My Project

Generated by Doxygen 1.8.6

Sun Dec 9 2018 18:38:00

Contents

1	EEC	E435L (Game		1
2	Hier	archica	l Index		3
	2.1	Class I	Hierarchy		3
3	Clas	s Index			5
	3.1	Class I	List		5
4	File	Index			7
	4.1	File Lis	st		7
5	Clas	s Docu	mentation		11
	5.1	boat C	lass Refere	ence	11
	5.2	Bug Cl	ass Refere	ence	11
	5.3	Bullet (Class Refe	rence	12
	5.4	Coffee	Cup Class	Reference	13
	5.5	game1	scene Cla	ss Reference	14
		5.5.1	Construc	tor & Destructor Documentation	15
			5.5.1.1	game1scene	15
		5.5.2	Member	Function Documentation	15
			5.5.2.1	getLevel	15
			5.5.2.2	profileParser	15
			5.5.2.3	setUpCountdownTimer	16
			5.5.2.4	startLevel	16
			5.5.2.5	updateCountdownTimer	16
			5.5.2.6	updatePosition1	16
			5.5.2.7	updatePosition2	16
			5.5.2.8	updatePosition3	16
			5.5.2.9	updatePosition4	16
			5.5.2.10	updatePosition5	16
			5.5.2.11	updatePosition6	16
			5.5.2.12	updatePosition7	17
			5.5.2.13	updatePosition8	17

iv CONTENTS

5.6	Game2	Scene Cla	ass Reference	17
	5.6.1	Member F	Function Documentation	19
		5.6.1.1	getHighScore	19
		5.6.1.2	updateHighScore	19
	5.6.2	Member [Data Documentation	19
		5.6.2.1	scoreL	19
5.7	gameO	over Class	Reference	19
5.8	games'	Widget Cla	ass Reference	20
	5.8.1	Member F	Function Documentation	21
		5.8.1.1	startGame1	21
5.9	hints C	lass Refere	ence	21
5.10	instruct	tion Class I	Reference	21
5.11	LevelPa	arser Class	s Reference	22
5.12	levels (Class Refer	rence	22
	5.12.1	Member F	Function Documentation	23
		5.12.1.1	getScore	23
		5.12.1.2	incrementHintNumber	24
		5.12.1.3	profileParser	24
		5.12.1.4	resetHintNumber	24
		5.12.1.5	resetLives	24
		5.12.1.6	setPopeyeX1	24
		5.12.1.7	updateLevel	24
		5.12.1.8	updateLevelOne	24
		5.12.1.9	updateScore	24
5.13	levelss	cene Class	Reference	24
	5.13.1	Construct	tor & Destructor Documentation	26
		5.13.1.1	levelsscene	26
	5.13.2	Member F	Function Documentation	26
		5.13.2.1	checkAnswer	26
		5.13.2.2	hideScene	27
		5.13.2.3	pauseLevel	27
		5.13.2.4	retryLevel	27
		5.13.2.5	setUpCountdownTimer	27
		5.13.2.6	updateCountdownTimer	27
		5.13.2.7	youLost	27
		5.13.2.8	youWon	27
5.14	LifeCou	unter Class	Reference	28
5.15	locks C	lass Refer	rence	28
5.16	loggInV	Widget Clas	ss Reference	28
5.17	LogOn'	Widget Cla	ass Reference	29

CONTENTS

5.10	lost Class Reference	30
	5.18.1 Constructor & Destructor Documentation	30
	5.18.1.1 lost	30
	5.18.1.2 lost	30
5.19	miniBug Class Reference	31
5.20	Popeye Class Reference	31
5.21	QualityControllcon Class Reference	32
5.22	river Class Reference	32
5.23	riverObstacle Class Reference	33
5.24	rock Class Reference	33
5.25	Shield Class Reference	34
5.26	signInWidget Class Reference	34
5.27	SignUpWidget Class Reference	35
5.28	smallRiver Class Reference	36
5.29	spinach Class Reference	36
5.30	Tester Class Reference	37
5.31	TestingIcon Class Reference	37
5.32	Wall Class Reference	38
5.33	Won Class Reference	38
	5.33.1 Constructor & Destructor Documentation	39
	5.33.1.1 Won	39
File I	Documentation (1997)	41
6.1	boat.cpp File Reference	
	and the second s	41
	6.1.1 Detailed Description	41 41
6.2	•	
6.2	boat.h File Reference	41
6.2	boat.h File Reference	41 41
	boat.h File Reference	41 41 41
	boat.h File Reference	41 41 41 41
6.3	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description	41 41 41 41
6.3	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference	41 41 41 42 42
6.3	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description	411 411 411 422 422
6.3	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference	411 411 411 422 422 422
6.36.46.5	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference 6.5.1 Detailed Description	41 41 41 42 42 42 42 42
6.36.46.5	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference 6.5.1 Detailed Description bullet.h File Reference	411 411 412 422 422 423 423
6.36.46.56.6	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference 6.5.1 Detailed Description bullet.h File Reference 6.6.1 Detailed Description	411 411 412 422 422 423 433 433
6.36.46.56.6	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference 6.5.1 Detailed Description bullet.h File Reference 6.6.1 Detailed Description coffeecup.cpp File Reference	41 41 41 42 42 42 42 43 43 43
6.3 6.4 6.5 6.6	boat.h File Reference 6.2.1 Detailed Description bug.cpp File Reference 6.3.1 Detailed Description bug.h File Reference 6.4.1 Detailed Description bullet.cpp File Reference 6.5.1 Detailed Description bullet.h File Reference 6.6.1 Detailed Description coffeecup.cpp File Reference 6.7.1 Detailed Description	411 411 412 422 422 423 433 433 433
	5.20 5.21 5.22 5.23 5.24 5.25 5.26 5.27 5.28 5.29 5.30 5.31 5.32 5.33	5.18.1.2 lost 5.19 miniBug Class Reference 5.20 Popeye Class Reference 5.21 QualityControllcon Class Reference 5.22 river Class Reference 5.23 riverObstacle Class Reference 5.24 rock Class Reference 5.25 Shield Class Reference 5.26 signInWidget Class Reference 5.27 SignUpWidget Class Reference 5.28 smallRiver Class Reference 5.29 spinach Class Reference 5.30 Tester Class Reference 5.31 TestingIcon Class Reference 5.32 Wall Class Reference 5.33 Won Class Reference 5.33 Won Class Reference 5.31 Constructor & Destructor Documentation 5.33.1.1 Won

vi CONTENTS

	6.9.1 Detailed Description	44
6.10	game1scene.h File Reference	44
	6.10.1 Detailed Description	44
6.11	game2scene.cpp File Reference	45
	6.11.1 Detailed Description	45
6.12	game2scene.h File Reference	45
	6.12.1 Detailed Description	45
6.13	gameswidget.cpp File Reference	46
	6.13.1 Detailed Description	46
6.14	hints.cpp File Reference	46
	6.14.1 Detailed Description	46
6.15	hints.h File Reference	46
	6.15.1 Detailed Description	47
6.16	instruction.cpp File Reference	47
	6.16.1 Detailed Description	47
6.17	instruction.h File Reference	47
	6.17.1 Detailed Description	47
6.18	levelparser.cpp File Reference	47
	6.18.1 Detailed Description	48
6.19	levelparser.h File Reference	48
	6.19.1 Detailed Description	48
6.20	levels.cpp File Reference	48
	6.20.1 Detailed Description	48
6.21	levels.h File Reference	48
	6.21.1 Detailed Description	49
6.22	levelsscene.cpp File Reference	49
	6.22.1 Detailed Description	49
6.23	lifecounter.cpp File Reference	49
	6.23.1 Detailed Description	49
6.24	lifecounter.h File Reference	49
	6.24.1 Detailed Description	50
6.25	••	50
	6.25.1 Detailed Description	50
6.26	locks.h File Reference	50
	6.26.1 Detailed Description	50
6.27	lost.cpp File Reference	51
	6.27.1 Detailed Description	51
6.28	lost.h File Reference	51
	6.28.1 Detailed Description	51
6.29	minibug.cpp File Reference	51

