Text = "&Edit";
00516         //
00517         // undoToolStripMenuItem
00518         //
00519         this.undoToolStripMenuItem.Name = "undoToolStripMenuItem";
00520         this.undoToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
```

```

        Keys) ((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.Z)));
00521      this.undoToolStripMenuItem.Size = new System.Drawing.Size(150, 22);
00522      this.undoToolStripMenuItem.Text = "&Undo";
00523      //
00524      // redoToolStripMenuItem
00525      //
00526      this.redoToolStripMenuItem.Name = "redoToolStripMenuItem";
00527      this.redoToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00528      Keys) ((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.Y)));
00529      this.redoToolStripMenuItem.Size = new System.Drawing.Size(150, 22);
00530      this.redoToolStripMenuItem.Text = "&Redo";
00531      //
00532      // toolStripSeparator3
00533      //
00534      this.toolStripSeparator3.Name = "toolStripSeparator3";
00535      this.toolStripSeparator3.Size = new System.Drawing.Size(147, 6);
00536      //
00537      // cutToolStripMenuItem
00538      //
00539      this.cutToolStripMenuItem.Image = ((System.Drawing.Image)(resources
00540      .GetObject("cutToolStripMenuItem.Image")));
00541      this.cutToolStripMenuItem.ImageTransparentColor = System.Drawing.
00542      Color.Magenta;
00543      this.cutToolStripMenuItem.Name = "cutToolStripMenuItem";
00544      this.cutToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00545      Keys) ((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.X)));
00546      this.cutToolStripMenuItem.Size = new System.Drawing.Size(150, 22);
00547      this.cutToolStripMenuItem.Text = "Cu&t";
00548      //
00549      // copyToolStripMenuItem
00550      //
00551      this.copyToolStripMenuItem.Image = ((System.Drawing.Image)(
00552      resources.GetObject("copyToolStripMenuItem.Image")));
00553      this.copyToolStripMenuItem.ImageTransparentColor = System.Drawing.
00554      Color.Magenta;
00555      this.copyToolStripMenuItem.Name = "copyToolStripMenuItem";
00556      this.copyToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00557      Keys) ((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.C)));
00558      this.copyToolStripMenuItem.Size = new System.Drawing.Size(150, 22);
00559      this.copyToolStripMenuItem.Text = "&Copy";
00560      //
00561      // pasteToolStripMenuItem
00562      //
00563      this.pasteToolStripMenuItem.Image = ((System.Drawing.Image)(
00564      resources.GetObject("pasteToolStripMenuItem.Image")));
00565      this.pasteToolStripMenuItem.ImageTransparentColor = System.Drawing.
00566      Color.Magenta;
00567      this.pasteToolStripMenuItem.Name = "pasteToolStripMenuItem";
00568      this.pasteToolStripMenuItem.ShortcutKeys = ((System.Windows.Forms.
00569      Keys) ((System.Windows.Forms.Keys.Control | System.Windows.Forms.Keys.V)));
00570      this.pasteToolStripMenuItem.Size = new System.Drawing.Size(150, 22)
00571      ;
00572      this.pasteToolStripMenuItem.Text = "&Paste";
00573      //
00574      // toolStripSeparator4
00575      //
00576      this.toolStripSeparator4.Name = "toolStripSeparator4";
00577      this.toolStripSeparator4.Size = new System.Drawing.Size(147, 6);
00578      //
00579      // selectAllToolStripMenuItem
00580      //
00581      this.selectAllToolStripMenuItem.Name = "selectAllToolStripMenuItem"
00582      ;
00583      this.selectAllToolStripMenuItem.Size = new System.Drawing.Size(150,
00584      22);
00585      this.selectAllToolStripMenuItem.Text = "Select &All";
00586      //
00587      // toolsToolStripMenuItem
00588      //
00589      this.toolsToolStripMenuItem.DropDownItems.AddRange(new System.
00590      Windows.Forms.ToolStripItem[] {
00591          this.customizeToolStripMenuItem,
00592          this.optionsToolStripMenuItem});
00593      this.toolsToolStripMenuItem.Name = "toolsToolStripMenuItem";

```

```
00580     this.toolsToolStripMenuItem.Size = new System.Drawing.Size(44, 21);
00581     this.toolsToolStripMenuItem.Text = "&Tools";
00582     // 
00583     // customizeToolStripMenuItem
00584     // 
00585     this.customizeToolStripMenuItem.Name = "customizeToolStripMenuItem"
00586     ;
00587     this.customizeToolStripMenuItem.Size = new System.Drawing.Size(134,
00588     22);
00589     this.customizeToolStripMenuItem.Text = "&Customize";
00590     // 
00591     // optionsToolStripMenuItem
00592     // 
00593     this.optionsToolStripMenuItem.Name = "optionsToolStripMenuItem";
00594     this.optionsToolStripMenuItem.Size = new System.Drawing.Size(134, 2
00595     2);
00596     this.optionsToolStripMenuItem.Text = "&Options";
00597     // 
00598     // helpToolStripMenuItem1
00599     // 
00600     this.helpToolStripMenuItem1.DropDownItems.AddRange(new System.
00601     Windows.Forms.ToolStripItem[] {
00602         this.contentsToolStripMenuItem,
00603         this.indexToolStripMenuItem,
00604         this.searchToolStripMenuItem,
00605         this.toolStripSeparator5,
00606         this.aboutToolStripMenuItem1});
00607     this.helpToolStripMenuItem1.Name = "helpToolStripMenuItem1";
00608     this.helpToolStripMenuItem1.Size = new System.Drawing.Size(40, 21);
00609     this.helpToolStripMenuItem1.Text = "&Help";
00610     // 
00611     // contentsToolStripMenuItem
00612     // 
00613     this.contentsToolStripMenuItem.Name = "contentsToolStripMenuItem";
00614     this.contentsToolStripMenuItem.Size = new System.Drawing.Size(129,
00615     22);
00616     ;
00617     this.contentsToolStripMenuItem.Text = "&Contents";
00618     // 
00619     // indexToolStripMenuItem
00620     // 
00621     this.indexToolStripMenuItem.Name = "indexToolStripMenuItem";
00622     this.indexToolStripMenuItem.Size = new System.Drawing.Size(129, 22
00623     );
00624     this.indexToolStripMenuItem.Text = "&Index";
00625     // 
00626     // searchToolStripMenuItem
00627     // 
00628     this.searchToolStripMenuItem.Name = "searchToolStripMenuItem";
00629     this.searchToolStripMenuItem.Size = new System.Drawing.Size(129, 22
00630     );
00631     this.searchToolStripMenuItem.Text = "&Search";
00632     // 
00633     // toolStripSeparator5
00634     // 
00635     this.toolStripSeparator5.Name = "toolStripSeparator5";
00636     this.toolStripSeparator5.Size = new System.Drawing.Size(126, 6);
00637     // 
00638     this.aboutToolStripMenuItem1.Name = "aboutToolStripMenuItem1";
00639     this.aboutToolStripMenuItem1.Size = new System.Drawing.Size(129, 22
00640     );
00641     this.aboutToolStripMenuItem1.Text = "&About...";
00642     // 
00643     // textBox1
00644     // 
00645     this.textBox1.BorderStyle = System.Windows.Forms.BorderStyle.
00646     FixedSingle;
00647     this.textBox1.Location = new System.Drawing.Point(3, 5);
00648     this.textBox1.Multiline = true;
00649     this.textBox1.Name = "textBox1";
00650     this.textBox1.Size = new System.Drawing.Size(301, 52);
00651     this.textBox1.TabIndex = 1;
00652     this.textBox1.Text = "Enter Message here..";
```

```

00645      //  

00646      // timer  

00647      //  

00648      this.timer.Interval = 25;  

00649      this.timer.Tick += new System.EventHandler(this.timer_Tick);  

00650      //  

00651      // trayICON  

00652      //  

00653      this.trayICON.BalloonTipText = "Raising Hands is a thing of the  

     past.";  

00654      this.trayICON.BalloonTipTitle = "Classroom Inq.";  

00655      this.trayICON.Icon = ((System.Drawing.Icon)(resources.GetObject("  

     trayICON.Icon")));  

00656      this.trayICON.Text = "Dbl Click to Minimize!";  

00657      this.trayICON.Visible = true;  

00658      this.trayICON.MouseDoubleClick += new System.Windows.Forms.  

     MouseEventHandler(this.trayICON_MouseDoubleClick);  

00659      //  

00660      // grpbxFeed  

00661      //  

00662      this.grpbxFeed.AutoScroll = true;  

00663      this.grpbxFeed.BackColor = System.Drawing.Color.FromArgb(((int)((  

     byte)(64))), ((int)((byte)(64))), ((int)((byte)(64))));  

00664      this.grpbxFeed.Location = new System.Drawing.Point(15, 28);  

00665      this.grpbxFeed.Name = "grpbxFeed";  

00666      this.grpbxFeed.Size = new System.Drawing.Size(355, 517);  

00667      this.grpbxFeed.TabIndex = 5;  

00668      this.grpbxFeed.MouseDown += new System.Windows.Forms.  

     MouseEventHandler(this.grpbxFeed_MouseDown);  

00669      //  

00670      // timer_SerialRead  

00671      //  

00672      this.timer_SerialRead.Enabled = true;  

00673      this.timer_SerialRead.Interval = 1000;  

00674      this.timer_SerialRead.Tick += new System.EventHandler(this.  

     timer_SerialRead_Tick);  

00675      //  

00676      // PanelPrefs  

00677      //  

00678      this.PanelPrefs.AutoScroll = true;  

00679      this.PanelPrefs.BackColor = System.Drawing.SystemColors.ControlText  

;  

00680      this.PanelPrefs.Controls.Add(this.chkbxSplash);  

00681      this.PanelPrefs.Controls.Add(this.lblNotifyTime);  

00682      this.PanelPrefs.Controls.Add(this.numUpDnNotify);  

00683      this.PanelPrefs.Controls.Add(this.lblSerPt);  

00684      this.PanelPrefs.Controls.Add(this.serialCOMcmbbx);  

00685      this.PanelPrefs.Controls.Add(this.lblAnimationPrefs);  

00686      this.PanelPrefs.Controls.Add(this.cmbbxAnimationSpeed);  

00687      this.PanelPrefs.Controls.Add(this.grpbxTheme);  

00688      this.PanelPrefs.Controls.Add(this.btnQuizMaker);  

00689      this.PanelPrefs.Controls.Add(this.chkbxCtrlHide);  

00690      this.PanelPrefs.Controls.Add(this.chkbxNotify);  

00691      this.PanelPrefs.Controls.Add(this.chkbxToolips);  

00692      this.PanelPrefs.Controls.Add(this.grpbxUnread);  

00693      this.PanelPrefs.Controls.Add(this.chkbxLameMode);  

00694      this.PanelPrefs.Controls.Add(this.btnStuMgmt_Prefs);  

00695      this.PanelPrefs.Controls.Add(this.chkbxTXSound);  

00696      this.PanelPrefs.Controls.Add(this.chkbxRXSound);  

00697      this.PanelPrefs.Controls.Add(this.btnPrefs_Cls);  

00698      this.PanelPrefs.Controls.Add(this.picbxPrefs);  

00699      this.PanelPrefs.ForeColor = System.Drawing.Color.DarkGoldenrod;  

00700      this.PanelPrefs.Location = new System.Drawing.Point(395, 28);  

00701      this.PanelPrefs.Name = "PanelPrefs";  

00702      this.PanelPrefs.Size = new System.Drawing.Size(355, 517);  

00703      this.PanelPrefs.TabIndex = 6;  

00704      //  

00705      // lblNotifyTime  

00706      //  

00707      this.lblNotifyTime.Font = new System.Drawing.Font("Microsoft Sans  

     Serif", 8.25F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.  

     Point, ((byte)(0)));  

00708      this.lblNotifyTime.Location = new System.Drawing.Point(169, 225);  

00709      this.lblNotifyTime.Name = "lblNotifyTime";

```

```
00710         this.lblNotifyTime.Size = new System.Drawing.Size(74, 33);
00711         this.lblNotifyTime.TabIndex = 19;
00712         this.lblNotifyTime.Text = "Notify panel\r\nnTimeout-->";
00713         this.lblNotifyTime.TextAlign = System.Drawing.ContentAlignment.
00714             MiddleCenter;
00715         // 
00716         // numUpDnNotify
00717         this.numUpDnNotify.BackColor = System.Drawing.Color.White;
00718         this.numUpDnNotify.BorderStyle = System.Windows.Forms.BorderStyle.
00719             FixedSingle;
00720         this.numUpDnNotify.DecimalPlaces = 1;
00721         this.numUpDnNotify.Increment = new decimal(new int[] {
00722             2,
00723             0,
00724             0,
00725             65536});
00726         this.numUpDnNotify.Location = new System.Drawing.Point(252, 232);
00727         this.numUpDnNotify.Maximum = new decimal(new int[] {
00728             50,
00729             0,
00730             0,
00731             65536});
00732         this.numUpDnNotify.Minimum = new decimal(new int[] {
00733             1,
00734             0,
00735             0});
00736         this.numUpDnNotify.Name = "numUpDnNotify";
00737         this.numUpDnNotify.ReadOnly = true;
00738         this.numUpDnNotify.Size = new System.Drawing.Size(49, 20);
00739         this.numUpDnNotify.TabIndex = 18;
00740         this.numUpDnNotify.TextAlign = System.Windows.Forms.
00741             HorizontalAlignment.Center;
00742         this.numUpDnNotify.Value = new decimal(new int[] {
00743             15,
00744             0,
00745             0,
00746             65536});
00747         // 
00748         // lblSerPt
00749         // 
00750         this.lblSerPt.Font = new System.Drawing.Font("Microsoft Sans Serif",
00751             , 8.25F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
00752         this.lblSerPt.Location = new System.Drawing.Point(169, 192);
00753         this.lblSerPt.Name = "lblSerPt";
00754         this.lblSerPt.Size = new System.Drawing.Size(74, 33);
00755         this.lblSerPt.TabIndex = 17;
00756         this.lblSerPt.Text = "Serial Port\r\nnSelect-->";
00757         this.lblSerPt.TextAlign = System.Drawing.ContentAlignment.
00758             MiddleCenter;
00759         // 
00760         // serialCOMcmbbx
00761         // 
00762         this.serialCOMcmbbx.FlatStyle = System.Windows.Forms.FlatStyle.Flat
00763         ;
00764         this.serialCOMcmbbx.FormattingEnabled = true;
00765         this.serialCOMcmbbx.Location = new System.Drawing.Point(252, 199);
00766         this.serialCOMcmbbx.Name = "serialCOMcmbbx";
00767         this.serialCOMcmbbx.Size = new System.Drawing.Size(64, 21);
00768         this.serialCOMcmbbx.TabIndex = 16;
00769         this.serialCOMcmbbx.Text = " Port?";
00770         this.serialCOMcmbbx.SelectedIndexChanged += new System.EventHandler
00771             (this.serialCOMcmbbx_SelectedIndexChanged);
00772         this.serialCOMcmbbx.Click += new System.EventHandler(this.
00773             serialCOMcmbbx_Click);
00774         // 
00775         // lblAnimationPrefs
00776         // 
00777         this.lblAnimationPrefs.Font = new System.Drawing.Font("Microsoft
00778             Sans Serif", 8.25F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.
00779             Point, ((byte)(0)));
00780         this.lblAnimationPrefs.Location = new System.Drawing.Point(169, 159
```

```

);
00773     this.lblAnimationPrefs.Name = "lblAnimationPrefs";
00774     this.lblAnimationPrefs.Size = new System.Drawing.Size(74, 33);
00775     this.lblAnimationPrefs.TabIndex = 15;
00776     this.lblAnimationPrefs.Text = "Animation\r\nSpeed -->";
00777     this.lblAnimationPrefs.TextAlign = System.Drawing.ContentAlignment.
00778     MiddleCenter;
00779     // 
00780     // cmbbxAnimationSpeed
00781     //
00782     this.cmbbxAnimationSpeed.FlatStyle = System.Windows.Forms.FlatStyle
00783     .Flat;
00784     this.cmbbxAnimationSpeed.FormattingEnabled = true;
00785     this.cmbbxAnimationSpeed.Items.AddRange(new object[] {
00786         "Slow",
00787         "Normal",
00788         "Fast",
00789         "Blazing"});
00790     this.cmbbxAnimationSpeed.Location = new System.Drawing.Point(252, 1
00791     65);
00792     this.cmbbxAnimationSpeed.Name = "cmbbxAnimationSpeed";
00793     this.cmbbxAnimationSpeed.Size = new System.Drawing.Size(71, 21);
00794     this.cmbbxAnimationSpeed.TabIndex = 14;
00795     this.cmbbxAnimationSpeed.Text = "Speed";
00796     this.cmbbxAnimationSpeed.SelectedIndexChanged += new System.
00797     EventHandler(this.cmbbxAnimationSpeed_SelectedIndexChanged);
00798     // 
00799     // grpbxTheme
00800     //
00801     this.grpbxTheme.Controls.Add(this.btnDefaultTheme);
00802     this.grpbxTheme.Controls.Add(this.btnForeColor);
00803     this.grpbxTheme.Controls.Add(this.btnBkgrndColor);
00804     this.grpbxTheme.Location = new System.Drawing.Point(17, 396);
00805     this.grpbxTheme.Name = "grpbxTheme";
00806     this.grpbxTheme.Size = new System.Drawing.Size(236, 85);
00807     this.grpbxTheme.TabIndex = 13;
00808     this.grpbxTheme.TabStop = false;
00809     this.grpbxTheme.Text = "Application Theme";
00810     // 
00811     // btnDefaultTheme
00812     //
00813     this.btnDefaultTheme.BackColor = System.Drawing.Color.Black;
00814     this.btnDefaultTheme.Location = new System.Drawing.Point(64, 56);
00815     this.btnDefaultTheme.Name = "btnDefaultTheme";
00816     this.btnDefaultTheme.Size = new System.Drawing.Size(98, 23);
00817     this.btnDefaultTheme.TabIndex = 13;
00818     this.btnDefaultTheme.Text = "Reset Default";
00819     this.btnDefaultTheme.UseVisualStyleBackColor = false;
00820     this.btnDefaultTheme.Click += new System.EventHandler(this.
00821     btnDefaultTheme_Click);
00822     // 
00823     // btnForeColor
00824     //
00825     this.btnForeColor.BackColor = System.Drawing.Color.Black;
00826     this.btnForeColor.Location = new System.Drawing.Point(15, 23);
00827     this.btnForeColor.Name = "btnForeColor";
00828     this.btnForeColor.Size = new System.Drawing.Size(98, 23);
00829     this.btnForeColor.TabIndex = 12;
00830     this.btnForeColor.Text = "Pick ForeColor";
00831     this.btnForeColor.UseVisualStyleBackColor = false;
00832     this.btnForeColor.Click += new System.EventHandler(this.
00833     btnForeColor_Click);
00834     // 
00835     // btnBkgrndColor
00836     //
00837     this.btnBkgrndColor.BackColor = System.Drawing.Color.Black;
00838     this.btnBkgrndColor.Location = new System.Drawing.Point(124, 23);
00839     this.btnBkgrndColor.Name = "btnBkgrndColor";
00840     this.btnBkgrndColor.Size = new System.Drawing.Size(98, 23);
00841     this.btnBkgrndColor.TabIndex = 11;
00842     this.btnBkgrndColor.Text = "Pick BackColor";
00843     this.btnBkgrndColor.UseVisualStyleBackColor = false;
00844     this.btnBkgrndColor.Click += new System.EventHandler(this.
00845     btnBkgrndColor_Click);

```

