

Documentação da Interface

Generated by Doxygen 1.9.4

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	7
4.1 File List	7
5 Namespace Documentation	9
5.1 Ui Namespace Reference	9
6 Class Documentation	11
6.1 Button Class Reference	11
6.1.1 Detailed Description	13
6.1.2 Constructor & Destructor Documentation	13
6.1.2.1 Button()	14
6.1.3 Member Function Documentation	14
6.1.3.1 buttonSizeConfiguration()	14
6.1.3.2 changeButton_style()	15
6.1.3.3 changeHeaderPage_InsideExperiment	16
6.1.3.4 changeInitShear_toFinishButton()	16
6.1.3.5 changePage_InsideExperiment	17
6.1.3.6 iconsConfiguration()	18
6.1.3.7 imageConfiguration()	18
6.1.3.8 initExperiment_ButtonStyle()	18
6.1.3.9 initialButtonStyling()	19
6.1.3.10 initialButtonStyling_Layout()	19
6.1.3.11 initialButtonStyling_Widget()	20
6.1.3.12 pressureButton_style()	21
6.1.3.13 setButton_style_icon()	22
6.1.3.14 setButtonShadow()	22
6.1.3.15 styleSheetConfiguration()	23
6.1.4 Member Data Documentation	23
6.1.4.1 button_icons	23
6.1.4.2 button_images	23
6.1.4.3 button_styleSheets	24
6.1.4.4 buttonMaximumSize	24
6.1.4.5 buttonMinimumSize	24
6.1.4.6 clickedButton	24
6.1.4.7 clickedButtonIcon	24

6.2 Charts Class Reference	25
6.2.1 Detailed Description	27
6.2.2 Constructor & Destructor Documentation	27
6.2.2.1 Charts()	27
6.2.3 Member Function Documentation	27
6.2.3.1 initialConfiguration()	27
6.2.3.2 reset_Chart()	28
6.2.3.3 updateDensificationCharts	28
6.2.3.4 updateShearCharts	29
6.2.4 Member Data Documentation	29
6.2.4.1 chart	29
6.2.4.2 chart_title	29
6.2.4.3 chartView	29
6.2.4.4 m_axisX	29
6.2.4.5 m_axisY	29
6.2.4.6 series	30
6.2.4.7 x_axis_title	30
6.2.4.8 y_axis_title	30
6.3 DBManager Class Reference	30
6.3.1 Detailed Description	33
6.3.2 Constructor & Destructor Documentation	34
6.3.2.1 DBManager()	34
6.3.2.2 ~DBManager()	34
6.3.3 Member Function Documentation	34
6.3.3.1 createTable()	34
6.3.3.2 getExperiment_id()	35
6.3.3.3 insertIntoTable()	35
6.3.3.4 insertValuesIntoBind_Densification()	37
6.3.3.5 insertValuesIntoBind_Experiment()	38
6.3.3.6 insertValuesIntoBind_FinalVariables()	40
6.3.3.7 insertValuesIntoBind_SampleVariables()	41
6.3.3.8 insertValuesIntoBind_SampleVariablesUpdate()	42
6.3.3.9 insertValuesIntoBind_Shear()	42
6.3.3.10 isOpen()	43
6.3.3.11 selectExperimentId()	44
6.3.3.12 tableExists()	44
6.3.3.13 update_database_table	45
6.3.3.14 updateTable()	46
6.3.4 Member Data Documentation	46
6.3.4.1 create_table	46
6.3.4.2 experiment_data	47
6.3.4.3 experiment_id	47

6.3.4.4 insert_into_table	47
6.3.4.5 prova_conceito_database	47
6.3.4.6 table_name	47
6.4 Densification Class Reference	48
6.4.1 Detailed Description	49
6.4.2 Constructor & Destructor Documentation	49
6.4.2.1 Densification()	49
6.4.3 Member Function Documentation	49
6.4.3.1 getDensificationResults()	49
6.4.3.2 getDensificationVariables()	50
6.4.3.3 getSample_number()	50
6.4.3.4 getVertical_displacement()	51
6.4.3.5 getVertical_load()	51
6.4.3.6 setDensificationVariables()	51
6.4.3.7 setDiff_sampleNumber_initExperiment()	52
6.4.4 Member Data Documentation	52
6.4.4.1 diff_sampleNumber_initExperiment	52
6.4.4.2 sample_number	53
6.4.4.3 vertical_displacement	53
6.4.4.4 vertical_load	53
6.5 Experiment Class Reference	53
6.5.1 Constructor & Destructor Documentation	57
6.5.1.1 Experiment()	57
6.5.2 Member Function Documentation	58
6.5.2.1 changePhase()	58
6.5.2.2 day_month_year()	58
6.5.2.3 getAllData_forInfoTable()	59
6.5.2.4 getAllData_forPhasesTable()	60
6.5.2.5 getArea()	60
6.5.2.6getAshto_class()	61
6.5.2.7 getAverageSpeed()	61
6.5.2.8 getBoring_number()	62
6.5.2.9 getDiameter()	62
6.5.2.10 getDuration()	62
6.5.2.11 getDuration_string()	63
6.5.2.12 getExperimentStarted()	64
6.5.2.13 getInitial_dry_density()	65
6.5.2.14 getInitial_height()	65
6.5.2.15 getInitial_moisture()	65
6.5.2.16 getInitial_position()	66
6.5.2.17 getInitial_saturation()	66
6.5.2.18 getInitial_time_milliseconds()	66

6.5.2.19 getInitial_time_seconds()	67
6.5.2.20 getInitial_timeString()	67
6.5.2.21 getinitial_void_ratio()	68
6.5.2.22 getinitial_volume()	68
6.5.2.23 getinitial_wet_density()	69
6.5.2.24 getInitial_wet_weight()	69
6.5.2.25 getLiquid_limit()	69
6.5.2.26 getName()	70
6.5.2.27 getNormalTension()	70
6.5.2.28 getOperator_name()	71
6.5.2.29 getPhase()	71
6.5.2.30 getPlastic_limit()	72
6.5.2.31 getPresent_time_miliseconds()	72
6.5.2.32 getPresent_time_seconds()	72
6.5.2.33 getPressure()	73
6.5.2.34 getSample_description()	73
6.5.2.35 getSample_id()	73
6.5.2.36 getSample_location()	74
6.5.2.37 getSample_period()	74
6.5.2.38 getSample_preparations()	74
6.5.2.39 getShearSample_number()	75
6.5.2.40 getShearTension()	75
6.5.2.41 getSpecimen_type()	76
6.5.2.42 getSpgr_solids()	76
6.5.2.43 getTest_type()	76
6.5.2.44 getUscs_class()	77
6.5.2.45 getwater_specific_weight()	77
6.5.2.46 hour_min_sec_ms()	77
6.5.2.47 insertData_inDatabase()	78
6.5.2.48 setAshto_class()	78
6.5.2.49 setBoring_number()	78
6.5.2.50 setDiameter()	79
6.5.2.51 setExperimentStarted()	79
6.5.2.52 setInitial_height()	79
6.5.2.53 setInitial_moisture()	80
6.5.2.54 setInitial_position()	80
6.5.2.55 setInitial_time()	80
6.5.2.56 setInitial_wet_weight()	81
6.5.2.57 setLiquid_limit()	81
6.5.2.58 setName()	81
6.5.2.59 setOperator_name()	82
6.5.2.60 setPlastic_limit()	82

6.5.2.61 setPressure()	82
6.5.2.62 setSample_description()	83
6.5.2.63 setSample_id()	83
6.5.2.64 setSample_location()	83
6.5.2.65 setSample_period()	84
6.5.2.66 setSample_preparations()	84
6.5.2.67 setSpecimen_type()	84
6.5.2.68 setSpgr_solids()	85
6.5.2.69 setTest_type()	85
6.5.2.70 setUscs_class()	85
6.5.2.71 updateDensificationChart	86
6.5.2.72 updateDensificationResultsTable()	86
6.5.2.73 updateDensificationTable()	87
6.5.2.74 updateShearChart	87
6.5.2.75 updateShearResultsTable()	88
6.5.2.76 updateShearTable()	88
6.5.3 Member Data Documentation	89
6.5.3.1 ashto_class	89
6.5.3.2 boring_number	89
6.5.3.3 densification_variables	89
6.5.3.4 diameter	90
6.5.3.5 experimentStarted	90
6.5.3.6 initial_height	90
6.5.3.7 initial_moisture	90
6.5.3.8 initial_position	90
6.5.3.9 initial_time_milliseconds	90
6.5.3.10 initial_time_seconds	90
6.5.3.11 initial_wet_weight	90
6.5.3.12 liquid_limit	91
6.5.3.13 name	91
6.5.3.14 operator_name	91
6.5.3.15 phase	91
6.5.3.16 plastic_limit	91
6.5.3.17 present_time_milliseconds	91
6.5.3.18 present_time_seconds	91
6.5.3.19 pressure	91
6.5.3.20 sample_description	92
6.5.3.21 sample_id	92
6.5.3.22 sample_location	92
6.5.3.23 sample_period	92
6.5.3.24 sample_preparations	92
6.5.3.25 shear_variables	92

6.5.3.26 specimen_type	92
6.5.3.27 spgr_solids	92
6.5.3.28 test_type	93
6.5.3.29 uscs_class	93
6.6 exportData Class Reference	93
6.6.1 Detailed Description	94
6.6.2 Constructor & Destructor Documentation	95
6.6.2.1 exportData()	95
6.6.2.2 ~exportData()	95
6.6.3 Member Function Documentation	95
6.6.3.1 exportCSV	95
6.6.3.2 write()	96
6.7 Field Class Reference	96
6.7.1 Detailed Description	99
6.7.2 Constructor & Destructor Documentation	99
6.7.2.1 Field()	99
6.7.3 Member Function Documentation	99
6.7.3.1 clearFields()	99
6.7.3.2 customizeField()	100
6.7.3.3 customizeOneField()	101
6.7.3.4 setVariables	103
6.7.4 Member Data Documentation	105
6.7.4.1 FieldMaximumSize	105
6.7.4.2 FieldMinimumSize	105
6.7.4.3 info_variables	105
6.8 interface_to_machine_message Union Reference	105
6.8.1 Detailed Description	106
6.8.2 Member Data Documentation	106
6.8.2.1	106
6.8.2.2 command	106
6.8.2.3 distance	106
6.8.2.4 enabled	107
6.8.2.5 payload	107
6.8.2.6 pressure	107
6.8.2.7 sample_period	107
6.8.2.8 velocity	107
6.9 machine_to_interface_message Union Reference	107
6.9.1 Detailed Description	108
6.9.2 Member Data Documentation	108
6.9.2.1	108
6.9.2.2 displacement	108
6.9.2.3 load	108

6.9.2.4 payload	108
6.9.2.5 sample_number	109
6.9.2.6 state	109
6.10 machineClient Class Reference	109
6.10.1 Detailed Description	110
6.10.2 Constructor & Destructor Documentation	110
6.10.2.1 machineClient()	110
6.10.2.2 ~machineClient()	110
6.10.3 Member Function Documentation	110
6.10.3.1 sendMessages()	110
6.10.4 Member Data Documentation	111
6.10.4.1 errorOccurred	111
6.10.4.2 socket_id	111
6.10.4.3 socket_name	111
6.11 machineServer Class Reference	112
6.11.1 Detailed Description	114
6.11.2 Constructor & Destructor Documentation	114
6.11.2.1 machineServer()	114
6.11.2.2 ~machineServer()	114
6.11.3 Member Function Documentation	114
6.11.3.1 init()	115
6.11.3.2 read_interface_message()	115
6.11.3.3 run()	115
6.11.4 Member Data Documentation	116
6.11.4.1 client_socket_id	116
6.11.4.2 errorOccurred	116
6.11.4.3 interface_message	116
6.11.4.4 server_socket_id	116
6.11.4.5 server_socket_name	116
6.12 MainWindow Class Reference	117
6.12.1 Detailed Description	120
6.12.2 Constructor & Destructor Documentation	120
6.12.2.1 MainWindow()	120
6.12.2.2 ~MainWindow()	121
6.12.3 Member Function Documentation	121
6.12.3.1 adjustVelocity_Distance	122
6.12.3.2 cancelExperiment	123
6.12.3.3 changeDistance	124
6.12.3.4 changeExportOption_Densification	124
6.12.3.5 changeExportOption_Shear	125
6.12.3.6 changeInitialPositionValue	125
6.12.3.7 changeOutsideExperimentPage	126

6.12.3.8 changePage()	126
6.12.3.9 changePhase	128
6.12.3.10 changeVelocity	128
6.12.3.11 connectButtonsToSlots_Layout()	129
6.12.3.12 connectButtonsToSlots_Widget()	130
6.12.3.13 connectButtonToSlots_WithArguments()	130
6.12.3.14 CreateDatabaseTables()	131
6.12.3.15 enableExportButton	131
6.12.3.16 enableShearInitButton	132
6.12.3.17 fillTextEditForTests()	132
6.12.3.18 InitialConfiguration_InsideExperimentButtons()	134
6.12.3.19 InitialConfiguration_OutsideExperimentHeaderButtons()	135
6.12.3.20 InitialConfiguration_PhasesButtons()	136
6.12.3.21 InitialConfiguration_PhasesFields()	136
6.12.3.22 InitialConfiguration_Tables()	137
6.12.3.23 initShearPhase	138
6.12.3.24 nextPhase	139
6.12.3.25 on_goBack_toolButton_clicked	140
6.12.3.26 on_initExperiment_toolButton_clicked	141
6.12.3.27 on_releasePressure_toolButton_clicked	141
6.12.3.28 onPositionButton_pressed	142
6.12.3.29 onPositionButton_released	143
6.12.3.30 updateResultsTables	144
6.12.4 Member Data Documentation	144
6.12.4.1 charts_variables	145
6.12.4.2 chosenTable	145
6.12.4.3 data_export	145
6.12.4.4 experiment_canceled	145
6.12.4.5 export_option	145
6.12.4.6 info_variables	145
6.12.4.7 my_db	145
6.12.4.8 previousIndex	145
6.12.4.9 receive_data	146
6.12.4.10 send_data	146
6.12.4.11 setupButtons	146
6.12.4.12 setupFields	146
6.12.4.13 tables	146
6.12.4.14 timer	146
6.12.4.15 ui	146
6.13 QTstyle_Test Class Reference	147
6.13.1 Detailed Description	148
6.13.2 Member Enumeration Documentation	148

6.13.2.1 TEnum	148
6.13.3 Constructor & Destructor Documentation	148
6.13.3.1 QTstyle_Test()	148
6.13.3.2 ~QTstyle_Test()	148
6.13.4 Member Function Documentation	149
6.13.4.1 testMe()	149
6.13.4.2 testMeToo()	149
6.13.5 Member Data Documentation	149
6.13.5.1 enumPtr	150
6.13.5.2 enumVar	150
6.13.5.3 handler	150
6.13.5.4 publicVar	150
6.14 receiveData Class Reference	151
6.14.1 Detailed Description	153
6.14.2 Constructor & Destructor Documentation	153
6.14.2.1 receiveData()	153
6.14.2.2 ~receiveData()	154
6.14.3 Member Function Documentation	154
6.14.3.1 data_arrived	154
6.14.3.2 init()	154
6.14.3.3 initSocketServer()	155
6.14.3.4 readClientMessage()	155
6.14.3.5 run()	156
6.14.4 Member Data Documentation	157
6.14.4.1 client_socket_id	157
6.14.4.2 errorOccurred	157
6.14.4.3 machine_message	157
6.14.4.4 server_socket_id	158
6.14.4.5 server_socket_name	158
6.14.4.6 shear_densification_variables	158
6.15 sendCommands Class Reference	158
6.15.1 Constructor & Destructor Documentation	159
6.15.1.1 sendCommands()	159
6.15.1.2 ~sendCommands()	159
6.15.2 Member Function Documentation	159
6.15.2.1 connectToMachine()	160
6.15.2.2 getDistance()	160
6.15.2.3 getEnabled()	160
6.15.2.4 getPressure()	160
6.15.2.5 getSamplingPeriod()	160
6.15.2.6 getVelocity()	160
6.15.2.7 sendMessage()	161

6.15.2.8 setCommand()	161
6.15.2.9 setDistance()	161
6.15.2.10 setEnabled()	162
6.15.2.11 setPressure()	162
6.15.2.12 setSamplingPeriod()	162
6.15.2.13 setVelocity()	163
6.15.3 Member Data Documentation	163
6.15.3.1 enabled	163
6.15.3.2 errorOccurred	163
6.15.3.3 interface_message	164
6.15.3.4 sampling_period	164
6.15.3.5 socket_id	164
6.15.3.6 socket_name	164
6.16 Shear Class Reference	164
6.16.1 Constructor & Destructor Documentation	166
6.16.1.1 Shear()	166
6.16.2 Member Function Documentation	166
6.16.2.1 getDate()	166
6.16.2.2 getDistance()	167
6.16.2.3 getHorizontal_displacement()	167
6.16.2.4 getHorizontal_load()	167
6.16.2.5 getHour_min_sec_ms()	168
6.16.2.6 getInitial_time_milliseconds()	168
6.16.2.7 getInitial_time_seconds()	168
6.16.2.8 getSample_number_diff()	169
6.16.2.9 getShearResults()	169
6.16.2.10 getShearVariables()	170
6.16.2.11 getVelocity()	170
6.16.2.12 setDate()	171
6.16.2.13 setDistance()	171
6.16.2.14 setHorizontal_displacement()	171
6.16.2.15 setHorizontal_load()	172
6.16.2.16 setHour_min_sec_ms()	172
6.16.2.17 setInitial_time()	172
6.16.2.18 setSample_number_diff()	173
6.16.2.19 setShearVariables()	173
6.16.2.20 setVelocity()	174
6.16.3 Member Data Documentation	174
6.16.3.1 date	174
6.16.3.2 distance	174
6.16.3.3 horizontal_displacement	174
6.16.3.4 horizontal_load	174

6.16.3.5 hour_min_sec_ms	174
6.16.3.6 initial_time_milliseconds	175
6.16.3.7 initial_time_seconds	175
6.16.3.8 sample_number_diff	175
6.16.3.9 velocity	175
6.17 SocketTests Class Reference	175
6.17.1 Detailed Description	178
6.17.2 Constructor & Destructor Documentation	178
6.17.2.1 SocketTests()	179
6.17.2.2 ~SocketTests()	179
6.17.3 Member Function Documentation	179
6.17.3.1 cleanupTestCase	179
6.17.3.2 compareStructFloatElements()	179
6.17.3.3 compareStructIntElements()	180
6.17.3.4 initTestCase	181
6.17.3.5 test_receiveDataFromInterface	181
6.17.3.6 test_receiveDataFromInterface_data	181
6.17.3.7 test_receiveDataFromMachine	182
6.17.3.8 test_receiveDataFromMachine_data	182
6.17.4 Member Data Documentation	182
6.17.4.1 info_variables	182
6.17.4.2 receiveDataFromInterface	182
6.17.4.3 receiveDataFromMachine	183
6.17.4.4 setupFields	183
6.17.4.5 tables	183
6.17.4.6 test_sendData	183
6.17.4.7 test_sendDataMachine	183
6.18 Table Class Reference	184
6.18.1 Constructor & Destructor Documentation	186
6.18.1.1 Table()	186
6.18.2 Member Function Documentation	186
6.18.2.1 clearDynamicTables()	186
6.18.2.2 clearStaticTables()	186
6.18.2.3 customizeTable()	187
6.18.2.4 initialConfig_DynamicTable()	187
6.18.2.5 initialConfig_ShearTable()	187
6.18.2.6 initialConfig_StaticTable()	187
6.18.2.7 test()	188
6.18.2.8 updateData_DynamicTable()	188
6.18.2.9 updateData_ShearTable()	188
6.18.2.10 updateData_StaticTable()	189
6.18.3 Member Data Documentation	190

6.18.3.1 columnValues	190
6.18.3.2 densificationTable	190
6.18.3.3 headerNames	190
6.18.3.4 lineNames	190
6.18.3.5 machineTablevalues	190
6.18.3.6 shearTable	190
6.18.3.7 table_variables	190
6.19 ThreadController Class Reference	191
6.19.1 Detailed Description	193
6.19.2 Constructor & Destructor Documentation	193
6.19.2.1 ThreadController()	193
6.19.2.2 ~ThreadController()	194
6.19.3 Member Function Documentation	194
6.19.3.1 receiveThreadFinishedSlot	194
6.19.4 Member Data Documentation	195
6.19.4.1 isThreadDestroyed	195
6.19.4.2 receiveDataThread	195
7 File Documentation	197
7.1 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/sendcommands.cpp File Reference	197
7.2 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/threadcontroller.cpp File Reference	197
7.3 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/field.cpp File Reference	198
7.4 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/mainwindow.cpp File Reference	199
7.5 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/table.cpp File Reference	199
7.6 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/button.cpp File Reference	200
7.7 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/shear.cpp File Reference	201
7.8 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/receivedata.cpp File Reference	201
7.9 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/experiment.cpp File Reference	202
7.10 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/main.cpp File Reference	203
7.10.1 Function Documentation	203
7.10.1.1 main()	203
7.11 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/charts.cpp File Reference	204
7.12 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface← Tests/inc/machineclient.h File Reference	204
7.13 machineclient.h	205

7.14	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface Tests/inc/machineserver.h File Reference	205
7.15	machineserver.h	206
7.16	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface Tests/src/machineserver.cpp File Reference	207
7.17	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface Tests/src/machineclient.cpp File Reference	207
7.18	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface Tests/src/tst_sockettests.cpp File Reference	208
7.18.1	Macro Definition Documentation	209
7.18.1.1	tests	209
7.19	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface Tests/src/testDoxygen.cpp File Reference	209
7.20	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/socket _local.h File Reference	209
7.20.1	Macro Definition Documentation	210
7.20.1.1	interface_payload_size	211
7.20.1.2	machine_payload_size	211
7.20.2	Typedef Documentation	211
7.20.2.1	interface_to_machine_message	211
7.20.2.2	machine_to_interface_message	211
7.21	socket_local.h	212
7.22	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/receivedata.h File Reference	213
7.23	receivedata.h	214
7.24	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/threadcontroller.h File Reference	214
7.25	threadcontroller.h	215
7.26	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/dbmanager.h File Reference	215
7.26.1	Macro Definition Documentation	217
7.26.1.1	densification_table	217
7.26.1.2	experiment_table	217
7.26.1.3	final_variables_table	217
7.26.1.4	sample_table	217
7.26.1.5	shear_table	217
7.27	dbmanager.h	218
7.28	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/shear.h File Reference	219
7.29	shear.h	220
7.30	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/densification.h File Reference	220
7.31	densification.h	221
7.32	/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/mainwindow.h File Reference	222

7.33 mainwindow.h	223
7.34 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/experiment.h File Reference	224
7.34.1 Macro Definition Documentation	226
7.34.1.1 densification_phase	226
7.34.1.2 pi_value	226
7.34.1.3 shear_phase	226
7.35 experiment.h	226
7.36 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/sendcommands.h File Reference	228
7.37 sendcommands.h	229
7.38 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/charts.h File Reference	230
7.38.1 Macro Definition Documentation	231
7.38.1.1 densification_chart	231
7.38.1.2 shear_chart	231
7.39 charts.h	232
7.40 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/button.h File Reference	232
7.40.1 Macro Definition Documentation	234
7.40.1.1 cancelButton_BackgroundColor	234
7.40.1.2 cancelButton_size	234
7.40.1.3 configurationButton_darkIcon	234
7.40.1.4 configurationButton_lightIcon	234
7.40.1.5 continueButton_BackgroundColor	234
7.40.1.6 continueButton_Icon	234
7.40.1.7 densificationButton_darkIcon	235
7.40.1.8 densificationButton_lightIcon	235
7.40.1.9 experimentButton_darkIcon	235
7.40.1.10 experimentButton_lightIcon	235
7.40.1.11 exportButton_size	235
7.40.1.12 finishButton_Icon	235
7.40.1.13 headerButton_darkBackgroundColor	235
7.40.1.14 headerButton_lightBackgroundColor	235
7.40.1.15 historyButton_darkIcon	236
7.40.1.16 historyButton_lightIcon	236
7.40.1.17 infoButton_darkIcon	236
7.40.1.18 infoButton_lightIcon	236
7.40.1.19 initShearButton_BackgroundColor	236
7.40.1.20 initShearButton_size	236
7.40.1.21 moveButton_DisabledBackgroundColor	236
7.40.1.22 no_icon	236
7.40.1.23 outsideExperiment_buttonSize	237

7.40.1.24 phases_buttonSize	237
7.40.1.25 phases_continueButtonSize	237
7.40.1.26 phasesButton_darkBackgroundColor	237
7.40.1.27 phasesButton_lightBackgroundColor	237
7.40.1.28 pressureButton_GreenBackgroundColor	237
7.40.1.29 pressureButton_RedBackgroundColor	237
7.40.1.30 shearButton_darkIcon	237
7.40.1.31 shearButton_lightIcon	238
7.40.1.32 velocityPositionBackgroundColor	238
7.40.1.33 velocityPositionButton_size	238
7.41 button.h	238
7.42 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/exportdata.h File Reference	239
7.43 exportdata.h	240
7.44 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/field.h File Reference	240
7.45 field.h	241
7.46 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/table.h File Reference	242
7.46.1 Macro Definition Documentation	243
7.46.1.1 densification_result_table	243
7.46.1.2 design_densification_table	243
7.46.1.3 design_shear_table	244
7.46.1.4 info_table	244
7.46.1.5 phases_table	244
7.46.1.6 shear_result_table	244
7.47 table.h	244
7.48 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/densification.cpp File Reference	245
7.49 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/exportdata.cpp File Reference	245
7.50 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/dbmanager.cpp File Reference	246
Index	247

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ui	9
--------------------	-------	---

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Densification	48
interface_to_machine_message	105
machine_to_interface_message	107
machineClient	109
QMainWindow	
MainWindow	117
QObject	
Button	11
Charts	25
DBManager	30
Experiment	53
Field	96
Shear	164
SocketTests	175
Table	184
ThreadController	191
exportData	93
QThread	
machineServer	112
receiveData	151
QTstyle_Test	147
sendCommands	158

Chapter 3

Class Index

3.1 Class List

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

Button	Classe dos botões	11
Charts	Classe responsável pelos gráficos	25
DBManager	Classe do banco de dados	30
Densification	Classe do adensamento	48
Experiment	53
exportData	Classe que exporta os dados para csv	93
Field	Classe dos campos de texto	96
interface_to_machine_message	Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina . . .	105
machine_to_interface_message	Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface . . .	107
machineClient	Classe que simula o envio de dados da máquina real	109
machineServer	Classe que simula o servidor da máquina real	112
MainWindow	Classe que instancia a janela principal da interface	117
QTstyle_Test	A test class	147
receiveData	Classe que recebe os dados da camada de controle	151
sendCommands	158
Shear	164
SocketTests	Classe de testes para o socket	175
Table	184
ThreadController	Classe que controla a thread criada por <code>receiveData</code>	191

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

```
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/button.cpp 200
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/charts.cpp 204
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/dbmanager.cpp
    246
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/densification.cpp
    245
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/experiment.cpp
    202
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/exportdata.cpp
    245
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/field.cpp . 198
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/main.cpp 203
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/mainwindow.cpp
    199
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/receivedata.cpp
    201
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/sendcommands.cpp
    197
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/shear.cpp 201
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/table.cpp 199
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/threadcontroller.cpp
    197
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/inc/machineclient.h
    204
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/inc/machineserver.h
    205
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/machineclient.cpp
    207
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/machineserver.cpp
    207
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/testDoxygen.cpp
    209
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/tst_sockettests.cpp
    208
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/button.h . 232
```

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[charts.h](#) . . 230
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[dbmanager.h](#)
215
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[densification.h](#)
220
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[experiment.h](#)
224
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[exportdata.h](#)
239
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[field.h](#) . . 240
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[mainwindow.h](#)
222
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[receivedata.h](#)
213
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[sendcommands.h](#)
228
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[shear.h](#) . . 219
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[socket_local.h](#)
209
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[table.h](#) . . 242
/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[threadcontroller.h](#)
214

Chapter 5

Namespace Documentation

5.1 Ui Namespace Reference

Chapter 6

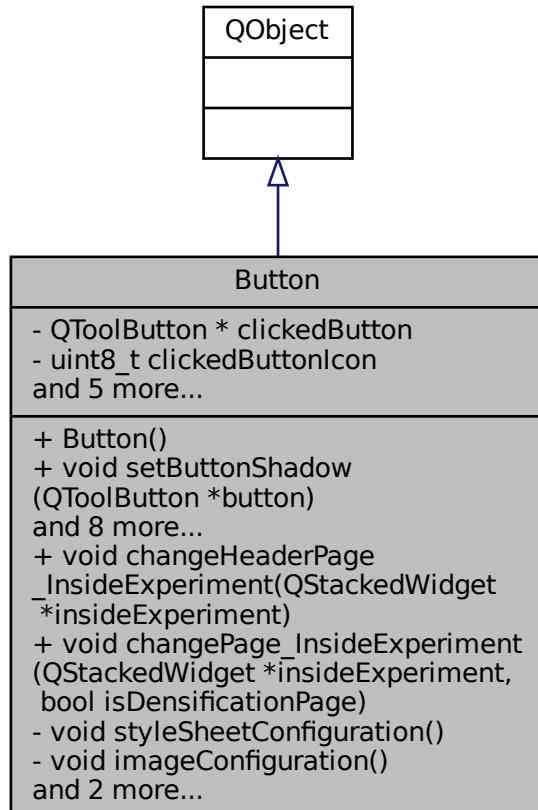
Class Documentation

6.1 Button Class Reference

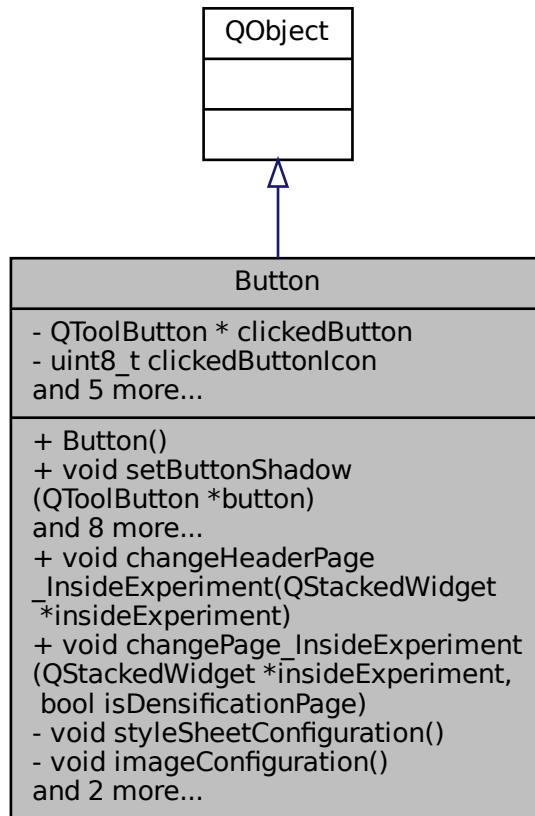
Classe dos botões.

```
#include <button.h>
```

Inheritance diagram for Button:



Collaboration diagram for Button:



Public Slots

- void [changeHeaderPage_InsideExperiment](#) (QStackedWidget *insideExperiment)
Responsável por alterar a página e o estilo dos botões do cabeçalho ao trocar de página.
- void [changePage_InsideExperiment](#) (QStackedWidget *insideExperiment, bool isDensificationPage)
Responsável por alterar entre as páginas de gráficos/tabelas/resultados dentro da página Adensamento ou Cisalhamento.

Public Member Functions

- [Button \(\)](#)
Constrói um novo objeto da classe [Button](#).
- void [setButtonShadow](#) (QToolButton *button)
Estiliza um QPushButton ou QToolButton para ter sombra.
- void [setButton_style_icon](#) (QToolButton *button, uint8_t style, uint8_t icon)
Altera o ícone e estilo do botão.
- void [initialButtonStyling_Layout](#) (QHBoxLayout *boxlayout, uint8_t style_option, uint8_t size)
Realiza a estilização de botões dentro de 1 layout.

- void [changeButton_style](#) (QToolButton *current_button, uint8_t icon, uint8_t style, uint8_t pos)
Altera a estilização do botão ao ser clicado.
- void [initialButtonStyling_Widget](#) (QObject *selectedWidget, uint8_t style_option, uint8_t size)
Estilização dos botões de um Widget escolhido.
- void [initExperiment_ButtonStyle](#) (QToolButton *play_button, bool enabled)
Estilização do botão de início do experimento.
- void [pressureButton_style](#) (QToolButton *pressure)
Estilização do botão da pressão.
- void [initialButtonStyling](#) (QToolButton *button, uint8_t style_option, uint8_t size)
Estilização inicial do botão, colocando-se sombra, estilo e tamanho.
- void [changeInitShear_toFinishButton](#) (QToolButton *button)
Altera a aparência do botão iniciar cisalhamento para finalizar o experimento.

Private Member Functions

- void [styleSheetConfiguration](#) ()
Cria as folhas de estilo para os botões.
- void [imageConfiguration](#) ()
Insere no array as imagens utilizadas na aplicação.
- void [iconsConfiguration](#) ()
Adiciona as imagens aos ícones para facilitar a estilização.
- void [buttonSizeConfiguration](#) ()
Insere os valores mínimos e máximos do tamanho dos botões.

Private Attributes

- QToolButton * [clickedButton](#) [3]
- uint8_t [clickedButtonIcon](#) [3]
- QString [button_styleSheets](#) [15]
- QString [button_images](#) [20]
- QIcon [button_icons](#) [20]
- QSize [buttonMaximumSize](#) [10]
- QSize [buttonMinimumSize](#) [10]

6.1.1 Detailed Description

Classe dos botões.

Esta classe é responsável por cuidar do comportamento e estilização dos botões.

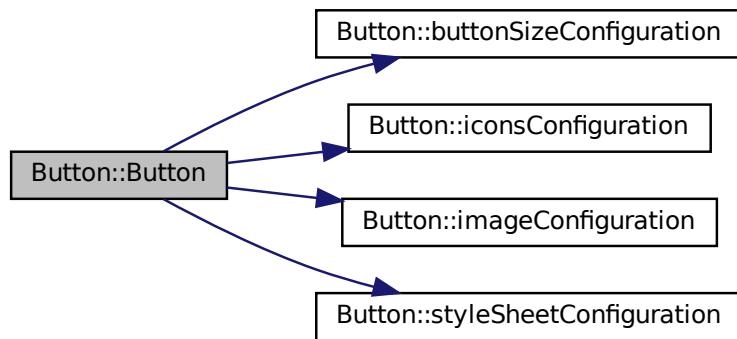
6.1.2 Constructor & Destructor Documentation

6.1.2.1 Button()

```
Button::Button ( )
```

Constrói um novo objeto da classe [Button](#).

Esta classe tem como objetivo criar as variáveis de estilo, imagem, ícones e outras configurações iniciais dos botões; Here is the call graph for this function:



6.1.3 Member Function Documentation

6.1.3.1 buttonSizeConfiguration()

```
void Button::buttonSizeConfiguration ( ) [private]
```

Insere os valores mínimos e máximos do tamanho dos botões.

Here is the caller graph for this function:



6.1.3.2 changeButton_style()

```
void Button::changeButton_style (
    QToolButton * current_button,
    uint8_t icon,
    uint8_t style,
    uint8_t pos )
```

Altera a estilização do botão ao ser clicado.

Parameters

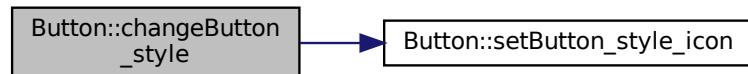
<i>current_button</i>	Botão que foi clicado
<i>icon</i>	Ícone do botão clicado

Os botões de página mudam de cor ao serem clicados

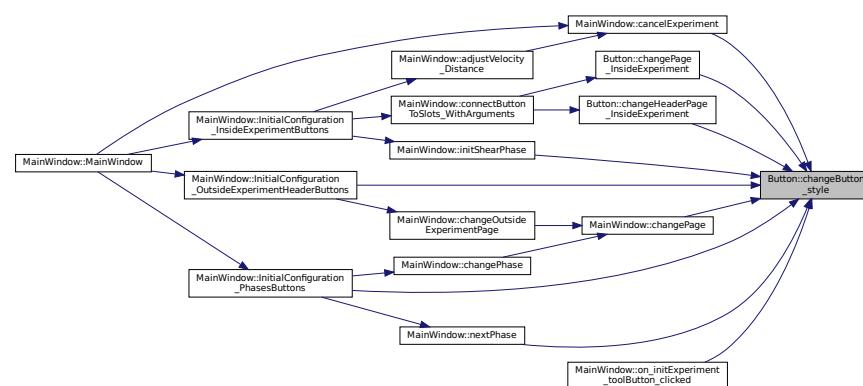
Parameters

<i>current_button</i>	Botão escolhido
<i>icon</i>	Ícone do botão
<i>style</i>	Estilo do botão
<i>pos</i>	Identificação do cabeçalho de páginas atual

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.3 changeHeaderPage_InsideExperiment

```
void Button::changeHeaderPage_InsideExperiment (
    QStackedWidget * insideExperiment ) [slot]
```

Responsável por alterar a página e o estilo dos botões do cabeçalho ao trocar de página.

Parameters

<i>insideExperiment</i>	Utilizado para trocar de StackWidget na troca de páginas.
-------------------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.4 changeInitShear_toFinishButton()

```
void Button::changeInitShear_toFinishButton (
    QToolButton * button )
```

Altera a aparência do botão iniciar cisalhamento para finalizar o experimento.

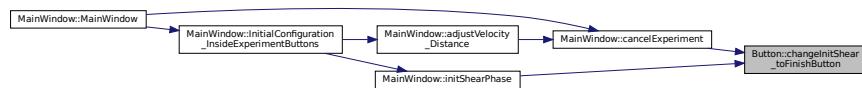
Parameters

<i>button</i>	Botão que será alterado.
---------------	--------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.5 changePage_InsideExperiment

```
void Button::changePage_InsideExperiment (
    QStackedWidget * insideExperiment,
    bool isDensificationPage ) [slot]
```

Responsável por alterar entre as páginas de gráficos/tabelas/resultados dentro da página Adensamento ou Cisalhamento.

Parameters

<i>insideExperiment</i>	Utilizado para trocar de StackWidget na troca de páginas.
<i>isDensificationPage</i>	Utilizado para identificar se é a página de Adensamento ou Cisalhamento

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.6 iconsConfiguration()

```
void Button::iconsConfiguration ( ) [private]
```

Adiciona as imagens aos ícones para facilitar a estilização.

Here is the caller graph for this function:



6.1.3.7 imageConfiguration()

```
void Button::imageConfiguration ( ) [private]
```

Insere no array as imagens utilizadas na aplicação.

Here is the caller graph for this function:



6.1.3.8 initExperiment_ButtonStyle()

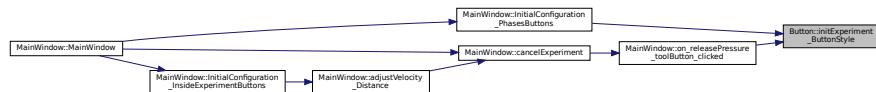
```
void Button::initExperiment_ButtonStyle (
    QToolButton * play_button,
    bool enabled )
```

Estilização do botão de início do experimento.

Parameters

<i>play_button</i>	Botão escolhido.
<i>enabled</i>	Se verdadeiro, o botão está habilitado, se falso, está desabilitado.

Here is the caller graph for this function:



6.1.3.9 initialButtonStyling()

```
void Button::initialButtonStyling (
    QToolButton * button,
    uint8_t style_option,
    uint8_t size )
```

Estilização inicial do botão, colocando-se sombra, estilo e tamanho.

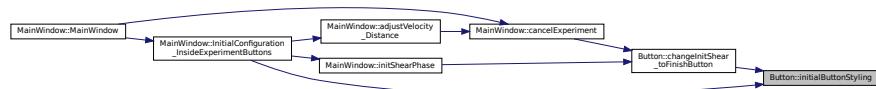
Parameters

<i>button</i>	O botão que será alterado.
<i>style_option</i>	A folha de estilo.
<i>size</i>	O tamanho do botão.

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.10 initialButtonStyling_Layout()

```
void Button::initialButtonStyling_Layout (
    QBoxLayout * boxlayout,
```

```
    uint8_t style_option,
    uint8_t size )
```

Realiza a estilização de botões dentro de 1 layout.

Essa estilização é realizada iterando sobre os elementos do layout.

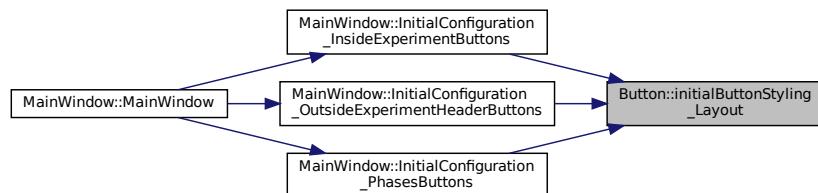
Parameters

<i>boxlayout</i>	Layout escolhido
<i>style_option</i>	Estilo dos botões escolhido
<i>size</i>	Tamanho dos botões escolhido

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.11 initialButtonStyling_Widget()

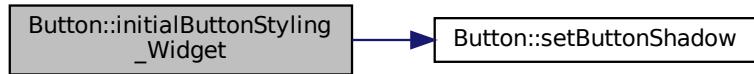
```
void Button::initialButtonStyling_Widget (
    QObject * selectedWidget,
    uint8_t style_option,
    uint8_t size )
```

Estilização dos botões de um Widget escolhido.

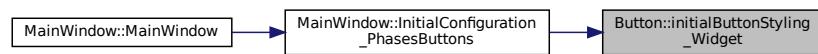
Parameters

<i>selectedWidget</i>	Widget escolhido.
<i>style_option</i>	Estilo do botão escolhido.
<i>size</i>	Tamanho do botão escolhido.

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.12 pressureButton_style()

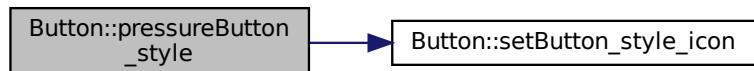
```
void Button::pressureButton_style ( QToolButton * pressure )
```

Estilização do botão da pressão.

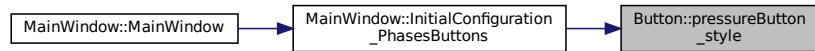
Parameters

<i>pressure</i>	Botão escolhido.
-----------------	------------------

Here is the call graph for this function:



Here is the caller graph for this function:



6.1.3.13 setButton_style_icon()

```

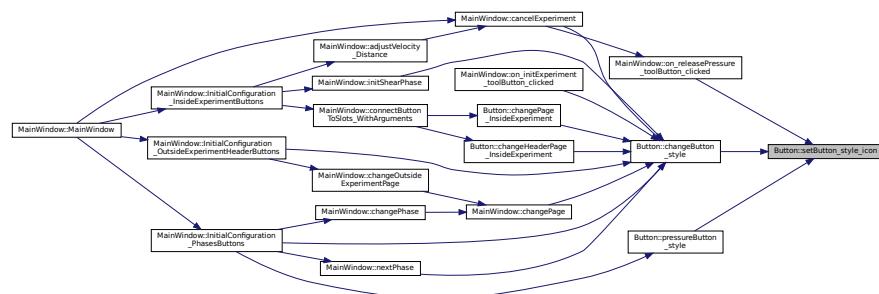
void Button::setButton_style_icon (
    QToolButton * button,
    uint8_t style,
    uint8_t icon )
  
```

Altera o ícone e estilo do botão.

Parameters

<i>button</i>	Botão que será alterado.
<i>style</i>	Estilo do botão, podendo ser claro ou escuro.
<i>icon</i>	Ícone do botão.

Here is the caller graph for this function:



6.1.3.14 setButtonShadow()

```

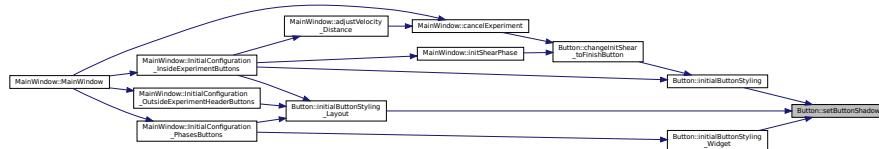
void Button::setButtonShadow (
    QToolButton * button )
  
```

Estiliza um QPushButton ou QToolButton para ter sombra.

Parameters

button O botão que será estilizado.

Here is the caller graph for this function:



6.1.3.15 styleSheetConfiguration()

```
void Button::styleSheetConfiguration ( ) [private]
```

Cria as folhas de estilo para os botões.

Here is the caller graph for this function:



6.1.4 Member Data Documentation

6.1.4.1 button_icons

```
QIcon Button::button_icons[20] [private]
```

Array com os ícones dos botões.

6.1.4.2 button images

```
QString Button::button_images[20] [private]
```

Array com as imagens dos botões.

6.1.4.3 **button_styleSheets**

```
QString Button::button_styleSheets[15] [private]
```

Array com os estilos claro e escudo.

6.1.4.4 **buttonMaximumSize**

```
QSize Button::buttonMaximumSize[10] [private]
```

Array com os tamanhos máximos dos botões.

6.1.4.5 **buttonMinimumSize**

```
QSize Button::buttonMinimumSize[10] [private]
```

Array com os tamanho mínimos dos botões.

6.1.4.6 **clickedButton**

```
QToolButton* Button::clickedButton[3] [private]
```

Botão do cabeçalho ou/e das etapas, que foi clicado.

6.1.4.7 **clickedButtonIcon**

```
uint8_t Button::clickedButtonIcon[3] [private]
```

Ícone do botão do cabeçalho ou/e de alguma subpágina que foi clicado.

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

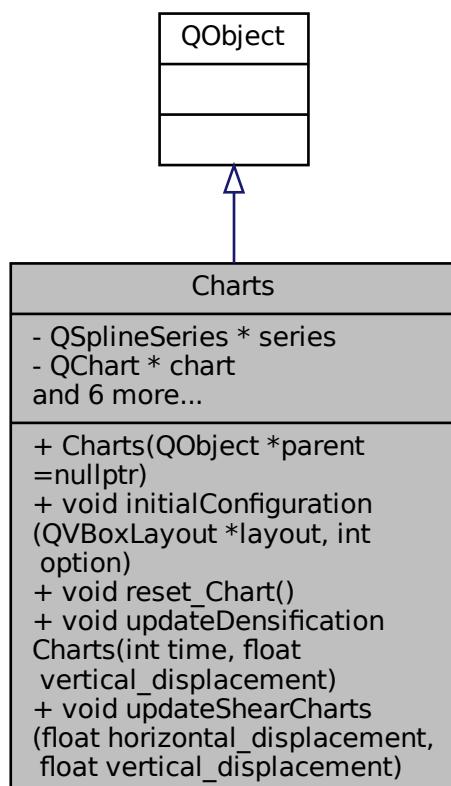
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[button.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[button.cpp](#)

6.2 Charts Class Reference

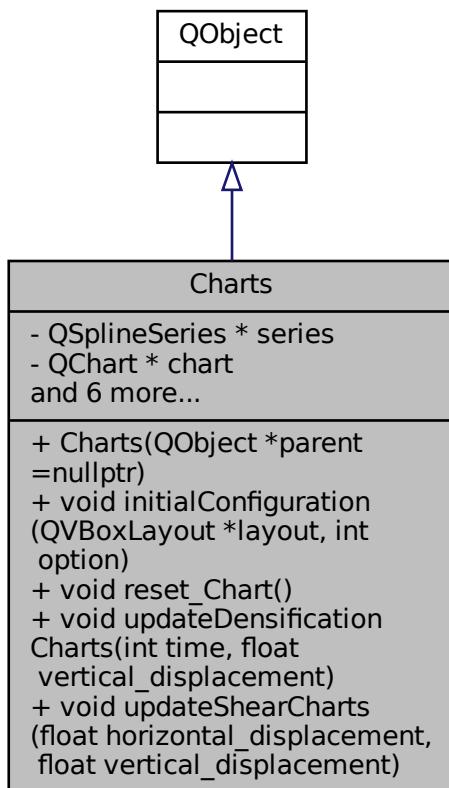
Classe responsável pelos gráficos.

```
#include <charts.h>
```

Inheritance diagram for Charts:



Collaboration diagram for Charts:



Public Slots

- void [updateDensificationCharts](#) (int time, float vertical_displacement)
Atualiza o gráfico de adensamento ao receber o sinal updateDensificationChart().
- void [updateShearCharts](#) (float horizontal_displacement, float vertical_displacement)
Atualiza o gráfico de cisalhamento ao receber o sinal updateShearChart().

Public Member Functions

- [Charts](#) (QObject *parent=nullptr)
Constrói uma nova instância da classe Charts.
- void [initialConfiguration](#) (QVBoxLayout *layout, int option)
Cria um gráfico e o insere em um layout.
- void [reset_Chart](#) ()
Apaga os gráficos atuais.

Private Attributes

- QSplineSeries * `series` [2]
- QChart * `chart` [2]
- QValueAxis * `m_axisX` [2]
- QValueAxis * `m_axisY` [2]
- QString `chart_title` [2]
- QString `x_axis_title` [2]
- QString `y_axis_title` [2]
- QChartView * `chartView` [2]

6.2.1 Detailed Description

Classe responsável pelos gráficos.

Esta classe é responsável por criar, estilizar e atualizar os gráficos.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 Charts()

```
Charts::Charts (
```

QObject * <i>parent</i> = nullptr) [explicit]
--

Constrói uma nova instância da classe `Charts`.

Define os títulos e nomes dos eixos dos gráficos.

Parameters

<i>parent</i>

6.2.3 Member Function Documentation

6.2.3.1 initialConfiguration()

```
void Charts::initialConfiguration (
```

QVBoxLayout * <i>layout</i> ,
int <i>option</i>)

Cria um gráfico e o insere em um layout.

Parameters

<i>layout</i>	O layout escolhido.
<i>option</i>	O tipo de gráfico escolhido.

Here is the caller graph for this function:



6.2.3.2 reset_Chart()

```
void Charts::reset_Chart ( )
```

Apaga os gráficos atuais.

Esta deleção dos gráficos atuais é utilizada quando o experimento se encerra ou é cancelado, previnindo assim, que os dados do experimento antigo apareçam no novo. Here is the caller graph for this function:



6.2.3.3 updateDensificationCharts

```
void Charts::updateDensificationCharts (
    int time,
    float vertical_displacement ) [slot]
```

Atualiza o gráfico de adensamento ao receber o sinal updateDensificationChart().

Parameters

<i>time</i>	Tempo decorrido
<i>vertical_displacement</i>	Deslocamento vertical

6.2.3.4 updateShearCharts

```
void Charts::updateShearCharts (
    float horizontal_displacement,
    float vertical_displacement ) [slot]
```

Atualiza o gráfico de cisalhamento ao receber o sinal updateShearChart().

Parameters

<i>horizontal_displacement</i>	Deslocamento horizontal
<i>vertical_displacement</i>	Deslocamento vertical

6.2.4 Member Data Documentation

6.2.4.1 chart

```
QChart* Charts::chart[2] [private]
```

Gráfico que recebe as séries e os eixos

6.2.4.2 chart_title

```
QString Charts::chart_title[2] [private]
```

Array com os títulos dos gráficos.

6.2.4.3 chartView

```
QChartView* Charts::chartView[2] [private]
```

Utilizado para inserir o gráfico em um layout.

6.2.4.4 m_axisX

```
QValueAxis* Charts::m_axisX[2] [private]
```

Array com os eixos X.

6.2.4.5 m_axisY

```
QValueAxis* Charts::m_axisY[2] [private]
```

Array com os eixos Y.

6.2.4.6 series

```
QSplineSeries* Charts::series[2] [private]
```

Cada elemento no array guarda os dados de algum gráfico.

6.2.4.7 x_axis_title

```
QString Charts::x_axis_title[2] [private]
```

Array com os títulos dos eixos x.

6.2.4.8 y_axis_title

```
QString Charts::y_axis_title[2] [private]
```

Array com os títulos dos eixos y.

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

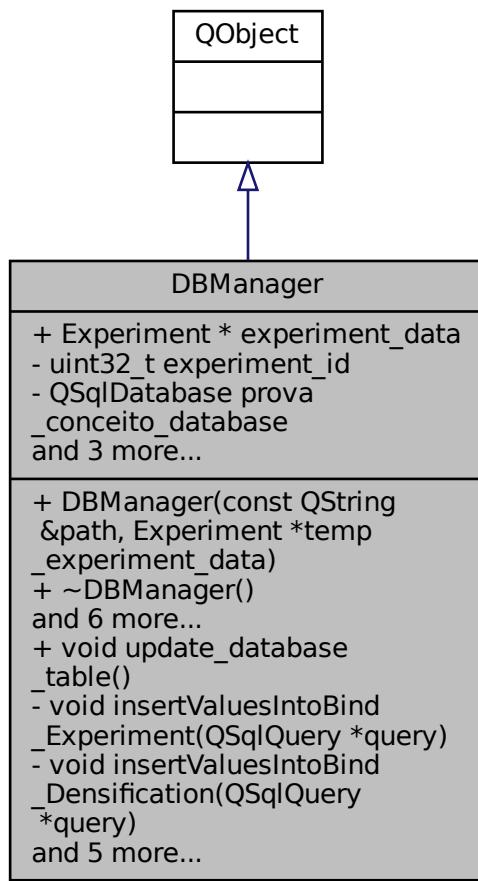
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[charts.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[charts.cpp](#)

6.3 DBManager Class Reference

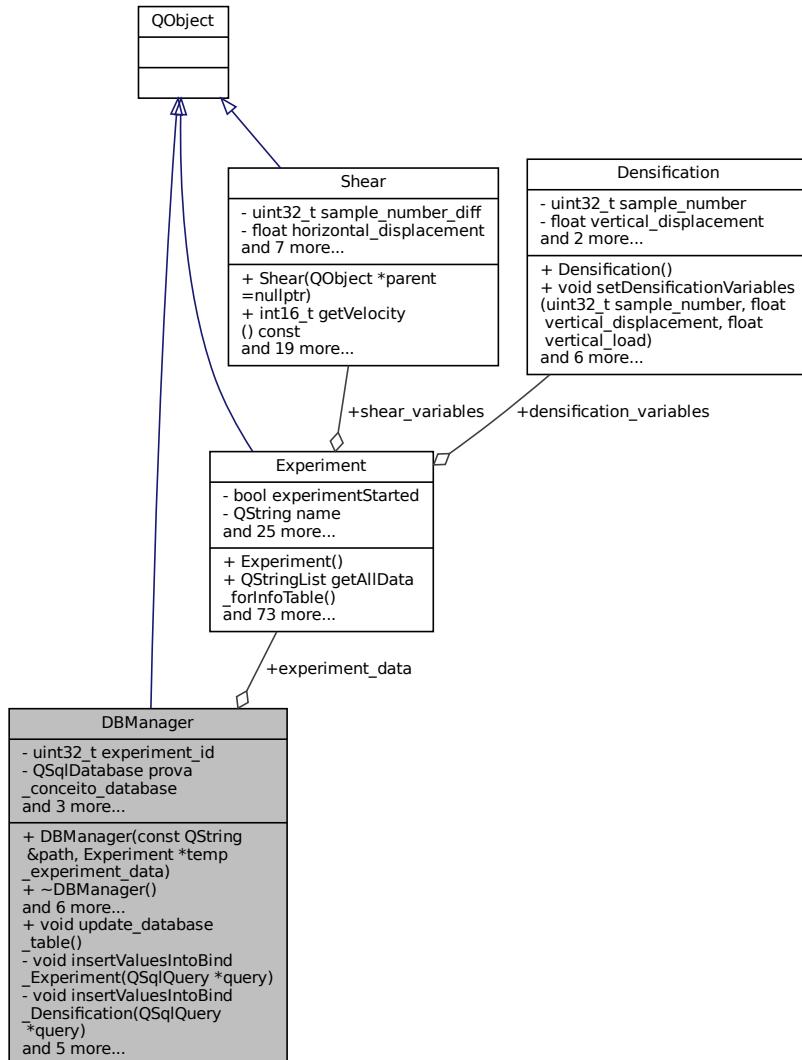
Classe do banco de dados.

```
#include <dbmanager.h>
```

Inheritance diagram for DBManager:



Collaboration diagram for DBManager:



Public Slots

- void [update_database_table \(\)](#)
Atualiza as tabelas de adensamento e cisalhamento.

