Space Invader

Generated by Doxygen 1.8.13

Contents

Chapter 1

Namespace Index

1.1 Namespace List

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

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

Namespace Documentation

4.1 game Namespace Reference

Classes

- class Alien
- class AlienBase
- class AlienBuilder
- class BarrierBlock
- class Base
- class Bullet
- class BulletBuilder
- class BulletBuilderInterface
- class Command
- class CommandClearStage
- class CommandGamePause
- class CommandGameStart
- class CommandGotoGameMode
- class CommandGotoLeaderBoardMode
- class CommandGotoStageMakerMode
- class CommandGotoTitleScreenMode
- class CommandRestartStage
- class Config
- class Cursor
- class CursorState
- class DiagonalBullet
- class FighterState
- class GameDialog
- class GameMenu
- class Hunter
- class InstructionRequest
- struct LaserBeam
- class LeaderBoard
- class MakerState
- class Menu
- class NormalState
- class PenState
- class Powerup

- class Ship
- struct SMakerPlacedObject
- class StageMaker
- class StatusBar
- class Swarm
- class SwarmInfo
- class UnitTestSpaceInvader

Enumerations

- enum **CURSOR_STATE** {
  NORMAL, **FIGHTER**, **PEN**, **STAGEMAKER**,
  END_OF_CURSOR_STATE }
- enum **GAME_STATUS** {
  GAME_STATUS_TITLE_SCREEN, **GAME_STATUS_IN_GAME**, **GAME_STATUS_LEADER_BOARD**,
  GAME_STATUS_STAGE_MAKER,
  GAME_STATUS_STAGE_MAKER_TESTING }
- enum **CannonType** { **Normal**, **MachineGun**, **Laser**, **END_OF_CANNON_TYPE** }
- enum **SMakerObjectType** {
  SMAKER_HOLDING_ALIEN_RED, **SMAKER_HOLDING_ALIEN_BLUE**, **SMAKER_HOLDING_ALIEN_HUNTER**, **SMAKER_HOLDING_ALIEN_DUMB**,
  SMAKER_HOLDING_INSTRUCTION_BOX, **SMAKER_HOLDING_BARRIER_BLOCK**, **SMAKER_HOLDING_LINE**, **SMAKER_HOLDING_NONE** }

4.1.1 Detailed Description

Commentary added at Stage 2 I copied the original kind of format for reading... lots of if/else statements. Made it a bit easier to extend SHIP - read with lots of IF ELSE statements.

List of aliens: actually a list of swarms, and swarms are composites of aliens. Aliens are read like this: position= 0 90, 0 100, 32 99 -> "format X[space]Y"; in other words X Y, X Y, X Y -> positions are adjusted during reading. i.e., if X,Y are NOT in window boundary, they will be set to the boundary. & if X or Y is excluded (can't tell which one) ignore that entry. -> if there is an empty space (e.g. '__,__,') where _ is blank space it will skip it. ->We can have overlapping aliens for an interesting effect. ->Of course, numAliens is numPositions. type=red -> e.g., red, blue, hunter... velocity is calculated in Builder. move=L,R,L,L (shoot is determined randomly within the swarm). If empty fills with a default move.. shoot=5 -> (at each move, swarms decrement this number until 0 and then 1 alien shoots). -> when it is 0, shoot will be then reset between 0∼5 again. [see Swarm]

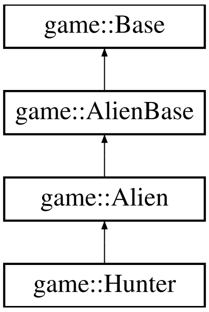
" How do we know when properties belong to a swarm, vs a different swarm or player ship?" -> the header [SHIP] will be used for the player [SWARM] for swarms [STAGE] when stage 3 of assignment is implemented. -> E.g., let's say we read [SHIP]. Enter 'SHIP' saving state. Lets say, we read position=.... but position is an ALIEN property, not player ship! Will just skip it. Keep reading until you reach another header [SWARM] or [STAGE]. -> if there is another [SHIP] ignore it and print a complaint. -> for [SWARM] there are multiple. So just start reading the next swarm.

