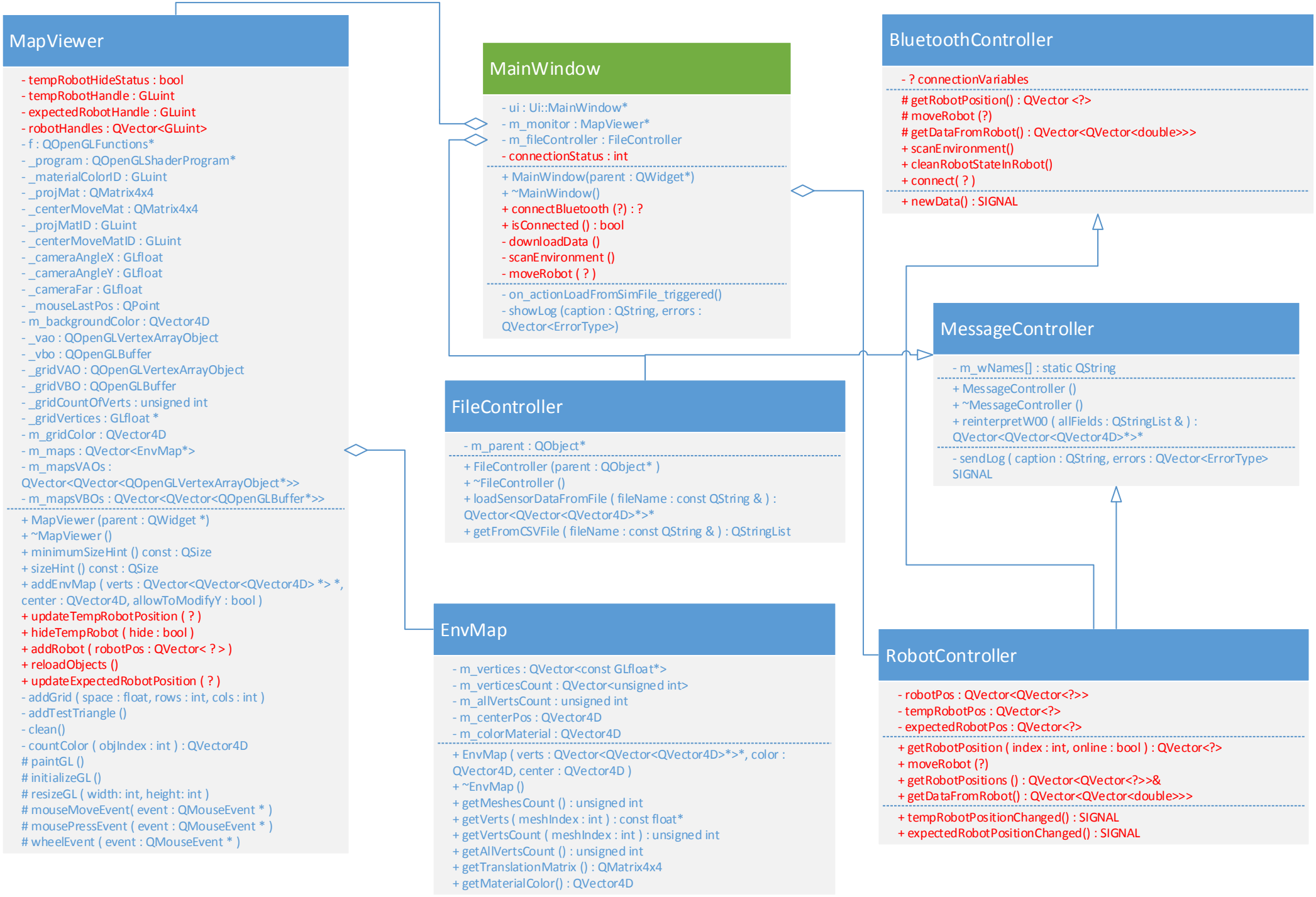
MapViewer
- tempRobotHideStatus : bool
- tempRobotHandle : GLuint
- expectedRobotHandle : GLuint
- robotHandles : QVector<GLuint>
- f : QOpenGLFunctions*
- _program : QOpenGLShaderProgram*
- _materialColorID : GLuint
- _projMat : QMatrix4x4
- _centerMoveMat : QMatrix4x4
- _projMatID : GLuint
- _centerMoveMatID : GLuint
- _cameraAngleX : GLfloat
- _cameraAngleY : GLfloat
- _cameraFar : GLfloat
- _mouseLastPos : QPoint
- m_backgroundColor : QVector4D
- _vao : QOpenGLVertexArrayObject
- _vbo : QOpenGLBuffer
- _gridVAO : QOpenGLVertexArrayObject
- _gridVBO : QOpenGLBuffer
- _gridCountOfVerts : unsigned int
- _gridVertices : GLfloat *
- m_gridColor : QVector4D
- m_maps : QVector<EnvMap*>
- m_mapsVAOs : QVector<QVector<QOpenGLVertexArrayObject*>>
- m_mapsVBOs : QVector<QVector<QOpenGLBuffer*>>
+ MapViewer (parent : QWidget *)
+ ~MapViewer ()
+ minimumSizeHint () const : QSize
+ sizeHint () const : QSize
+ addEnvMap ( verts : QVector<QVector<QVector4D> *> *, center : QVector4D, allowToModifyY : bool )
+ updateTempRobotPosition ( ? )
+ hideTempRobot ( hide : bool )
+ addRobot ( robotPos : QVector< ? > )
+ reloadObjects ()
+ updateExpectedRobotPosition ( ? )
- addGrid ( space : float, rows : int, cols : int )
- addTestTriangle ()
- clean()
- countColor ( objIndex : int ) : QVector4D
# paintGL ()
# initializeGL ()
# resizeGL ( width: int, height: int )
# mouseMoveEvent( event : QMouseEvent * )
# mousePressEvent ( event : QMouseEvent * )
# wheelEvent ( event : QMouseEvent * )
MainWindow
- ui : Ui::MainWindow*
- m_monitor : MapViewer*
- m_fileController : FileController
- connectionStatus : int
+ MainWindow(parent : QWidget*)
+ ~MainWindow()
+ connectBluetooth (?) : ?
+ isConnected () : bool