```
00839      //  
00840      // btnQuizMaker  
00841      //  
00842      this.btnQuizMaker.AccessibleRole = System.Windows.Forms.  
        AccessibleRole.None;  
00843      this.btnQuizMaker.BackColor = System.Drawing.Color.Black;  
00844      this.btnQuizMaker.Location = new System.Drawing.Point(20, 367);  
00845      this.btnQuizMaker.Name = "btnQuizMaker";  
00846      this.btnQuizMaker.Size = new System.Drawing.Size(103, 23);  
00847      this.btnQuizMaker.TabIndex = 10;  
00848      this.btnQuizMaker.Text = "Quiz Maker";  
00849      this.btnQuizMaker.UseVisualStyleBackColor = false;  
00850      this.btnQuizMaker.Click += new System.EventHandler(this.  
        btnQuizMaker_Click);  
00851      //  
00852      // chkbxCtrlHide  
00853      //  
00854      this.chkbxCtrlHide.FlatStyle = System.Windows.Forms.FlatStyle.Flat;  
00855      this.chkbxCtrlHide.Location = new System.Drawing.Point(18, 276);  
00856      this.chkbxCtrlHide.Name = "chkbxCtrlHide";  
00857      this.chkbxCtrlHide.Size = new System.Drawing.Size(130, 33);  
00858      this.chkbxCtrlHide.TabIndex = 9;  
00859      this.chkbxCtrlHide.Text = "Hide '\x' in title bar";  
00860      this.chkbxCtrlHide.TextAlign = System.Drawing.ContentAlignment.  
        MiddleCenter;  
00861      this.chkbxCtrlHide.UseVisualStyleBackColor = true;  
00862      this.chkbxCtrlHide.CheckedChanged += new System.EventHandler(this.  
        chkbxCtrlHide_CheckedChanged);  
00863      //  
00864      // chkbxNotify  
00865      //  
00866      this.chkbxNotify.Checked = true;  
00867      this.chkbxNotify.CheckState = System.Windows.Forms.CheckState.  
        Checked;  
00868      this.chkbxNotify.FlatStyle = System.Windows.Forms.FlatStyle.Flat;  
00869      this.chkbxNotify.Location = new System.Drawing.Point(18, 237);  
00870      this.chkbxNotify.Name = "chkbxNotify";  
00871      this.chkbxNotify.Size = new System.Drawing.Size(130, 33);  
00872      this.chkbxNotify.TabIndex = 8;  
00873      this.chkbxNotify.Text = "Notifications in Tray";  
00874      this.chkbxNotify.TextAlign = System.Drawing.ContentAlignment.  
        MiddleCenter;  
00875      this.chkbxNotify.UseVisualStyleBackColor = true;  
00876      //  
00877      // chkbxToolips  
00878      //  
00879      this.chkbxToolips.FlatStyle = System.Windows.Forms.FlatStyle.Flat;  
00880      this.chkbxToolips.Location = new System.Drawing.Point(18, 198);  
00881      this.chkbxToolips.Name = "chkbxToolips";  
00882      this.chkbxToolips.Size = new System.Drawing.Size(130, 33);  
00883      this.chkbxToolips.TabIndex = 7;  
00884      this.chkbxToolips.Text = "Toolips on Questions";  
00885      this.chkbxToolips.TextAlign = System.Drawing.ContentAlignment.  
        MiddleCenter;  
00886      this.chkbxToolips.UseVisualStyleBackColor = true;  
00887      //  
00888      // grpbxUnread  
00889      //  
00890      this.grpbxUnread.Controls.Add(this.rdbtnClick);  
00891      this.grpbxUnread.Controls.Add(this.rdbtnHover);  
00892      this.grpbxUnread.Location = new System.Drawing.Point(165, 81);  
00893      this.grpbxUnread.Name = "grpbxUnread";  
00894      this.grpbxUnread.Size = new System.Drawing.Size(158, 72);  
00895      this.grpbxUnread.TabIndex = 6;  
00896      this.grpbxUnread.TabStop = false;  
00897      this.grpbxUnread.Text = "Mark as Read by:";  
00898      //  
00899      // rdbtnClick  
00900      //  
00901      this.rdbtnClick.AutoSize = true;  
00902      this.rdbtnClick.FlatStyle = System.Windows.Forms.FlatStyle.Flat;  
00903      this.rdbtnClick.Location = new System.Drawing.Point(31, 42);  
00904      this.rdbtnClick.Name = "rdbtnClick";  
00905      this.rdbtnClick.Size = new System.Drawing.Size(82, 17);
```

```

00906     this.rdbtnClick.TabIndex = 1;
00907     this.rdbtnClick.TabStop = true;
00908     this.rdbtnClick.Text = "Mouse Click";
00909     this.rdbtnClick.UseVisualStyleBackColor = true;
00910     //
00911     // rdbtnHover
00912     //
00913     this.rdbtnHover.AutoSize = true;
00914     this.rdbtnHover.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
00915     this.rdbtnHover.Location = new System.Drawing.Point(31, 19);
00916     this.rdbtnHover.Name = "rdbtnHover";
00917     this.rdbtnHover.Size = new System.Drawing.Size(88, 17);
00918     this.rdbtnHover.TabIndex = 0;
00919     this.rdbtnHover.TabStop = true;
00920     this.rdbtnHover.Text = "Mouse Hover";
00921     this.rdbtnHover.UseVisualStyleBackColor = true;
00922     //
00923     // chkbxLameMode
00924     //
00925     this.chkbxLameMode.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
00926     this.chkbxLameMode.Location = new System.Drawing.Point(18, 159);
00927     this.chkbxLameMode.Name = "chkbxLameMode";
00928     this.chkbxLameMode.Size = new System.Drawing.Size(112, 33);
00929     this.chkbxLameMode.TabIndex = 5;
00930     this.chkbxLameMode.Text = "I\\'m Lame\r\n(No Animations)";
00931     this.chkbxLameMode.UseVisualStyleBackColor = true;
00932     //
00933     // btnStuMgmt_Prefs
00934     //
00935     this.btnStuMgmt_Prefs.AccessibleRole = System.Windows.Forms.
00936     AccessibleRole.None;
00937     this.btnStuMgmt_Prefs.BackColor = System.Drawing.Color.Black;
00938     this.btnStuMgmt_Prefs.Location = new System.Drawing.Point(138, 367)
00939     ;
00940     this.btnStuMgmt_Prefs.Name = "btnStuMgmt_Prefs";
00941     this.btnStuMgmt_Prefs.Size = new System.Drawing.Size(103, 23);
00942     this.btnStuMgmt_Prefs.TabIndex = 4;
00943     this.btnStuMgmt_Prefs.Text = "Manage Students";
00944     this.btnStuMgmt_Prefs.UseVisualStyleBackColor = false;
00945     this.btnStuMgmt_Prefs.Click += new System.EventHandler(this.
00946     btnStuMgmt_Prefs_Click);
00947     //
00948     // chkbxTXSound
00949     //
00950     this.chkbxTXSound.Checked = true;
00951     this.chkbxTXSound.CheckState = System.Windows.Forms.CheckState.
00952     Checked;
00953     this.chkbxTXSound.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
00954     this.chkbxTXSound.Location = new System.Drawing.Point(18, 120);
00955     this.chkbxTXSound.Name = "chkbxTXSound";
00956     this.chkbxTXSound.Size = new System.Drawing.Size(112, 33);
00957     this.chkbxTXSound.TabIndex = 3;
00958     this.chkbxTXSound.Text = "Play Sound on\r\nmessage sent";
00959     this.chkbxTXSound.UseVisualStyleBackColor = true;
00960     //
00961     // chkbxRXSound
00962     //
00963     this.chkbxRXSound.Checked = true;
00964     this.chkbxRXSound.CheckState = System.Windows.Forms.CheckState.
00965     Checked;
00966     this.chkbxRXSound.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
00967     this.chkbxRXSound.Location = new System.Drawing.Point(18, 81);
00968     this.chkbxRXSound.Name = "chkbxRXSound";
00969     this.chkbxRXSound.Size = new System.Drawing.Size(112, 33);
00970     this.chkbxRXSound.TabIndex = 2;
00971     this.chkbxRXSound.Text = "Play Sound on\r\nmessage received";
00972     this.chkbxRXSound.UseVisualStyleBackColor = true;
00973     //
00974     // btnPrefs_Cls
00975     //
00976     this.btnPrefs_Cls.BackColor = System.Drawing.Color.Black;
00977     this.btnPrefs_Cls.Location = new System.Drawing.Point(267, 452);
00978     this.btnPrefs_Cls.Name = "btnPrefs_Cls";
00979     this.btnPrefs_Cls.Size = new System.Drawing.Size(75, 23);

```

```
00975         this.btnAdd_Prefs.TabIndex = 1;
00976         this.btnAdd_Prefs.Text = "Close";
00977         this.btnAdd_Prefs.UseVisualStyleBackColor = false;
00978         this.btnAdd_Prefs.Click += new System.EventHandler(this.
00979             btnAdd_Prefs_Click);
00979         //
00980         // picbxPrefs
00981         //
00982         this.picbxPrefs.Image = ((System.Drawing.Image)(resources.GetObject
00983             ("picbxPrefs.Image")));
00983         this.picbxPrefs.Location = new System.Drawing.Point(16, 17);
00984         this.picbxPrefs.Name = "picbxPrefs";
00985         this.picbxPrefs.Size = new System.Drawing.Size(321, 50);
00986         this.picbxPrefs.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
00987             StretchImage;
00987         this.picbxPrefs.TabIndex = 0;
00988         this.picbxPrefs.TabStop = false;
00989         //
00990         // DirectMsgPanel
00991         //
00992         this.DirectMsgPanel.AutoScroll = true;
00993         this.DirectMsgPanel.BackColor = System.Drawing.SystemColors.
00993             ControlText;
00994         this.DirectMsgPanel.Controls.Add(this.pictureBox1);
00995         this.DirectMsgPanel.Controls.Add(this.btnAdd_DM);
00996         this.DirectMsgPanel.Controls.Add(this.btnAdd_Cls);
00997         this.DirectMsgPanel.Controls.Add(this.btnAdd_Clr);
00998         this.DirectMsgPanel.Controls.Add(this.cmbxDM);
00999         this.DirectMsgPanel.Controls.Add(this.txtbxDM);
01000         this.DirectMsgPanel.Location = new System.Drawing.Point(-395, 28);
01001         this.DirectMsgPanel.Name = "DirectMsgPanel";
01002         this.DirectMsgPanel.Size = new System.Drawing.Size(355, 517);
01003         this.DirectMsgPanel.TabIndex = 6;
01004         //
01005         // pictureBox1
01006         //
01007         this.pictureBox1.Image = ((System.Drawing.Image)(resources.
01007             GetObject("pictureBox1.Image")));
01008         this.pictureBox1.Location = new System.Drawing.Point(16, 17);
01009         this.pictureBox1.Name = "pictureBox1";
01010         this.pictureBox1.Size = new System.Drawing.Size(324, 73);
01011         this.pictureBox1.SizeMode = System.Windows.Forms.PictureBoxSizeMode
01011             .StretchImage;
01012         this.pictureBox1.TabIndex = 5;
01013         this.pictureBox1.TabStop = false;
01014         //
01015         // btnDM_Send
01016         //
01017         this.btnAdd_DM.ForeColor = System.Drawing.Color.DarkGoldenrod;
01018         this.btnAdd_DM.Location = new System.Drawing.Point(260, 262);
01019         this.btnAdd_DM.Name = "btnDM_Send";
01020         this.btnAdd_DM.Size = new System.Drawing.Size(85, 37);
01021         this.btnAdd_DM.TabIndex = 1;
01022         this.btnAdd_DM.Text = "Send";
01023         this.btnAdd_DM.UseVisualStyleBackColor = true;
01024         this.btnAdd_DM.Click += new System.EventHandler(this.
01024             btnAdd_DM_Click);
01025         //
01026         // btnDM_Cls
01027         //
01028         this.btnAdd_Cls.ForeColor = System.Drawing.Color.DarkGoldenrod;
01029         this.btnAdd_Cls.Location = new System.Drawing.Point(135, 262);
01030         this.btnAdd_Cls.Name = "btnDM_Cls";
01031         this.btnAdd_Cls.Size = new System.Drawing.Size(85, 37);
01032         this.btnAdd_Cls.TabIndex = 3;
01033         this.btnAdd_Cls.Text = "Close";
01034         this.btnAdd_Cls.UseVisualStyleBackColor = true;
01035         this.btnAdd_Cls.Click += new System.EventHandler(this.
01035             btnAdd_Cls_Click);
01036         //
01037         // btnDM_Clr
01038         //
01039         this.btnAdd_Clr.ForeColor = System.Drawing.Color.DarkGoldenrod;
01040         this.btnAdd_Clr.Location = new System.Drawing.Point(10, 262);
```

```

01041         this.btnAdd.Name = "btnDM_Clr";
01042         this.btnAdd.Size = new System.Drawing.Size(85, 37);
01043         this.btnAdd.TabIndex = 2;
01044         this.btnAdd.Text = "Clear";
01045         this.btnAdd.UseVisualStyleBackColor = true;
01046         this.btnAdd.Click += new System.EventHandler(this.
01047             btnDM_Clr_Click);
01047         //
01048         // cmbxDM
01049         //
01050         this.cmbxDM.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01051         this.cmbxDM.FormattingEnabled = true;
01052         this.cmbxDM.Location = new System.Drawing.Point(96, 119);
01053         this.cmbxDM.Name = "cmbxDM";
01054         this.cmbxDM.Size = new System.Drawing.Size(163, 21);
01055         this.cmbxDM.TabIndex = 4;
01056         this.cmbxDM.Text = "Select Student";
01057         this.cmbxDM.SelectedIndexChanged += new System.EventHandler(this.
01058             cmbxDM_SelectedIndexChanged);
01059         this.cmbxDM.Click += new System.EventHandler(this.cmbxDM_Click);
01060         //
01061         // txtbxDM
01062         //
01063         this.txtbxDM.AutoCompleteMode = System.Windows.Forms.
01064             AutoCompleteMode.Suggest;
01063         this.txtbxDM.AutoCompleteSource = System.Windows.Forms.
01064             AutoCompleteSource.AllSystemSources;
01064         this.txtbxDM.BorderStyle = System.Windows.Forms.BorderStyle.
01064             FixedSingle;
01065         this.txtbxDM.Location = new System.Drawing.Point(10, 160);
01066         this.txtbxDM.Multiline = true;
01067         this.txtbxDM.Name = "txtbxDM";
01068         this.txtbxDM.Size = new System.Drawing.Size(335, 71);
01069         this.txtbxDM.TabIndex = 0;
01070         //
01071         // PanelFAQ
01072         //
01073         this.PanelFAQ.AutoScroll = true;
01074         this.PanelFAQ.BackColor = System.Drawing.SystemColors.ControlText;
01075         this.PanelFAQ.Controls.Add(this.lnklblIssues);
01076         this.PanelFAQ.Controls.Add(this.lblFAQ);
01077         this.PanelFAQ.Controls.Add(this.txtbxFAQ);
01078         this.PanelFAQ.Controls.Add(this.btnAddcls);
01079         this.PanelFAQ.Controls.Add(this.pictureBox2);
01080         this.PanelFAQ.ForeColor = System.Drawing.Color.DarkGoldenrod;
01081         this.PanelFAQ.Location = new System.Drawing.Point(12, -495);
01082         this.PanelFAQ.Name = "PanelFAQ";
01083         this.PanelFAQ.Size = new System.Drawing.Size(361, 522);
01084         this.PanelFAQ.TabIndex = 7;
01085         //
01086         // lnklblIssues
01087         //
01088         this.lnklblIssues.AutoSize = true;
01089         this.lnklblIssues.LinkColor = System.Drawing.Color.Gold;
01090         this.lnklblIssues.Location = new System.Drawing.Point(125, 486);
01091         this.lnklblIssues.Name = "lnklblIssues";
01092         this.lnklblIssues.Size = new System.Drawing.Size(85, 13);
01093         this.lnklblIssues.TabIndex = 5;
01094         this.lnklblIssues.TabStop = true;
01095         this.lnklblIssues.Text = "GitHub - Issues";
01096         this.lnklblIssues.VisitedLinkColor = System.Drawing.Color.Chocolate
01097 ;
01097         this.lnklblIssues.LinkClicked += new System.Windows.Forms.
01098             LinkLabelLinkClickedEventHandler(this.lnklblIssues_LinkClicked);
01098         //
01099         // lblFAQ
01100         //
01101         this.lblFAQ.AutoSize = true;
01102         this.lblFAQ.Location = new System.Drawing.Point(8, 486);
01103         this.lblFAQ.Name = "lblFAQ";
01104         this.lblFAQ.Size = new System.Drawing.Size(111, 13);
01105         this.lblFAQ.TabIndex = 4;
01106         this.lblFAQ.Text = "Contact Developer at:";
01107         //

```

```
01108         // txtbxFAQ
01109         //
01110         this.txtbxFAQ.BackColor = System.Drawing.Color.Black;
01111         this.txtbxFAQ.BorderStyle = System.Windows.Forms.BorderStyle.None;
01112         this.txtbxFAQ.Cursor = System.Windows.Forms.Cursors.Default;
01113         this.txtbxFAQ.ForeColor = System.Drawing.Color.DarkGoldenrod;
01114         this.txtbxFAQ.Location = new System.Drawing.Point(19, 83);
01115         this.txtbxFAQ.Multiline = true;
01116         this.txtbxFAQ.Name = "txtbxFAQ";
01117         this.txtbxFAQ.ReadOnly = true;
01118         this.txtbxFAQ.ScrollBars = System.Windows.Forms.ScrollBars.Vertical
01119         ;
01120         this.txtbxFAQ.Size = new System.Drawing.Size(332, 392);
01121         this.txtbxFAQ.TabIndex = 2;
01122         this.txtbxFAQ.Text = resources.GetString("txtbxFAQ.Text");
01123         //
01124         // btnFAQcls
01125         //
01126         this.btnFAQcls.BackColor = System.Drawing.Color.Black;
01127         this.btnFAQcls.Location = new System.Drawing.Point(276, 481);
01128         this.btnFAQcls.Name = "btnFAQcls";
01129         this.btnFAQcls.Size = new System.Drawing.Size(75, 23);
01130         this.btnFAQcls.TabIndex = 1;
01131         this.btnFAQcls.Text = "Close";
01132         this.btnFAQcls.UseVisualStyleBackColor = false;
01133         this.btnFAQcls.Click += new System.EventHandler(this.
01134             btnFAQcls_Click);
01135         //
01136         // pictureBox2
01137         //
01138         this.pictureBox2.Image = ((System.Drawing.Image)(resources.
01139             GetObject("pictureBox2.Image")));
01140         this.pictureBox2.Location = new System.Drawing.Point(16, 17);
01141         this.pictureBox2.Name = "pictureBox2";
01142         this.pictureBox2.Size = new System.Drawing.Size(321, 50);
01143         this.pictureBox2.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
01144             StretchImage;
01145         this.pictureBox2.TabIndex = 0;
01146         this.pictureBox2.TabStop = false;
01147         //
01148         // PanelStudents
01149         //
01150         this.PanelStudents.AutoScroll = true;
01151         this.PanelStudents.BackColor = System.Drawing.SystemColors.
01152             ControlText;
01153         this.PanelStudents.Controls.Add(this.btnConvView);
01154         this.PanelStudents.Controls.Add(this.grpbxEDITStudents);
01155         this.PanelStudents.Controls.Add(this.btnCloseStudents);
01156         this.PanelStudents.Controls.Add(this.btnCloseEDTStudents);
01157         this.PanelStudents.Controls.Add(this.lstbxStudents);
01158         this.PanelStudents.Controls.Add(this.btnCLSStudents);
01159         this.PanelStudents.Controls.Add(this.pictureBox3);
01160         this.PanelStudents.ForeColor = System.Drawing.Color.DarkGoldenrod;
01161         this.PanelStudents.Location = new System.Drawing.Point(15, 545);
01162         this.PanelStudents.Name = "PanelStudents";
01163         this.PanelStudents.Size = new System.Drawing.Size(361, 499);
01164         this.PanelStudents.TabIndex = 9;
01165         //
01166         // btnConvView
01167         //
01168         this.btnConvView.BackColor = System.Drawing.Color.Black;
01169         this.btnConvView.Location = new System.Drawing.Point(191, 99);
01170         this.btnConvView.Name = "btnConvView";
01171         this.btnConvView.Size = new System.Drawing.Size(127, 23);
01172         this.btnConvView.TabIndex = 6;
01173         this.btnConvView.Text = "Conversation Viewer";
01174         this.btnConvView.UseVisualStyleBackColor = false;
01175         this.btnConvView.Click += new System.EventHandler(this.
01176             btnConvView_Click);
01177         //
01178         // grpbxEDITStudents
01179         //
01180         this.grpbxEDITStudents.Controls.Add(this.btnDoneStudents);
01181         this.grpbxEDITStudents.Controls.Add(this.lblStudentsName);
```