Chapter 5

Class Documentation

5.1 game::Alien Class Reference

Inheritance diagram for game::Alien:



Public Member Functions

- **Alien** (QPixmap image, int x, int y, int velocity, int score, QString baseType)
- virtual void **move** (QString direction)
- QList< Bullet * > **shoot** (QString type)
- virtual QList< Bullet * > **react** ()
- int **get_score** () const
- QList< AlienBase * > **getAliens** () const
- bool **add** (AlienBase *toAdd)
- void **remove** (AlienBase *toDelete)
- virtual void **paint** (QPainter &painter)
- void **set_y** (int y)

Protected Attributes

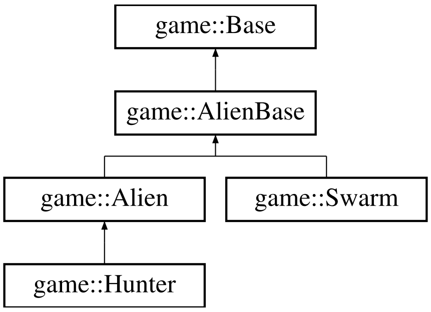
- int **velocity**
- int **score**
- QSoundEffect **bulletSFX**
- BulletBuilder **builder**

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

- alien.h
- alien.cpp

5.2 game::AlienBase Class Reference

Inheritance diagram for game::AlienBase:



Public Member Functions

- **AlienBase** (QPixmap image, double scale, int x, int y, int windowWidth, int windowHeight, int minX)
- virtual QList< Bullet ∗ > **shoot** (QString type)=0
- virtual void **move** (QString direction)=0
- virtual int **get_score** () const =0
- virtual QList< AlienBase ∗ > **getAliens** () const =0
- virtual bool **add** (AlienBase ∗toAdd)=0
- virtual void **remove** (AlienBase ∗toDelete)=0
- virtual QList< Bullet ∗ > **react** ()=0
- virtual void **paint** (QPainter &painter)=0

Additional Inherited Members

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

- alienbase.h
- composite.cpp

5.3 game::AlienBuilder Class Reference

Public Member Functions

- **AlienBuilder** (QPixmap &image, QString &type, Base &ship)
- Alien ∗ **buildAlien** (QString &type, int x, int y)
- int **velocityCalculator** (QString &type)
- int **scoreCalculator** (QString &type)

Protected Attributes

- QPixmap **image**
- QString **type**
- int **score**
- int **velocity**
- int **id**

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

- alienbuilder.h
- alienbuilder.cpp

5.4 Background Class Reference

Public Member Functions

- **Background** (int screenWidth, int screenHeight)
- void **draw** (QPainter ∗p)
- void **nextFrame** ()

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

- background.h
- background.cpp

5.5 game::BarrierBlock Class Reference

Public Member Functions

- **BarrierBlock** (int x, int y, unsigned width)
- void **draw** (QPainter ∗p)

Public Attributes

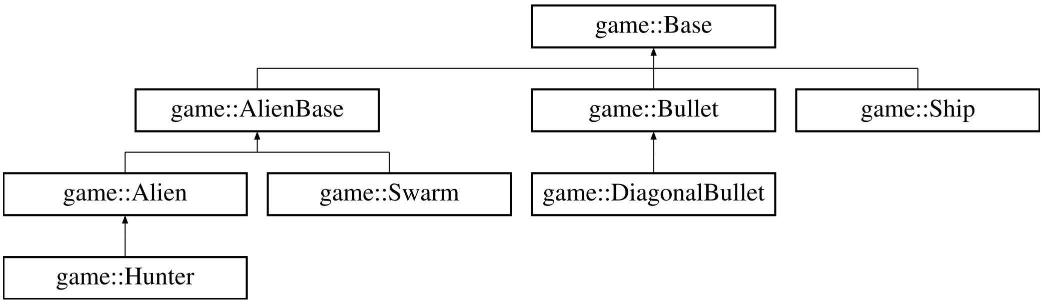
- QPixmap **pixmap**
- int **x**
- int **y**
- unsigned **width**

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

- barrierblock.h
- barrierblock.cpp

5.6 game::Base Class Reference

Inheritance diagram for game::Base:



Public Member Functions

- **Base** (QPixmap image, double scale, int x, int y=0, int boundaryX=800, int boundaryY=600, int minX=0)
- bool **checkXY** (int x1, int x2, int myX)
- bool **collides** (int x1, int x2, int y1, int y2)
- bool **collides** (Base &base)
- void **set_image** (QPixmap image)
- virtual void **set_x** (int x)
- virtual void **set_y** (int y)
- const QPixmap & **get_image** () const
- double **get_scale** () const
- int **get_x** () const
- int **get_y** () const