Public Member Functions

- [DBManager \(const QString &path, Experiment *temp_experiment_data\)](#)
Constrói um novo objeto da classe `DBManager`.
- [~DBManager \(\)](#)
Fechá o banco de dados e destrói a instância dele.
- bool [isOpen \(\) const](#)
Verifica se o banco de dados está aberto.

- bool `createTable` (uint8_t option)
Cria uma tabela no banco de dados.
- bool `tableExists` (uint8_t option)
Verifica se a tabela existe no banco de dados.
- bool `insertIntoTable` (uint8_t option)
Insere dados em uma tabela escolhida.
- bool `updateTable` (uint8_t option)
Atualiza os dados de uma tabela escolhida.
- uint32_t `getExperiment_id` () const
`DBManager::getExperiment_id.`

Public Attributes

- Experiment * `experiment_data`

Private Member Functions

- void `insertValuesBind_Experiment` (QSqlQuery *query)
Insere valores no bind da inserção da tabela EXPERIMENT_TABLE.
- void `insertValuesBind_Densification` (QSqlQuery *query)
Insere valores no bind da inserção da tabela DENSIFICATION_TABLE.
- void `insertValuesBind_Shear` (QSqlQuery *query)
Insere valores no bind da inserção da tabela SHEAR_TABLE.
- void `insertValuesBind_FinalVariables` (QSqlQuery *query)
Insere valores no bind da inserção da tabela FINAL_VARIABLES_TABLE.
- void `insertValuesBind_SampleVariables` (QSqlQuery *query)
Insere valores no bind da inserção da tabela SAMPLE_VARIABLES_TABLE.
- void `insertValuesBind_SampleVariablesUpdate` (QSqlQuery *query)
Insere valores no bind da atualização da tabela SAMPLE_VARIABLES_TABLE.
- bool `selectExperimentId` ()
Insere o valor da variável this->experiment_id após consulta no banco.

Private Attributes

- uint32_t `experiment_id`
- QSqlDatabase `prova_conceito_database`
- QStringList `table_name` [5] = {"EXPERIMENT_TABLE","DENSIFICATION_TABLE","SHEAR_TABLE","FINAL_VARIABLES_TABLE","SAMPLE_TABLE"}
- QStringList `create_table` [5]
- QStringList `insert_into_table` [6]

6.3.1 Detailed Description

Classe do banco de dados.

Esta classe é responsável por administrar todas as operações relacionadas ao banco de dados.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 DBManager()

```
DBManager::DBManager (
    const QString & path,
    Experiment * temp_experiment_data )
```

Constrói um novo objeto da classe [DBManager](#).

Esta classe cria o banco de dados, se ele não existir, e insere os valores nas variáveis `this->create_table` e `this->insert_into_table`.

Parameters

<i>path</i>	Nome do banco de dados com o caminho. Exemplo: <code>home/user/databaseExperiment</code>
<i>temp_experiment_data</i>	Ponteiro para a intância da classe Experiment inicializada na classe MainWindow .

6.3.2.2 ~DBManager()

```
DBManager::~DBManager ( )
```

Fecha o banco de dados e destrói a instância dele.

O arquivo do banco de dados não é detruído.

6.3.3 Member Function Documentation

6.3.3.1 createTable()

```
bool DBManager::createTable (
    uint8_t option )
```

Cria uma tabela no banco de dados.

Parameters

<i>option</i>	Qual tabela será criada.
---------------	--------------------------

Returns

true Se a operação foi bem sucedida.
false Se a operação falhou.

Here is the caller graph for this function:



6.3.3.2 getExperiment_id()

```
uint32_t DBManager::getExperiment_id ( ) const  
DBManager::getExperiment_id.
```

Returns

experiment_id da Tabela EXPERIMENT_TABLE no banco

6.3.3.3 insertIntoTable()

```
bool DBManager::insertIntoTable (   
        uint8_t option )
```

Insere dados em uma tabela escolhida.

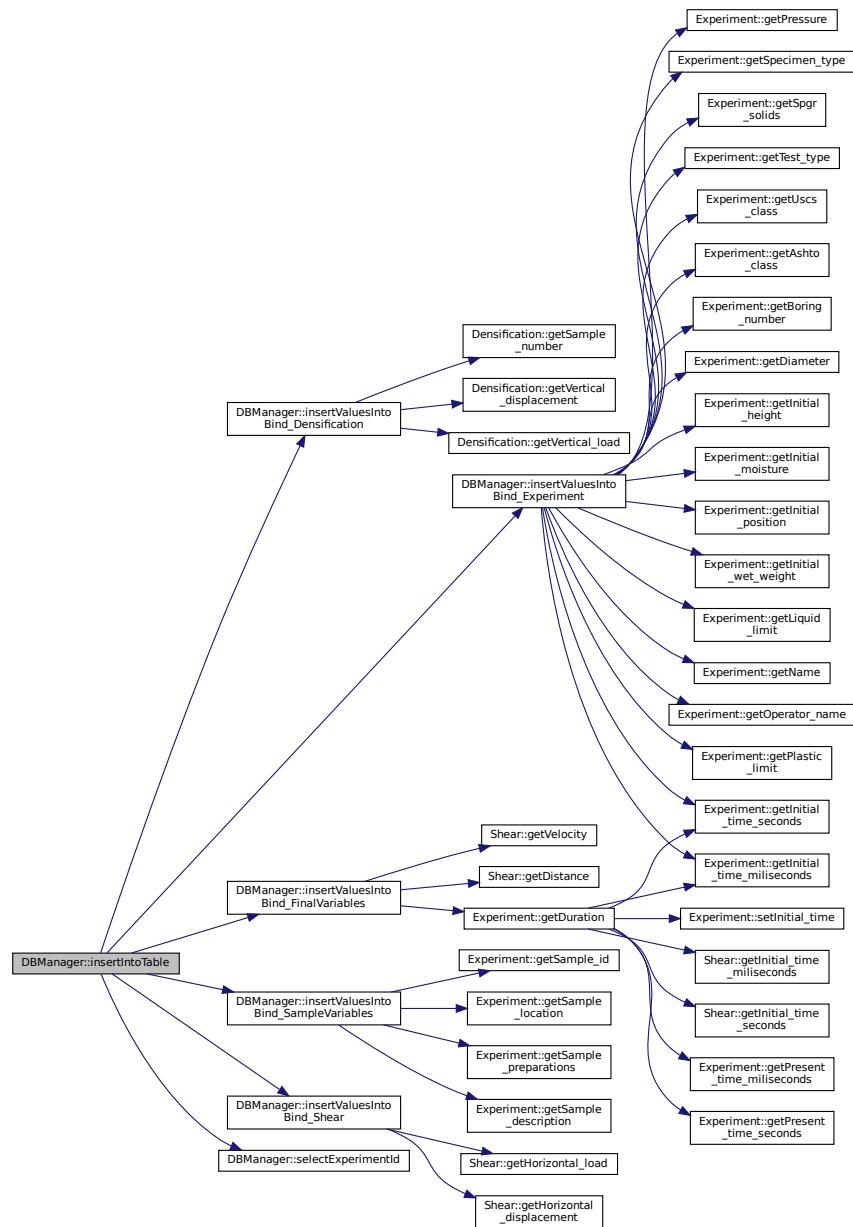
Parameters

<i>option</i>	A tabela escolhida.
---------------	---------------------

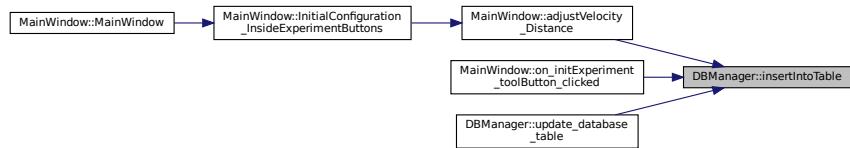
Returns

true Se a operação foi bem sucedida;
 false Se a operação falhou.

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.4 `insertValuesIntoBind_Densification()`

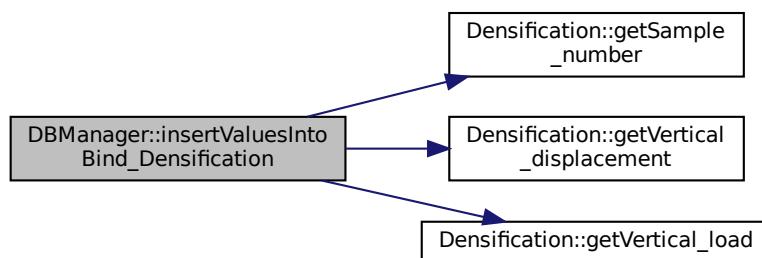
```
void DBManager::insertValuesIntoBind_Densification (
    QSqlQuery * query ) [private]
```

Insere valores no bind da inserção da tabela DENSIFICATION_TABLE.

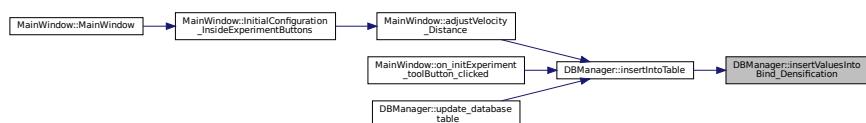
Parameters

<code>query</code>	Variável utilizada para executar operações SQL.
--------------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.5 insertValuesIntoBind_Experiment()

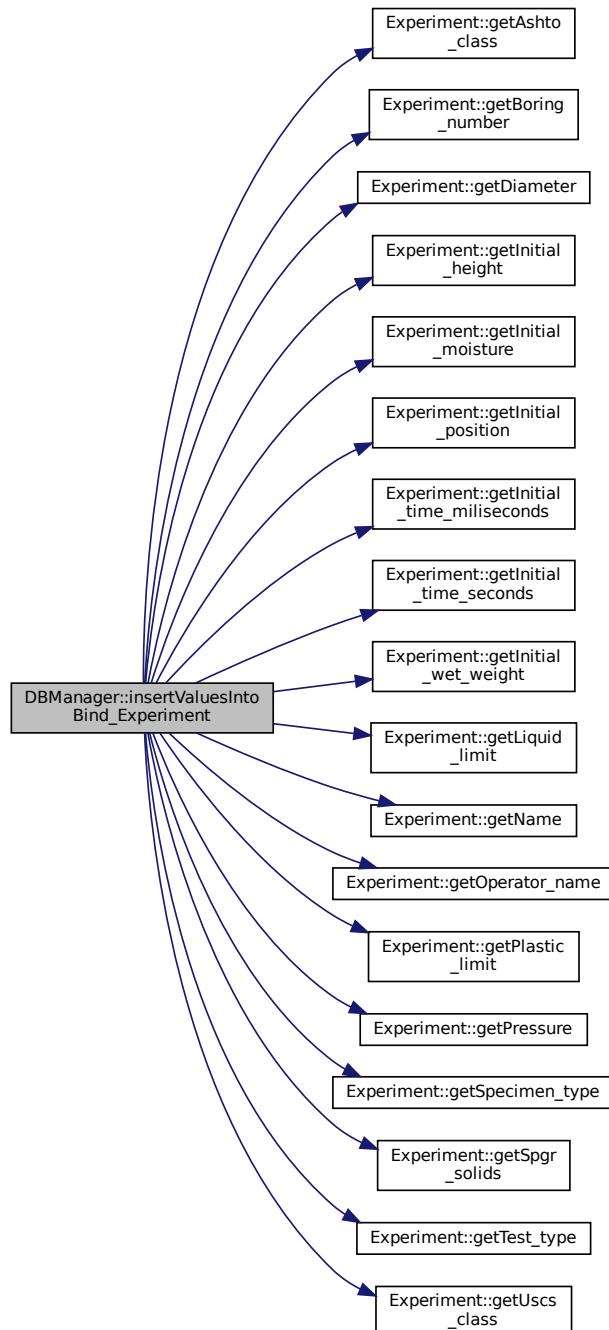
```
void DBManager::insertValuesIntoBind_Experiment (
    QSqlQuery * query ) [private]
```

Insere valores no bind da inserção da tabela EXPERIMENT_TABLE.

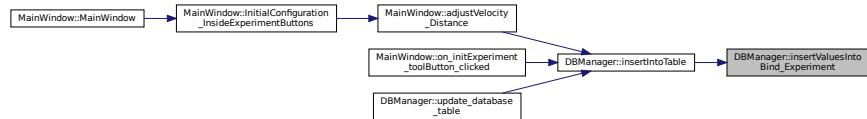
Parameters

<i>query</i>	Variável utilizada para executar operações SQL.
--------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.6 insertValuesIntoBind_FinalVariables()

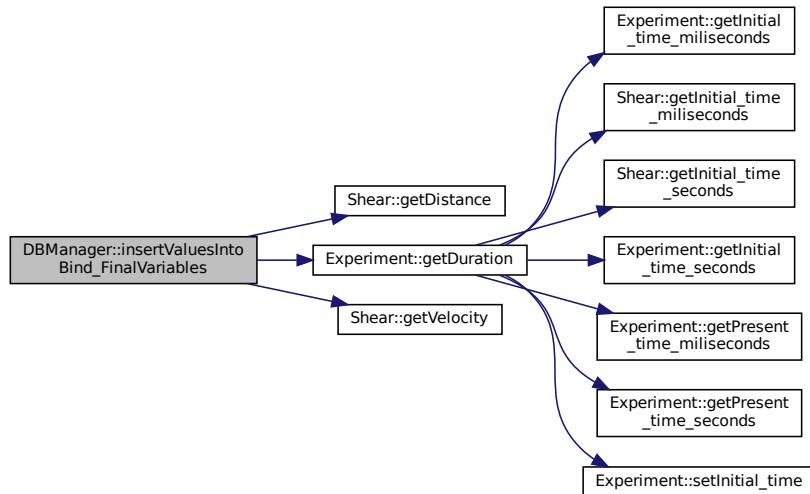
```
void DBManager::insertValuesIntoBind_FinalVariables (
    QSqlQuery * query ) [private]
```

Insere valores no bind da inserção da tabela FINAL_VARIABLES_TABLE.

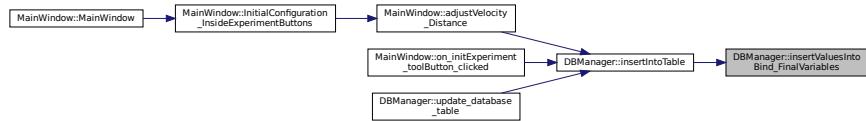
Parameters

<i>query</i>	Variável utilizada para executar operações SQL.
--------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.7 insertValuesIntoBind_SampleVariables()

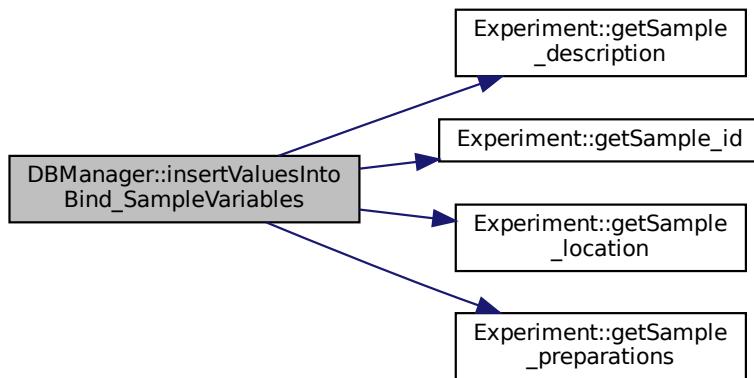
```
void DBManager::insertValuesIntoBind_SampleVariables (
    QSqlQuery * query ) [private]
```

Insere valores no bind da inserção da tabela SAMPLE_VARIABLES_TABLE.

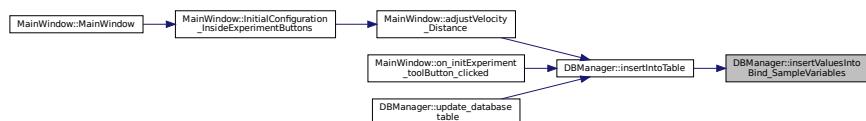
Parameters

<i>query</i>	Variável utilizada para executar operações SQL.
--------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.8 insertValuesIntoBind_SampleVariablesUpdate()

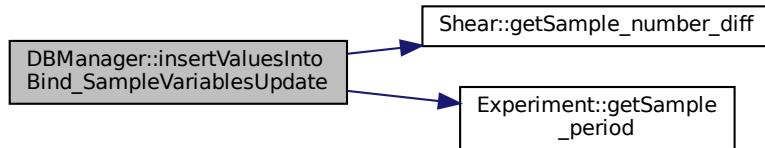
```
void DBManager::insertValuesIntoBind_SampleVariablesUpdate (
    QSqlQuery * query ) [private]
```

Insere valores no bind da atualização da tabela SAMPLE_VARIABLES_TABLE.

Parameters

<i>query</i>	Variável utilizada para executar operações SQL.
--------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.9 insertValuesIntoBind_Shear()

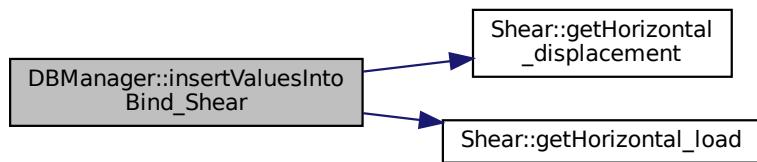
```
void DBManager::insertValuesIntoBind_Shear (
    QSqlQuery * query ) [private]
```

Insere valores no bind da inserção da tabela SHEAR_TABLE.

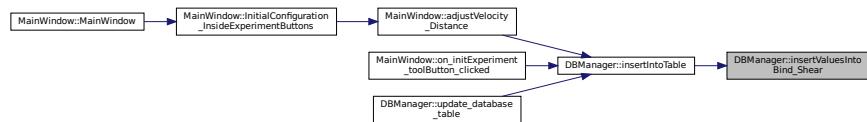
Parameters

<i>query</i>	Variável utilizada para executar operações SQL.
--------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.3.10 isOpen()

```
bool DBManager::isOpen( ) const
```

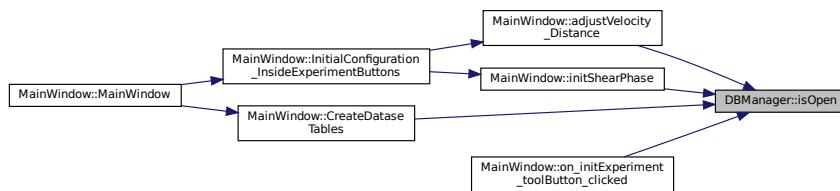
Verifica se o banco de dados está aberto.

Returns

true Caso esteja aberto.

false Caso esteja fechado.

Here is the caller graph for this function:



6.3.3.11 selectExperimentId()

```
bool DBManager::selectExperimentId ( ) [private]
```

Insere o valor da variável this->experiment_id após consulta no banco.

Array com as string utilizadas para inserir dados nas tabelas.

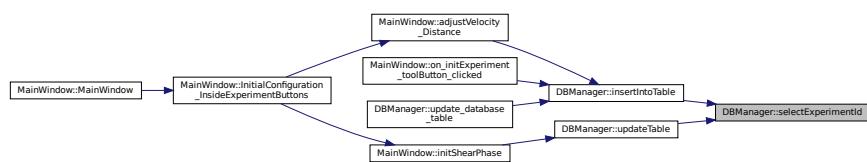
Esta função captura o maior valor do experiment_id da tabela EXPERIMENT_TABLE, e o utiliza como chave estrangeira nas outras tabelas. Esse esquema funciona atualmente, pois a chave primária da classe EXPERIMENT_TABLE é gerada automaticamente e é incremental, 1,2,3,4... e assim por diante.

Returns

true Se a operação foi bem sucedida.

false Se a operação falhou.

Here is the caller graph for this function:



6.3.3.12 tableExists()

```
bool DBManager::tableExists (
    uint8_t option )
```

Verifica se a tabela existe no banco de dados.

Parameters

<i>option</i>	A tabela escolhida.
---------------	---------------------

Returns

true Se a operação foi bem sucedida.
false Se a operação falhou.

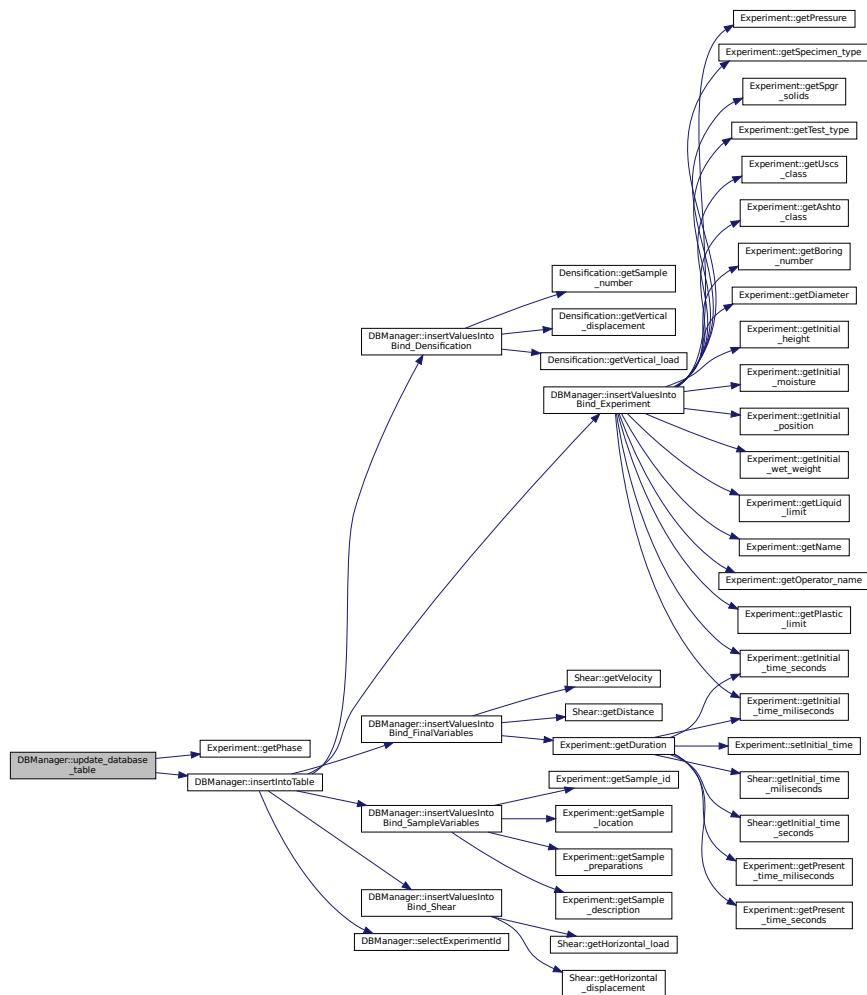
Here is the caller graph for this function:

**6.3.3.13 update_database_table**

```
void DBManager::update_database_table ( ) [slot]
```

Atualiza as tabelas de adensamento e cisalhamento.

Recebe o sinal `data_arrived()` da classe `receiveData` para executar a atualização. Here is the call graph for this function:



6.3.3.14 updateTable()

```
bool DBManager::updateTable (
    uint8_t option )
```

Atualiza os dados de uma tabela escolhida.

Parameters

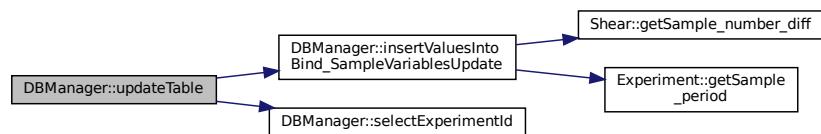
<i>option</i>	A tabela escolhida.
---------------	---------------------

Returns

true Se a operação foi bem sucedida.

false Se a operação falhou.

Here is the call graph for this function:



Here is the caller graph for this function:



6.3.4 Member Data Documentation

6.3.4.1 create_table

```
QString DBManager::create_table[5] [private]
```

Array com as strings utilizadas para criar as tabelas.

6.3.4.2 experiment_data

```
Experiment* DBManager::experiment_data
```

Instância da classe [Experiment](#) utilizada para atualizar/inserir dados nas tabelas.

6.3.4.3 experiment_id

```
uint32_t DBManager::experiment_id [private]
```

Id do experimento utilizado para preencher a chave estrangeira em outras tabelas.

6.3.4.4 insert_into_table

```
QString DBManager::insert_into_table[6] [private]
```

6.3.4.5 prova_conceito_database

```
QSqlDatabase DBManager::prova_conceito_database [private]
```

Variável do banco de dados.

6.3.4.6 table_name

```
QString DBManager::table_name[5] = {"EXPERIMENT_TABLE", "DENSIFICATION_TABLE", "SHEAR_TABLE", "FINAL←  
_VARIABLES_TABLE", "SAMPLE_TABLE"} [private]
```

Array com o nome das tabelas.

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

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[dbmanager.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[dbmanager.cpp](#)

6.4 Densification Class Reference

Classe do adensamento.

```
#include <densification.h>
```

Collaboration diagram for Densification:

Densification
<ul style="list-style-type: none"> - uint32_t sample_number - float vertical_displacement and 2 more...
<ul style="list-style-type: none"> + Densification() + void setDensificationVariables(uint32_t sample_number, float vertical_displacement, float vertical_load) and 6 more...

Public Member Functions

- **Densification ()**
Constrói uma nova instância da classe objeto.
- void **setDensificationVariables (uint32_t sample_number, float vertical_displacement, float vertical_load)**
Insere os dados recebidos nas variáveis da classe this->sample_number, this->vertical_displacement e this->vertical_load.
- QStringList **getDensificationVariables ()**
Retorna alguns dos valores utilizados na tabela de adensamento.
- QStringList **getDensificationResults ()**
Retorna alguns dos valores utilizados na tabela de resultados do adensamento.
- uint32_t **getSample_number () const**
Retorna o valor do número de amostra atual.
- float **getVertical_displacement () const**
Retorna o valor do deslocamento vertical.
- float **getVertical_load () const**
Retorna o valor da carga vertical.
- void **setDiff_sampleNumber_initExperiment (uint32_t diff)**
Define o valor de this->diff_sampleNumber_initExperiment.

Private Attributes

- uint32_t **sample_number**
- float **vertical_displacement**
- float **vertical_load**
- uint32_t **diff_sampleNumber_initExperiment**

Último número de amostra antes depois do posicionamento inicial.

6.4.1 Detailed Description

Classe do adensamento.

Esta classe é responsável por administrar as operações e dados relacionados ao adensamento no experimento.

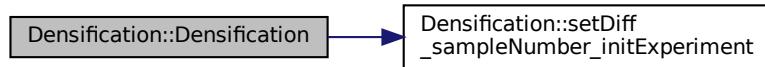
6.4.2 Constructor & Destructor Documentation

6.4.2.1 Densification()

```
Densification::Densification ( )
```

Constrói uma nova instância da classe objeto.

Inicializa o sample_number e o diff_sampleNumber_initExperiment com o valor 0. Here is the call graph for this function:



6.4.3 Member Function Documentation

6.4.3.1 getDensificationResults()

```
QStringList Densification::getDensificationResults ( )
```

Retorna alguns dos valores utilizados na tabela de resultados do adensamento.

Returns

QStringList Lista de QStrings com os valores.

Here is the caller graph for this function:



6.4.3.2 getDensificationVariables()

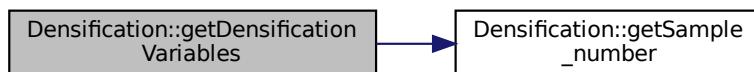
```
QStringList Densification::getDensificationVariables ( )
```

Retorna alguns dos valores utilizados na tabela de adensamento.

Returns

QStringList Lista do tipo QString que guarda os valores.

Here is the call graph for this function:



Here is the caller graph for this function:



6.4.3.3 getSample_number()

```
uint32_t Densification::getSample_number ( ) const
```

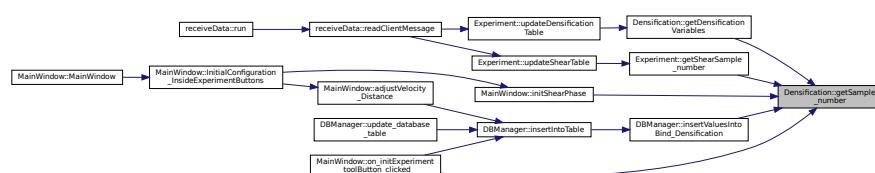
Retorna o valor do número de amostra atual.

A máquina sempre manda o número de amostra incrementando, 1,2,3,4 ... e assim por diante. Durante o posicionamento inicial o número de amostra é incrementado e assim, para o adensamento começar com o número de amostra 0 é necessário realizar uma diferença entre o número atual e este número.

Returns

uint32_t Número da amostra.

Here is the caller graph for this function:



6.4.3.4 getVertical_displacement()

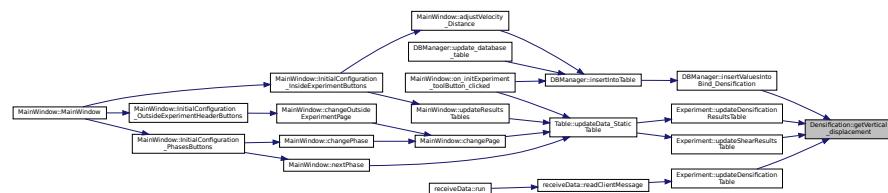
```
float Densification::getVertical_displacement ( ) const
```

Retorna o valor do deslocamento vertical.

Returns

float Valor do deslocamento vertical atual.

Here is the caller graph for this function:



6.4.3.5 getVertical_load()

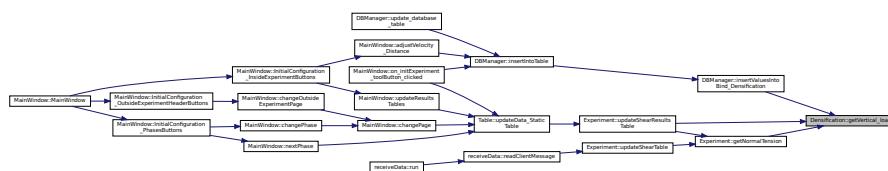
```
float Densification::getVertical_load ( ) const
```

Retorna o valor da carga vertical.

Returns

float Valor da carga vertical atual.

Here is the caller graph for this function:



6.4.3.6 setDensificationVariables()

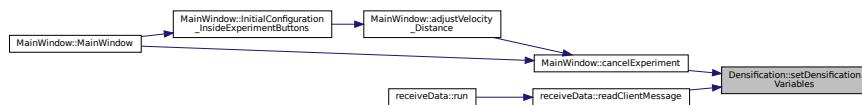
```
void Densification::setDensificationVariables (
    uint32_t sample_number,
    float vertical_displacement,
    float vertical_load )
```

Insere os dados recebidos nas variáveis da classe this->sample_number, this->vertical_displacement e this->vertical_load.

Parameters

<i>sample_number</i>	Número da amostra recebido pela camada de controle.
<i>vertical_displacement</i>	Deslocamento vertical recebido pela camada de controle.
<i>vertical_load</i>	Carga vertical recebido pela camada de controle.

Here is the caller graph for this function:



6.4.3.7 setDiff_sampleNumber_initExperiment()

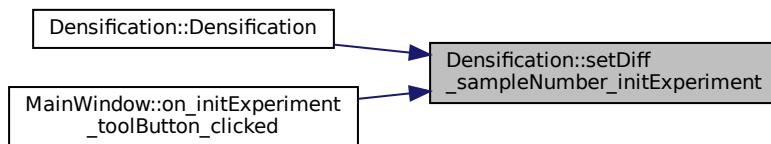
```
void Densification::setDiff_sampleNumber_initExperiment (
    uint32_t diff )
```

Define o valor de this->diff_sampleNumber_initExperiment.

Parameters

<i>diff</i>	Valor definido.
-------------	-----------------

Here is the caller graph for this function:



6.4.4 Member Data Documentation

6.4.4.1 diff_sampleNumber_initExperiment

```
uint32_t Densification::diff_sampleNumber_initExperiment [private]
```

Último número de amostra antes depois do posicionamento inicial.

A máquina sempre manda o número de amostra incrementando, 1,2,3,4 ... e assim por diante. Durante o posicionamento inicial o número de amostra é incrementado e assim, para o adensamento começar com o número de amostra 0 é necessário realizar uma diferença entre o número atual e este número.

6.4.4.2 sample_number

```
uint32_t Densification::sample_number [private]
```

Número da amostra atual.

6.4.4.3 vertical_displacement

```
float Densification::vertical_displacement [private]
```

Deslocamento vertical.

6.4.4.4 vertical_load

```
float Densification::vertical_load [private]
```

Carga vertical.

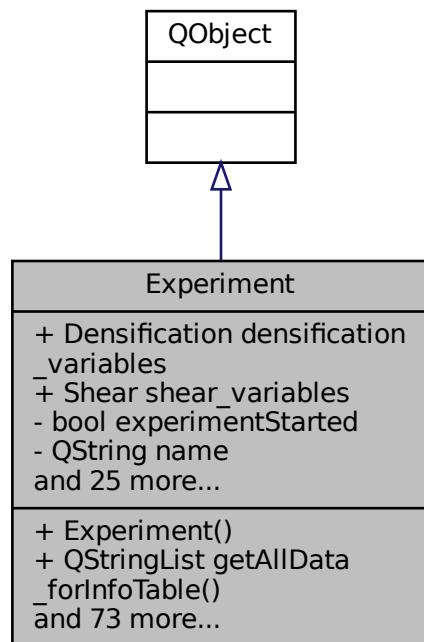
The documentation for this class was generated from the following files:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[densification.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[densification.cpp](#)

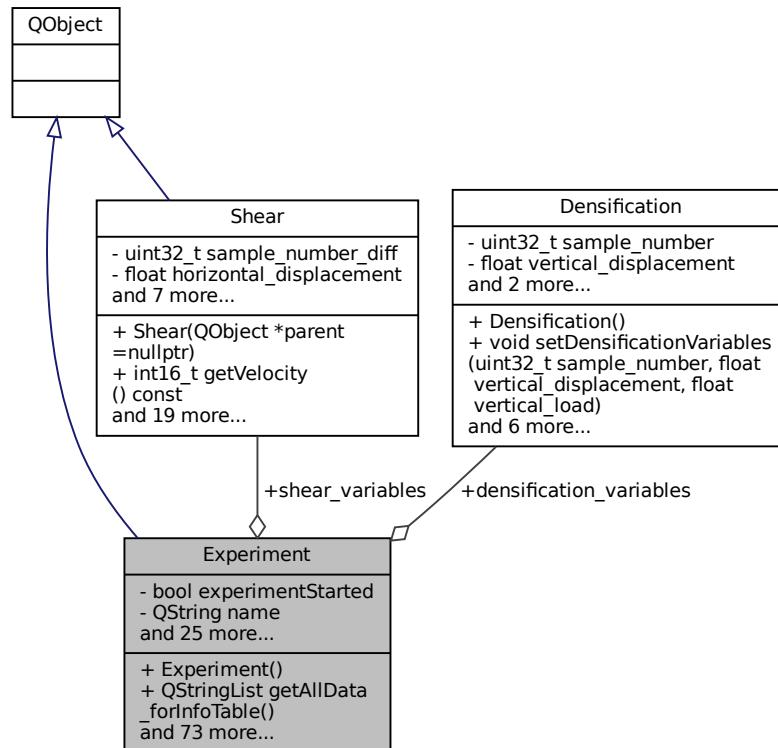
6.5 Experiment Class Reference

```
#include <experiment.h>
```

Inheritance diagram for Experiment:



Collaboration diagram for Experiment:



Signals

- void `updateDensificationChart` (int time, float vertical_displacement)
- void `updateShearChart` (float horizontal_displacement, float vertical_displacement)

Public Member Functions

- `Experiment ()`
- `QStringList getAllData_forInfoTable ()`
- `QStringList getAllData_forPhasesTable ()`
- `QStringList updateDensificationTable ()`
- `QStringList updateDensificationResultsTable ()`
- `QStringList updateShearTable ()`
- `QStringList updateShearResultsTable ()`
- `void insertData_inDatabase ()`
- `const QString & getName () const`
- `void setName (const QString &newName)`
- `const QString & getOperator_name () const`
- `void setOperator_name (const QString &newOperator_name)`
- `const QString & getTest_type () const`
- `void setTest_type (const QString &newTest_type)`
- `const QString & getSpecimen_type () const`

- void `setSpecimen_type` (const QString &newSpecimen_type)
- const QString & `getUscs_class` () const
- void `setUscs_class` (const QString &newUscs_class)
- const QString & `getAshto_class` () const
- void `setAshto_class` (const QString &newAshto_class)
- const QString & `getSample_preparations` () const
- void `setSample_preparations` (const QString &newSample_preparations)
- int `getSample_id` () const
- void `setSample_id` (int newSample_id)
- int `getBoring_number` () const
- void `setBoring_number` (int newBoring_number)
- const QString & `getSample_location` () const
- void `setSample_location` (const QString &newSample_location)
- const QString & `getSample_description` () const
- void `setSample_description` (const QString &newSample_description)
- float `getInitial_height` () const
- void `setInitial_height` (float newInitial_height)
- float `getInitial_wet_weight` () const
- void `setInitial_wet_weight` (float newInitial_wet_weight)
- float `getInitial_moisture` () const
- void `setInitial_moisture` (float newInitial_moisture)
- float `getSpgr_solids` () const
- void `setSpgr_solids` (float newSpgr_solids)
- float `getPlastic_limit` () const
- void `setPlastic_limit` (float newPlastic_limit)
- float `getLiquid_limit` () const
- void `setLiquid_limit` (float newLiquid_limit)
- float `getInitial_position` () const
- void `setInitial_position` (float newInitial_position)
- float `getDiameter` () const
- void `setDiameter` (float newDiameter)
- float `getArea` ()
- float `getInitial_volume` ()
- float `getInitial_wet_density` ()
- float `getInitial_dry_density` ()
- float `getInitial_void_ratio` ()
- float `getwater_specific_weight` ()
- float `getInitial_saturation` ()
- float `getPressure` () const
- void `setPressure` (float newPressure)
- int `getPhase` () const
- void `changePhase` ()
- uint64_t `getDuration` (bool isDensification)
- uint64_t `getPresent_time_seconds` () const
- uint64_t `getInitial_time_seconds` () const
- uint64_t `getPresent_time_milliseconds` () const
- uint64_t `getInitial_time_milliseconds` () const
- void `setInitial_time` (bool isInitial)
- QString `getInitial_timeString` ()
- QString `hour_min_sec_ms` ()
- QString `day_month_year` ()
- bool `getExperimentStarted` () const
- void `setExperimentStarted` (bool newExperimentStarted)
- QString `getDuration_string` (bool isDensification)
- int `getSample_period` () const

- void `setSample_period` (int newSample_period)
- uint32_t `getShearSample_number` () const
- float `getNormalTension` ()
- float `getShearTension` ()
- float `getAverageSpeed` ()

Public Attributes

- Densification `densification_variables`
- Shear `shear_variables`

Private Attributes

- bool `experimentStarted` = false
- QString `name`
- QString `operator_name`
- QString `test_type`
- QString `specimen_type`
- QString `uscs_class`
- QString `ashto_class`
- QString `sample_preparations`
- int `sample_id`
- int `boring_number`
- QString `sample_location`
- QString `sample_description`
- float `initial_height`
- float `initial_wet_weight`
- float `initial_moisture`
- float `spgr_solids`
- float `plastic_limit`
- float `liquid_limit`
- float `initial_position`
- float `diameter`
- float `pressure`
- uint64_t `initial_time_seconds`
- uint64_t `present_time_seconds`
- uint64_t `initial_time_milliseconds`
- uint64_t `present_time_milliseconds`
- int `phase` = `densification_phase`
- int `sample_period`

6.5.1 Constructor & Destructor Documentation

6.5.1.1 Experiment()

```
Experiment::Experiment ( )
```

6.5.2 Member Function Documentation

6.5.2.1 changePhase()

```
void Experiment::changePhase ( )
```

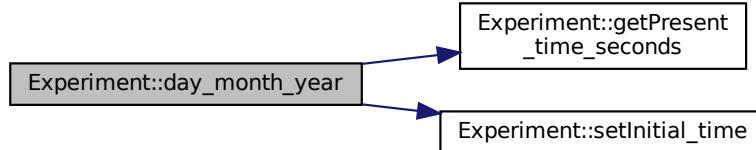
Here is the caller graph for this function:



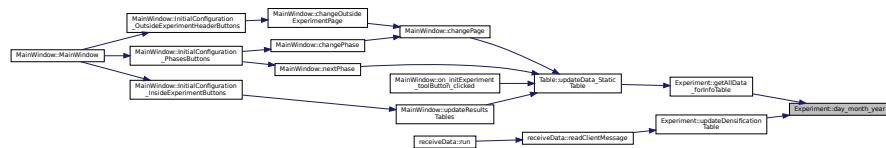
6.5.2.2 day_month_year()

```
QString Experiment::day_month_year ( )
```

Here is the call graph for this function:



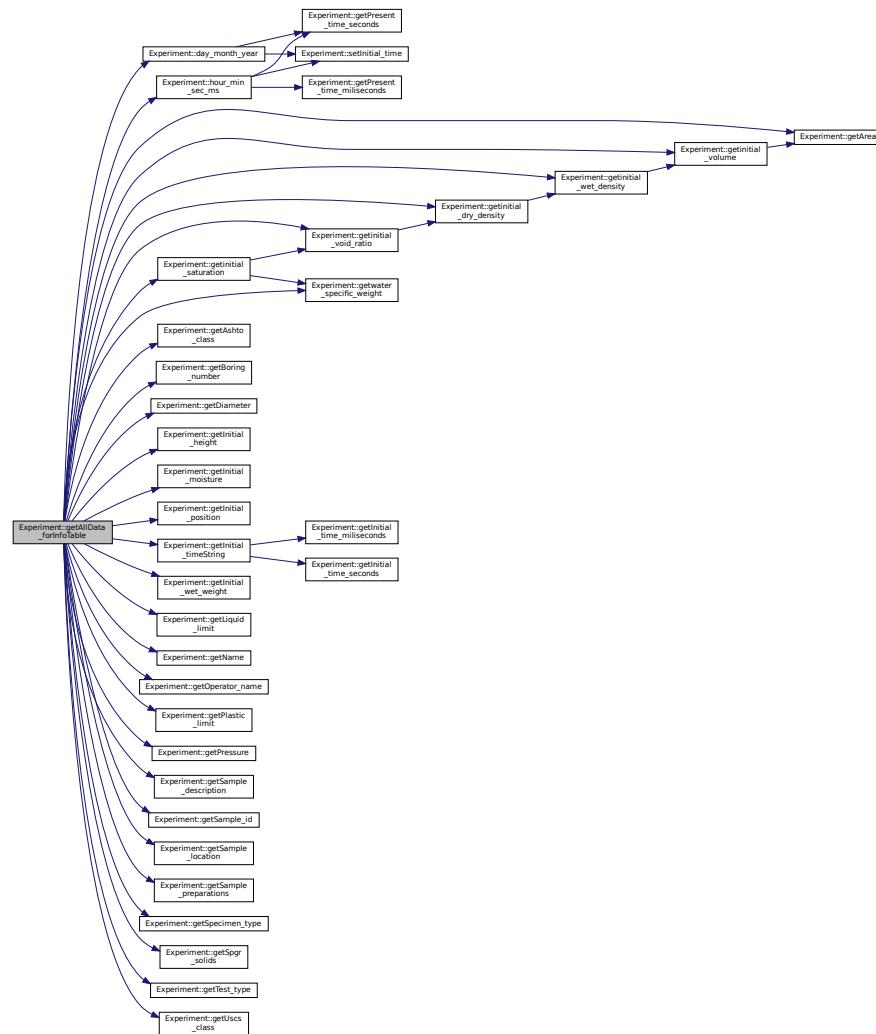
Here is the caller graph for this function:



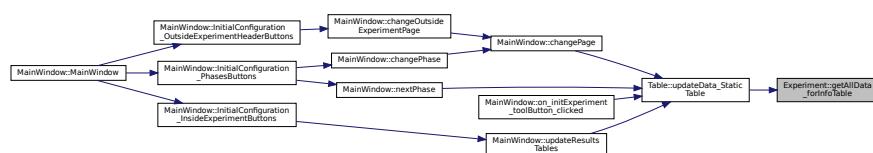
6.5.2.3 getAllData_forInfoTable()

```
QStringList Experiment::getAllData_forInfoTable ( )
```

Here is the call graph for this function:



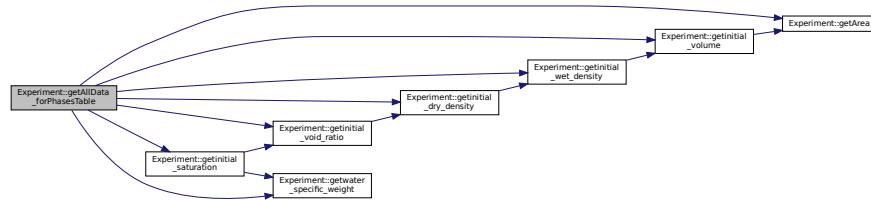
Here is the caller graph for this function:



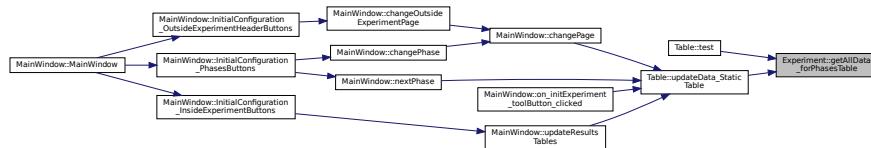
6.5.2.4 getAllData_forPhasesTable()

```
QStringList Experiment::getAllData_forPhasesTable ( )
```

Here is the call graph for this function:



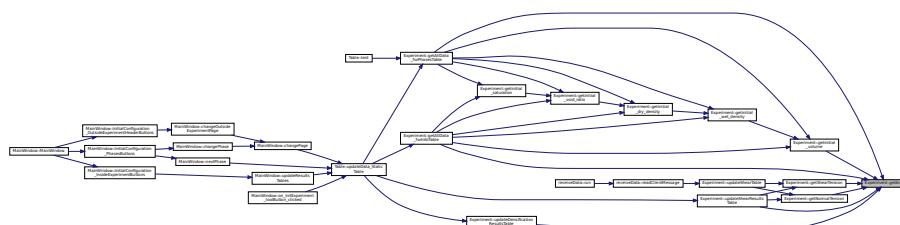
Here is the caller graph for this function:



6.5.2.5 getArea()

```
float Experiment::getArea ( )
```

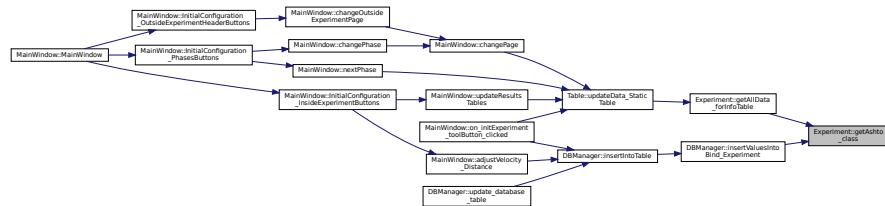
Here is the caller graph for this function:



6.5.2.6 getAshto_class()

```
const QString & Experiment::getAshto_class() const
```

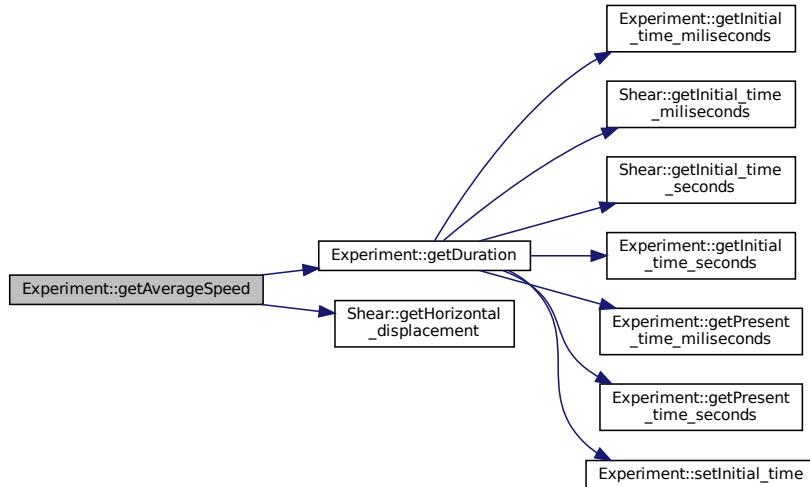
Here is the caller graph for this function:



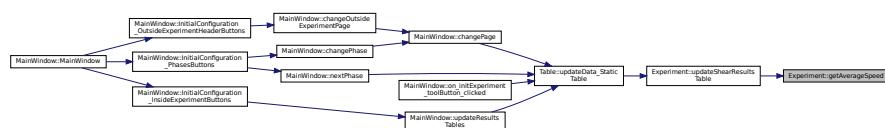
6.5.2.7 getAverageSpeed()

```
float Experiment::getAverageSpeed() const
```

Here is the call graph for this function:



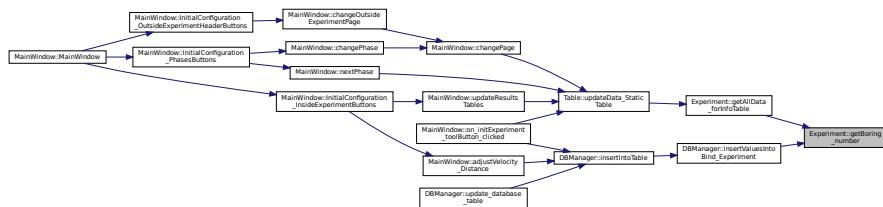
Here is the caller graph for this function:



6.5.2.8 getBoring_number()

```
int Experiment::getBoring_number ( ) const
```

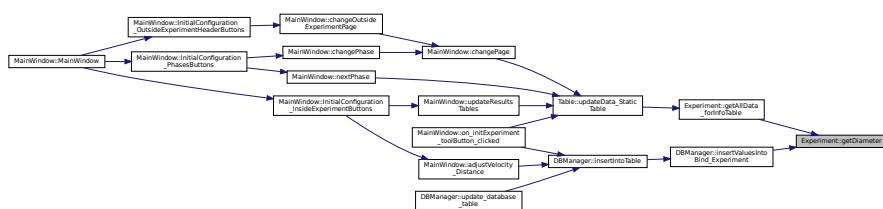
Here is the caller graph for this function:



6.5.2.9 getDiameter()

```
float Experiment::getDiameter ( ) const
```

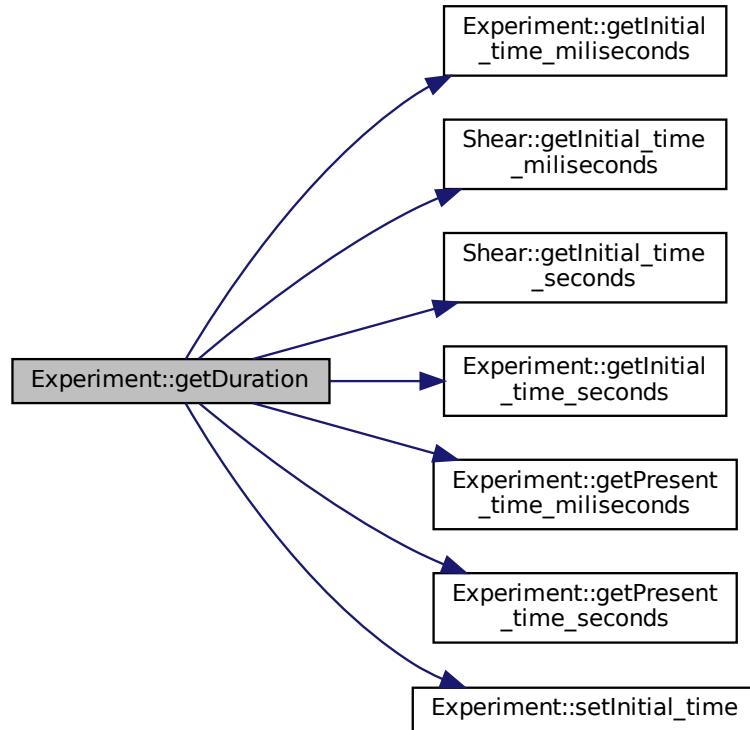
Here is the caller graph for this function:



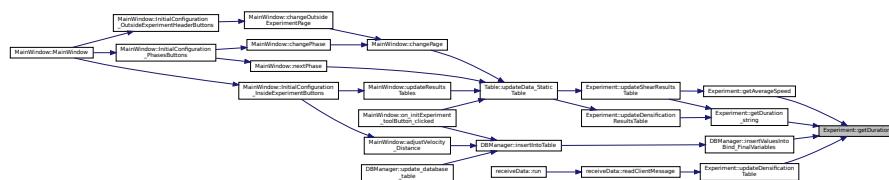
6.5.2.10 getDuration()

```
uint64_t Experiment::getDuration ( <br> bool isDensification )
```

Here is the call graph for this function:



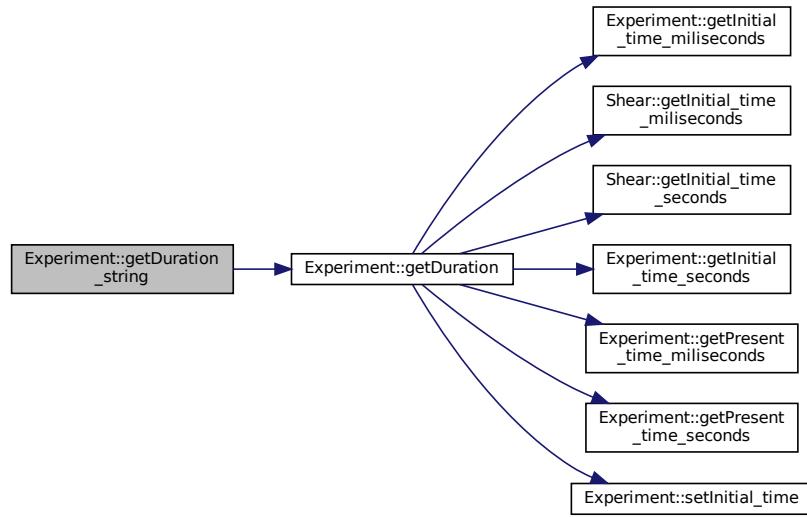
Here is the caller graph for this function:



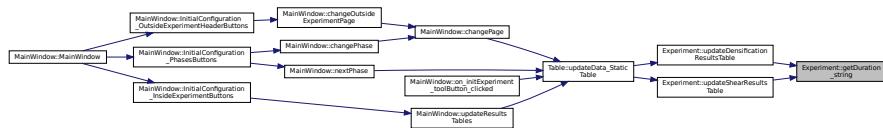
6.5.2.11 `getDuration_string()`

```
QString Experiment::getDuration_string (
    bool isDensification )
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.5.2.12 getExperimentStarted()