CONTENTS vii

	6.29.1 Detailed Description	51
6.30	popeye.cpp File Reference	51
	6.30.1 Detailed Description	52
6.31	popeye.h File Reference	52
	6.31.1 Detailed Description	52
6.32	qualitycontrolicon.cpp File Reference	52
	6.32.1 Detailed Description	52
6.33	qualitycontrolicon.h File Reference	52
	6.33.1 Detailed Description	53
6.34	river.cpp File Reference	53
	6.34.1 Detailed Description	53
6.35	river.h File Reference	53
	6.35.1 Detailed Description	53
6.36	riverobstacle.cpp File Reference	53
	6.36.1 Detailed Description	54
6.37	riverobstacle.h File Reference	54
	6.37.1 Detailed Description	54
6.38	rock.cpp File Reference	54
	6.38.1 Detailed Description	54
6.39	rock.h File Reference	54
	6.39.1 Detailed Description	55
6.40	shield.cpp File Reference	55
	6.40.1 Detailed Description	55
6.41	shield.h File Reference	55
	6.41.1 Detailed Description	55
6.42	signinwidget.h File Reference	56
	6.42.1 Detailed Description	56
6.43	signupwidget.h File Reference	56
	6.43.1 Detailed Description	56
6.44	smallriver.cpp File Reference	56
	6.44.1 Detailed Description	57
6.45	smallriver.h File Reference	57
	6.45.1 Detailed Description	57
6.46	spinach.cpp File Reference	57
	6.46.1 Detailed Description	57
6.47	spinach.h File Reference	57
	6.47.1 Detailed Description	58
6.48	tester.cpp File Reference	58
	6.48.1 Detailed Description	58
6.49	tester.h File Reference	58

viii CONTENTS

	6.49.1 Detailed Description	58
6.50	testingicon.cpp File Reference	59
	6.50.1 Detailed Description	59
6.51	testingicon.h File Reference	59
	6.51.1 Detailed Description	59
6.52	won.cpp File Reference	59
	6.52.1 Detailed Description	59
6.53	won.h File Reference	60
	6.53.1 Detailed Description	60

Chapter 1

EECE435L Game

A game containing two sub-games designed to help kids learn programming

Author

Ali Al Akbar Haidoura Camille farhat

Date

9-12-2018

2 EECE435L Game

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

LevelParser	22
levels	22
QGraphicsPixmapItem	
boat	11
Bug	11
Bullet	12
CoffeeCup	13
LifeCounter	28
locks	28
miniBug	31
Popeye	31
QualityControllcon	32
river	32
riverObstacle	33
rock	33
Shield	34
smallRiver	36
spinach	36
Tester	37
Testinglcon	37
Wall	38
QGraphicsScene	
game1scene	14
Game2Scene	17
levelsscene	24
QObject	
boat	11
Bug	11
Bullet	12
locks	28
miniBug	31
Popeye	31
river	
riverObstacle	
rock	
smallRiver	
spinach	
Tester	

4 Hierarchical Index

Testinglcon																				 		37
QWidget																						
gameOver .																						
gamesWidge	et																 			 		20
hints																	 			 		21
instruction .																						
loggInWidget	t																 			 		28
LogOnWidge	et																 			 		29
lost																	 			 		30
signInWidget																						
SignUpWidge	et																 			 		35
Won																						38

Chapter 3

Class Index

3.1 Class List

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

boat	11
Bug	11
Bullet	12
CoffeeCup	13
game1scene	14
Game2Scene	17
gameOver	19
	20
	21
instruction	21
LevelParser	22
levels	22
	24
	28
	28
	28
	29
	30
	31
	31
QualityControllcon	32
	32
	33
	33
	34
	34
	35
	36
	36
	37
	37
	38
Won	38

6 Class Index

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

boat.cpp		
	Contains Popeye class definition	41
boat.h	Doct close definition	4 -
bug.cpp	Boat class definition	41
bug.cpp	Bug class definitions	41
bug.h		
	QGraphicsPixmapItem representing the bugs	42
bullet.cp	p	
	Bullet class definitions	42
bullet.h		
	QGraphicsPixmapItem representing the bullets	43
coffeecu	• • • •	
	Contains CoffeeCup Class definitions	43
coffeecu		
	QGraphicsPixmapItem representing the Coffee Cup	43
game1so		
	Game1's class definition	44
game1so		
_	The Game's scene	44
game2so		
	Contains Game2Scene Class definitions	45
game2so		
	The scene in which all the logic and items of Game2 are located	45
_	er.h	??
gamesw	idget.cpp	40
	Game1-2 Links	46
•	ridget.h	??
hints.cpp		40
hints.h	Shows the appropriate hints in a pop up window	46
1111113.11	The Level's scene's Hints	46
instruction		+0
ii isti uotio	Shows the initial instructions to start the game in a pop up window	47
instruction		F /
oti dotic	The Level's scene's levels pop up used for displaying instructions	47
levelpars		.,
- 1 - 1p - 31 - 0	Contains LevelParser Class definitions	47

8 File Index

levelparser.h	
Parses the game2 levels from the text files	. 48
levels.cpp	
Setting up the different specs for each level to load them dynamically everytime a user starts a	
level	. 48
levels.h	
The Level Object	. 48
levelsscene.cpp	
Contains Game1's class definition	. 49
levelsscene.h	. ??
lifecounter.cpp	
Contains LifeCounter Class definitions	. 49
lifecounter.h	
QGraphicsPixmapItem representing	. 49
locks.cpp	
Contains Lock class definition	. 50
locks.h	
Contains Lock class definition	. 50
logginwidget.h	
logonwidget.h	
lost.cpp	•••
Shows the appropriate loose message in a pop up window	. 51
lost.h	. 51
The Level's scene's loose pop up	. 51
···	. 31
minibug.cpp	E4
Contains miniBug Class definitions	
minibug.h	. ??
popeye.cpp	
Contains Popeye class definition	. 51
popeye.h	
Contains Popeye class definition	. 52
qualitycontrolicon.cpp	
Contains QualityControllcon Class definitions	. 52
qualitycontrolicon.h	
QGraphicsPixmapItem representing the quality control icon	. 52
river.cpp	
Contains river class definition	. 53
river.h	
Contains Popeye class definition	. 53
riverobstacle.cpp	
Contains River Obstacle class definition	. 53
riverobstacle.h	
Contains River Obstacle class definition	. 54
rock.cpp	
Rock class definition	. 54
rock.h	
Rock class definition	. 54
	. 54
shield.cpp Contains shield Class definitions	. 55
	. 55
Shield.h	
QGraphicsPixmapItem representing the shield icon	. 55
signinwidget.h	
User sign in widget	. 56
signupwidget.h	
User sign up widget	. 56
smallriver.cpp	
Small River class definition	. 56

4.1 File List

smallriver.	h h	
9	Small River class definition	
spinach.cp	ор	
(Contains Spinach class definition	
spinach.h		
(Contains Spinach class definition	
tester.cpp		
(Contains tester Class definitions	
tester.h		
(QGraphicsPixmapItem representing the Tester character	
testingicor	n.cpp	
(Contains testinglcon Class definitions	
testingicor	n.h	
(QGraphicsPixmapItem representing the testingIcon	
wall.h .		
won.cpp		
(Shows the appropriate win message in a pop up window	
won.h		
9	Shows the appropriate win message in a pop up window	

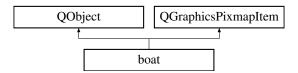
10 File Index

Chapter 5

Class Documentation

5.1 boat Class Reference

Inheritance diagram for boat:



Public Member Functions

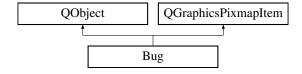
boat (QObject *parent=nullptr)
 Setting Popeye's Image.