Protected Attributes

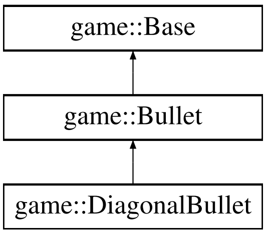
- QPixmap **image**
- int **boundaryX**
- int **boundaryY**
- int **minX**
- double **scale**
- int **x**
- int **y**

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

- base.h
- base.cpp

5.7 game::Bullet Class Reference

Inheritance diagram for game::Bullet:



Public Member Functions

- **Bullet** (QPixmap image, int x, int y, int bullet_velocity, bool friendly)
- virtual void **move** ()
- int **get_bullet_velocity** () const
- bool **isFriendly** ()

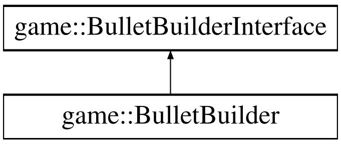
Additional Inherited Members

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

- bullet.h
- bullet.cpp

5.8 game::BulletBuilder Class Reference

Inheritance diagram for game::BulletBuilder:



Public Member Functions

- **BulletBuilder** (int velocity, Base &ship, QString baseType, bool friendly)
- void **set_velocity** (int velocity)
- Bullet ∗ **build_bullet** (QString type)

Protected Member Functions

- QPixmap **calculate_image** (QString &type)

Protected Attributes

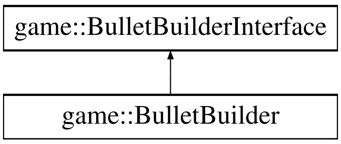
- bool **friendly**
- int **velocity**
- QPixmap **image**
- QString **baseType**
- Base & **ship**

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

- bulletbuilder.h
- bulletbuilder.cpp

5.9 game::BulletBuilderInterface Class Reference

Inheritance diagram for game::BulletBuilderInterface:



Public Member Functions

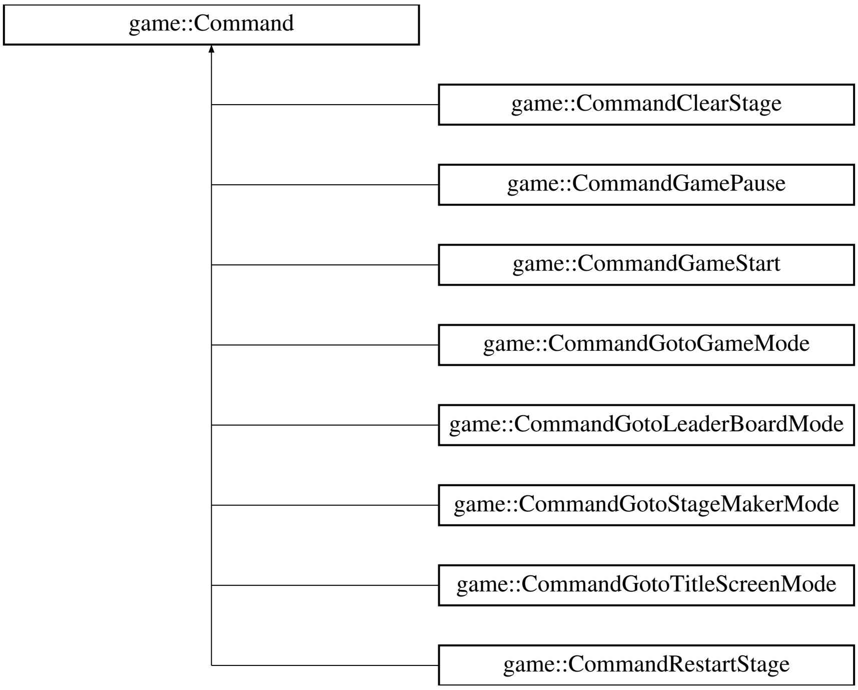
- virtual Bullet ∗ **build_bullet** (QString type)=0
- virtual void **set_velocity** (int velocity)=0