```

01176     this.grpbxEDITStudents.Controls.Add(this.txtbxStudentsName);
01177     this.grpbxEDITStudents.Location = new System.Drawing.Point(150, 178
01178 );
01179     this.grpbxEDITStudents.Name = "grpbxEDITStudents";
01180     this.grpbxEDITStudents.Size = new System.Drawing.Size(200, 142);
01181     this.grpbxEDITStudents.TabIndex = 5;
01182     this.grpbxEDITStudents.TabStop = false;
01183     this.grpbxEDITStudents.Text = "StudentNumber";
01184 
01185 // 
01186 // btnDoneStudents
01187 // 
01188 this.btnDoneStudents.BackColor = System.Drawing.Color.Black;
01189 this.btnDoneStudents.Location = new System.Drawing.Point(63, 99);
01190 this.btnDoneStudents.Name = "btnDoneStudents";
01191 this.btnDoneStudents.Size = new System.Drawing.Size(75, 23);
01192 this.btnDoneStudents.TabIndex = 6;
01193 this.btnDoneStudents.Text = "Done";
01194 this.btnDoneStudents.UseVisualStyleBackColor = false;
01195 this.btnDoneStudents.Click += new System.EventHandler(this.
01196 btnDoneStudents_Click);
01197 // 
01198 // lblStudentsName
01199 // 
01200 this.lblStudentsName.AutoSize = true;
01201 this.lblStudentsName.Location = new System.Drawing.Point(15, 29);
01202 this.lblStudentsName.Name = "lblStudentsName";
01203 this.lblStudentsName.Size = new System.Drawing.Size(82, 13);
01204 this.lblStudentsName.TabIndex = 1;
01205 this.lblStudentsName.Text = "Student's Name";
01206 // 
01207 // txtbxStudentsName
01208 // 
01209 this.txtbxStudentsName.BorderStyle = System.Windows.Forms.
01210 BorderStyle.FixedSingle;
01211 this.txtbxStudentsName.Location = new System.Drawing.Point(19, 61);
01212 this.txtbxStudentsName.Name = "txtbxStudentsName";
01213 this.txtbxStudentsName.Size = new System.Drawing.Size(169, 20);
01214 this.txtbxStudentsName.TabIndex = 0;
01215 this.txtbxStudentsName.Text = "----- Select a Student";
01216 // 
01217 // btnCLRStudents
01218 // 
01219 this.btnCLRStudents.BackColor = System.Drawing.Color.Black;
01220 this.btnCLRStudents.Location = new System.Drawing.Point(265, 139);
01221 this.btnCLRStudents.Name = "btnCLRStudents";
01222 this.btnCLRStudents.Size = new System.Drawing.Size(75, 23);
01223 this.btnCLRStudents.TabIndex = 4;
01224 this.btnCLRStudents.Text = "Clear";
01225 this.btnCLRStudents.UseVisualStyleBackColor = false;
01226 this.btnCLRStudents.Click += new System.EventHandler(this.
01227 btnCLRStudents_Click);
01228 // 
01229 // btnEDTStudents
01230 // 
01231 this.btnEDTStudents.BackColor = System.Drawing.Color.Black;
01232 this.btnEDTStudents.Location = new System.Drawing.Point(168, 139);
01233 this.btnEDTStudents.Name = "btnEDTStudents";
01234 this.btnEDTStudents.Size = new System.Drawing.Size(75, 23);
01235 this.btnEDTStudents.TabIndex = 3;
01236 this.btnEDTStudents.Text = "Edit";
01237 this.btnEDTStudents.UseVisualStyleBackColor = false;
01238 this.btnEDTStudents.Click += new System.EventHandler(this.
01239 btnEDTStudents_Click);
01240 // 
01241 // lstbxStudents
01242 // 
01243 this.lstbxStudents.FormattingEnabled = true;
01244 this.lstbxStudents.Location = new System.Drawing.Point(20, 90);
01245 this.lstbxStudents.Name = "lstbxStudents";
01246 this.lstbxStudents.Size = new System.Drawing.Size(120, 342);
01247 this.lstbxStudents.TabIndex = 2;
01248 // 
01249 // btnCLSStudents
01250 // 

```

```
01245     this.btnCloseStudents.BackColor = System.Drawing.Color.Black;
01246     this.btnCloseStudents.Location = new System.Drawing.Point(276, 462);
01247     this.btnCloseStudents.Name = "btnCLSStudents";
01248     this.btnCloseStudents.Size = new System.Drawing.Size(75, 23);
01249     this.btnCloseStudents.TabIndex = 1;
01250     this.btnCloseStudents.Text = "Close";
01251     this.btnCloseStudents.UseVisualStyleBackColor = false;
01252     this.btnCloseStudents.Click += new System.EventHandler(this.
    btnCLSStudents_Click);
01253     //
01254     // pictureBox3
01255     //
01256     this.pictureBox3.Image = global::SrP_ClassroomInq.Properties.
    Resources.StuMngmntCtr;
01257     this.pictureBox3.Location = new System.Drawing.Point(6, 7);
01258     this.pictureBox3.Name = "pictureBox3";
01259     this.pictureBox3.Size = new System.Drawing.Size(345, 77);
01260     this.pictureBox3.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
    StretchImage;
01261     this.pictureBox3.TabIndex = 0;
01262     this.pictureBox3.TabStop = false;
01263     //
01264     // timer_ControlsCreate
01265     //
01266     this.timer_ControlsCreate.Enabled = true;
01267     this.timer_ControlsCreate.Interval = 1000;
01268     this.timer_ControlsCreate.Tick += new System.EventHandler(this.
    timer_ControlsCreate_Tick);
01269     //
01270     // PanelConvView
01271     //
01272     this.PanelConvView.AutoScroll = true;
01273     this.PanelConvView.BackColor = System.Drawing.SystemColors.
    ControlText;
01274     this.PanelConvView.Controls.Add(this.btnClose_REFRESH);
01275     this.PanelConvView.Controls.Add(this.btnClose_PRINT);
01276     this.PanelConvView.Controls.Add(this.lstbxConvView);
01277     this.PanelConvView.Controls.Add(this.btnClose_CLS_ConvView);
01278     this.PanelConvView.Controls.Add(this.pictureBox4);
01279     this.PanelConvView.ForeColor = System.Drawing.Color.DarkGoldenrod;
01280     this.PanelConvView.Location = new System.Drawing.Point(0, -530);
01281     this.PanelConvView.Name = "PanelConvView";
01282     this.PanelConvView.Size = new System.Drawing.Size(384, 545);
01283     this.PanelConvView.TabIndex = 10;
01284     //
01285     // btnCV_REFRESH
01286     //
01287     this.btnClose_REFRESH.BackColor = System.Drawing.Color.Black;
01288     this.btnClose_REFRESH.Location = new System.Drawing.Point(274, 505);
01289     this.btnClose_REFRESH.Name = "btnCV_REFRESH";
01290     this.btnClose_REFRESH.Size = new System.Drawing.Size(75, 23);
01291     this.btnClose_REFRESH.TabIndex = 5;
01292     this.btnClose_REFRESH.Text = "Refresh";
01293     this.btnClose_REFRESH.UseVisualStyleBackColor = false;
01294     this.btnClose_REFRESH.Click += new System.EventHandler(this.
    btnCV_REFRESH_Click);
01295     //
01296     // btnCV_PRINT
01297     //
01298     this.btnClose_PRINT.BackColor = System.Drawing.Color.Black;
01299     this.btnClose_PRINT.Location = new System.Drawing.Point(21, 505);
01300     this.btnClose_PRINT.Name = "btnCV_PRINT";
01301     this.btnClose_PRINT.Size = new System.Drawing.Size(75, 23);
01302     this.btnClose_PRINT.TabIndex = 4;
01303     this.btnClose_PRINT.Text = "Print";
01304     this.btnClose_PRINT.UseVisualStyleBackColor = false;
01305     this.btnClose_PRINT.Click += new System.EventHandler(this.
    btnCV_PRINT_Click);
01306     //
01307     // lstbxConvView
01308     //
01309     this.lstbxConvView.BackColor = System.Drawing.Color.Black;
01310     this.lstbxConvView.BorderStyle = System.Windows.Forms.BorderStyle.
    None;
```

```

01311     this.lstbxConvView.Cursor = System.Windows.Forms.Cursors.Default;
01312     this.lstbxConvView.ForeColor = System.Drawing.Color.DarkGoldenrod;
01313     this.lstbxConvView.FormattingEnabled = true;
01314     this.lstbxConvView.Items.AddRange(new object[] {
01315         "Conversation Viewer for ClassroomInq"});
01316     this.lstbxConvView.Location = new System.Drawing.Point(12, 73);
01317     this.lstbxConvView.Name = "lstbxConvView";
01318     this.lstbxConvView.SelectionMode = System.Windows.Forms.
01319         SelectionMode.MultiExtended;
01320     this.lstbxConvView.Size = new System.Drawing.Size(361, 416);
01321     this.lstbxConvView.TabIndex = 3;
01322     this.lstbxConvView.MouseLeave += new System.EventHandler(this.
01323         lstbxConvView_MouseLeave);
01324     //
01325     // btnCLS_ConvView
01326     //
01327     this.btnCLS_ConvView.BackColor = System.Drawing.Color.Black;
01328     this.btnCLS_ConvView.Location = new System.Drawing.Point(153, 505);
01329     this.btnCLS_ConvView.Name = "btnCLS_ConvView";
01330     this.btnCLS_ConvView.Size = new System.Drawing.Size(75, 23);
01331     this.btnCLS_ConvView.TabIndex = 1;
01332     this.btnCLS_ConvView.Text = "Close";
01333     this.btnCLS_ConvView.UseVisualStyleBackColor = false;
01334     this.btnCLS_ConvView.Click += new System.EventHandler(this.
01335         btnCLS_ConvView_Click);
01336     //
01337     // pictureBox4
01338     //
01339     this.pictureBox4.Image = ((System.Drawing.Image)(resources.
01340         GetObject("pictureBox4.Image")));
01341     this.pictureBox4.Location = new System.Drawing.Point(32, 17);
01342     this.pictureBox4.Name = "pictureBox4";
01343     this.pictureBox4.Size = new System.Drawing.Size(321, 50);
01344     this.pictureBox4.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
01345         StretchImage;
01346     this.pictureBox4.TabIndex = 0;
01347     this.pictureBox4.TabStop = false;
01348     //
01349     // Printer
01350     //
01351     this.Printer.DocumentName = "Student-Conversation";
01352     this.Printer.OriginAtMargins = true;
01353     this.Printer.PrintPage += new System.Drawing.Printing.
01354         PrintPageEventHandler(this.Printer_PrintPage);
01355     //
01356     // cntxtMenu
01357     //
01358     this.cntxtMenu.Popup += new System.EventHandler(this.
01359         cntxtMenu_arr_Popup);
01360     //
01361     // muIItemDelete
01362     //
01363     this.muIItemDelete.Index = -1;
01364     this.muIItemDelete.Text = "Delete this question?";
01365     this.muIItemDelete.Click += new System.EventHandler(this.
01366         muIItemDelete_Click);
01367     //
01368     // muIItemQuizDelete
01369     //
01370     this.muIItemQuizDelete.Index = -1;
01371     this.muIItemQuizDelete.Text = "Undo Delete";
01372     this.muIItemQuizDelete.Click += new System.EventHandler(this.
01373         muIItemQuizDelete_Click);
01374     //
01375     // muIItemMarkAsRead
01376     //
01377     this.muIItemMarkAsRead.Index = -1;
01378     this.muIItemMarkAsRead.Text = "Mark as Read";
01379     this.muIItemMarkAsRead.Click += new System.EventHandler(this.
01380         muIItemMarkAsRead_Click);
01381     //
01382     // muIItemMarkAsUnRead
01383     //
01384     this.muIItemMarkAsUnRead.Index = -1;

```

```
01375         this.muItmMarkAsUnRead.Text = "Mark as Unread";
01376         this.muItmMarkAsUnRead.Click += new System.EventHandler(this.
01377             muItmMarkAsUnRead_Click);
01377         //
01378         // muItmPrefs
01379         //
01380         this.muItmPrefs.Index = -1;
01381         this.muItmPrefs.Text = "Preferences";
01382         this.muItmPrefs.Click += new System.EventHandler(this.
01383             muItmPrefs_Click);
01383         //
01384         // muItmDM
01385         //
01386         this.muItmDM.Index = -1;
01387         this.muItmDM.Text = "Direct Message";
01388         this.muItmDM.Click += new System.EventHandler(this.muItmDM_Click);
01389         //
01390         // muItmFAQ
01391         //
01392         this.muItmFAQ.Index = -1;
01393         this.muItmFAQ.Text = "FAQ";
01394         this.muItmFAQ.Click += new System.EventHandler(this.muItmFAQ_Click)
01394 ;
01395         //
01396         // muItmQuit
01397         //
01398         this.muItmQuit.Index = -1;
01399         this.muItmQuit.Text = "sudo killall ClassroomInq";
01400         this.muItmQuit.Click += new System.EventHandler(this.
01400             muItmQuit_Click);
01401         //
01402         // muItmUndo
01403         //
01404         this.muItmUndo.Index = -1;
01405         this.muItmUndo.Text = "Undo Delete";
01406         this.muItmUndo.Click += new System.EventHandler(this.
01406             muItmUndo_Click);
01407         //
01408         // muItmName
01409         //
01410         this.muItmName.Index = -1;
01411         this.muItmName.Text = "Toggle Name";
01412         this.muItmName.Click += new System.EventHandler(this.
01412             muItmName_Click);
01413         //
01414         // muItmAttendance
01415         //
01416         this.muItmAttendance.Index = -1;
01417         this.muItmAttendance.Text = "Attendance";
01418         this.muItmAttendance.Click += new System.EventHandler(this.
01418             muItmAttendance_Click);
01419         //
01420         // muItmClassVote
01421         //
01422         this.muItmClassVote.Index = -1;
01423         this.muItmClassVote.Text = "Class Vote";
01424         this.muItmClassVote.Click += new System.EventHandler(this.
01424             muItmClassVote_Click);
01425         //
01426         // muItmQuiz
01427         //
01428         this.muItmQuiz.Index = -1;
01429         this.muItmQuiz.Text = "Quiz";
01430         this.muItmQuiz.Click += new System.EventHandler(this.
01430             muItmQuiz_Click);
01431         //
01432         // muItmBrdCst
01433         //
01434         this.muItmBrdCst.Index = -1;
01435         this.muItmBrdCst.Text = "BroadCast";
01436         this.muItmBrdCst.Click += new System.EventHandler(this.
01436             muItmBrdCst_Click);
01437         //
01438         // PanelQuizMaker
```

```

01439      // 
01440      this.PanelQuizMaker.AutoScroll = true;
01441      this.PanelQuizMaker.BackColor = System.Drawing.SystemColors.
01442      ControlText;
01443      this.PanelQuizMaker.Controls.Add(this.btnAddQM);
01444      this.PanelQuizMaker.Controls.Add(this.btnQMDel);
01445      this.PanelQuizMaker.Controls.Add(this.btnQMClr);
01446      this.PanelQuizMaker.Controls.Add(this.txtbxQM);
01447      this.PanelQuizMaker.Controls.Add(this.btnQMcls);
01448      this.PanelQuizMaker.Controls.Add(this.btnQMEdit);
01449      this.PanelQuizMaker.Controls.Add(this.lstbxQuizMaker);
01450      this.PanelQuizMaker.Controls.Add(this.pictureBox5);
01451      this.PanelQuizMaker.Controls.Add(this.btnQuizMode);
01452      this.PanelQuizMaker.ForeColor = System.Drawing.Color.DarkGoldenrod;
01453      this.PanelQuizMaker.Location = new System.Drawing.Point(400, 0);
01454      this.PanelQuizMaker.Name = "PanelQuizMaker";
01455      this.PanelQuizMaker.Size = new System.Drawing.Size(384, 524);
01456      this.PanelQuizMaker.TabIndex = 11;
01457      // 
01458      // btnAddQM
01459      // 
01460      this.btnAddQM.BackColor = System.Drawing.Color.Black;
01461      this.btnAddQM.Location = new System.Drawing.Point(162, 423);
01462      this.btnAddQM.Name = "btnAddQM";
01463      this.btnAddQM.Size = new System.Drawing.Size(60, 23);
01464      this.btnAddQM.TabIndex = 8;
01465      this.btnAddQM.Text = "Add";
01466      this.btnAddQM.UseVisualStyleBackColor = false;
01467      this.btnAddQM.Click += new System.EventHandler(this.btnAddQM_Click)
01468      ;
01469      // 
01470      // btnQMDel
01471      // 
01472      this.btnQMDel.BackColor = System.Drawing.Color.Black;
01473      this.btnQMDel.Location = new System.Drawing.Point(252, 423);
01474      this.btnQMDel.Name = "btnQMDel";
01475      this.btnQMDel.Size = new System.Drawing.Size(60, 23);
01476      this.btnQMDel.TabIndex = 10;
01477      this.btnQMDel.Text = "Delete";
01478      this.btnQMDel.UseVisualStyleBackColor = false;
01479      this.btnQMDel.MouseDown += new System.Windows.Forms.MouseEventHandler(this.btnQMDel_MouseDown);
01480      // 
01481      // btnQMClr
01482      // 
01483      this.btnQMClr.BackColor = System.Drawing.Color.Black;
01484      this.btnQMClr.Location = new System.Drawing.Point(72, 423);
01485      this.btnQMClr.Name = "btnQMClr";
01486      this.btnQMClr.Size = new System.Drawing.Size(60, 23);
01487      this.btnQMClr.TabIndex = 9;
01488      this.btnQMClr.Text = "Clear";
01489      this.btnQMClr.UseVisualStyleBackColor = false;
01490      this.btnQMClr.Click += new System.EventHandler(this.btnQMClr_Click)
01491      ;
01492      // 
01493      // txtbxQM
01494      // 
01495      this.txtbxQM.BorderStyle = System.Windows.Forms.BorderStyle.
01496      FixedSingle;
01497      this.txtbxQM.Location = new System.Drawing.Point(12, 380);
01498      this.txtbxQM.Name = "txtbxQM";
01499      this.txtbxQM.Size = new System.Drawing.Size(358, 20);
01500      this.txtbxQM.TabIndex = 7;
01501      this.txtbxQM.Text = "^^^^ Select a question to edit. ^^^^";
01502      this.txtbxQM.TextAlign = System.Windows.Forms.HorizontalAlignment.
01503      Center;
01504      this.txtbxQM.Click += new System.EventHandler(this.txtbxQM_Click);
01505      // 
01506      // btnQMcls
01507      // 
01508      this.btnQMcls.BackColor = System.Drawing.Color.Black;
01509      this.btnQMcls.Location = new System.Drawing.Point(252, 486);
01510      this.btnQMcls.Name = "btnQMcls";
01511      this.btnQMcls.Size = new System.Drawing.Size(111, 23);

```

```

01507         this.btnClose.TabIndex = 1;
01508         this.btnClose.Text = "Close Quiz Maker";
01509         this.btnClose.UseVisualStyleBackColor = false;
01510         this.btnClose.Click += new System.EventHandler(this.btnClose_Click);
01511     ;
01512     // btnQMEdit
01513     //
01514     this.btnQMEdit.BackColor = System.Drawing.Color.Black;
01515     this.btnQMEdit.Location = new System.Drawing.Point(142, 333);
01516     this.btnQMEdit.Name = "btnQMEdit";
01517     this.btnQMEdit.Size = new System.Drawing.Size(101, 23);
01518     this.btnQMEdit.TabIndex = 6;
01519     this.btnQMEdit.Text = "Edit Quiz";
01520     this.btnQMEdit.UseVisualStyleBackColor = false;
01521     this.btnQMEdit.Click += new System.EventHandler(this.
01522         btnQMEdit_Click);
01523     //
01524     // lstbxQuizMaker
01525     //
01526     this.lstbxQuizMaker.BackColor = System.Drawing.Color.Black;
01527     this.lstbxQuizMaker.ForeColor = System.Drawing.Color.DarkGoldenrod;
01528     this.lstbxQuizMaker.FormattingEnabled = true;
01529     this.lstbxQuizMaker.Items.AddRange(new object[] {
01530         "Quiz Questions",
01531         "Will go here"});
01532     this.lstbxQuizMaker.Location = new System.Drawing.Point(15, 78);
01533     this.lstbxQuizMaker.Name = "lstbxQuizMaker";
01534     this.lstbxQuizMaker.Size = new System.Drawing.Size(355, 251);
01535     this.lstbxQuizMaker.TabIndex = 5;
01536     //
01537     // pictureBox5
01538     //
01539     this.pictureBox5.Image = ((System.Drawing.Image)(resources.
01540         GetObject("pictureBox5.Image")));
01541     this.pictureBox5.Location = new System.Drawing.Point(32, 17);
01542     this.pictureBox5.Name = "pictureBox5";
01543     this.pictureBox5.Size = new System.Drawing.Size(321, 50);
01544     this.pictureBox5.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
01545         StretchImage;
01546     this.pictureBox5.TabIndex = 0;
01547     this.pictureBox5.TabStop = false;
01548     //
01549     // btnQuizMode
01550     //
01551     this.btnQuizMode.BackColor = System.Drawing.Color.Black;
01552     this.btnQuizMode.Location = new System.Drawing.Point(21, 486);
01553     this.btnQuizMode.Name = "btnQuizMode";
01554     this.btnQuizMode.Size = new System.Drawing.Size(110, 23);
01555     this.btnQuizMode.TabIndex = 4;
01556     this.btnQuizMode.Text = "Enter Quiz Mode";
01557     this.btnQuizMode.UseVisualStyleBackColor = false;
01558     this.btnQuizMode.Click += new System.EventHandler(this.
01559         btnQuizMode_Click);
01560     //
01561     // PanelQuizMode
01562     //
01563     this.PanelQuizMode.AutoScroll = true;
01564     this.PanelQuizMode.BackColor = System.Drawing.SystemColors.
01565         ControlText;
01566     this.PanelQuizMode.Controls.Add(this.pnlQuiz);
01567     this.PanelQuizMode.Controls.Add(this.btnExitQuiz);
01568     this.PanelQuizMode.Controls.Add(this.pictureBox6);
01569     this.PanelQuizMode.ForeColor = System.Drawing.Color.DarkGoldenrod;
01570     this.PanelQuizMode.Location = new System.Drawing.Point(-400, 0);
01571     this.PanelQuizMode.Name = "PanelQuizMode";
01572     this.PanelQuizMode.Size = new System.Drawing.Size(384, 550);
01573     this.PanelQuizMode.TabIndex = 12;
01574     //
01575     // pnlQuiz
01576     //
01577     this.pnlQuiz.AutoScroll = true;
01578     this.pnlQuiz.BackColor = System.Drawing.SystemColors.
01579         ControlDarkDark;

```