```
bool Experiment::getExperimentStarted ( ) const
```

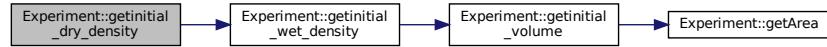
Here is the caller graph for this function:



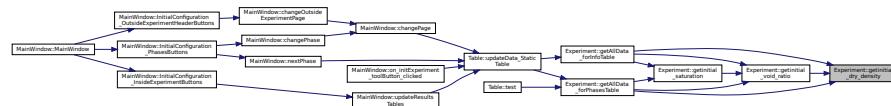
6.5.2.13 getinitial_dry_density()

```
float Experiment::getinitial_dry_density ( )
```

Here is the call graph for this function:



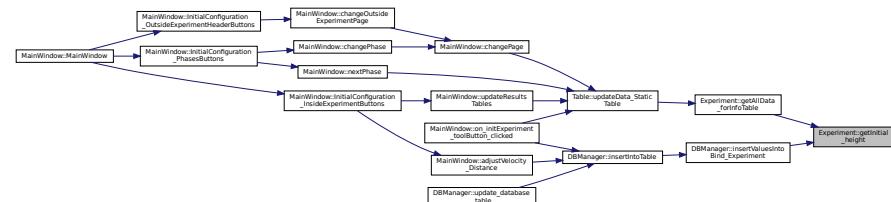
Here is the caller graph for this function:



6.5.2.14 getInitial_height()

```
float Experiment::getInitial_height ( ) const
```

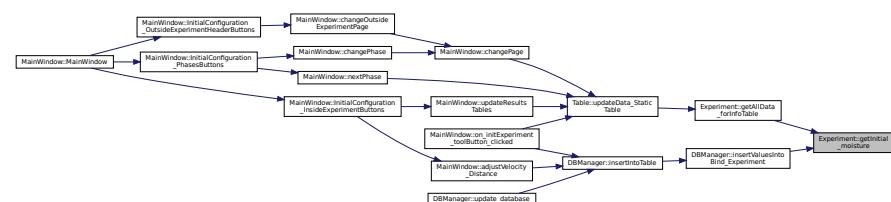
Here is the caller graph for this function:



6.5.2.15 getInitial_moisture()

```
float Experiment::getInitial_moisture ( ) const
```

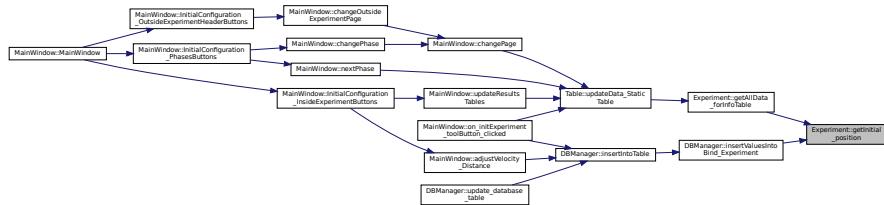
Here is the caller graph for this function:



6.5.2.16 getInitial_position()

```
float Experiment::getInitial_position ( ) const
```

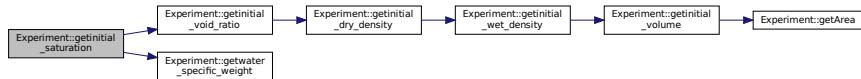
Here is the caller graph for this function:



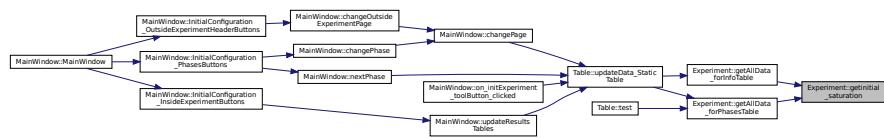
6.5.2.17 `getinitial_saturation()`

```
float Experiment::getinitial_saturation ( )
```

Here is the call graph for this function:



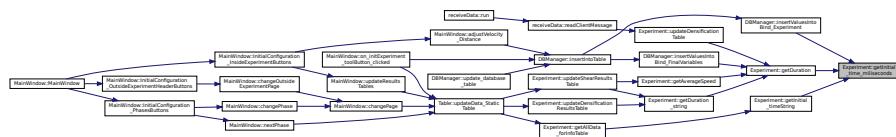
Here is the caller graph for this function:



6.5.2.18 getInitial_time_milliseconds()

```
uint64_t Experiment::getInitial_time_milliseconds ( ) const
```

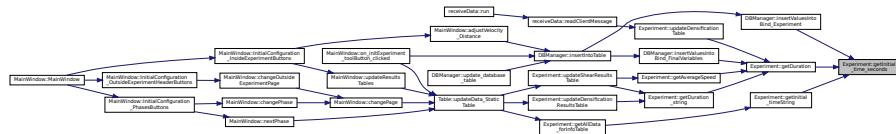
Here is the caller graph for this function:



6.5.2.19 getInitial_time_seconds()

```
uint64_t Experiment::getInitial_time_seconds ( ) const
```

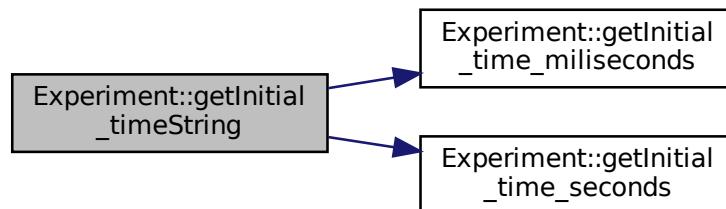
Here is the caller graph for this function:



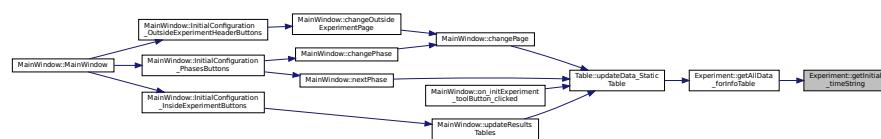
6.5.2.20 getInitialTimeString()

```
QString Experiment::getInitialTimeString ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



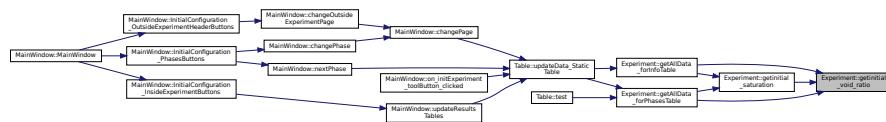
6.5.2.21 getinitial_void_ratio()

```
float Experiment::getinitial_void_ratio ( )
```

Here is the call graph for this function:



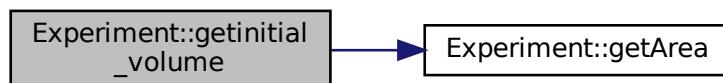
Here is the caller graph for this function:



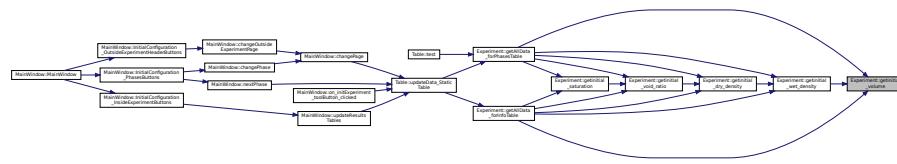
6.5.2.22 getinitial_volume()

```
float Experiment::getinitial_volume ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



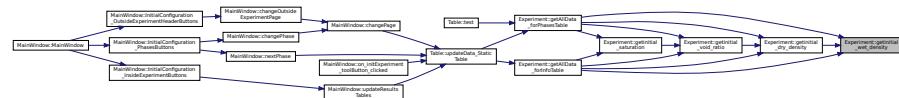
6.5.2.23 getinitial_wet_density()

```
float Experiment::getinitial_wet_density ( )
```

Here is the call graph for this function:



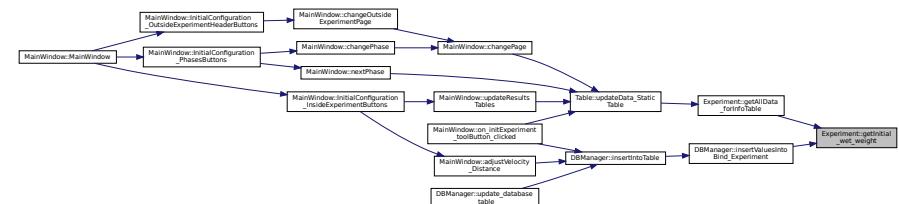
Here is the caller graph for this function:



6.5.2.24 getInitial_wet_weight()

```
float Experiment::getInitial_wet_weight ( ) const
```

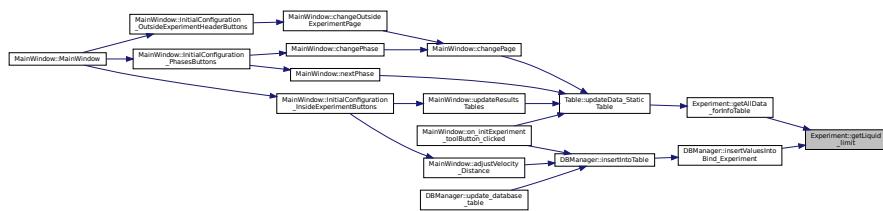
Here is the caller graph for this function:



6.5.2.25 getLiquid_limit()

```
float Experiment::getLiquid_limit ( ) const
```

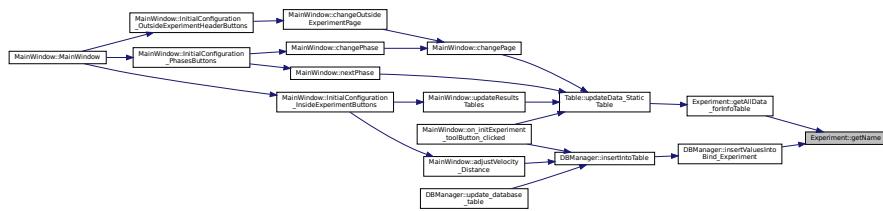
Here is the caller graph for this function:



6.5.2.26 getName()

```
const QString & Experiment::getName () const
```

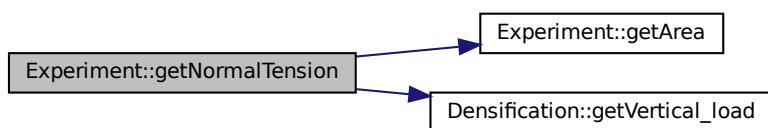
Here is the caller graph for this function:



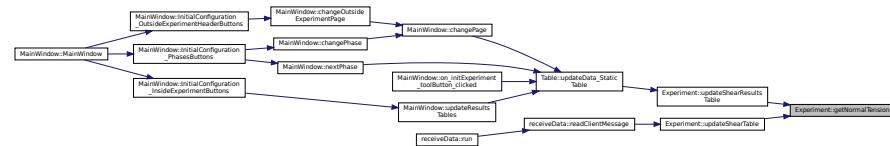
6.5.2.27 getNormalTension()

```
float Experiment::getNormalTension ()
```

Here is the call graph for this function:



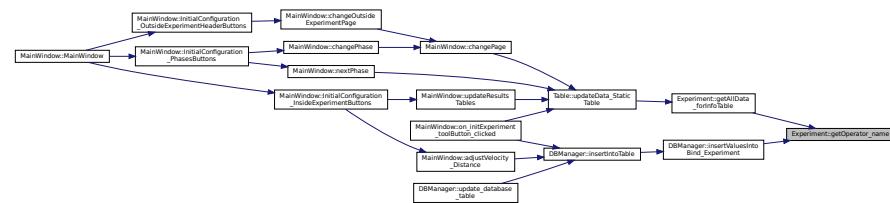
Here is the caller graph for this function:



6.5.2.28 getOperator_name()

```
const QString & Experiment::getOperator_name() const
```

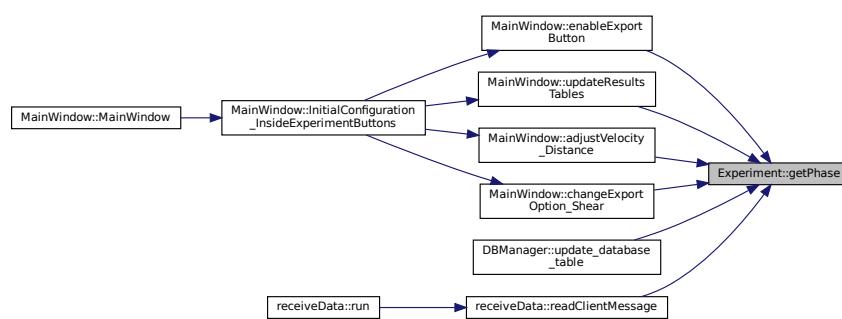
Here is the caller graph for this function:



6.5.2.29 getPhase()

```
int Experiment::getPhase() const
```

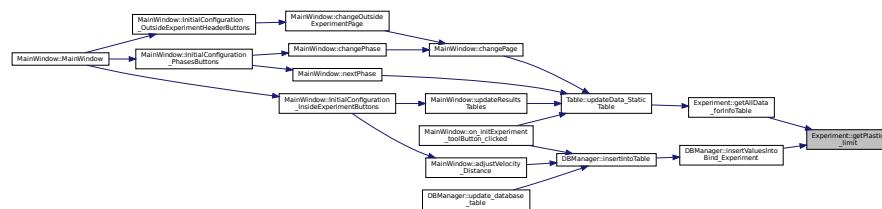
Here is the caller graph for this function:



6.5.2.30 getPlastic_limit()

```
float Experiment::getPlastic_limit () const
```

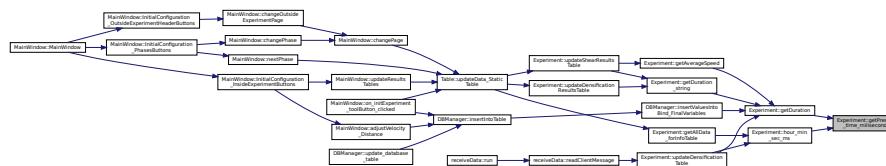
Here is the caller graph for this function:



6.5.2.31 getPresent_time_milliseconds()

```
uint64_t Experiment::getPresent_time_milliseconds () const
```

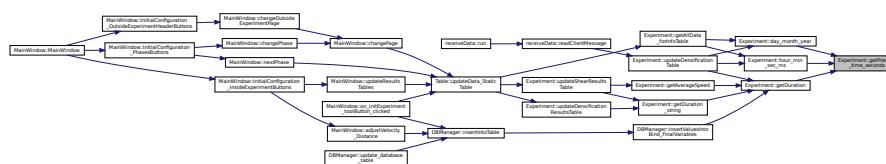
Here is the caller graph for this function:



6.5.2.32 getPresent_time_seconds()

```
uint64_t Experiment::getPresent_time_seconds () const
```

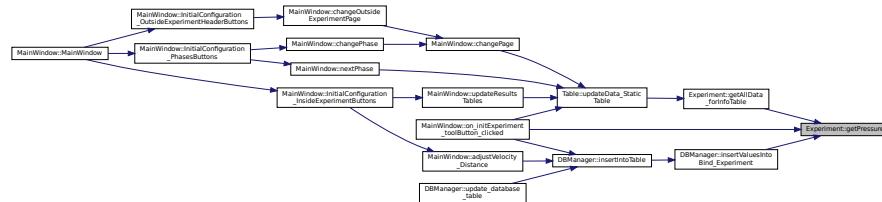
Here is the caller graph for this function:



6.5.2.33 getPressure()

```
float Experiment::getPressure ( ) const
```

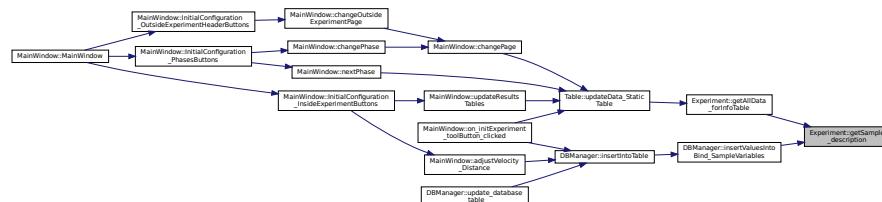
Here is the caller graph for this function:



6.5.2.34 getSample_description()

```
const QString & Experiment::getSample_description ( ) const
```

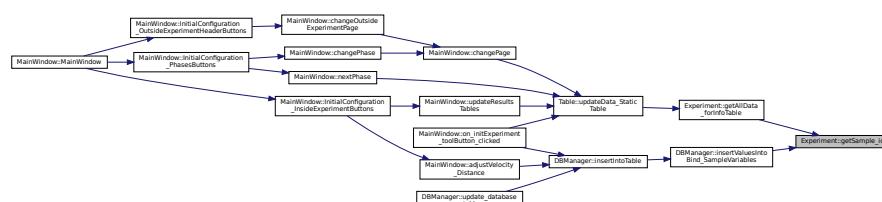
Here is the caller graph for this function:



6.5.2.35 getSample_id()

```
int Experiment::getSample_id ( ) const
```

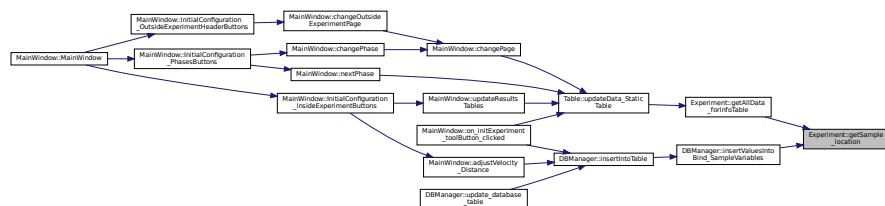
Here is the caller graph for this function:



6.5.2.36 getSample_location()

```
const QString & Experiment::getSample_location() const
```

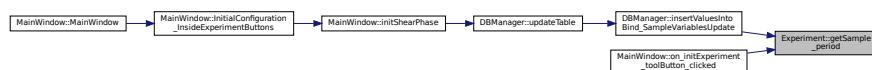
Here is the caller graph for this function:



6.5.2.37 getSample_period()

```
int Experiment::getSample_period() const
```

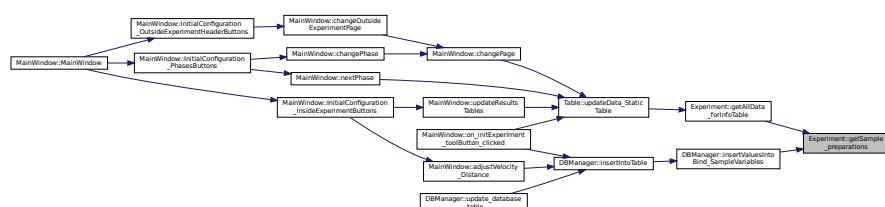
Here is the caller graph for this function:



6.5.2.38 getSample_preparations()

```
const QString & Experiment::getSample_preparations() const
```

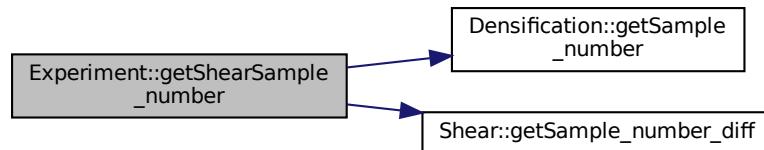
Here is the caller graph for this function:



6.5.2.39 getShearSample_number()

```
uint32_t Experiment::getShearSample_number ( ) const
```

Here is the call graph for this function:



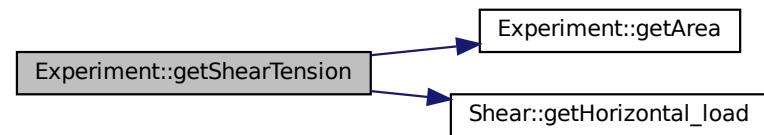
Here is the caller graph for this function:



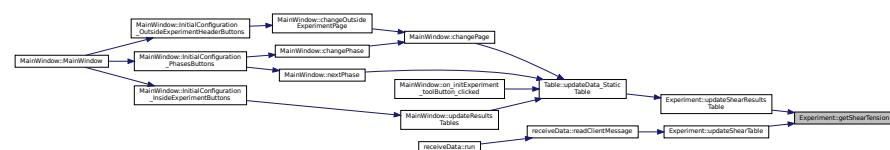
6.5.2.40 getShearTension()

```
float Experiment::getShearTension ( )
```

Here is the call graph for this function:



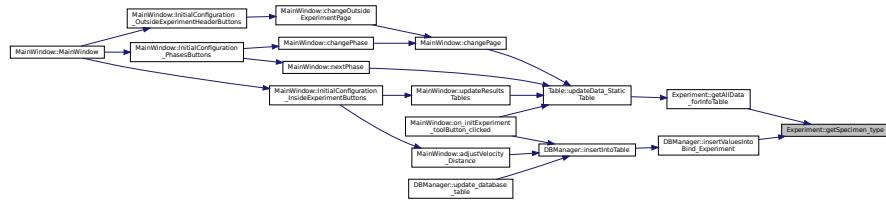
Here is the caller graph for this function:



6.5.2.41 getSpecimen_type()

```
const QString & Experiment::getSpecimen_type ( ) const
```

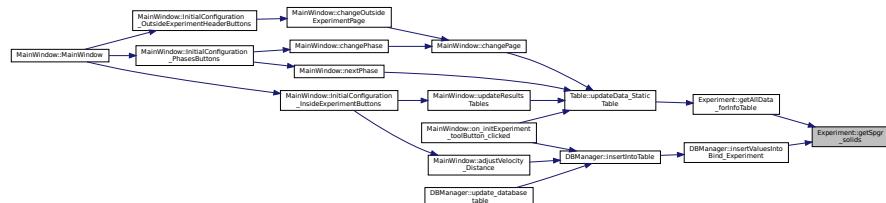
Here is the caller graph for this function:



6.5.2.42 getSpgr_solids()

```
float Experiment::getSpgr_solids ( ) const
```

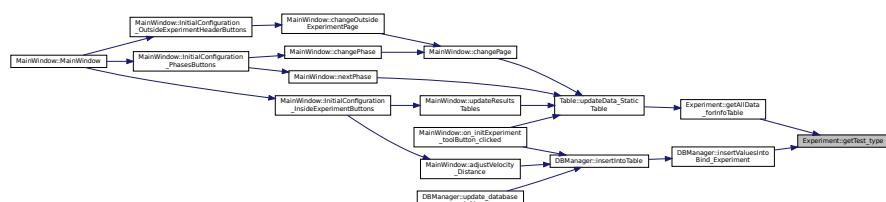
Here is the caller graph for this function:



6.5.2.43 getTest type()

```
const QString & Experiment::getTest_type () const
```

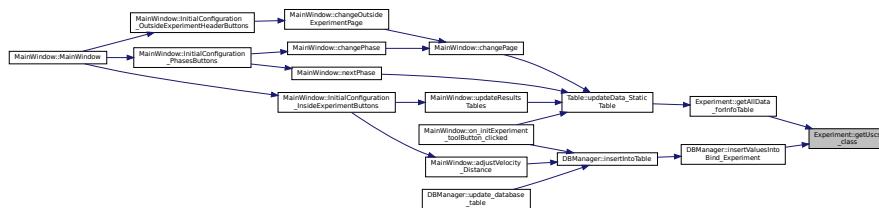
Here is the caller graph for this function:



6.5.2.44 getUscs_class()

```
const QString & Experiment::getUscs_class ( ) const
```

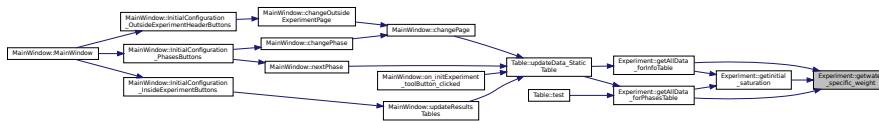
Here is the caller graph for this function:



6.5.2.45 getwater_specific_weight()

```
float Experiment::getwater_specific_weight ( )
```

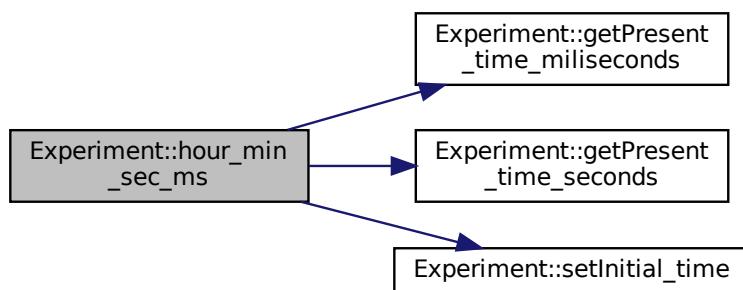
Here is the caller graph for this function:



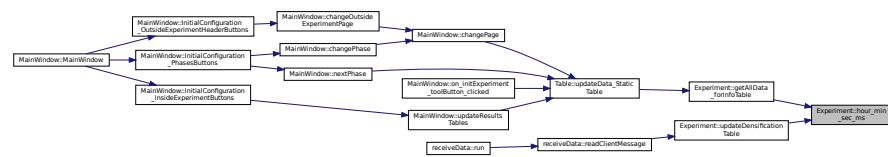
6.5.2.46 hour_min_sec_ms()

```
QString Experiment::hour_min_sec_ms ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



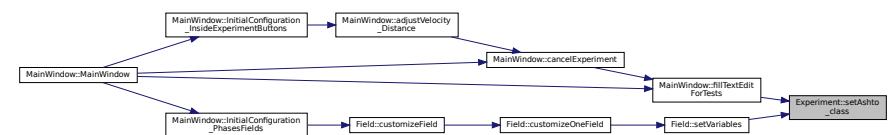
6.5.2.47 insertData_inDatabase()

```
void Experiment::insertData_inDatabase ( )
```

6.5.2.48 setAshto_class()

```
void Experiment::setAshto_class (
    const QString & newAshto_class )
```

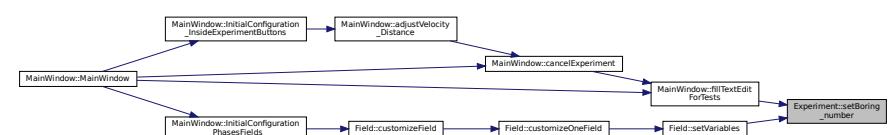
Here is the caller graph for this function:



6.5.2.49 setBoring_number()

```
void Experiment::setBoring_number (
    int newBoring_number )
```

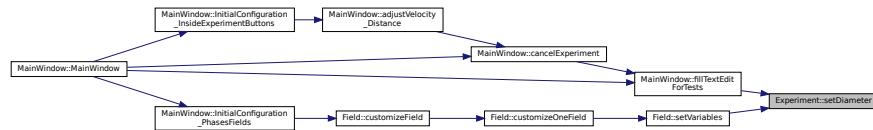
Here is the caller graph for this function:



6.5.2.50 setDiameter()

```
void Experiment::setDiameter (
    float newDiameter )
```

Here is the caller graph for this function:



6.5.2.51 setExperimentStarted()

```
void Experiment::setExperimentStarted (
    bool newExperimentStarted )
```

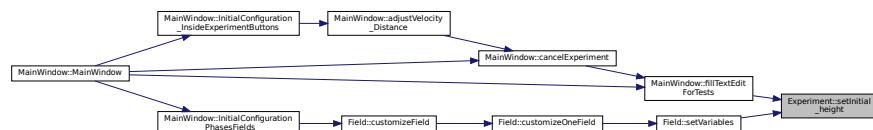
Here is the caller graph for this function:



6.5.2.52 setInitial_height()

```
void Experiment::setInitial_height (
    float newInitial_height )
```

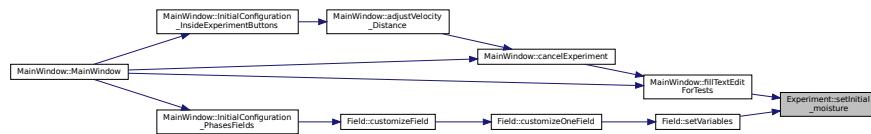
Here is the caller graph for this function:



6.5.2.53 setInitial_moisture()

```
void Experiment::setInitial_moisture (
    float newInitial_moisture )
```

Here is the caller graph for this function:



6.5.2.54 setInitial_position()

```
void Experiment::setInitial_position (
    float newInitial_position )
```

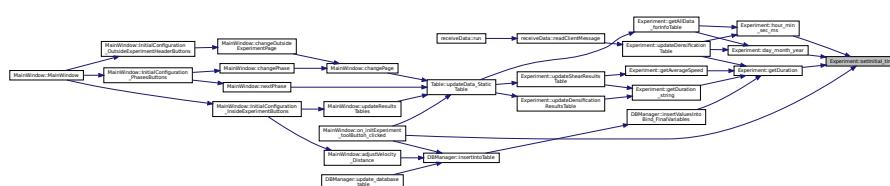
Here is the caller graph for this function:



6.5.2.55 setInitial_time()

```
void Experiment::setInitial_time (
    bool isInitial )
```

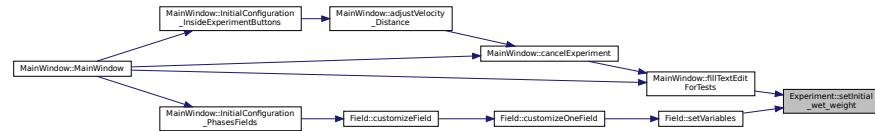
Here is the caller graph for this function:



6.5.2.56 setInitial_wet_weight()

```
void Experiment::setInitial_wet_weight (
    float newInitial_wet_weight )
```

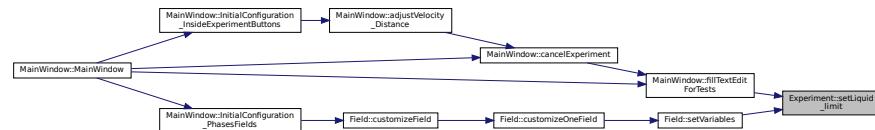
Here is the caller graph for this function:



6.5.2.57 setLiquid_limit()

```
void Experiment::setLiquid_limit (
    float newLiquid_limit )
```

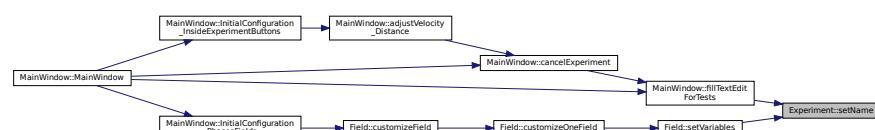
Here is the caller graph for this function:



6.5.2.58 setName()

```
void Experiment::setName (
    const QString & newName )
```

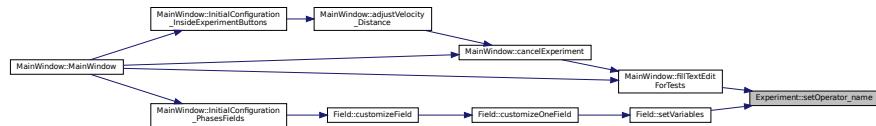
Here is the caller graph for this function:



6.5.2.59 setOperator_name()

```
void Experiment::setOperator_name (
    const QString & newOperator_name )
```

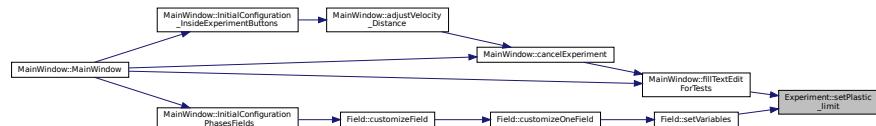
Here is the caller graph for this function:



6.5.2.60 setPlastic_limit()

```
void Experiment::setPlastic_limit (
    float newPlastic_limit )
```

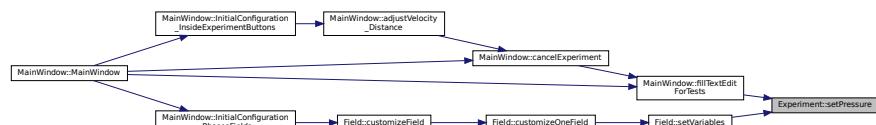
Here is the caller graph for this function:



6.5.2.61 setPressure()

```
void Experiment::setPressure (
    float newPressure )
```

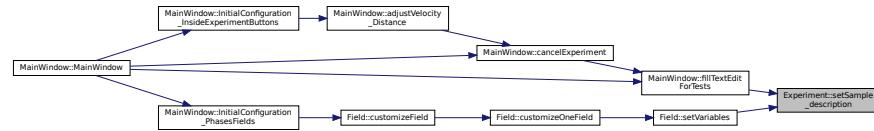
Here is the caller graph for this function:



6.5.2.62 setSample_description()

```
void Experiment::setSample_description (
    const QString & newSample_description )
```

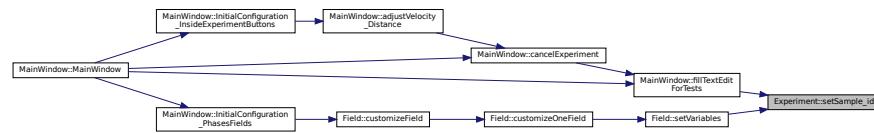
Here is the caller graph for this function:



6.5.2.63 setSample_id()

```
void Experiment::setSample_id (
    int newSample_id )
```

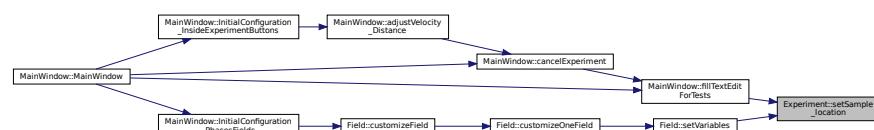
Here is the caller graph for this function:



6.5.2.64 setSample_location()

```
void Experiment::setSample_location (
    const QString & newSample_location )
```

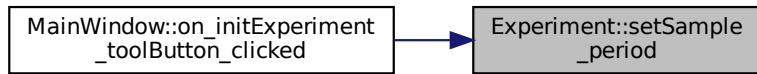
Here is the caller graph for this function:



6.5.2.65 setSample_period()

```
void Experiment::setSample_period (
    int newSample_period )
```

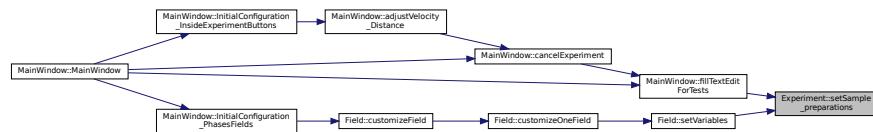
Here is the caller graph for this function:



6.5.2.66 setSample_preparations()

```
void Experiment::setSample_preparations (
    const QString & newSample_preparations )
```

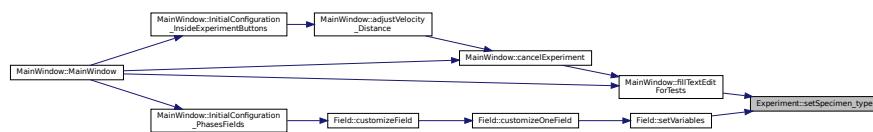
Here is the caller graph for this function:



6.5.2.67 setSpecimen_type()

```
void Experiment::setSpecimen_type (
    const QString & newSpecimen_type )
```

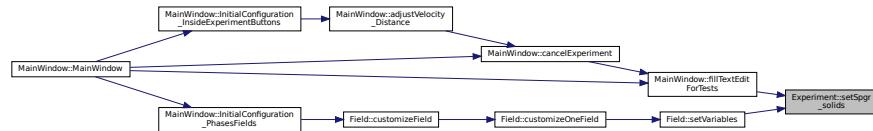
Here is the caller graph for this function:



6.5.2.68 setSpgr_solids()

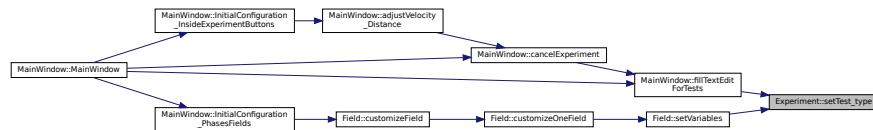
```
void Experiment::setSpgr_solids (
    float newSpgr_solids )
```

Here is the caller graph for this function:

**6.5.2.69 setTest_type()**

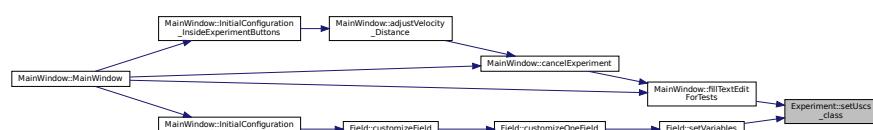
```
void Experiment::setTest_type (
    const QString & newTest_type )
```

Here is the caller graph for this function:

**6.5.2.70 setUscs_class()**

```
void Experiment::setUscs_class (
    const QString & newUscs_class )
```

Here is the caller graph for this function:



6.5.2.71 updateDensificationChart

```
void Experiment::updateDensificationChart (
    int time,
    float vertical_displacement ) [signal]
```

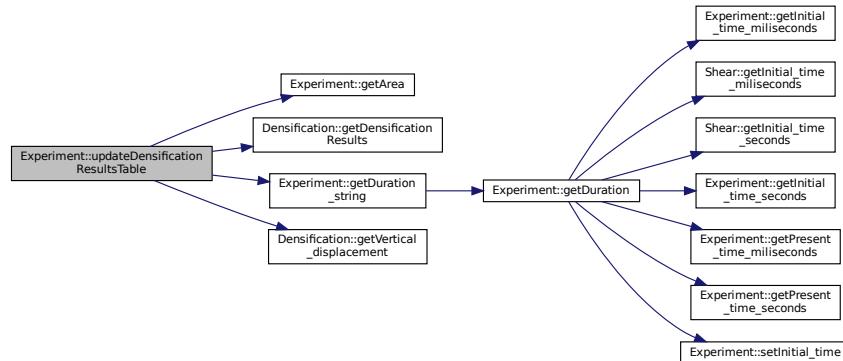
Here is the caller graph for this function:



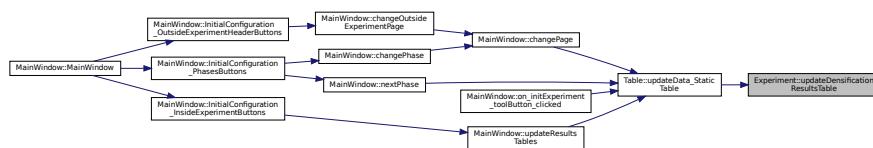
6.5.2.72 updateDensificationResultsTable()

```
QStringList Experiment::updateDensificationResultsTable ( )
```

Here is the call graph for this function:



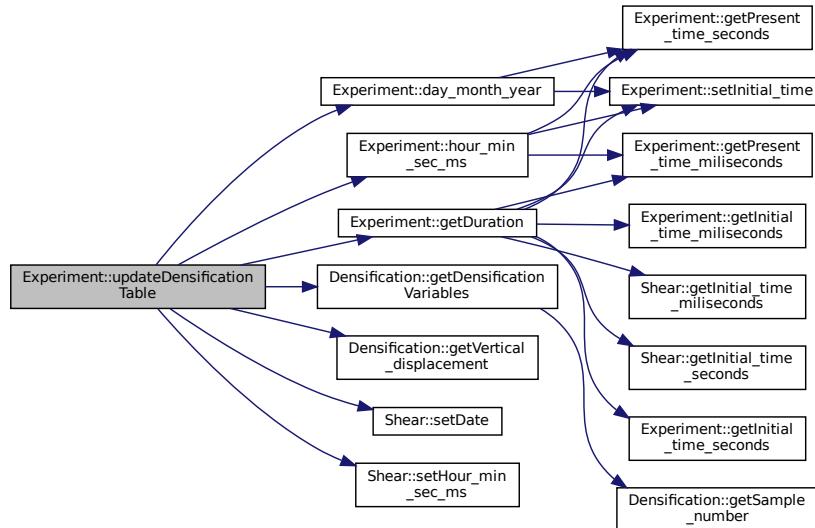
Here is the caller graph for this function:



6.5.2.73 updateDensificationTable()

```
QStringList Experiment::updateDensificationTable ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.5.2.74 updateShearChart

```
void Experiment::updateShearChart (
    float horizontal_displacement,
    float vertical_displacement ) [signal]
```

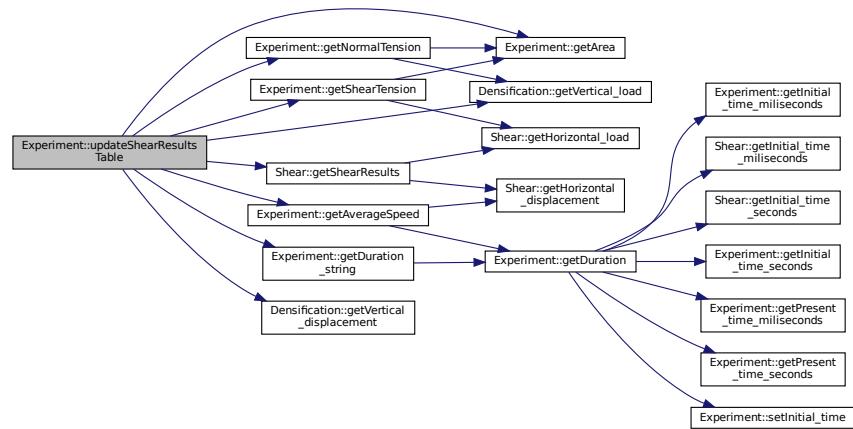
Here is the caller graph for this function:



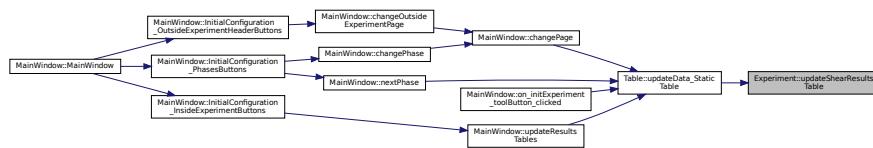
6.5.2.75 updateShearResultsTable()

```
QStringList Experiment::updateShearResultsTable ( )
```

Here is the call graph for this function:



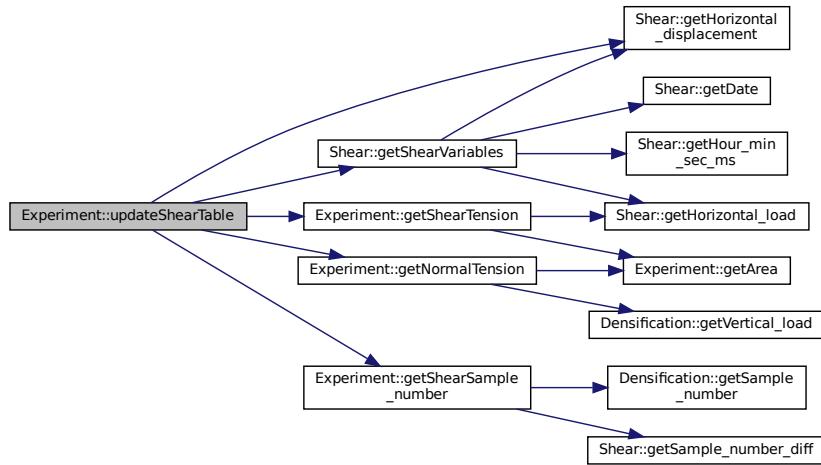
Here is the caller graph for this function:



6.5.2.76 updateShearTable()

```
QStringList Experiment::updateShearTable ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.5.3 Member Data Documentation

6.5.3.1 ashto_class

```
QString Experiment::ashto_class [private]
```

6.5.3.2 boring_number

```
int Experiment::boring_number [private]
```

6.5.3.3 densification_variables

```
Densification Experiment::densification_variables
```

6.5.3.4 diameter

```
float Experiment::diameter [private]
```

6.5.3.5 experimentStarted

```
bool Experiment::experimentStarted = false [private]
```

6.5.3.6 initial_height

```
float Experiment::initial_height [private]
```

6.5.3.7 initial_moisture

```
float Experiment::initial_moisture [private]
```

6.5.3.8 initial_position

```
float Experiment::initial_position [private]
```

6.5.3.9 initial_time_milliseconds

```
uint64_t Experiment::initial_time_milliseconds [private]
```

6.5.3.10 initial_time_seconds

```
uint64_t Experiment::initial_time_seconds [private]
```

6.5.3.11 initial_wet_weight

```
float Experiment::initial_wet_weight [private]
```

6.5.3.12 liquid_limit

```
float Experiment::liquid_limit [private]
```

6.5.3.13 name

```
QString Experiment::name [private]
```

6.5.3.14 operator_name

```
QString Experiment::operator_name [private]
```

6.5.3.15 phase

```
int Experiment::phase = densification_phase [private]
```

6.5.3.16 plastic_limit

```
float Experiment::plastic_limit [private]
```

6.5.3.17 present_time_milliseconds

```
uint64_t Experiment::present_time_milliseconds [private]
```

6.5.3.18 present_time_seconds

```
uint64_t Experiment::present_time_seconds [private]
```

6.5.3.19 pressure

```
float Experiment::pressure [private]
```

6.5.3.20 sample_description

```
QString Experiment::sample_description [private]
```

6.5.3.21 sample_id

```
int Experiment::sample_id [private]
```

6.5.3.22 sample_location

```
QString Experiment::sample_location [private]
```

6.5.3.23 sample_period

```
int Experiment::sample_period [private]
```

6.5.3.24 sample_preparations

```
QString Experiment::sample_preparations [private]
```

6.5.3.25 shear_variables

```
Shear Experiment::shear_variables
```

6.5.3.26 specimen_type

```
QString Experiment::specimen_type [private]
```

6.5.3.27 spgr_solids

```
float Experiment::spgr_solids [private]
```

6.5.3.28 test_type

```
QString Experiment::test_type [private]
```

6.5.3.29 uscs_class

```
QString Experiment::uscs_class [private]
```

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

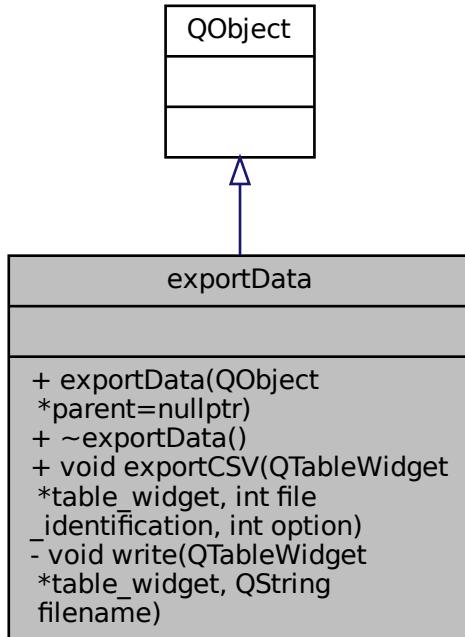
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/experiment.h
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/experiment.cpp

6.6 exportData Class Reference

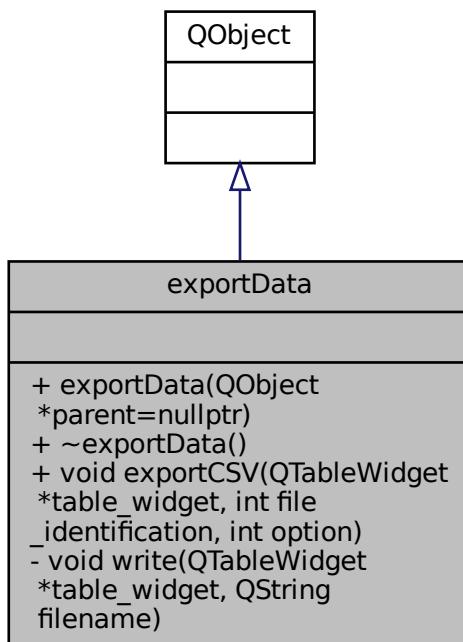
Classe que exporta os dados para csv.

```
#include <exportdata.h>
```

Inheritance diagram for exportData:



Collaboration diagram for exportData:



Public Slots

- void [exportCSV](#) (QTableWidget *table_widget, int file_identification, int option)
Essa função exporta o arquivo csv para o Github.

Public Member Functions

- [exportData](#) (QObject *parent=nullptr)
Constrói uma nova instância da classe `exportData`.
- [~exportData](#) ()
Destroi a instância de `exportData` e retorna para a branch main.

Private Member Functions

- void [write](#) (QTableWidget *table_widget, QString filename)
Escreve os dados de `table_widget` em um arquivo csv.

6.6.1 Detailed Description

Classe que exporta os dados para csv.

Esta classe é responsável por exportar os dados para o Github.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 `exportData()`

```
exportData::exportData (
    QObject * parent = nullptr ) [explicit]
```

Constrói uma nova instância da classe `exportData`.

Parameters

<code>parent</code>	Classe pai
---------------------	------------

6.6.2.2 `~exportData()`

```
exportData::~exportData ( )
```

Destrói a instância de `exportData` e retorna para a branch main.

6.6.3 Member Function Documentation

6.6.3.1 `exportCSV`

```
void exportData::exportCSV (
    QTableWidget * table_widget,
    int file_identification,
    int option ) [slot]
```

Essa função exporta o arquivo csv para o Github.

Esse slot é disparado quando o botão exportar recebe o sinal clicked.

Parameters

<code>table_widget</code>	Tabela com os dados
<code>file_identification</code>	Número do experimento. Utilizado para nomear o arquivo.
<code>option</code>	Nome do arquivo, baseado na tabela escolhida.

Here is the call graph for this function:



6.6.3.2 write()

```
void exportData::write (
    QTableWidget * table_widget,
    QString filename ) [private]
```

Escreve os dados de table_widget em um arquivo csv.

Parameters

<i>table_widget</i>	Tabela com os dados.
<i>filename</i>	Nome do arquivo.

Here is the caller graph for this function:



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

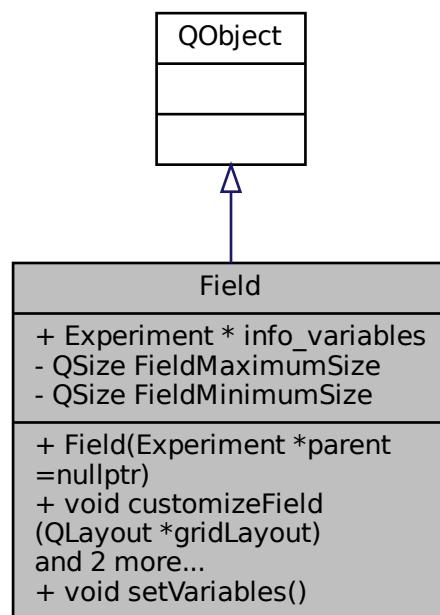
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[exportdata.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[exportdata.cpp](#)

6.7 Field Class Reference

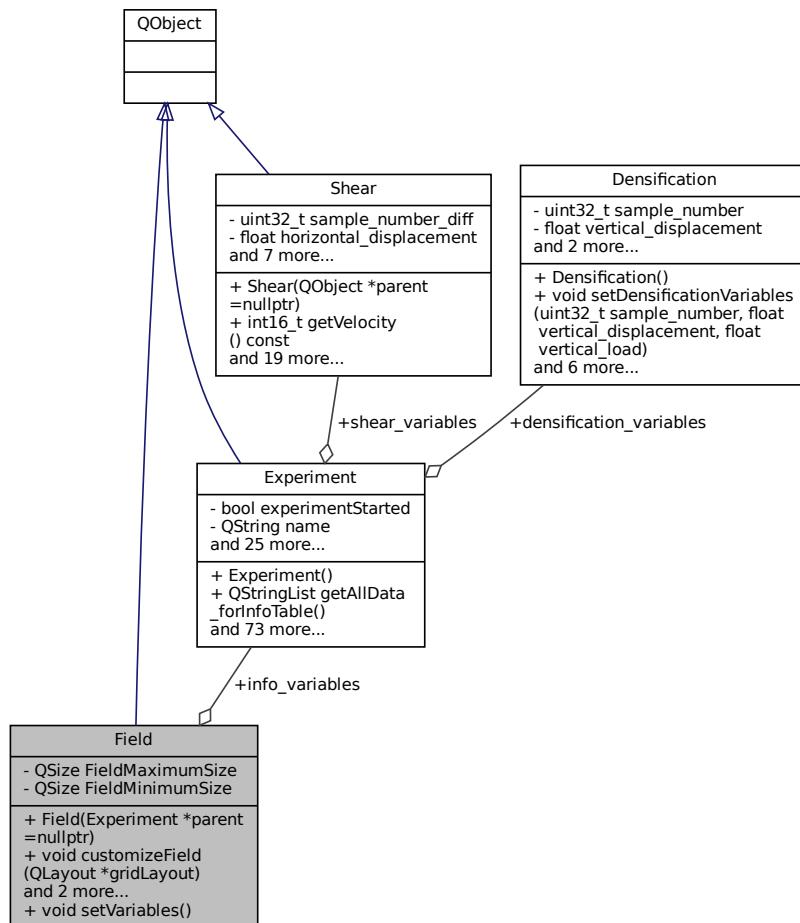
Classe dos campos de texto.

```
#include <field.h>
```

Inheritance diagram for Field:



Collaboration diagram for Field:



Public Slots

- `void setVariables ()`
Define os valores das variáveis do experimento com base nos campos de texto.

Public Member Functions

- `Field (Experiment *parent=nullptr)`
Constrói uma nova instância da classe `Field`.
- `void customizeField (QWidget *gridLayout)`
Customização dos campos de texto com a label superior de um layout escolhido.
- `void customizeOneField (QLabel *label, QLineEdit *lineEdit)`
Realiza a customização de um campo de texto e de uma label.
- `void clearFields (QWidget *gridLayout)`
Limpa os campos de texto de uma layout escolhido.

Public Attributes

- `Experiment * info_variables`

Private Attributes

- `QSize FieldMaximumSize [10]`
- `QSize FieldMinimumSize [10]`

6.7.1 Detailed Description

Classe dos campos de texto.

Utilizada para estilizar os campos de texto.

6.7.2 Constructor & Destructor Documentation

6.7.2.1 Field()

```
Field::Field (
    Experiment * parent = nullptr )
```

Constrói uma nova instância da classe `Field`.

Insere os valores mínimos e máximos do campo de texto.

Parameters

<code>parent</code>	instância da classe experimento criada na classe <code>MainWindow</code> .
---------------------	--

6.7.3 Member Function Documentation

6.7.3.1 clearFields()

```
void Field::clearFields (
    QLayout * gridLayout )
```

Limpa os campos de texto de uma layout escolhido.

Parameters

<i>gridLayout</i>	Layout escolhido.
-------------------	-------------------

Here is the caller graph for this function:



6.7.3.2 customizeField()

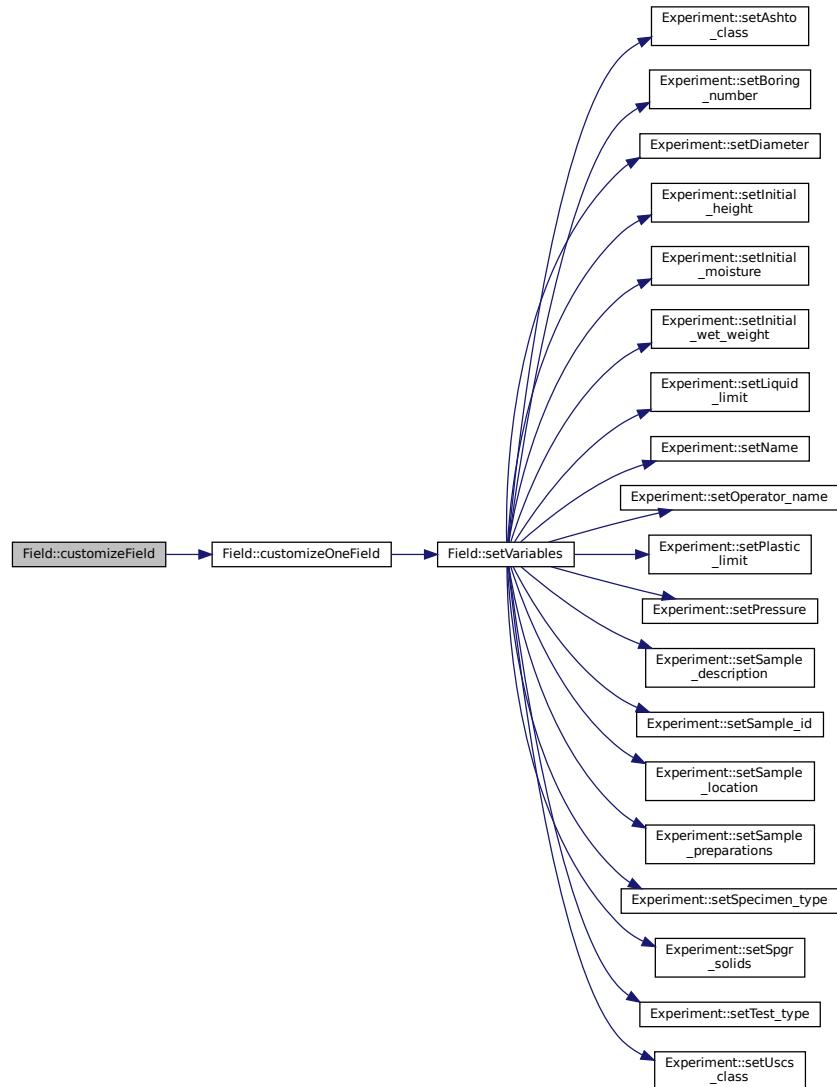
```
void Field::customizeField ( QLayout * gridLayout )
```

Customização dos campos de texto com a label superior de um layout escolhido.

