

435L Project

Generated by Doxygen 1.8.6

Thu Nov 26 2020 23:50:11

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	Arrow Class Reference	5
3.1.1	Member Function Documentation	6
3.1.1.1	rotate	6
3.1.1.2	shoot	6
3.1.1.3	spacePressed	6
3.2	Board Class Reference	6
3.2.1	Member Function Documentation	6
3.2.1.1	cannotPlay	6
3.2.1.2	setScore	6
3.3	Disk Class Reference	7
3.4	Game1scene Class Reference	7
3.4.1	Member Function Documentation	8
3.4.1.1	addCircle	8
3.4.1.2	addVirus	8
3.4.1.3	cleanPage	9
3.4.1.4	collisionVirusSyring	9
3.4.1.5	connectButtons	9
3.4.1.6	displayScores	9
3.4.1.7	fillScene	9
3.4.1.8	increaseLevel	9
3.4.1.9	setScoreLabels	9
3.4.1.10	startGame	9
3.4.1.11	updateGameName	9
3.4.1.12	updateGameOver	9
3.4.1.13	updateGameWon	9

3.4.1.14	updateUserScores	9
3.4.1.15	userFailed	10
3.4.1.16	userWon	10
3.5	Game2scene Class Reference	10
3.5.1	Member Function Documentation	11
3.5.1.1	cellClicked	11
3.5.1.2	changeToPlayer	11
3.5.1.3	changeTurn	11
3.5.1.4	cleanPage	11
3.5.1.5	fillScene	11
3.5.1.6	fillSceneHelper	11
3.5.1.7	gameOver	11
3.5.1.8	legalMovesCheck	11
3.5.1.9	restartGame	12
3.5.1.10	startGame	12
3.5.1.11	updateBoard	12
3.5.1.12	updateUserScores	12
3.6	Json Class Reference	12
3.6.1	Member Function Documentation	12
3.6.1.1	appendToUserDocument	12
3.6.1.2	checkUser	12
3.6.1.3	EncodeImage	13
3.6.1.4	getJsonDocument	13
3.6.1.5	updateUserScores	13
3.7	Line Class Reference	13
3.7.1	Member Function Documentation	13
3.7.1.1	checkLine	13
3.7.1.2	getChangedTiles	14
3.8	LoginPage Class Reference	14
3.8.1	Member Function Documentation	14
3.8.1.1	fixLabels	14
3.8.1.2	loginUser	15
3.9	MainWindow Class Reference	15
3.9.1	Member Function Documentation	15
3.9.1.1	getJoke	15
3.9.1.2	playMusic	16
3.10	Point Class Reference	16
3.11	RollingBg Class Reference	16
3.11.1	Member Function Documentation	17
3.11.1.1	paintEvent	17

3.12 SignupPage Class Reference	17
3.12.1 Member Function Documentation	18
3.12.1.1 cleanPage	18
3.12.1.2 createUser	18
3.12.1.3 selectImage	18
3.12.1.4 setUser	18
3.13 Strategy Class Reference	18
3.13.1 Member Function Documentation	18
3.13.1.1 play	18
3.14 Syringe Class Reference	19
3.15 User Class Reference	19
3.15.1 Constructor & Destructor Documentation	20
3.15.1.1 User	20
3.15.2 Member Function Documentation	20
3.15.2.1 getScoresAsJson	20
3.15.2.2 isUnique	20
3.15.2.3 isValid	20
3.15.2.4 toJsonObject	20
3.16 Util Class Reference	21
3.16.1 Member Function Documentation	21
3.16.1.1 checkPassword	21
3.16.1.2 hashPassword	21
3.17 VirusLarge Class Reference	21
3.17.1 Member Function Documentation	22
3.17.1.1 collidedWithSyringe	22
3.17.1.2 update	22
3.17.1.3 userFailed	22
3.18 WelcomeWindow Class Reference	23
3.18.1 Member Function Documentation	23
3.18.1.1 checkBday	23
3.18.1.2 cleanPage	23
3.18.1.3 fillScene	23
3.18.1.4 fixLabels	24
3.18.1.5 fixPixmapItems	24
3.18.1.6 fixWidgets	24
3.18.1.7 updateProfilePic	24
3.18.1.8 updateScores	24