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

- · boat.h
- boat.cpp

5.2 Bug Class Reference

Inheritance diagram for Bug:



Public Slots

• void guard ()

The slot that moves the bug on each timer timeout.

• void shoot ()

Responsible for the shooting logic of the Bug.

Public Member Functions

• Bug (Game2Scene *)

Bug constructor.

• void decrementLives ()

Responsible for collision logic with bullet.

· void pause ()

Responsible for keeping the bug in place when pausing the game.

• void resume ()

responsible for resuming after pausing

Public Attributes

• Game2Scene * scene

The scene where the bug is added.

• QTimer * timer

QTimer for moving the Bug periodically.

QTimer * shootingTimer

QTimer for shooting miniBugs.

• QPixmap * icon

QPixmap holding the image of the Bug.

- · QTransform transform
- int lives

Integer holding the number of Bug's lives left.

• int dir

Integer holding the direction of movement of the Bug.

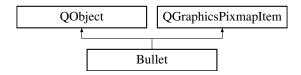
• int **i**

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

- bug.h
- bug.cpp

5.3 Bullet Class Reference

Inheritance diagram for Bullet:



Public Slots

• void move ()

The slot that moves the Bullet on the timer's timeout.

Public Member Functions

Bullet (int direction, int x, int y, Game2Scene *scene)
 Bullet Constructor.

Public Attributes

• QPixmap * icon

QPixmap holding the image of the bullet.

• QTimer * timer

QTimer that trigger s==s.

• Game2Scene * scene

The Scene where the Bullet is.

• int step

A steps counter used for deleting the bullet at a certain range.

· int direction

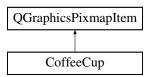
Integer holding the direction of the Bullet according to the last step the tester made.

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

- bullet.h
- bullet.cpp

5.4 CoffeeCup Class Reference

Inheritance diagram for CoffeeCup:



Public Member Functions

• CoffeeCup ()

CoffeeCup Constructor.

Public Attributes

• QPixmap * icon

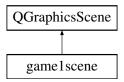
QPixmap holding the image of the CoffeeCup.

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

- coffeecup.h
- · coffeecup.cpp

5.5 game1scene Class Reference

Inheritance diagram for game1scene:



Public Slots

• void startLevel ()

start the appropriate level

void updateCountdownTimer ()

Function that will update the timer and then display it.

Public Member Functions

• game1scene (QString user)

Default constructor.

void updatePosition1 ()

update the position of popeye, placing him on top of the locks for level 1

• void updatePosition2 ()

update the position of popeye, placing him on top of the locks for level 2

• void updatePosition3 ()

update the position of popeye, placing him on top of the locks for level 3

void updatePosition4 ()

update the position of popeye, placing him on top of the locks for level 4

• void updatePosition5 ()

update the position of popeye, placing him on top of the locks for level 5

• void updatePosition6 ()

update the position of popeye, placing him on top of the locks for level 6

• void updatePosition7 ()

update the position of popeye, placing him on top of the locks for level 7

· void updatePosition8 ()

update the position of popeye, placing him on top of the locks for level 8

• void hideLevelScene ()

update the position of popeye, placing him on top of the locks for level 1

• int getLevel ()

get the user's level from the text file

QStringList profileParser (QString line)

get a list of string when parsing from text file the attributes that are seprated by spaces

void setUpCountdownTimer ()

< Timer controlling the countdown

Public Attributes

• Popeye * popeye = new Popeye()

Creating the references to the Objects.

- locks * lock1 = new locks()
- locks * lock2 = new locks()
- locks * lock3 = new locks()
- locks * lock4 = new locks()
- locks * lock5 = new locks()
- locks * lock6 = new locks()
- locks * lock7 = new locks()
- locks * lock8 = new locks()
- QPushButton * start
- · QString user
- levelsscene * scene2

pointer to an Object of type levelsscene

QGraphicsView * view2

pointer to an Object of type QGraphicsView

- QGraphicsTextItem * countDownText
- int **countdown** = 12000
- QTimer * countdownTimer

< value of the countdown at the start of the game that is = to 120 i.e. 20 mins to reach Olive

5.5.1 Constructor & Destructor Documentation

5.5.1.1 game1scene::game1scene (QString user)

Default constructor.

Setting the initial scene and object's positions.

5.5.2 Member Function Documentation

5.5.2.1 int game1scene::getLevel ()

get the user's level from the text file

Getting the level number from the appropriate user text file (stored as the 10th entry)

Parameters

event only argument, key press

Returns

int that is the Level Number

< to check if it is entering the file, and it is

5.5.2.2 QStringList game1scene::profileParser (QString line)

get a list of string when parsing from text file the attributes that are seprated by spaces

Parsing the text file that was filled and has his attributes seperated by tabs.

Returns

List of strings

```
void game1scene::setUpCountdownTimer ( )
< Timer controlling the countdown
Initializes TextItem that indicates the Countdown at the start of each level.
Setting up the timer
5.5.2.4 void game1scene::startLevel() [slot]
start the appropriate level
Starting the corresponsding level by calling the levelsscene constructure and updating the lock's positions accord-
ingly. < pointer to an Object of type levelsscene
< pointer to an Object of type QGraphicsView
5.5.2.5 void game1scene::updateCountdownTimer() [slot]
Function that will update the timer and then display it.
update SLOT that controls the Countdown to display it when adjusted, every second
5.5.2.6 void game1scene::updatePosition1 ( )
update the position of popeye, placing him on top of the locks for level 1
Function Updating Popeye's position and hidding locks, according to the level Users' in.
5.5.2.7 void game1scene::updatePosition2 ( )
update the position of popeye, placing him on top of the locks for level 2
Function Updating Popeye's position and hidding locks, according to the level Users' in.
5.5.2.8 void game1scene::updatePosition3 ( )
update the position of popeye, placing him on top of the locks for level 3
Function Updating Popeye's position and hidding locks, according to the level Users' in.
5.5.2.9 void game1scene::updatePosition4 ( )
update the position of popeye, placing him on top of the locks for level 4
Function Updating Popeye's position and hidding locks, according to the level Users' in.
5.5.2.10 void game1scene::updatePosition5 ( )
update the position of popeye, placing him on top of the locks for level 5
Function Updating Popeye's position and hidding locks, according to the level Users' in.
5.5.2.11 void game1scene::updatePosition6 ( )
update the position of popeye, placing him on top of the locks for level 6
```

Function Updating Popeye's position and hidding locks, according to the level Users' in.

5.5.2.12 void game1scene::updatePosition7 ()

update the position of popeye, placing him on top of the locks for level 7

Function Updating Popeye's position and hidding locks, according to the level Users' in.

5.5.2.13 void game1scene::updatePosition8 ()

update the position of popeye, placing him on top of the locks for level 8

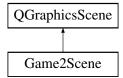
Function Updating Popeye's position and hidding locks, according to the level Users' in.

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

- · game1scene.h
- game1scene.cpp

5.6 Game2Scene Class Reference

Inheritance diagram for Game2Scene:



Public Slots

• void deactivateShield ()

Responsible of deactivating the shield on shieldTimer timeout.

• void updateTimer ()

Responsible of updating remSec on every timer timeout.

void pauseOrResume ()

Responsible of pausing or resuming the game on pause PushButton press.

• void startLevel ()

Responsible of starting a new level after passing the previous one.

• void logScore ()

Responsible of logging the score to file after winning the game.

• void retry ()

Responsible of restarting the game after losing.