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

- bulletbuilderinterface.h

5.10 game::Command Class Reference

Inheritance diagram for game::Command:


game::Command
game::CommandClearStage
game::CommandGamePause
game::CommandGameStart
game::CommandGotoGameMode
game::CommandGotoLeaderBoardMode
game::CommandGotoStageMakerMode
game::CommandGotoTitleScreenMode
game::CommandRestartStage

Public Member Functions

- **Command** (GameDialog ∗gamedialog)
- virtual void **execute** ()=0

Protected Attributes

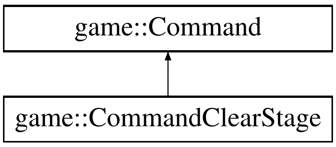
- GameDialog ∗ **gDialog**

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

- command.h

5.11 game::CommandClearStage Class Reference

Inheritance diagram for game::CommandClearStage:



Public Member Functions

- **CommandClearStage** (GameDialog ∗gamedialog)
- virtual void **execute** ()

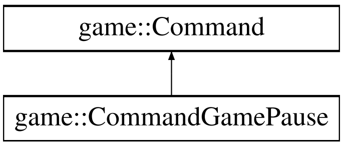
Additional Inherited Members

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

- commandclearstage.h
- commandclearstage.cpp

5.12 game::CommandGamePause Class Reference

Inheritance diagram for game::CommandGamePause:



Public Member Functions

- **CommandGamePause** (GameDialog ∗gamedialog)
- virtual void **execute** ()

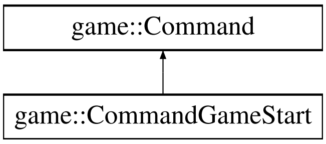
Additional Inherited Members

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

- commandgamepause.h
- commandgamepause.cpp

5.13 game::CommandGameStart Class Reference

Inheritance diagram for game::CommandGameStart:



Public Member Functions

- **CommandGameStart** (GameDialog ∗gamedialog)
- virtual void **execute** ()

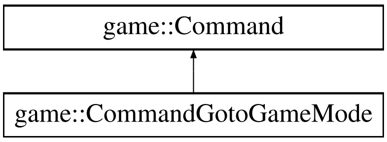
Additional Inherited Members

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

- commandgamestart.h
- commandgamestart.cpp

5.14 game::CommandGotoGameMode Class Reference

Inheritance diagram for game::CommandGotoGameMode:



Public Member Functions

- **CommandGotoGameMode** (GameDialog ∗gamedialog)
- virtual void **execute** ()

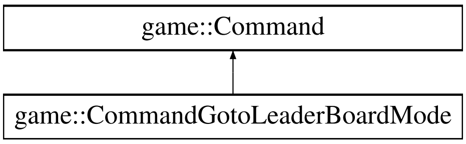
Additional Inherited Members

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

- commandgotogamemode.h
- commandgotogamemode.cpp

5.15 game::CommandGotoLeaderBoardMode Class Reference

Inheritance diagram for game::CommandGotoLeaderBoardMode:



Public Member Functions

- **CommandGotoLeaderBoardMode** (GameDialog ∗gamedialog)
- virtual void **execute** ()

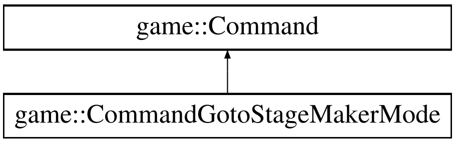
Additional Inherited Members

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

- commandgotoleaderboardmode.h
- commandgotoleaderboardmode.cpp

5.16 game::CommandGotoStageMakerMode Class Reference

Inheritance diagram for game::CommandGotoStageMakerMode:



Public Member Functions

- **CommandGotoStageMakerMode** (GameDialog ∗gamedialog)
- virtual void **execute** ()

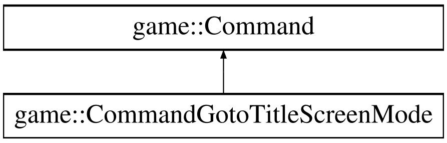
Additional Inherited Members

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

- commandgotostagemakermode.h
- commandgotostagemakermode.cpp

5.17 game::CommandGotoTitleScreenMode Class Reference

Inheritance diagram for game::CommandGotoTitleScreenMode:



Public Member Functions

- **CommandGotoTitleScreenMode** (GameDialog ∗gamedialog)
- virtual void **execute** ()