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Board	6
Json	12
Line	13
Point	16
QGraphicsPixmapItem	
Arrow	5
Syringe	19
VirusLarge	21
QGraphicsScene	
Game1scene	7
Game2scene	10
LoginPage	14
MainWindow	15
WelcomeWindow	23
QLabel	
Disk	7
QObject	
Arrow	5
Syringe	19
User	19
Util	21
VirusLarge	21
QWidget	
RollingBg	16
SignupPage	17
Strategy	18

Chapter 2

Class Index

2.1 Class List

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

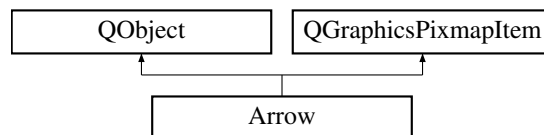
Arrow	5
Board	6
Disk	7
Game1scene	7
Game2scene	10
Json	12
Line	13
LoginPage	14
MainWindow	15
Point	16
RollingBg	16
SignupPage	17
Strategy	18
Syringe	19
User	19
Util	21
VirusLarge	21
WelcomeWindow	23

Chapter 3

Class Documentation

3.1 Arrow Class Reference

Inheritance diagram for Arrow:



Public Slots

- void [rotate](#) ()
- void [shoot](#) ()

Signals

- void **failure** ()

Public Member Functions

- **Arrow** (QObject *parent=nullptr)
- void [spacePressed](#) ()

Public Attributes

- [Syringe](#) * **syringe**
- QString **arrowPicPath** = "../game1images/arrow.png"
- int **direction** = 1
- int **rotationDegree** = 0
- int **timerRotateSpeed** = 70
- QTimer * **timerRotate**
- QTimer * **timerShoot**

3.1.1 Member Function Documentation

3.1.1.1 void Arrow::rotate () [slot]

A function to rotate the arrow

3.1.1.2 void Arrow::shoot () [slot]

A function to shoot the arrow

3.1.1.3 void Arrow::spacePressed ()

Creates a new [Syringe](#) that gets initialized whenever a space is pressed

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

- game1-kill-covid-19/arrow.h
- game1-kill-covid-19/arrow.cpp

3.2 Board Class Reference

Public Member Functions

- void [setScore](#) ()
- QList< [Point](#) > **getChangedTiles** (int player)
- bool [cannotPlay](#) (int player)

Public Attributes

- int **countBlackDisks** = 0
- int **countWhiteDisks** = 0
- const int **playerWhite** = -1
- const int **playerBlack** = 1
- int ** **gameboard**

3.2.1 Member Function Documentation

3.2.1.1 bool Board::cannotPlay (int *player*)

Checks if a [User](#) or the AI can play or not

Returns

True if they can, false if not

3.2.1.2 void Board::setScore ()

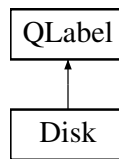
Count the number of black and white disks in the board

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

- game-2-reversi/board.h
- game-2-reversi/board.cpp

3.3 Disk Class Reference

Inheritance diagram for Disk:



Signals

- void **isClicked** (int x, int y)

Public Member Functions

- **Disk** (int x, int y, QWidget *parent=0)

Protected Member Functions

- void **mousePressEvent** (QMouseEvent *event)

Protected Attributes

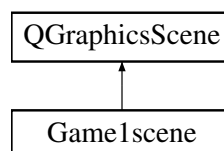
- int **x**
- int **y**

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

- game-2-reversi/disk.h
- game-2-reversi/disk.cpp

3.4 Game1scene Class Reference

Inheritance diagram for Game1scene:



Public Slots

- void [startGame](#) ()
- void [updateGameName](#) ()
- void [updateGameOver](#) ()
- void [updateGameWon](#) ()
- void [collisionVirusSyring](#) ()
- void [userFailed](#) ()

Public Member Functions

- void **fixWidgets** ()
- void **fillScene** ()
- void **setScoreLabels** ()
- void **connectButtons** ()
- void **addCircle** ()
- void **displayScores** ()
- void **addVirus** ()
- void **updateUserScores** ()
- void **increaseLevel** ()
- void **userWon** ()
- void **cleanPage** ()