Public Member Functions

· Game2Scene (QString user)

Constructor.

void updateLifeScore ()

Responsible of updating the tester's lives when colliding with a bug or a miniBug.

• int getHighScore ()

Responsible of fetching the HighScore from the profile file.

void updateHighScore (QString user)

Responsible of updating the HighScore in the profile file if needed.

Public Attributes

• int dir

Integer holding the direction of the tester.

· int ammo

Integer holding the amount of ammo left.

int bugs

Integer holding the number of bugs left on the scene.

· int remSec

Integer holding the number of seconds left.

· int HighScore

Integer holding the HighScore of the Player.

· int score

Integer holding the current score of the player.

int level

Integer holding the level the player is currently playing.

bool playing

Boolean indicating if the player is active or not.

· bool hasShield

Boolean indicating if the tester has an active shiekd.

· bool QCshown

Boolean indicating if the Quality Control icon is hidden or shown.

· bool paused

Boolean indicating if the game is paused.

QList< Bug * > bugList

QList of Bug objects pointers.

• QLabel * announcement

QLabel holding the win/loss announcements.

QLabel * timerLabel

QLabel holding the string Timer.

QLabel * ammoLabel

QLabel holding the word Ammo.

QLabel * livesLabel

QLabel holding the word Lives.

- QLabel * HiScore
- QLabel * scoreL

QLabel holding the word HighScore.

• QPushButton * next

QPushButton used to continue when level finishes;.

QPushButton * pause

QPsuhButton used to pause/resume the.

• LifeCounter * lifeCounter

Q GraphicsPixmapItem representing the number of lives of the tester.

• LevelParser * parser

Used to parse the levels from text files.

QString user

QString holding the current user.

· Tester * tester

Q GraphicsPixmapItem representing the tester character.

QualityControllcon * QClcon

QGraphicsPixmapItem representing the Quality control Icon.

QTimer * shieldTimer

QTimer responsible for deactivating the shield after 5 seconds.

• QTimer * timer

QTimer responsible for updating the time each second.

- QGraphicsPixmapItem * soul1
- QGraphicsPixmapItem * soul2
- QGraphicsPixmapItem * soul3

QGraphicsPixmapItem representing the souls of the tester.

- QGraphicsPixmapItem * tens
- QGraphicsPixmapItem * units

QGraphicsPixmapItem representing the time left.

- QGraphicsPixmapItem * ammoTens
- QGraphicsPixmapItem * ammoUnits

QGraphicsPixmapItem representing the ammo left.

5.6.1 Member Function Documentation

5.6.1.1 int Game2Scene::getHighScore ()

Responsible of fetching the HighScore from the profile file.

< to check if it is entering the file, and it is

5.6.1.2 void Game2Scene::updateHighScore (QString user)

Responsible of updating the HighScore in the profile file if needed.

<To update the level number in the text file (stored as the 10th entry)

5.6.2 Member Data Documentation

5.6.2.1 QLabel* Game2Scene::scoreL

QLabel holding the word HighScore.

QLabel holding the word score

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

- game2scene.h
- · game2scene.cpp

5.7 gameOver Class Reference

Inheritance diagram for gameOver:



Public Member Functions

gameOver (QWidget *parent=nullptr)
 Setting the message pop up window content.

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

- · gameover.h
- · gameover.cpp

5.8 gamesWidget Class Reference

Inheritance diagram for gamesWidget:



Public Slots

• void startGame1 ()

Slot to take the user to Game1Scene.

• void startGame2 ()

Slot to take the user to Game2Scene.

Public Member Functions

• gamesWidget (QString user)

Setting the widget's layout.

void setVerticalLayout ()

Setting the Vertical Layout.

• void setGridLayout ()

Setting the Grid Layout.

Public Attributes

- QPushButton * game1
- QPushButton * game2
- QString user
- QVBoxLayout * VerticalL
- QGridLayout * GridL
- game1scene * scene1

pointer to an Object of type game1scene

- Game2Scene * scene2
- QGraphicsView * view1

pointer to an Object of type QGraphicsView

QGraphicsView * view2

5.9 hints Class Reference 21

5.8.1 Member Function Documentation

5.8.1.1 void gamesWidget::startGame1 () [slot]

Slot to take the user to Game1Scene.

- < pointer to an Object of type game1scene
- < pointer to an Object of type QGraphicsView

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

- · gameswidget.h
- · gameswidget.cpp

5.9 hints Class Reference

Inheritance diagram for hints:



Public Member Functions

- hints (QWidget *parent=nullptr)
 Setting the hints pop up window content.
- hints (QString hint, QWidget *parent=nullptr)

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

- · hints.h
- · hints.cpp

5.10 instruction Class Reference

Inheritance diagram for instruction:



Public Member Functions

• instruction (QWidget *parent=nullptr)

Setting the instruction pop up window content.

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

- · instruction.h
- · instruction.cpp

5.11 LevelParser Class Reference

Public Member Functions

• LevelParser (QString filePath)

Constructor.

• void parse (Game2Scene *scene)

Responsible of actually parsing the file and adding the elements to the scene.

Public Attributes

QDir * filePath

Directory to the file being parsed.

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

- · levelparser.h
- · levelparser.cpp

5.12 levels Class Reference

Public Member Functions

· levels (int x, QString user)

Setting the specs for each level.

void setPopeyeX1 (int newX)

Setting up popeye's X and Y positions.

void setPopeyeY1 (int newY)

Setting Popeye's New Y position.

void incrementHintNumber ()

Incrementing the hint Number.

void resetHintNumber ()

Restting the hint Number.

• void decrementLifes ()

Decrementing the number of lifes.

· void incrementLevelNumb ()

Incrementing the Level Number.

• void incrementScore ()

Incrementing the score.

• void decrementScore ()

Decrementing the score.

void updateLevel (QString user)

Updating the level Number when user passes a level.

void updateLevelOne (QString user)

Reseting the user's level to 1 if he looses the game.

• int getScore (QString user)

- · void updateScore (QString user)
- void resetLives ()

Reseting the lifes to it's initial count of 5.

· void resetScore ()

Reseting the score to it's initial count of 100.

QStringList profileParser (QString)

parse the line in a text file and return a list of strings

Public Attributes

- QString hint [3]
- QString instructions
- int popeyeX1
- int popeyeY1

popeye's X and Y positions

- int spinachX1
- · int spinachY1
- int spinachX2
- int spinachY2
- int spinachX3
- int spinachY3

spinach can's X and Y positions

- int river1X
- · int river1Y

the river's X and Y positions

- int boat1X
- int boat1Y

the boat's X and Y positions

- int rock1X
- int rock1Y

the rock's X and Y positions

- int obstacleX
- · int obstacleY

the obstacle's X and Y positions

- int smallRiver1X
- int smallRiver1Y

the first small river's X and Y positions

- · int smallRiver2X
- int smallRiver2Y

the second small river \boldsymbol{X} and \boldsymbol{Y} positions

- · int levelNumb
- · int hintNumber
- · int lifes

User's level Number, hint number and life number count.

int score

User's score count.