Parameters

<i>gridLayout</i>	Layout escolhido.
-------------------	-------------------

Here is the call graph for this function:



Here is the caller graph for this function:



6.7.3.3 customizeOneField()

```

void Field::customizeOneField (
    QLabel * label,
    QLineEdit * lineEdit )
  
```

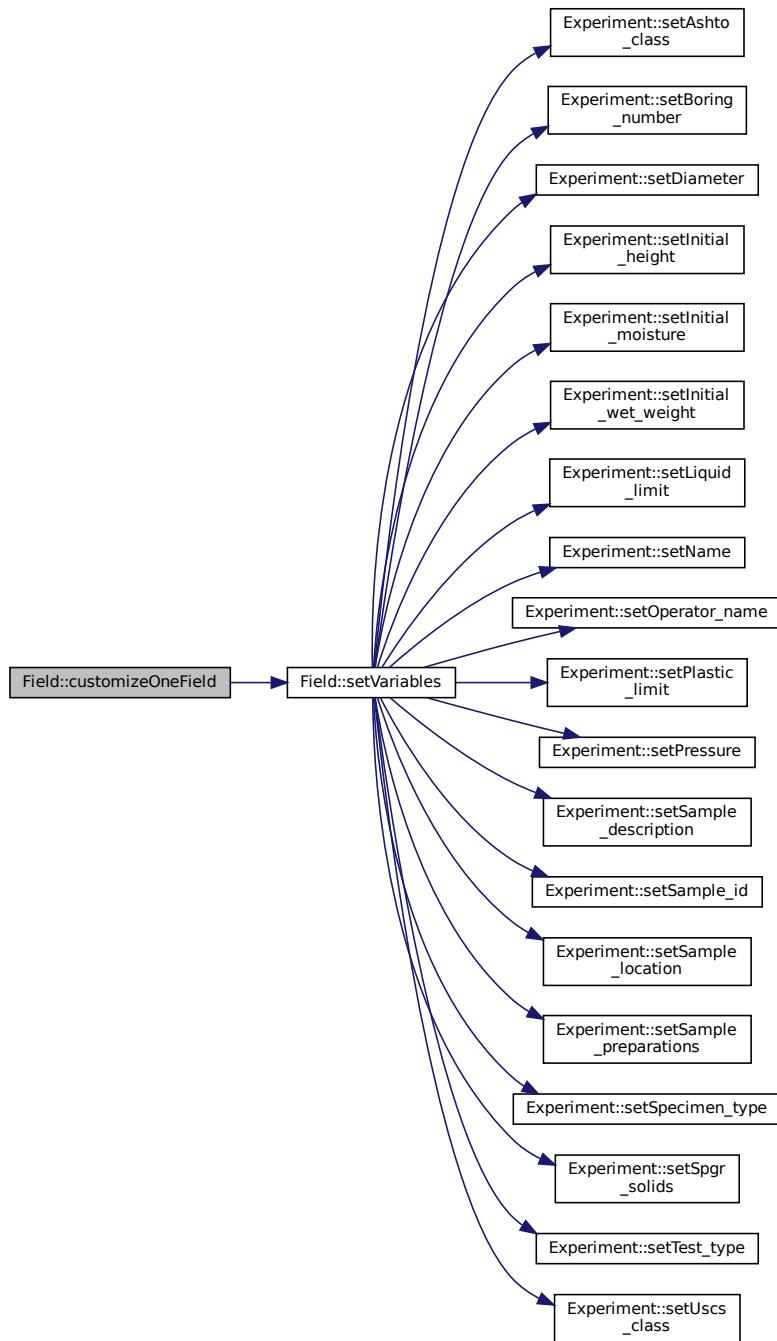
Realiza a customização de um campo de texto e de uma label.

Conecta os campos de texto ao SLOT editingFinished().

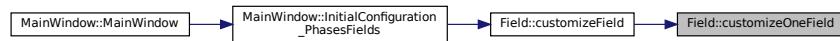
Parameters

<i>label</i>	Label escolhida.
<i>lineEdit</i>	Campo de texto escolhido.

Here is the call graph for this function:



Here is the caller graph for this function:



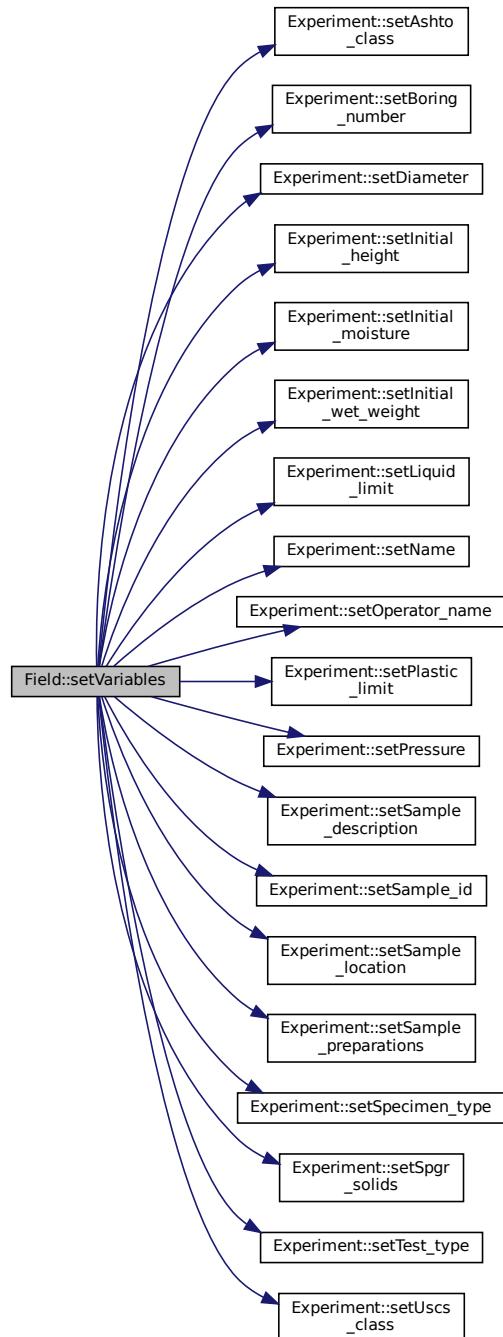
6.7.3.4 setVariables

```
void Field::setVariables ( ) [slot]
```

Define os valores das variáveis do experimento com base nos campos de texto.

Esse slot é disparado toda vez que o sinal editingFinished() é emitido pelos campos de texto. Here is the call graph

for this function:



Here is the caller graph for this function:



6.7.4 Member Data Documentation

6.7.4.1 FieldMaximumSize

```
QSize Field::FieldMaximumSize[10] [private]
```

Array com os valores máximos de tamanho que o campo de texto pode ter.

6.7.4.2 FieldMinimumSize

```
QSize Field::FieldMinimumSize[10] [private]
```

Array com os valores mínimos de tamanho que o campo de texto pode ter.

6.7.4.3 info_variables

```
Experiment* Field::info_variables
```

Variável da classe [Experiment](#) que receberá a instância da classe [MainWindow](#).

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

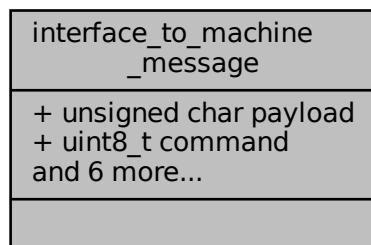
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[field.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[field.cpp](#)

6.8 interface_to_machine_message Union Reference

Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina.

```
#include <socket_local.h>
```

Collaboration diagram for interface_to_machine_message:



Public Attributes

- unsigned char `payload` [`interface_payload_size`]
- struct {
 - uint8_t `command`
Identificação do comando a ser enviado.
 - union {
 - uint16_t `pressure`
 - struct {
 int16_t `distance`
 int16_t `velocity`
 - struct {
 uint8_t `enabled`
 uint16_t `sample_period`

};

6.8.1 Detailed Description

Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina.

Utilizando a union, a transformação dos dados para string (unsigned char [] neste caso) é facilitada, pois os dados ocupam a mesma região de memória que a string. Assim para enviar os dados pelo socket basta enviar a variável `payload`.

6.8.2 Member Data Documentation

6.8.2.1

```
struct { ... } interface_to_machine_message::@1
```

6.8.2.2 command

```
uint8_t interface_to_machine_message::command
```

Identificação do comando a ser enviado.

0 para pressão, 1 para distância e velocidade e 2 para o período de amostragem e permitir o recebimento de dados.

6.8.2.3 distance

```
int16_t interface_to_machine_message::distance
```

Distância que a máquina percorrerá na fase de cisalhamento.

6.8.2.4 enabled

```
uint8_t interface_to_machine_message::enabled
```

Indica que a interface já pode receber os dados da máquina.

6.8.2.5 payload

```
unsigned char interface_to_machine_message::payload[interface_payload_size]
```

String com os dados a serem enviados.

6.8.2.6 pressure

```
uint16_t interface_to_machine_message::pressure
```

Pressão da máquina desejada pelo usuário

6.8.2.7 sample_period

```
uint16_t interface_to_machine_message::sample_period
```

Período de amostragem do recebimento dos dados vindos da máquina.

6.8.2.8 velocity

```
int16_t interface_to_machine_message::velocity
```

Velocidade que a máquina terá na fase de cisalhamento.

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

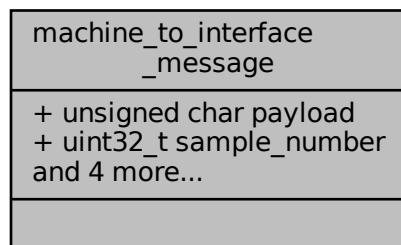
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/socket_local.h

6.9 machine_to_interface_message Union Reference

Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface.

```
#include <socket_local.h>
```

Collaboration diagram for machine_to_interface_message:



Public Attributes

- unsigned char `payload` [`machine_payload_size`]
- struct {
 - uint32_t `sample_number`
 - float `displacement` [2]
 - float `load` [2]
 - uint8_t `state`
};

6.9.1 Detailed Description

Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface.

Utilizando a union, a transformação dos dados para string (unsigned char [] neste caso) é facilitada, pois os dados ocupam a mesma região de memória que a string. Assim para enviar os dados pelo socket basta enviar a variável payload.

6.9.2 Member Data Documentation

6.9.2.1

```
struct { ... } machine_to_interface_message::@9
```

6.9.2.2 displacement

```
float machine_to_interface_message::displacement [2]
```

Valor do deslocamento, [0] para os deslocamentos verticais de adensamento e [1] para os horizontais de cisalhamento.

6.9.2.3 load

```
float machine_to_interface_message::load[2]
```

Valor da carga, [0] para as cargas verticais de adensamento e [1] para as horizontais de cisalhamento.

6.9.2.4 payload

```
unsigned char machine_to_interface_message::payload[machine_payload_size]
```

String com os dados a serem enviados.

6.9.2.5 sample_number

```
uint32_t machine_to_interface_message::sample_number
```

6.9.2.6 state

```
uint8_t machine_to_interface_message::state
```

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

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[socket_local.h](#)

6.10 machineClient Class Reference

Classe que simula o envio de dados da máquina real.

```
#include <machineclient.h>
```

Collaboration diagram for machineClient:

machineClient
- char socket_name - int socket_id - uint8_t errorOccurred
+ machineClient() + ~machineClient() + void sendMessages(int32_t i)

Public Member Functions

- [machineClient \(\)](#)
Construtor da classe [machineClient](#).
- [~machineClient \(\)](#)
Destruitor da classe [machineClient](#).
- [void sendMessages \(int32_t i\)](#)
Envia mensagens para o servidor da interface caso a conexão no construtor tenha sucesso.

Private Attributes

- char `socket_name` [200]
Nome do socket local.
- int `socket_id`
Identificador do socket local criado na função socket().
- uint8_t `errorOccurred`
Identifica se ocorreu um erro na conexão com o servidor da interface.

6.10.1 Detailed Description

Classe que simula o envio de dados da máquina real.

Esta classe tem como objetivo auxiliar o teste do socket, simulando o envio de dados da máquina real.

6.10.2 Constructor & Destructor Documentation

6.10.2.1 `machineClient()`

```
machineClient::machineClient ( )
```

Construtor da classe `machineClient`.

Este construtor cria uma conexão socket com o servidor da interface, o qual é aberto pela classe `receiveData`. Lembrando que a classe `ThreadController` que cria objetos da classe `receiveData`.

6.10.2.2 `~machineClient()`

```
machineClient::~machineClient ( )
```

Destrutor da classe `machineClient`.

Este destrutor encerra a conexão com o servidor da interface, fechando o socket. Isto é feito na função `close(this->socket_id);`

6.10.3 Member Function Documentation

6.10.3.1 `sendMessages()`

```
void machineClient::sendMessages (
    int32_t i )
```

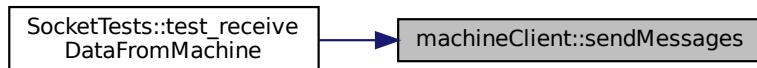
Envia mensagens para o servidor da interface caso a conexão no construtor tenha sucesso.

Esta função envia mensagens para o servidor da interface, simulando o comportamento da máquina.

Parameters

<i>i</i>	Valor utilizado para alterar o conteúdo da mensagem, e permitir assim, uma maior variedade de testes.
----------	---

Here is the caller graph for this function:



6.10.4 Member Data Documentation

6.10.4.1 errorOccurred

```
uint8_t machineClient::errorOccurred [private]
```

Identifica se ocorreu um erro na conexão com o servidor da interface.

O valor 1 significa que um erro ocorreu e que deve ocorrer uma nova tentativa de reconexão. O valor 0 significa que não houve erro.

6.10.4.2 socket_id

```
int machineClient::socket_id [private]
```

Identificador do socket local criado na função socket().

Um exemplo: this->socket_id = socket(PF_LOCAL, SOCK_STREAM, 0).

6.10.4.3 socket_name

```
char machineClient::socket_name[200] [private]
```

Nome do socket local.

Deve possuir o mesmo nome do socket aberto pelo servidor da interface na classe [receiveData](#).

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

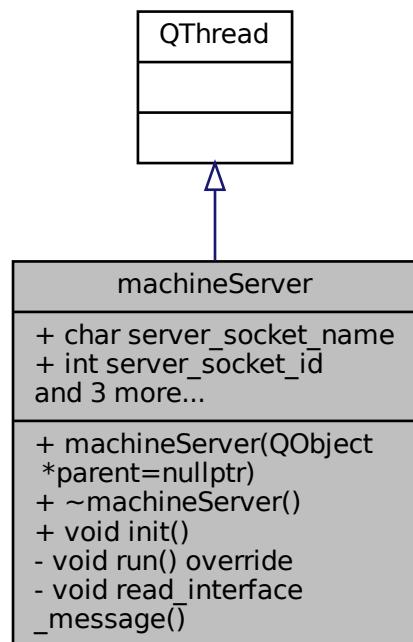
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/inc/[machineclient.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/[machineclient.cpp](#)

6.11 machineServer Class Reference

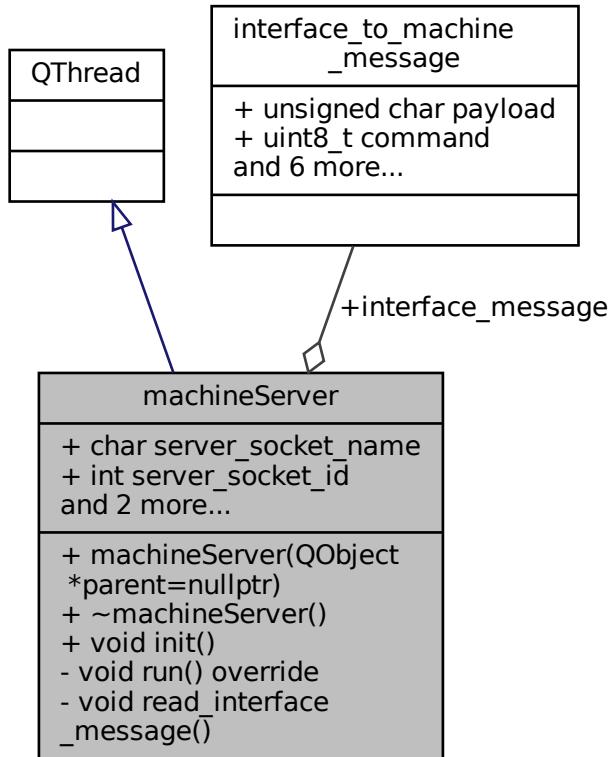
Classe que simula o servidor da máquina real.

```
#include <machineserver.h>
```

Inheritance diagram for machineServer:



Collaboration diagram for machineServer:



Public Member Functions

- `machineServer (QObject *parent=nullptr)`
Constrói uma instância da classe `machineServer`.
- `~machineServer ()`
Destruitor da classe `machineServer`.
- `void init ()`
Função utilizada para realizar qualquer operação antes do início da thread.

Public Attributes

- `char server_socket_name [100]`
- `int server_socket_id`
- `int client_socket_id`
- `interface_to_machine_message interface_message`
- `uint8_t errorOccurred`

Private Member Functions

- void [run \(\)](#) override
É a função que roda durante toda a vida da thread.
- void [read_interface_message \(\)](#)
Lê as mensagens enviadas pelo cliente da interface.

6.11.1 Detailed Description

Classe que simula o servidor da máquina real.

Esta classe é uma thread e tem como objetivo auxiliar o teste do socket, simulando o recebimento de dados pela máquina real.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 [machineServer\(\)](#)

```
machineServer::machineServer (   
    QObject * parent = nullptr )
```

Constrói uma instância da classe [machineServer](#).

Neste construtor é criado o socket servidor, realizado seu bind e determinado a quantidade de clientes que irá receber.

Parameters

<i>parent</i>	<input type="text"/>
---------------	----------------------

6.11.2.2 [~machineServer\(\)](#)

```
machineServer::~machineServer ( )
```

Destrutor da classe [machineServer](#).

Este destrutor tem como objetivo destruir os sockets do servidor e do cliente utilizados para realizar o envio de dados da interface para a máquina.

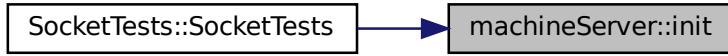
6.11.3 Member Function Documentation

6.11.3.1 init()

```
void machineServer::init ( )
```

Função utilizada para realizar qualquer operação antes do início da thread.

Essa função chama a função que inicia a thread, [init\(\)](#) chama start(). Here is the caller graph for this function:

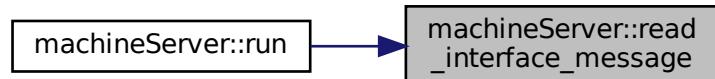


6.11.3.2 read_interface_message()

```
void machineServer::read_interface_message ( ) [private]
```

Lê as mensagens enviadas pelo cliente da interface.

A leitura é igual a leitura de um arquivo. Here is the caller graph for this function:



6.11.3.3 run()

```
void machineServer::run ( ) [override], [private]
```

É a função que roda durante toda a vida da thread.

Essa função fica esperando novos clientes da interface se conectarem e depois chama a função [read_interface_message\(\)](#) para ler as mensagens enviadas pela interface. Here is the call graph for this function:



6.11.4 Member Data Documentation

6.11.4.1 client_socket_id

```
int machineServer::client_socket_id
```

Identificador do socket do cliente aceitado pelo servidor.

6.11.4.2 errorOccurred

```
uint8_t machineServer::errorOccurred
```

Sinaliza se algum erro ocorreu no construtor da classe.

6.11.4.3 interface_message

```
interface_to_machine_message machineServer::interface_message
```

Union que contém a estrutura dos dados enviados pela interface.

6.11.4.4 server_socket_id

```
int machineServer::server_socket_id
```

Identificador do socket do servidor da máquina.

6.11.4.5 server_socket_name

```
char machineServer::server_socket_name[100]
```

Nome do socket servidor da máquina. O nome deve ser igual ao do socket cliente da Interface.

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

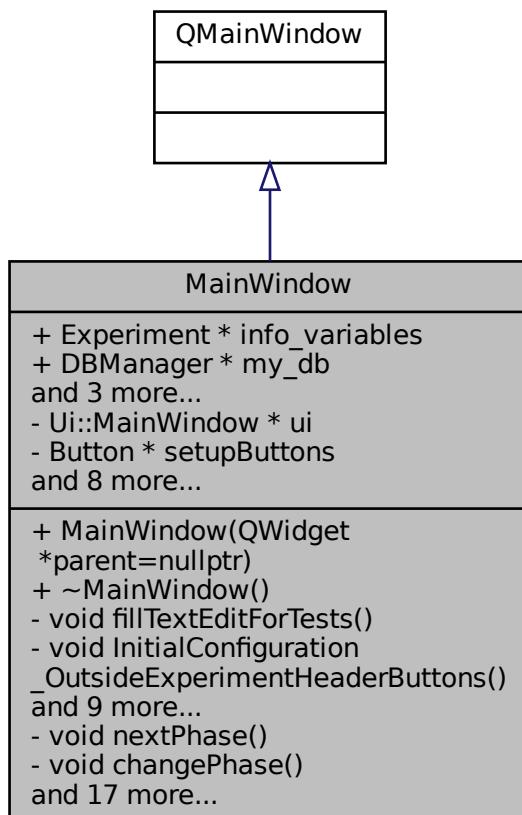
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/inc/machineserver.h
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/machineserver.cpp

6.12 MainWindow Class Reference

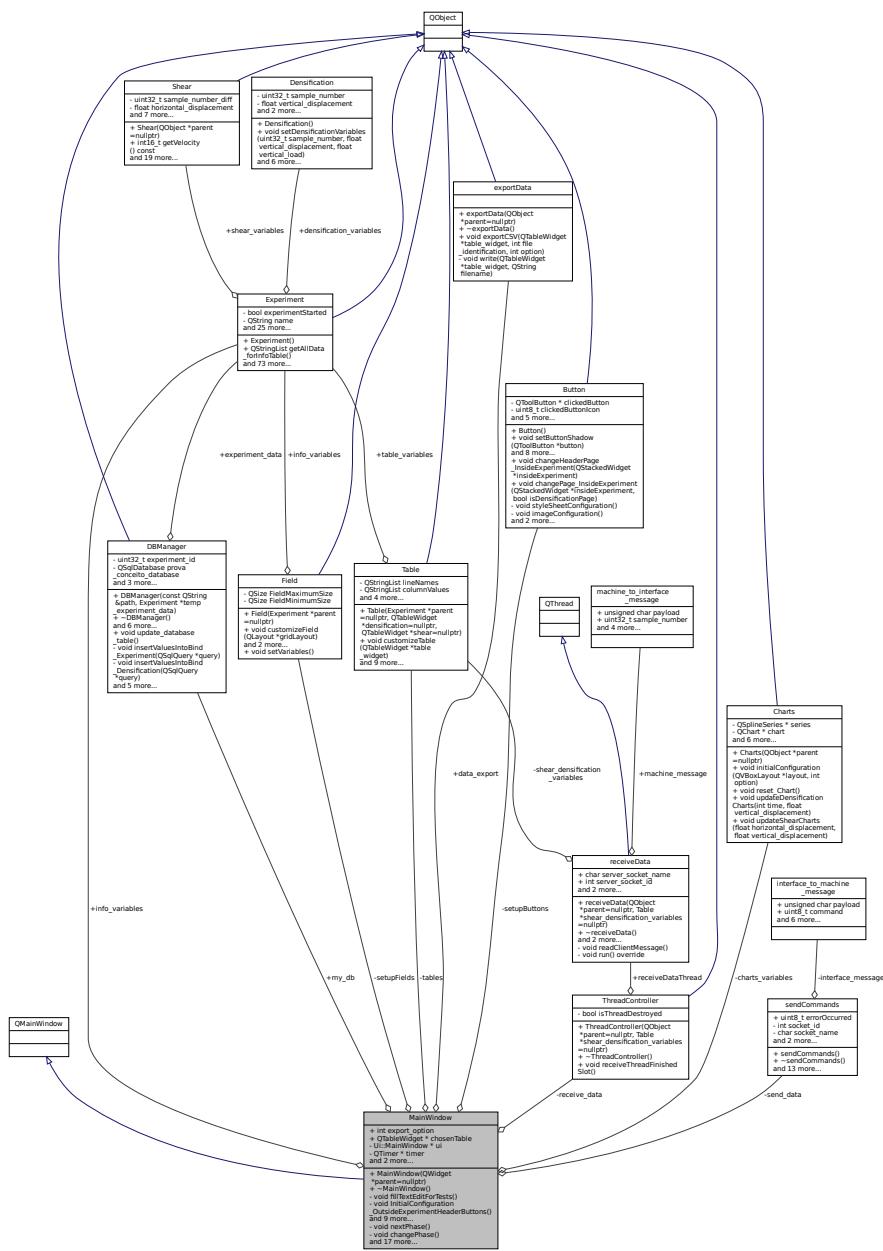
Classe que instancia a janela principal da interface.

```
#include <mainwindow.h>
```

Inheritance diagram for MainWindow:



Collaboration diagram for MainWindow:



Public Member Functions

- **MainWindow** (QWidget *parent=nullptr)
Constrói uma nova janela principal.
 - **~MainWindow ()**
Destroi a janela principal.

Public Attributes

- Experiment * info variables

- `DBManager * my_db`
- `exportData * data_export`
- `int export_option`
- `QTableWidget * chosenTable`

Private Slots

- `void nextPhase ()`
- `void changePhase ()`
- `void changeOutsideExperimentPage ()`
- `void changeInitialPositionValue ()`
- `void onPositionButton_pressed ()`
- `void onPositionButton_released ()`
- `void cancelExperiment ()`
- `void on_initExperiment_toolButton_clicked ()`
- `void on_releasePressure_toolButton_clicked ()`
- `void updateResultsTables ()`
- `void adjustVelocity_Distance ()`
- `void initShearPhase ()`
- `void on_goBack_toolButton_clicked ()`
- `void changeVelocity ()`
- `void changeDistance ()`
- `void enableShearInitButton (int index)`
- `void enableExportButton (int index)`
- `void changeExportOption_Densification (int index)`
- `void changeExportOption_Shear (int index)`

Private Member Functions

- `void fillTextEditForTests ()`
- `void InitialConfiguration_OutsideExperimentHeaderButtons ()`
- `void InitialConfiguration_InsideExperimentButtons ()`
- `void InitialConfiguration_PhasesButtons ()`
- `void InitialConfiguration_PhasesFields ()`
- `void InitialConfiguration_Tables ()`
- `void CreateDataTables ()`
- `void connectButtonToSlots_WithArguments (QHBoxLayout *boxlayout, QStackedWidget *stack_widget, int option)`
- `void connectButtonsToSlots_Layout (QHBoxLayout *list, const char *signal, const char *slot)`
- `void connectButtonsToSlots_Widget (QObject *list, const char *signal, const char *slot)`
- `void changePage (QToolButton *buttonSender, QString buttons_name[6], uint8_t array_size, QStackedWidget *page_stack, uint8_t icon[3], uint8_t style)`

Private Attributes

- `Ui::MainWindow * ui`
- `Button * setupButtons`
- `Field * setupFields`
- `Table * tables`
- `ThreadController * receive_data`
- `sendCommands * send_data`
- `QTimer * timer`
- `int previousIndex`
- `bool experiment_canceled = false`
- `Charts * charts_variables`

6.12.1 Detailed Description

Classe que instancia a janela principal da interface.

É a janela principal da aplicação.

6.12.2 Constructor & Destructor Documentation

6.12.2.1 MainWindow()

```
MainWindow::MainWindow (   
    QWidget * parent = nullptr )
```

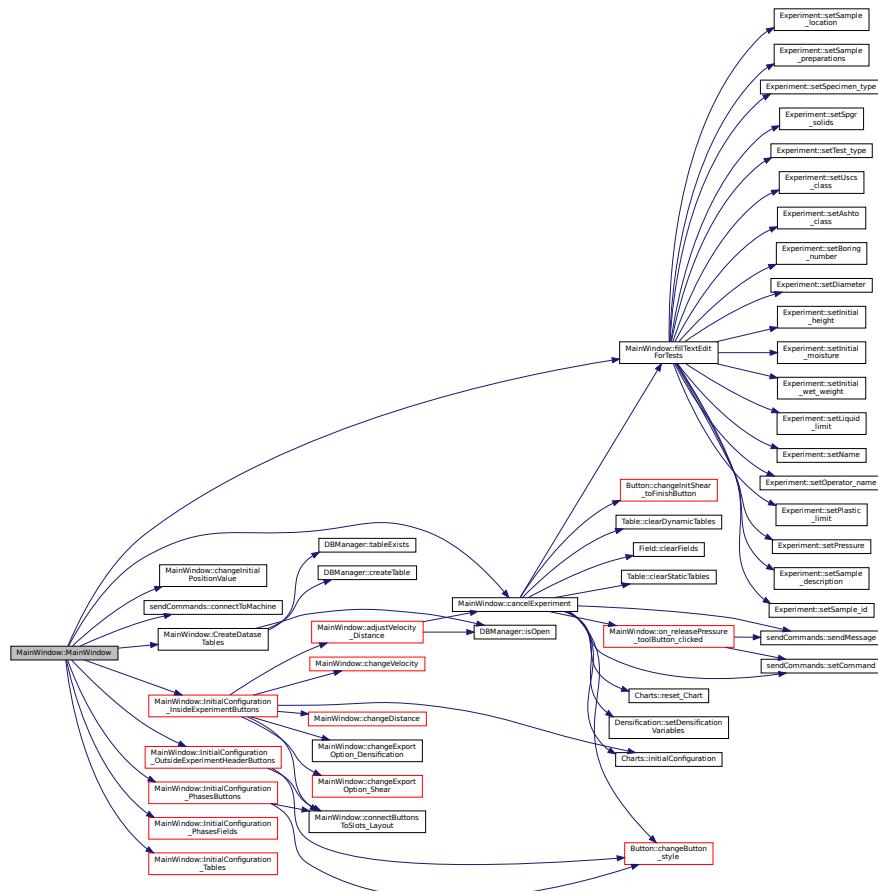
Constrói uma nova janela principal.

É o construtor da janela principal da interface.

Parameters

<i>parent</i>	
---------------	--

Here is the call graph for this function:



6.12.2.2 ~MainWindow()

```
MainWindow::~MainWindow ( )
```

Destroi a janela principal.

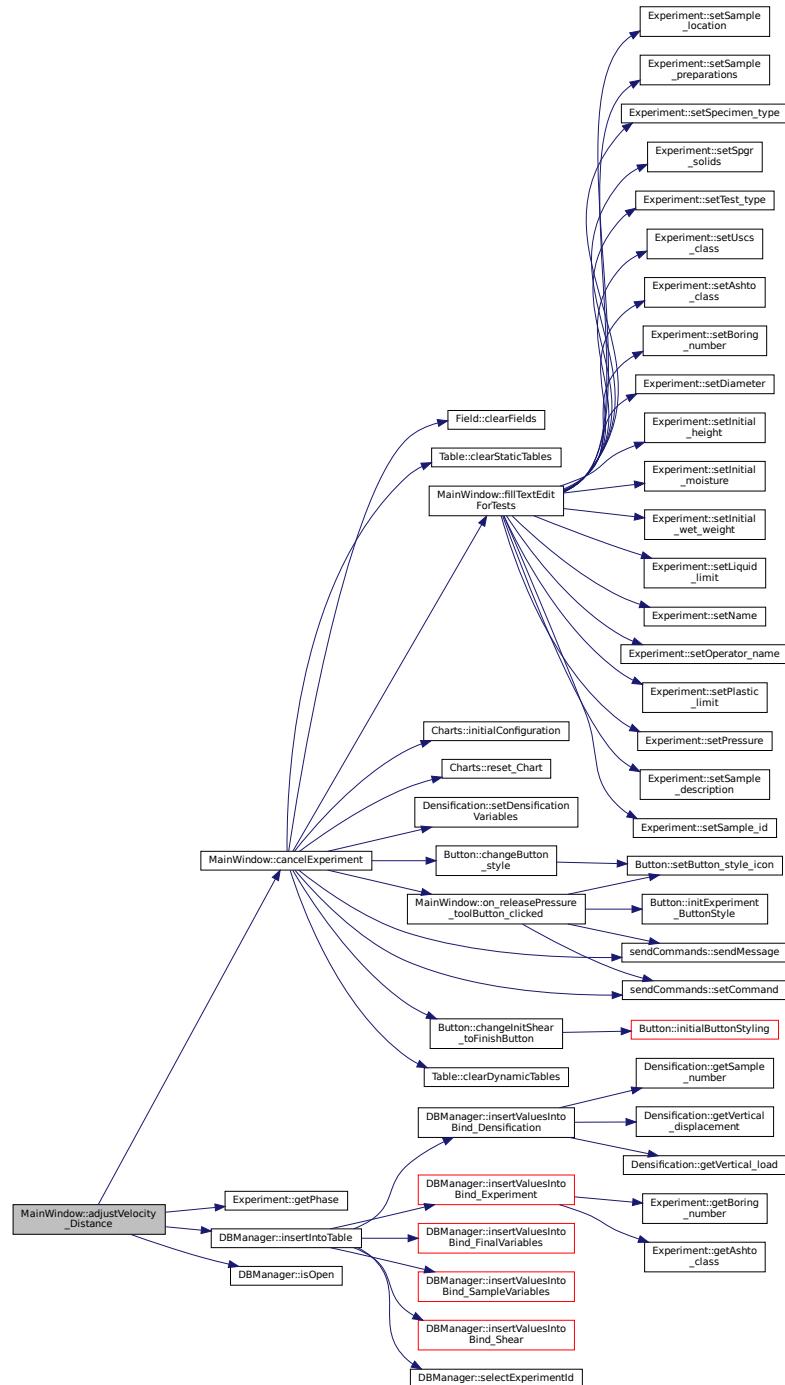
Destroi a instânciá ui;

6.12.3 Member Function Documentation

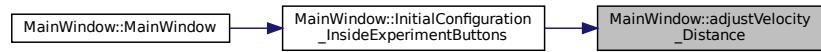
6.12.3.1 `adjustVelocity_Distance`

```
void MainWindow::adjustVelocity_Distance ( ) [private], [slot]
```

Here is the call graph for this function:



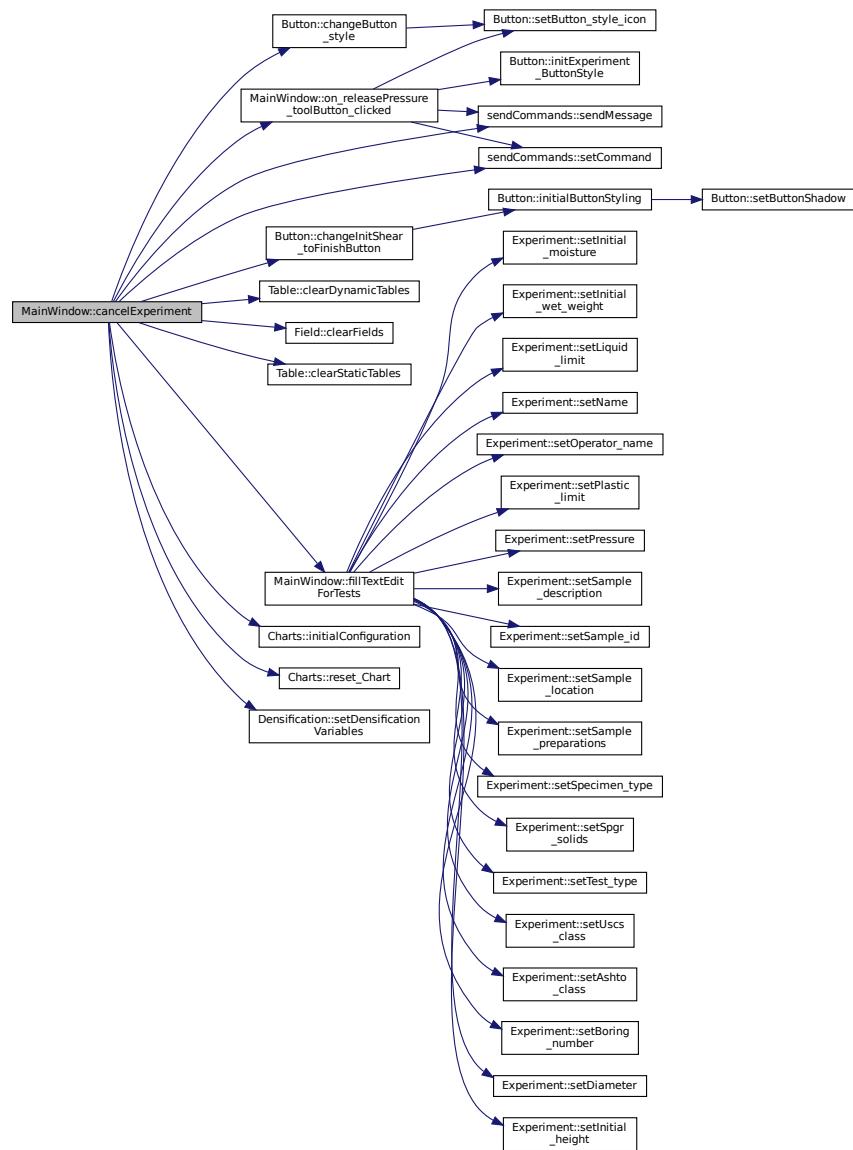
Here is the caller graph for this function:



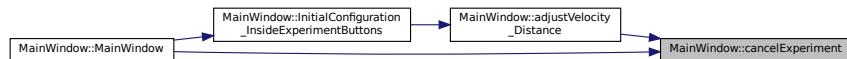
6.12.3.2 cancelExperiment

```
void MainWindow::cancelExperiment () [private], [slot]
```

Here is the call graph for this function:



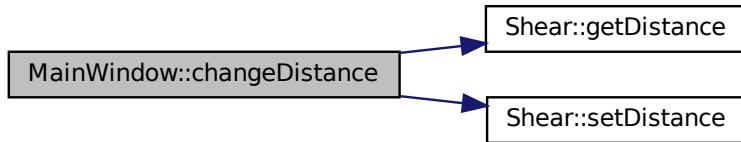
Here is the caller graph for this function:



6.12.3.3 changeDistance

```
void MainWindow::changeDistance ( ) [private], [slot]
```

Here is the call graph for this function:



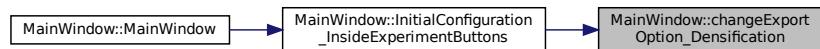
Here is the caller graph for this function:



6.12.3.4 changeExportOption_Densification

```
void MainWindow::changeExportOption_Densification (
    int index ) [private], [slot]
```

Here is the caller graph for this function:



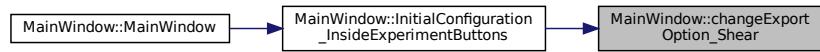
6.12.3.5 changeExportOption_Shear

```
void MainWindow::changeExportOption_Shear ( int index ) [private], [slot]
```

Here is the call graph for this function:



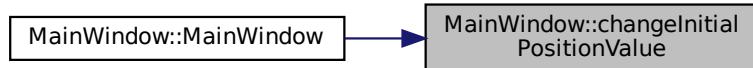
Here is the caller graph for this function:



6.12.3.6 changeInitialPositionValue

```
void MainWindow::changeInitialPositionValue ( ) [private], [slot]
```

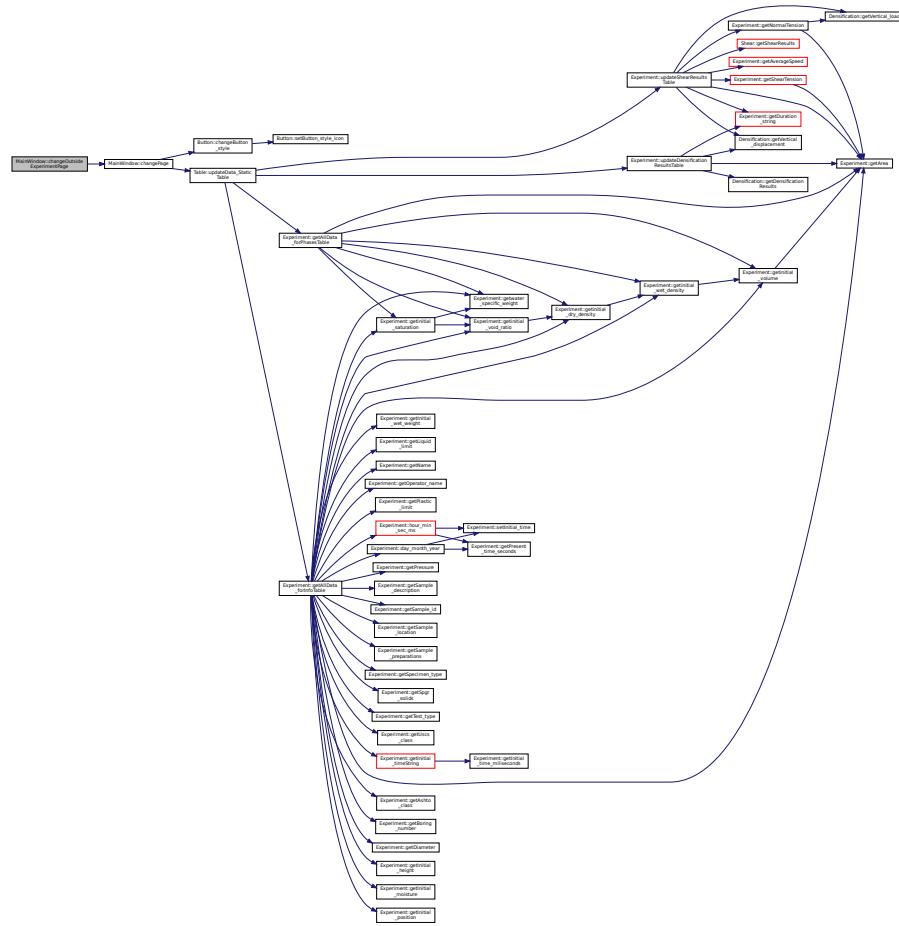
Here is the caller graph for this function:



6.12.3.7 changeOutsideExperimentPage

```
void MainWindow::changeOutsideExperimentPage () [private], [slot]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.12.3.8 changePage()

```
void MainWindow::changePage (
```

`QToolButton * buttonSender,`

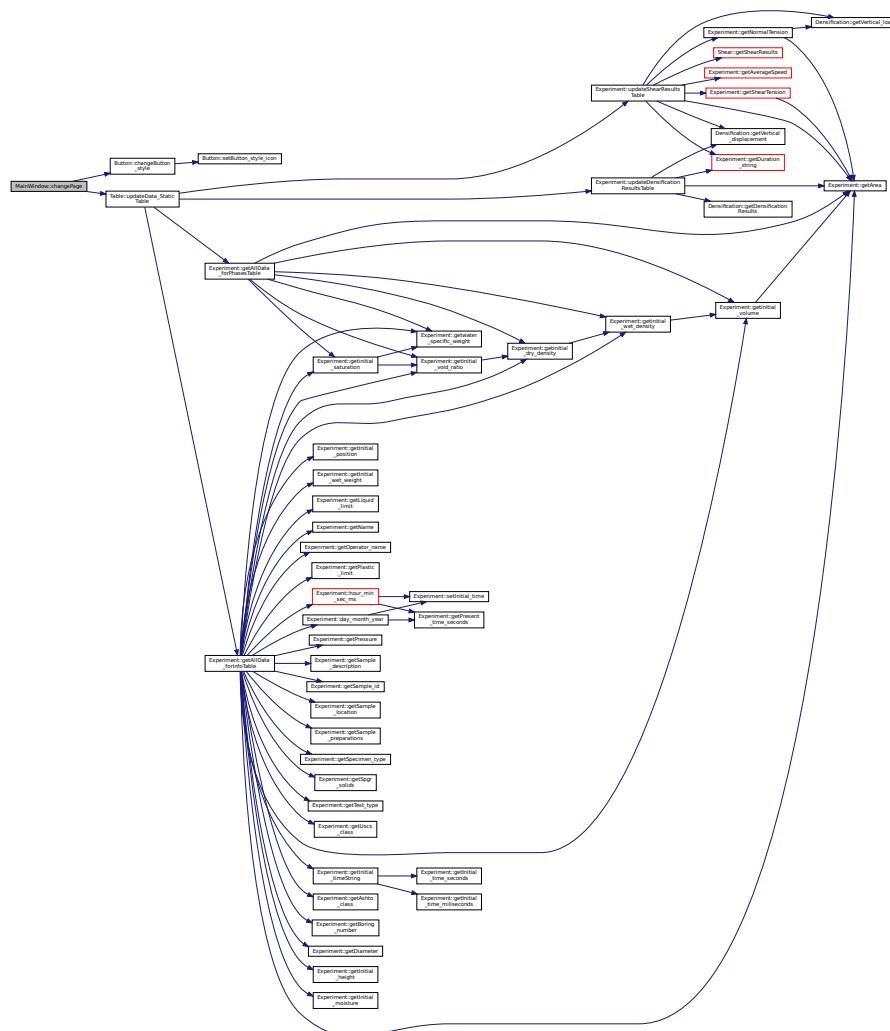
`QString buttons_name[6],`

```

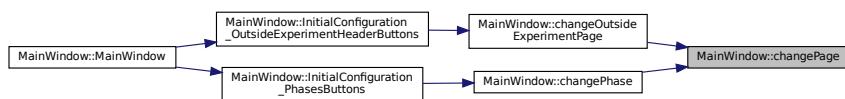
    uint8_t array_size,
    QStackedWidget * page_stack,
    uint8_t icon[3],
    uint8_t style ) [private]

```

Here is the call graph for this function:



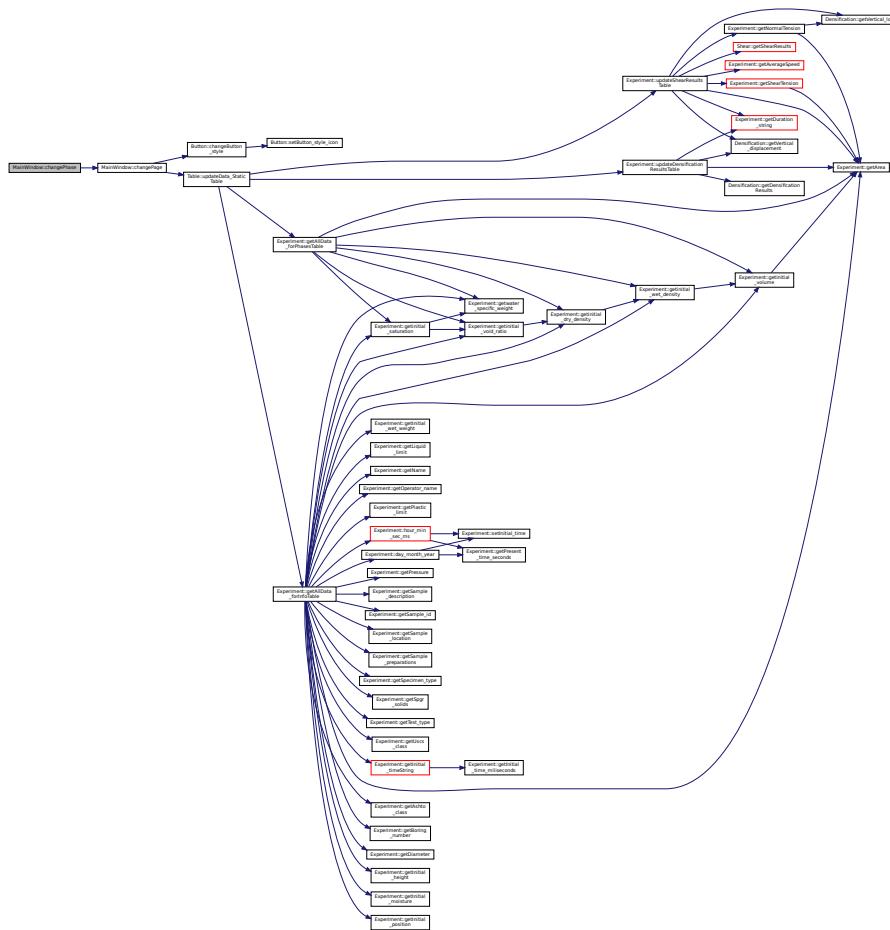
Here is the caller graph for this function:



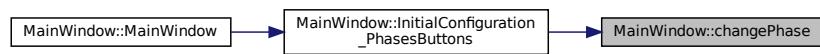
6.12.3.9 changePhase

```
void MainWindow::changePhase ( ) [private], [slot]
```

Here is the call graph for this function:



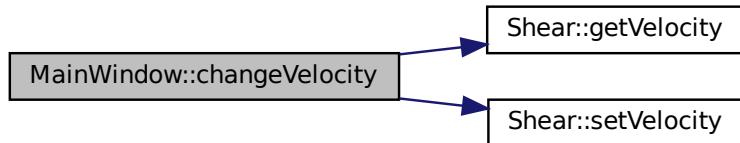
Here is the caller graph for this function:



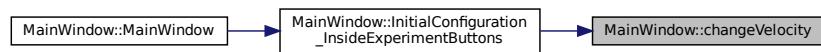
6.12.3.10 changeVelocity

```
void MainWindow::changeVelocity ( ) [private], [slot]
```

Here is the call graph for this function:



Here is the caller graph for this function:

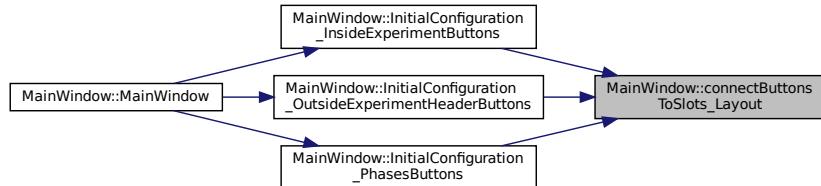


6.12.3.11 `connectButtonsToSlots_Layout()`

```
void MainWindow::connectButtonsToSlots_Layout (
```

QHBoxLayout * <i>list</i> ,	
const char * <i>signal</i> ,	
const char * <i>slot</i>) [private]	

Here is the caller graph for this function:



6.12.3.12 connectButtonsToSlots_Widget()

```
void MainWindow::connectButtonsToSlots_Widget (
    QObject * list,
    const char * signal,
    const char * slot ) [private]
```

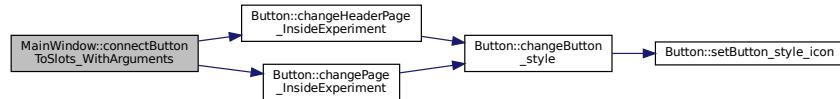
Here is the caller graph for this function:



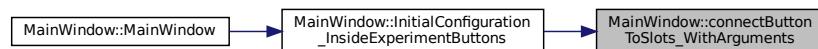
6.12.3.13 connectButtonToSlots_WithArguments()

```
void MainWindow::connectButtonToSlots_WithArguments (
    QBoxLayout * boxlayout,
    QStackedWidget * stack_widget,
    int option ) [private]
```

Here is the call graph for this function:



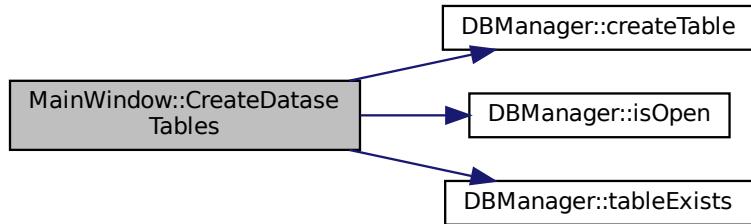
Here is the caller graph for this function:



6.12.3.14 CreateDatabaseTables()

```
void MainWindow::CreateDatabaseTables ( ) [private]
```

Here is the call graph for this function:



Here is the caller graph for this function:



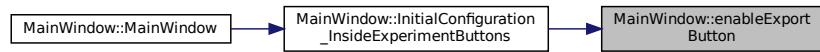
6.12.3.15 enableExportButton

```
void MainWindow::enableExportButton (
    int index ) [private], [slot]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.12.3.16 enableShearInitButton

```
void MainWindow::enableShearInitButton ( int index ) [private], [slot]
```

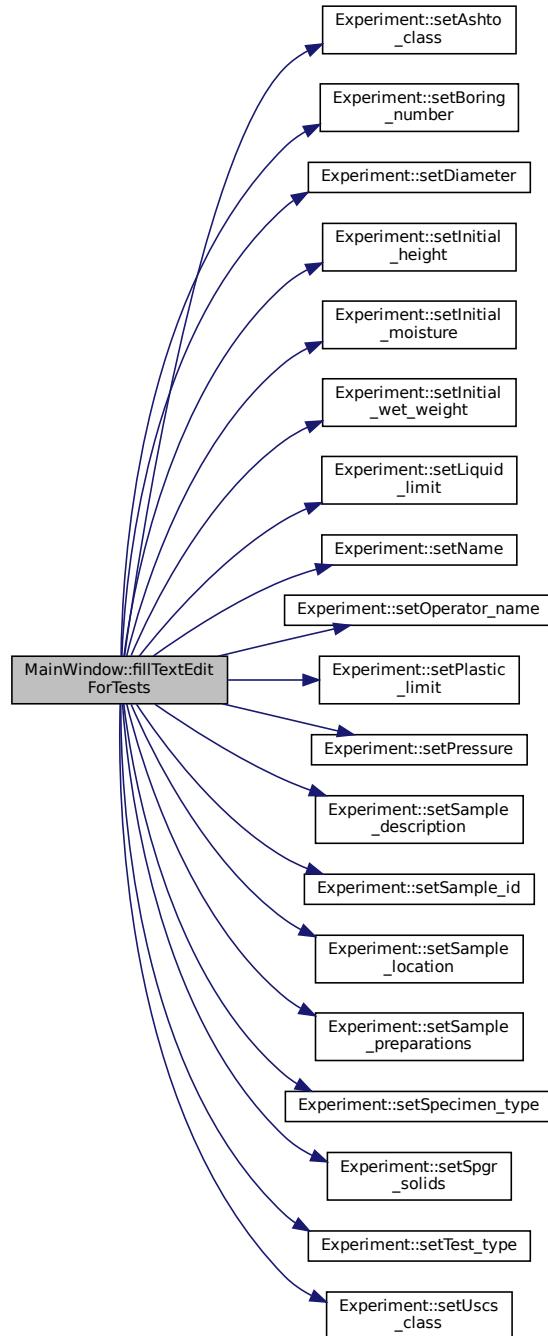
Here is the caller graph for this function:



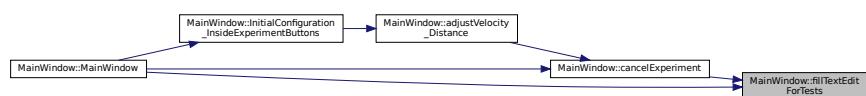
6.12.3.17 fillTextEditForTests()

```
void MainWindow::fillTextEditForTests ( ) [private]
```

Here is the call graph for this function:



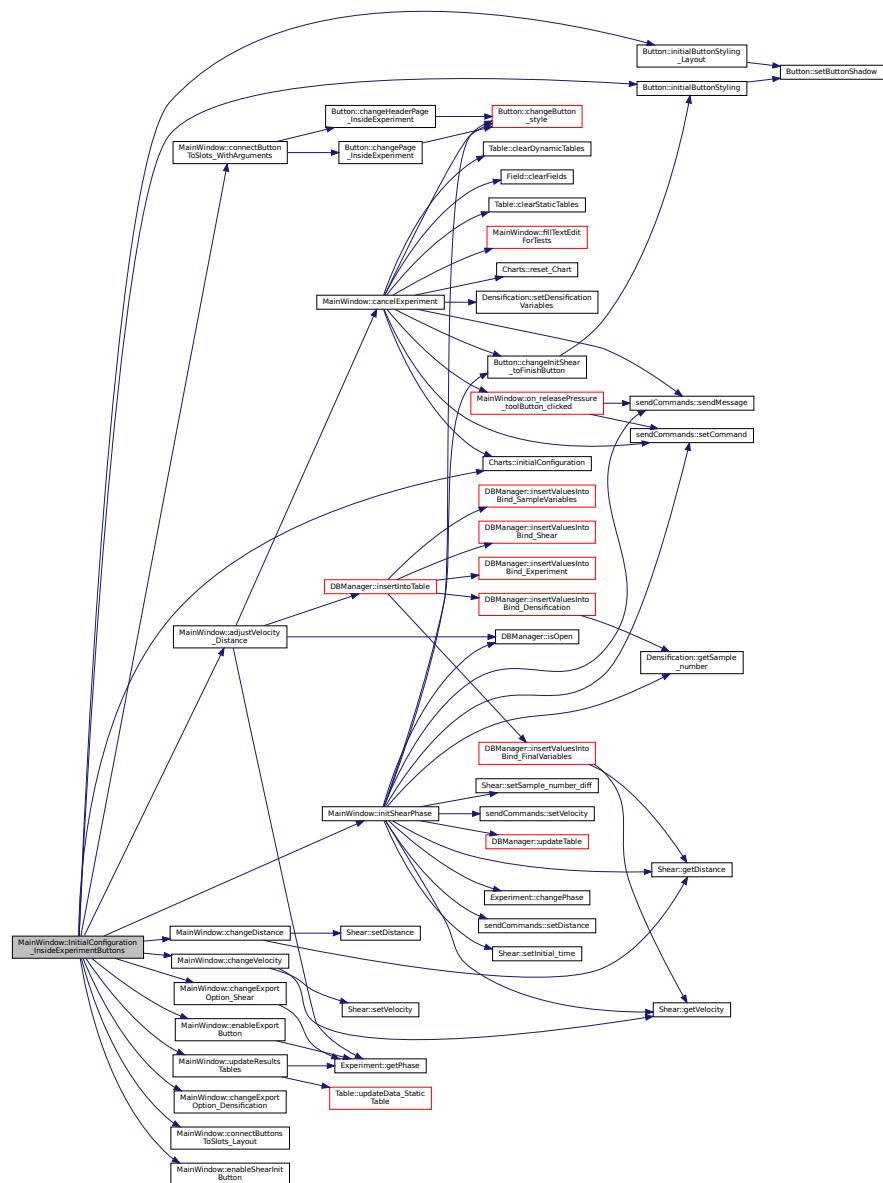
Here is the caller graph for this function:



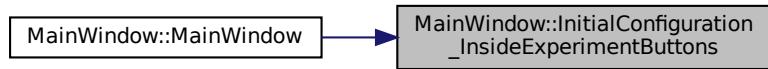
6.12.3.18 InitialConfiguration_InsideExperimentButtons()

```
void MainWindow::InitialConfiguration_InsideExperimentButtons ( ) [private]
```

Here is the call graph for this function:



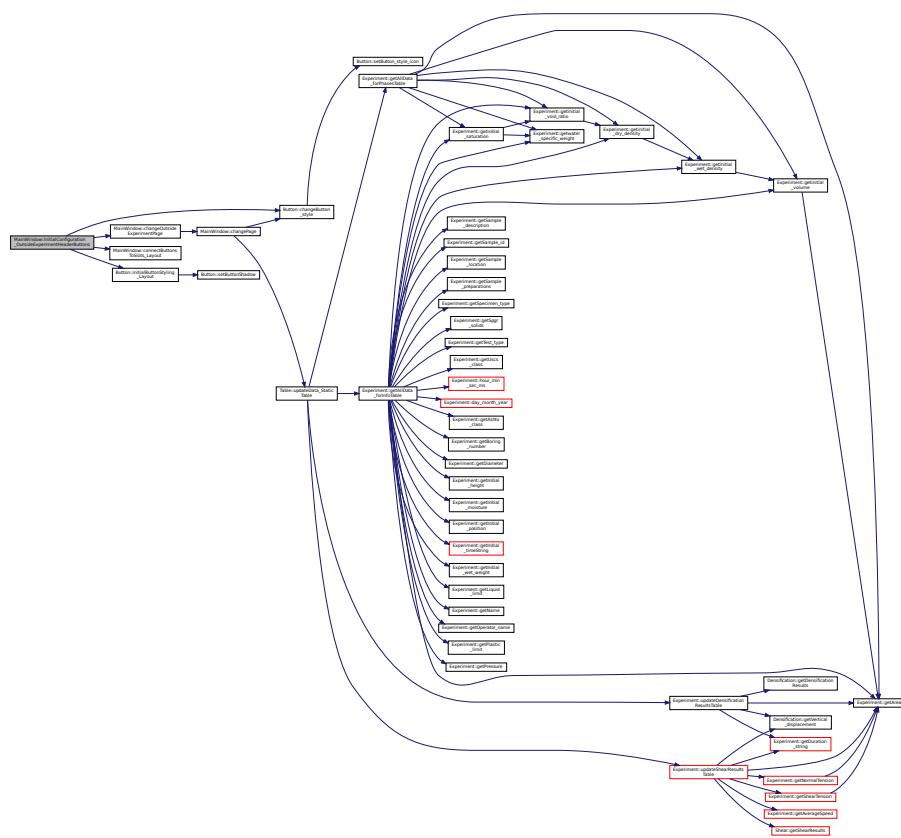
Here is the caller graph for this function:



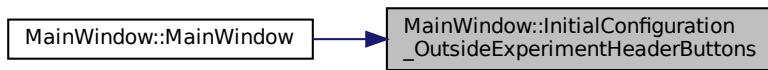
6.12.3.19 InitialConfiguration_OutsideExperimentHeaderButtons()

```
void MainWindow::InitialConfiguration_OutsideExperimentHeaderButtons () [private]
```

Here is the call graph for this function:



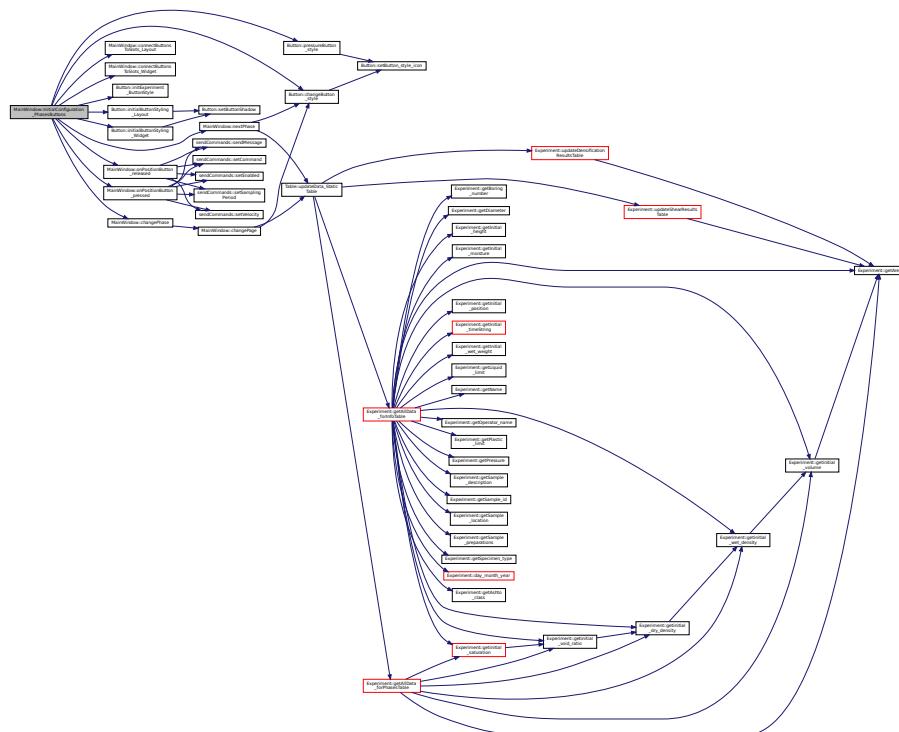
Here is the caller graph for this function:



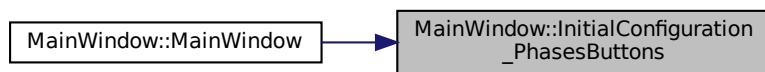
6.12.3.20 InitialConfiguration_PhasesButtons()

```
void MainWindow::InitialConfiguration_PhasesButtons ( ) [private]
```

Here is the call graph for this function:



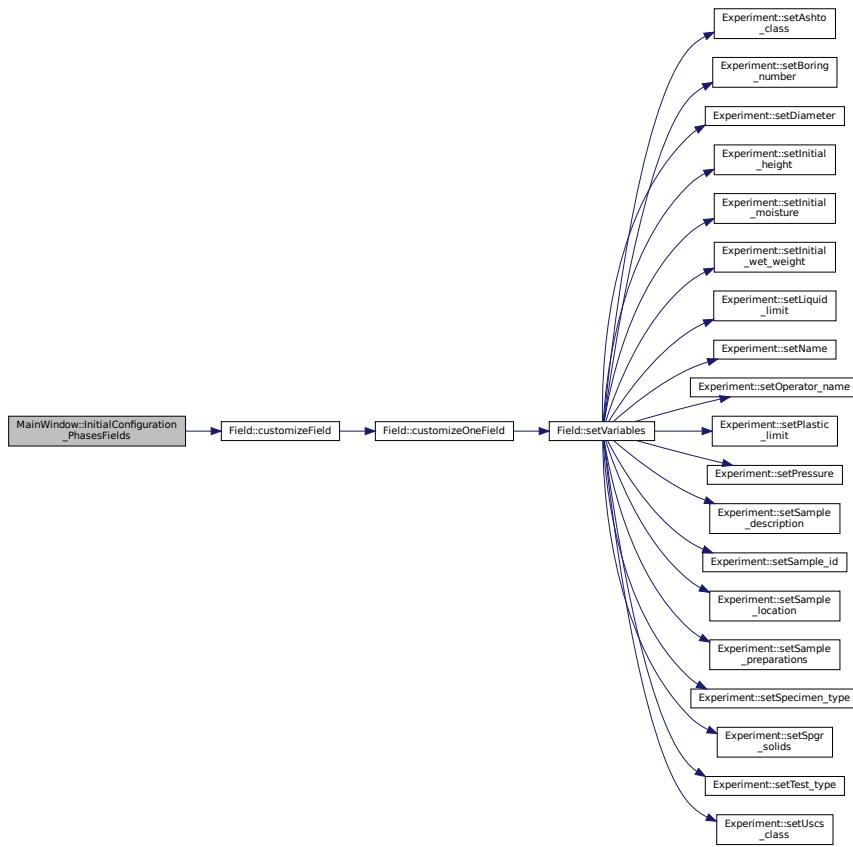
Here is the caller graph for this function:



6.12.3.21 InitialConfiguration PhasesFields()

```
void MainWindow::InitialConfiguration_PhasesFields ( ) [private]
```

Here is the call graph for this function:



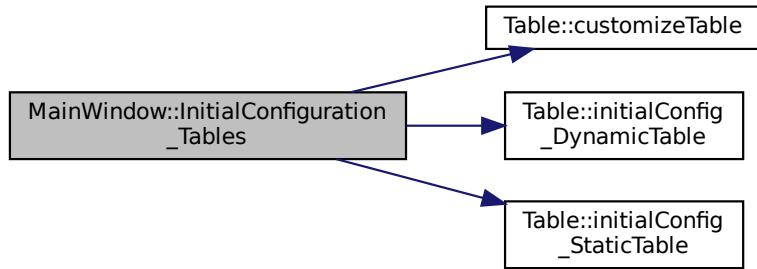
Here is the caller graph for this function:



6.12.3.22 InitialConfiguration_Tables()

```
void MainWindow::InitialConfiguration_Tables ( ) [private]
```

Here is the call graph for this function:



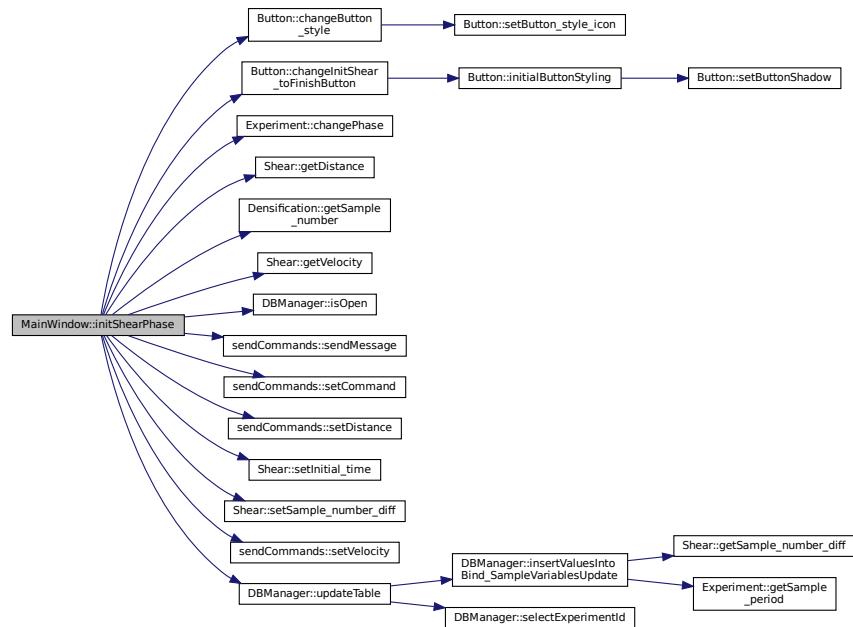
Here is the caller graph for this function:



6.12.3.23 initShearPhase

```
void MainWindow::initShearPhase ( ) [private], [slot]
```

Here is the call graph for this function:



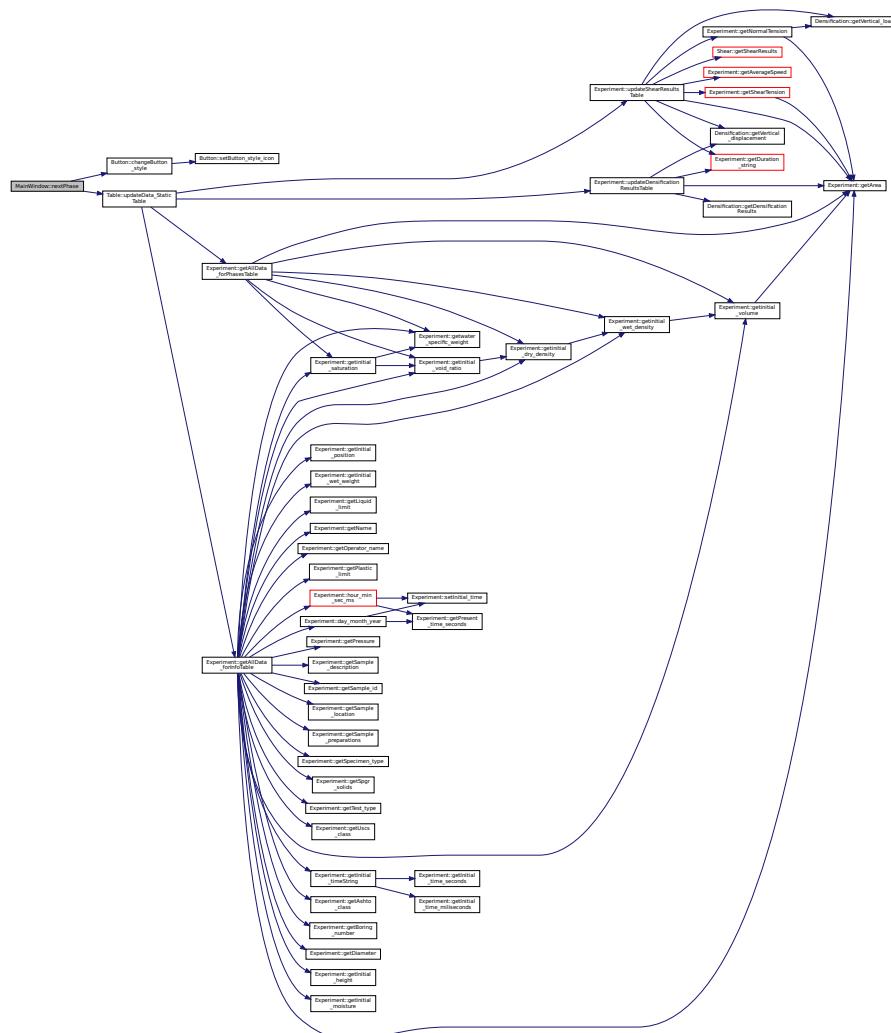
Here is the caller graph for this function:



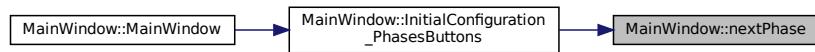
6.12.3.24 nextPhase

```
void MainWindow::nextPhase ( ) [private], [slot]
```

Here is the call graph for this function:



Here is the caller graph for this function:



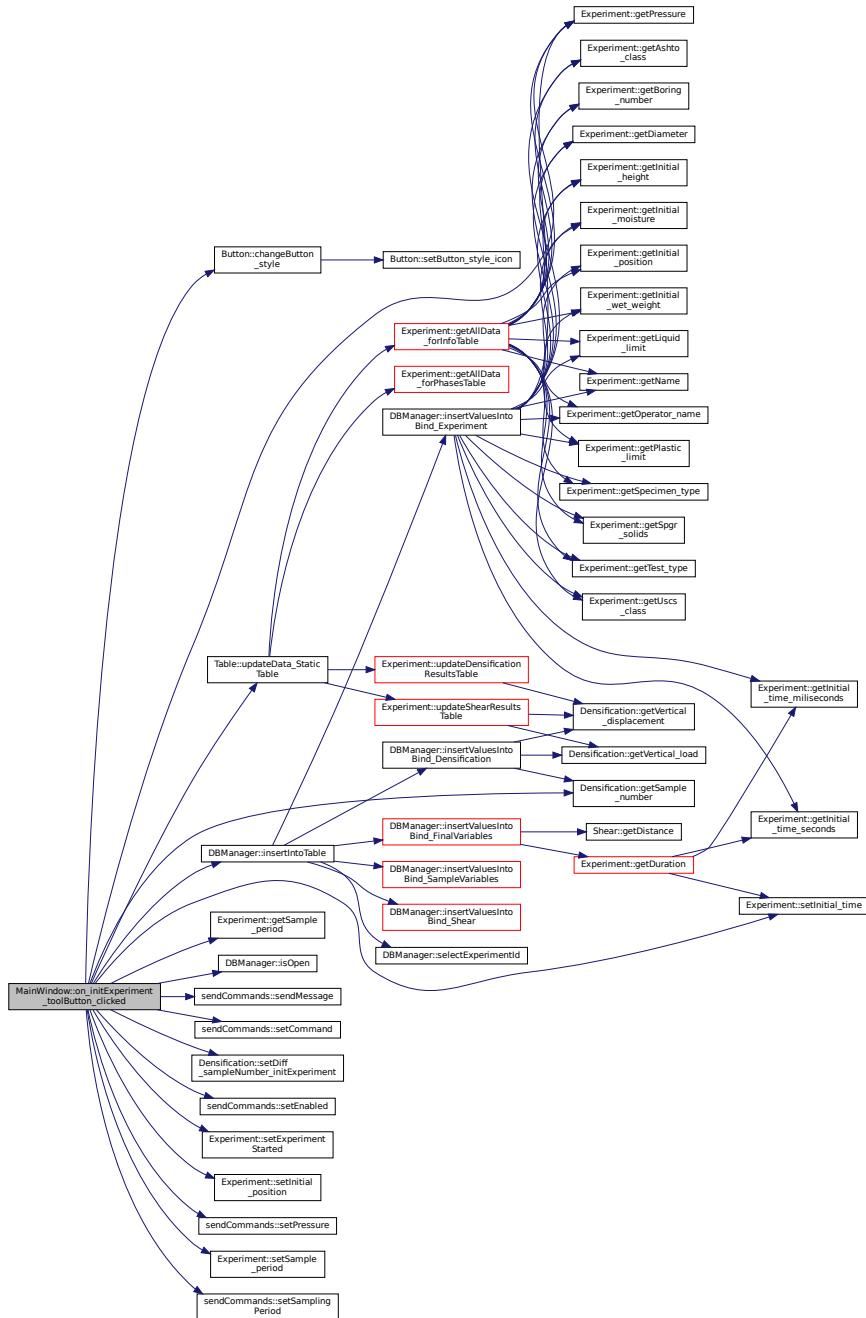
6.12.3.25 on_goBack_toolButton_clicked

```
void MainWindow::on_goBack_toolButton_clicked () [private], [slot]
```

6.12.3.26 on_initExperiment_toolButton_clicked

```
void MainWindow::on_initExperiment_toolButton_clicked () [private], [slot]
```

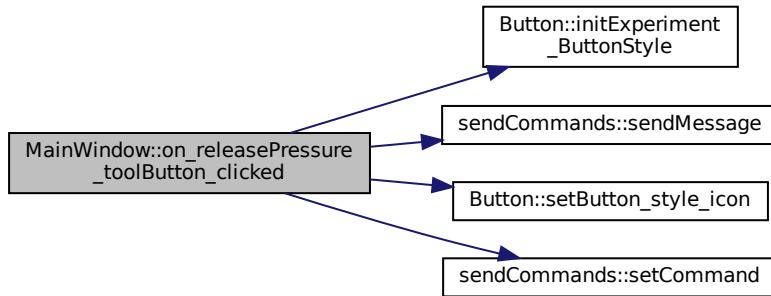
Here is the call graph for this function:



6.12.3.27 on_releasePressure_toolButton_clicked

```
void MainWindow::on_releasePressure_toolButton_clicked () [private], [slot]
```

Here is the call graph for this function:



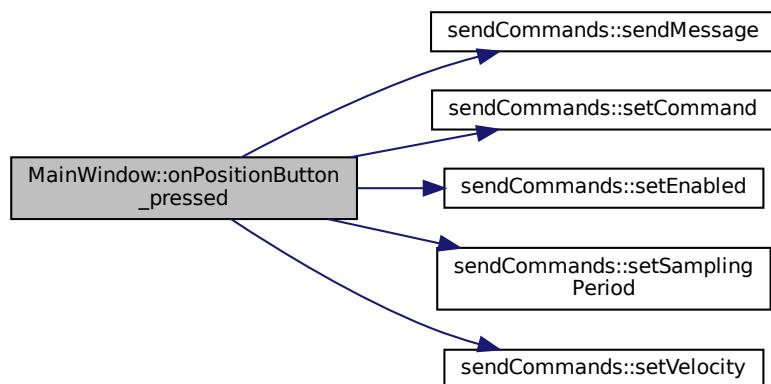
Here is the caller graph for this function:



6.12.3.28 `onPositionButton_pressed`

```
void MainWindow::onPositionButton_pressed ( ) [private], [slot]
```

Here is the call graph for this function:



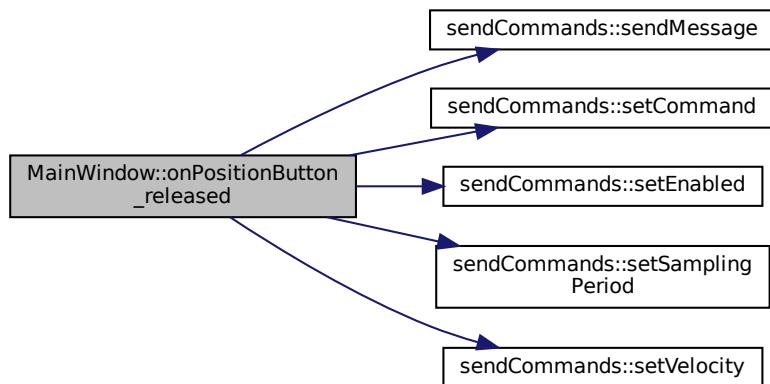
Here is the caller graph for this function:



6.12.3.29 onPositionButton_released

```
void MainWindow::onPositionButton_released () [private], [slot]
```

Here is the call graph for this function:



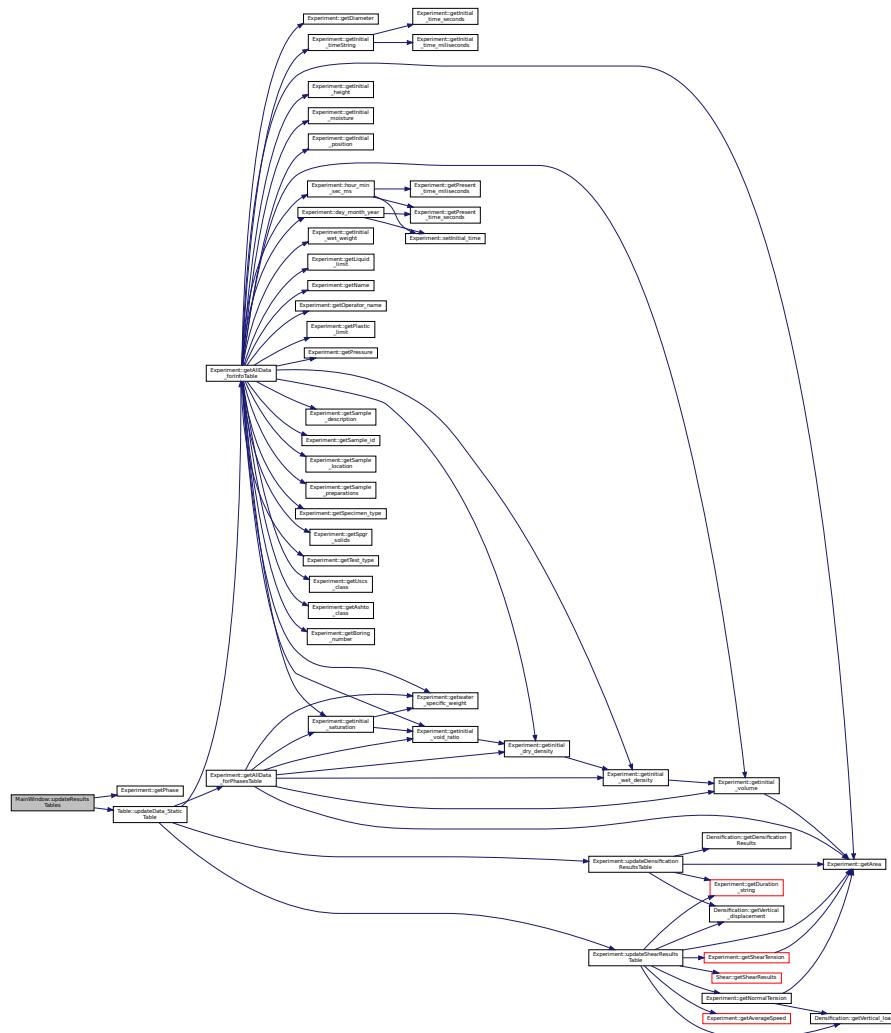
Here is the caller graph for this function:



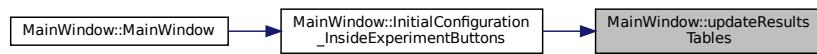
6.12.3.30 updateResultsTables

```
void MainWindow::updateResultsTables ( ) [private], [slot]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.12.4 Member Data Documentation

6.12.4.1 charts_variables

```
Charts* MainWindow::charts_variables [private]
```

6.12.4.2 chosenTable

```
QTableWidget* MainWindow::chosenTable
```

6.12.4.3 data_export

```
exportData* MainWindow::data_export
```

6.12.4.4 experiment_canceled

```
bool MainWindow::experiment_canceled = false [private]
```

6.12.4.5 export_option

```
int MainWindow::export_option
```

6.12.4.6 info_variables

```
Experiment* MainWindow::info_variables
```

6.12.4.7 my_db

```
DBManager* MainWindow::my_db
```

6.12.4.8 previousIndex

```
int MainWindow::previousIndex [private]
```

6.12.4.9 receive_data

```
ThreadController* MainWindow::receive_data [private]
```

6.12.4.10 send_data

```
sendCommands* MainWindow::send_data [private]
```

6.12.4.11 setupButtons

```
Button* MainWindow::setupButtons [private]
```

Instância da classe [Button](#) para a estilização dos botões.

6.12.4.12 setupFields

```
Field* MainWindow::setupFields [private]
```

6.12.4.13 tables

```
Table* MainWindow::tables [private]
```

6.12.4.14 timer

```
QTimer* MainWindow::timer [private]
```

6.12.4.15 ui

```
Ui::MainWindow* MainWindow::ui [private]
```

Instância da classe e serve para operar seus elementos, como botões.

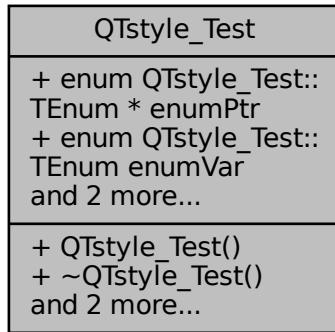
The documentation for this class was generated from the following files:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[mainwindow.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[mainwindow.cpp](#)

6.13 QTstyle_Test Class Reference

A test class.

Collaboration diagram for QTstyle_Test:



Public Types

- enum `TEnum { TVal1 , TVal2 , TVal3 }`

An enum.

Public Member Functions

- `QTstyle_Test ()`
A constructor.
- `~QTstyle_Test ()`
A destructor.
- int `testMe` (int a, const char *s)
A normal member taking two arguments and returning an integer value.
- virtual void `testMeToo` (char c1, char c2)=0
A pure virtual member.

Public Attributes

- enum `QTstyle_Test::TEnum * enumPtr`
Enum pointer.
- enum `QTstyle_Test::TEnum enumVar`
Enum variable.
- int `publicVar`
A public variable.
- int(* `handler`)(int a, int b)
A function variable.

6.13.1 Detailed Description

A test class.

A more elaborate class description.

6.13.2 Member Enumeration Documentation

6.13.2.1 TEnum

```
enum QTstyle_Test::TEnum
```

An enum.

More detailed enum description.

Enumerator

TVal1	Enum value TVal1.
TVal2	Enum value TVal2.
TVal3	Enum value TVal3.

6.13.3 Constructor & Destructor Documentation

6.13.3.1 QTstyle_Test()

```
QTstyle_Test::QTstyle_Test ( )
```

A constructor.

A more elaborate description of the constructor.

6.13.3.2 ~QTstyle_Test()

```
QTstyle_Test::~QTstyle_Test ( )
```

A destructor.

A more elaborate description of the destructor.

6.13.4 Member Function Documentation

6.13.4.1 testMe()

```
int QTstyle_Test::testMe (
    int a,
    const char * s )
```

A normal member taking two arguments and returning an integer value.

Parameters

<i>a</i>	an integer argument.
<i>s</i>	a constant character pointer.

Returns

The test results

See also

[QTstyle_Test\(\)](#), [~QTstyle_Test\(\)](#), [testMeToo\(\)](#) and [publicVar\(\)](#)

6.13.4.2 testMeToo()

```
virtual void QTstyle_Test::testMeToo (
    char c1,
    char c2 ) [pure virtual]
```

A pure virtual member.

See also

[testMe\(\)](#)

Parameters

<i>c1</i>	the first argument.
<i>c2</i>	the second argument.

6.13.5 Member Data Documentation

6.13.5.1 enumPtr

```
enum QTstyle_Test::TEnum * QTstyle_Test::enumPtr
```

Enum pointer.

Details.

6.13.5.2 enumVar

```
enum QTstyle_Test::TEnum QTstyle_Test::enumVar
```

Enum variable.

Details.

6.13.5.3 handler

```
int(* QTstyle_Test::handler) (int a, int b)
```

A function variable.

Details.

6.13.5.4 publicVar

```
int QTstyle_Test::publicVar
```

A public variable.

Details.

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

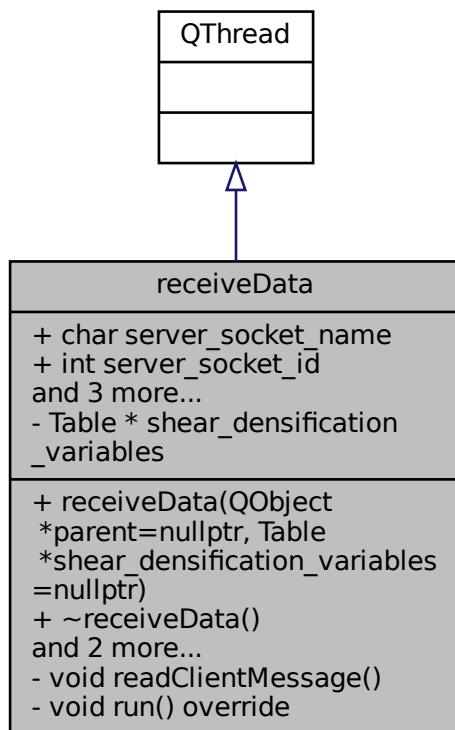
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/testDoxygen.cpp

6.14 receiveData Class Reference

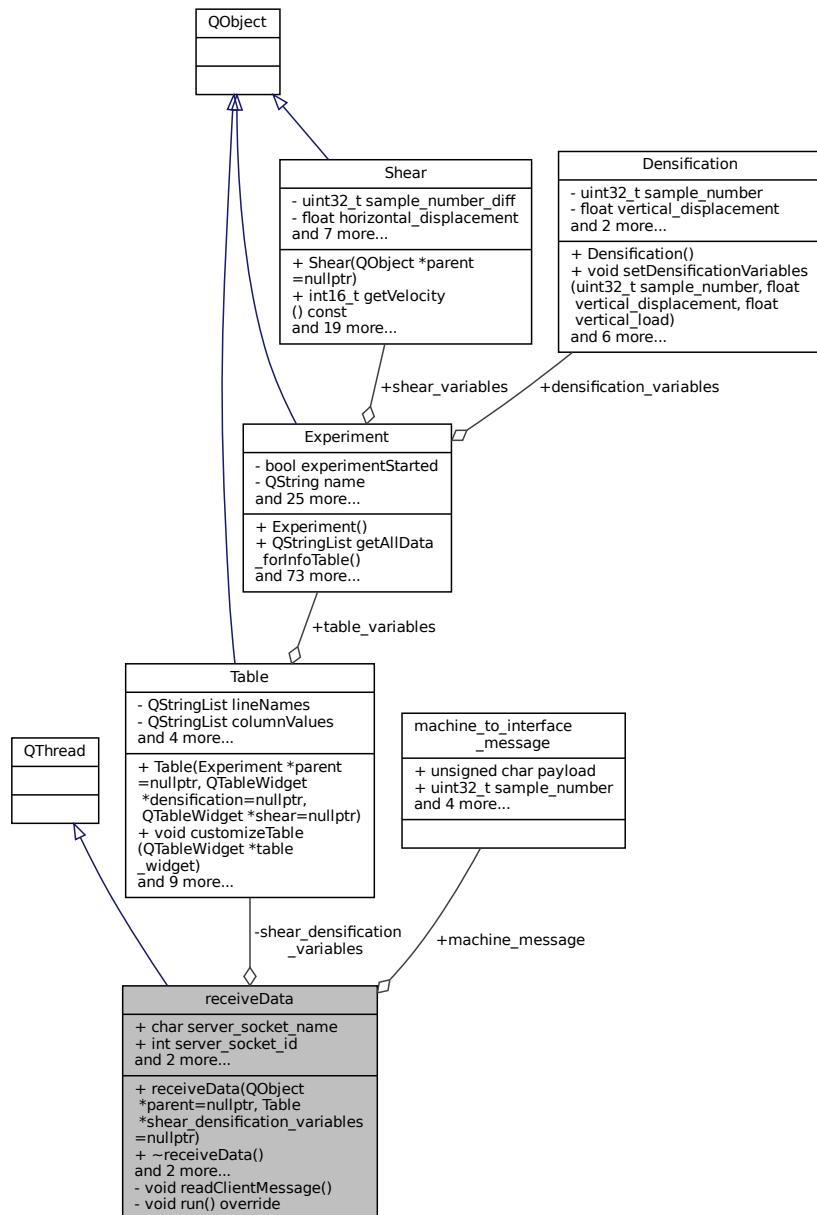
Classe que recebe os dados da camada de controle.

```
#include <receivedata.h>
```

Inheritance diagram for receiveData:



Collaboration diagram for receiveData:



Signals

- `void data_arrived ()`

Public Member Functions

- `receiveData (QObject *parent=nullptr, Table *shear_densification_variables=nullptr)`
Constrói uma nova instância da classe receiveData.
- `~receiveData ()`

Destroi a instânciada classe `receiveData`.

- `uint8_t initSocketServer ()`
Configuração inicial do socket servidor.
- `void init ()`
Inicia a thread.

Public Attributes

- `char server_socket_name [100]`
- `int server_socket_id`
- `int client_socket_id`
- `uint8_t errorOccurred`
- `machine_to_interface_message machine_message`

Private Member Functions

- `void readClientMessage ()`
Lê as mensagens enviadas pela camada de controle.
- `void run () override`
Roda a thread.

Private Attributes

- `Table * shear_densification_variables`

6.14.1 Detailed Description

Classe que recebe os dados da camada de controle.

Esta classe é responsável por administrar a recepção de dados, enviados pela camada de controle.

Esta classe é uma Thread.

6.14.2 Constructor & Destructor Documentation

6.14.2.1 `receiveData()`

```
receiveData::receiveData (
    QObject * parent = nullptr,
    Table * shear_densification_variables = nullptr )
```

Constrói uma nova instância da classe `receiveData`.

Iguala o ponteiro da classe `Table` à instância criada na `MainWindow`.

Parameters

<i>parent</i>	Objeto pai
<i>shear_densification_variables</i>	instância criada na MainWindow da classe Table

6.14.2.2 ~receiveData()

```
receiveData::~receiveData ( )
```

Destroi a instânciada classe [receiveData](#).

Fechas os sockets do servidor e cliente.

6.14.3 Member Function Documentation**6.14.3.1 data_arrived**

```
void receiveData::data_arrived ( ) [signal]
```

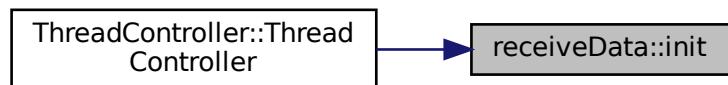
Here is the caller graph for this function:

**6.14.3.2 init()**

```
void receiveData::init ( )
```

Inicia a thread.

Here is the caller graph for this function:



6.14.3.3 initSocketServer()

```
uint8_t receiveData::initSocketServer ( )
```

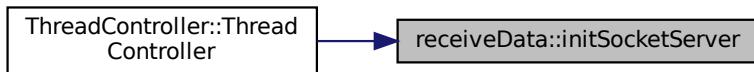
Configuração inicial do socket servidor.

Nesta função se realiza toda a configuração inicial do socket servidor, para ser utilizada posteriormente.

Returns

uint8_t 0 em caso de erro e 1 para sucesso.

Here is the caller graph for this function:



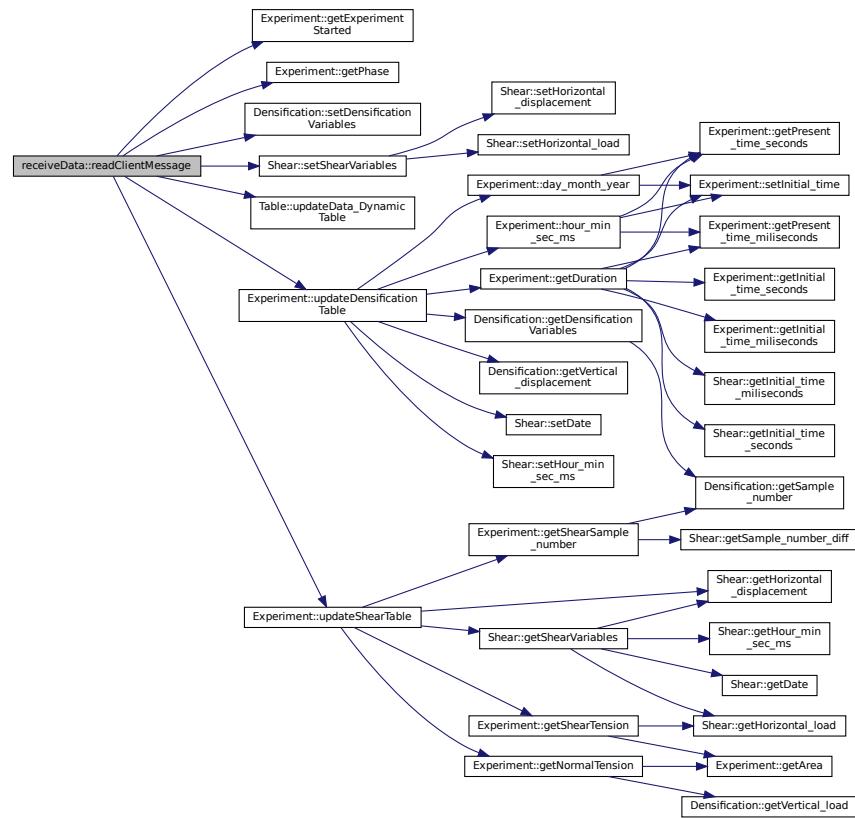
6.14.3.4 readClientMessage()

```
void receiveData::readClientMessage ( ) [private]
```

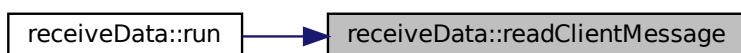
Lê as mensagens enviadas pela camada de controle.

Além da leitura, esta função também define as variáveis de adensamento e cisalhamento, atualiza as tabelas e

envia um sinal para o banco de dados salvar novos dados. Here is the call graph for this function:



Here is the caller graph for this function:

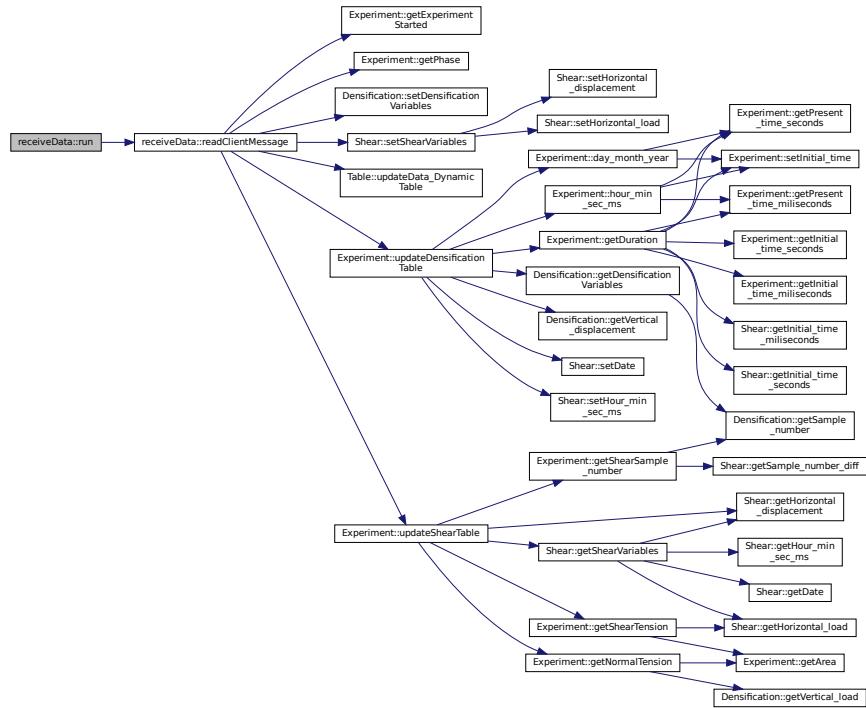


6.14.3.5 run()

```
void receiveData::run( ) [override], [private]
```

Roda a thread.

Esta função é chamada logo após o início da thread e o fim de sua execução significa o fim da thread. Here is the call graph for this function:



6.14.4 Member Data Documentation

6.14.4.1 client_socket_id

```
int receiveData::client_socket_id
```

Identificador do socket cliente (Camada de controle).

6.14.4.2 errorOccurred

```
uint8_t receiveData::errorOccurred
```

Identifica se algum erro ocorreu nas configurações iniciais.

6.14.4.3 machine_message

```
machine_to_interface_message receiveData::machine_message
```

Essa é a estrutura de envio de dados da camada de controle para a Interface.

6.14.4.4 server_socket_id

```
int receiveData::server_socket_id
```

Identificador do socket servidor (Interface).

6.14.4.5 server_socket_name

```
char receiveData::server_socket_name[100]
```

Nome do socket, o socket cliente da camada de controle deve possuir o mesmo nome .

6.14.4.6 shear_densification_variables

```
Table* receiveData::shear_densification_variables [private]
```

Receberá a instância da classe [Table](#) criada na [MainWindow](#).

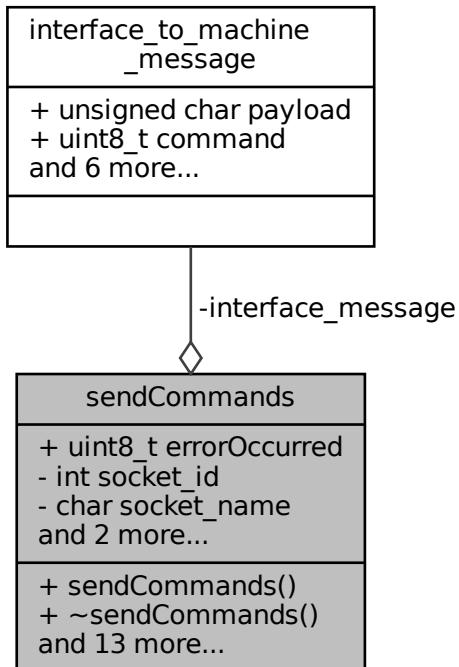
The documentation for this class was generated from the following files:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[receivedata.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[receivedata.cpp](#)

6.15 sendCommands Class Reference

```
#include <sendcommands.h>
```

Collaboration diagram for sendCommands:



Public Member Functions

- `sendCommands ()`
- `~sendCommands ()`
- `void setCommand (uint8_t command)`
- `uint8_t connectToMachine ()`
- `int16_t sendMessage ()`
- `void setVelocity (int16_t velocity)`
- `void setDistance (int16_t distance)`
- `void setPressure (uint8_t pressure)`
- `void setEnabled (uint8_t enabled)`
- `void setSamplingPeriod (uint16_t sample_period)`
- `int16_t getVelocity ()`
- `int16_t getDistance ()`
- `uint16_t getPressure ()`
- `uint8_t getEnabled ()`
- `uint16_t getSamplingPeriod ()`

Public Attributes

- `uint8_t errorOccurred`

Private Attributes

- `int socket_id`
- `char socket_name [100]`
- `interface_to_machine_message interface_message`
- `bool enabled`
- `int16_t sampling_period`

6.15.1 Constructor & Destructor Documentation

6.15.1.1 `sendCommands()`

```
sendCommands::sendCommands ( )
```

6.15.1.2 `~sendCommands()`

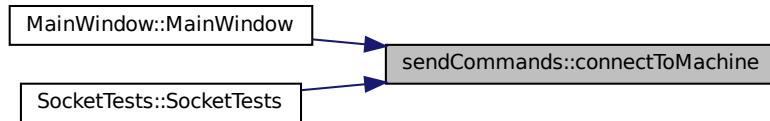
```
sendCommands::~sendCommands ( )
```

6.15.2 Member Function Documentation

6.15.2.1 `connectToMachine()`

```
uint8_t sendCommands::connectToMachine ( )
```

Here is the caller graph for this function:



6.15.2.2 `getDistance()`

```
int16_t sendCommands::getDistance ( )
```

6.15.2.3 `getEnabled()`

```
uint8_t sendCommands::getEnabled ( )
```

6.15.2.4 `getPressure()`

```
uint16_t sendCommands::getPressure ( )
```

6.15.2.5 `getSamplingPeriod()`

```
uint16_t sendCommands::getSamplingPeriod ( )
```

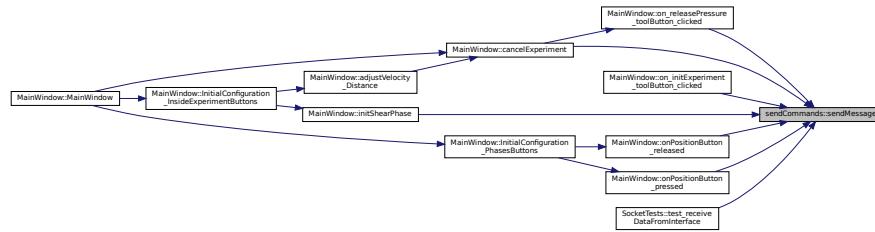
6.15.2.6 `getVelocity()`

```
int16_t sendCommands::getVelocity ( )
```

6.15.2.7 sendMessage()

```
int16_t sendCommands::sendMessage ( )
```

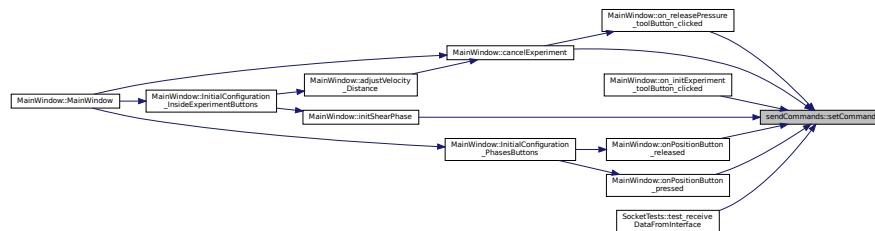
Here is the caller graph for this function:



6.15.2.8 setCommand()

```
void sendCommands::setCommand (
    uint8_t command )
```

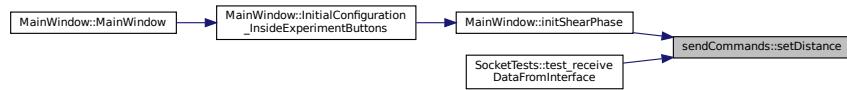
Here is the caller graph for this function:



6.15.2.9 setDistance()

```
void sendCommands::setDistance (
    int16_t distance )
```

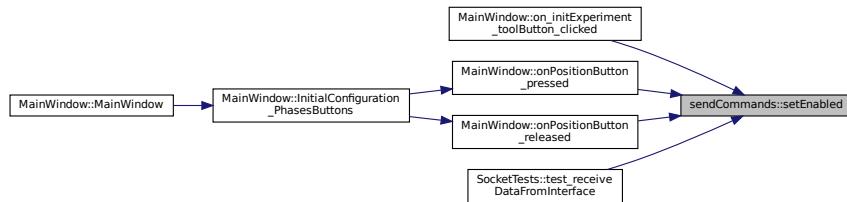
Here is the caller graph for this function:



6.15.2.10 setEnabled()

```
void sendCommands::setEnabled (
    uint8_t enabled )
```

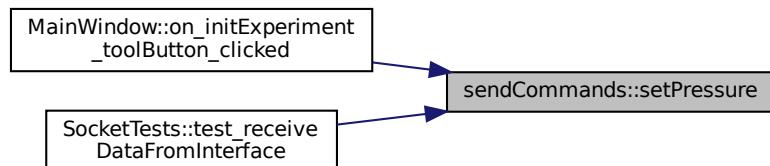
Here is the caller graph for this function:



6.15.2.11 setPressure()

```
void sendCommands::setPressure (
    uint8_t pressure )
```

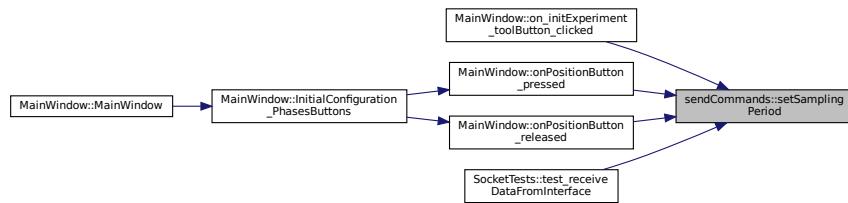
Here is the caller graph for this function:



6.15.2.12 setSamplingPeriod()

```
void sendCommands::setSamplingPeriod (
    uint16_t sample_period )
```

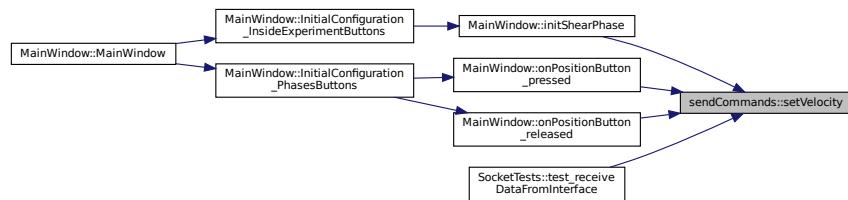
Here is the caller graph for this function:



6.15.2.13 setVelocity()

```
void sendCommands::setVelocity (
    int16_t velocity )
```

Here is the caller graph for this function:



6.15.3 Member Data Documentation

6.15.3.1 enabled

```
bool sendCommands::enabled [private]
```

6.15.3.2 errorOccurred

```
uint8_t sendCommands::errorOccurred
```

6.15.3.3 interface_message

```
interface_to_machine_message sendCommands::interface_message [private]
```

6.15.3.4 sampling_period

```
int16_t sendCommands::sampling_period [private]
```

6.15.3.5 socket_id

```
int sendCommands::socket_id [private]
```

6.15.3.6 socket_name

```
char sendCommands::socket_name[100] [private]
```

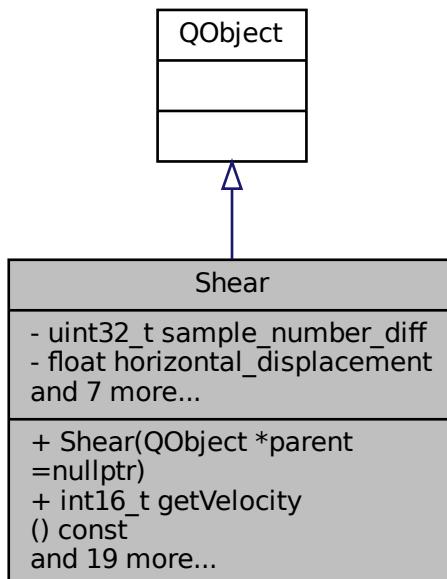
The documentation for this class was generated from the following files:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/sendcommands.h
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/sendcommands.cpp

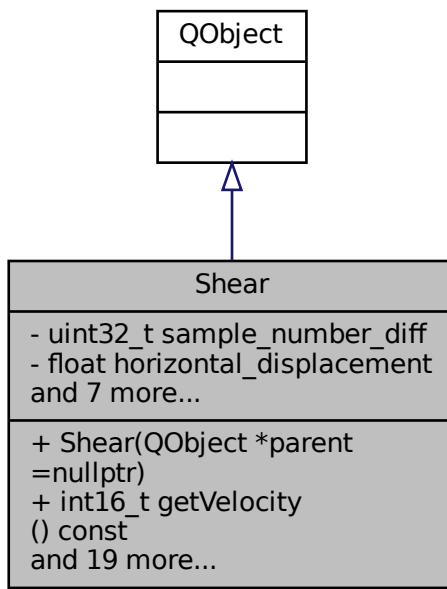
6.16 Shear Class Reference