Public Attributes

- **RollingBg** * **rollingbg**
- **Syringe** * **syringe**
- **Arrow** * **arrow**
- **VirusLarge** * **virusLarge**
- QPushButton * **homeButton**
- QPushButton * **startButton**
- QGraphicsPixmapItem * **gameName**
- QGraphicsPixmapItem * **gameOver**
- QGraphicsPixmapItem * **gameWon**
- QGraphicsPixmapItem * **circle**
- QTimer * **gameNameTimer**
- QTimer * **gameOverTimer**
- QTimer * **gameWonTimer**
- int **gameNamey** = 130
- int **gameOvery** = 130
- int **gameWony** = 130
- **User** * **curUser** = NULL
- QLabel * **highscoreL**
- QLabel * **currentScoreL**
- QLabel * **scoreHistoryL**
- QLabel * **scoreHistory**
- int **highscore** = 0
- int **currentScore** = 0
- int **countLarge** = 0
- int **countMedium** = 0
- int **countSmall** = 0
- int **counter** = 0
- int **levelSpeed** = 50

3.4.1 Member Function Documentation

3.4.1.1 void Game1scene::addCircle ()

Adds the semi-circle that serves as a stand to the arrow

3.4.1.2 void Game1scene::addVirus ()

Creates a new virus and adds it to the game scene

3.4.1.3 void Game1scene::cleanPage ()

Resets the page before admitting a new user

3.4.1.4 void Game1scene::collisionVirusSyring () [slot]

Slot to the collision() signal emitted by the virusLarge class

3.4.1.5 void Game1scene::connectButtons ()

Connects all slots and buttons except the virus related ones since we keep creating a new virus everytime. Check [Game1scene::addVirus\(\)](#) to see the virus related slots.

3.4.1.6 void Game1scene::displayScores ()

updates the score labels on the gameScene

3.4.1.7 void Game1scene::fillScene ()

fixes the buttons and adds them to the game Scene

3.4.1.8 void Game1scene::increaseLevel ()

Called whenever the user hits 5 viruses. increases the speed of rotation of arrows as well as the speed of the viruses (levelSpeed) and rolling background

3.4.1.9 void Game1scene::setScoreLabels ()

Fixes the score labels and adds them to the scene

3.4.1.10 void Game1scene::startGame () [slot]

Resets the current scores and starts a new game

3.4.1.11 void Game1scene::updateGameName () [slot]

Makes gameName label move up and down

3.4.1.12 void Game1scene::updateGameOver () [slot]

Makes gameOver label move up and down

3.4.1.13 void Game1scene::updateGameWon () [slot]

Makes gameWon label move up and down

3.4.1.14 void Game1scene::updateUserScores ()

Edits the current [User](#)'s vector of game1 scores as well as his highscore

3.4.1.15 void Game1scene::userFailed () [slot]

Whenever the arrow goes out of bound or when a virus leaves the boundary without being hit, this function receives the failure() signal and ends the current game

3.4.1.16 void Game1scene::userWon ()

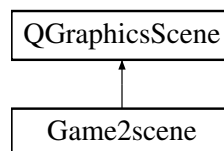
whenever the user reaches a score ≥ 150 this function signals that the user wins and ends the current game

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

- game1-kill-covid-19/game1scene.h
- game1-kill-covid-19/game1scene.cpp

3.5 Game2scene Class Reference

Inheritance diagram for Game2scene:



Public Slots

- void [cellClicked](#) ()
- void [changeTurn](#) ()
- void [restartGame](#) ()

Public Member Functions

- void [startGame](#) ()
- void [changeToPlayer](#) ()
- bool [legalMovesCheck](#) ()
- void [updateBoard](#) ()
- void [gameOver](#) ()
- void [fillScene](#) ()
- void [fillSceneHelper](#) (QLabel *label, int fontSize)
- void [updateUserScores](#) ()
- void [cleanPage](#) ()

Public Attributes

- [User](#) * **curUser** = NULL
- int **highscore** = 0
- [Board](#) * **board**
- QTimer * **timer**
- QGraphicsPixmapItem * **boardPicture**
- QGraphicsPixmapItem * **gameNamePicture**
- QString **whiteImage** = "../game2images/WhiteDisk.png"
- QString **blackImage** = "../game2images/BlackDisk.png"

- QLabel * **row**
- QLabel * **column**
- QLabel * **finalScore**
- QLabel * **gameName**
- QLabel * **rowNumbers**
- QLabel * **colNumbers**
- QLineEdit * **rowEdit**
- QLineEdit * **columnEdit**
- QPushButton * **enterMove**
- QPushButton * **homeButton**
- QPushButton * **restartButton**
- QGraphicsPixmapItem *** **boardGame**
- int **gameStatus** = -1

3.5.1 Member Function Documentation

3.5.1.1 void Game2scene::cellClicked () [slot]

A function that updates the board whenever an input from the user is given.