5.12.1 Member Function Documentation

5.12.1.1 int levels::getScore (QString user)

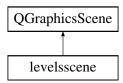
< to check if it is entering the file, and it is

```
5.12.1.2 void levels::incrementHintNumber ( )
Incrementing the hint Number.
Increasing the hint Number.
5.12.1.3 QStringList levels::profileParser ( QString line )
parse the line in a text file and return a list of strings
<parse the line and return a list.</pre>
5.12.1.4 void levels::resetHintNumber ( )
Restting the hint Number.
Reseting the Hint Number.
5.12.1.5 void levels::resetLives ( )
Reseting the lifes to it's initial count of 5.
Reseting the number of lives.
5.12.1.6 void levels::setPopeyeX1 (int newX)
Setting up popeye's X and Y positions.
Setting Popeye's new X position.
5.12.1.7 void levels::updateLevel ( QString user )
Updating the level Number when user passes a level.
<To update the level number in the text file (stored as the 10th entry)
5.12.1.8 void levels::updateLevelOne ( QString user )
Reseting the user's level to 1 if he looses the game.
<To update the level number in the text file (stored as the 10th entry)
5.12.1.9 void levels::updateScore ( QString user )
<To update the level number in the text file (stored as the 10th entry)
The documentation for this class was generated from the following files:
```

- · levels.h
- · levels.cpp

5.13 levelsscene Class Reference

Inheritance diagram for levelsscene:



Public Slots

void checkAnswer ()

Checking the User's input and moving popeye and the boat accordingly. If the User Clicks on the pickUp Spinach cans dissapear.

· void displayHint ()

Displaying the 3 hints and repeating them if passed over all of them and decrement score accordingly.

void pauseLevel ()

Pause the level the user's in.

void retryLevel ()

Retry level and reset positons to initial hard coded positions.

· void hideScene ()

Hide the scene.

void updateCountdownTimer ()

Function that will update the timer and then display it.

Public Member Functions

• levelsscene (QString user)

The constructor the this class is called, everytime the user clicks on the Let's Go! Button and thus, the corresponding level is loaded from the appropriate text file. Each Level have different specs and object that are loaded to their appropriate positions.

- void looseLife ()
- · void gameOver ()
- void youLost (bool)

Decrementing lifes, score and showing appropriate pop up window when users' looses.

void youWon ()

Incrementing score, proceeding to the game1scene and showing appropriate pop up window when user wins.

- void setUpCountdownTimer ()
 - < Timer controlling the countdown

Public Attributes

- · QString user
- Popeye * popeye = new Popeye()

Creating the references to the Objects.

- spinach * spinach1 = new spinach()
- spinach * spinach2 = new spinach()
- spinach * spinach3 = new spinach()
- river * river1 = new river()
- boat * boat1 = new boat()
- rock * rock1 = new rock()
- riverObstacle * obstacle = new riverObstacle
- smallRiver * smallRiver1 = new smallRiver()
- smallRiver * smallRiver2 = new smallRiver()

- levels * I
- QPushButton * run
- QPushButton * hint
- QPushButton * pause
- QPushButton * retry
- QPushButton * proceed

QWidgets that will displayed on the screen.

- QTextEdit * text
- QLabel * instructions
- QGraphicsTextItem * countDownText
- int countdown = 120
- QTimer * countdownTimer

< value of the countdown at the start of the game that is = to 120 i.e. 2 mins for each level

5.13.1 Constructor & Destructor Documentation

5.13.1.1 levelsscene::levelsscene (QString user)

The constructor the this class is called, everytime the user clicks on the Let's Go! Button and thus, the corresponding level is loaded from the appropriate text file. Each Level have different specs and object that are loaded to their appropriate positions.

- < User can only pause after level 1 is completed
- < if the levels has ot not other spinach cans

5.13.2 Member Function Documentation

5.13.2.1 void levelsscene::checkAnswer() [slot]

Checking the User's input and moving popeye and the boat accordingly. If the User Clicks on the pickUp Spinach cans dissapear.

- < Run push button function
- <direction of movement, 0 to the right and anticlockwise
- <check the argument of the Repeat(args) from the user</p>
- <get the number of iterations
- <start reading after we get the number of iterations
- < iterate j times over the user's Code
- < i is either = 0 or 2 and at the end of the first for loop i is resetted to 2 if there is repetition

popeye cannot move in level 5; boat collides with him only

to check if popeye can pass on the bridge or if there is a river

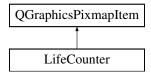
- If popeye collides with the boat, they should both move in the same manner if Boat.Move is inputed
- < In level 7 the boat can only move on a horizontal line
- < Each Rotate will rotate popeye by 90 degrees
- <if the colliding object is of type spinach
- < decrement the number of lifes before showing it to the user and show appropriate syntax error

```
5.13.2.2 void levelsscene::hideScene() [slot]
Hide the scene.
Hide the levelsscene.
5.13.2.3 void levelsscene::pauseLevel() [slot]
Pause the level the user's in.
<Since everything is loaded from the text file; the lives, scores, timer and level number, the pause simply closes the
scene and when the user re-logs in, he just continues where he left off when he clicks on the Let's Start! button
5.13.2.4 void levelsscene::retryLevel( ) [slot]
Retry level and reset positions to initial hard coded positions.
<retry -> reset popeye's position
< if the levels has ot not other spinach cans
5.13.2.5 void levelsscene::setUpCountdownTimer ( )
< Timer controlling the countdown
Initializes TextItem that indicates the Countdown at the start of each level.
Setting up the timer
\textbf{5.13.2.6} \quad \textbf{void levels scene::} \textbf{updateCountdownTimer( )} \quad [\, \texttt{slot} \, ]
Function that will update the timer and then display it.
update SLOT that controls the Countdown to display it when adjusted, every second
5.13.2.7 void levelsscene::youLost (bool timelsUp)
Decrementing lifes, score and showing appropriate pop up window when users' looses.
<reset the user to level 1
<reset the user to level 1
5.13.2.8 void levelsscene::youWon()
Incrementing score, proceeding to the game1scene and showing appropriate pop up window when user wins.
<Increment the score according the count of lifes, the higher the lifes, the higher the score</p>
< Increment the score according to the player's time
The documentation for this class was generated from the following files:
```

- · levelsscene.h
- levelsscene.cpp

5.14 LifeCounter Class Reference

Inheritance diagram for LifeCounter:



Public Member Functions

• LifeCounter (int life)

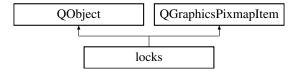
Constructor.

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

- · lifecounter.h
- · lifecounter.cpp

5.15 locks Class Reference

Inheritance diagram for locks:



Public Member Functions

• locks (QObject *parent=nullptr)

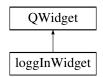
Setting the lock Image.

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

- locks.h
- locks.cpp

5.16 loggInWidget Class Reference

Inheritance diagram for loggInWidget:



Public Slots

· void startGames ()

Public Member Functions

- logglnWidget (QString user="guest")
- void setVerticalLayout ()
- void setGridLayout ()

Public Attributes

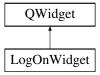
- QLabel * userName
- QLabel * logIn
- QPushButton * games
- QString user
- QVBoxLayout * VerticalL
- QGridLayout * GridL

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

- · logginwidget.h
- · logginwidget.cpp
- logginwidget.cpp.BACKUP.3062.cpp
- logginwidget.cpp.BASE.3062.cpp
- logginwidget.cpp.LOCAL.3062.cpp
- logginwidget.cpp.REMOTE.3062.cpp

5.17 LogOnWidget Class Reference

Inheritance diagram for LogOnWidget:



Public Slots

- void GotoSignUpPage ()
- void LogInAsGuest ()
- void GotoSignInPage ()

Public Member Functions

- LogOnWidget (QWidget *parent=nullptr)
- void setVerticalLayout ()
- void setGridLayout ()

30 Class Documentation

Public Attributes

- QPushButton * SignUpButton
- QPushButton * SignInButton
- QPushButton * GuestButton
- QLabel * TitleLabel
- QLabel * InfoLabel
- QVBoxLayout * VerticalLayout
- QGridLayout * GridLayout
- QGraphicsScene * Scene

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

- logonwidget.h
- · logonwidget.cpp

5.18 lost Class Reference

Inheritance diagram for lost:



Public Member Functions

lost (QWidget *parent=nullptr)

Setting the lost pop up window content, displaying the total number of lifes left and any syntax error when needed.