```

01574     this.pnlQuiz.Location = new System.Drawing.Point(12, 78);
01575     this.pnlQuiz.Name = "pnlQuiz";
01576     this.pnlQuiz.Size = new System.Drawing.Size(361, 402);
01577     this.pnlQuiz.TabIndex = 2;
01578     //
01579     // btnExitQuiz
01580     //
01581     this.btnExitQuiz.BackColor = System.Drawing.Color.Black;
01582     this.btnExitQuiz.Location = new System.Drawing.Point(137, 486);
01583     this.btnExitQuiz.Name = "btnExitQuiz";
01584     this.btnExitQuiz.Size = new System.Drawing.Size(111, 23);
01585     this.btnExitQuiz.TabIndex = 1;
01586     this.btnExitQuiz.Text = "Exit Quiz Mode";
01587     this.btnExitQuiz.UseVisualStyleBackColor = false;
01588     this.btnExitQuiz.Click += new System.EventHandler(this.
01589         btnExitQuiz_Click);
01590     //
01591     // pictureBox6
01592     //
01593     this.pictureBox6.Image = ((System.Drawing.Image)(resources.
01594         GetObject("pictureBox6.Image")));
01595     this.pictureBox6.Location = new System.Drawing.Point(32, 17);
01596     this.pictureBox6.Name = "pictureBox6";
01597     this.pictureBox6.Size = new System.Drawing.Size(321, 50);
01598     this.pictureBox6.SizeMode = System.Windows.Forms.PictureBoxSizeMode
01599         .StretchImage;
01600     this.pictureBox6.TabIndex = 0;
01601     this.pictureBox6.TabStop = false;
01602     //
01603     // PanelAttendance
01604     //
01605     this.PanelAttendance.BackColor = System.Drawing.SystemColors.
01606     ControlText;
01607     ;
01608     this.PanelAttendance.Location = new System.Drawing.Point(190, 265);
01609     this.PanelAttendance.Name = "PanelAttendance";
01610     this.PanelAttendance.Size = new System.Drawing.Size(0, 0);
01611     this.PanelAttendance.TabIndex = 13;
01612     this.PanelAttendance.Visible = false;
01613     //
01614     // lstbxAttendance
01615     //
01616     this.lstbxAttendance.BackColor = System.Drawing.Color.DarkGoldenrod
01617     ;
01618     this.lstbxAttendance.BorderStyle = System.Windows.Forms.BorderStyle
01619         .FixedSingle;
01620     this.lstbxAttendance.ForeColor = System.Drawing.Color.Black;
01621     this.lstbxAttendance.FormattingEnabled = true;
01622     this.lstbxAttendance.Items.AddRange(new object[] {
01623         "Student's will show up here when they've said they're here:"});
01624     this.lstbxAttendance.Location = new System.Drawing.Point(32, 89);
01625     this.lstbxAttendance.Name = "lstbxAttendance";
01626     this.lstbxAttendance.Size = new System.Drawing.Size(321, 379);
01627     this.lstbxAttendance.TabIndex = 2;
01628     //
01629     // btnExitAttendance
01630     //
01631     this.btnExitAttendance.BackColor = System.Drawing.Color.Black;
01632     this.btnExitAttendance.Location = new System.Drawing.Point(125, 486
01633     );
01634     this.btnExitAttendance.Name = "btnExitAttendance";
01635     this.btnExitAttendance.Size = new System.Drawing.Size(135, 23);
01636     this.btnExitAttendance.TabIndex = 1;
01637     this.btnExitAttendance.Text = "Exit Attendance Mode";
01638     this.btnExitAttendance.UseVisualStyleBackColor = false;
01639     this.btnExitAttendance.Click += new System.EventHandler(this.
01640         btnExitAttendance_Click);
01641     //
01642     // pictureBox7
01643     //

```

```
01639         this.pictureBox7.Image = ((System.Drawing.Image)(resources.
01640             GetObject("pictureBox7.Image")));
01641         this.pictureBox7.Location = new System.Drawing.Point(32, 18);
01642         this.pictureBox7.Name = "pictureBox7";
01643         this.pictureBox7.Size = new System.Drawing.Size(321, 50);
01644         this.pictureBox7.SizeMode = System.Windows.Forms.PictureBoxSizeMode.
01645             StretchImage;
01646         this.pictureBox7.TabIndex = 0;
01647         this.pictureBox7.TabStop = false;
01648         //
01649         // PanelClassVote
01650         //
01651         this.PanelClassVote.BackColor = System.Drawing.SystemColors.
01652             ControlText;
01653         this.PanelClassVote.Controls.Add(this.btnClose);
01654         this.PanelClassVote.Controls.Add(this.txtbxCVStats);
01655         this.PanelClassVote.Controls.Add(this.lblOption2);
01656         this.PanelClassVote.Controls.Add(this.lblOption1);
01657         this.PanelClassVote.Controls.Add(this.lblCVPipe);
01658         this.PanelClassVote.Controls.Add(this.lblCVRight);
01659         this.PanelClassVote.Controls.Add(this.lblCVLeft);
01660         this.PanelClassVote.Controls.Add(this.btnExitClassVote);
01661         this.PanelClassVote.ForeColor = System.Drawing.Color.DarkGoldenrod;
01662         this.PanelClassVote.Location = new System.Drawing.Point(190, 265);
01663         this.PanelClassVote.Name = "PanelClassVote";
01664         this.PanelClassVote.Size = new System.Drawing.Size(0, 0);
01665         this.PanelClassVote.TabIndex = 14;
01666         this.PanelClassVote.Visible = false;
01667         //
01668         // btnCVReset
01669         //
01670         this.btnClose.BackColor = System.Drawing.Color.Black;
01671         this.btnClose.Location = new System.Drawing.Point(32, 486);
01672         this.btnClose.Name = "btnCVReset";
01673         this.btnClose.Size = new System.Drawing.Size(135, 23);
01674         this.btnClose.TabIndex = 8;
01675         this.btnClose.Text = "Reset Voting";
01676         this.btnClose.UseVisualStyleBackColor = false;
01677         this.btnClose.Visible = false;
01678         this.btnClose.Click += new System.EventHandler(this.
01679             btnCVReset_Click);
01680         //
01681         // txtbxCVStats
01682         //
01683         this.txtbxCVStats.BackColor = System.Drawing.Color.DarkGoldenrod;
01684         this.txtbxCVStats.BorderStyle = System.Windows.Forms.BorderStyle.
01685             FixedSingle;
01686         this.txtbxCVStats.Font = new System.Drawing.Font("Century Gothic",
01687             10F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
01688             ((byte)(0)));
01689         this.txtbxCVStats.ForeColor = System.Drawing.Color.Black;
01690         this.txtbxCVStats.Location = new System.Drawing.Point(40, 297);
01691         this.txtbxCVStats.Multiline = true;
01692         this.txtbxCVStats.Name = "txtbxCVStats";
01693         this.txtbxCVStats.ReadOnly = true;
01694         this.txtbxCVStats.Size = new System.Drawing.Size(304, 50);
01695         this.txtbxCVStats.TabIndex = 7;
01696         this.txtbxCVStats.Text = "...Stats of Vote...";
01697         this.txtbxCVStats.TextAlign = System.Windows.Forms.
01698             HorizontalAlignment.Center;
01699         this.txtbxCVStats.Visible = false;
01700         //
01701         // lblOption2
01702         //
01703         this.lblOption2.Font = new System.Drawing.Font("Century Gothic", 18
01704             F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point,
01705             ((byte)(0)));
01706         this.lblOption2.Location = new System.Drawing.Point(213, 95);
01707         this.lblOption2.Name = "lblOption2";
01708         this.lblOption2.Size = new System.Drawing.Size(156, 32);
01709         this.lblOption2.TabIndex = 6;
01710         this.lblOption2.Text = "Option #2";
01711         this.lblOption2.TextAlign = System.Drawing.ContentAlignment.
```

```

        MiddleCenter;
01703            this.lblOption2.Visible = false;
01704            //
01705            // lblOption1
01706            //
01707            this.lblOption1.Font = new System.Drawing.Font("Century Gothic", 18
F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
01708            this.lblOption1.Location = new System.Drawing.Point(26, 95);
01709            this.lblOption1.Name = "lblOption1";
01710            this.lblOption1.Size = new System.Drawing.Size(156, 32);
01711            this.lblOption1.TabIndex = 5;
01712            this.lblOption1.Text = "Option #1";
01713            this.lblOption1.TextAlign = System.Drawing.ContentAlignment.
MiddleCenter;
01714            this.lblOption1.Visible = false;
01715            //
01716            // lblCVPipe
01717            //
01718            this.lblCVPipe.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01719            this.lblCVPipe.Font = new System.Drawing.Font("Poor Richard", 84F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0))
));
01720            this.lblCVPipe.Location = new System.Drawing.Point(163, 124);
01721            this.lblCVPipe.Name = "lblCVPipe";
01722            this.lblCVPipe.Size = new System.Drawing.Size(65, 156);
01723            this.lblCVPipe.TabIndex = 4;
01724            this.lblCVPipe.Text = "|";
01725            this.lblCVPipe.TextAlign = System.Drawing.ContentAlignment.
MiddleCenter;
01726            this.lblCVPipe.Visible = false;
01727            //
01728            // lblCVRight
01729            //
01730            this.lblCVRight.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01731            this.lblCVRight.Font = new System.Drawing.Font("Poor Richard", 84F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0))
));
01732            this.lblCVRight.Location = new System.Drawing.Point(210, 124);
01733            this.lblCVRight.Name = "lblCVRight";
01734            this.lblCVRight.Size = new System.Drawing.Size(155, 156);
01735            this.lblCVRight.TabIndex = 3;
01736            this.lblCVRight.Text = "0";
01737            this.lblCVRight.TextAlign = System.Drawing.ContentAlignment.
MiddleCenter;
01738            this.lblCVRight.Visible = false;
01739            //
01740            // lblCVLeft
01741            //
01742            this.lblCVLeft.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01743            this.lblCVLeft.Font = new System.Drawing.Font("Poor Richard", 84F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0))
));
01744            this.lblCVLeft.Location = new System.Drawing.Point(16, 124);
01745            this.lblCVLeft.Name = "lblCVLeft";
01746            this.lblCVLeft.Size = new System.Drawing.Size(155, 156);
01747            this.lblCVLeft.TabIndex = 2;
01748            this.lblCVLeft.Text = "0";
01749            this.lblCVLeft.TextAlign = System.Drawing.ContentAlignment.
MiddleCenter;
01750            this.lblCVLeft.Visible = false;
01751            //
01752            // btnExitClassVote
01753            //
01754            this.btnExitClassVote.BackColor = System.Drawing.Color.Black;
01755            this.btnExitClassVote.Location = new System.Drawing.Point(218, 486)
;
01756            this.btnExitClassVote.Name = "btnExitClassVote";
01757            this.btnExitClassVote.Size = new System.Drawing.Size(135, 23);
01758            this.btnExitClassVote.TabIndex = 1;
01759            this.btnExitClassVote.Text = "Exit Class Vote Mode";
01760            this.btnExitClassVote.UseVisualStyleBackColor = false;
01761            this.btnExitClassVote.Visible = false;
01762            this.btnExitClassVote.Click += new System.EventHandler(this.

```

```
        btnExitClassVote_Click);
01763        // 
01764        // pictureBox8
01765        // 
01766        this.pictureBox8.Image = ((System.Drawing.Image)(resources.
01767        GetObject("pictureBox8.Image")));
01768        this.pictureBox8.Location = new System.Drawing.Point(32, 18);
01769        this.pictureBox8.Name = "pictureBox8";
01770        this.pictureBox8.Size = new System.Drawing.Size(321, 50);
01771        this.pictureBox8.SizeMode = System.Windows.Forms.PictureBoxSizeModeSizeMode
01772        .StretchImage;
01773        this.pictureBox8.TabIndex = 0;
01774        this.pictureBox8.TabStop = false;
01775        this.pictureBox8.Visible = false;
01776        // 
01777        // colorDlg
01778        // 
01779        this.colorDlg.AnyColor = true;
01780        this.colorDlg.FullOpen = true;
01781        // 
01782        this.pnlBrdCst.Controls.Add(this.btnSend);
01783        this.pnlBrdCst.Controls.Add(this.textBox1);
01784        this.pnlBrdCst.Location = new System.Drawing.Point(0, -45);
01785        this.pnlBrdCst.Name = "pnlBrdCst";
01786        this.pnlBrdCst.Size = new System.Drawing.Size(385, 66);
01787        this.pnlBrdCst.TabIndex = 15;
01788        this.pnlBrdCst.Leave += new System.EventHandler(this.
01789        pnlBrdCst_Leave);
01790        // 
01791        // btnSend
01792        // 
01793        this.btnSend.Location = new System.Drawing.Point(310, 5);
01794        this.btnSend.Name = "btnSend";
01795        this.btnSend.Size = new System.Drawing.Size(72, 52);
01796        this.btnSend.TabIndex = 3;
01797        this.btnSend.Text = "Send Msg";
01798        this.btnSend.UseVisualStyleBackColor = true;
01799        this.btnSend.Click += new System.EventHandler(this.btnSend_Click);
01800        // 
01801        // PanelNotify
01802        // 
01803        this.PanelNotify.BackColor = System.Drawing.Color.DarkGoldenrod;
01804        this.PanelNotify.Controls.Add(this.lblNotify);
01805        this.PanelNotify.Controls.Add(this.pictureBoxNotify);
01806        this.PanelNotify.ForeColor = System.Drawing.Color.Black;
01807        this.PanelNotify.Location = new System.Drawing.Point(0, 550);
01808        this.PanelNotify.Name = "PanelNotify";
01809        this.PanelNotify.Size = new System.Drawing.Size(385, 35);
01810        this.PanelNotify.TabIndex = 16;
01811        // 
01812        // lblNotify
01813        // 
01814        this.lblNotify.AutoEllipsis = true;
01815        this.lblNotify.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01816        this.lblNotify.Font = new System.Drawing.Font("Century Gothic", 10F
01817        , System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0
01818        )));
01819        this.lblNotify.Location = new System.Drawing.Point(3, 3);
01820        this.lblNotify.Name = "lblNotify";
01821        this.lblNotify.Size = new System.Drawing.Size(345, 31);
01822        this.lblNotify.TabIndex = 1;
01823        this.lblNotify.Text = " ";
01824        this.lblNotify.TextAlign = System.Drawing.ContentAlignment.
01825        MiddleCenter;
01826        this.lblNotify.MouseDown += new System.Windows.Forms.
01827        MouseEventHandler(this.lblNotify_MouseDown);
01828        // 
01829        // pictureBoxNotify
01830        // 
01831        this.pictureBoxNotify.BackgroundImage = global::SrP_ClassroomInq.
01832        Properties.Resources.comment_bubble;
01833        this.pictureBoxNotify.BackgroundImageLayout = System.Windows.Forms.
```

```

        ImageLayout.Stretch;
01828            this.pictureBoxNotify.Location = new System.Drawing.Point(348, 3);
01829            this.pictureBoxNotify.Name = "pictureBoxNotify";
01830            this.pictureBoxNotify.Size = new System.Drawing.Size(34, 31);
01831            this.pictureBoxNotify.TabIndex = 0;
01832            this.pictureBoxNotify.TabStop = false;
01833            this.pictureBoxNotify.Click += new System.EventHandler(this.
    pictureBoxNotify_Click);
01834            //
01835            // chkbxSplash
01836            //
01837            this.chkbxSplash.Checked = true;
01838            this.chkbxSplash.CheckState = System.Windows.Forms.CheckState.
    Checked;
01839            this.chkbxSplash.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
01840            this.chkbxSplash.Location = new System.Drawing.Point(18, 315);
01841            this.chkbxSplash.Name = "chkbxSplash";
01842            this.chkbxSplash.Size = new System.Drawing.Size(130, 33);
01843            this.chkbxSplash.TabIndex = 20;
01844            this.chkbxSplash.Text = "Splash Screen";
01845            this.chkbxSplash.TextAlign = System.Drawing.ContentAlignment.
    MiddleCenter;
01846            this.chkbxSplash.UseVisualStyleBackColor = true;
01847            //
01848            // frmClassrromInq
01849            //
01850            this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
01851            this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
01852            this.AutoValidate = System.Windows.Forms.AutoValidate.
    EnablePreventFocusChange;
01853            this.BackColor = System.Drawing.SystemColors.ControlText;
01854            this.ClientSize = new System.Drawing.Size(384, 550);
01855            this.Controls.Add(this.menuStrip1);
01856            this.Controls.Add(this.PanelConvView);
01857            this.Controls.Add(this.PanelClassVote);
01858            this.Controls.Add(this.pnlBrdCst);
01859            this.Controls.Add(this.PanelFAQ);
01860            this.Controls.Add(this.PanelPrefs);
01861            this.Controls.Add(this.PanelNotify);
01862            this.Controls.Add(this.grpbxFeed);
01863            this.Controls.Add(this.PanelAttendance);
01864            this.Controls.Add(this.PanelQuizMode);
01865            this.Controls.Add(this.PanelQuizMaker);
01866            this.Controls.Add(this.DirectMsgPanel);
01867            this.Controls.Add(this.PanelStudents);
01868            this.DoubleBuffered = true;
01869            this.ForeColor = System.Drawing.Color.DarkGoldenrod;
01870            this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.
    FixedDialog;
01871            this.Icon = ((System.Drawing.Icon)(resources.GetObject("$this.Icon"
    )));
01872            this.MainMenuStrip = this.menuStrip1;
01873            this.MaximizeBox = false;
01874            this.MinimizeBox = false;
01875            this.Name = "frmClassrromInq";
01876            this.Text = " Classroom Inquisition | Home";
01877            this.FormClosing += new System.Windows.Forms.
    FormClosingEventHandler(this.Form1_FormClosing);
01878            this.Load += new System.EventHandler(this.Form1_Load);
01879            this.menuStrip1.ResumeLayout(false);
01880            this.menuStrip1.PerformLayout();
01881            this.PanelPrefs.ResumeLayout(false);
01882            ((System.ComponentModel.ISupportInitialize)(this.numUpDnNotify)).EndInit();
01883            this.grpbxTheme.ResumeLayout(false);
01884            this.grpbxUnread.ResumeLayout(false);
01885            this.grpbxUnread.PerformLayout();
01886            ((System.ComponentModel.ISupportInitialize)(this.picbxPrefs)).EndInit();
01887            this.DirectMsgPanel.ResumeLayout(false);
01888            this.DirectMsgPanel.PerformLayout();
01889            ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).EndInit();
01890            this.PanelFAQ.ResumeLayout(false);

```

```
01891         this.PanelFAQ.PerformLayout();
01892         ((System.ComponentModel.ISupportInitialize)(this.pictureBox2)).BeginInit();
01893         this.PanelStudents.ResumeLayout(false);
01894         this.grpbxEDITStudents.ResumeLayout(false);
01895         this.grpbxEDITStudents.PerformLayout();
01896         ((System.ComponentModel.ISupportInitialize)(this.pictureBox3)).BeginInit();
01897         this.PanelConvView.ResumeLayout(false);
01898         ((System.ComponentModel.ISupportInitialize)(this.pictureBox4)).BeginInit();
01899         this.PanelQuizMaker.ResumeLayout(false);
01900         this.PanelQuizMaker.PerformLayout();
01901         ((System.ComponentModel.ISupportInitialize)(this.pictureBox5)).BeginInit();
01902         this.PanelQuizMode.ResumeLayout(false);
01903         ((System.ComponentModel.ISupportInitialize)(this.pictureBox6)).BeginInit();
01904         this.PanelAttendance.ResumeLayout(false);
01905         ((System.ComponentModel.ISupportInitialize)(this.pictureBox7)).BeginInit();
01906         this.PanelClassVote.ResumeLayout(false);
01907         this.PanelClassVote.PerformLayout();
01908         ((System.ComponentModel.ISupportInitialize)(this.pictureBox8)).BeginInit();
01909         this.pnlBrdCst.ResumeLayout(false);
01910         this.pnlBrdCst.PerformLayout();
01911         this.PanelNotify.ResumeLayout(false);
01912         ((System.ComponentModel.ISupportInitialize)(this.pictureBoxNotify)).BeginInit();
01913         this.ResumeLayout(false);
01914         this.PerformLayout();
01915     }
01916 }
01917 #endregion
01918 private System.Windows.Forms.MenuStrip menuStrip1;
01919 private System.Windows.Forms.ToolStripMenuItem sendToolStripMenuItem;
01920 private System.Windows.Forms.ToolStripMenuItem broadcastToolStripMenuItem;
01921 private System.Windows.Forms.ToolStripMenuItem directMessageToolStripMenuItem;
01922 private System.Windows.Forms.ToolStripMenuItem viewToolStripMenuItem;
01923 private System.Windows.Forms.ToolStripMenuItem helpToolStripMenuItem;
01924 private System.Windows.Forms.TextBox textBox1;
01925 private System.Windows.Forms.ToolStripMenuItem PrefsToolStripMenuItem;
01926 private System.Windows.Forms.ToolStripMenuItem quitToolStripMenuItem;
01927 private System.Windows.Forms.Timer timer;
01928 private System.Windows.Forms.NotifyIcon trayICON;
01929 private System.Windows.Forms.Panel grpbxFeed;
01930 private System.IO.Ports.SerialPort SerialPort;
01931 private System.Windows.Forms.ToolStripMenuItem modesToolStripMenuItem;
01932 private System.Windows.Forms.ToolStripMenuItem attendedanceToolStripMenuItem;
01933 private System.Windows.Forms.ToolStripMenuItem classVoteToolStripMenuItem;
01934 private System.Windows.Forms.ToolStripMenuItem normalToolStripMenuItem;
01935 private System.Windows.Forms.ToolStripMenuItem quizToolStripMenuItem;
01936 private System.Windows.Forms.ToolStripMenuItem showNamesToolStripMenuItem;
01937 private System.Windows.Forms.ToolStripMenuItem generalFAQToolStripMenuItem;
01938 private System.Windows.Forms.ToolStripMenuItem aboutToolStripMenuItem;
01939 private System.Windows.Forms.Timer timer_SerialRead;
01940 private System.Windows.Forms.Panel PanelPrefs;
01941 private System.Windows.Forms.Panel DirectMsgPanel;
01942 private System.Windows.Forms.PictureBox pictureBox1;
01943 private System.Windows.Forms.Button btnDM_Send;
01944 private System.Windows.Forms.Button btnDM_Cls;
01945 private System.Windows.Forms.Button btnDM_Clr;
01946 private System.Windows.Forms.ComboBox cmbxDM;
01947 private System.Windows.Forms.TextBox txtbxDM;
01948 private System.Windows.Forms.PictureBox picbxPrefs;
```