```
#include <shear.h>
```

Inheritance diagram for Shear:



Collaboration diagram for Shear:



Public Member Functions

- `Shear (QObject *parent=nullptr)`
- `int16_t getVelocity () const`
- `void setVelocity (int16_t newVelocity)`
- `int16_t getDistance () const`
- `void setDistance (int16_t newDistance)`
- `void setShearVariables (float horizontal_displacement, float horizontal_load)`
- `QStringList getShearVariables ()`
- `QStringList getShearResults ()`
- `uint32_t getSample_number_diff () const`
- `void setSample_number_diff (uint32_t newSample_number_diff)`
- `float getHorizontal_displacement () const`
- `void setHorizontal_displacement (float newHorizontal_displacement)`
- `float getHorizontal_load () const`
- `void setHorizontal_load (float newHorizontal_load)`
- `const QString & getDate () const`
- `void setDate (const QString &newDate)`
- `const QString & getHour_min_sec_ms () const`
- `void setHour_min_sec_ms (const QString &newHour_min_sec_ms)`
- `uint64_t getInitial_time_seconds () const`
- `uint64_t getInitial_time_milliseconds () const`
- `void setInitial_time ()`

Private Attributes

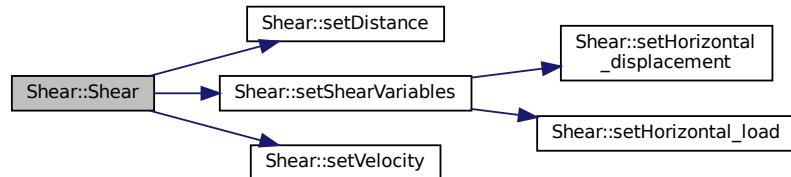
- `uint32_t sample_number_diff`
- `float horizontal_displacement`
- `float horizontal_load`
- `int16_t velocity`
- `int16_t distance`
- `QString date`
- `QString hour_min_sec_ms`
- `uint64_t initial_time_seconds`
- `uint64_t initial_time_milliseconds`

6.16.1 Constructor & Destructor Documentation

6.16.1.1 Shear()

```
Shear::Shear (
    QObject * parent = nullptr ) [explicit]
```

Here is the call graph for this function:



6.16.2 Member Function Documentation

6.16.2.1 getDate()

```
const QString & Shear::getDate ( ) const
```

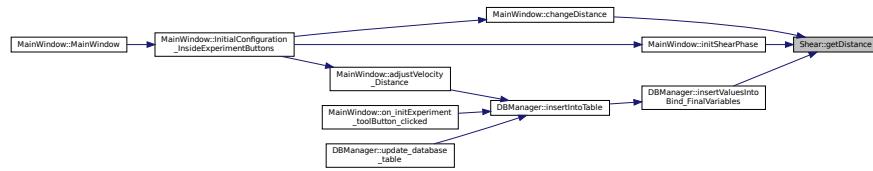
Here is the caller graph for this function:



6.16.2.2 getDistance()

```
int16_t Shear::getDistance ( ) const
```

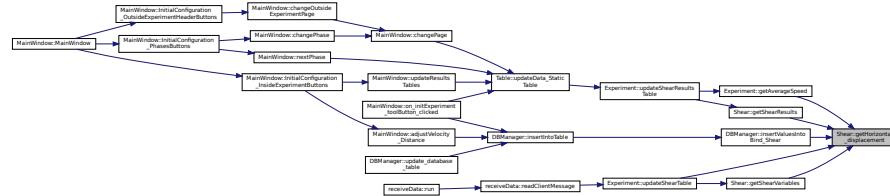
Here is the caller graph for this function:



6.16.2.3 getHorizontal_displacement()

```
float Shear::getHorizontal_displacement ( ) const
```

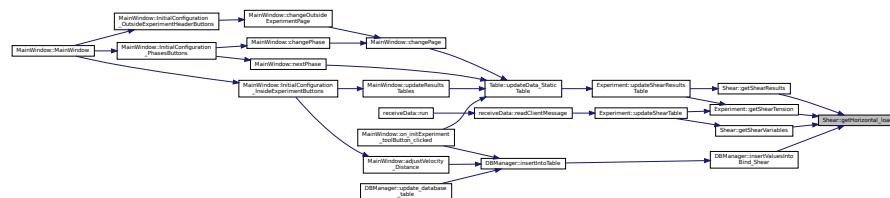
Here is the caller graph for this function:



6.16.2.4 getHorizontal_load()

```
float Shear::getHorizontal_load ( ) const
```

Here is the caller graph for this function:



6.16.2.5 getHour_min_sec_ms()

```
const QString & Shear::getHour_min_sec_ms ( ) const
```

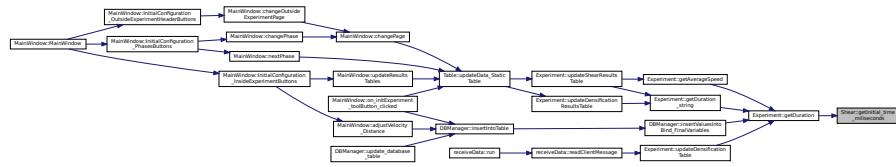
Here is the caller graph for this function:



6.16.2.6 getInitial_time_milliseconds()

```
uint64_t Shear::getInitial_time_milliseconds ( ) const
```

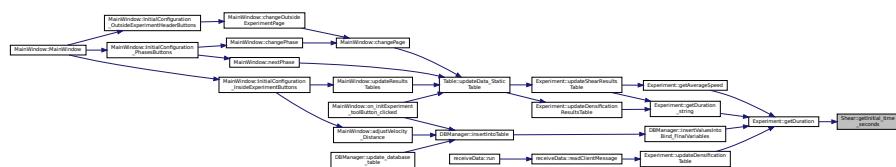
Here is the caller graph for this function:



6.16.2.7 getInitial_time_seconds()

```
uint64_t Shear::getInitial_time_seconds ( ) const
```

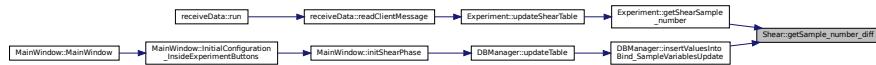
Here is the caller graph for this function:



6.16.2.8 getSample_number_diff()

```
uint32_t Shear::getSample_number_diff ( ) const
```

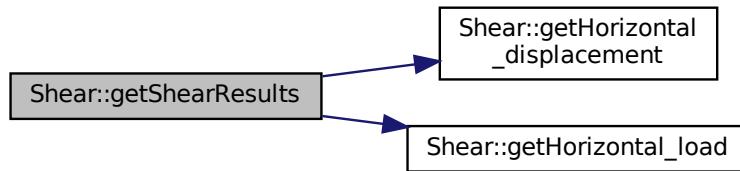
Here is the caller graph for this function:



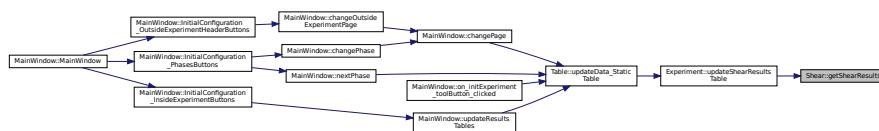
6.16.2.9 getShearResults()

```
QStringList Shear::getShearResults ( )
```

Here is the call graph for this function:



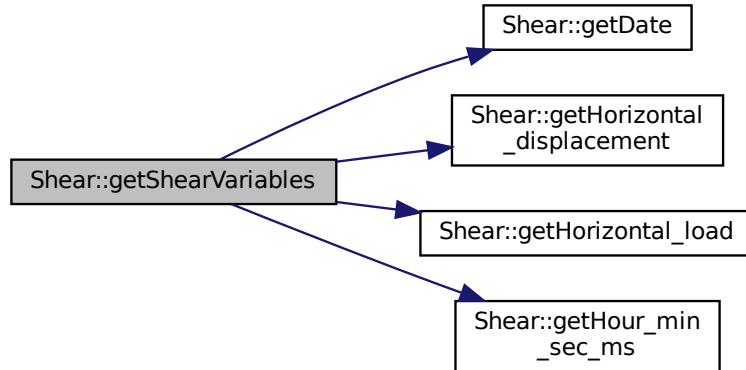
Here is the caller graph for this function:



6.16.2.10 getShearVariables()

```
QStringList Shear::getShearVariables ( )
```

Here is the call graph for this function:



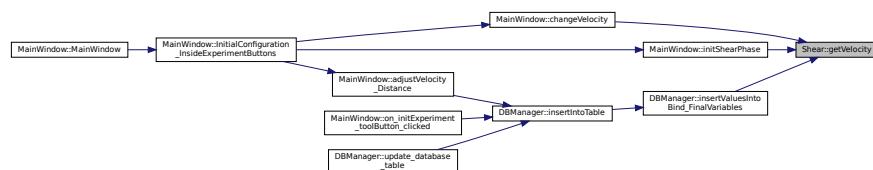
Here is the caller graph for this function:



6.16.2.11 getVelocity()

```
int16_t Shear::getVelocity ( ) const
```

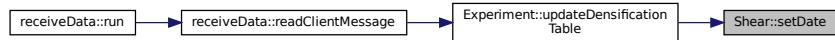
Here is the caller graph for this function:



6.16.2.12 setDate()

```
void Shear::setDate (
    const QString & newDate )
```

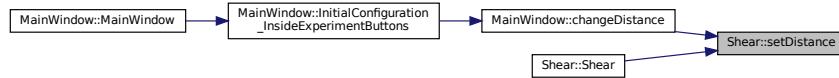
Here is the caller graph for this function:



6.16.2.13 setDistance()

```
void Shear::setDistance (
    int16_t newDistance )
```

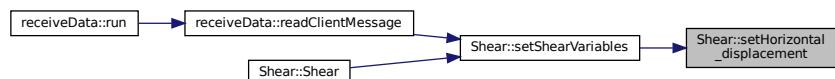
Here is the caller graph for this function:



6.16.2.14 setHorizontal_displacement()

```
void Shear::setHorizontal_displacement (
    float newHorizontal_displacement )
```

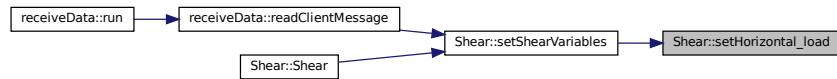
Here is the caller graph for this function:



6.16.2.15 setHorizontal_load()

```
void Shear::setHorizontal_load (
    float newHorizontal_load )
```

Here is the caller graph for this function:



6.16.2.16 setHour_min_sec_ms()

```
void Shear::setHour_min_sec_ms (
    const QString & newHour_min_sec_ms )
```

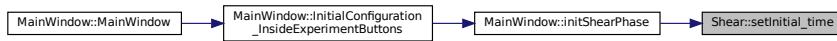
Here is the caller graph for this function:



6.16.2.17 setInitial_time()

```
void Shear::setInitial_time ( )
```

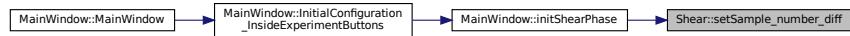
Here is the caller graph for this function:



6.16.2.18 setSample_number_diff()

```
void Shear::setSample_number_diff (
    uint32_t newSample_number_diff )
```

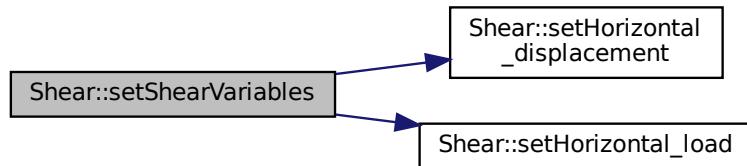
Here is the caller graph for this function:



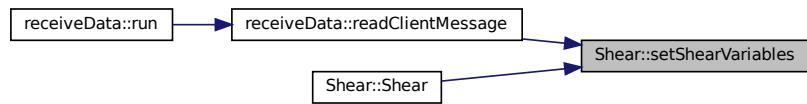
6.16.2.19 setShearVariables()

```
void Shear::setShearVariables (
    float horizontal_displacement,
    float horizontal_load )
```

Here is the call graph for this function:



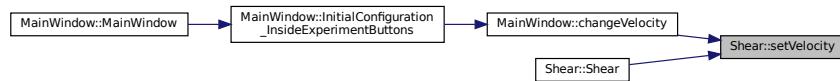
Here is the caller graph for this function:



6.16.2.20 setVelocity()

```
void Shear::setVelocity (
    int16_t newVelocity )
```

Here is the caller graph for this function:



6.16.3 Member Data Documentation

6.16.3.1 date

```
QString Shear::date [private]
```

6.16.3.2 distance

```
int16_t Shear::distance [private]
```

6.16.3.3 horizontal_displacement

```
float Shear::horizontal_displacement [private]
```

6.16.3.4 horizontal_load

```
float Shear::horizontal_load [private]
```

6.16.3.5 hour_min_sec_ms

```
QString Shear::hour_min_sec_ms [private]
```

6.16.3.6 `initial_time_milliseconds`

```
uint64_t Shear::initial_time_milliseconds [private]
```

6.16.3.7 `initial_time_seconds`

```
uint64_t Shear::initial_time_seconds [private]
```

6.16.3.8 `sample_number_diff`

```
uint32_t Shear::sample_number_diff [private]
```

6.16.3.9 `velocity`

```
int16_t Shear::velocity [private]
```

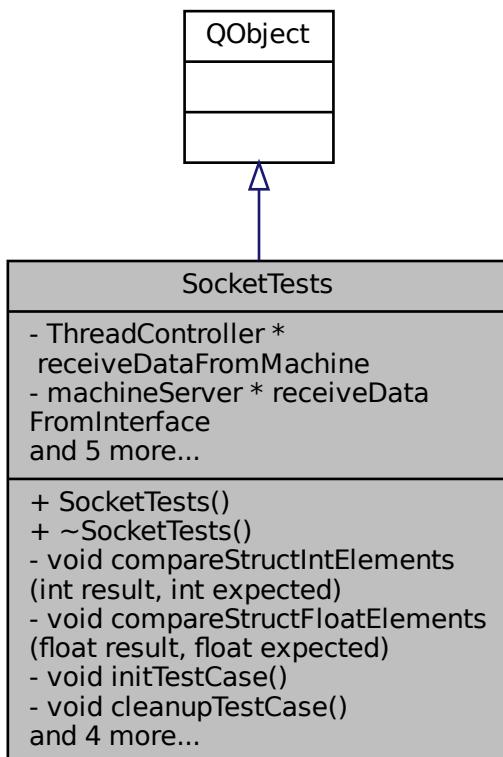
The documentation for this class was generated from the following files:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[shear.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[shear.cpp](#)

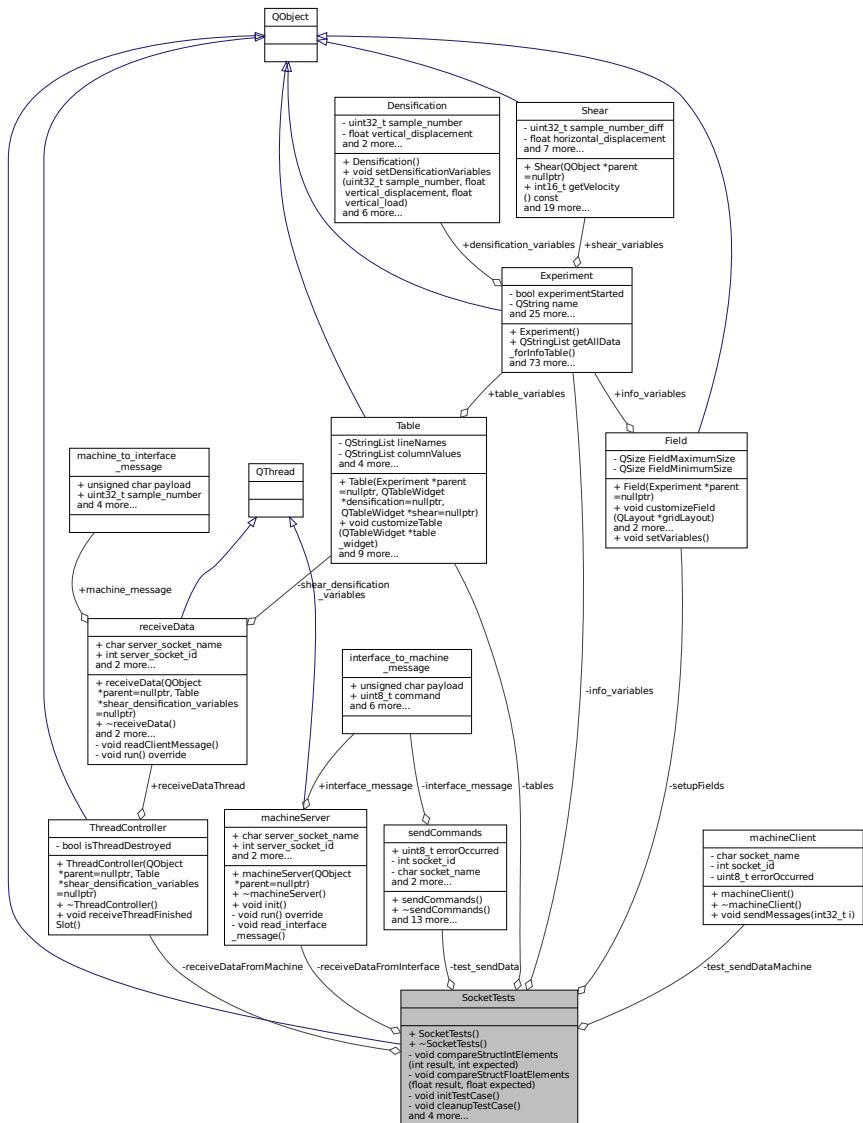
6.17 SocketTests Class Reference

Classe de testes para o socket.

Inheritance diagram for SocketTests:



Collaboration diagram for SocketTests:



Public Member Functions

- [SocketTests \(\)](#)

Contrutor da classe de testes do socket. Pode ser utilizado para instanciar classes, diferentes da atual, ou para configurações iniciais.

- [~SocketTests \(\)](#)

Destruitor da classe de testes do socket.

Private Slots

- [void initTestCase \(\)](#)

Primeiro teste.

- [void cleanupTestCase \(\)](#)

Último teste.

- void [test_receiveDataFromMachine \(\)](#)
Teste que verifica se a interface está recebendo corretamente os dados da máquina.
- void [test_receiveDataFromMachine_data \(\)](#)
Função que possui os dados necessários para realizar o teste test_receiveDataFromMachine.
- void [test_receiveDataFromInterface \(\)](#)
Teste que verifica se a interface está recebendo corretamente os dados da máquina.
- void [test_receiveDataFromInterface_data \(\)](#)
Função que possui os dados necessários para realizar o teste test_receiveDataFromInterface.

Private Member Functions

- void [compareStructIntElements \(int result, int expected\)](#)
Compara a variável result com a expected.
- void [compareStructFloatElements \(float result, float expected\)](#)
Compara a variável result com a expected.

Private Attributes

- [ThreadController * receiveDataFromMachine](#)
A classe ThreadController cria o socket servidor que irá aguardar a conexão do cliente da Máquina.
- [machineServer * receiveDataFromInterface](#)
A classe machineServer criar o socket servidor que irá aguardar a conexão do cliente da Interface.
- [Experiment * info_variables](#)
- [Field * setupFields](#)
- [Table * tables](#)
- [sendCommands test_sendData](#)
- [machineClient * test_sendDataMachine](#)

6.17.1 Detailed Description

Classe de testes para o socket.

Esta classe testa se o envio e recebimento das mensagens através do socket local funciona.

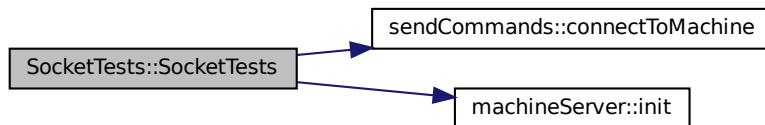
6.17.2 Constructor & Destructor Documentation

6.17.2.1 `SocketTests()`

```
SocketTests::SocketTests ( )
```

Contrutor da classe de testes do socket. Pode ser utilizado para instanciar classes, diferentes da atual, ou para configurações iniciais.

Here is the call graph for this function:



6.17.2.2 `~SocketTests()`

```
SocketTests::~SocketTests ( )
```

Destrutor da classe de testes do socket.

No destrutor é eliminada a instância `receiveDataFromMachine` da [ThreadController](#).

6.17.3 Member Function Documentation

6.17.3.1 `cleanupTestCase`

```
void SocketTests::cleanupTestCase ( ) [private], [slot]
```

Último teste.

Teste que roda ao final de todos os outros testes. Pode ser utilizado para alguma configuração final dos dados. Como liberação de memória ou algum outro pós-processamento.

6.17.3.2 `compareStructFloatElements()`

```
void SocketTests::compareStructFloatElements (
    float result,
    float expected ) [private]
```

Compara a variável `result` com a `expected`.

Esta função realiza uma comparação entre os 2 parâmetros do tipo `float` e apresenta um log caso haja diferenças. Ela foi criada para permitir diversas comparações em apenas 1 teste.

Parameters

<i>result</i>	Resultado do envio de dados através do socket.
<i>expected</i>	Valor esperado que chegue através do socket.

Here is the caller graph for this function:



6.17.3.3 compareStructIntElements()

```
void SocketTests::compareStructIntElements (
    int result,
    int expected ) [private]
```

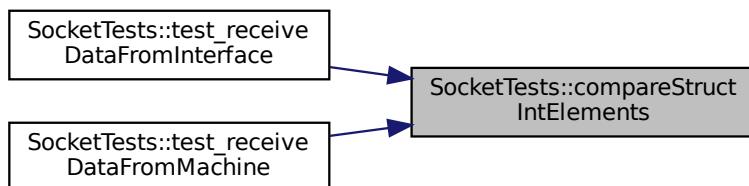
Compara a variável *result* com a *expected*.

Esta função realiza uma comparação entre os 2 parâmetros do tipo int e apresenta um log caso haja diferenças. Ela foi criada para permitir diversas comparações em apenas 1 teste.

Parameters

<i>result</i>	Resultado do envio de dados através do socket.
<i>expected</i>	Valor esperado que chegue através do socket.

Here is the caller graph for this function:



6.17.3.4 initTestCase

```
void SocketTests::initTestCase ( ) [private], [slot]
```

Primeiro teste.

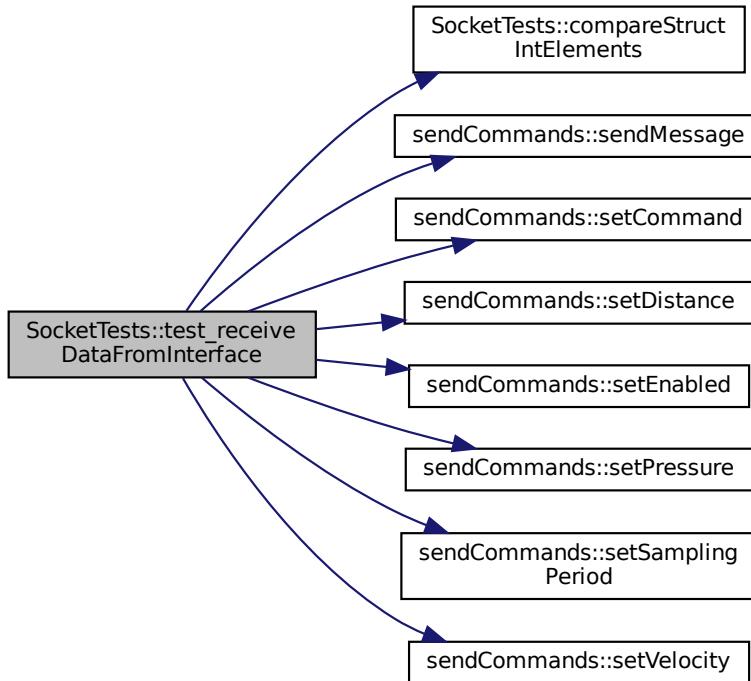
Teste que roda antes de todos os outros testes. Pode ser utilizado para alguma configuração inicial.

6.17.3.5 test_receiveDataFromInterface

```
void SocketTests::test_receiveDataFromInterface ( ) [private], [slot]
```

Teste que verifica se a interface está recebendo corretamente os dados da máquina.

Este teste é uma simulação, não é utilizada a máquina real para este teste. Here is the call graph for this function:



6.17.3.6 test_receiveDataFromInterface_data

```
void SocketTests::test_receiveDataFromInterface_data ( ) [private], [slot]
```

Função que possui os dados necessários para realizar o teste test_receiveDataFromInterface.

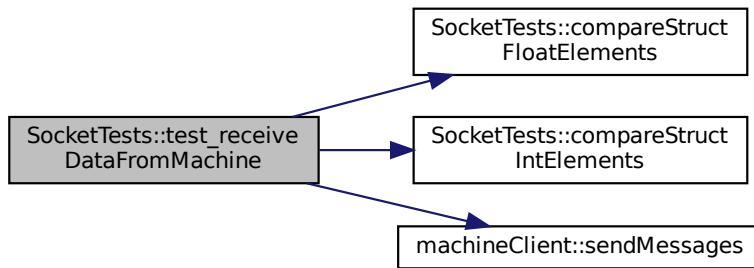
Funções que terminam com _data nos testes, tem a função de fornecerem os dados para o teste ser executado. Exemplo: test_receiveDataFromInterface e test_receiveDataFromInterface_data.

6.17.3.7 test_receiveDataFromMachine

```
void SocketTests::test_receiveDataFromMachine ( ) [private], [slot]
```

Teste que verifica se a interface está recebendo corretamente os dados da máquina.

Este teste é uma simulação, não é utilizada a máquina real para este teste. Here is the call graph for this function:



6.17.3.8 test_receiveDataFromMachine_data

```
void SocketTests::test_receiveDataFromMachine_data ( ) [private], [slot]
```

Função que possui os dados necessários para realizar o teste test_receiveDataFromMachine.

Funções que terminam com _data nos testes, tem a função de fornecerem os dados para o teste ser executado. Exemplo: test_receiveDataFromMachine e test_receiveDataFromMachine_data.

6.17.4 Member Data Documentation

6.17.4.1 info_variables

```
Experiment* SocketTests::info_variables [private]
```

6.17.4.2 receiveDataFromInterface

```
machineServer* SocketTests::receiveDataFromInterface [private]
```

A classe [machineServer](#) criar o socket servidor que irá aguardar a conexão do cliente da Interface.

Instâncie um objeto da classe machine server. Este objeto inicia uma thread, garantindo que o servidor da Máquina e o cliente da Interface rodem em paralelo.

6.17.4.3 receiveDataFromMachine

```
ThreadController* SocketTests::receiveDataFromMachine [private]
```

A classe [ThreadController](#) cria o socket servidor que irá aguardar a conexão do cliente da Máquina.

Instância um objeto da classe [ThreadController](#), responsável por criar um objeto da classe [receiveData](#) (Thread). Fazendo com que a interface e o serviço dos sockets rodem em paralelo.

6.17.4.4 setupFields

```
Field* SocketTests::setupFields [private]
```

6.17.4.5 tables

```
Table* SocketTests::tables [private]
```

6.17.4.6 test_sendData

```
sendCommands SocketTests::test_sendData [private]
```

6.17.4.7 test_sendDataMachine

```
machineClient* SocketTests::test_sendDataMachine [private]
```

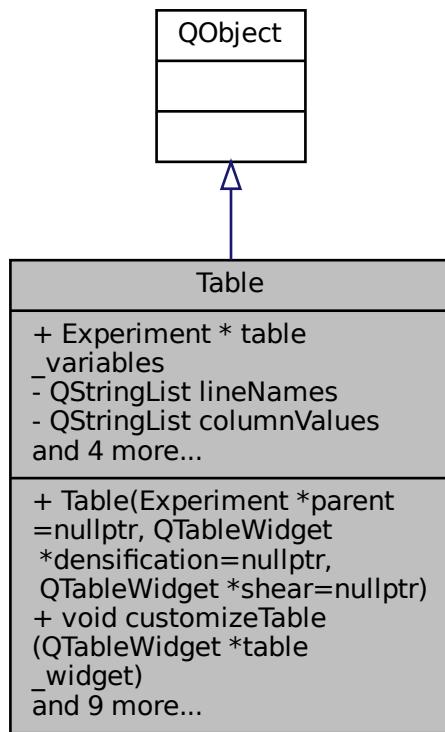
The documentation for this class was generated from the following file:

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/[tst_sockettests.cpp](#)

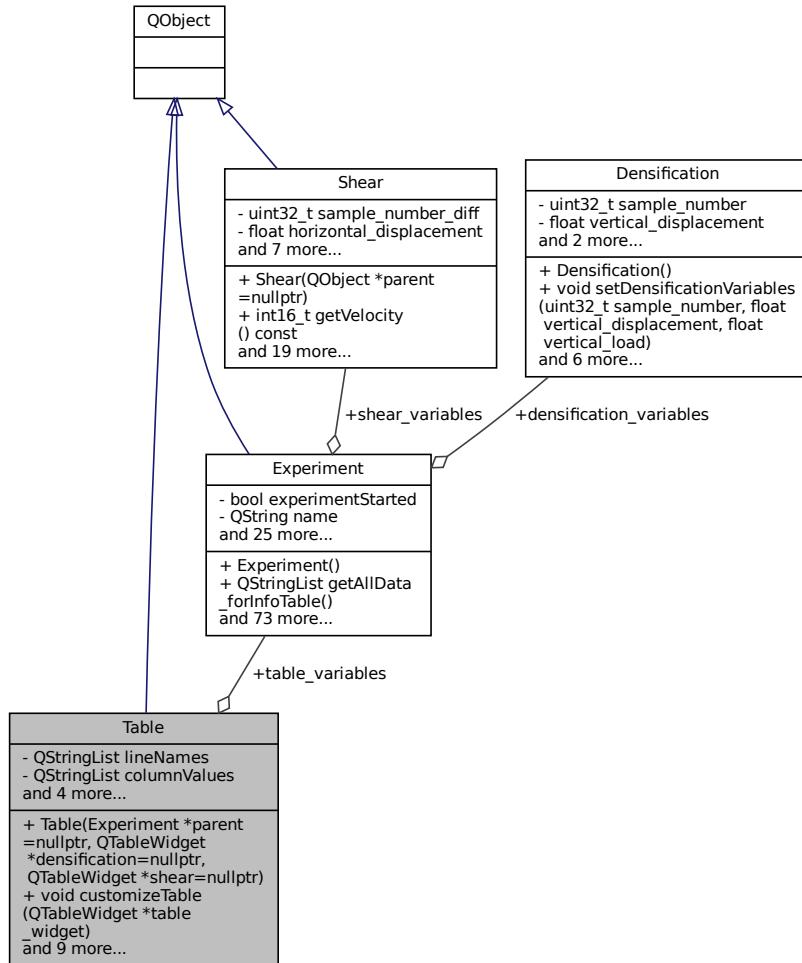
6.18 Table Class Reference

```
#include <table.h>
```

Inheritance diagram for Table:



Collaboration diagram for Table:



Public Member Functions

- `Table (Experiment *parent=nullptr, QTableWidget *densification=nullptr, QTableWidget *shear=nullptr)`
- `void customizeTable (QTableWidget *table_widget)`
- `void initialConfig_ShearTable (QTableWidget *table_widget)`
- `void updateData_ShearTable (QTableWidget *table_widget)`
- `void initialConfig_StaticTable (QTableWidget *table_widget, uint8_t option)`
- `void updateData_StaticTable (QTableWidget *table_widget, uint8_t option)`
- `void updateData_DynamicTable (QStringList data, uint8_t option)`
- `void initialConfig_DynamicTable (QTableWidget *table_widget, uint8_t option)`
- `void clearDynamicTables (QTableWidget *table_widget)`
- `void clearStaticTables (QTableWidget *table_widget)`
- `void test ()`

Public Attributes

- `Experiment * table_variables`

Private Attributes

- QStringList `lineNames` [4]
- QStringList `columnValues`
- QStringList `headerNames` [2]
- QStringList `machineTablevalues` [2]
- QTableWidget * `densificationTable`
- QTableWidget * `shearTable`

6.18.1 Constructor & Destructor Documentation

6.18.1.1 Table()

```
Table::Table (
    Experiment * parent = nullptr,
    QTableWidget * densification = nullptr,
    QTableWidget * shear = nullptr ) [explicit]
```

6.18.2 Member Function Documentation

6.18.2.1 clearDynamicTables()

```
void Table::clearDynamicTables (
    QTableWidget * table_widget )
```

Here is the caller graph for this function:



6.18.2.2 clearStaticTables()

```
void Table::clearStaticTables (
    QTableWidget * table_widget )
```

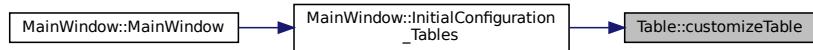
Here is the caller graph for this function:



6.18.2.3 customizeTable()

```
void Table::customizeTable (
    QTableWidget * table_widget )
```

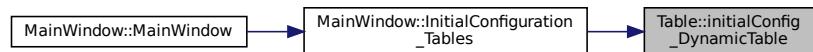
Here is the caller graph for this function:



6.18.2.4 initialConfig_DynamicTable()

```
void Table::initialConfig_DynamicTable (
    QTableWidget * table_widget,
    uint8_t option )
```

Here is the caller graph for this function:



6.18.2.5 initialConfig_ShearTable()

```
void Table::initialConfig_ShearTable (
    QTableWidget * table_widget )
```

6.18.2.6 initialConfig_StaticTable()

```
void Table::initialConfig_StaticTable (
    QTableWidget * table_widget,
    uint8_t option )
```

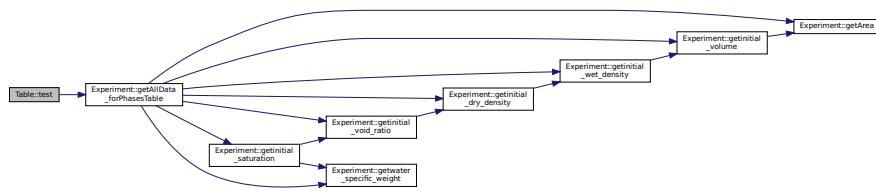
Here is the caller graph for this function:



6.18.2.7 test()

```
void Table::test ( )
```

Here is the call graph for this function:



6.18.2.8 updateData_DynamicTable()

```
void Table::updateData_DynamicTable (
    QStringList data,
    uint8_t option )
```

Here is the caller graph for this function:



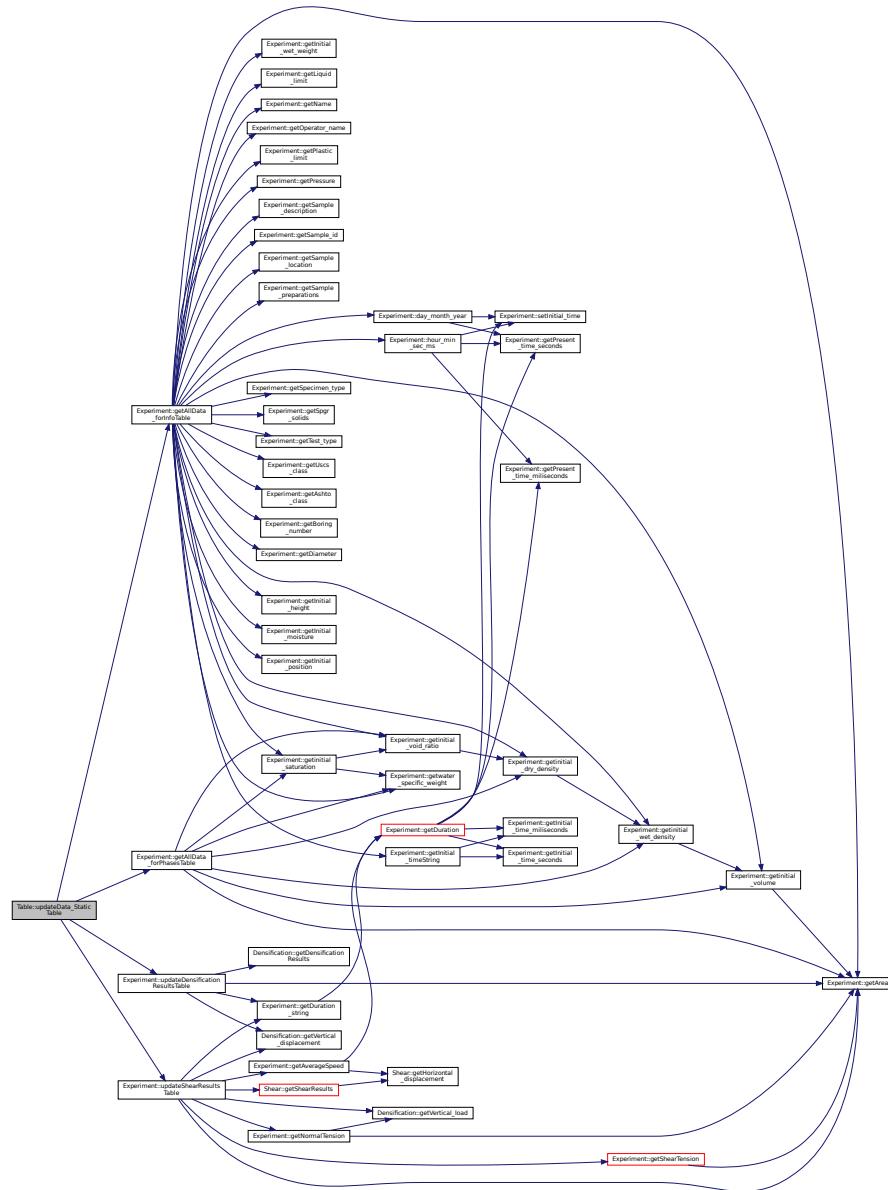
6.18.2.9 updateData_ShearTable()

```
void Table::updateData_ShearTable (
    QTableWidget * table_widget )
```

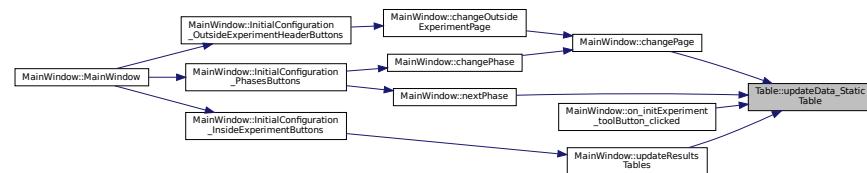
6.18.2.10 updateData_StaticTable()

```
void Table::updateData_StaticTable (
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.18.3 Member Data Documentation

6.18.3.1 columnValues

```
QStringList Table::columnValues [private]
```

6.18.3.2 densificationTable

```
QTableWidget* Table::densificationTable [private]
```

6.18.3.3 headerNames

```
QStringList Table::headerNames[2] [private]
```

6.18.3.4 lineNames

```
QStringList Table::lineNames[4] [private]
```

6.18.3.5 machineTablevalues

```
QStringList Table::machineTablevalues[2] [private]
```

6.18.3.6 shearTable

```
QTableWidget* Table::shearTable [private]
```

6.18.3.7 table_variables

```
Experiment* Table::table_variables
```

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

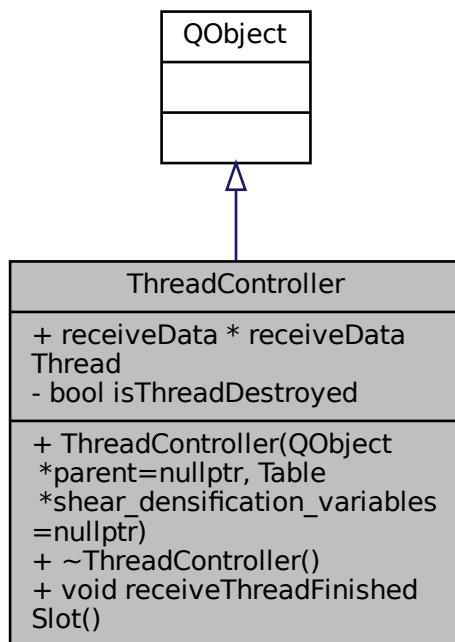
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[table.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[table.cpp](#)

6.19 ThreadController Class Reference

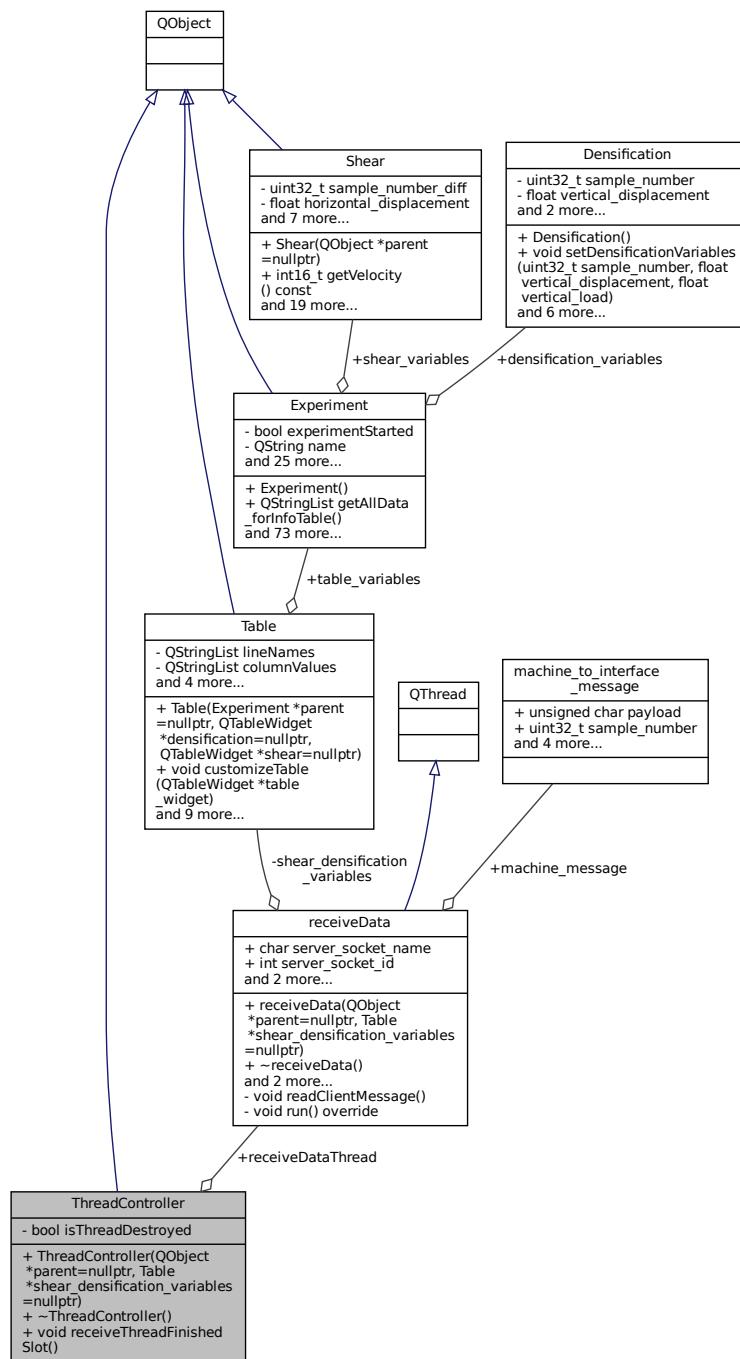
Classe que controla a thread criada por [receiveData](#).

```
#include <threadcontroller.h>
```

Inheritance diagram for ThreadController:



Collaboration diagram for ThreadController:



Public Slots

- void `receiveThreadFinishedSlot ()`

Esse slot é disparado quando a thread da classe `receiveData` termina antes da classe `ThreadController`.

Public Member Functions

- `ThreadController (QObject *parent=nullptr, Table *shear_densification_variables=nullptr)`
Construtor da classe Thread Controller.
- `~ThreadController ()`
Destruitor da classe ThreadController.

Public Attributes

- `receiveData * receiveDataThread`

Private Attributes

- `bool isThreadDestroyed`

6.19.1 Detailed Description

Classe que controla a thread criada por `receiveData`.

Esta classe controla a criação e destruição da Thread `receiveData`.

6.19.2 Constructor & Destructor Documentation

6.19.2.1 ThreadController()

```
ThreadController::ThreadController (
    QObject * parent = nullptr,
    Table * shear_densification_variables = nullptr )
```

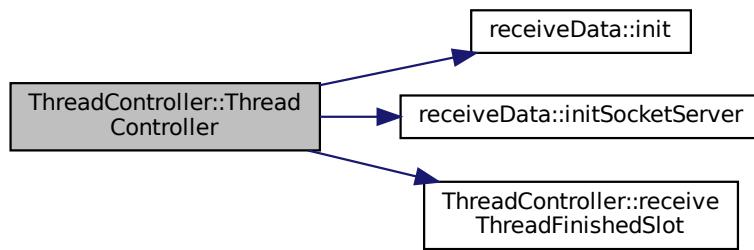
Construtor da classe Thread Controller.

Responsável por instância e iniciar a thread da classe `receiveData`.

Parameters

<code>parent</code>	<input type="text"/>
---------------------	----------------------

Here is the call graph for this function:



6.19.2.2 ~ThreadController()

```
ThreadController::~ThreadController ( )
```

Destrutor da classe [ThreadController](#).

Responsável por encerrar corretamente a thread da classe [receiveData](#).

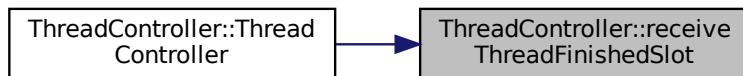
6.19.3 Member Function Documentation

6.19.3.1 receiveThreadFinishedSlot

```
void ThreadController::receiveThreadFinishedSlot ( ) [slot]
```

Esse slot é disparado quando a thread da classe [receiveData](#) termina antes da classe [ThreadController](#).

Quando isso ocorre, é deletado a instância da classe [receiveData](#). Here is the caller graph for this function:



6.19.4 Member Data Documentation

6.19.4.1 isThreadDestroyed

```
bool ThreadController::isThreadDestroyed [private]
```

Informa se a thread foi destruída ou não.

6.19.4.2 receiveDataThread

```
receiveData* ThreadController::receiveDataThread
```

Instância da classe [receiveData\(Thread\)](#).

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

- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/[threadcontroller.h](#)
- /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/[threadcontroller.cpp](#)

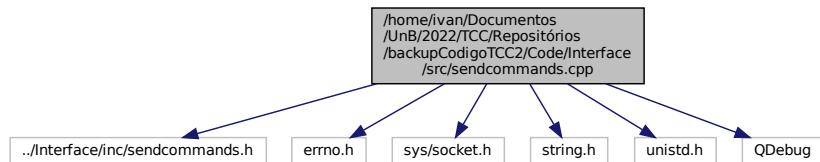
Chapter 7

File Documentation

7.1 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/src/sendcommands.cpp File Reference

```
#include "../Interface/inc/sendcommands.h"
#include <errno.h>
#include <sys/socket.h>
#include <string.h>
#include <unistd.h>
#include <QDebug>
```

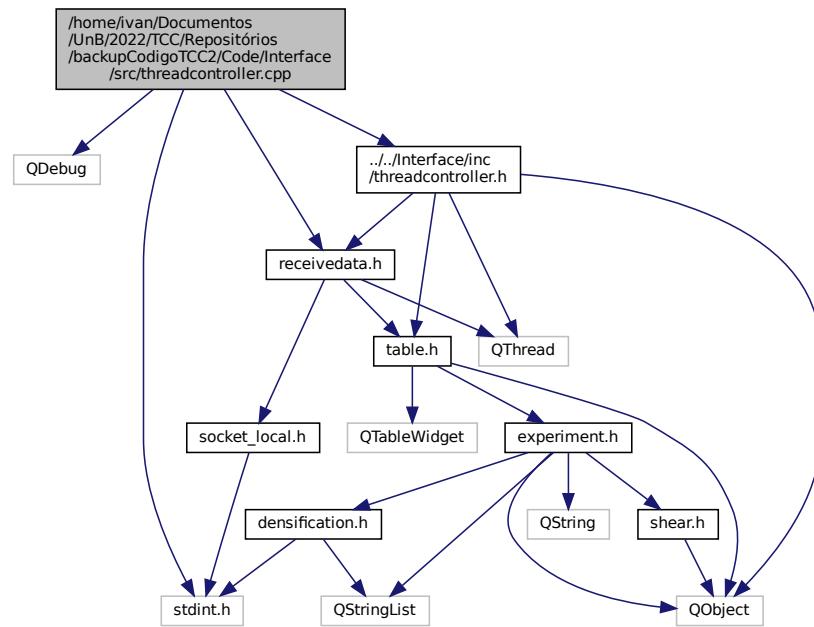
Include dependency graph for sendcommands.cpp:



7.2 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/src/threadcontroller.cpp File Reference

```
#include <QDebug>
#include <stdint.h>
#include "../..../Interface/inc/threadcontroller.h"
```

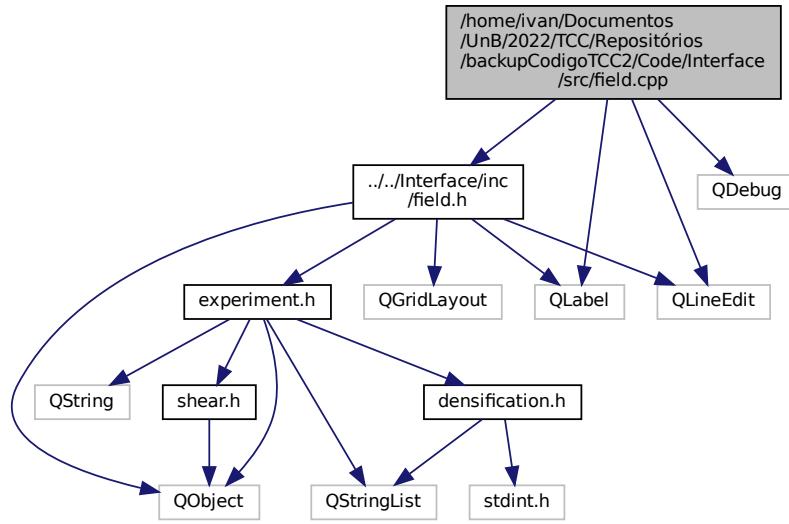
```
#include "../../Interface/inc/receivedata.h"
Include dependency graph for threadcontroller.cpp:
```



7.3 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/src/field.cpp File Reference

```
#include "../../Interface/inc/field.h"
#include <QLineEdit>
#include <QLabel>
#include <QDebug>
```

Include dependency graph for field.cpp:



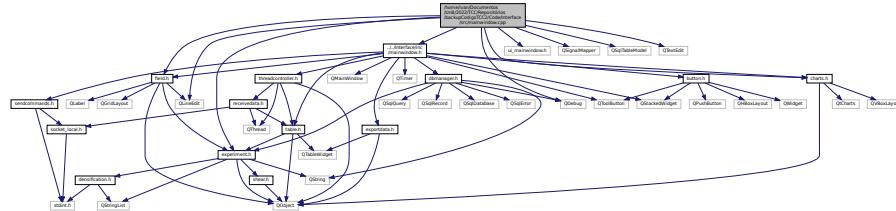
7.4 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/src/mainwindow.cpp File Reference

```

#include "../../../Interface/inc/mainwindow.h"
#include "ui_mainwindow.h"
#include "inc/field.h"
#include <QDebug>
#include <QLineEdit>
#include <QSignalMapper>
#include "inc/experiment.h"
#include <QSqlTableModel>
#include <QTextEdit>
#include "inc/charts.h"

```

Include dependency graph for mainwindow.cpp:



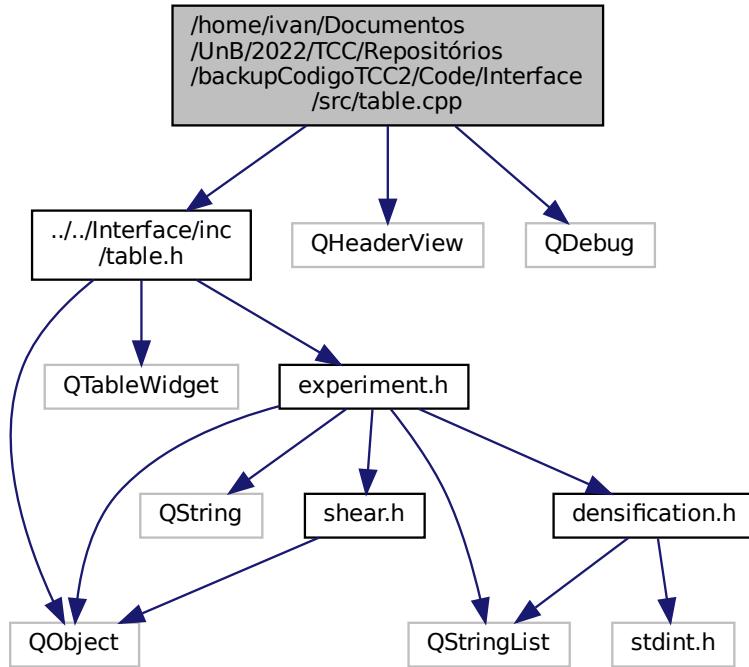
7.5 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/src/table.cpp File Reference

```

#include "../../../Interface/inc/table.h"
#include <QHeaderView>

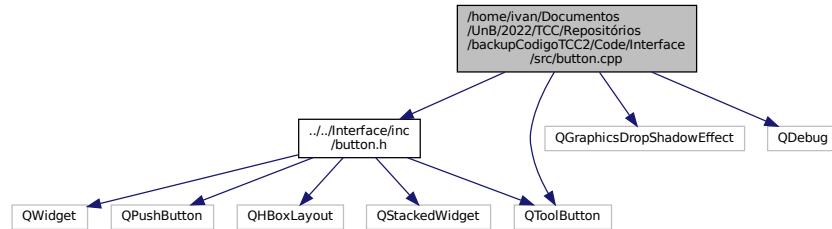
```

```
#include <QDebug>
Include dependency graph for table.cpp:
```



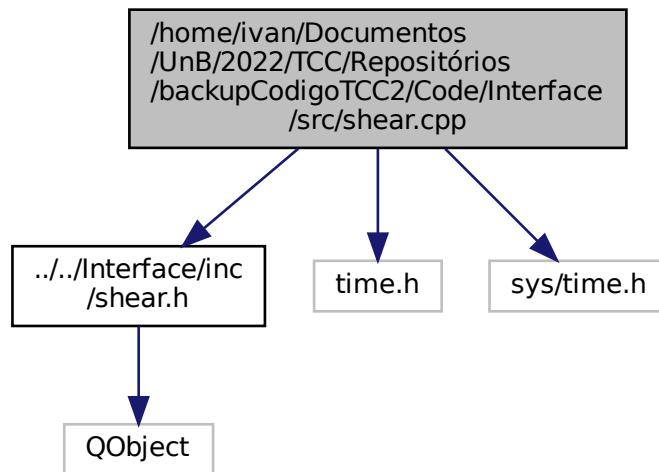
7.6 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/src/button.cpp File Reference

```
#include "../../Interface/inc/button.h"
#include <QGraphicsDropShadowEffect>
#include <QDebug>
#include <QToolButton>
Include dependency graph for button.cpp:
```



7.7 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/shear.cpp File Reference