- lost (int lifes, QWidget *parent=nullptr)
- lost (int lifes, QString command, QWidget *parent=nullptr)

5.18.1 Constructor & Destructor Documentation

```
5.18.1.1 lost::lost ( int lifes, QWidget * parent = nullptr )
```

<If the lives count is less than 0 the user would have lost completely and therefore would have to restart the game</p>

```
5.18.1.2 lost::lost (int lifes, QString command, QWidget * parent = nullptr)
```

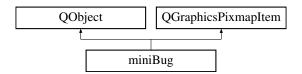
< not checking for the in entered in the Move(#) function

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

- · lost.h
- lost.cpp

5.19 miniBug Class Reference

Inheritance diagram for miniBug:



Public Slots

• void move ()

responsible of moving the miniBug on each timer timeout

Public Member Functions

miniBug (Game2Scene *, Bug *, int)
 Constructor.

Public Attributes

• QPixmap * icon

Pixmap holding the image.

• Game2Scene * scene

The scene where the miniBug is located.

• Bug * bug

The parent Bug.

• QTimer * timer

QTimer used to move the miniBug periodically.

· int steps

Number of steps taken already used in determining the range.

int dir

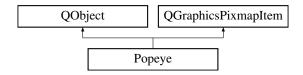
Integer holding the direction of the miniBug.

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

- minibug.h
- minibug.cpp

5.20 Popeye Class Reference

Inheritance diagram for Popeye:



32 Class Documentation

Public Member Functions

Popeye (QObject *parent=nullptr)
 Setting Popeye's Image.

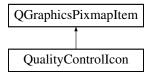
• void collision ()

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

- · popeye.h
- popeye.cpp

5.21 QualityControllcon Class Reference

Inheritance diagram for QualityControllcon:



Public Member Functions

• QualityControllcon ()

Constructor.

Public Attributes

• QPixmap * icon

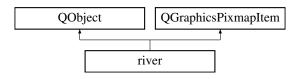
Pixmap holding the image of the QC icon.

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

- · qualitycontrolicon.h
- qualitycontrolicon.cpp

5.22 river Class Reference

Inheritance diagram for river:



Public Member Functions

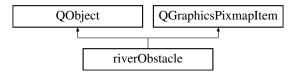
river (QObject *parent=nullptr)
 Setting the river's Image.

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

- · river.h
- · river.cpp

5.23 riverObstacle Class Reference

Inheritance diagram for riverObstacle:



Public Member Functions

• riverObstacle (QObject *parent=nullptr)

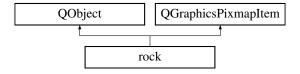
Setting the river obstacle's Image.

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

- · riverobstacle.h
- · riverobstacle.cpp

5.24 rock Class Reference

Inheritance diagram for rock:



Public Member Functions

rock (QObject *parent=nullptr)
 Setting the rock's Image.

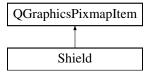
The documentation for this class was generated from the following files:

- · rock.h
- rock.cpp

34 Class Documentation

5.25 Shield Class Reference

Inheritance diagram for Shield:



Public Member Functions

• Shield ()

Constructor.

Public Attributes

• QPixmap * icon

Pixmap holding the image of the shield.

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

- · shield.h
- shield.cpp

5.26 signInWidget Class Reference

Inheritance diagram for signInWidget:



Public Slots

- void loggedIn ()
- void homepage ()

Public Member Functions

- signInWidget (QWidget *parent=nullptr)
- void setVerticalLayout ()
- void setGridLayout ()

Public Attributes

- QLabel * userName
- QLabel * password
- QLabel * signIn
- QLineEdit * LuserName
- QLineEdit * Lpassword
- QPushButton * submit
- QPushButton * back
- const QString esc ="7727"
- QVBoxLayout * VerticalL
- QGridLayout * GridL

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

- · signinwidget.h
- · signinwidget.cpp

5.27 SignUpWidget Class Reference

Inheritance diagram for SignUpWidget:



Public Slots

- void VerifySubmitSlot ()
- void GoBackToLogOnSlot ()
- void homepage ()

Public Member Functions

• SignUpWidget (QWidget *parent=nullptr)

Public Attributes

- QLabel * FirstName
- QLabel * LastName
- QLabel * UserName
- QLabel * Password
- QLabel * ConfirmPass
- QLabel * ProfilePicture
- QLabel * Gender
- QLineEdit * First
- QLineEdit * Last
- QLineEdit * User
- QLineEdit * Pass

36 Class Documentation

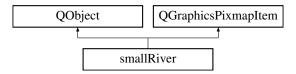
- QLineEdit * Confirm
- QPushButton * Submit
- QPushButton * back
- QRadioButton * Male
- QRadioButton * Female
- QRadioButton * ProfilePic1
- QRadioButton * ProfilePic2
- QRadioButton * ProfilePic3
- QRadioButton * ProfilePic4
- QGroupBox * groupBox
- QGridLayout * GridLayout
- QVBoxLayout * VerticalLayout
- QGroupBox * previewGroupBox
- QGridLayout * previewLayout
- QCalendarWidget * calendar
- const QString esc ="7727"

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

- · signupwidget.h
- signupwidget.cpp

5.28 smallRiver Class Reference

Inheritance diagram for smallRiver:



Public Member Functions

• smallRiver (QObject *parent=nullptr)

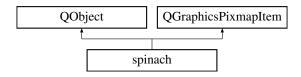
Setting the small river's Image.

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

- · smallriver.h
- · smallriver.cpp

5.29 spinach Class Reference

Inheritance diagram for spinach:



5.30 Tester Class Reference 37

Public Member Functions

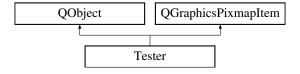
spinach (QObject *parent=nullptr)
 Setting the spinach can's Image.

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

- spinach.h
- spinach.cpp

5.30 Tester Class Reference

Inheritance diagram for Tester:



Public Member Functions

- Tester (int, int)
- void decrementLives ()
- · void loseLife ()
- void drinkCoffee ()
- void showStats ()

Public Attributes

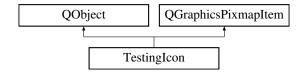
- QPixmap * icon
- QTimer * timer
- int lives
- · int souls
- int startingX
- · int startingY

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

- · tester.h
- · tester.cpp

5.31 TestingIcon Class Reference

Inheritance diagram for TestingIcon:



38 Class Documentation

Public Slots

• void reActivate ()

Public Member Functions

• void deActivate ()

Public Attributes

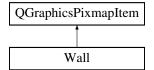
- QPixmap * icon
- bool isHidden
- QTimer * timer

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

- · testingicon.h
- testingicon.cpp

5.32 Wall Class Reference

Inheritance diagram for Wall:



Public Attributes

• QPixmap * icon

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

- wall.h
- wall.cpp

5.33 Won Class Reference

Inheritance diagram for Won:



5.33 Won Class Reference 39

Public Member Functions

- Won (QWidget *parent)

 Setting the win pop up window content.
- Won (int, int)

5.33.1 Constructor & Destructor Documentation

5.33.1.1 Won::Won (int level, int lives)

< If level is 8 and user won therefore he would have the entire game.

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

- won.h
- won.cpp

40 **Class Documentation**

Chapter 6

File Documentation

6.1 boat.cpp File Reference

Contains Popeye class definition.

```
#include "boat.h"
```

6.1.1 Detailed Description

Contains Popeye class definition.