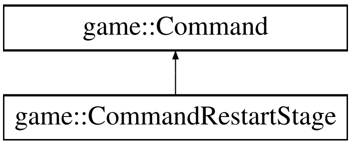
Additional Inherited Members

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

- commandgototitlescreenmode.h
- commandgototitlescreenmode.cpp

5.18 game::CommandRestartStage Class Reference

Inheritance diagram for game::CommandRestartStage:



Public Member Functions

- **CommandRestartStage** (GameDialog ∗gamedialog)
- virtual void **execute** ()

Additional Inherited Members

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

- commonrestartstage.h
- commonrestartstage.cpp

5.19 game::Config Class Reference

Public Member Functions

- QString **get_name** ()
- double **get_scale** ()
- int **get_startpos** ()
- QStringList **get_instructs** ()
- std::vector< QList< SwarmInfo > > **getSwarmList** ()
- int **get_frames** ()
- int **get_SCALEDWIDTH** ()
- int **get_SCALEDHEIGHT** ()

Static Public Member Functions

- static Config ∗ **getInstance** ()

Public Attributes

- bool **shipUseXwing**

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

- config.h
- config.cpp

5.20 game::Cursor Class Reference

Public Member Functions

- **Cursor** (GameDialog ∗gDialog)
- CursorState ∗ **getCurState** ()
- void **setCursorState** (CURSOR_STATE state)

Public Attributes

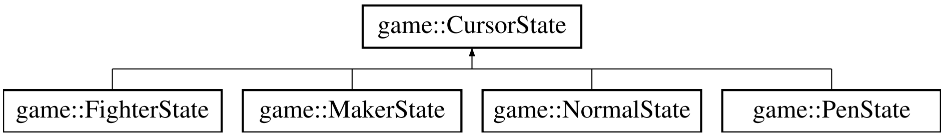
- int **radius**
- CURSOR_STATE **state**
- CURSOR_STATE **preState**
- CursorState ∗ **currentState**
- std::map< CURSOR_STATE, CursorState ∗ > **cursorStatesList**

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

- cursor.h
- cursor.cpp

5.21 game::CursorState Class Reference

Inheritance diagram for game::CursorState:



Public Member Functions

- **CursorState** (Cursor ∗c, GameDialog ∗gDialog)
- virtual void **processMouseEvent** (QMouseEvent ∗event)=0
- virtual void **processMousePress** (QMouseEvent ∗event)=0
- virtual void **processMouseRelease** (QMouseEvent ∗event)=0
- virtual void **updateCursorDisplay** ()=0
- virtual void **update** ()=0

Public Attributes

- int **cursorX**
- int **cursorY**

Protected Attributes

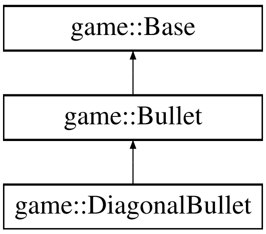
- Cursor ∗ **cursor**
- GameDialog ∗ **gDialog**
- bool **leftPressing**

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

- cursorstate.h

5.22 game::DiagonalBullet Class Reference

Inheritance diagram for game::DiagonalBullet:



Public Member Functions

- **DiagonalBullet** (QPixmap image, int x, int y, int bullet_velocity, bool friendly, bool right)
- void **move** ()

Protected Attributes

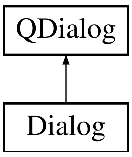
- bool **right**

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

- diagonalbullet.h
- diagonalbullet.cpp

5.23 Dialog Class Reference

Inheritance diagram for Dialog:



Public Member Functions

- **Dialog** (QWidget ∗parent=0)

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

- dialog.h
- dialog.cpp

5.24 Explosion Class Reference

Public Member Functions

- **Explosion** (int x, int y, int scaledToWidth, ExplosionType type)
- void **draw** (QPainter ∗painter)
- void **nextFrame** ()

Public Attributes

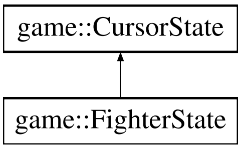
- int **x**
- int **y**
- bool **finished**

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

- explosion.h
- explosion.cpp

5.25 game::FighterState Class Reference

Inheritance diagram for game::FighterState:



Public Member Functions

- **FighterState** (Cursor ∗c, GameDialog ∗dialog)
- void **processMouseEvent** (QMouseEvent ∗event)
- void **processMousePress** (QMouseEvent ∗event)
- void **processMouseRelease** (QMouseEvent ∗event)
- void **updateCursorDisplay** ()
- void **draw** (QPainter ∗p)
- void **update** ()
- void **setCursorDisplay** (bool normal)