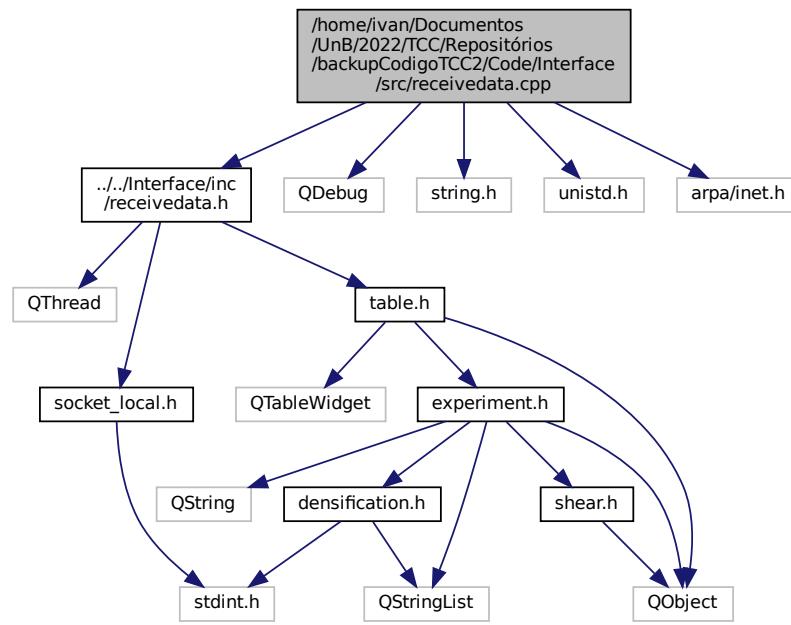
```
#include "../../Interface/inc/shear.h"
#include <time.h>
#include <sys/time.h>
Include dependency graph for shear.cpp:
```



7.8 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/receivedata.cpp File Reference

```
#include "../../Interface/inc/receivedata.h"
#include <QDebug>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
```

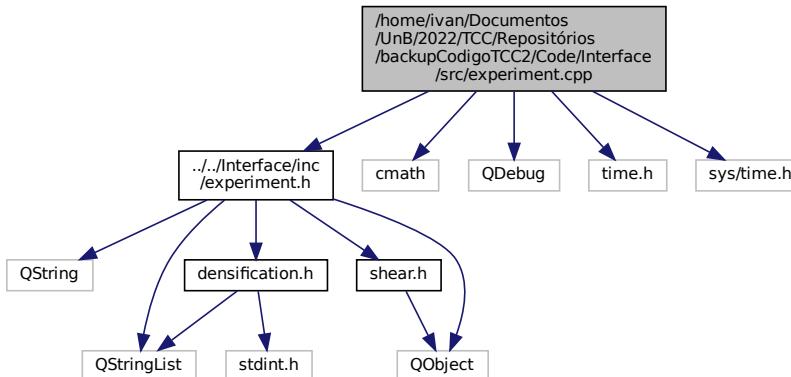
Include dependency graph for receivedata.cpp:



7.9 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/experiment.cpp File Reference

```
#include "../../Interface/inc/experiment.h"
#include <cmath>
#include <QDebug>
#include <time.h>
#include <sys/time.h>
```

Include dependency graph for experiment.cpp:



7.10 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup

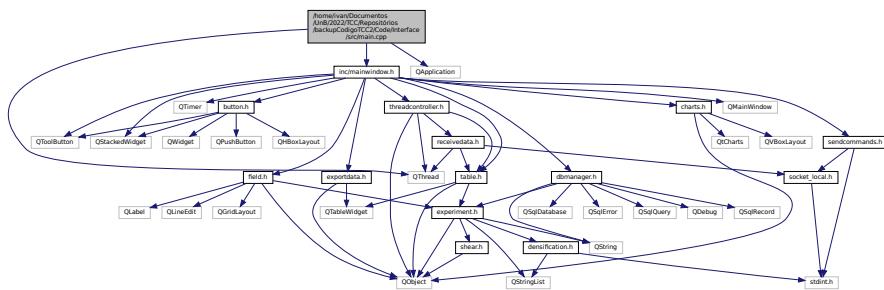
CodigoTCC2/Code/Interface/src/main.cpp File Reference

```
#include "inc/mainwindow.h"
```

```
#include <QApplication>
```

```
#include <QThread>
```

Include dependency graph for main.cpp:



Functions

- int `main` (int argc, char *argv[])

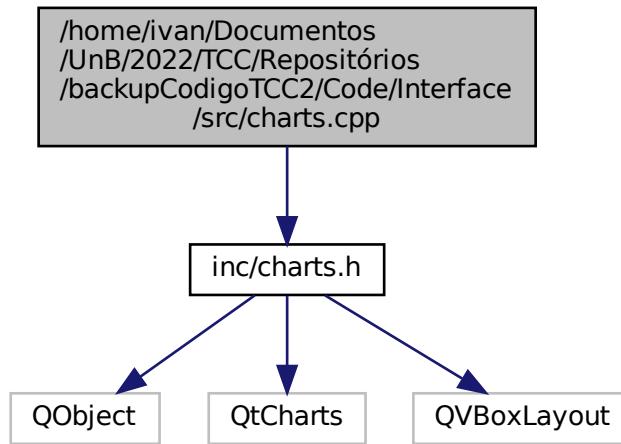
7.10.1 Function Documentation

7.10.1.1 main()

```
int main (
    int argc,
    char * argv[ ] )
```

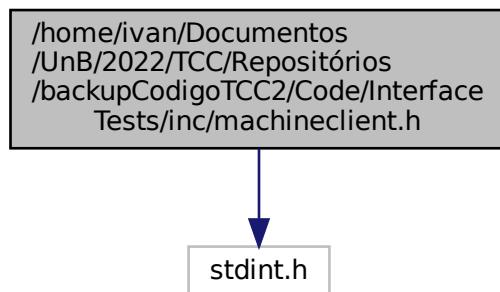
7.11 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/src/charts.cpp File Reference

```
#include "inc/charts.h"
Include dependency graph for charts.cpp:
```

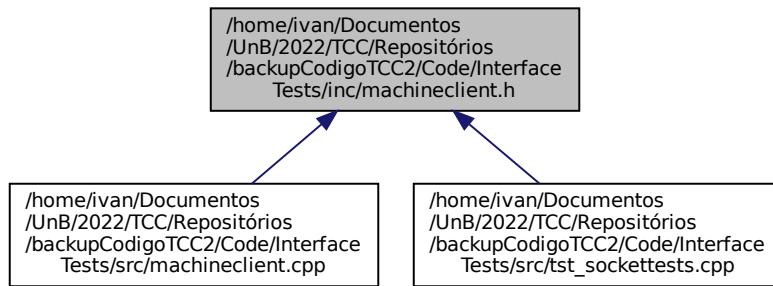


7.12 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/InterfaceTests/inc/machineclient.h File Reference

```
#include <stdint.h>
Include dependency graph for machineclient.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [machineClient](#)
Classe que simula o envio de dados da máquina real.

7.13 machineclient.h

[Go to the documentation of this file.](#)

```

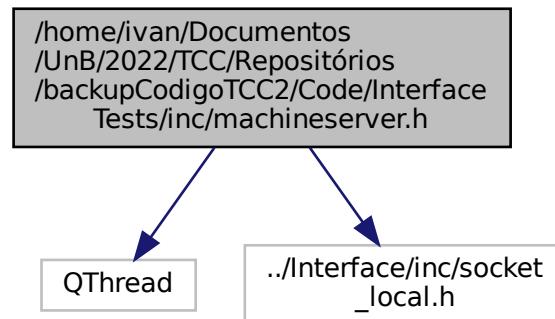
1 #ifndef MACHINECLIENT_H
2 #define MACHINECLIENT_H
3 #include<stdint.h>
10 class machineClient
11 {
12     public:
13         machineClient();
14         ~machineClient();
15         void sendMessages(int32_t i);
16     private:
22         char socket_name[200];
28         int socket_id;
35         uint8_t errorOccurred;
36
37
38
39 };
40
41 #endif // MACHINECLIENT_H

```

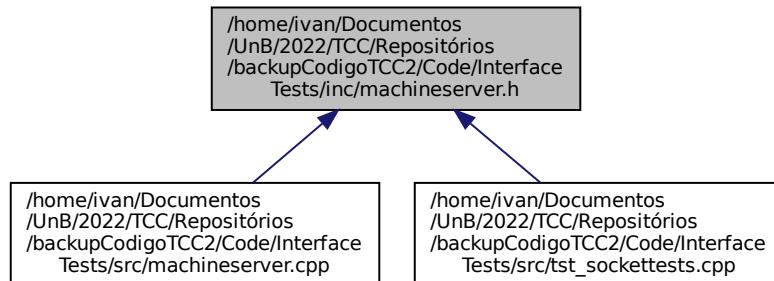
7.14 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/InterfaceTests/inc/machineserver.h File Reference

```
#include <QThread>
#include "../Interface/inc/socket_local.h"
```

Include dependency graph for machineserver.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [machineServer](#)
Classe que simula o servidor da máquina real.

7.15 machineserver.h

[Go to the documentation of this file.](#)

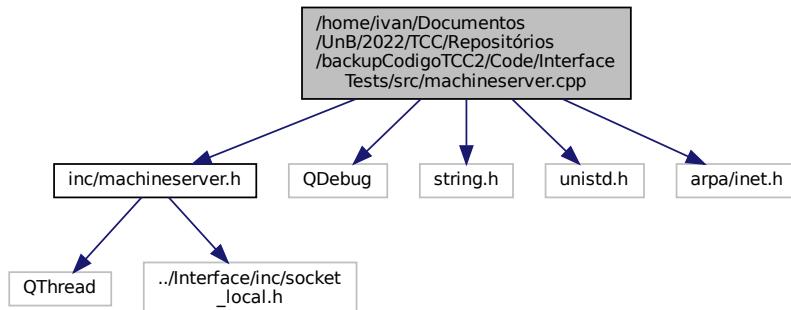
```

1 #ifndef MACHINESERVER_H
2 #define MACHINESERVER_H
3
4 #include <QThread>
5 #include "../Interface/inc/socket_local.h"
6
13 class machineServer : public QThread
14 {
15     Q_OBJECT
16     public:
  
```

```
17     machineServer(QObject * parent = nullptr);
18     ~machineServer();
19     void init();
20     char server_socket_name[100];
21     int server_socket_id;
22     int client_socket_id;
23     interface_to_machine_message interface_message;
24     uint8_t errorOccurred;
25 private:
26     void run() override;
27     void read_interface_message();
28 };
29 }
30 #endif // MACHINESEVER_H
```

7.16 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/machineserver.cpp File Reference

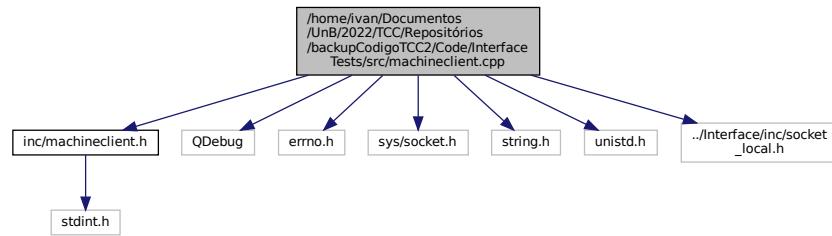
```
#include "inc/machineserver.h"
#include <QDebug>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
Include dependency graph for machineserver.cpp:
```



7.17 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/InterfaceTests/src/machineclient.cpp File Reference

```
#include "inc/machineclient.h"
#include <QDebug>
#include <errno.h>
#include <sys/socket.h>
#include <string.h>
#include <unistd.h>
```

```
#include "../Interface/inc/socket_local.h"
Include dependency graph for machineclient.cpp:
```



7.18 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup←
CódigoTCC2/Code/InterfaceTests/src/tst_sockettests.cpp File
Reference

```
#include <QtTest>
#include <QCoreApplication>
#include <QDebug>
#include "inc/machineclient.h"
#include "../Interface/inc/threadcontroller.h"
#include "../Interface/inc/socket_local.h"
#include "../Interface/inc/sendcommands.h"
#include "../Interface/inc/experiment.h"
#include "../Interface/inc/dbmanager.h"
#include "../Interface/inc/densification.h"
#include "../Interface/inc/shear.h"
#include "../Interface/inc/field.h"
#include "../Interface/inc/table.h"
#include "inc/machineserver.h"
#include <unistd.h>
#include <QTableWidget>
#include "tst_sockettests.moc"
Include dependency graph for tst_sockettests.cpp:
```

Include dependency graph for `tst_sockettests.cpp`:



Classes

- class `SocketTests`

Classe de testes para o socket.

Macros

- `#define tests`

7.18.1 Macro Definition Documentation

7.18.1.1 tests

```
#define tests
```

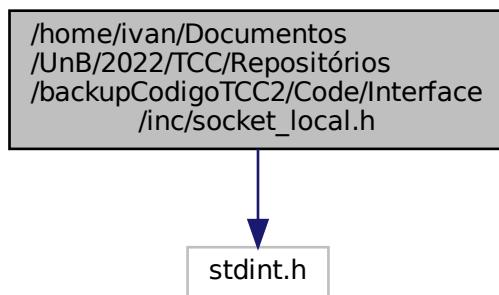
7.19 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup CodigoTCC2/Code/InterfaceTests/src/testDoxygen.cpp File Reference

Classes

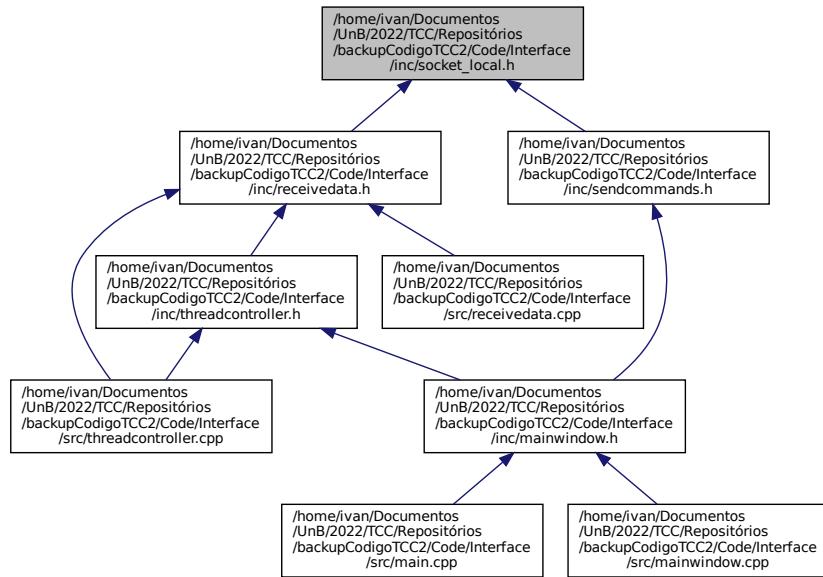
- class [QTstyle_Test](#)
A test class.

7.20 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup CodigoTCC2/Code/Interface/inc/socket_local.h File Reference

```
#include <stdint.h>  
Include dependency graph for socket_local.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- union `interface_to_machine_message`

Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina.

- union `machine_to_interface_message`

Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface.

Macros

- #define `interface_payload_size` 3 * sizeof(uint16_t)
- #define `machine_payload_size` 6 * sizeof(uint32_t)

TypeDefs

- typedef union `interface_to_machine_message` `interface_to_machine_message`

Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina.

- typedef union `machine_to_interface_message` `machine_to_interface_message`

Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface.

7.20.1 Macro Definition Documentation

7.20.1.1 interface_payload_size

```
#define interface_payload_size 3 * sizeof(uint16_t)
```

Tamanho do payload da mensagem enviada pela interface.

7.20.1.2 machine_payload_size

```
#define machine_payload_size 6 * sizeof(uint32_t)
```

Tamanho do payload da mensagem enviada pela máquina.

7.20.2 Typedef Documentation

7.20.2.1 interface_to_machine_message

```
typedef union interface_to_machine_message interface_to_machine_message
```

Essa union tem como objetivo facilitar o envio de mensagens da interface para a máquina.

Utilizando a union, a transformação dos dados para string (unsigned char [] neste caso) é facilitada, pois os dados ocupam a mesma região de memória que a string. Assim para enviar os dados pelo socket basta enviar a variável payload.

7.20.2.2 machine_to_interface_message

```
typedef union machine_to_interface_message machine_to_interface_message
```

Essa union tem como objetivo facilitar o envio de mensagem da máquina para a interface.

Utilizando a union, a transformação dos dados para string (unsigned char [] neste caso) é facilitada, pois os dados ocupam a mesma região de memória que a string. Assim para enviar os dados pelo socket basta enviar a variável payload.

7.21 socket_local.h

[Go to the documentation of this file.](#)

```

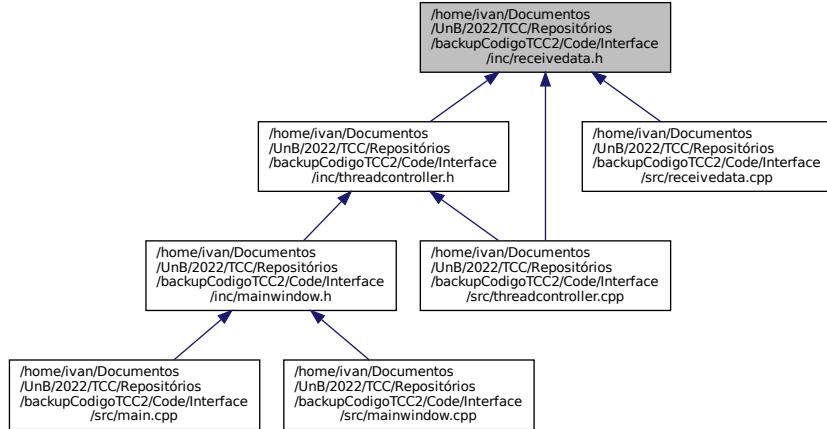
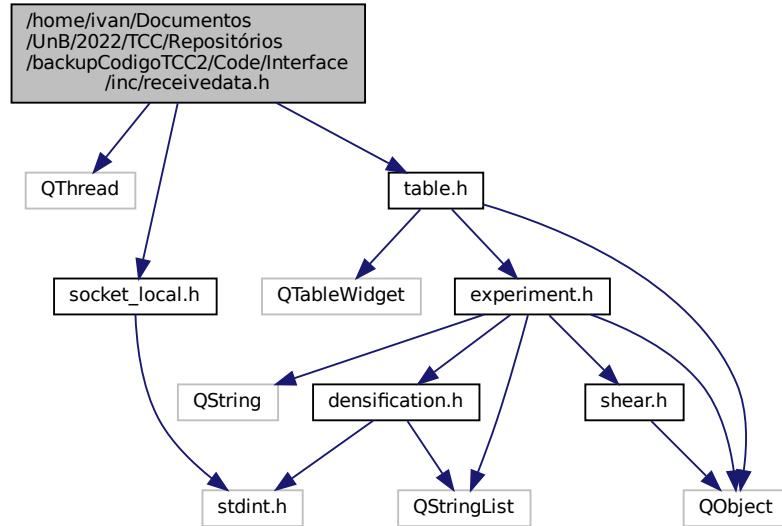
1 #ifndef SOCKET_LOCAL_H
2 #define SOCKET_LOCAL_H
3
4 #include<stdint.h>
5
6 //##define tests
7
8 #ifdef tests
9 #include <QMetaType>
10#endif
11
12#define interface_payload_size 3 * sizeof(uint16_t)
13#define machine_payload_size 6 * sizeof(uint32_t)
14
15typedef union interface_to_machine_message{
16    unsigned char payload[interface_payload_size];
17    struct {
18        uint8_t command;
19        union {
20            uint16_t pressure;
21            struct {
22                int16_t distance;
23                int16_t velocity;
24            };
25            struct {
26                uint8_t enabled;
27                uint16_t sample_period;
28            };
29        };
30    };
31} interface_to_machine_message;
32
33typedef union machine_to_interface_message{
34    unsigned char payload[machine_payload_size];
35    struct {
36        uint32_t sample_number; /* Número da amostra, [0] para as amostras de adensamento e [1] para as
37        de cisalhamento.*/
38        /*uint32_t date_time;/*!< Dia, hora, minuto e segundo que o essa union foi enviada para a
39        interface.*/
40        float displacement[2];
41        float load[2];
42        uint8_t state;
43    };
44} machine_to_interface_message;
45
46/*
47 1 tabela de experimento
48
49 n amostras
50
51 1 cisalhamento e de adensamento
52 id do experimento
53 fase
54 sample_number
55 displacement[2]
56 load[2]
57 sample_number_diff
58
59 */
60
61
62
63
64
65 1 amostras
66
67
68
69 1 cisalhamento e de adensamento
70 id do experimento
71 fase
72 sample_number
73 displacement[2]
74 load[2]
75 sample_number_diff
76
77 */
78
79
80
81#endif
82 Q_DECLARE_METATYPE(interface_to_machine_message);
83 Q_DECLARE_METATYPE(machine_to_interface_message);
84#endif
85
86
87
88// mandar só um numero de amostra e quando iniciar a fase de cisalhamento, subtrair o número de
89// amostra atual pelo último numero de
90// amostra antes de iniciar o cisalhamento. Exemplo: 802 - 800 = numero de amostra 2 para cisalhamento
91// número 802 para adensamento.
92
93
94
95
96
97
98
99#endif // SOCKET_LOCAL_H

```

CodigoTCC2/Code/Interface/inc/receivedata.h File Reference

```
#include <QThread>
#include "socket_local.h"
#include "table.h"
```

Include dependency graph for receivedata.h:



Classes

- class [receiveData](#)

Classe que recebe os dados da camada de controle.

7.23 receivedata.h

Go to the documentation of this file.

```

1 #ifndef RECEIVEDATA_H
2 #define RECEIVEDATA_H
3
4 #include <QThread>
5
6 #include "socket_local.h"
7 #include "table.h"
8
17 class receiveData : public QThread
18 {
19     Q_OBJECT
20 private:
21     void readClientMessage();
22     void run() override;
23     Table * shear_densification_variables;
24 public:
25     char server_socket_name[100];
26     int server_socket_id;
27     int client_socket_id;
28     uint8_t errorOccurred;
29     receiveData(QObject * parent = nullptr, Table * shear_densification_variables = nullptr);
30     ~receiveData();
31     uint8_t initSocketServer();
32     void init();
33     machine_to_interface_message machine_message;
34     public slots:
35     signals:
36         void data_arrived();
37 };
38
39 #endif // RECEIVEDATA_H

```

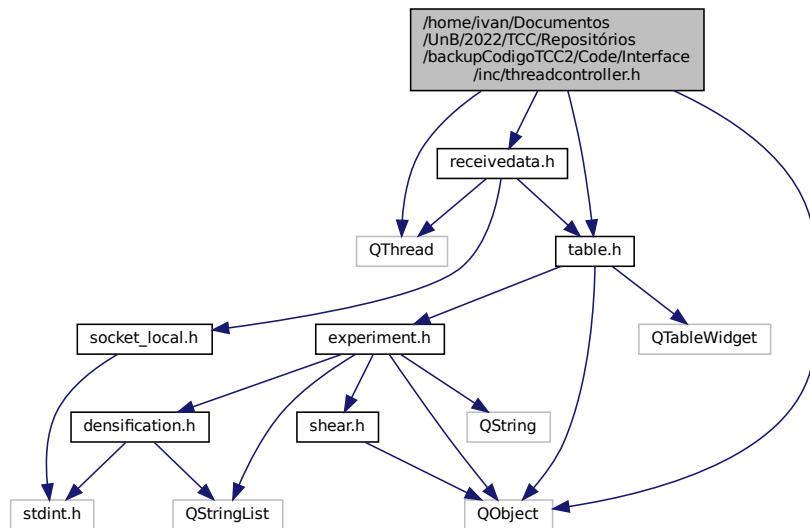
7.24 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/threadcontroller.h File Reference

```

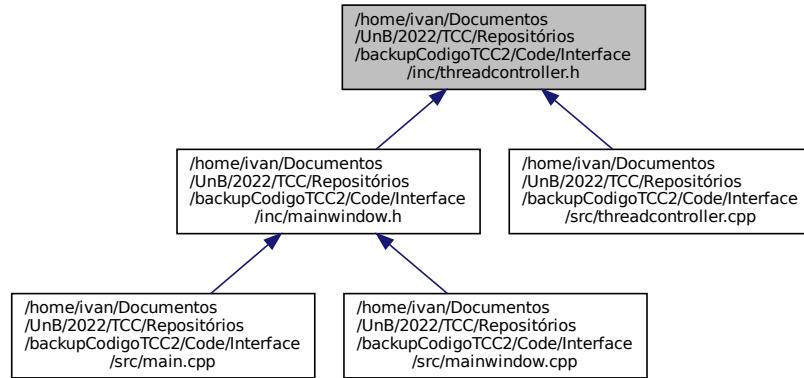
#include <QObject>
#include <QThread>
#include "receivedata.h"
#include "table.h"

```

Include dependency graph for threadcontroller.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [ThreadController](#)

Classe que controla a thread criada por `receiveData`.

7.25 threadcontroller.h

[Go to the documentation of this file.](#)

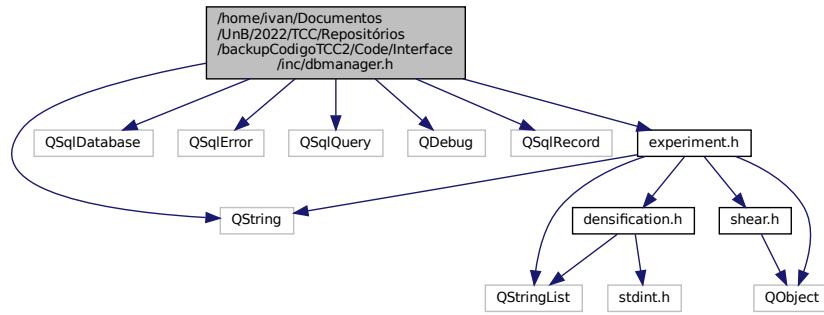
```

1 ifndef THREADCONTROLLER_H
2 define THREADCONTROLLER_H
3
4 include <QObject>
5 include <QThread>
6
7 include "receivedata.h"
8 include "table.h"
9
10
17 class ThreadController : public QObject
18 {
19     Q_OBJECT
20     private:
21         bool isThreadDestroyed;
22     public:
23         receiveData *receiveDataThread;
24         ThreadController(QObject *parent = nullptr, Table * shear_densification_variables = nullptr);
25         ~ThreadController();
26     public slots:
27         void receiveThreadFinishedSlot();
28     signals:
29 };
30
31 endif // THREADCONTROLLER_H
  
```

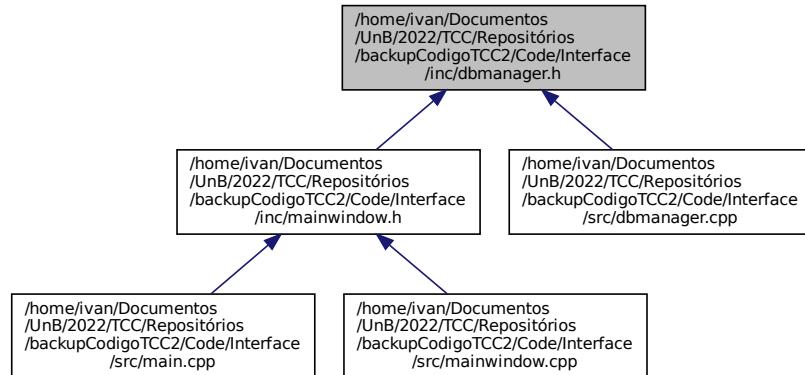
7.26 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/inc/dbmanager.h File Reference

```
#include <QString>
#include <QSqlDatabase>
```

```
#include <QSqlError>
#include <QSqlQuery>
#include <QDebug>
#include <QSqlRecord>
#include "experiment.h"
Include dependency graph for dbmanager.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [DBManager](#)

Classe do banco de dados.

Macros

- `#define experiment_table 0`
- `#define densification_table 1`
- `#define shear_table 2`
- `#define final_variables_table 3`
- `#define sample_table 4`

7.26.1 Macro Definition Documentation

7.26.1.1 densification_table

```
#define densification_table 1
```

Número identificador da tabela DENSIFICATION_TABLE.

7.26.1.2 experiment_table

```
#define experiment_table 0
```

Número identificador da tabela EXPERIMENT_TABLE.

7.26.1.3 final_variables_table

```
#define final_variables_table 3
```

Número identificador da tabela FINAL_VARIABLES_TABLE.

7.26.1.4 sample_table

```
#define sample_table 4
```

Número identificador da tabela SAMPLE_TABLE.

7.26.1.5 shear_table

```
#define shear_table 2
```

Número identificador da tabela SHEAR_TABLE.

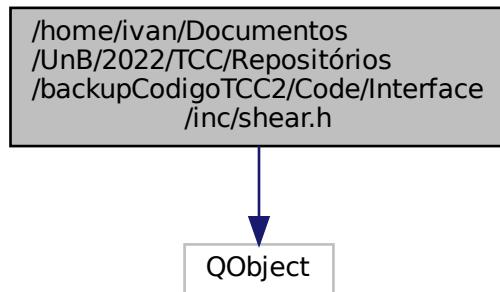
7.27 dbmanager.h

[Go to the documentation of this file.](#)

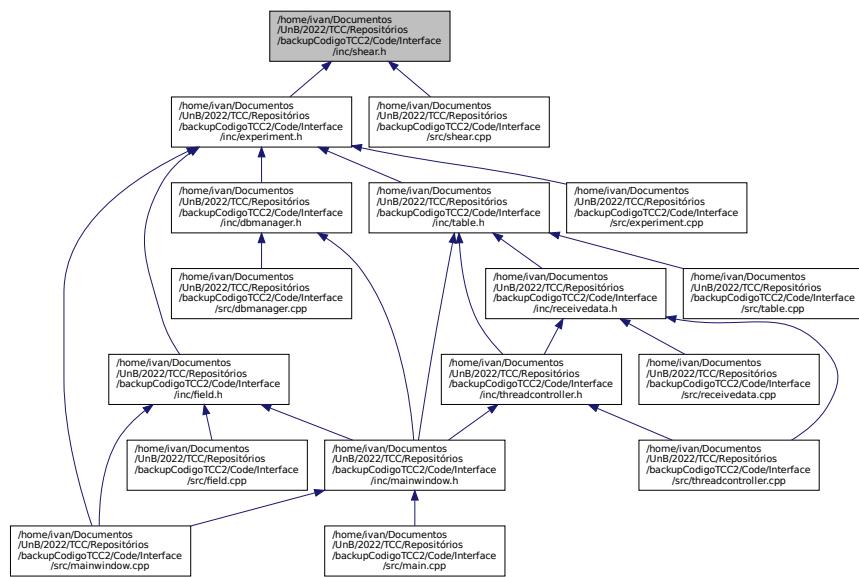
```
1 #ifndef DBMANAGER_H
2 #define DBMANAGER_H
3
4 #include <QString>
5 #include <QSqlDatabase>
6 #include <QSqlError>
7 #include <QSqlQuery>
8 #include <QDebug>
9 #include <QSqlRecord>
10 #include "experiment.h"
11
12 #define experiment_table 0
13 #define densification_table 1
14 #define shear_table 2
15 #define final_variables_table 3
16 #define sample_table 4
23 class DBManager : public QObject
24 {
25     Q_OBJECT
26 public:
27     DBManager(const QString & path, Experiment * temp_experiment_data);
28     ~DBManager();
29     bool isOpen() const;
30     bool createTable(uint8_t option);
31     bool tableExists(uint8_t option);
32     bool insertIntoTable(uint8_t option);
33     bool updateTable(uint8_t option);
34
35     Experiment * experiment_data;
38     uint32_t getExperiment_id() const;
39
40 private:
41     uint32_t experiment_id;
43     QSqlDatabase prova_conceito_database;
44     QString table_name[5] =
45 {"EXPERIMENT_TABLE","DENSIFICATION_TABLE","SHEAR_TABLE","FINAL_VARIABLES_TABLE","SAMPLE_TABLE"};
46     void insertValuesIntoBind_Experiment(QSqlQuery *query);
47     void insertValuesIntoBind_Densification(QSqlQuery *query);
48     void insertValuesIntoBind_Shear(QSqlQuery *query);
49     void insertValuesIntoBind_FinalVariables(QSqlQuery *query);
50     void insertValuesIntoBind_SampleVariables(QSqlQuery *query);
51     void insertValuesIntoBind_SampleVariablesUpdate(QSqlQuery *query);
52
53     QString create_table[5];
54     QString insert_into_table[6];
57     bool selectExperimentId();
58
59
60 public slots:
61     void update_database_table();
62
63 };
64 #endif // DBMANAGER_H
```

7.28 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/shear.h File Reference

```
#include <QObject>
Include dependency graph for shear.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class Shear

7.29 shear.h

[Go to the documentation of this file.](#)

```

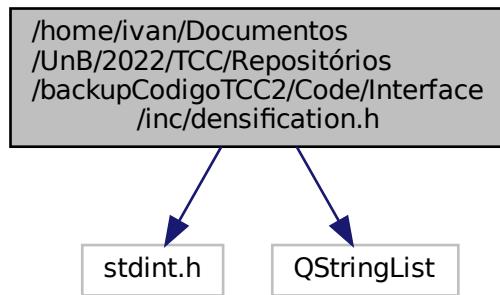
1 #ifndef SHEAR_H
2 #define SHEAR_H
3
4 #include <QObject>
5
6 class Shear : public QObject
7 {
8     Q_OBJECT
9 public:
10     explicit Shear(QObject *parent = nullptr);
11
12     int16_t getVelocity() const;
13     void setVelocity(int16_t newVelocity);
14
15     int16_t getDistance() const;
16     void setDistance(int16_t newDistance);
17
18     void setShearVariables(float horizontal_displacement, float horizontal_load);
19     QStringList getShearVariables();
20     QStringList getShearResults();
21
22     uint32_t getSample_number_diff() const;
23     void setSample_number_diff(uint32_t newSample_number_diff);
24
25     float getHorizontal_displacement() const;
26     void setHorizontal_displacement(float newHorizontal_displacement);
27
28     float getHorizontal_load() const;
29     void setHorizontal_load(float newHorizontal_load);
30
31     const QString &getDate() const;
32     void setDate(const QString &newDate);
33
34     const QString &getHour_min_sec_ms() const;
35     void setHour_min_sec_ms(const QString &newHour_min_sec_ms);
36
37     uint64_t getInitial_time_seconds() const;
38
39     uint64_t getInitial_time_miliseconds() const;
40
41
42     void setInitial_time();
43
44 private:
45     uint32_t sample_number_diff;
46     float horizontal_displacement;
47     float horizontal_load;
48
49     int16_t velocity;
50     int16_t distance;
51
52     QString date;
53     QString hour_min_sec_ms;
54
55     uint64_t initial_time_seconds;
56     uint64_t initial_time_miliseconds;
57
58 //     vertical_displacement // recebido
59 //     horizontal_displacement // recebido
60 //     shear_load // recebido
61 //     normal_tension // normal_load/area
62 //     shear_tension // shear_load/area
63
64 signals:
65
66 };
67
68 };
69
70 #endif // SHEAR_H

```

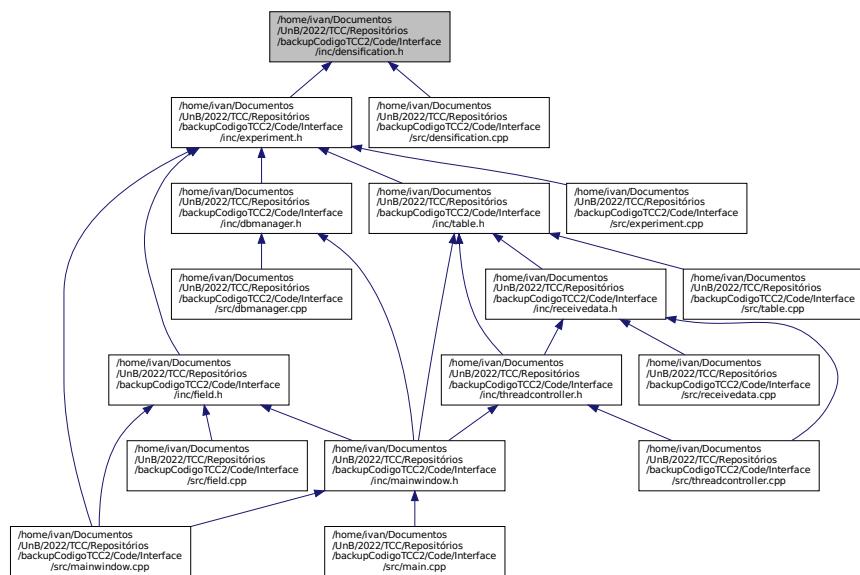
7.30 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/densification.h File Reference

```
#include <stdint.h>
#include <QStringList>
```

Include dependency graph for densification.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Densification](#)

Classe do adensamento.

7.31 densification.h

[Go to the documentation of this file.](#)

```

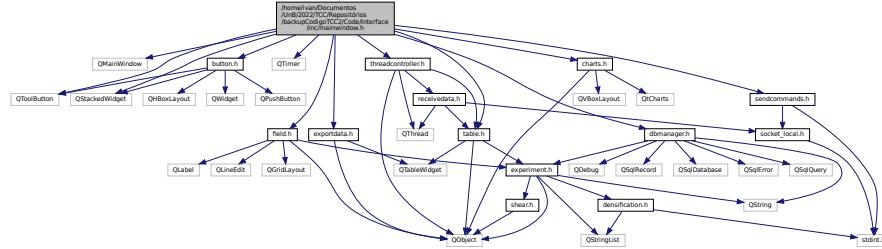
1 #ifndef DENSIFICATION_H
2 #define DENSIFICATION_H
  
```

```
3
4 #include <stdint.h>
5 #include <QStringList>
12 class Densification
13 {
14 public:
15     Densification();
16     void setDensificationVariables(uint32_t sample_number, float vertical_displacement,
17                                     float vertical_load);
18
19     QStringList getDensificationVariables();
20     QStringList getDensificationResults();
21
22     uint32_t getSample_number() const;
23
24     float getVertical_displacement() const;
25     float getVertical_load() const;
26
27     void setDiff_sampleNumber_initExperiment(uint32_t diff);
28
29 private:
30     uint32_t sample_number;
31     float vertical_displacement;
32     float vertical_load;
33     uint32_t diff_sampleNumber_initExperiment;
42
43
44 };
45
46 #endif // DENSIFICATION_H
```

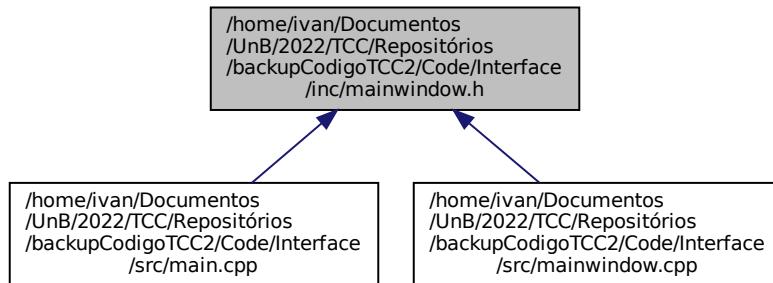
7.32 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup←
CódigoTCC2/Code/Interface/inc/mainwindow.h File Reference

```
#include < QMainWindow>
#include < QToolButton>
#include < QStackedWidget>
#include < QTimer>
#include "button.h"
#include "field.h"
#include "table.h"
#include "threadcontroller.h"
#include "sendcommands.h"
#include "dbmanager.h"
#include "exportdata.h"
#include "charts.h"
Include dependency graph for mainwindow.h
```

Include dependency graph for mainwindow.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [MainWindow](#)
Classe que instancia a janela principal da interface.

Namespaces

- namespace [Ui](#)

7.33 mainwindow.h

[Go to the documentation of this file.](#)

```

1 #ifndef MAINWINDOW_H
2 #define MAINWINDOW_H
3
4 #include <QMainWindow>
5 #include <QToolButton>
6 #include <QStackedWidget>
7 #include <QTimer>
8 #include "button.h"
9 #include "field.h"
10 #include "table.h"
11 #include "threadcontroller.h"
12 #include "sendcommands.h"
13 #include "dbmanager.h"
14 #include "exportdata.h"
15 #include "charts.h"
16 QT_BEGIN_NAMESPACE
17 namespace Ui { class MainWindow; }
18 QT_END_NAMESPACE
25 class MainWindow : public QMainWindow
26 {
27     Q_OBJECT
28
29 public:
30     MainWindow(QWidget *parent = nullptr);
31     ~MainWindow();
32     Experiment * info_variables;
33     DBManager * my_db;
34     exportData * data_export;
35     int export_option;
36     QTableWidget * chosenTable;
37 private slots:
38
39     void nextPhase();
40
41     void changePhase();

```

```

42     void changeOutsideExperimentPage();
43
44     void changeInitialPositionValue();
45
46     void onPositionButton_pressed();
47
48     void onPositionButton_released();
49
50
51     void cancelExperiment();
52
53
54     void on_initExperiment_toolButton_clicked();
55
56     void on_releasePressure_toolButton_clicked();
57
58     void updateResultsTables();
59
60     void adjustVelocity_Distance();
61
62     void initShearPhase();
63
64     void on_goBack_toolButton_clicked();
65
66     void changeVelocity();
67
68     void changeDistance();
69
70     void enableShearInitButton(int index);
71     void enableExportButton(int index);
72
73     void changeExportOption_Densification(int index);
74     void changeExportOption_Shear(int index);
75
76 private:
77     Ui::MainWindow *ui;
78     Button *setupButtons;
79     Field *setupFields;
80     Table *tables;
81     ThreadController *receive_data;
82     sendCommands *send_data;
83     QTimer * timer;
84     int previousIndex;
85     bool experiment_canceled = false;
86     Charts * charts_variables;
87
88     void fillTextEditForTests();
89     void InitialConfiguration_OutsideExperimentHeaderButtons();
90     void InitialConfiguration_InsideExperimentButtons();
91     void InitialConfiguration_PhasesButtons();
92     void InitialConfiguration_PhasesFields();
93     void InitialConfiguration_Tables();
94     void CreateDataTables();
95
96     void connectButtonToSlots_WithArguments(QHBoxLayout * boxlayout, QStackedWidget * stack_widget, int option);
97     void connectButtonsToSlots_Layout(QHBoxLayout * list, const char *signal, const char *slot);
98     void connectButtonsToSlots_Widget(QObject * list, const char *signal, const char *slot);
99     void changePage(QPushButton* buttonSender,QString buttons_name[6], uint8_t array_size,
100                  QStackedWidget * page_stack,uint8_t icon[3],uint8_t style);
101
102 };
103 #endif // MAINWINDOW_H

```

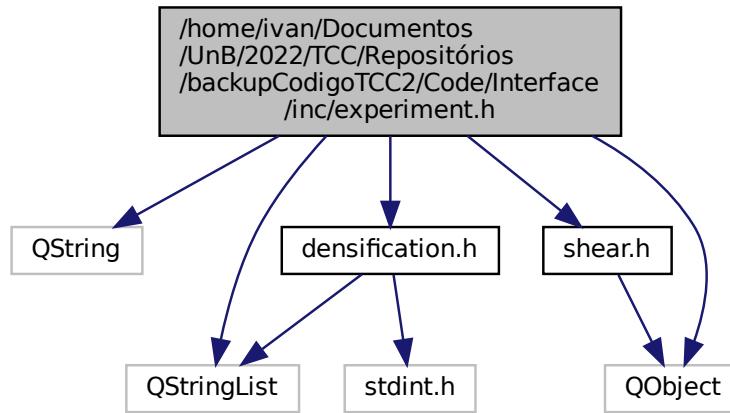
7.34 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/experiment.h File Reference

```

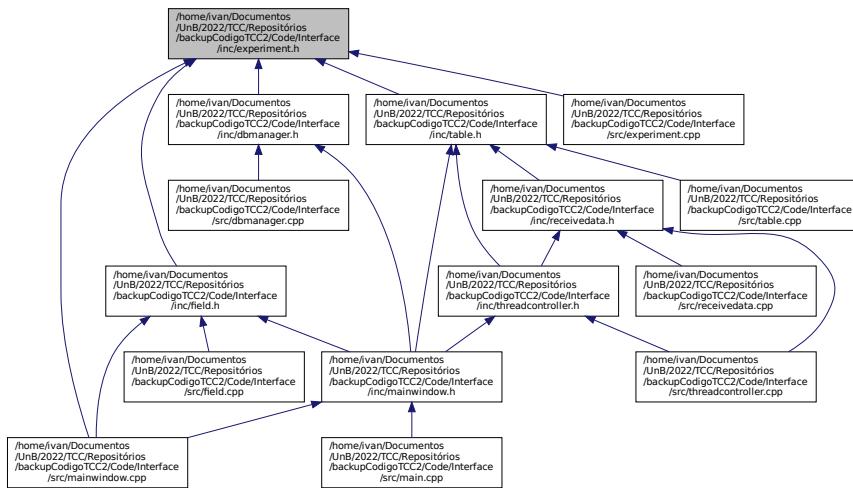
#include <QString>
#include <QStringList>
#include "densification.h"
#include "shear.h"
#include <QObject>

```

Include dependency graph for experiment.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Experiment](#)

Macros

- `#define pi_value 3.1415f`
- `#define densification_phase 0`
- `#define shear_phase 1`

7.34.1 Macro Definition Documentation

7.34.1.1 densification_phase

```
#define densification_phase 0
```

7.34.1.2 pi_value

```
#define pi_value 3.1415f
```

7.34.1.3 shear_phase

```
#define shear_phase 1
```

7.35 experiment.h

[Go to the documentation of this file.](#)

```
1 #ifndef EXPERIMENT_H
2 #define EXPERIMENT_H
3
4 #include <QString>
5 #include <QStringList>
6 #include "densification.h"
7 #include "shear.h"
8 #include <QObject>
9 #define pi_value 3.1415f
10 #define densification_phase 0
11 #define shear_phase 1
12
13 class Experiment : public QObject
14 {
15     Q_OBJECT
16
17 public:
18
19     Experiment();
20     QStringList getAllData_forInfoTable();
21     QStringList getAllData_forPhasesTable();
22
23     QStringList updateDensificationTable();
24     QStringList updateDensificationResultsTable();
25
26     QStringList updateShearTable();
27     QStringList updateShearResultsTable();
28
29     void insertData_inDatabase();
30
31     const QString &getName() const;
32     void setName(const QString &newName);
33
34     const QString &getOperator_name() const;
35     void setOperator_name(const QString &newOperator_name);
36
37     const QString &getTest_type() const;
38     void setTest_type(const QString &newTest_type);
39
40     const QString &getSpecimen_type() const;
```

```
42     void setSpecimen_type(const QString &newSpecimen_type);
43
44     const QString &getUscs_class() const;
45     void setUscs_class(const QString &newUscs_class);
46
47     const QString &getAshto_class() const;
48     void setAshto_class(const QString &newAshto_class);
49
50     const QString &getSample_preparations() const;
51     void setSample_preparations(const QString &newSample_preparations);
52
53     int getSample_id() const;
54     void setSample_id(int newSample_id);
55
56     int getBoring_number() const;
57     void setBoring_number(int newBoring_number);
58
59     const QString &getSample_location() const;
60     void setSample_location(const QString &newSample_location);
61
62     const QString &getSample_description() const;
63     void setSample_description(const QString &newSample_description);
64
65     float getInitial_height() const;
66     void setInitial_height(float newInitial_height);
67
68     float getInitial_wet_weight() const;
69     void setInitial_wet_weight(float newInitial_wet_weight);
70
71     float getInitial_moisture() const;
72     void setInitial_moisture(float newInitial_moisture);
73
74     float getSpgr_solids() const;
75     void setSpgr_solids(float newSpgr_solids);
76
77     float getPlastic_limit() const;
78     void setPlastic_limit(float newPlastic_limit);
79
80     float getLiquid_limit() const;
81     void setLiquid_limit(float newLiquid_limit);
82
83     float getInitial_position() const;
84     void setInitial_position(float newInitial_position);
85
86     float getDiameter() const;
87     void setDiameter(float newDiameter);
88
89     float getArea();
90     float getinitial_volume();
91     float getinitial_wet_density();
92     float getinitial_dry_density();
93     float getinitial_void_ratio();
94     float getwater_specific_weight();
95     float getinitial_saturation();
96
97
98     float getPressure() const;
99     void setPressure(float newPressure);
100    int getPhase() const;
101    void changePhase();
102
103   uint64_t getDuration(bool isDensification);
104
105   uint64_t getPresent_time_seconds() const;
106   uint64_t getInitial_time_seconds() const;
107
108   uint64_t getPresent_time_milliseconds() const;
109   uint64_t getInitial_time_milliseconds() const;
110
111   void setInitial_time(bool isInitial);
112
113   QString getInitialTimeString();
114
115   QString hour_min_sec_ms();
116   QString day_month_year();
117
118
119   bool getExperimentStarted() const;
120   void setExperimentStarted(bool newExperimentStarted);
121
122   QString getDuration_string(bool isDensification);
123
124   Densification densification_variables;
125   Shear shear_variables;
126
127   int getSample_period() const;
128   void setSample_period(int newSample_period);
```

```

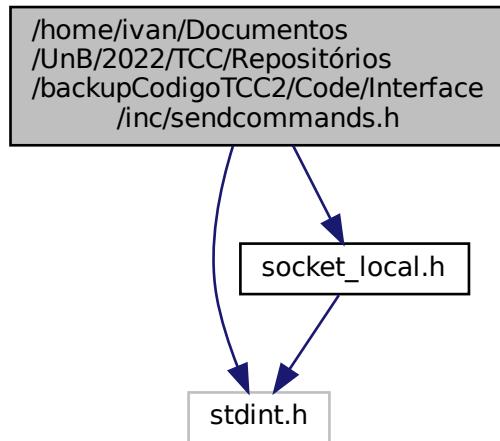
129
130     uint32_t getShearSample_number() const;
131     float getNormalTension();
132     float getShearTension();
133
134     float getAverageSpeed();
135
136 private:
137
138     bool experimentStarted = false;
139     QString name;
140     QString operator_name;
141     QString test_type;
142     QString specimen_type;
143     QString uscs_class;
144     QString ashto_class;
145     QString sample_preparations;
146     int sample_id;
147     int boring_number;
148     QString sample_location;
149     QString sample_description;
150     float initial_height;
151     float initial_wet_weight;
152     float initial_moisture;
153     float spgr_solids;
154     float plastic_limit;
155     float liquid_limit;
156     float initial_position;
157     float diameter;
158     float pressure;
159     uint64_t initial_time_seconds;
160     uint64_t present_time_seconds;
161     uint64_t initial_time_milliseconds;
162     uint64_t present_time_milliseconds;
163
164     int phase = densification_phase;
165     int sample_period;
166
167
168 signals:
169     void updateDensificationChart(int time, float vertical_displacement);
170     void updateShearChart(float horizontal_displacement, float vertical_displacement);
171
172
173 };
174
175 #endif // EXPERIMENT_H

```

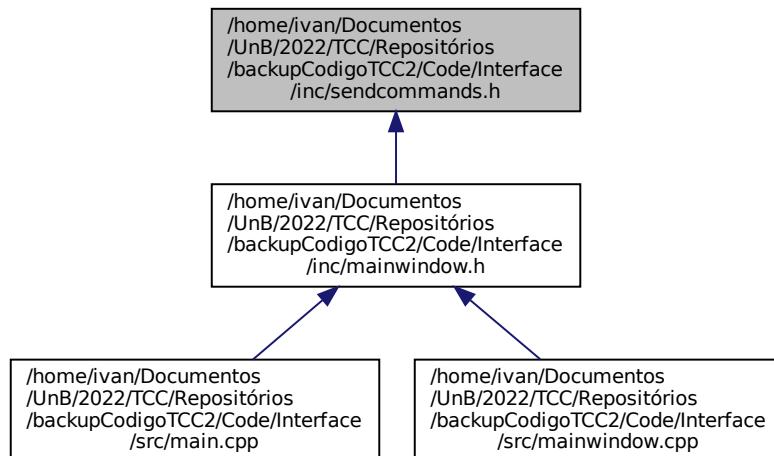
7.36 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/sendcommands.h File Reference

```
#include <stdint.h>
#include "socket_local.h"
```

Include dependency graph for sendcommands.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [sendCommands](#)

7.37 sendcommands.h

[Go to the documentation of this file.](#)

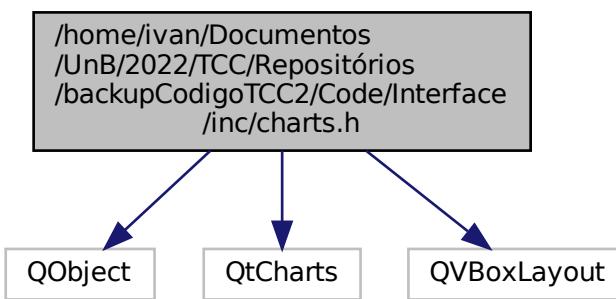
```

1 #ifndef SENDCOMMANDS_H
2 #define SENDCOMMANDS_H
3
4 #include<stdint.h>
5 #include "socket_local.h"
6
7 class sendCommands
8 {
9     private:
10     int socket_id;
11     char socket_name[100];
12     interface_to_machine_message interface_message;
13
14     bool enabled;
15     int16_t sampling_period;
16
17
18
19
20 public:
21     sendCommands();
22     ~sendCommands();
23     void setCommand(uint8_t command);
24     uint8_t errorOccurred;
25
26     uint8_t connectToMachine();
27     int16_t sendMessage();
28
29     void setVelocity(int16_t velocity);
30     void setDistance(int16_t distance);
31     void setPressure(uint8_t pressure);
32     void setEnabled(uint8_t enabled);
33     void setSamplingPeriod(uint16_t sample_period);
34     int16_t getVelocity();
35     int16_t getDistance();
36     uint16_t getPressure();
37     uint8_t getEnabled();
38     uint16_t getSamplingPeriod();
39
40 };
41
42 #endif // SENDCOMMANDS_H

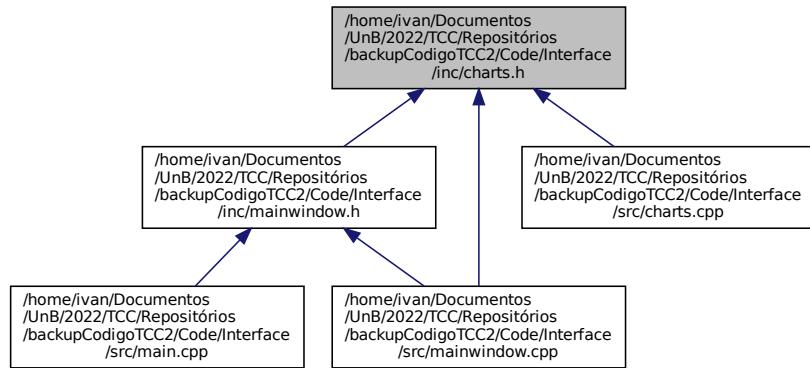
```

7.38 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/charts.h File Reference

```
#include <QObject>
#include <QtCharts>
#include <QVBoxLayout>
Include dependency graph for charts.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [Charts](#)

Classe responsável pelos gráficos.

Macros

- `#define densification_chart 0`
- `#define shear_chart 1`

7.38.1 Macro Definition Documentation

7.38.1.1 `densification_chart`

```
#define densification_chart 0
```

7.38.1.2 `shear_chart`

```
#define shear_chart 1
```

7.39 charts.h

[Go to the documentation of this file.](#)