Author

Camille Farhat & Ali Haidoura

6.2 boat.h File Reference

contains boat class definition

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

class boat

6.2.1 Detailed Description

contains boat class definition

Author

Camille Farhat & Ali Haidar

6.3 bug.cpp File Reference

contains Bug class definitions

```
#include "bug.h"
#include "tester.h"
#include "minibug.h"
#include <QTransform>
```

6.3.1 Detailed Description

contains Bug class definitions

Author

Ali Al Akbar Haidoura

6.4 bug.h File Reference

QGraphicsPixmapItem representing the bugs.

```
#include <QGraphicsPixmapItem>
#include <QTimer>
```

Classes

· class Bug

6.4.1 Detailed Description

QGraphicsPixmapItem representing the bugs. Bug objects move horizentally, shoot at the player when in sight also contains collision logic

Author

Ali Al Akbar Haidoura

6.5 bullet.cpp File Reference

contains **Bullet** class definitions

```
#include "bullet.h"
#include "wall.h"
#include "bug.h"
#include "minibug.h"
#include <QTransform>
```

6.5.1 Detailed Description

contains Bullet class definitions

Author

Ali Al Akbar Haidoura

6.6 bullet.h File Reference 43

6.6 bullet.h File Reference

QGraphicsPixmapItem representing the bullets.

```
#include <QGraphicsPixmapItem>
#include <QTimer>
#include <QDebug>
#include "game2scene.h"
```

Classes

· class Bullet

6.6.1 Detailed Description

QGraphicsPixmapItem representing the bullets.

Author

Ali Al Akbar Haidoura

6.7 coffeecup.cpp File Reference

Contains CoffeeCup Class definitions.

```
#include "coffeecup.h"
```

6.7.1 Detailed Description

Contains CoffeeCup Class definitions.

Author

Ali Al Akbar Haidoura

6.8 coffeecup.h File Reference

QGraphicsPixmapItem representing the Coffee Cup.

```
#include <QGraphicsPixmapItem>
```

Classes

class CoffeeCup

5.8.1 Detailed Description

QGraphicsPixmapItem representing the Coffee Cup.

Author

Ali Al Akbar Haidoura

6.9 game1scene.cpp File Reference

contains Game1's class definition

```
#include <QObject>
#include <QGraphicsPixmapItem>
#include <QGraphicsScene>
#include <QtEvents>
#include <QTimer>
#include "popeye.h"
#include "locks.h"
#include "game1scene.h"
#include "gameswidget.h"
#include "gameover.h"
```

6.9.1 Detailed Description

contains Game1's class definition This is the scene where we see Popeye, Olive and the locks. Each time time popeye passes a level, the corresponding lock is hiden and popeye's position is updated. The let's Go Button is the button that lunches the user to the corresponding level.

Author

Camille Farhat & Ali Haidoura

6.10 game1scene.h File Reference

The Game's scene.

```
#include <QObject>
#include <QGraphicsScene>
#include <QGraphicsPixmapItem>
#include <QtEvents>
#include <QTimer>
#include <QtWidgets>
#include "popeye.h"
#include "locks.h"
#include "levelsscene.h"
```

Classes

class game1scene

6.10.1 Detailed Description

The Game's scene. This is the header file for our first game's scene. It publicly inherits from QGraphicsScene.

6.11 game2scene.cpp File Reference

Contains Game2Scene Class definitions.

```
#include "game2scene.h"
#include "levelparser.h"
#include "bug.h"
#include "wall.h"
#include "coffeecup.h"
#include "qualitycontrolicon.h"
#include "testingicon.h"
#include "shield.h"
#include "bullet.h"
#include <QDebug>
#include <QGraphicsItem>
```

6.11.1 Detailed Description

Contains Game2Scene Class definitions.

Author

Ali Al Akbar Haidoura

6.12 game2scene.h File Reference

The scene in which all the logic and items of Game2 are located.

```
#include <QGraphicsScene>
#include <QGraphicsView>
#include <QTimer>
#include <QLabel>
#include <QPushButton>
#include "tester.h"
#include "qualitycontrolicon.h"
#include "lifecounter.h"
#include "testingicon.h"
#include "bug.h"
```

Classes

· class Game2Scene

6.12.1 Detailed Description

The scene in which all the logic and items of Game2 are located.

Author

Ali Al Akbar Haidoura

6.13 gameswidget.cpp File Reference

contains Game1-2 Links

```
#include <QtWidgets>
#include <QObject>
#include <QPixmap>
#include <QGraphicsItem>
#include <QGraphicsPixmapItem>
#include <QGraphicsScene>
#include <QGraphicsView>
#include <QApplication>
#include "game1scene.h"
#include "gameswidget.h"
#include "levelparser.h"
```

6.13.1 Detailed Description

contains Game 1-2 Links

Author

Camille Farhat & Ali Haidoura

6.14 hints.cpp File Reference

Shows the appropriate hints in a pop up window.

```
#include "hints.h"
#include "levels.h"
#include <QLabel>
#include <QVBoxLayout>
```

6.14.1 Detailed Description

Shows the appropriate hints in a pop up window.

Author

Camille Farhat & Ali Haidoura

6.15 hints.h File Reference

The Level's scene's Hints.

```
#include <QWidget>
```

Classes

class hints

6.15.1 Detailed Description

The Level's scene's Hints. Pop up window that will display the hints.

6.16 instruction.cpp File Reference

Shows the initial instructions to start the game in a pop up window.

```
#include "instruction.h"
#include <QLabel>
#include <QLayout>
```

6.16.1 Detailed Description

Shows the initial instructions to start the game in a pop up window.

Author

Camille Farhat & Ali Haidoura

6.17 instruction.h File Reference

The Level's scene's levels pop up used for displaying instructions.

```
#include <QWidget>
```

Classes

· class instruction

6.17.1 Detailed Description

The Level's scene's levels pop up used for displaying instructions. Pop up window that will display the Initial Instructions before starting the Game.

6.18 levelparser.cpp File Reference

Contains LevelParser Class definitions.

```
#include "levelparser.h"
#include <QDebug>
#include <bug.h>
#include "wall.h"
#include "tester.h"
#include "testingicon.h"
#include "shield.h"
#include "coffeecup.h"
#include "qualitycontrolicon.h"
```

6.18.1 Detailed Description

Contains LevelParser Class definitions.

Author

Ali Al Akbar Haidoura

6.19 levelparser.h File Reference

Parses the game2 levels from the text files.

```
#include <QString>
#include <QDir>
#include "game2scene.h"
```

Classes

class LevelParser

6.19.1 Detailed Description

Parses the game2 levels from the text files.

Author

Ali Al Akbar Haidoura

6.20 levels.cpp File Reference

Setting up the different specs for each level to load them dynamically everytime a user starts a level.

```
#include "levels.h"
```

6.20.1 Detailed Description

Setting up the different specs for each level to load them dynamically everytime a user starts a level.

Author

Camille Farhat & Ali Haidoura

6.21 levels.h File Reference

The Level Object.

```
#include <QWidget>
#include <QString>
#include <QTimer>
#include <QtWidgets>
#include <QObject>
#include "spinach.h"
#include "popeye.h"
```

Classes

· class levels

6.21.1 Detailed Description

The Level Object. Creating a Level Object that will be passed to every scene to load the respective level.

6.22 levelsscene.cpp File Reference

Contains Game1's class definition.

```
#include "levelsscene.h"
#include "levels.h"
#include "hints.h"
#include "lost.h"
#include "won.h"
#include "instruction.h"
#include "boat.h"
```

6.22.1 Detailed Description

Contains Game1's class definition. This file is creating the enviorment of our game. Setting everything from background image to the timer, popeye, the spinach cans, the boat and the river obstacles and other level-specific specifications. The user will be writing code into the text editor. This code will be chekcked and popeye will move accordingly.

Author

Camille Farhat & Ali Haidoura

6.23 lifecounter.cpp File Reference

```
Contains LifeCounter Class definitions.
```

```
#include "lifecounter.h"
```

6.23.1 Detailed Description

Contains LifeCounter Class definitions.