```

01951     private System.Windows.Forms.Button btnPrefs_Cls;
01952     private System.Windows.Forms.CheckBox chkbxRXSound;
01953     private System.Windows.Forms.CheckBox chkbxTXSound;
01954     private System.Windows.Forms.Panel PanelFAQ;
01955     private System.Windows.Forms.Button btnFAQcls;
01956     private System.Windows.Forms.PictureBox pictureBox2;
01957     private System.Windows.Forms.TextBox txtbxFAQ;
01958     private System.Windows.Forms.Label lblFAQ;
01959     private System.Windows.Forms.ToolStripComboBox toolStripComboBox1;
01960     private System.Windows.Forms.ToolStripItem fileToolStripMenuItem;
01961     private System.Windows.Forms.ToolStripItem newToolStripMenuItem;
01962     private System.Windows.Forms.ToolStripItem openToolStripMenuItem;
01963     private System.Windows.Forms.ToolStripSeparator toolStripSeparator;
01964     private System.Windows.Forms.ToolStripItem saveToolStripMenuItem;
01965     private System.Windows.Forms.ToolStripItem saveAsToolStripMenuItem;
01966     private System.Windows.Forms.ToolStripSeparator toolStripSeparator1;
01967     private System.Windows.Forms.ToolStripItem printToolStripMenuItem;
01968     private System.Windows.Forms.ToolStripItem
01969         printPreviewToolStripMenuItem;
01970     private System.Windows.Forms.ToolStripSeparator toolStripSeparator2;
01971     private System.Windows.Forms.ToolStripItem exitToolStripMenuItem;
01972     private System.Windows.Forms.ToolStripItem editToolStripMenuItem;
01973     private System.Windows.Forms.ToolStripItem undoToolStripMenuItem;
01974     private System.Windows.Forms.ToolStripItem redoToolStripMenuItem;
01975     private System.Windows.Forms.ToolStripSeparator toolStripSeparator3;
01976     private System.Windows.Forms.ToolStripItem cutToolStripMenuItem;
01977     private System.Windows.Forms.ToolStripItem copyToolStripMenuItem;
01978     private System.Windows.Forms.ToolStripItem pasteToolStripMenuItem;
01979     private System.Windows.Forms.ToolStripSeparator toolStripSeparator4;
01980     private System.Windows.Forms.ToolStripItem
01981         selectAllToolStripMenuItem;
01982     private System.Windows.Forms.ToolStripItem toolsToolStripMenuItem;
01983     private System.Windows.Forms.ToolStripItem
01984         customizeToolStripMenuItem;
01985     private System.Windows.Forms.ToolStripItem optionsToolStripMenuItem
01986 ;
01987     private System.Windows.Forms.ToolStripItem helpToolStripMenuItem;
01988     private System.Windows.Forms.ToolStripItem
01989         contentsToolStripMenuItem;
01990     private System.Windows.Forms.ToolStripSeparator toolStripSeparator5;
01991     private System.Windows.Forms.GroupBox grpbxEDITStudents;
01992     private System.Windows.Forms.Label lblStudentsName;
01993     private System.Windows.Forms.TextBox txtbxStudentsName;
01994     private System.Windows.Forms.Button btnCLRStudents;
01995     private System.Windows.Forms.Button btnEDTStudents;
01996     private System.Windows.Forms.ListBox lstbxStudents;
01997     private System.Windows.Forms.Button btnCLSStudents;
01998     private System.Windows.Forms.PictureBox pictureBox3;
01999     private System.Windows.Forms.Button btnDoneStudents;
02000     private System.Windows.Forms.Button btnStuMgmt_Prefs;
02001     private System.Windows.Forms.CheckBox chkbxLameMode;
02002     private System.Windows.Forms.Timer timer_ControlsCreate;
02003     private System.Windows.Forms.GroupBox grpbxUnread;
02004     private System.Windows.Forms.RadioButton rdbtnClick;
02005     private System.Windows.Forms.LinkLabel lnklblIssues;
02006     private System.Windows.Forms.Panel PanelConvView;
02007     private System.Windows.Forms.Button btnCLS_ConvView;
02008     private System.Windows.Forms.PictureBox pictureBox4;
02009     private System.Windows.Forms.ListBox lstbxConvView;
02010     private System.Windows.Forms.Button btnConvView;
02011     private System.Windows.Forms.CheckBox chkbxToolips;
02012     private System.Windows.Forms.CheckBox chkbxNotify;
02013     private System.Windows.Forms.Button btnCV_Refresh;
02014     private System.Windows.Forms.Button btnCV_Print;
02015     private System.Drawing.Printing.PrintDocument Printer;
02016     private System.Windows.Forms.MenuItem multiDelete;
02017     private System.Windows.Forms.MenuItem multiQuizDelete;
02018     private System.Windows.Forms.ContextMenu ctxtMenu;
02019     private System.Windows.Forms.MenuItem multiPrefs;

```

```
02020     private System.Windows.Forms.MenuItem muItmDM;
02021     private System.Windows.Forms.MenuItem muItmFAQ;
02022     private System.Windows.Forms.MenuItem muItmQuit;
02023     private System.Windows.Forms.MenuItem muItmUndo;
02024     private System.Windows.Forms.MenuItem muItmName;
02025     private System.Windows.Forms.MenuItem muItmClassVote;
02026     private System.Windows.Forms.MenuItem muItmQuiz;
02027     private System.Windows.Forms.MenuItem muItmAttendance;
02028     private System.Windows.Forms.MenuItem muItmBrdCst;
02029     private System.Windows.Forms.MenuItem muItmMarkAsRead;
02030     private System.Windows.Forms.MenuItem muItmMarkAsUnRead;
02031     private System.Windows.Forms.CheckBox chkbxCtrlHide;
02032     private System.Windows.Forms.Button btnQuizMaker;
02033     private System.Windows.Forms.Panel PanelQuizMaker;
02034     private System.Windows.Forms.Button btnQuizMode;
02035     private System.Windows.Forms.Button btnQMcls;
02036     private System.Windows.Forms.PictureBox pictureBox5;
02037     private System.Windows.Forms.Button btnQMDel;
02038     private System.Windows.Forms.Button btnAddQM;
02039     private System.Windows.Forms.Button btnQMclr;
02040     private System.Windows.Forms.TextBox txtbxQM;
02041     private System.Windows.Forms.Button btnQMEdit;
02042     private System.Windows.Forms.ListBox lstbxQuizMaker;
02043     private System.Windows.Forms.Panel PanelQuizMode;
02044     private System.Windows.Forms.Panel pnlQuiz;
02045     private System.Windows.Forms.Button btnExitQuiz;
02046     private System.Windows.Forms.PictureBox pictureBox6;
02047     private System.Windows.Forms.Panel PanelAttendance;
02048     private System.Windows.Forms.Button btnExitAttendance;
02049     private System.Windows.Forms.PictureBox pictureBox7;
02050     private System.Windows.Forms.Panel PanelClassVote;
02051     private System.Windows.Forms.Button btnExitClassVote;
02052     private System.Windows.Forms.PictureBox pictureBox8;
02053     private System.Windows.Forms.Button btnForeColor;
02054     private System.Windows.Forms.Button btnBkgndColor;
02055     private System.Windows.Forms.ColorDialog colorDlg;
02056     private System.Windows.Forms.GroupBox grpbxTheme;
02057     private System.Windows.Forms.Button btnDefaultTheme;
02058     private System.Windows.Forms.Label lblAnimationPrefs;
02059     private System.Windows.Forms.ComboBox cmbbxAnimationSpeed;
02060     private System.Windows.Forms.Panel pnlBrdCst;
02061     private System.Windows.Forms.Button btnSend;
02062     private System.Windows.Forms.Label lblSerPt;
02063     private System.Windows.Forms.ComboBox serialCOMcmbbx;
02064     private System.Windows.Forms.ListBox lstbxAttendance;
02065     private System.Windows.Forms.Panel PanelNotify;
02066     private System.Windows.Forms.Label lblNotify;
02067     private System.Windows.Forms.PictureBox pictureBoxNotify;
02068     private System.Windows.Forms.NumericUpDown numUpDnNotify;
02069     private System.Windows.Forms.Label lblNotifyTime;
02070     private System.Windows.Forms.TextBox txtbxCVStats;
02071     private System.Windows.Forms.Label lblOption2;
02072     private System.Windows.Forms.Label lblOption1;
02073     private System.Windows.Forms.Label lblCVpipe;
02074     private System.Windows.Forms.Label lblCVRight;
02075     private System.Windows.Forms.Label lblCVLeft;
02076     private System.Windows.Forms.Button btnCVReset;
02077     private System.Windows.Forms.CheckBox chkbxSplash;
02078 }
02079 }
02080 }
```

**6.11 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/-
SrP_ClassroomInq/Form1.cs File Reference**

Classes

- class SrP_ClassroomInq::Form1

Packages

- package SrP_ClassroomInq

6.12 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1.cs

```

00001 using System;
00002 using System.Collections.Generic;
00003 using System.ComponentModel;
00004 using System.Data;
00005 using System.Drawing;
00006 using System.Linq;
00007 using System.Text;
00008 using System.Windows.Forms;
00009
00010 namespace SrP_ClassroomInq
00011 {
00012     public partial class Form1 : Form
00013     {
00014         public Form1()
00015         {
00016             InitializeComponent();
00017         }
00018         byte i = 0;
00019         byte k = 0;
00020         byte j = 0;
00021         bool textbox1WASclicked = new bool();
00022         bool grpbxRPL_WASclicked = new bool();
00023         bool btnCLS_WASclicked = new bool();
00024
00025         #region Click Events
00026
00027         private void quitToolStripMenuItem_Click(object sender, EventArgs e)
00028         {
00029             string message = "Leaving so soon?";
00030
00031             string caption = "Are you Sure you want to do that...?";
00032
00033             MessageBoxButtons buttons = MessageBoxButtons.YesNo;
00034             DialogResult result;
00035             result = MessageBox.Show(this, message, caption, buttons,
00036                         MessageBoxIcon.Question);
00037
00038             if (result == DialogResult.Yes)
00039             {
00040                 this.Close();
00041             }
00042
00043             private void textBox1_MouseClick(object sender, MouseEventArgs e)
00044             {
00045                 timer.Enabled = true;
00046                 textbox1WASclicked = true;
00047             }
00048
00049
00050             private void trayICON_MouseDoubleClick(object sender, MouseEventArgs e)
00051             {
00052                 if (FormWindowState.Minimized == this.WindowState)
00053                 {

```

```
00054         this.Show();
00055         this.WindowState = FormWindowState.Normal;
00056     }
00057     else if (FormWindowState.Normal == this.WindowState)
00058     {
00059         this.Hide();
00060         trayICON.ShowBalloonTip(500);
00061         this.WindowState = FormWindowState.Minimized;
00062     }
00063 }
00064 }
00065 #endregion Click Events
00066
00068     private void timer_Tick(object sender, EventArgs e)
00069     {
00070         #region Send Button
00071         if ((btnSend.Visible == false) && (textBox1WASclicked == true))// if not visible
00072         {
00073             if (i < 3)
00074             {
00075                 grpbxFeed.SetBounds(grpbxFed.Location.X, grpbxFeed.
00076                     Location.Y + 5, grpbxFeed.Size.Width, grpbxFeed.Size.Height - 2);
00077                 i++;
00078             }
00079             else if ((i > 2)&&(i < 9))
00080             {
00081                 grpbxFeed.SetBounds(grpbxFed.Location.X, grpbxFeed.
00082                     Location.Y + 4, grpbxFeed.Size.Width, grpbxFeed.Size.Height - 3);
00083                 i++;
00084             }
00085             else
00086             {
00087                 i = 0;
00088                 timer.Enabled = false;
00089                 btnSend.Visible = true;
00090                 textBox1WASclicked = false;
00091             }
00092         }
00093         #endregion
00094         #region GrpBxRPL_Clicked
00095         if ((grpbx_REPLY.Height < 137) && (grpbxRPL_WASclicked == true))// if not visible
00096         {
00097             if (k < 3)
00098             {
00099                 grpbx_REPLY.SetBounds(grpbx_REPLY.Location.X, grpbx_REPLY.
00100                     Location.Y, grpbx_REPLY.Size.Width, grpbx_REPLY.Size.Height + 3);
00101                 k++;
00102             }
00103             else if ((k > 2) && (k < 19))
00104             {
00105                 grpbx_REPLY.SetBounds(grpbx_REPLY.Location.X, grpbx_REPLY.
00106                     Location.Y, grpbx_REPLY.Size.Width, grpbx_REPLY.Size.Height + 6);
00107                 k++;
00108             }
00109             else
00110             {
00111                 k = 0;
00112                 timer.Enabled = false;
00113                 grpbxRPL_WASclicked = false;
00114             }
00115             textBox1.AppendText(grpbx_REPLY.Height.ToString() + Environment
00116 .NewLine); //troubleshooting
00117         }
00118         #endregion
00119         #region Destroy Reply
00120         if ((grpbx_REPLY.Height > 32) && (btnCLS_WASclicked == true))// if not visible
00121         {
```

```

00120             if (j < 3)
00121             {
00122                 grpbx_Reply.SetBounds(grpbx_Reply.Location.X, grpbx_Reply.
00123                     Location.Y, grpbx_Reply.Size.Width, grpbx_Reply.Size.Height - 3);
00124                 j++;
00125             }
00126             else if ((j > 2) && (j < 19))
00127             {
00128                 grpbx_Reply.SetBounds(grpbx_Reply.Location.X, grpbx_Reply.
00129                     Location.Y, grpbx_Reply.Size.Width, grpbx_Reply.Size.Height - 6);
00130                 j++;
00131             }
00132             else
00133             {
00134                 j = 0;
00135                 timer.Enabled = false;
00136                 btnCLS_WASclicked = false;
00137             }
00138         #endregion
00139     }
00140 }
00141
00142 private void btnCLR_Click(object sender, EventArgs e)
00143 {
00144     txtbx_Reply.ResetText(); //clear text
00145 }
00146
00147 private void lbl_question_Click(object sender, EventArgs e)
00148 {
00149     timer.Enabled = true;
00150     grpbxRPL_WASclicked = true;
00151 }
00152
00153 private void replyToolStripMenuItem_Click(object sender, EventArgs e)
00154 {
00155     timer.Enabled = true;
00156     grpbxRPL_WASclicked = true;
00157 }
00158
00159 private void btnCLS_Click(object sender, EventArgs e)
00160 {
00161     timer.Enabled = true;
00162     btnCLS_WASclicked = true;
00163 }
00164
00165 private void Form1_Load(object sender, EventArgs e)
00166 {
00167 }
00168
00169
00170
00171
00172
00173
00174 }
00175 }
```

6.13 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1.Designer.cs File - Reference

Classes

- class [SrP_ClassroomInq::Form1](#)

Packages

- package [SrP_ClassroomInq](#)

6.14 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Form1.Designer.cs

```
00001 namespace SrP_ClassroomInq
00002 {
00003     partial class Form1
00004     {
00008         private System.ComponentModel.IContainer components = null;
00009
0014     protected override void Dispose(bool disposing)
00015     {
00016         if (disposing && (components != null))
00017         {
00018             components.Dispose();
00019         }
00020         base.Dispose(disposing);
00021     }
00022
00023     #region Windows Form Designer generated code
00024
00029     private void InitializeComponent()
00030     {
00031         this.components = new System.ComponentModel.Container();
00032         System.ComponentModel.ComponentResourceManager resources = new
System.ComponentModel.ComponentResourceManager(typeof(Form1));
00033         this.menuStrip1 = new System.Windows.Forms.MenuStrip();
00034         this.sendToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00035         this.broadcastToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00036         this.directMessageToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00037         this.replyToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00038         this.controlsToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00039         this.viewToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00040         this.helpToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00041         this.quitToolStripMenuItem = new System.Windows.Forms.
ToolStripMenuItem();
00042         this.textBox1 = new System.Windows.Forms.TextBox();
00043         this.statusStrip1 = new System.Windows.Forms.StatusStrip();
00044         this.sb_send_status = new System.Windows.Forms.ToolStripStatusLabel
();
00045         this.timer = new System.Windows.Forms.Timer(this.components);
00046         this.btnSend = new System.Windows.Forms.Button();
00047         this.trayICON = new System.Windows.Forms.NotifyIcon(this.components
);
00048         this.grpbxFeed = new System.Windows.Forms.GroupBox();
00049         this.vScrollBar1 = new System.Windows.Forms.VScrollBar();
00050         this.grpbx_Reply = new System.Windows.Forms.GroupBox();
00051         this.btnRPL = new System.Windows.Forms.Button();
00052         this.btnCLS = new System.Windows.Forms.Button();
00053         this.btnCLR = new System.Windows.Forms.Button();
00054         this.txtbx_Reply = new System.Windows.Forms.TextBox();
```

```

00055         this.lbl_question = new System.Windows.Forms.Label();
00056         this.menuStrip1.SuspendLayout();
00057         this.statusStrip1.SuspendLayout();
00058         this.grpbxFeed.SuspendLayout();
00059         this.grpbx_Reply.SuspendLayout();
00060         this.SuspendLayout();
00061         //
00062         // menuStrip1
00063         //
00064         this.menuStrip1.BackColor = System.Drawing.SystemColors.Menu;
00065         this.menuStrip1.Items.AddRange(new System.Windows.Forms.
00066         ToolStripItem[] {
00067             this.sendToolStripMenuItem,
00068             this.replyToolStripMenuItem,
00069             this.controlsToolStripMenuItem,
00070             this.viewToolStripMenuItem,
00071             this.helpToolStripMenuItem,
00072             this.quitToolStripMenuItem});
00073         this.menuStrip1.Location = new System.Drawing.Point(0, 0);
00074         this.menuStrip1.Name = "menuStrip1";
00075         this.menuStrip1.Size = new System.Drawing.Size(364, 24);
00076         this.menuStrip1.TabIndex = 0;
00077         this.menuStrip1.Text = "menuStrip1";
00078         //
00079         // sendToolStripMenuItem
00080         //
00081         this.sendToolStripMenuItem.DropDownItems.AddRange(new System.
00082         Windows.Forms.ToolStripItem[] {
00083             this.broadcastToolStripMenuItem,
00084             this.directMessageToolStripMenuItem});
00085         this.sendToolStripMenuItem.Name = "sendToolStripMenuItem";
00086         this.sendToolStripMenuItem.Size = new System.Drawing.Size(63, 20);
00087         this.sendToolStripMenuItem.Text = "Compose";
00088         //
00089         // broadcastToolStripMenuItem
00090         //
00091         this.broadcastToolStripMenuItem.Name = "broadcastToolStripMenuItem"
00092         ;
00093         this.broadcastToolStripMenuItem.Size = new System.Drawing.Size(147,
00094         22);
00095         this.broadcastToolStripMenuItem.Text = "Broadcast";
00096         //
00097         // directMessageToolStripMenuItem
00098         //
00099         this.directMessageToolStripMenuItem.Name = "
00100         directMessageToolStripMenuItem";
00101         this.directMessageToolStripMenuItem.Size = new System.Drawing.Size(
00102         147, 22);
00103         this.directMessageToolStripMenuItem.Text = "Direct Message";
00104         //
00105         // replyToolStripMenuItem
00106         //
00107         this.replyToolStripMenuItem.Name = "replyToolStripMenuItem";
00108         this.replyToolStripMenuItem.Size = new System.Drawing.Size(46, 20);
00109         this.replyToolStripMenuItem.Text = "Reply";
00110         this.replyToolStripMenuItem.Click += new System.EventHandler(this.
00111         replyToolStripMenuItem_Click);
00112         //
00113         // controlsToolStripMenuItem
00114         //
00115         this.controlsToolStripMenuItem.Name = "controlsToolStripMenuItem";
00116         this.controlsToolStripMenuItem.Size = new System.Drawing.Size(59, 2
00117         0);
00118         this.controlsToolStripMenuItem.Text = "Controls";
00119         //
00120         // viewToolStripMenuItem
00121         //
00122         this.viewToolStripMenuItem.Name = "viewToolStripMenuItem";
00123         this.viewToolStripMenuItem.Size = new System.Drawing.Size(41, 20);
00124         this.viewToolStripMenuItem.Text = "View";
00125         //
00126         // helpToolStripMenuItem
00127         //
00128         this.helpToolStripMenuItem.Name = "helpToolStripMenuItem";

```