```

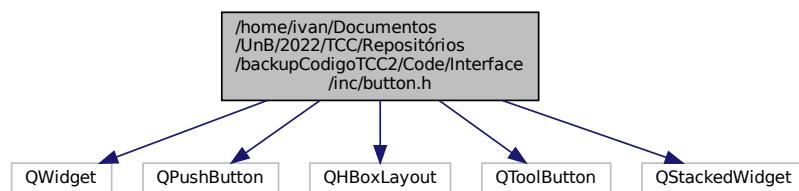
1 #ifndef CHARTS_H
2 #define CHARTS_H
3
4 #include <QObject>
5 #include <QtCharts>
6 #include <QVBoxLayout>
7
8 #define densification_chart 0
9 #define shear_chart 1
10 class Charts : public QObject
11 {
12     Q_OBJECT
13 public:
14     explicit Charts(QObject *parent = nullptr);
15     void initialConfiguration(QVBoxLayout * layout,int option);
16     void reset_Chart();
17 private:
18     QSplineSeries *series[2];
19     QChart *chart[2];
20     QValueAxis *m_axisX[2];
21     QValueAxis *m_axisY[2];
22     QString chart_title[2];
23     QString x_axis_title[2];
24     QString y_axis_title[2];
25     QChartView *chartView[2];
26 signals:
27
28 public slots:
29     void updateDensificationCharts(int time, float vertical_displacement);
30     void updateShearCharts(float horizontal_displacement, float vertical_displacement);
31 };
32
33
34
35
36
37
38 };
39
40 #endif // CHARTS_H

```

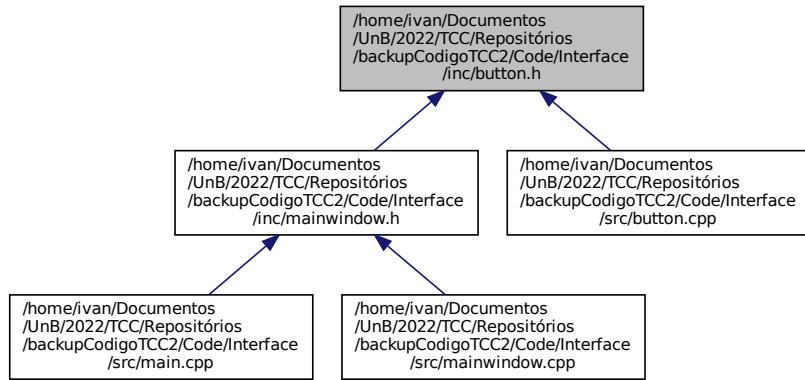
7.40 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup ↵ CodigoTCC2/Code/Interface/inc/button.h File Reference

```
#include <QWidget>
#include <QPushButton>
#include <QHBoxLayout>
#include <QToolButton>
#include <QStackedWidget>
```

Include dependency graph for button.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Button](#)

Classe dos botões.

Macros

- #define headerButton_lightBackgroundColor 0
- #define headerButton_darkBackgroundColor 1
- #define phasesButton_lightBackgroundColor 2
- #define phasesButton_darkBackgroundColor 3
- #define continueButton_BackgroundColor 4
- #define pressureButton_GreenBackgroundColor 5
- #define pressureButton_RedBackgroundColor 6
- #define moveButton_DisabledBackgroundColor 7
- #define cancelButton_BackgroundColor 8
- #define initShearButton_BackgroundColor 9
- #define velocityPositionBackgroundColor 10
- #define historyButton_lightIcon 0
- #define historyButton_darkIcon 1
- #define experimentButton_lightIcon 2
- #define experimentButton_darkIcon 3
- #define configurationButton_lightIcon 4
- #define configurationButton_darkIcon 5
- #define continueButton_Icon 6
- #define densificationButton_lightIcon 7
- #define densificationButton_darkIcon 8
- #define shearButton_lightIcon 9
- #define shearButton_darkIcon 10
- #define infoButton_lightIcon 11
- #define infoButton_darkIcon 12
- #define finishButton_Icon 13
- #define no_icon 255
- #define outsideExperiment_buttonSize 0

- #define phases_buttonSize 1
- #define phases_continueButtonSize 2
- #define cancelButton_size 3
- #define initShearButton_size 4
- #define velocityPositionButton_size 5
- #define exportButton_size 6

7.40.1 Macro Definition Documentation

7.40.1.1 cancelButton_BackgroundColor

```
#define cancelButton_BackgroundColor 8
```

Número identificador do estilo do botão de cancelar o experimento.

7.40.1.2 cancelButton_size

```
#define cancelButton_size 3
```

Número identificador do tamanho do botão de cancelar o experimento .

7.40.1.3 configurationButton_darkIcon

```
#define configurationButton_darkIcon 5
```

Número identificador do ícone do botão configurações para ser utilizado com o background escuro.

7.40.1.4 configurationButton_lightIcon

```
#define configurationButton_lightIcon 4
```

Número identificador do ícone do botão configurações para ser utilizado com o background claro.

7.40.1.5 continueButton_BackgroundColor

```
#define continueButton_BackgroundColor 4
```

Número identificador do estilo do botão de continuar nas etapas.

7.40.1.6 continueButton_Icon

```
#define continueButton_Icon 6
```

Número identificador do ícone do botão continuar

7.40.1.7 **densificationButton_darkIcon**

```
#define densificationButton_darkIcon 8
```

Número identificador do ícone do botão adensamento para ser utilizado com o background escuro.

7.40.1.8 **densificationButton_lightIcon**

```
#define densificationButton_lightIcon 7
```

Número identificador do ícone do botão adensamento para ser utilizado com o background claro.

7.40.1.9 **experimentButton_darkIcon**

```
#define experimentButton_darkIcon 3
```

Número identificador do ícone do botão experimento para ser utilizado com o background escuro.

7.40.1.10 **experimentButton_lightIcon**

```
#define experimentButton_lightIcon 2
```

Número identificador do ícone do botão experimento para ser utilizado com o background claro.

7.40.1.11 **exportButton_size**

```
#define exportButton_size 6
```

Número identificador do tamanho do botão de exportar tabelas.

7.40.1.12 **finishButton_Icon**

```
#define finishButton_Icon 13
```

Número identificador do ícone do botão finalizar.

7.40.1.13 **headerButton_darkBackgroundColor**

```
#define headerButton_darkBackgroundColor 1
```

Número identificador do estilo do botão header com cores escuras.

7.40.1.14 **headerButton_lightBackgroundColor**

```
#define headerButton_lightBackgroundColor 0
```

Número identificador do estilo do botão header com cores claras.

7.40.1.15 historyButton_darkIcon

```
#define historyButton_darkIcon 1
```

Número identificador do ícone do botão histórico para ser utilizado com o background escuro.

7.40.1.16 historyButton_lightIcon

```
#define historyButton_lightIcon 0
```

Número identificador do ícone do botão histórico para ser utilizado com o background claro.

7.40.1.17 infoButton_darkIcon

```
#define infoButton_darkIcon 12
```

Número identificador do ícone do botão informações para ser utilizado com o background escuro.

7.40.1.18 infoButton_lightIcon

```
#define infoButton_lightIcon 11
```

Número identificador do ícone do botão informações para ser utilizado com o background claro.

7.40.1.19 initShearButton_BackgroundColor

```
#define initShearButton_BackgroundColor 9
```

Número identificador do estilo do botão de iniciar o cisalhamento.

7.40.1.20 initShearButton_size

```
#define initShearButton_size 4
```

Número identificador do tamanho do botão de iniciar o cisalhamento .

7.40.1.21 moveButton_DisabledBackgroundColor

```
#define moveButton_DisabledBackgroundColor 7
```

Número identificador do estilo do botão de mover desativado.

7.40.1.22 no_icon

```
#define no_icon 255
```

Número que informa que o botão não possui ícone.

7.40.1.23 outsideExperiment_buttonSize

```
#define outsideExperiment_buttonSize 0
```

Número identificador do tamanho dos botões no cabeçalho .

7.40.1.24 phases_buttonSize

```
#define phases_buttonSize 1
```

Número identificador do tamanho dos botões das fases (etapa1,etapa2 ...).

7.40.1.25 phases_continueButtonSize

```
#define phases_continueButtonSize 2
```

Número identificador do tamanho dos botões de continuar .

7.40.1.26 phasesButton_darkBackgroundColor

```
#define phasesButton_darkBackgroundColor 3
```

Número identificador do estilo do botão etapas com cores escuras.

7.40.1.27 phasesButton_lightBackgroundColor

```
#define phasesButton_lightBackgroundColor 2
```

Número identificador do estilo do botão etapas com cores claras.

7.40.1.28 pressureButton_GreenBackgroundColor

```
#define pressureButton_GreenBackgroundColor 5
```

Número identificador do estilo do botão de pressão ligado.

7.40.1.29 pressureButton_RedBackgroundColor

```
#define pressureButton_RedBackgroundColor 6
```

Número identificador do estilo do botão de pressão desligado.

7.40.1.30 shearButton_darkIcon

```
#define shearButton_darkIcon 10
```

Número identificador do ícone do botão cisalhamento para ser utilizado com o background escuro.

7.40.1.31 shearButton_lightIcon

```
#define shearButton_lightIcon 9
```

Número identificador do ícone do botão cisalhamento para ser utilizado com o background claro.

7.40.1.32 velocityPositionBackgroundColor

```
#define velocityPositionBackgroundColor 10
```

Número identificador do estilo dos botões de alterar a posição e velocidade.

7.40.1.33 velocityPositionButton_size

```
#define velocityPositionButton_size 5
```

Número identificador do tamanho dos botões de ajustar as variáveis de velocidade e distância .

7.41 button.h

[Go to the documentation of this file.](#)

```
1 #ifndef BUTTON_H
2 #define BUTTON_H
3
4 #include <QWidget>
5 #include <QPushButton>
6 #include <QHBoxLayout>
7 #include <QToolButton>
8 #include <QStackedWidget>
9
10 #define headerButton_lightBackgroundColor 0
11 #define headerButton_darkBackgroundColor 1
12 #define phasesButton_lightBackgroundColor 2
13 #define phasesButton_darkBackgroundColor 3
14 #define continueButton_BackgroundColor 4
15 #define pressureButton_GreenBackgroundColor 5
16 #define pressureButton_RedBackgroundColor 6
17 #define moveButton_DisabledBackgroundColor 7
18 #define cancelButton_BackgroundColor 8
19 #define initShearButton_BackgroundColor 9
20 #define velocityPositionBackgroundColor 10
22 #define historyButton_lightIcon 0
23 #define historyButton_darkIcon 1
24 #define experimentButton_lightIcon 2
25 #define experimentButton_darkIcon 3
26 #define configurationButton_lightIcon 4
27 #define configurationButton_darkIcon 5
28 #define continueButton_Icon 6
30 #define densificationButton_lightIcon 7
31 #define densificationButton_darkIcon 8
32 #define shearButton_lightIcon 9
33 #define shearButton_darkIcon 10
34 #define infoButton_lightIcon 11
35 #define infoButton_darkIcon 12
36 #define finishButton_Icon 13
38 #define no_icon 255
39 #define outsideExperiment_buttonSize 0
40 #define phases_buttonSize 1
41 #define phases_continueButtonSize 2
42 #define cancelButton_size 3
43 #define initShearButton_size 4
44 #define velocityPositionButton_size 5
45 #define exportButton_size 6
51 class Button: public QObject
52 {
53     Q_OBJECT
54 public:
```

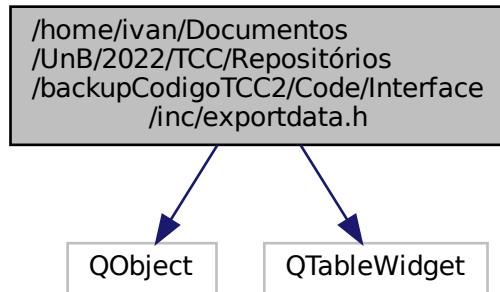
```

55     Button();
56     void setButtonShadow(QToolButton * button);
57     void setButton_style_icon(QToolButton * button, uint8_t style, uint8_t icon);
58     void initialButtonStyling_Layout(QHBoxLayout * boxlayout, uint8_t style_option, uint8_t size);
59     void changeButton_style(QToolButton *current_button, uint8_t icon, uint8_t style, uint8_t pos);
60     void initialButtonStyling_Widget(QObject * selectedWidget, uint8_t style_option, uint8_t size);
61     void initExperiment_ButtonStyle(QToolButton *play_button, bool enabled);
62     void pressureButton_style(QToolButton * pressure);
63     void initialButtonStyling(QToolButton * button, uint8_t style_option, uint8_t size);
64
65     void changeInitShear_toFinishButton(QToolButton * button);
66
67 private:
68     QToolButton * clickedButton[3];
69     uint8_t clickedButtonIcon[3];
70     QString button_styleSheets[15];
71     QString button_images[20];
72     QIcon button_icons[20];
73     QSize buttonMaximumSize[10];
74     QSize buttonMinimumSize[10];
75     void styleSheetConfiguration();
76     void imageConfiguration();
77     void iconsConfiguration();
78     void buttonSizeConfiguration();
79
80 signals:
81
82 public slots:
83     void changeHeaderPage_InsideExperiment(QStackedWidget *insideExperiment);
84     void changePage_InsideExperiment(QStackedWidget *insideExperiment, bool isDensificationPage);
85
86
87 };
88
89 #endif // BUTTON_H

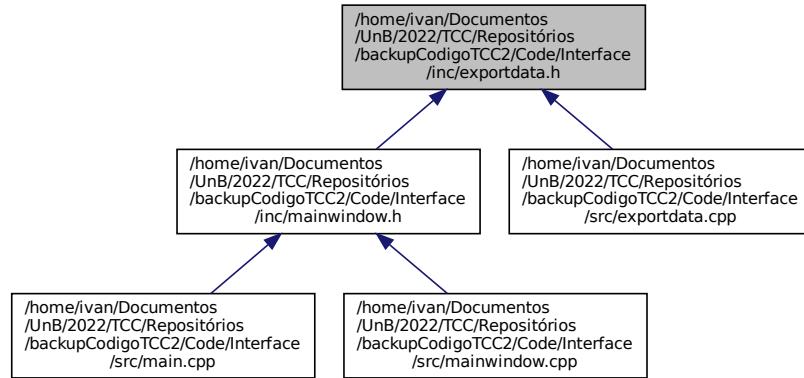
```

7.42 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/exportdata.h File Reference

```
#include <QObject>
#include <QTableWidget>
Include dependency graph for exportdata.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [exportData](#)

Classe que exporta os dados para csv.

7.43 exportdata.h

[Go to the documentation of this file.](#)

```

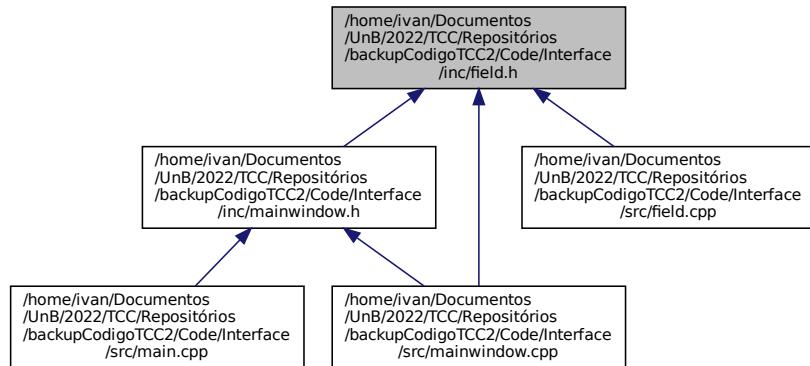
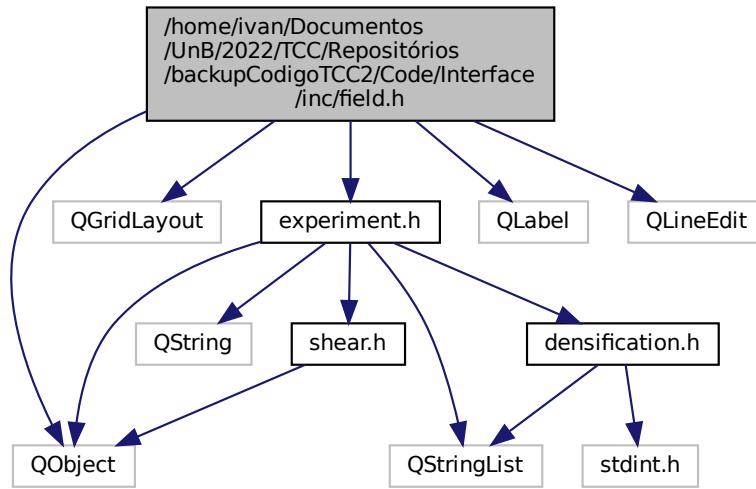
1 #ifndef EXPORTDATA_H
2 #define EXPORTDATA_H
3
4 #include <QObject>
5 #include <QTableWidget>
12 class exportData : public QObject
13 {
14     Q_OBJECT
15 public:
16     explicit exportData(QObject *parent = nullptr);
17     ~exportData();
18
19 private:
20     void write(QTableWidget *table_widget,QString filename);
21 signals:
22
23 public slots:
24     void exportCSV(QTableWidget *table_widget, int file_identification, int option);
25 };
26
27 #endif // EXPORTDATA_H
  
```

7.44 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/field.h File Reference

```

#include <QObject>
#include <QGridLayout>
#include "experiment.h"
#include <QLabel>
  
```

```
#include <QLineEdit>
Include dependency graph for field.h:
```



Classes

- class [Field](#)

Classe dos campos de texto.

7.45 field.h

[Go to the documentation of this file.](#)

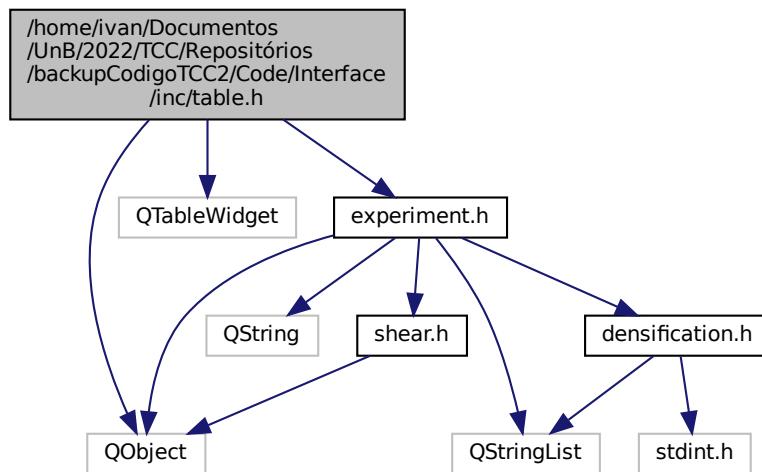
```

1 #ifndef FIELD_H
2 #define FIELD_H
3
4 #include <QObject>
5 #include <QGridLayout>
6 #include "experiment.h"
7 #include <QLabel>
8 #include <QLineEdit>
15 class Field : public QObject
16 {
17     Q_OBJECT
18 public:
19     Field(Experiment *parent = nullptr);
20     void customizeField(QLayout *gridLayout);
21     void customizeOneField(QLabel *label, QLineEdit *lineEdit);
22     void clearFields(QLayout *gridLayout);
23     Experiment * info_variables;
25 private:
26     QSize FieldMaximumSize[10];
27     QSize FieldMinimumSize[10];
29 signals:
30
31 public slots:
32     void setVariables();
33
34
35 };
36
37 #endif // FIELD_H

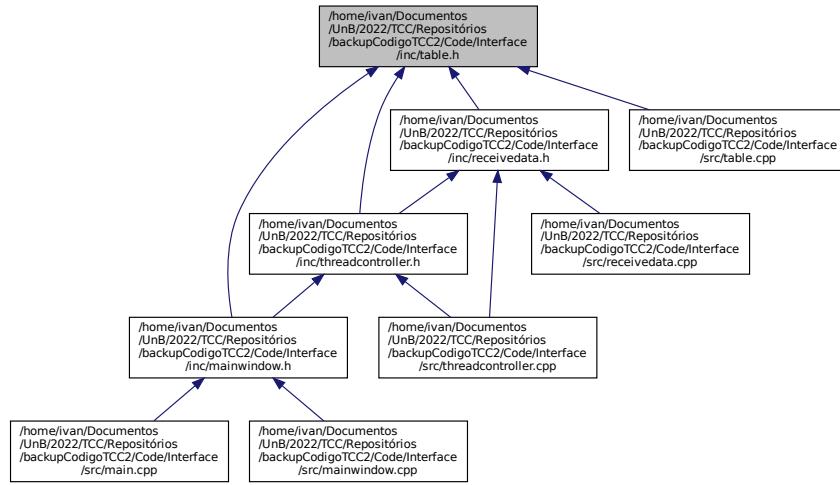
```

7.46 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backup← CodigoTCC2/Code/Interface/inc/table.h File Reference

```
#include <QObject>
#include <QTableWidget>
#include "experiment.h"
Include dependency graph for table.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [Table](#)

Macros

- `#define phases_table 0`
- `#define info_table 1`
- `#define densification_result_table 2`
- `#define shear_result_table 3`
- `#define design_densification_table 0`
- `#define design_shear_table 1`

7.46.1 Macro Definition Documentation

7.46.1.1 `densification_result_table`

```
#define densification_result_table 2
```

7.46.1.2 `design_densification_table`

```
#define design_densification_table 0
```

7.46.1.3 design_shear_table

```
#define design_shear_table 1
```

7.46.1.4 info_table

```
#define info_table 1
```

7.46.1.5 phases_table

```
#define phases_table 0
```

7.46.1.6 shear_result_table

```
#define shear_result_table 3
```

7.47 table.h

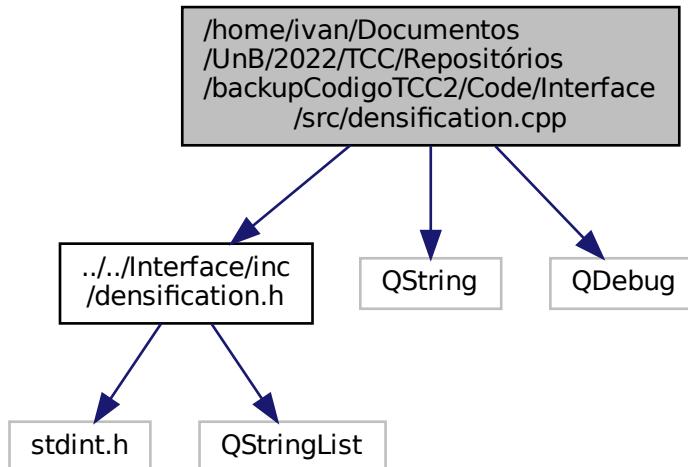
[Go to the documentation of this file.](#)

```
1 #ifndef TABLE_H
2 #define TABLE_H
3
4 #include <QObject>
5 #include <QTableWidget>
6 #include "experiment.h"
7
8 #define phases_table 0
9 #define info_table 1
10 #define densification_result_table 2
11 #define shear_result_table 3
12 #define design_densification_table 0
13 #define design_shear_table 1
14
15 class Table : public QObject
16 {
17     Q_OBJECT
18 public:
19     explicit Table(Experiment *parent = nullptr, QTableWidget *densification = nullptr
20                 , QTableWidget *shear = nullptr);
21     void customizeTable(QTableWidget *table_widget);
22
23     void initialConfig_ShearTable(QTableWidget *table_widget);
24     void updateData_ShearTable(QTableWidget *table_widget);
25
26     void initialConfig_StaticTable(QTableWidget *table_widget, uint8_t option);
27     void updateData_StaticTable(QTableWidget *table_widget, uint8_t option);
28
29     void updateData_DynamicTable(QStringList data, uint8_t option);
30     void initialConfig_DynamicTable(QTableWidget *table_widget, uint8_t option);
31
32     void clearDynamicTables(QTableWidget *table_widget);
33     void clearStaticTables(QTableWidget *table_widget);
34
35
36
37     void test();
```

```
39     Experiment * table_variables;
40
41 private:
42
43     QStringList lineNames[4];
44     QStringList columnValues;
45
46     QStringList headerNames[2];
47     QStringList machineTablevalues[2];
48
49     QTableWidget *densificationTable;
50     QTableWidget *shearTable;
51
52 signals:
53
54 };
55
56 #endif // TABLE_H
```

7.48 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/densification.cpp File Reference

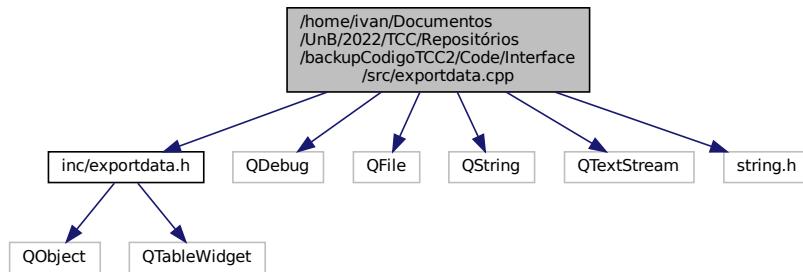
```
#include "../../Interface/inc/densification.h"
#include <QString>
#include <QDebug>
Include dependency graph for densification.cpp:
```



7.49 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/exportdata.cpp File Reference

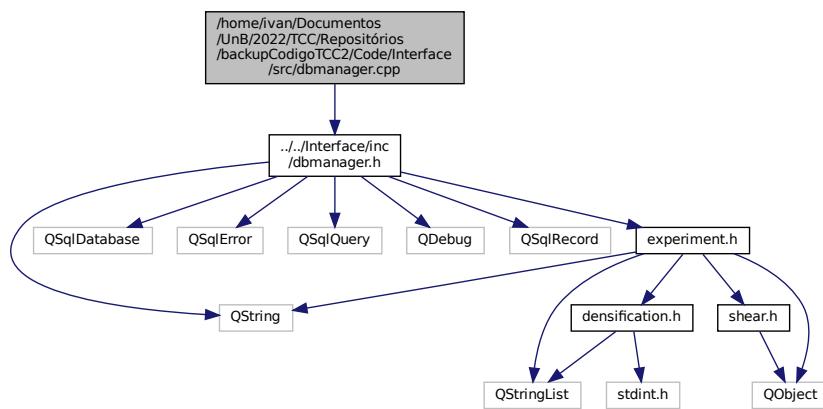
```
#include "inc/exportdata.h"
#include <QDebug>
#include <QFile>
#include <QString>
```

```
#include <QTextStream>
#include <string.h>
Include dependency graph for exportdata.cpp:
```



7.50 /home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/dbmanager.cpp File Reference

```
#include "../../../Interface/inc/dbmanager.h"
Include dependency graph for dbmanager.cpp:
```



Index

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
232, 238
197

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
230, 232
201

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
215, 218
199

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
220, 221
197

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
224, 226
204, 205

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
239, 240
205, 206

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
240, 241
207

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
222, 223
207

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
213, 214
209

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2
228, 229
208

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/shear.h,
219, 220
DBManager, 34

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/socket_local.h,
209, 212
MainWindow, 121

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/table.h,
242, 244
QTstyle_Test, 148

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/inc/threadcontroller.h,
214, 215
SocketTests, 179

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/button.cpp,
200
ThreadController, 194

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/charts.cpp,
204
exportData, 95

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/dbmanager.cpp,
246
machineClient, 110

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/densification.cpp,
245
machineServer, 114

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/experiment.cpp,
202
receiveData, 154

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/exportdata.cpp,
245
sendCommands, 159

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/field.cpp,
198
adjustVelocity_Distance

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/main.cpp,
203
MainWindow, 121

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/mainwindow.cpp,
199
boring_number

/home/ivan/Documentos/UnB/2022/TCC/Repositórios/backupCodigoTCC2/Code/Interface/src/receivedata.cpp,
201
Experiment, 89

Button, 11

Button, 13

button_icons, 23
 button_images, 23
 button_styleSheets, 23
 buttonMaximumSize, 24
 buttonMinimumSize, 24
 buttonSizeConfiguration, 14
 changeButton_style, 14
 changeHeaderPage_InsideExperiment, 16
 changeInitShear_toFinishButton, 16
 changePage_InsideExperiment, 17
 clickedButton, 24
 clickedButtonIcon, 24
 iconsConfiguration, 17
 imageConfiguration, 18
 initExperiment_ButtonStyle, 18
 initialButtonStyling, 19
 initialButtonStyling_Layout, 19
 initialButtonStyling_Widget, 20
 pressureButton_style, 21
 setButton_style_icon, 22
 setButtonShadow, 22
 styleSheetConfiguration, 23
 button.h
 cancelButton_BackgroundColor, 234
 cancelButton_size, 234
 configurationButton_darkIcon, 234
 configurationButton_lightIcon, 234
 continueButton_BackgroundColor, 234
 continueButton_Icon, 234
 densificationButton_darkIcon, 234
 densificationButton_lightIcon, 235
 experimentButton_darkIcon, 235
 experimentButton_lightIcon, 235
 exportButton_size, 235
 finishButton_Icon, 235
 headerButton_darkBackgroundColor, 235
 headerButton_lightBackgroundColor, 235
 historyButton_darkIcon, 235
 historyButton_lightIcon, 236
 infoButton_darkIcon, 236
 infoButton_lightIcon, 236
 initShearButton_BackgroundColor, 236
 initShearButton_size, 236
 moveButton_DisabledBackgroundColor, 236
 no_icon, 236
 outsideExperiment_buttonSize, 236
 phases_buttonSize, 237
 phases_continueButtonSize, 237
 phasesButton_darkBackgroundColor, 237
 phasesButton_lightBackgroundColor, 237
 pressureButton_GreenBackgroundColor, 237
 pressureButton_RedBackgroundColor, 237
 shearButton_darkIcon, 237
 shearButton_lightIcon, 237
 velocityPositionBackgroundColor, 238
 velocityPositionButton_size, 238
 button_icons
 Button, 23
 button_images
 Button, 23
 button_styleSheets
 Button, 23
 buttonMaximumSize
 Button, 24
 buttonMinimumSize
 Button, 24
 buttonSizeConfiguration
 Button, 14
 cancelButton_BackgroundColor
 button.h, 234
 cancelButton_size
 button.h, 234
 cancelExperiment
 MainWindow, 123
 changeButton_style
 Button, 14
 changeDistance
 MainWindow, 124
 changeExportOption_Densification
 MainWindow, 124
 changeExportOption_Shear
 MainWindow, 124
 changeHeaderPage_InsideExperiment
 Button, 16
 changeInitialPositionValue
 MainWindow, 125
 changeInitShear_toFinishButton
 Button, 16
 changeOutsideExperimentPage
 MainWindow, 125
 changePage
 MainWindow, 126
 changePage_InsideExperiment
 Button, 17
 changePhase
 Experiment, 58
 MainWindow, 127
 changeVelocity
 MainWindow, 128
 chart
 Charts, 29
 chart_title
 Charts, 29
 Charts, 25
 chart, 29
 chart_title, 29
 Charts, 27
 chartView, 29
 initialConfiguration, 27
 m_axisX, 29
 m_axisY, 29
 reset_Chart, 28
 series, 29
 updateDensificationCharts, 28
 updateShearCharts, 28
 x_axis_title, 30

y_axis_title, 30
charts.h
densification_chart, 231
shear_chart, 231
charts_variables
MainWindow, 144
chartView
Charts, 29
chosenTable
MainWindow, 145
cleanupTestCase
SocketTests, 179
clearDynamicTables
Table, 186
clearFields
Field, 99
clearStaticTables
Table, 186
clickedButton
Button, 24
clickedButtonIcon
Button, 24
client_socket_id
machineServer, 116
receiveData, 157
columnValues
Table, 190
command
interface_to_machine_message, 106
compareStructFloatElements
SocketTests, 179
compareStructIntElements
SocketTests, 180
configurationButton_darkIcon
button.h, 234
configurationButton_lightIcon
button.h, 234
connectButtonsToSlots_Layout
MainWindow, 129
connectButtonsToSlots_Widget
MainWindow, 129
connectButtonToSlots_WithArguments
MainWindow, 130
connectToMachine
sendCommands, 159
continueButton_BackgroundColor
button.h, 234
continueButton_Icon
button.h, 234
create_table
DBManager, 46
CreateDatabaseTables
MainWindow, 130
createTable
DBManager, 34
customizeField
Field, 100
customizeOneField
Field, 101
customizeTable
Table, 186
data_arrived
receiveData, 154
data_export
MainWindow, 145
date
Shear, 174
day_month_year
Experiment, 58
DBManager, 30
~DBManager, 34
create_table, 46
createTable, 34
DBManager, 34
experiment_data, 46
experiment_id, 47
getExperiment_id, 35
insert_into_table, 47
insertIntoTable, 35
insertValuesIntoBind_Densification, 37
insertValuesIntoBind_Experiment, 37
insertValuesIntoBind_FinalVariables, 40
insertValuesIntoBind_SampleVariables, 41
insertValuesIntoBind_SampleVariablesUpdate, 42
insertValuesIntoBind_Shear, 42
isOpen, 43
prova_conceito_database, 47
selectExperimentId, 43
table_name, 47
tableExists, 44
update_database_table, 45
updateTable, 46
dbmanager.h
densification_table, 217
experiment_table, 217
final_variables_table, 217
sample_table, 217
shear_table, 217
Densification, 48
Densification, 49
diff_sampleNumber_initExperiment, 52
getDensificationResults, 49
getDensificationVariables, 49
getSample_number, 50
getVertical_displacement, 50
getVertical_load, 51
sample_number, 53
setDensificationVariables, 51
setDiff_sampleNumber_initExperiment, 52
vertical_displacement, 53
vertical_load, 53
densification_chart
charts.h, 231
densification_phase
experiment.h, 226
densification_result_table

table.h, 243
 densification_table
 dbmanager.h, 217
 densification_variables
 Experiment, 89
 densificationButton_darkIcon
 button.h, 234
 densificationButton_lightIcon
 button.h, 235
 densificationTable
 Table, 190
 design_densification_table
 table.h, 243
 design_shear_table
 table.h, 243
 diameter
 Experiment, 89
 diff_sampleNumber_initExperiment
 Densification, 52
 displacement
 machine_to_interface_message, 108
 distance
 interface_to_machine_message, 106
 Shear, 174
 enabled
 interface_to_machine_message, 106
 sendCommands, 163
 enableExportButton
 MainWindow, 131
 enableShearInitButton
 MainWindow, 132
 enumPtr
 QTstyle_Test, 149
 enumVar
 QTstyle_Test, 150
 errorOccurred
 machineClient, 111
 machineServer, 116
 receiveData, 157
 sendCommands, 163
 Experiment, 53
 ashto_class, 89
 boring_number, 89
 changePhase, 58
 day_month_year, 58
 densification_variables, 89
 diameter, 89
 Experiment, 57
 experimentStarted, 90
 getAllData_forInfoTable, 58
 getAllData_forPhasesTable, 59
 getArea, 60
 getAshto_class, 60
 getAverageSpeed, 61
 getBoring_number, 61
 getDiameter, 62
 getDuration, 62
 getDuration_string, 63
 getExperimentStarted, 64
 getInitial_dry_density, 64
 getInitial_height, 65
 getInitial_moisture, 65
 getInitial_position, 65
 getInitial_saturation, 66
 getInitial_time_milliseconds, 66
 getInitial_time_seconds, 66
 getInitialTimeString, 67
 getInitial_void_ratio, 67
 getInitial_volume, 68
 getInitial_wet_density, 68
 getInitial_wet_weight, 69
 getLiquid_limit, 69
 getName, 70
 getNormalTension, 70
 getOperator_name, 71
 getPhase, 71
 getPlastic_limit, 71
 getPresent_time_milliseconds, 72
 getPresent_time_seconds, 72
 getPressure, 72
 getSample_description, 73
 getSample_id, 73
 getSample_location, 73
 getSample_period, 74
 getSample_preparations, 74
 getShearSample_number, 74
 getShearTension, 75
 getSpecimen_type, 75
 getSpgr_solids, 76
 getTest_type, 76
 getUscs_class, 76
 getwater_specific_weight, 77
 hour_min_sec_ms, 77
 initial_height, 90
 initial_moisture, 90
 initial_position, 90
 initial_time_milliseconds, 90
 initial_time_seconds, 90
 initial_wet_weight, 90
 insertData_inDatabase, 78
 liquid_limit, 90
 name, 91
 operator_name, 91
 phase, 91
 plastic_limit, 91
 present_time_milliseconds, 91
 present_time_seconds, 91
 pressure, 91
 sample_description, 91
 sample_id, 92
 sample_location, 92
 sample_period, 92
 sample_preparations, 92
 setAshto_class, 78
 setBoring_number, 78
 setDiameter, 78

setExperimentStarted, 79
setInitial_height, 79
setInitial_moisture, 79
setInitial_position, 80
setInitial_time, 80
setInitial_wet_weight, 80
setLiquid_limit, 81
setName, 81
setOperator_name, 81
setPlastic_limit, 82
setPressure, 82
setSample_description, 82
setSample_id, 83
setSample_location, 83
setSample_period, 83
setSample_preparations, 84
setSpecimen_type, 84
setSpgr_solids, 84
setTest_type, 85
setUscs_class, 85
shear_variables, 92
specimen_type, 92
spgr_solids, 92
test_type, 92
updateDensificationChart, 85
updateDensificationResultsTable, 86
updateDensificationTable, 86
updateShearChart, 87
updateShearResultsTable, 87
updateShearTable, 88
uscs_class, 93
experiment.h
densification_phase, 226
pi_value, 226
shear_phase, 226
experiment_canceled
 MainWindow, 145
experiment_data
 DBManager, 46
experiment_id
 DBManager, 47
experiment_table
 dbmanager.h, 217
experimentButton_darkIcon
 button.h, 235
experimentButton_lightIcon
 button.h, 235
experimentStarted
 Experiment, 90
export_option
 MainWindow, 145
exportButton_size
 button.h, 235
exportCSV
 exportData, 95
exportData, 93
 ~exportData, 95
 exportCSV, 95
exportData, 95
Field, 96
 clearFields, 99
 customizeField, 100
 customizeOneField, 101
 Field, 99
 FieldMaximumSize, 105
 FieldMinimumSize, 105
 info_variables, 105
 setVariables, 103
FieldMaximumSize
 Field, 105
FieldMinimumSize
 Field, 105
fillTextEditForTests
 MainWindow, 132
final_variables_table
 dbmanager.h, 217
finishButton_icon
 button.h, 235
getAllData_forInfoTable
 Experiment, 58
getAllData_forPhasesTable
 Experiment, 59
getArea
 Experiment, 60
getAshto_class
 Experiment, 60
getAverageSpeed
 Experiment, 61
getBoring_number
 Experiment, 61
getDate
 Shear, 166
getDensificationResults
 Densification, 49
getDensificationVariables
 Densification, 49
getDiameter
 Experiment, 62
getDistance
 sendCommands, 160
 Shear, 166
getDuration
 Experiment, 62
getDuration_string
 Experiment, 63
getEnabled
 sendCommands, 160
getExperiment_id
 DBManager, 35
getExperimentStarted
 Experiment, 64
getHorizontal_displacement
 Shear, 167
getHorizontal_load

Shear, 167
getHour_min_sec_ms
 Shear, 167
getinitial_dry_density
 Experiment, 64
getInitial_height
 Experiment, 65
getInitial_moisture
 Experiment, 65
getInitial_position
 Experiment, 65
getinitial_saturation
 Experiment, 66
getInitial_time_milliseconds
 Experiment, 66
 Shear, 168
getInitial_time_seconds
 Experiment, 66
 Shear, 168
getInitialTimeString
 Experiment, 67
getinitial_void_ratio
 Experiment, 67
getinitial_volume
 Experiment, 68
getinitial_wet_density
 Experiment, 68
getinitial_wet_weight
 Experiment, 69
getLiquid_limit
 Experiment, 69
getName
 Experiment, 70
getNormalTension
 Experiment, 70
getOperator_name
 Experiment, 71
getPhase
 Experiment, 71
getPlastic_limit
 Experiment, 71
getPresent_time_milliseconds
 Experiment, 72
getPresent_time_seconds
 Experiment, 72
getPressure
 Experiment, 72
 sendCommands, 160
getSample_description
 Experiment, 73
getSample_id
 Experiment, 73
getSample_location
 Experiment, 73
getSample_number
 Densification, 50
getSample_number_diff
 Shear, 168
getSample_period
 Experiment, 74
getSample_preparations
 Experiment, 74
getSamplingPeriod
 sendCommands, 160
getShearResults
 Shear, 169
getShearSample_number
 Experiment, 74
getShearTension
 Experiment, 75
getShearVariables
 Shear, 169
getSpecimen_type
 Experiment, 75
getSpgr_solids
 Experiment, 76
getTest_type
 Experiment, 76
getUscs_class
 Experiment, 76
getVelocity
 sendCommands, 160
 Shear, 170
getVertical_displacement
 Densification, 50
getVertical_load
 Densification, 51
getwater_specific_weight
 Experiment, 77
handler
 QTstyle_Test, 150
headerButton_darkBackgroundColor
 button.h, 235
headerButton_lightBackgroundColor
 button.h, 235
headerNames
 Table, 190
historyButton_darkIcon
 button.h, 235
historyButton_lightIcon
 button.h, 236
horizontal_displacement
 Shear, 174
horizontal_load
 Shear, 174
hour_min_sec_ms
 Experiment, 77
 Shear, 174
iconsConfiguration
 Button, 17
imageConfiguration
 Button, 18
info_table
 table.h, 244
info_variables

Field, 105
MainWindow, 145
SocketTests, 182
infoButton_darkIcon
button.h, 236
infoButton_lightIcon
button.h, 236
init
machineServer, 114
receiveData, 154
initExperiment_ButtonStyle
Button, 18
initial_height
Experiment, 90
initial_moisture
Experiment, 90
initial_position
Experiment, 90
initial_time_milliseconds
Experiment, 90
Shear, 174
initial_time_seconds
Experiment, 90
Shear, 175
initial_wet_weight
Experiment, 90
initialButtonStyling
Button, 19
initialButtonStyling_Layout
Button, 19
initialButtonStyling_Widget
Button, 20
initialConfig_DynamicTable
Table, 187
initialConfig_ShearTable
Table, 187
initialConfig_StaticTable
Table, 187
initialConfiguration
Charts, 27
InitialConfiguration_InsideExperimentButtons
MainWindow, 134
InitialConfiguration_OutsideExperimentHeaderButtons
MainWindow, 135
InitialConfiguration_PhasesButtons
MainWindow, 136
InitialConfiguration_PhasesFields
MainWindow, 136
InitialConfiguration_Tables
MainWindow, 137
initShearButton_BackgroundColor
button.h, 236
initShearButton_size
button.h, 236
initShearPhase
MainWindow, 138
initSocketServer
receiveData, 154
initTestCase
SocketTests, 180
insert_into_table
DBManager, 47
insertData_inDatabase
Experiment, 78
insertIntoTable
DBManager, 35
insertValuesIntoBind_Densification
DBManager, 37
insertValuesIntoBind_Experiment
DBManager, 37
insertValuesIntoBind_FinalVariables
DBManager, 40
insertValuesIntoBind_SampleVariables
DBManager, 41
insertValuesIntoBind_SampleVariablesUpdate
DBManager, 42
insertValuesIntoBind_Shear
DBManager, 42
interface_message
machineServer, 116
sendCommands, 163
interface_payload_size
socket_local.h, 210
interface_to_machine_message, 105
command, 106
distance, 106
enabled, 106
payload, 107
pressure, 107
sample_period, 107
socket_local.h, 211
velocity, 107
isOpen
DBManager, 43
isThreadDestroyed
ThreadController, 195
lineNames
Table, 190
liquid_limit
Experiment, 90
load
machine_to_interface_message, 108
m_axisX
Charts, 29
m_axisY
Charts, 29
machine_message
receiveData, 157
machine_payload_size
socket_local.h, 211
machine_to_interface_message, 107
displacement, 108
load, 108
payload, 108
sample_number, 108

socket_local.h, 211
 state, 109
 machineClient, 109
 ~machineClient, 110
 errorOccurred, 111
 machineClient, 110
 sendMessages, 110
 socket_id, 111
 socket_name, 111
 machineServer, 112
 ~machineServer, 114
 client_socket_id, 116
 errorOccurred, 116
 init, 114
 interface_message, 116
 machineServer, 114
 read_interface_message, 115
 run, 115
 server_socket_id, 116
 server_socket_name, 116
 machineTablevalues
 Table, 190
 main
 main.cpp, 203
 main.cpp
 main, 203
 MainWindow, 117
 ~MainWindow, 121
 adjustVelocity_Distance, 121
 cancelExperiment, 123
 changeDistance, 124
 changeExportOption_Densification, 124
 changeExportOption_Shear, 124
 changeInitialPositionValue, 125
 changeOutsideExperimentPage, 125
 changePage, 126
 changePhase, 127
 changeVelocity, 128
 charts_variables, 144
 chosenTable, 145
 connectButtonsToSlots_Layout, 129
 connectButtonsToSlots_Widget, 129
 connectButtonToSlots_WithArguments, 130
 CreateDatabaseTables, 130
 data_export, 145
 enableExportButton, 131
 enableShearInitButton, 132
 experiment_canceled, 145
 export_option, 145
 fillTextEditForTests, 132
 info_variables, 145
 InitialConfiguration_InsideExperimentButtons, 134
 InitialConfiguration_OutsideExperimentHeaderButtons,
 135
 InitialConfiguration_PhasesButtons, 136
 InitialConfiguration_PhasesFields, 136
 InitialConfiguration_Tables, 137
 initShearPhase, 138
 MainWindow, 120
 my_db, 145
 nextPhase, 139
 on_goBack_toolButton_clicked, 140
 on_initExperiment_toolButton_clicked, 140
 on_releasePressure_toolButton_clicked, 141
 onPositionButton_pressed, 142
 onPositionButton_released, 143
 previousIndex, 145
 receive_data, 145
 send_data, 146
 setupButtons, 146
 setupFields, 146
 tables, 146
 timer, 146
 ui, 146
 updateResultsTables, 143
 moveButton_DisabledBackgroundColor
 button.h, 236
 my_db
 MainWindow, 145
 name
 Experiment, 91
 nextPhase
 MainWindow, 139
 no_icon
 button.h, 236
 on_goBack_toolButton_clicked
 MainWindow, 140
 on_initExperiment_toolButton_clicked
 MainWindow, 140
 on_releasePressure_toolButton_clicked
 MainWindow, 141
 onPositionButton_pressed
 MainWindow, 142
 onPositionButton_released
 MainWindow, 143
 operator_name
 Experiment, 91
 outsideExperiment_buttonSize
 button.h, 236
 payload
 interface_to_machine_message, 107
 machine_to_interface_message, 108
 phase
 Experiment, 91
 phases_buttonSize
 button.h, 237
 phases_continueButtonSize
 button.h, 237
 phases_table
 table.h, 244
 phasesButton_darkBackgroundColor
 button.h, 237
 phasesButton_lightBackgroundColor
 button.h, 237

pi_value
 experiment.h, 226

plastic_limit
 Experiment, 91

present_time_milliseconds
 Experiment, 91

present_time_seconds
 Experiment, 91

pressure
 Experiment, 91

 interface_to_machine_message, 107

pressureButton_GreenBackgroundColor
 button.h, 237

pressureButton_RedBackgroundColor
 button.h, 237

pressureButton_style
 Button, 21

previousIndex
 MainWindow, 145

prova_conceito_database
 DBManager, 47

publicVar
 QTstyle_Test, 150

QTstyle_Test, 147
 ~QTstyle_Test, 148

 enumPtr, 149

 enumVar, 150

 handler, 150

 publicVar, 150

 QTstyle_Test, 148

 TEnum, 148

 testMe, 149

 testMeToo, 149

 TVal1, 148

 TVal2, 148

 TVal3, 148

read_interface_message
 machineServer, 115

readClientMessage
 receiveData, 155

receive_data
 MainWindow, 145

receiveData, 151
 ~receiveData, 154

 client_socket_id, 157

 data_arrived, 154

 errorOccurred, 157

 init, 154

 initSocketServer, 154

 machine_message, 157

 readClientMessage, 155

 receiveData, 153

 run, 156

 server_socket_id, 157

 server_socket_name, 158

 shear_densification_variables, 158

receiveDataFromInterface
 SocketTests, 182

receiveDataFromMachine
 SocketTests, 182

receiveDataThread
 ThreadController, 195

receiveThreadFinishedSlot
 ThreadController, 194

reset_Chart
 Charts, 28

run
 machineServer, 115

 receiveData, 156

sample_description
 Experiment, 91

sample_id
 Experiment, 92

sample_location
 Experiment, 92

sample_number
 Densification, 53

 machine_to_interface_message, 108

sample_number_diff
 Shear, 175

sample_period
 Experiment, 92

 interface_to_machine_message, 107

sample_preparations
 Experiment, 92

sample_table
 dbmanager.h, 217

sampling_period
 sendCommands, 164

selectExperimentId
 DBManager, 43

send_data
 MainWindow, 146

sendCommands, 158
 ~sendCommands, 159

 connectToMachine, 159

 enabled, 163

 errorOccurred, 163

 getDistance, 160

 getEnabled, 160

 getPressure, 160

 getSamplingPeriod, 160

 getVelocity, 160

 interface_message, 163

 sampling_period, 164

 sendCommands, 159

 sendMessage, 160

 setCommand, 161

 setDistance, 161

 setEnabled, 161

 setPressure, 162

 setSamplingPeriod, 162

 setVelocity, 163

 socket_id, 164

 socket_name, 164

sendMessage
 sendCommands, 160
sendMessages
 machineClient, 110
series
 Charts, 29
server_socket_id
 machineServer, 116
 receiveData, 157
server_socket_name
 machineServer, 116
 receiveData, 158
setAshto_class
 Experiment, 78
setBoring_number
 Experiment, 78
setButton_style_icon
 Button, 22
setButtonShadow
 Button, 22
setCommand
 sendCommands, 161
setDate
 Shear, 170
setDensificationVariables
 Densification, 51
setDiameter
 Experiment, 78
setDiff_sampleNumber_initExperiment
 Densification, 52
setDistance
 sendCommands, 161
 Shear, 171
setEnabled
 sendCommands, 161
setExperimentStarted
 Experiment, 79
setHorizontal_displacement
 Shear, 171
setHorizontal_load
 Shear, 171
setHour_min_sec_ms
 Shear, 172
setInitial_height
 Experiment, 79
setInitial_moisture
 Experiment, 79
setInitial_position
 Experiment, 80
setInitial_time
 Experiment, 80
 Shear, 172
setInitial_wet_weight
 Experiment, 80
setLiquid_limit
 Experiment, 81
setName
 Experiment, 81

setOperator_name
 Experiment, 81
setPlastic_limit
 Experiment, 82
setPressure
 Experiment, 82
 sendCommands, 162
setSample_description
 Experiment, 82
setSample_id
 Experiment, 83
setSample_location
 Experiment, 83
setSample_number_diff
 Shear, 172
setSample_period
 Experiment, 83
setSample_preparations
 Experiment, 84
setSamplingPeriod
 sendCommands, 162
setShearVariables
 Shear, 173
setSpecimen_type
 Experiment, 84
setSpgr_solids
 Experiment, 84
setTest_type
 Experiment, 85
setupButtons
 MainWindow, 146
setupFields
 MainWindow, 146
 SocketTests, 183
setUscs_class
 Experiment, 85
setVariables
 Field, 103
setVelocity
 sendCommands, 163
 Shear, 173
Shear, 164
 date, 174
 distance, 174
 getDate, 166
 getDistance, 166
 getHorizontal_displacement, 167
 getHorizontal_load, 167
 getHour_min_sec_ms, 167
 getInitial_time_milliseconds, 168
 getInitial_time_seconds, 168
 getSample_number_diff, 168
 getShearResults, 169
 getShearVariables, 169
 getVelocity, 170
 horizontal_displacement, 174
 horizontal_load, 174
 hour_min_sec_ms, 174

initial_time_milliseconds, 174
initial_time_seconds, 175
sample_number_diff, 175
setDate, 170
setDistance, 171
setHorizontal_displacement, 171
setHorizontal_load, 171
setHour_min_sec_ms, 172
setInitial_time, 172
setSample_number_diff, 172
setShearVariables, 173
setVelocity, 173
Shear, 166
velocity, 175
shear_chart
charts.h, 231
shear_densification_variables
receiveData, 158
shear_phase
experiment.h, 226
shear_result_table
table.h, 244
shear_table
dbmanager.h, 217
shear_variables
Experiment, 92
shearButton_darkIcon
button.h, 237
shearButton_lightIcon
button.h, 237
shearTable
Table, 190
socket_id
machineClient, 111
sendCommands, 164
socket_local.h
interface_payload_size, 210
interface_to_machine_message, 211
machine_payload_size, 211
machine_to_interface_message, 211
socket_name
machineClient, 111
sendCommands, 164
SocketTests, 175
~SocketTests, 179
cleanupTestCase, 179
compareStructFloatElements, 179
compareStructIntElements, 180
info_variables, 182
initTestCase, 180
receiveDataFromInterface, 182
receiveDataFromMachine, 182
setupFields, 183
SocketTests, 178
tables, 183
test_receiveDataFromInterface, 181
test_receiveDataFromInterface_data, 181
test_receiveDataFromMachine, 181
test_receiveDataFromMachine_data, 182
test_receiveDataFromMachine_data, 182
test_sendData, 183
test_sendDataMachine, 183
specimen_type
Experiment, 92
spgr_solids
Experiment, 92
state
machine_to_interface_message, 109
styleSheetConfiguration
Button, 23
Table, 184
clearDynamicTables, 186
clearStaticTables, 186
columnValues, 190
customizeTable, 186
densificationTable, 190
headerNames, 190
initialConfig_DynamicTable, 187
initialConfig_ShearTable, 187
initialConfig_StaticTable, 187
lineNames, 190
machineTablevalues, 190
shearTable, 190
Table, 186
table_variables, 190
test, 187
updateData_DynamicTable, 188
updateData_ShearTable, 188
updateData_StaticTable, 188
table.h
densification_result_table, 243
design_densification_table, 243
design_shear_table, 243
info_table, 244
phases_table, 244
shear_result_table, 244
table_name
DBManager, 47
table_variables
Table, 190
tableExists
DBManager, 44
tables
MainWindow, 146
SocketTests, 183
TEnum
QTstyle_Test, 148
test
Table, 187
test_receiveDataFromInterface
SocketTests, 181
test_receiveDataFromInterface_data
SocketTests, 181
test_receiveDataFromMachine
SocketTests, 181
test_receiveDataFromMachine_data
SocketTests, 182

test_sendData
 SocketTests, 183
test_sendDataMachine
 SocketTests, 183
test_type
 Experiment, 92
testMe
 QTstyle_Test, 149
testMeToo
 QTstyle_Test, 149
tests
 tst_sockettests.cpp, 209
ThreadController, 191
 ~ThreadController, 194
 isThreadDestroyed, 195
 receiveDataThread, 195
 receiveThreadFinishedSlot, 194
 ThreadController, 193
timer
 MainWindow, 146
tst_sockettests.cpp
 tests, 209
TVal1
 QTstyle_Test, 148
TVal2
 QTstyle_Test, 148
TVal3
 QTstyle_Test, 148

Ui, 9
ui
 MainWindow, 146
update_database_table
 DBManager, 45
updateData_DynamicTable
 Table, 188
updateData_ShearTable
 Table, 188
updateData_StaticTable
 Table, 188
updateDensificationChart
 Experiment, 85
updateDensificationCharts
 Charts, 28
updateDensificationResultsTable
 Experiment, 86
updateDensificationTable
 Experiment, 86
updateResultsTables
 MainWindow, 143
updateShearChart
 Experiment, 87
updateShearCharts
 Charts, 28
updateShearResultsTable
 Experiment, 87
updateShearTable
 Experiment, 88
updateTable

DBManager, 46
uscs_class
 Experiment, 93

velocity
 interface_to_machine_message, 107
 Shear, 175
velocityPositionBackgroundColor
 button.h, 238
velocityPositionButton_size
 button.h, 238
vertical_displacement
 Densification, 53
vertical_load
 Densification, 53

write
 exportData, 96

x_axis_title
 Charts, 30

y_axis_title
 Charts, 30