3.5.1.2 void Game2scene::changeToPlayer ()

A function used to do the logic whenever we change the player's turn

3.5.1.3 void Game2scene::changeTurn () [slot]

A function used to change turns between players

3.5.1.4 void Game2scene::cleanPage ()

A function that cleans the page and remove items from the board when a user chooses to go back to the home page

3.5.1.5 void Game2scene::fillScene ()

A function to fill the scene

3.5.1.6 void Game2scene::fillSceneHelper (QLabel * *label*, int *fontSize*)

A helper function to fill the Scene

3.5.1.7 void Game2scene::gameOver ()

A function used to do the logic when a game finishes.

3.5.1.8 bool Game2scene::legalMovesCheck ()

Checks if the play or AI can make a move

Returns

True if they can, false if not

3.5.1.9 void Game2scene::restartGame () [slot]

A function used to restart the game whenever the button is clicked

3.5.1.10 void Game2scene::startGame ()

A function Used by the constructor to start the game and initalize the board and images

3.5.1.11 void Game2scene::updateBoard ()

Updates the board's pixmaps whenever a turn is finished.

3.5.1.12 void Game2scene::updateUserScores ()

Edits the current [User](#)'s vector of game2 scores as well as his highscore

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

- game-2-reversi/game2scene.h
- game-2-reversi/game2scene.cpp

3.6 Json Class Reference

Public Member Functions

- QJsonDocument [getJsonDocument](#) ()
Gets the JsonDocument of the file path.
- void [appendToUserDocument](#) (QJsonObject user)
- void [updateUserScores](#) (QString username, QVector< int > gameScores, int highscore, int gameNumber)
- QJsonObject [checkUser](#) (QJsonArray &usersArray, QString &username, QString &password)
- QJsonValue [EncodeImage](#) (const QPixmap &p)
- QPixmap [DecodeImage](#) (QJsonValue val)

Public Attributes

- QString **filePath** = "/home/eece435l/Desktop/435L/project-eece435l-game-center/JSON/users.json"
- [Util](#) **util**

3.6.1 Member Function Documentation

3.6.1.1 void Json::appendToUserDocument (QJsonObject user)

Takes a newly created user and appends it to the users.json document

3.6.1.2 QJsonObject Json::checkUser (QJsonArray & usersArray, QString & username, QString & password)

Checks if a user who attempted to login has already signed up before

Returns

If the login was successful, returns the user as a QJsonObject. Else returns an empty QJsonObject

3.6.1.3 QJsonValue Json::EncodeImage (const QPixmap & p)

Takes a picture, encodes it, and returns the corresponding hashed QJsonValue

Returns

QJsonValue for the encoded image

3.6.1.4 QJsonDocument Json::getJsonDocument ()

Gets the JsonDocument of the file path.

Returns

QJsonDocument of the file path

3.6.1.5 void Json::updateUserScores (QString username, QVector< int > gameScores, int highscore, int gameNumber)

Update the [User](#) Scores in the [Json](#) object of the [User](#)

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

- accounts-and-framework/json.h
- accounts-and-framework/json.cpp

3.7 Line Class Reference

Public Member Functions

- QList< [Point](#) > [getChangedTiles](#) ()
- QList< [Point](#) > [checkLine](#) ([Point](#) pattern)
- **Line** (int diskColor, [Point](#) point, int **board)

Public Attributes

- int **diskColor**
- [Point](#) **point**
- int ** **boardgame**

3.7.1 Member Function Documentation

3.7.1.1 QList< Point > Line::checkLine (Point pattern)

A function used by getChangedTiles to see whether a line is affected by the point

Returns

If it is affected, it returns a list of all points affected. If not, it just returns an empty list

3.7.1.2 QList< Point > Line::getChangedTiles ()

A function that get all the changed tiles that are affected by choosing the point attribute