```
00121      this.helpToolStripMenuItem.Size = new System.Drawing.Size(40, 20);
00122      this.helpToolStripMenuItem.Text = "Help";
00123      //
00124      // quitToolStripMenuItem
00125      //
00126      this.quitToolStripMenuItem.Alignment = System.Windows.Forms.
00127      ToolStripItemAlignment.Right;
00127      this.quitToolStripMenuItem.Name = "quitToolStripMenuItem";
00128      this.quitToolStripMenuItem.Size = new System.Drawing.Size(39, 20);
00129      this.quitToolStripMenuItem.Text = "Quit";
00130      this.quitToolStripMenuItem.Click += new System.EventHandler(this.
00131      quitToolStripMenuItem_Click);
00131      //
00132      // textBox1
00133      //
00134      this.textBox1.Location = new System.Drawing.Point(12, 27);
00135      this.textBox1.Multiline = true;
00136      this.textBox1.Name = "textBox1";
00137      this.textBox1.Size = new System.Drawing.Size(340, 45);
00138      this.textBox1.TabIndex = 1;
00139      this.textBox1.Text = "Enter Message here..";
00140      this.textBox1.MouseClick += new System.Windows.Forms.
00141      MouseEventHandler(this.textBox1_MouseClick);
00141      //
00142      // statusStrip1
00143      //
00144      this.statusStrip1.Items.AddRange(new System.Windows.Forms.
00145      ToolStripItem[] {
00145      this.sb_send_status});
00146      this.statusStrip1.Location = new System.Drawing.Point(0, 528);
00147      this.statusStrip1.Name = "statusStrip1";
00148      this.statusStrip1.Size = new System.Drawing.Size(364, 22);
00149      this.statusStrip1.TabIndex = 2;
00150      this.statusStrip1.Text = "statusStrip1";
00151      //
00152      // sb_send_status
00153      //
00154      this.sb_send_status.Name = "sb_send_status";
00155      this.sb_send_status.Size = new System.Drawing.Size(167, 17);
00156      this.sb_send_status.Text = "Welcome to Classroom Inquisition";
00157      //
00158      // timer
00159      //
00160      this.timer.Interval = 75;
00161      this.timer.Tick += new System.EventHandler(this.timer_Tick);
00162      //
00163      // btnSend
00164      //
00165      this.btnSend.Location = new System.Drawing.Point(277, 80);
00166      this.btnSend.Name = "btnSend";
00167      this.btnSend.Size = new System.Drawing.Size(75, 29);
00168      this.btnSend.TabIndex = 4;
00169      this.btnSend.Text = "Send Msg";
00170      this.btnSend.UseVisualStyleBackColor = true;
00171      this.btnSend.Visible = false;
00172      //
00173      // trayICON
00174      //
00175      this.trayICON.BalloonTipText = "Raising Hands is a thing of the
00175      past.";
00176      this.trayICON.BalloonTipTitle = "Classroom Inq.";
00177      this.trayICON.Icon = ((System.Drawing.Icon)(resources.GetObject("
00177      trayICON.Icon")));
00178      this.trayICON.Text = "Classroom Inq.";
00179      this.trayICON.Visible = true;
00180      this.trayICON.MouseDoubleClick += new System.Windows.Forms.
00180      MouseEventHandler(this.trayICON_MouseDoubleClick);
00181      //
00182      // grpbxFeed
00183      //
00184      this.grpbxFeed.BackColor = System.Drawing.SystemColors.Control;
00185      this.grpbxFeed.Controls.Add(this.vScrollBar1);
00186      this.grpbxFeed.Controls.Add(this.grpbx_Reply);
00187      this.grpbxFeed.Location = new System.Drawing.Point(12, 80);
```

```

00188     this.grpbxFeed.Name = "grpbxFeed";
00189     this.grpbxFeed.Size = new System.Drawing.Size(340, 432);
00190     this.grpbxFeed.TabIndex = 5;
00191     this.grpbxFeed.TabStop = false;
00192     //
00193     // vScrollBar1
00194     //
00195     this.vScrollBar1.Dock = System.Windows.Forms.DockStyle.Right;
00196     this.vScrollBar1.Location = new System.Drawing.Point(321, 16);
00197     this.vScrollBar1.Name = "vScrollBar1";
00198     this.vScrollBar1.Size = new System.Drawing.Size(16, 413);
00199     this.vScrollBar1.TabIndex = 3;
00200     this.vScrollBar1.Visible = false;
00201     //
00202     // grpbx_Reply
00203     //
00204     this.grpbx_Reply.BackColor = System.Drawing.SystemColors.
ControlDarkDark;
00205     this.grpbx_Reply.Controls.Add(this.btnRPL);
00206     this.grpbx_Reply.Controls.Add(this.btnCLS);
00207     this.grpbx_Reply.Controls.Add(this.btnCLR);
00208     this.grpbx_Reply.Controls.Add(this.txtbx_Reply);
00209     this.grpbx_Reply.Controls.Add(this.lbl_question);
00210     this.grpbx_Reply.ForeColor = System.Drawing.SystemColors.
HighlightText;
00211     this.grpbx_Reply.Location = new System.Drawing.Point(6, 19);
00212     this.grpbx_Reply.Name = "grpbx_Reply";
00213     this.grpbx_Reply.Size = new System.Drawing.Size(328, 32);
00214     this.grpbx_Reply.TabIndex = 2;
00215     this.grpbx_Reply.TabStop = false;
00216     //
00217     // btnRPL
00218     //
00219     this.btnRPL.Location = new System.Drawing.Point(230, 98);
00220     this.btnRPL.Name = "btnRPL";
00221     this.btnRPL.Size = new System.Drawing.Size(67, 25);
00222     this.btnRPL.TabIndex = 4;
00223     this.btnRPL.Text = "Repl&y";
00224     this.btnRPL.UseVisualStyleBackColor = true;
00225     //
00226     // btnCLS
00227     //
00228     this.btnCLS.Location = new System.Drawing.Point(129, 98);
00229     this.btnCLS.Name = "btnCLS";
00230     this.btnCLS.Size = new System.Drawing.Size(67, 25);
00231     this.btnCLS.TabIndex = 3;
00232     this.btnCLS.Text = "Clos&e";
00233     this.btnCLS.UseVisualStyleBackColor = true;
00234     this.btnCLS.Click += new System.EventHandler(this.btnCLS_Click);
00235     //
00236     // btnCLR
00237     //
00238     this.btnCLR.Location = new System.Drawing.Point(23, 98);
00239     this.btnCLR.Name = "btnCLR";
00240     this.btnCLR.Size = new System.Drawing.Size(67, 25);
00241     this.btnCLR.TabIndex = 2;
00242     this.btnCLR.Text = "Clea&r";
00243     this.btnCLR.UseVisualStyleBackColor = true;
00244     this.btnCLR.Click += new System.EventHandler(this.btnCLR_Click);
00245     //
00246     // txtbx_Reply
00247     //
00248     this.txtbx_Reply.AcceptsReturn = true;
00249     this.txtbx_Reply.Location = new System.Drawing.Point(23, 32);
00250     this.txtbx_Reply.Multiline = true;
00251     this.txtbx_Reply.Name = "txtbx_Reply";
00252     this.txtbx_Reply.Size = new System.Drawing.Size(274, 51);
00253     this.txtbx_Reply.TabIndex = 1;
00254     //
00255     // lbl_question
00256     //
00257     this.lbl_question.AutoSize = true;
00258     this.lbl_question.Location = new System.Drawing.Point(62, 13);
00259     this.lbl_question.Name = "lbl_question";

```

```
00260         this.lbl_question.Size = new System.Drawing.Size(191, 13);
00261         this.lbl_question.TabIndex = 0;
00262         this.lbl_question.Text = "Why is there evil in the world??\t-Joseph
";
00263         this.lbl_question.Click += new System.EventHandler(this.
lbl_question_Click);
00264         // 
00265         // Form1
00266         //
00267         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
00268         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
00269         this.BackColor = System.Drawing.SystemColors.Control;
00270         this.ClientSize = new System.Drawing.Size(364, 550);
00271         this.ControlBox = false;
00272         this.Controls.Add(this.grpbxFeed);
00273         this.Controls.Add(this.btnSend);
00274         this.Controls.Add(this.statusStrip1);
00275         this.Controls.Add(this.textBox1);
00276         this.Controls.Add(this.menuStrip1);
00277         this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.
FixedSingle;
00278         this.Icon = ((System.Drawing.Icon)(resources.GetObject("$this.Icon"
)));
00279         this.MainMenuStrip = this.menuStrip1;
00280         this.MaximizeBox = false;
00281         this.Name = "Form1";
00282         this.Text = " Classroom Inquisition";
00283         this.Load += new System.EventHandler(this.Form1_Load);
00284         this.menuStrip1.ResumeLayout(false);
00285         this.menuStrip1.PerformLayout();
00286         this.statusStrip1.ResumeLayout(false);
00287         this.statusStrip1.PerformLayout();
00288         this.grpbx_Feed.ResumeLayout(false);
00289         this.grpbx_Feed.PerformLayout();
00290         this.grpbx_Reply.ResumeLayout(false);
00291         this.grpbx_Reply.PerformLayout();
00292         this.ResumeLayout(false);
00293     }
00294 }
00295 #endregion
00296
00297 private System.Windows.Forms.MenuStrip menuStrip1;
00298 private System.Windows.Forms.ToolStripMenuItem sendToolStripMenuItem;
00299 private System.Windows.Forms.ToolStripMenuItem broadcastToolStripMenuItem;
00300 private System.Windows.Forms.ToolStripMenuItem directMessageToolStripMenuItem;
00301 private System.Windows.Forms.ToolStripMenuItem replyToolStripMenuItem;
00302 private System.Windows.Forms.ToolStripMenuItem viewToolStripMenuItem;
00303 private System.Windows.Forms.ToolStripMenuItem helpToolStripMenuItem;
00304 private System.Windows.Forms.TextBox textBox1;
00305 private System.Windows.Forms.StatusStrip statusStrip1;
00306 private System.Windows.Forms.ToolStripStatusLabel sb_send_status;
00307 private System.Windows.Forms.ToolStripStatusLabel controlsToolStripMenuItem;
00308 ;
00309 private System.Windows.Forms.ToolStripMenuItem quitToolStripMenuItem;
00310 private System.Windows.Forms.Timer timer;
00311 private System.Windows.Forms.Button btnSend;
00312 private System.Windows.Forms.NotifyIcon trayICON;
00313 private System.Windows.Forms.GroupBox grpbxFeed;
00314 private System.Windows.Forms.GroupBox grpbx_Reply;
00315 private System.Windows.Forms.Button btnCLR;
00316 private System.Windows.Forms.TextBox txtbx_Reply;
00317 private System.Windows.Forms.Label lbl_question;
00318 private System.Windows.Forms.Button btnRPL;
00319 private System.Windows.Forms.Button btnCLS;
00320 private System.Windows.Forms.VScrollBar vScrollBar1;
00321 }
00322 }
00323 }
```

6.15 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Program.cs File Reference

Classes

- class [SrP_ClassroomInq::Program](#)

Packages

- package [SrP_ClassroomInq](#)

Variables

- using [System](#)

6.15.1 Variable Documentation

6.15.1.1 using System

Definition at line 18 of file [Program.cs](#).

6.16 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Program.cs

```

00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 ****/
00019 using System;
00020 using System.Collections.Generic;
00021 using System.Linq;
00022 using System.Windows.Forms;
00023 namespace SrP_ClassroomInq
00024 {
00025     static class Program
00026     {
00030         [STAThread]

```

```
00031     static void Main()
00032     {
00033         Application.EnableVisualStyles();
00034         Application.SetCompatibleTextRenderingDefault(false);
00035         Application.Run(new frmClassroomInq());
00036     }
00037 }
00038 }
```

6.17 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/AssemblyInfo.cs - File Reference

6.18 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/AssemblyInfo.cs

```
00001 using System.Reflection;
00002 using System.Runtime.CompilerServices;
00003 using System.Runtime.InteropServices;
00004
00005 // General Information about an assembly is controlled through the following
00006 // set of attributes. Change these attribute values to modify the information
00007 // associated with an assembly.
00008 [assembly: AssemblyTitle("ClassroomInq")]
00009 [assembly: AssemblyDescription("")]
00010 [assembly: AssemblyConfiguration("")]
00011 [assembly: AssemblyCompany("Classroom Inquistion")]
00012 [assembly: AssemblyProduct("ClassroomInq")]
00013 [assembly: AssemblyCopyright("Copyright © David Christensen 2012")]
00014 [assembly: AssemblyTrademark("")]
00015 [assembly: AssemblyCulture("")]
00016
00017 // Setting ComVisible to false makes the types in this assembly not visible
00018 // to COM components. If you need to access a type in this assembly from
00019 // COM, set the ComVisible attribute to true on that type.
00020 [assembly: ComVisible(false)]
00021
00022 // The following GUID is for the ID of the typelib if this project is exposed
00023 // to COM
00024 [assembly: Guid("44f847c1-0eed-4f4c-90fe-a036fd556b8c")]
00025
00026 // Version information for an assembly consists of the following four values:
00027 // Major Version
00028 // Minor Version
00029 // Build Number
00030 // Revision
00031 //
00032 // You can specify all the values or you can default the Build and Revision
00033 // Numbers
00034 // by using the '*' as shown below:
00035 [assembly: AssemblyVersion("1.0.*")]
00036 [assembly: AssemblyFileVersion("1.7.0.0")]
00037 [assembly: AssemblyFileVersion("1.7.3.0")]
```

6.19 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/Resources.Designer.cs File Reference

Classes

- class [SrP_ClassroomInq::Properties::Resources](#)
A strongly-typed resource class, for looking up localized strings, etc.

Packages

- package [SrP_ClassroomInq.Properties](#)

6.20 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/Resources.Designer.cs

```

00001 // -----
00002 // <auto-generated>
00003 //   This code was generated by a tool.
00004 //   Runtime Version:4.0.30319.239
00005 //
00006 //   Changes to this file may cause incorrect behavior and will be lost if
00007 //   the code is regenerated.
00008 // </auto-generated>
00009 //

00010 -----
00011 namespace SrP_ClassroomInq.Properties {
00012     using System;
00013
00014
00015     // This class was auto-generated by the StronglyTypedResourceBuilder
00016     // class via a tool like ResGen or Visual Studio.
00017     // To add or remove a member, edit your .ResX file then rerun ResGen
00018     // with the /str option, or rebuild your VS project.
00019     [global::System.CodeDom.Compiler.GeneratedCodeAttribute(
00020         "System.Resources.Tools.StronglyTypedResourceBuilder", "4.0.0.0")]
00021     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00022     [global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
00023     internal class Resources {
00024
00025         private static global::System.Resources.ResourceManager resourceMan;
00026
00027         private static global::System.Globalization.CultureInfo resourceCulture
00028 ;
00029
00030
00031         [global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute(""
00032             Microsoft.Performance", "CA1811:AvoidUncalledPrivateCode")]
00033         internal Resources() {
00034
00035
00036         [global::System.ComponentModel.EditorBrowsableAttribute(global::System.
00037             ComponentModel.EditorBrowsableState.Advanced)]
00038         internal static global::System.Resources.ResourceManager
00039             ResourceManager {
00040                 get {
00041                     if (object.ReferenceEquals(resourceMan, null)) {
00042                         global::System.Resources.ResourceManager temp = new global
00043                             ::System.Resources.ResourceManager("SrP_ClassroomInq.Properties.Resources",

```

```
00043     typeof(Resources).Assembly);
00044         resourceMan = temp;
00045     }
00046 }
00047 }
00048
00049 [global::System.ComponentModel.EditorBrowsableAttribute(global::System.
00050 ComponentModel.EditorBrowsableState.Advanced)]
00051     internal static global::System.Globalization.CultureInfo Culture {
00052     get {
00053         return resourceCulture;
00054     }
00055     set {
00056         resourceCulture = value;
00057     }
00058 }
00059
00060     internal static System.Drawing.Bitmap Background {
00061     get {
00062         object obj = ResourceManager.GetObject("Background",
00063         resourceCulture);
00064         return ((System.Drawing.Bitmap)(obj));
00065     }
00066 }
00067
00068     internal static System.Drawing.Bitmap bubbles2 {
00069     get {
00070         object obj = ResourceManager.GetObject("bubbles2",
00071         resourceCulture);
00072         return ((System.Drawing.Bitmap)(obj));
00073     }
00074 }
00075
00076     internal static System.Drawing.Bitmap comment_bubble {
00077     get {
00078         object obj = ResourceManager.GetObject("comment_bubble",
00079         resourceCulture);
00080         return ((System.Drawing.Bitmap)(obj));
00081     }
00082 }
00083
00084     internal static System.Drawing.Bitmap reply_arrow {
00085     get {
00086         object obj = ResourceManager.GetObject("reply_arrow",
00087         resourceCulture);
00088         return ((System.Drawing.Bitmap)(obj));
00089     }
00090 }
00091     internal static System.Drawing.Bitmap Splash {
00092     get {
00093         object obj = ResourceManager.GetObject("Splash",
00094         resourceCulture);
00095         return ((System.Drawing.Bitmap)(obj));
00096     }
00097 }
00098     internal static System.Drawing.Bitmap StuMngmntCtr {
00099     get {
00100         object obj = ResourceManager.GetObject("StuMngmntCtr",
00101         resourceCulture);
00102         return ((System.Drawing.Bitmap)(obj));
00103     }
00104 }
00105     internal static System.Drawing.Bitmap user_comment {
00106     get {
00107         object obj = ResourceManager.GetObject("user_comment",
00108         resourceCulture);
00109         return ((System.Drawing.Bitmap)(obj));
00110     }
00111 }
```

```
00112 }
```

6.21 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/Settings.Designer.cs

File Reference

Classes

- class [SrP_ClassroomInq::Properties::Settings](#)

Packages

- package [SrP_ClassroomInq.Properties](#)

6.22 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Properties/Settings.Designer.cs

```
00001 // -----
00002 // <auto-generated>
00003 //     This code was generated by a tool.
00004 //     Runtime Version:4.0.30319.239
00005 //
00006 //     Changes to this file may cause incorrect behavior and will be lost if
00007 //     the code is regenerated.
00008 // </auto-generated>
00009 //

-----
00010
00011 namespace SrP_ClassroomInq.Properties {
00012
00013
00014     [global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
00015     [global::System.CodeDom.Compiler.GeneratedCodeAttribute(
00016         "Microsoft.VisualStudioEditors.SettingsDesigner.SettingsSingleFileGenerator", "10.0.0.0")]
00017     internal sealed partial class Settings : global::System.Configuration.
00018         ApplicationSettingsBase {
00019
00020     private static Settings defaultInstance = ((Settings)(global::System.
00021         Configuration.ApplicationSettingsBase.Synchronized(new Settings())));
00022
00023     public static Settings Default {
00024         get {
00025             return defaultInstance;
00026         }
00027     }
00028
00029     [global::System.Configuration.UserScopedSettingAttribute()]
00030     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00031     [global::System.Configuration.DefaultSettingValueAttribute("this-isn\'t-secure")]
00032     public string key {
00033         get {
00034             return ((string)(this["key"]));
00035         }
00036         set {
00037             this["key"] = value;
00038         }
00039     }
00040 }
```

```
00035         }
00036     }
00037
00038     [global::System.Configuration.UserScopedSettingAttribute()]
00039     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00040     [global::System.Configuration.DefaultSettingValueAttribute("False")]
00041     public bool Animations {
00042         get {
00043             return ((bool)(this["Animations"]));
00044         }
00045         set {
00046             this["Animations"] = value;
00047         }
00048     }
00049
00050     [global::System.Configuration.UserScopedSettingAttribute()]
00051     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00052     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00053     public bool SoundRX {
00054         get {
00055             return ((bool)(this["SoundRX"]));
00056         }
00057         set {
00058             this["SoundRX"] = value;
00059         }
00060     }
00061
00062     [global::System.Configuration.UserScopedSettingAttribute()]
00063     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00064     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00065     public bool SoundTX {
00066         get {
00067             return ((bool)(this["SoundTX"]));
00068         }
00069         set {
00070             this["SoundTX"] = value;
00071         }
00072     }
00073
00074     [global::System.Configuration.UserScopedSettingAttribute()]
00075     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00076     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00077     public bool HoverRead {
00078         get {
00079             return ((bool)(this["HoverRead"]));
00080         }
00081         set {
00082             this["HoverRead"] = value;
00083         }
00084     }
00085
00086     [global::System.Configuration.UserScopedSettingAttribute()]
00087     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00088     [global::System.Configuration.DefaultSettingValueAttribute("False")]
00089     public bool ClickRead {
00090         get {
00091             return ((bool)(this["ClickRead"]));
00092         }
00093         set {
00094             this["ClickRead"] = value;
00095         }
00096     }
00097
00098     [global::System.Configuration.UserScopedSettingAttribute()]
00099     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00100     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00101     public bool Tooltips {
00102         get {
00103             return ((bool)(this["Tooltips"]));
00104         }
00105         set {
00106             this["Tooltips"] = value;
00107         }
00108     }
```