Author

Ali Al Akbar Haidoura

6.24 lifecounter.h File Reference

QGraphicsPixmapItem representing.

```
#include <QGraphicsPixmapItem>
```

Classes

· class LifeCounter

6.24.1 Detailed Description

QGraphicsPixmapItem representing.

Author

Ali Al Akbar Haidoura

6.25 locks.cpp File Reference

Contains Lock class definition.

```
#include "locks.h"
#include <QGraphicsPixmapItem>
```

6.25.1 Detailed Description

Contains Lock class definition.

Author

Camille Farhat & Ali Haidar

6.26 locks.h File Reference

Contains Lock class definition.

```
#include <QWidget>
#include <QGraphicsPixmapItem>
#include <QObject>
```

Classes

· class locks

6.26.1 Detailed Description

Contains Lock class definition.

Author

Camille Farhat & Ali Haidar

6.27 lost.cpp File Reference

Shows the appropriate loose message in a pop up window.

```
#include "lost.h"
#include <QLabel>
#include <QVBoxLayout>
```

6.27.1 Detailed Description

Shows the appropriate loose message in a pop up window.

Author

Camille Farhat & Ali Haidoura

6.28 lost.h File Reference

The Level's scene's loose pop up.

```
#include <QWidget>
```

Classes

· class lost

6.28.1 Detailed Description

The Level's scene's loose pop up. Pop up window that will display the number of lifes remaining and that the player lost.

6.29 minibug.cpp File Reference

Contains miniBug Class definitions.

```
#include "minibug.h"
#include "bullet.h"
```

6.29.1 Detailed Description

Contains miniBug Class definitions.

Author

Ali Al Akbar Haidoura

6.30 popeye.cpp File Reference

Contains Popeye class definition.

```
#include "popeye.h"
```

6.30.1 Detailed Description

Contains Popeye class definition.

Author

Camille Farhat & Ali Haidoura

6.31 popeye.h File Reference

Contains Popeye class definition.

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

• class Popeye

6.31.1 Detailed Description

Contains Popeye class definition.

Author

Camille Farhat & Ali Haidar

6.32 qualitycontrolicon.cpp File Reference

Contains QualityControllcon Class definitions.

```
#include "qualitycontrolicon.h"
```

6.32.1 Detailed Description

Contains QualityControllcon Class definitions.

Author

Ali Al Akbar Haidoura

6.33 qualitycontrolicon.h File Reference

QGraphicsPixmapItem representing the quality control icon.

```
#include <QGraphicsPixmapItem>
```

Classes

· class QualityControllcon

6.33.1 Detailed Description

QGraphicsPixmapItem representing the quality control icon.

Author

Ali Al Akbar Haidoura

6.34 river.cpp File Reference

Contains river class definition.

```
#include "river.h"
#include <QGraphicsPixmapItem>
```

6.34.1 Detailed Description

Contains river class definition.

Author

Camille Farhat & Ali Haidoura

6.35 river.h File Reference

Contains Popeye class definition.

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

· class river

6.35.1 Detailed Description

Contains Popeye class definition.

Author

Camille Farhat & Ali Haidar

6.36 riverobstacle.cpp File Reference

Contains River Obstacle class definition.

```
#include "riverobstacle.h"
#include <QGraphicsPixmapItem>
```

6.36.1 Detailed Description

Contains River Obstacle class definition.

Author

Camille Farhat & Ali Haidar

6.37 riverobstacle.h File Reference

Contains River Obstacle class definition.

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

· class riverObstacle

6.37.1 Detailed Description

Contains River Obstacle class definition.

Author

Camille Farhat & Ali Haidar

6.38 rock.cpp File Reference

contains rock class definition

```
#include "rock.h"
#include <QGraphicsPixmapItem>
```

6.38.1 Detailed Description

contains rock class definition

Author

Camille Farhat & Ali Haidoura

6.39 rock.h File Reference

contains rock class definition

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

• class rock

6.39.1 Detailed Description

contains rock class definition

Author

Camille Farhat & Ali Haidar

6.40 shield.cpp File Reference

Contains shield Class definitions.

```
#include "shield.h"
```

6.40.1 Detailed Description

Contains shield Class definitions.

Author

Ali Al Akbar Haidoura

6.41 shield.h File Reference

QGraphicsPixmapItem representing the shield icon.

```
#include <QGraphicsPixmapItem>
```

Classes

class Shield

6.41.1 Detailed Description

 $QGraphics Pix map I tem\ representing\ the\ shield\ icon.$

Author

Ali Al Akbar Haidoura

6.42 signinwidget.h File Reference

User sign in widget.

```
#include <QWidget>
#include <QtWidgets>
#include <QObject>
#include <QDir>
#include <QFileDialog>
#include <iostream>
#include <QFile>
#include <fstream>
```

Classes

· class signInWidget

6.42.1 Detailed Description

User sign in widget. This is the header file for our the sign in widget where the user enters his username and password to Log in

6.43 signupwidget.h File Reference

User sign up widget.

```
#include "logonwidget.h"
#include <QWidget>
#include <QtWidgets>
#include <fstream>
#include <QFileInfo>
#include <QCalendarWidget>
```

Classes

class SignUpWidget

6.43.1 Detailed Description

User sign up widget. This is the header file for our the sign up widget where the user enters his information before signing in

6.44 smallriver.cpp File Reference

contains Small River class definition

```
#include "smallriver.h"
#include <QGraphicsPixmapItem>
```

6.44.1 Detailed Description

contains Small River class definition

Author

Camille Farhat & Ali Haidoura

6.45 smallriver.h File Reference

contains Small River class definition

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

· class smallRiver

6.45.1 Detailed Description

contains Small River class definition

Author

Camille Farhat & Ali Haidar

6.46 spinach.cpp File Reference

Contains Spinach class definition.

```
#include "spinach.h"
#include <QObject>
```

6.46.1 Detailed Description

Contains Spinach class definition.

Author

Camille Farhat & Ali Haidoura

6.47 spinach.h File Reference

Contains Spinach class definition.

```
#include <QObject>
#include <QGraphicsPixmapItem>
```

Classes

· class spinach

6.47.1 Detailed Description

Contains Spinach class definition.

Author

Camille Farhat & Ali Haidar

6.48 tester.cpp File Reference

Contains tester Class definitions.

```
#include "tester.h"
#include <QTimer>
```

6.48.1 Detailed Description

Contains tester Class definitions.

Author

Ali Al Akbar Haidoura

6.49 tester.h File Reference

QGraphicsPixmapItem representing the Tester character.

```
#include <QGraphicsPixmapItem>
#include <QKeyEvent>
#include <QObject>
#include <QTimer>
```

Classes

· class Tester

6.49.1 Detailed Description

QGraphicsPixmapItem representing the Tester character.

Author

Ali Al Akbar Haidoura

6.50 testingicon.cpp File Reference

Contains testingIcon Class definitions.

```
#include "testingicon.h"
```

6.50.1 Detailed Description

Contains testingIcon Class definitions.

Author

Ali Al Akbar Haidoura

6.51 testingicon.h File Reference

QGraphicsPixmapItem representing the testingIcon.

```
#include <QGraphicsPixmapItem>
#include <QTimer>
#include <QObject>
```

Classes

• class TestingIcon

6.51.1 Detailed Description

QGraphicsPixmapItem representing the testingIcon.

Author

Ali Al Akbar Haidoura

6.52 won.cpp File Reference

Shows the appropriate win message in a pop up window.

```
#include "won.h"
#include <QLabel>
#include <QLayout>
```

6.52.1 Detailed Description

Shows the appropriate win message in a pop up window.

Author

Camille Farhat & Ali Haidoura

6.53 won.h File Reference

Shows the appropriate win message in a pop up window.

#include <QWidget>

Classes

class Won

6.53.1 Detailed Description

Shows the appropriate win message in a pop up window.

Author

Camille Farhat & Ali Haidar