Returns

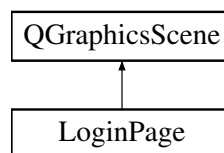
A list of all changed points on the board

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

- game-2-reversi/line.h
- game-2-reversi/line.cpp

3.8 LoginPage Class Reference

Inheritance diagram for LoginPage:



Public Slots

- void [loginUser](#) ()

Public Member Functions

- **LoginPage** (QWidget *parent=nullptr)
- void **fillScene** ()
- void [fixLabels](#) ()

Public Attributes

- [Json](#) **json**
- [Util](#) **util**
- [User](#) * **curUser**
- QLabel * **loginLabel**
- QLabel * **usernameLabel**
- QLabel * **passwordLabel**
- QLabel * **noAccountLabel**
- QLineEdit * **usernameLineEdit**
- QLineEdit * **passwordLineEdit**
- QPushButton * **loginButton**
- QPushButton * **signupButton**
- QPushButton * **homeButton**

3.8.1 Member Function Documentation

3.8.1.1 void LoginPage::fixLabels ()

adjusts the design of the labels (color, background, font, ...)

3.8.1.2 void LoginPage::loginUser () [slot]

Gets called after a user has attempted to login. using the json.cpp utility class:

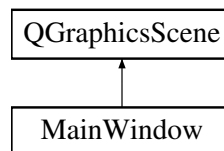
- loads the users.json file using json.getJsonDocument()
- checks if the attempted login was successful using json.checkUser(usersArray,username,password) if successful, the curUser instance gets updated to the currently logged in user else, curUser becomes NULL

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

- accounts-and-framework/loginpage.h
- accounts-and-framework/loginpage.cpp

3.9 MainWindow Class Reference

Inheritance diagram for MainWindow:



Public Member Functions

- void **fixLabels** ()
- void **fillScene** ()
- void **playMusic** ()
- QString **getJoke** ()

Public Attributes

- **Json json**
- QLabel * **helloLabel**
- QLabel * **jokeLabel**
- QLabel * **joke**
- QPushButton * **loginButton**
- QPushButton * **signupButton**
- QPushButton * **guestButton**

3.9.1 Member Function Documentation

3.9.1.1 QString MainWindow::getJoke ()

loads the jokes.json document selects a random joke from it

Returns

QString as a Joke.

3.9.1.2 void MainWindow::playMusic ()

Set the Background music.

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

- accounts-and-framework/mainwindow.h
- accounts-and-framework/mainwindow.cpp

3.10 Point Class Reference

Public Member Functions

- bool **operator!=** ([Point](#) point)
- bool **operator==** ([Point](#) point)
- **Point** (int x, int y)

Public Attributes

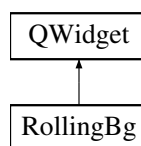
- int **x**
- int **y**

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

- game-2-reversi/point.h
- game-2-reversi/point.cpp

3.11 RollingBg Class Reference

Inheritance diagram for RollingBg:



Public Member Functions

- **RollingBg** (QWidget *parent=nullptr)

Public Attributes

- QTimer * **timer**
- int **timerSpeed** = 50

Protected Member Functions

- void [paintEvent](#) (QPaintEvent *event) override

3.11.1 Member Function Documentation

3.11.1.1 void RollingBg::paintEvent (QPainter * event) [override],[protected]

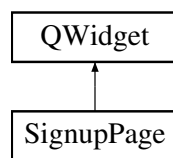
A function to create a QPainter and draw on it

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

- game1-kill-covid-19/rollingbg.h
- game1-kill-covid-19/rollingbg.cpp

3.12 SignupPage Class Reference

Inheritance diagram for SignupPage:



Public Slots

- void [setUser](#) ()
- void [selectImage](#) ()

Public Member Functions

- **SignupPage** (QWidget *parent=nullptr)
- [User](#) * [createUser](#) ()
- void [cleanPage](#) ()

Public Attributes

- [User](#) * **curUser** = NULL
- [Json](#) **json**
- [Util](#) **util**
- QLabel * **Header**
- QLabel * **SignInLabel**
- QLineEdit * **FirstNameEdit**
- QLineEdit * **SecondNameEdit**
- QLineEdit * **UsernameEdit**
- QLineEdit * **PasswordEdit**
- QLineEdit * **ConfirmPasswordEdit**
- QComboBox * **GenderBox**
- QSpinBox * **AgeBox**
- QSpinBox * **DayBox**
- QSpinBox * **MonthBox**
- QSpinBox * **YearBox**
- QPushButton * **SignUpButton**
- QPushButton * **SignInButton**
- QPushButton * **SelectImage**