```

00109
00110     [global::System.Configuration.UserScopedSettingAttribute()]
00111     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00112     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00113     public bool ShowNames {
00114         get {
00115             return ((bool)(this["ShowNames"]));
00116         }
00117         set {
00118             this["ShowNames"] = value;
00119         }
00120     }
00121
00122     [global::System.Configuration.UserScopedSettingAttribute()]
00123     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00124     [global::System.Configuration.DefaultSettingValueAttribute("True")]
00125     public bool Notify {
00126         get {
00127             return ((bool)(this["Notify"]));
00128         }
00129         set {
00130             this["Notify"] = value;
00131         }
00132     }
00133
00134     [global::System.Configuration.UserScopedSettingAttribute()]
00135     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00136     [global::System.Configuration.DefaultSettingValueAttribute("False")]
00137     public bool CtrlHide {
00138         get {
00139             return ((bool)(this["CtrlHide"]));
00140         }
00141         set {
00142             this["CtrlHide"] = value;
00143         }
00144     }
00145
00146     [global::System.Configuration.UserScopedSettingAttribute()]
00147     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00148     [global::System.Configuration.DefaultSettingValueAttribute(""
00149         DarkGoldenrod")]
00150     public global::System.Drawing.Color ForeColorTheme {
00151         get {
00152             return ((global::System.Drawing.Color)(this["ForeColorTheme"]));
00153         }
00154         set {
00155             this["ForeColorTheme"] = value;
00156         }
00157     }
00158
00159     [global::System.Configuration.UserScopedSettingAttribute()]
00160     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00161     [global::System.Configuration.DefaultSettingValueAttribute("Black")]
00162     public global::System.Drawing.Color BackColorTheme {
00163         get {
00164             return ((global::System.Drawing.Color)(this["BackColorTheme"]));
00165         }
00166         set {
00167             this["BackColorTheme"] = value;
00168         }
00169     }
00170
00171     [global::System.Configuration.UserScopedSettingAttribute()]
00172     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00173     [global::System.Configuration.DefaultSettingValueAttribute("24")]
00174     public byte AnimationSpeed {
00175         get {
00176             return ((byte)(this["AnimationSpeed"]));
00177         }
00178         set {
00179             this["AnimationSpeed"] = value;
00180         }
00181     }

```

```
00180      }
00181
00182      [global::System.Configuration.UserScopedSettingAttribute()]
00183      [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00184      [global::System.Configuration.DefaultSettingValueAttribute("1.6")]
00185      public decimal NotifyTimeout {
00186          get {
00187              return ((decimal)(this["NotifyTimeout"]));
00188          }
00189          set {
00190              this["NotifyTimeout"] = value;
00191          }
00192      }
00193
00194      [global::System.Configuration.UserScopedSettingAttribute()]
00195      [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
00196      [global::System.Configuration.DefaultSettingValueAttribute("True")]
00197      public bool Splash {
00198          get {
00199              return ((bool)(this["Splash"]));
00200          }
00201          set {
00202              this["Splash"] = value;
00203          }
00204      }
00205  }
00206 }
```

6.23 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Splash.cs File Reference

Classes

- class [SrP_ClassroomInq::Splash](#)

Packages

- package [SrP_ClassroomInq](#)

Variables

- using [System](#)

6.23.1 Variable Documentation

6.23.1.1 using System

Definition at line 18 of file [Splash.cs](#).

6.24 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Splash.cs

```

00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 *****/
00019 using System;
00020 using System.Collections.Generic;
00021 using System.ComponentModel;
00022 using System.Data;
00023 using System.Drawing;
00024 using System.Linq;
00025 using System.Text;
00026 using System.Windows.Forms;
00027
00028 namespace SrP_ClassroomInq
00029 {
00030     public partial class Splash : Form
00031     {
00032         public Splash()
00033         {
00034             InitializeComponent();
00035         }
00036         byte i = 0;
00037
00038         private void FatherTime_Tick(object sender, EventArgs e)
00039         {
00040             if (i < 3)
00041             {
00042                 //tick is once a second
00043                 i++;
00044             }
00045             else if (i < 10)
00046             {
00047                 if (i == 3)
00048                 {
00049                     FatherTime.Interval = 100; //change to 10 times faster
00050                 }
00051                 this.Opacity = this.Opacity - (double)0.1; //fade out by
00052                 percent
00053                 i++;
00054             }
00055             else
00056             {
00057                 this.DialogResult = System.Windows.Forms.DialogResult.OK; //tell the main form we're good to go
00058             }
00059         }
00060     }
00061 }
```

6.25 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition-/SrP_ClassroomInq/Splash.Designer.cs File

Reference **215**
6.25 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Splash.Designer.cs File - Reference

Variables

- namespace [SrP_ClassroomInq](#)

6.25.1 Variable Documentation

6.25.1.1 namespace SrP_ClassroomInq

Definition at line 19 of file [Splash.Designer.cs](#).

6.26 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/Splash.Designer.cs

```
00001 /*****
00002 * Copyright (C) 2012 David V. Christensen
00003 *
00004 * This program is free software; you can redistribute it and/or
00005 * modify it under the terms of the GNU General Public License
00006 * as published by the Free Software Foundation; either version 2
00007 * of the License, or (at your option) any later version.
00008 *
00009 * This program is distributed in the hope that it will be useful,
00010 * but WITHOUT ANY WARRANTY; without even the implied warranty of
00011 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00012 * GNU General Public License for more details.
00013 *
00014 * You should have received a copy of the GNU General Public License
00015 * along with this program; if not, write to the Free Software
00016 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00017 * USA.
00018 *****/
00019 namespace SrP_ClassroomInq
00020 {
00021     partial class Splash
00022     {
00023         private System.ComponentModel.IContainer components = null;
00024
00025         protected override void Dispose(bool disposing)
00026         {
00027             if (disposing && (components != null))
00028             {
00029                 components.Dispose();
00030             }
00031             base.Dispose(disposing);
00032         }
00033
00034         #region Windows Form Designer generated code
00035
00036         private void InitializeComponent()
00037         {
00038             this.components = new System.ComponentModel.Container();
00039             this.lblVersion = new System.Windows.Forms.Label();
00040             this.FatherTime = new System.Windows.Forms.Timer(this.components);
00041
00042             ...
00043
00044         }
00045
00046         #endregion
00047
00048         private void Splash_Load(object sender, EventArgs e)
00049         {
00050             ...
00051
00052             ...
00053
00054             ...
00055
00056             ...
00057
00058             ...
00059
00060             ...
00061
00062             ...
00063
00064             ...
00065
00066             ...
00067
00068             ...
00069
00070             ...
00071
00072             ...
00073
00074             ...
00075
00076             ...
00077
00078             ...
00079
00080             ...
00081
00082             ...
00083
00084             ...
00085
00086             ...
00087
00088             ...
00089
00090             ...
00091
00092             ...
00093
00094             ...
00095
00096             ...
00097
00098             ...
00099
00100             ...
00101
00102             ...
00103
00104             ...
00105
00106             ...
00107
00108             ...
00109
00110             ...
00111
00112             ...
00113
00114             ...
00115
00116             ...
00117
00118             ...
00119
00120             ...
00121
00122             ...
00123
00124             ...
00125
00126             ...
00127
00128             ...
00129
00130             ...
00131
00132             ...
00133
00134             ...
00135
00136             ...
00137
00138             ...
00139
00140             ...
00141
00142             ...
00143
00144             ...
00145
00146             ...
00147
00148             ...
00149
00150             ...
00151
00152             ...
00153
00154             ...
00155
00156             ...
00157
00158             ...
00159
00160             ...
00161
00162             ...
00163
00164             ...
00165
00166             ...
00167
00168             ...
00169
00170             ...
00171
00172             ...
00173
00174             ...
00175
00176             ...
00177
00178             ...
00179
00180             ...
00181
00182             ...
00183
00184             ...
00185
00186             ...
00187
00188             ...
00189
00190             ...
00191
00192             ...
00193
00194             ...
00195
00196             ...
00197
00198             ...
00199
00200             ...
00201
00202             ...
00203
00204             ...
00205
00206             ...
00207
00208             ...
00209
00210             ...
00211
00212             ...
00213
00214             ...
00215
00216             ...
00217
00218             ...
00219
00220             ...
00221
00222             ...
00223
00224             ...
00225
00226             ...
00227
00228             ...
00229
00230             ...
00231
00232             ...
00233
00234             ...
00235
00236             ...
00237
00238             ...
00239
00240             ...
00241
00242             ...
00243
00244             ...
00245
00246             ...
00247
00248             ...
00249
00250             ...
00251
00252             ...
00253
00254             ...
00255
00256             ...
00257
00258             ...
00259
00260             ...
00261
00262             ...
00263
00264             ...
00265
00266             ...
00267
00268             ...
00269
00270             ...
00271
00272             ...
00273
00274             ...
00275
00276             ...
00277
00278             ...
00279
00280             ...
00281
00282             ...
00283
00284             ...
00285
00286             ...
00287
00288             ...
00289
00290             ...
00291
00292             ...
00293
00294             ...
00295
00296             ...
00297
00298             ...
00299
00300             ...
00301
00302             ...
00303
00304             ...
00305
00306             ...
00307
00308             ...
00309
00310             ...
00311
00312             ...
00313
00314             ...
00315
00316             ...
00317
00318             ...
00319
00320             ...
00321
00322             ...
00323
00324             ...
00325
00326             ...
00327
00328             ...
00329
00330             ...
00331
00332             ...
00333
00334             ...
00335
00336             ...
00337
00338             ...
00339
00340             ...
00341
00342             ...
00343
00344             ...
00345
00346             ...
00347
00348             ...
00349
00350             ...
00351
00352             ...
00353
00354             ...
00355
00356             ...
00357
00358             ...
00359
00360             ...
00361
00362             ...
00363
00364             ...
00365
00366             ...
00367
00368             ...
00369
00370             ...
00371
00372             ...
00373
00374             ...
00375
00376             ...
00377
00378             ...
00379
00380             ...
00381
00382             ...
00383
00384             ...
00385
00386             ...
00387
00388             ...
00389
00390             ...
00391
00392             ...
00393
00394             ...
00395
00396             ...
00397
00398             ...
00399
00400             ...
00401
00402             ...
00403
00404             ...
00405
00406             ...
00407
00408             ...
00409
00410             ...
00411
00412             ...
00413
00414             ...
00415
00416             ...
00417
00418             ...
00419
00420             ...
00421
00422             ...
00423
00424             ...
00425
00426             ...
00427
00428             ...
00429
00430             ...
00431
00432             ...
00433
00434             ...
00435
00436             ...
00437
00438             ...
00439
00440             ...
00441
00442             ...
00443
00444             ...
00445
00446             ...
00447
00448             ...
00449
00450             ...
00451
00452             ...
00453
00454             ...
00455
00456             ...
00457
00458             ...
00459
00460             ...
00461
00462             ...
00463
00464             ...
00465
00466             ...
00467
00468             ...
00469
00470             ...
00471
00472             ...
00473
00474             ...
00475
00476             ...
00477
00478             ...
00479
00480             ...
00481
00482             ...
00483
00484             ...
00485
00486             ...
00487
00488             ...
00489
00490             ...
00491
00492             ...
00493
00494             ...
00495
00496             ...
00497
00498             ...
00499
00500             ...
00501
00502             ...
00503
00504             ...
00505
00506             ...
00507
00508             ...
00509
00510             ...
00511
00512             ...
00513
00514             ...
00515
00516             ...
00517
00518             ...
00519
00520             ...
00521
00522             ...
00523
00524             ...
00525
00526             ...
00527
00528             ...
00529
00530             ...
00531
00532             ...
00533
00534             ...
00535
00536             ...
00537
00538             ...
00539
00540             ...
00541
00542             ...
00543
00544             ...
00545
00546             ...
00547
00548             ...
00549
00550             ...
00551
00552             ...
00553
00554             ...
00555
00556             ...
00557
00558             ...
00559
00560             ...
00561
00562             ...
00563
00564             ...
00565
00566             ...
00567
00568             ...
00569
00570             ...
00571
00572             ...
00573
00574             ...
00575
00576             ...
00577
00578             ...
00579
00580             ...
00581
00582             ...
00583
00584             ...
00585
00586             ...
00587
00588             ...
00589
00590             ...
00591
00592             ...
00593
00594             ...
00595
00596             ...
00597
00598             ...
00599
00600             ...
00601
00602             ...
00603
00604             ...
00605
00606             ...
00607
00608             ...
00609
00610             ...
00611
00612             ...
00613
00614             ...
00615
00616             ...
00617
00618             ...
00619
00620             ...
00621
00622             ...
00623
00624             ...
00625
00626             ...
00627
00628             ...
00629
00630             ...
00631
00632             ...
00633
00634             ...
00635
00636             ...
00637
00638             ...
00639
00640             ...
00641
00642             ...
00643
00644             ...
00645
00646             ...
00647
00648             ...
00649
00650             ...
00651
00652             ...
00653
00654             ...
00655
00656             ...
00657
00658             ...
00659
00660             ...
00661
00662             ...
00663
00664             ...
00665
00666             ...
00667
00668             ...
00669
00670             ...
00671
00672             ...
00673
00674             ...
00675
00676             ...
00677
00678             ...
00679
00680             ...
00681
00682             ...
00683
00684             ...
00685
00686             ...
00687
00688             ...
00689
00690             ...
00691
00692             ...
00693
00694             ...
00695
00696             ...
00697
00698             ...
00699
00700             ...
00701
00702             ...
00703
00704             ...
00705
00706             ...
00707
00708             ...
00709
00710             ...
00711
00712             ...
00713
00714             ...
00715
00716             ...
00717
00718             ...
00719
00720             ...
00721
00722             ...
00723
00724             ...
00725
00726             ...
00727
00728             ...
00729
00730             ...
00731
00732             ...
00733
00734             ...
00735
00736             ...
00737
00738             ...
00739
00740             ...
00741
00742             ...
00743
00744             ...
00745
00746             ...
00747
00748             ...
00749
00750             ...
00751
00752             ...
00753
00754             ...
00755
00756             ...
00757
00758             ...
00759
00760             ...
00761
00762             ...
00763
00764             ...
00765
00766             ...
00767
00768             ...
00769
00770             ...
00771
00772             ...
00773
00774             ...
00775
00776             ...
00777
00778             ...
00779
00780             ...
00781
00782             ...
00783
00784             ...
00785
00786             ...
00787
00788             ...
00789
00790             ...
00791
00792             ...
00793
00794             ...
00795
00796             ...
00797
00798             ...
00799
00800             ...
00801
00802             ...
00803
00804             ...
00805
00806             ...
00807
00808             ...
00809
00810             ...
00811
00812             ...
00813
00814             ...
00815
00816             ...
00817
00818             ...
00819
00820             ...
00821
00822             ...
00823
00824             ...
00825
00826             ...
00827
00828             ...
00829
00830             ...
00831
00832             ...
00833
00834             ...
00835
00836             ...
00837
00838             ...
00839
00840             ...
00841
00842             ...
00843
00844             ...
00845
00846             ...
00847
00848             ...
00849
00850             ...
00851
00852             ...
00853
00854             ...
00855
00856             ...
00857
00858             ...
00859
00860             ...
00861
00862             ...
00863
00864             ...
00865
00866             ...
00867
00868             ...
00869
00870             ...
00871
00872             ...
00873
00874             ...
00875
00876             ...
00877
00878             ...
00879
00880             ...
00881
00882             ...
00883
00884             ...
00885
00886             ...
00887
00888             ...
00889
00890             ...
00891
00892             ...
00893
00894             ...
00895
00896             ...
00897
00898             ...
00899
00900             ...
00901
00902             ...
00903
00904             ...
00905
00906             ...
00907
00908             ...
00909
00910             ...
00911
00912             ...
00913
00914             ...
00915
00916             ...
00917
00918             ...
00919
00920             ...
00921
00922             ...
00923
00924             ...
00925
00926             ...
00927
00928             ...
00929
00930             ...
00931
00932             ...
00933
00934             ...
00935
00936             ...
00937
00938             ...
00939
00940             ...
00941
00942             ...
00943
00944             ...
00945
00946             ...
00947
00948             ...
00949
00950             ...
00951
00952             ...
00953
00954             ...
00955
00956             ...
00957
00958             ...
00959
00960             ...
00961
00962             ...
00963
00964             ...
00965
00966             ...
00967
00968             ...
00969
00970             ...
00971
00972             ...
00973
00974             ...
00975
00976             ...
00977
00978             ...
00979
00980             ...
00981
00982             ...
00983
00984             ...
00985
00986             ...
00987
00988             ...
00989
00990             ...
00991
00992             ...
00993
00994             ...
00995
00996             ...
00997
00998             ...
00999
01000             ...
01001
01002             ...
01003
01004             ...
01005
01006             ...
01007
01008             ...
01009
01010             ...
01011
01012             ...
01013
01014             ...
01015
01016             ...
01017
01018             ...
01019
01020             ...
01021
01022             ...
01023
01024             ...
01025
01026             ...
01027
01028             ...
01029
01030             ...
01031
01032             ...
01033
01034             ...
01035
01036             ...
01037
01038             ...
01039
01040             ...
01041
01042             ...
01043
01044             ...
01045
01046             ...
01047
01048             ...
01049
01050             ...
01051
01052             ...
01053
01054             ...
01055
01056             ...
01057
01058             ...
01059
01060             ...
01061
01062             ...
01063
01064             ...
01065
01066             ...
01067
01068             ...
01069
01070             ...
01071
01072             ...
01073
01074             ...
01075
01076             ...
01077
01078             ...
01079
01080             ...
01081
01082             ...
01083
01084             ...
01085
01086             ...
01087
01088             ...
01089
01090             ...
01091
01092             ...
01093
01094             ...
01095
01096             ...
01097
01098             ...
01099
01100             ...
01101
01102             ...
01103
01104             ...
01105
01106             ...
01107
01108             ...
01109
01110             ...
01111
01112             ...
01113
01114             ...
01115
01116             ...
01117
01118             ...
01119
01120             ...
01121
01122             ...
01123
01124             ...
01125
01126             ...
01127
01128             ...
01129
01130             ...
01131
01132             ...
01133
01134             ...
01135
01136             ...
01137
01138             ...
01139
01140             ...
01141
01142             ...
01143
01144             ...
01145
01146             ...
01147
01148             ...
01149
01150             ...
01151
01152             ...
01153
01154             ...
01155
01156             ...
01157
01158             ...
01159
01160             ...
01161
01162             ...
01163
01164             ...
01165
01166             ...
01167
01168             ...
01169
01170             ...
01171
01172             ...
01173
01174             ...
01175
01176             ...
01177
01178             ...
01179
01180             ...
01181
01182             ...
01183
01184             ...
01185
01186             ...
01187
01188             ...
01189
01190             ...
01191
01192             ...
01193
01194             ...
01195
01196             ...
01197
01198             ...
01199
01200             ...
01201
01202             ...
01203
01204             ...
01205
01206             ...
01207
01208             ...
01209
01210             ...
01211
01212             ...
01213
01214             ...
01215
01216             ...
01217
01218             ...
01219
01220             ...
01221
01222             ...
01223
01224             ...
01225
01226             ...
01227
01228             ...
01229
01230             ...
01231
01232             ...
01233
01234             ...
01235
01236             ...
01237
01238             ...
01239
01240             ...
01241
01242             ...
01243
01244             ...
01245
01246             ...
01247
01248             ...
01249
01250             ...
01251
01252             ...
01253
01254             ...
01255
01256             ...
01257
01258             ...
01259
01260             ...
01261
01262             ...
01263
01264             ...
01265
01266             ...
01267
01268             ...
01269
01270             ...
01271
01272             ...
01273
01274             ...
01275
01276             ...
01277
01278             ...
01279
01280             ...
01281
01282             ...
01283
01284             ...
01285
01286             ...
01287
01288             ...
01289
01290             ...
01291
01292             ...
01293
01294             ...
01295
01296             ...
01297
01298             ...
01299
01300             ...
01301
01302             ...
01303
01304             ...
01305
01306             ...
01307
01308             ...
01309
01310             ...
01311
01312             ...
01313
01314             ...
01315
01316             ...
01317
01318             ...
01319
01320             ...
01321
01322             ...
01323
01324             ...
01325
01326             ...
01327
01328             ...
01329
01330             ...
01331
01332             ...
01333
01334             ...
01335
01336             ...
01337
01338             ...
01339
01340             ...
01341
01342             ...
01343
01344             ...
01345
01346             ...
01347
01348             ...
01349
01350             ...
01351
01352             ...
01353
01354             ...
01355
01356             ...
01357
01358             ...
01359
01360             ...
01361
01362             ...
01363
01364             ...
01365
01366             ...
01367
01368             ...
01369
01370             ...
01371
01372             ...
01373
01374             ...
01375
01376             ...
01377
01378             ...
01379
01380             ...
01381
01382             ...
01383
01384             ...
01385
01386             ...
01387
01388             ...
01389
01390             ...
01391
01392             ...
01393
01394             ...
01395
01396             ...
01397
01398             ...
01399
01400             ...
01401
01402             ...
01403
01404             ...
01405
01406             ...
01407
01408             ...
01409
01410             ...
01411
01412             ...
01413
01414             ...
01415
01416             ...
01417
01418             ...
01419
01420             ...
01421
01422             ...
01423
01424             ...
01425
01426             ...
01427
01428             ...
01429
01430             ...
01431
01432             ...
01433
01434             ...
01435
01436             ...
01437
01438             ...
01439
01440             ...
01441
01442             ...
01443
01444             ...
01445
01446             ...
01447
01448             ...
01449
01450             ...
01451
01452             ...
01453
01454             ...
01455
01456             ...
01457
01458             ...
01459
01460             ...
01461
01462             ...
01463
01464             ...
01465
01466             ...
01467
01468             ...
01469
01470             ...
01471
01472             ...
01473
01474             ...
01475
01476             ...
01477
01478             ...
01479
01480             ...
01481
01482             ...
01483
01484             ...
01485
01486             ...
01487
01488             ...
01489
01490             ...
01491
01492             ...
01493
01494             ...
01495
01496             ...
01497
01498             ...
01499
01500             ...
01501
01502             ...
01503
01504             ...
01505
01506             ...
01507
01508             ...
01509
01510             ...
01511
01512             ...
01513
01514             ...
01515
01516             ...
01517
01518             ...
01519
01520             ...
01521
01522             ...
01523
01524             ...
01525
01526             ...
01527
01528             ...
01529
01530             ...
01531
01532             ...
01533
01534             ...
01535
01536             ...
01537
01538             ...
01539
01540             ...
01541
01542             ...
01543
01544             ...
01545
01546             ...
01547
01548             ...
01549
01550             ...
01551
01552             ...
01553
01554             ...
01555
01556             ...
01557
01558             ...
01559
01560             ...
01561
01562             ...
01563
01564             ...
01565
01566             ...
01567
01568             ...
01569
01570             ...
01571
01572             ...
01573
01574             ...
01575
01576             ...
01577
01578             ...
01579
01580             ...
01581
01582             ...
01583
01584             ...
01585
01586             ...
01587
01588             ...
01589
01590             ...
01591
01592             ...
01593
01594             ...
01595
01596             ...
01597
01598             ...
01599
01600             ...
01601
01602             ...
01603
01604             ...
0160
```