- downloadData ()
- scanEnvironment ()
- moveRobot ( ? )
- on_actionLoadFromSimFile_triggered()
- showLog (caption : QString, errors : QVector<ErrorType>)
BluetoothController
- ? connectionVariables
# getRobotPosition() : QVector <?>
# moveRobot (?)
# getDataFromRobot() : QVector<QVector<double>>>
+ scanEnvironment()
+ cleanRobotStateInRobot()
+ connect( ? )
+ newData() : SIGNAL
MessageController
- m_wNames[] : static QString
+ MessageController ()
+ ~MessageController ()
+ reinterpretW00 ( allFields : QStringList & ) : QVector<QVector<QVector4D>*>*
- sendLog ( caption : QString, errors : QVector<ErrorType>
SIGNAL
FileController
- m_parent : QObject*
+ FileController (parent : QObject* )
+ ~FileController ()
+ loadSensorDataFromFile ( fileName : const QString & ) : QVector<QVector<QVector4D>*>*
+ getFromCSVFile ( fileName : const QString & ) : QStringList
EnvMap
- m_vertices : QVector<const GLfloat*>
- m_verticesCount : QVector<unsigned int>
- m_allVertsCount : unsigned int
- m_centerPos : QVector4D
- m_colorMaterial : QVector4D
+ EnvMap ( verts : QVector<QVector<QVector4D>*>*, color : QVector4D, center : QVector4D )
+ ~EnvMap ()
+ getMeshesCount () : unsigned int
+ getVerts ( meshIndex : int ) : const float*
+ getVertsCount ( meshIndex : int ) : unsigned int
+ getAllVertsCount () : unsigned int
+ getTranslationMatrix () : QMatrix4x4
+ getMaterialColor() : QVector4D
RobotController
- robotPos : QVector<QVector<?>>
- tempRobotPos : QVector<?>
- expectedRobotPos : QVector<?>
+ getRobotPosition ( index : int, online : bool ) : QVector<?>
+ moveRobot (?)
+ getRobotPositions () : QVector<QVector<?>>&
+ getDataFromRobot() : QVector<QVector<double>>>
+ tempRobotPositionChanged() : SIGNAL
+ expectedRobotPositionChanged() : SIGNAL