- QVBoxLayout * **BoxLayout**
- QFormLayout * **FormLayout**
- QHBoxLayout * **DateLayout**
- QHBoxLayout * **SignInLayout**
- QGroupBox * **GroupBox**
- QString **file_name**

3.12.1 Member Function Documentation

3.12.1.1 void SignupPage::cleanPage ()

- this methods resets all the widgets that took user input
- Gets called whenever a user wants to signup

3.12.1.2 User * SignupPage::createUser ()

called from the [setUser\(\)](#) SLOT Reads the input from the widgets and attempts to create a new user

Returns

if successful, returns the new user (not yet added to users.json) else, returns NULL

3.12.1.3 void SignupPage::selectImage () [slot]

Takes profile picture file path from user updates the corresponding class member

3.12.1.4 void SignupPage::setUser () [slot]

Called whenever the signup button is pressed calls [createUser\(\)](#) in order to check all necessary conditions before adding a new user to the users.json file if [createUser\(\)](#) returned a user, [setUser\(\)](#) appends it to users.json

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

- accounts-and-framework/signuppage.h
- accounts-and-framework/signuppage.cpp

3.13 Strategy Class Reference

Public Member Functions

- **Strategy** (int player)
- void [play](#) ([Board](#) *board)

3.13.1 Member Function Documentation

3.13.1.1 void Strategy::play (Board * board)

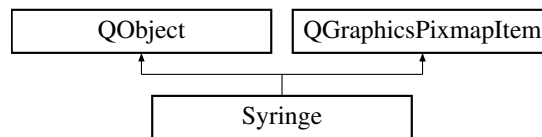
A funciton used by the AI to play its strategy

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

- game-2-reversi/strategy.h
- game-2-reversi/strategy.cpp

3.14 Syringe Class Reference

Inheritance diagram for Syringe:



Public Member Functions

- **Syringe** (QObject *parent=nullptr)

Public Attributes

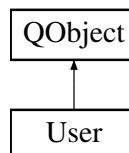
- QString **syringePicPath** = "../game1images/syringe.png"

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

- game1-kill-covid-19/syringe.h
- game1-kill-covid-19/syringe.cpp

3.15 User Class Reference

Inheritance diagram for User:



Public Member Functions

- **User** (QObject *parent=nullptr)
- **User** (JsonObject userJson)
- void **read** (const JsonObject &json)
- void **write** (JsonObject &json) const
- bool **isUnique** ()
- bool **isValid** ()
- JsonObject **toJsonObject** ()
- QJsonArray **getScoresAsJson** (QVector< int > &scores)

Public Attributes

- **Json json**
- QString **firstName**
- QString **lastName**
- QString **gender**

- QString **username**
- QString **dateOfBirth**
- QJsonValue **picture**
- int **hashedpassword**
- int **age**
- int **game1_highest** = 0
- int **game2_highest** = 0
- QVector< int > **game1_scores** = {0,0,0,0,0}
- QVector< int > **game2_scores** = {0,0,0,0,0}

3.15.1 Constructor & Destructor Documentation

3.15.1.1 User::User (QJsonObject *result*) [explicit]

Gets the [User](#) from a QJsonObject

Returns

a user from the users.json

3.15.2 Member Function Documentation

3.15.2.1 QJsonArray User::getScoresAsJson (QVector< int > & *scores*)

Transforms a vector of scores to QJsonArray

Returns

QJsonArray of scores

3.15.2.2 bool User::isUnique ()

Checks whether a [User](#) is unique or not

Returns

True if unique, False otherwise.

3.15.2.3 bool User::isValid ()

Checks whether [User](#)'s input is valid

Returns

true if valid, false otherwise.