```

00051         this.SuspendLayout();
00052         //
00053         // lblVersion
00054         //
00055         this.lblVersion.Font = new System.Drawing.Font("Century Gothic", 24
F, System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
00056         this.lblVersion.ForeColor = System.Drawing.Color.DarkGoldenrod;
00057         this.lblVersion.Location = new System.Drawing.Point(152, 210);
00058         this.lblVersion.Name = "lblVersion";
00059         this.lblVersion.Size = new System.Drawing.Size(160, 71);
00060         this.lblVersion.TabIndex = 0;
00061         this.lblVersion.Text = "1.7.3";
00062         this.lblVersion.TextAlign = System.Drawing.ContentAlignment.
MiddleCenter;
00063         //
00064         // FatherTime
00065         //
00066         this.FatherTime.Enabled = true;
00067         this.FatherTime.Interval = 1000;
00068         this.FatherTime.Tick += new System.EventHandler(this.
FatherTime_Tick);
00069         //
00070         // Splash
00071         //
00072         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.None;
00073         this.BackColor = System.Drawing.Color.Black;
00074         this.BackgroundImage = global::SrP_ClassroomInq.Properties.
Resources.Splash;
00075         this.BackgroundImageLayout = System.Windows.Forms.ImageLayout.
Stretch;
00076         this.ClientSize = new System.Drawing.Size(465, 281);
00077         this.ControlBox = false;
00078         this.Controls.Add(this.lblVersion);
00079         this.DoubleBuffered = true;
00080         this.Font = new System.Drawing.Font("Mistral", 18F, System.Drawing.
FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(0)));
00081         this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None;
00082         this.Margin = new System.Windows.Forms.Padding(2, 3, 2, 3);
00083         this.MaximizeBox = false;
00084         this.MinimizeBox = false;
00085         this.Name = "Splash";
00086         this.Opacity = 0.9D;
00087         this.ShowIcon = false;
00088         this.ShowInTaskbar = false;
00089         this.StartPosition = System.Windows.Forms.FormStartPosition.
CenterScreen;
00090         this.Text = "ClassroomInq-Splash";
00091         this.TopMost = true;
00092         this.ResumeLayout(false);
00093     }
00094 }
00095 #
00096 #endregion
00097
00098     private System.Windows.Forms.Label lblVersion;
00099     private System.Windows.Forms.Timer FatherTime;
00100 }
00101 }
```

6.27 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/themes.cs File Reference

Classes

- class [SrP_ClassroomInq::frmClassrromInq](#)

Packages

- package SrP_ClassroomInq

6.28 /home/bondslaveofJesus/Documents/SourceCode/Classroom-Inquisition/SrP_ClassroomInq/themes.cs

```
00001 /*****
00002   *
00003   * Copyright (C) 2012 David V. Christensen
00004   *
00005   * This program is free software; you can redistribute it and/or
00006   * modify it under the terms of the GNU General Public License
00007   * as published by the Free Software Foundation; either version 2
00008   * of the License, or (at your option) any later version.
00009   *
00010   * This program is distributed in the hope that it will be useful,
00011   * but WITHOUT ANY WARRANTY; without even the implied warranty of
00012   * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00013   * GNU General Public License for more details.
00014   *
00015   * You should have received a copy of the GNU General Public License
00016   * along with this program; if not, write to the Free Software
00017   * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
00018   * USA.
00019   */
00020 using System;
00021 using System.Collections.Generic;
00022 using System.ComponentModel;
00023 using System.Data;
00024 using System.Drawing;
00025 using System.Drawing.Printing;
00026 using System.Linq;
00027 using System.Media;
00028 using System.Text;
00029 using System.Windows.Forms;
00030 using System.IO;
00031 using System.Security.Cryptography;
00032 using System.Runtime.InteropServices;
00033 namespace SrP_ClassroomInq
00034 {
00035     public partial class frmClassrromInq : Form
00036     {
00037         /**
00038         *
00039         * Classroom Inquisition Teacher Application
00040         * Programmer: David Christensen
00041         * Bug Tracking: https://github.com/dchriste/Classroom-Inquisition/
00042         *
00043         */
00044     /*This Method allows for the theming of all controls globally*/
00045     public void ThemeApply(Color Fore, Color Back)
00046     {
00047         this.Hide(); //hide while we repaint everything
00048         this.SuspendLayout();
00049
00050         #region Prefs Panel Controls
00051         btnForeColor.ForeColor = Fore;
00052         btnForeColor.BackColor = Back;
00053
00054         btnBkgrndColor.ForeColor = Fore;
00055     }
}
```

```

00056     btnBkgrndColor.BackColor = Back;
00057
00058     btnDefaultTheme.ForeColor = Fore;
00059     btnDefaultTheme.BackColor = Back;
00060
00061     btnPrefs_Cls.ForeColor = Fore;
00062     btnPrefs_Cls.BackColor = Back;
00063
00064     btnStuMgmt_Prefs.ForeColor = Fore;
00065     btnStuMgmt_Prefs.BackColor = Back;
00066
00067     btnQuizMaker.ForeColor = Fore;
00068     btnQuizMaker.BackColor = Back;
00069
00070     chkbxCtrlHide.ForeColor = Fore;
00071     chkbxCtrlHide.BackColor = Back;
00072
00073     chkbxLameMode.ForeColor = Fore;
00074     chkbxLameMode.BackColor = Back;
00075
00076     chkbxNotify.ForeColor = Fore;
00077     chkbxNotify.BackColor = Back;
00078
00079     chkbxRXSound.ForeColor = Fore;
00080     chkbxRXSound.BackColor = Back;
00081
00082     chkbxToolips.ForeColor = Fore;
00083     chkbxToolips.BackColor = Back;
00084
00085     chkbxTXSound.ForeColor = Fore;
00086     chkbxTXSound.BackColor = Back;
00087
00088     grpbxTheme.ForeColor = Fore;
00089     grpbxTheme.BackColor = Back;
00090
00091     grpbxUnread.ForeColor = Fore;
00092     grpbxUnread.BackColor = Back;
00093
00094     lblAnimationPrefs.BackColor = Back;
00095     lblAnimationPrefs.ForeColor = Fore;
00096
00097     lblNotifyTime.BackColor = Back;
00098     lblNotifyTime.ForeColor = Fore;
00099
00100     lblSerPt.BackColor = Back;
00101     lblSerPt.ForeColor = Fore;
00102
00103     if (Fore == Color.Lime)
00104     {
00105         cmbbxAnimationSpeed.BackColor = SystemColors.ControlDarkDark;
00106         cmbbxAnimationSpeed.ForeColor = Fore;
00107     }
00108     else
00109     {
00110         cmbbxAnimationSpeed.BackColor = Color.White;
00111         cmbbxAnimationSpeed.ForeColor = Color.Black;
00112     }
00113
00114     PanelPrefs.ForeColor = Fore;
00115     PanelPrefs.BackColor = Back;
00116 #endregion
00117
00118 #region Home Controls
00119 this.ForeColor = Fore;
00120 this.BackColor = Back;
00121
00122     if (Fore == Color.Lime)
00123     {
00124         textBox1.BackColor = SystemColors.ControlDarkDark;
00125         textBox1.ForeColor = Fore;
00126
00127         serialCOMcmbbx.BackColor = SystemColors.ControlDarkDark;
00128         serialCOMcmbbx.ForeColor = Fore;
00129     }

```

```
00130         else
00131     {
00132         textBox1.BackColor = Color.White;
00133         textBox1.ForeColor = Color.Black;
00134
00135         serialCOMcmbbx.BackColor = Color.White;
00136         serialCOMcmbbx.ForeColor = Color.Black;
00137     }
00138
00139     menuStrip1.BackColor = Fore; //for contrast between form back
00140     menuStrip1.ForeColor = Back;
00141
00142     viewToolStripMenuItem.BackColor = Fore;
00143     viewToolStripMenuItem.ForeColor = Back;
00144
00145     broadcastToolStripMenuItem.BackColor = Fore;
00146     broadcastToolStripMenuItem.ForeColor = Back;
00147
00148     directMessageToolStripMenuItem.BackColor = Fore;
00149     directMessageToolStripMenuItem.ForeColor = Back;
00150
00151     modesToolStripMenuItem.BackColor = Fore;
00152     modesToolStripMenuItem.ForeColor = Back;
00153
00154     showNamesToolStripMenuItem.BackColor = Fore;
00155     showNamesToolStripMenuItem.ForeColor = Back;
00156
00157     quizToolStripMenuItem.BackColor = Fore;
00158     quizToolStripMenuItem.ForeColor = Back;
00159
00160     normalToolStripMenuItem.BackColor = Fore;
00161     normalToolStripMenuItem.ForeColor = Back;
00162
00163     attedanceToolStripMenuItem.BackColor = Fore;
00164     attedanceToolStripMenuItem.ForeColor = Back;
00165
00166     classVoteToolStripMenuItem.BackColor = Fore;
00167     classVoteToolStripMenuItem.ForeColor = Back;
00168
00169     generalFAQToolStripMenuItem.BackColor = Fore;
00170     generalFAQToolStripMenuItem.ForeColor = Back;
00171
00172     aboutToolStripMenuItem.BackColor = Fore;
00173     aboutToolStripMenuItem.ForeColor = Back;
00174
00175     quitToolStripMenuItem.BackColor = Fore;
00176     quitToolStripMenuItem.ForeColor = Back;
00177
00178     for (int i = 0; i < NumQuestions; i++) //re-skin the questions
00179     present
00180     {
00181         reply_arr[i].BackColor = BackColorTheme;
00182         reply_arr[i].ForeColor = ForeColorTheme;
00183
00184         close_arr[i].BackColor = BackColorTheme;
00185         close_arr[i].ForeColor = ForeColorTheme;
00186
00187         clear_arr[i].BackColor = BackColorTheme;
00188         clear_arr[i].ForeColor = ForeColorTheme;
00189
00190         if (Fore == Color.Lime)
00191         {
00192             txtbx_reply_arr[i].BackColor = SystemColors.ControlDarkDark;
00193
00194             txtbx_reply_arr[i].ForeColor = Fore;
00195         }
00196         else
00197         {
00198             txtbx_reply_arr[i].BackColor = Color.White;
00199             txtbx_reply_arr[i].ForeColor = Color.Black;
00200
00201             group_arr[i].BackColor = BackColorTheme;
00202             group_arr[i].ForeColor = ForeColorTheme;
```

```

00202
00203     lbl_arr[i].BackColor = BackColorTheme;
00204     lbl_arr[i].ForeColor = ForeColorTheme;
00205
00206     picbx_arr[i].BackColor = BackColorTheme;
00207
00208     picbx_ConvView_arr[i].BackColor = BackColorTheme;
00209
00210     tt_picbxCV_arr[i].BackColor = BackColorTheme;
00211     tt_picbxCV_arr[i].ForeColor = ForeColorTheme;
00212 }
00213
00214 #endregion
00215
00216 #region Direct Message Controls
00217 DirectMsgPanel.BackColor = Back;
00218 DirectMsgPanel.ForeColor = Fore;
00219
00220 if (Fore == Color.Lime)
00221 {
00222     txtbxDM.BackColor = SystemColors.ControlDarkDark;
00223     txtbxDM.ForeColor = Fore;
00224
00225     cmbxDM.BackColor = SystemColors.ControlDarkDark;
00226     cmbxDM.ForeColor = Fore;
00227 }
00228 else
00229 {
00230     txtbxDM.BackColor = Color.White;
00231     txtbxDM.ForeColor = Color.Black;
00232
00233     cmbxDM.BackColor = Color.White;
00234     cmbxDM.ForeColor = Color.Black;
00235 }
00236
00237 btnDM_Clr.BackColor = Back;
00238 btnDM_Clr.ForeColor = Fore;
00239
00240 btnDM_Cls.BackColor = Back;
00241 btnDM_Cls.ForeColor = Fore;
00242
00243 btnDM_Send.BackColor = Back;
00244 btnDM_Send.ForeColor = Fore;
00245 #endregion
00246
00247 #region BroadCast Controls
00248 pnlBrdCst.BackColor = Fore;
00249 pnlBrdCst.ForeColor = Back;
00250
00251 btnSend.ForeColor = Fore;
00252 btnSend.BackColor = Back;
00253 #endregion
00254
00255 #region FAQ Panel Controls
00256 PanelFAQ.BackColor = Back;
00257 PanelFAQ.ForeColor = Fore;
00258
00259 txtbxFAQ.BackColor = Back;
00260 txtbxFAQ.ForeColor = Fore;
00261
00262 lnklblIssues.BackColor = Back;
00263 lnklblIssues.ForeColor = Fore;
00264
00265 btnFAQcls.BackColor = Back;
00266 btnFAQcls.ForeColor = Fore;
00267 #endregion
00268
00269 #region StuMgmt Controls
00270 PanelStudents.BackColor = Back;
00271 PanelStudents.ForeColor = Fore;
00272
00273 lstbxStudents.BackColor = Back;
00274 lstbxStudents.ForeColor = Fore;
00275

```

```
00276     if (Fore == Color.Lime)
00277     {
00278         txtbxStudentsName.BackColor = SystemColors.ControlDarkDark;
00279         txtbxStudentsName.ForeColor = Fore;
00280     }
00281     else
00282     {
00283         txtbxStudentsName.BackColor = Color.White;
00284         txtbxStudentsName.ForeColor = Color.Black;
00285     }
00286
00287     btnCLRStudents.BackColor = Back;
00288     btnCLRStudents.ForeColor = Fore;
00289
00290     btnEDTStudents.BackColor = Back;
00291     btnEDTStudents.ForeColor = Fore;
00292
00293     btnDoneStudents.BackColor = Back;
00294     btnDoneStudents.ForeColor = Fore;
00295
00296     btnCLSStudents.BackColor = Back;
00297     btnCLSStudents.ForeColor = Fore;
00298
00299     btnConvView.BackColor = Back;
00300     btnConvView.ForeColor = Fore;
00301 #endregion
00302
00303 #region Quiz Maker Controls
00304 PanelQuizMaker.BackColor = Back;
00305 PanelQuizMaker.ForeColor = Fore;
00306
00307     lstbxQuizMaker.BackColor = Back;
00308     lstbxQuizMaker.ForeColor = Fore;
00309
00310     if (Fore == Color.Lime)
00311     {
00312         txtbxQM.BackColor = SystemColors.ControlDarkDark;
00313         txtbxQM.ForeColor = Fore;
00314     }
00315     else
00316     {
00317         txtbxQM.BackColor = Color.White;
00318         txtbxQM.ForeColor = Color.Black;
00319     }
00320
00321     btnQMclr.BackColor = Back;
00322     btnQMclr.ForeColor = Fore;
00323
00324     btnQMcls.BackColor = Back;
00325     btnQMcls.ForeColor = Fore;
00326
00327     btnQMDel.BackColor = Back;
00328     btnQMDel.ForeColor = Fore;
00329
00330     btnQMEdit.BackColor = Back;
00331     btnQMEdit.ForeColor = Fore;
00332
00333     btnAddQM.BackColor = Back;
00334     btnAddQM.ForeColor = Fore;
00335
00336     btnQuizMode.BackColor = Back;
00337     btnQuizMode.ForeColor = Fore;
00338 #endregion
00339
00340 #region Quiz Mode Controls
00341 PanelQuizMode.BackColor = Back;
00342 PanelQuizMode.ForeColor = Fore;
00343
00344     btnExitQuiz.BackColor = Back;
00345     btnExitQuiz.ForeColor = Fore;
00346
00347 //dynamically themed when opened
00348 #endregion
00349
```

```

00350     #region Conversation Viewer Controls
00351     PanelConvView.BackColor = Back;
00352     PanelConvView.ForeColor = Fore;
00353
00354     lstbxConvView.BackColor = Back;
00355     lstbxConvView.ForeColor = Fore;
00356
00357     btnCLS_ConvView.BackColor = Back;
00358     btnCLS_ConvView.ForeColor = Fore;
00359
00360     btnCV_Print.BackColor = Back;
00361     btnCV_Print.ForeColor = Fore;
00362
00363     btnCV_Refresh.BackColor = Back;
00364     btnCV_Refresh.ForeColor = Fore;
00365
#endregion
00366
00367     #region Attendance Mode Controls
00368     PanelAttendance.BackColor = Back;
00369     PanelAttendance.ForeColor = Fore;
00370
00371     btnExitAttendance.BackColor = Back;
00372     btnExitAttendance.ForeColor = Fore;
00373
00374     lstbxAttendance.BackColor = Fore;
00375     lstbxAttendance.ForeColor = Back;
#endregion
00376
00377
00378     #region Class Vote Mode Controls
00379     PanelClassVote.BackColor = Back;
00380     PanelClassVote.ForeColor = Fore;
00381
00382     lblCVLeft.BackColor = Back;
00383     lblCVLeft.ForeColor = Fore;
00384
00385     lblCVRight.BackColor = Back;
00386     lblCVRight.ForeColor = Fore;
00387
00388     lblOption1.BackColor = Back;
00389     lblOption1.ForeColor = Fore;
00390
00391     lblOption2.BackColor = Back;
00392     lblOption2.ForeColor = Fore;
00393
00394     lblCVPipe.BackColor = Back;
00395     lblCVPipe.ForeColor = Fore;
00396
00397     txtbxCVStats.BackColor = Fore;
00398     txtbxCVStats.ForeColor = Back;
00399
00400     btnCVReset.BackColor = Back;
00401     btnCVReset.ForeColor = Fore;
00402
00403     btnExitClassVote.BackColor = Back;
00404     btnExitClassVote.ForeColor = Fore;
#endregion
00405
00406
00407     #region About Dialog
00408     About.BackColor = Back;
00409     About.ForeColor = Fore;
00410
#endregion
00411
00412     #region Notify Panel
00413     PanelNotify.BackColor = Fore;
00414     PanelNotify.ForeColor = Back;
00415
00416     pictureBoxNotify.BackColor = Fore;
00417
00418     lblNotify.BackColor = Fore;
00419     lblNotify.ForeColor = Back;
#endregion
00420
00421
00422     this.ResumeLayout();
00423     //this.Show(); //tada!!

```

```
00424      }
00425      }
00426 }
```