3.15.2.4 QJsonObject User::toJsonObject ()

Transforms a [User](#) to a QJsonObject

Returns

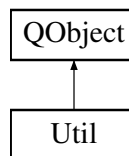
Corresponding QJsonObject

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

- accounts-and-framework/user.h
- accounts-and-framework/user.cpp

3.16 Util Class Reference

Inheritance diagram for Util:

**Public Member Functions**

- **Util** (QObject *parent=nullptr)
- int [hashPassword](#) (QString password)
- bool [checkPassword](#) (QString password)

3.16.1 Member Function Documentation

3.16.1.1 bool Util::checkPassword (QString password)

Checks if a password is valid, of size at least 4 and have special chars /return True if valid, false otherwise.

3.16.1.2 int Util::hashPassword (QString password)

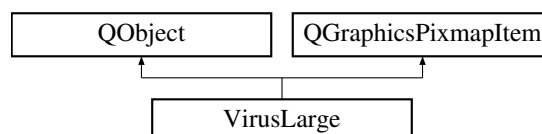
Hashes a password using a hashfunction /return Hashed password as int

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

- accounts-and-framework/util.h
- accounts-and-framework/util.cpp

3.17 VirusLarge Class Reference

Inheritance diagram for VirusLarge:



Public Slots

- void [update](#) ()

Signals

- void **collision** ()
- void **failure** ()

Public Member Functions

- **VirusLarge** (QObject *parent=nullptr, int levelSpeed=50)
- void [collidedWithSyringe](#) ()
- void [userFailed](#) ()

Public Attributes

- QString **picturePath** = ":/game1images/virus-green.png"
- QString **smashedPicPath** = ":/game1images/mike.png"
- bool **smashed** = false
- int **x**
- int **y**
- int **virusType** = 1
- int **virusScore** = 0
- int **timerSpeed** = 50
- QTimer * **timer**

3.17.1 Member Function Documentation

3.17.1.1 void VirusLarge::collidedWithSyringe ()

A function that emits a signal whenever a collision happens with a virus

3.17.1.2 void VirusLarge::update () [slot]

A function that updates the coordinates of a virus

3.17.1.3 void VirusLarge::userFailed ()

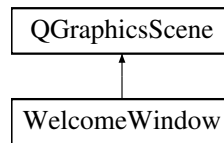
emit a signal whenever the syringe fails to hit the Virus - Game Over!

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

- game1-kill-covid-19/viruslarge.h
- game1-kill-covid-19/viruslarge.cpp

3.18 WelcomeWindow Class Reference

Inheritance diagram for WelcomeWindow:



Public Member Functions

- **WelcomeWindow** (QObject *parent=nullptr)
- void [fixPixmapItems](#) ()
- void [fixWidgets](#) ()
- void [fixLabels](#) ()
- void [fillScene](#) ()
- void [checkBday](#) ()
- void [updateProfilePic](#) ()
- void [updateScores](#) ()
- void [cleanPage](#) ()

Public Attributes

- [User](#) * **curUser** = NULL
- [Json](#) **json**
- QLabel * **helloLabel**
- QLabel * **happyBirthday**
- QGraphicsPixmapItem * **game1Pic**
- QGraphicsPixmapItem * **game2Pic**
- QGraphicsPixmapItem * **profilePicture**
- QPushButton * **game1Button**
- QPushButton * **game2Button**
- QPushButton * **homeButton**
- QLabel * **game1Scores**
- QLabel * **game2Scores**

3.18.1 Member Function Documentation

3.18.1.1 void WelcomeWindow::checkBday ()

checks if its the current user's birthday. if so, displays a happy birthday message

3.18.1.2 void WelcomeWindow::cleanPage ()

Called when we need to go to the maingview Cleans all widgets in order to prepare for another user to login/signup

3.18.1.3 void WelcomeWindow::fillScene ()

Function Used to fill the Scene - For readability

3.18.1.4 void WelcomeWindow::fixLabels ()

Function used to fix Labels - For readability

3.18.1.5 void WelcomeWindow::fixPixmapItems ()

Sets the icons of the games in their corresponding place on the scene

3.18.1.6 void WelcomeWindow::fixWidgets ()

Sets the geometry of the widgets

3.18.1.7 void WelcomeWindow::updateProfilePic ()

Decodes a user's profile picture from a QJsonValue into a QPixmap sets the QPixmap p to the corresponding profile pic location on the scene

3.18.1.8 void WelcomeWindow::updateScores ()

Displays a user's scores to the scene for each corresponding game

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

- accounts-and-framework/welcomewindow.h
- accounts-and-framework/welcomewindow.cpp