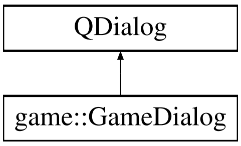
Additional Inherited Members

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

- fighterstate.h
- fighterstate.cpp

5.26 game::GameDialog Class Reference

Inheritance diagram for game::GameDialog:



Public Slots

- void **nextFrame** ()
- void **showScore** ()

Public Member Functions

- **GameDialog** (QWidget ∗parent=nullptr)
- void **generateAliens** (const QList< SwarmInfo > &swarms)
- void **paintEvent** (QPaintEvent ∗event)
- void **updateBullets** ()
- void **checkSwarmCollisions** (AlienBase ∗&root)
- void **requestName** (QString info)
- void **keyPressEvent** (QKeyEvent ∗event)
- void **keyReleaseEvent** (QKeyEvent ∗event)
- void **mousePressEvent** (QMouseEvent ∗event)
- void **mouseReleaseEvent** (QMouseEvent ∗event)
- void **mouseMoveEvent** (QMouseEvent ∗event)
- int **get_collided_swarm** (Bullet ∗&b, AlienBase ∗&root)
- int **get_collided** (Bullet ∗&b, AlienBase ∗&root)
- void **addBullets** (const QList< Bullet ∗> &list)
- int **countAliens** (AlienBase ∗root)
- bool **updateBullets_barrierChkHelper** (int x, int y)
- void **printDebugInfo** (QPainter ∗p)
- void **initCommands** ()

Static Public Member Functions

- static int **randInt** (int low, int high)
- static void **SeedRandInt** ()

Public Attributes

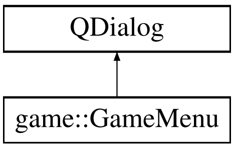
- QTimer ∗ **timer**
- Ship ∗ **ship**
- std::vector< Bullet ∗ > **bullets**
- AlienBase ∗ **swarms**
- QSoundEffect **shipFiringSound**
- int **next_instruct**
- int **frames**
- const int **WIDTH** = 800
- const int **HEIGHT** = 600
- int **SCALEDWIDTH**
- int **SCALEDHEIGHT**
- int **STATUSBARHEIGHT**
- bool **paused**
- Menu ∗ **menu**
- StageMaker **stageMaker**
- bool **debugMode**
- double **timerModifier**
- int **powerUpDropRate**
- int **gameScore**
- bool **legacyMode**
- int **curStageNum**
- QRect **stageTransitionBox**
- bool **stageTransition**
- std::vector< Explosion > **explosions**
- std::vector< BarrierBlock > **barriers**
- std::vector< Powerup > **powerups**
- LaserBeam **laserBeam**
- StatusBar **statusBar**
- Config ∗ **c**
- LeaderBoard **leaderBoard**
- GameMenu **gameMenu**
- LeaderBoardNameRequest **leaderBoardNameRequest**
- bool **playerOverride**
- std::map< int, bool > **pressedKeys**
- Background **bg**
- Cursor **cursor**
- GAME_STATUS **currentState**
- std::unique_ptr< Command > **commandGameStart**
- std::unique_ptr< Command > **commandGamePause**
- std::unique_ptr< Command > **commandClearStage**
- std::unique_ptr< Command > **commandRestartStage**
- std::unique_ptr< Command > **commandGoToTitleScreenMode**
- std::unique_ptr< Command > **commandGoToGameMode**
- std::unique_ptr< Command > **commandGoToStageMakerMode**
- std::unique_ptr< Command > **commandGoToLeaderBoardMode**

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

- gamedialog.h
- gamedialog.cpp

5.27 game::GameMenu Class Reference

Inheritance diagram for game::GameMenu:



Public Member Functions

- **GameMenu** (GameDialog ∗gDialog, QWidget ∗parent=0)

Public Attributes

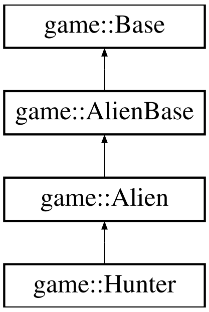
- Ui::GameMenu ∗ **ui**

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

- gamemenu.h
- gamemenu.cpp

5.28 game::Hunter Class Reference

Inheritance diagram for game::Hunter:



Public Member Functions

- **Hunter** (QPixmap image, int x, int y, int velocity, int score, QString baseType, Base &ship, int stray)
- void **move** (QString direction)
- QList< Bullet ∗ > **shoot** (QString type)
- QList< Bullet ∗ > **react** ()

Protected Member Functions

- QString **calculateDirection** ()

Protected Attributes

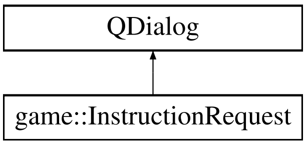
- int **stray**
- Base & **ship**
- QString **baseType**

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

- hunter.h
- hunter.cpp

5.29 game::InstructionRequest Class Reference

Inheritance diagram for game::InstructionRequest:



Public Member Functions

- **InstructionRequest** (QWidget ∗parent=0)

Public Attributes

- QString **instructions**
- SMakerPlacedObject ∗ **instructionBox**

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

- instructionrequest.h
- instructionrequest.cpp

5.30 game::LaserBeam Struct Reference

Public Attributes

- int **originX**
- int **originY**
- int **width**
- bool **exists**

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

- bullet.h

5.31 game::LeaderBoard Class Reference

Public Member Functions

- void **init** (int ScreenWidth, int ScreenHeight, QString filename)
- void **draw** (QPainter ∗p)
- void **update** ()
- void **reset** ()

Public Attributes

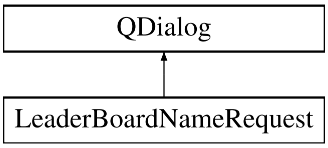
- QRect **leaderBoardTitle**
- std::vector< std::pair< QRect, QStringList > > **players**
- bool **finished**
- QString **filename**
- int **ScreenWidth**
- int **ScreenHeight**

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

- leaderboard.h
- leaderboard.cpp

5.32 LeaderBoardNameRequest Class Reference

Inheritance diagram for LeaderBoardNameRequest:



Public Member Functions

- **LeaderBoardNameRequest** (QString filename, game::GameDialog ∗gDialog, QWidget ∗parent=0)

Public Attributes

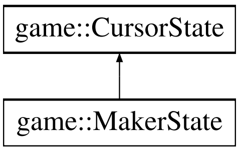
- Ui::leaderBoardNameRequest ∗ **ui**
- QString **filename**
- game::GameDialog ∗ **gDialog**

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

- leaderboardnamerequest.h
- leaderboardnamerequest.cpp

5.33 game::MakerState Class Reference

Inheritance diagram for game::MakerState:



Public Member Functions

- **MakerState** (Cursor ∗c, GameDialog ∗dialog)
- void **processMouseEvent** (QMouseEvent ∗event)
- void **processMousePress** (QMouseEvent ∗event)
- void **processMouseRelease** (QMouseEvent ∗event)
- void **updateCursorDisplay** ()
- void **draw** (QPainter ∗p)
- void **update** ()

Additional Inherited Members

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

- makerstate.h
- makerstate.cpp

5.34 game::Menu Class Reference

Public Member Functions

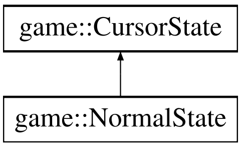
- **Menu** (QWidget ∗parent, QString name, int &playeScore, QList< QPair< QString, int >> scores)
- void **displayMenu** (bool paused)
- void **openScore** ()

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

- menu.h
- menu.cpp

5.35 game::NormalState Class Reference

Inheritance diagram for game::NormalState:



Public Member Functions

- **NormalState** (Cursor ∗c, GameDialog ∗gDialog)
- void **processMouseEvent** (QMouseEvent ∗event)
- void **processMousePress** (QMouseEvent ∗event)
- void **processMouseRelease** (QMouseEvent ∗event)
- void **updateCursorDisplay** ()
- void **draw** (QPainter ∗p)
- void **update** ()

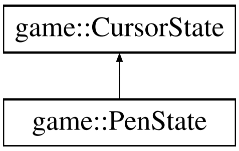
Additional Inherited Members

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

- normalstate.h
- normalstate.cpp

5.36 game::PenState Class Reference

Inheritance diagram for game::PenState:



Public Member Functions

- **PenState** (Cursor ∗c, GameDialog ∗gDialog)
- void **processMouseEvent** (QMouseEvent ∗event)
- void **processMousePress** (QMouseEvent ∗event)
- void **processMouseRelease** (QMouseEvent ∗event)
- void **updateCursorDisplay** ()
- void **draw** (QPainter ∗p)
- void **update** ()

Additional Inherited Members

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

- penstate.h
- penstate.cpp

5.37 game::Powerup Class Reference

Public Member Functions

- **Powerup** (PowerupType type, int x, int y, int radius)
- void **draw** (QPainter ∗p)
- void **update** ()
- int **x** ()
- int **y** ()

Static Public Member Functions

- static Powerup **generateRandomPowerup** (int x, int y, int radius)

Public Attributes

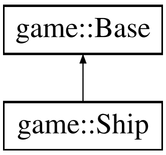
- int **radius**
- PowerupType **type**
- QPixmap **pixmap**

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

- powerup.h
- powerup.cpp

5.38 game::Ship Class Reference

Inheritance diagram for game::Ship:



Public Member Functions

- **Ship** (QPixmap image, double scale, int x, int y)
- Bullet ∗ **shoot** ()
- void **move_left** ()
- void **move_right** ()
- void **update** ()

Public Attributes

- bool **dead**
- CannonType **cannonType**
- int **cannonAmmo**
- bool **machineGunShootAtLeft**

Additional Inherited Members

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

- ship.h
- ship.cpp

5.39 game::SMakerPlacedObject Struct Reference

Public Attributes

- QRect **hitBox**
- SMakerObjectType **type**
- QPixmap **pixmap**
- bool **connected**
- QPoint **connectedPoint**
- QString **instructions**

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

- stagemaker.h

5.40 game::StageMaker Class Reference

Public Member Functions

- **StageMaker** (GameDialog ∗gDialog)
- void **init** ()
- void **draw** (QPainter ∗p)
- void **update** ()
- void **buttonPressed** ()
- void **buttonReleased** ()
- void **clearAll** ()
- void **testStage** ()

Public Attributes

- std::map< SMakerObjectType, SMakerPlacedObject > **objectTemplate**
- QPoint **lineOrigin**
- QRect **clearAllBtn**
- QRect **testStageBtn**
- SMakerObjectType **holdingObject**
- GameDialog ∗ **gDialog**

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

- stagemaker.h
- stagemaker.cpp

5.41 star Struct Reference

Public Attributes

- int **x**
- int **y**
- QColor **color**
- int **speed**

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

- background.h

5.42 game::StatusBar Class Reference

Public Member Functions

- **StatusBar** (GameDialog ∗dialog)
- void **draw** (QPainter ∗p)
- void **update** ()
- void **buildBrush** ()

Public Attributes

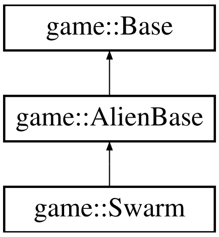
- double **plasmaEnergy**
- double **barrierEnergy**
- GameDialog ∗ **gd**
- QRect **containerOuter**
- QRect **containerInner**
- QRect **statusBar**
- QRect **plasmaBar**
- QRect **barrierBar**
- QBrush **statusBarBrush**
- QBrush **plasmaBarBrush**
- QBrush **barrierBarBrush**
- bool **plasmaDrained**

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

- statusbar.h
- statusbar.cpp

5.43 game::Swarm Class Reference

Inheritance diagram for game::Swarm:



Public Member Functions

- Swarm (SwarmInfo &swarmInfo, Base &ship)
- bool **add** (AlienBase ∗toAdd)
- void **remove** (AlienBase ∗toDelete)
- void **move** (QString direction)
- QList< Bullet ∗ > **shoot** (QString type)
- int **get_score** () const
- QList< AlienBase ∗ > **getAliens** () const
- QList< Bullet ∗ > **react** ()
- virtual void **paint** (QPainter &painter)

Additional Inherited Members

5.43.1 Constructor & Destructor Documentation

5.43.1.1 Swarm()

```
game::Swarm::Swarm (
            SwarmInfo & swarmInfo,
            Base & ship )
```

CONSTRUCTOR

FOR AN EMPTY SWARM at init (e.g., root node)

it should ALWAYS shoot and move (see max pixels) These following properties for Root swarm nodes are the defaults for swarminfo

- moves list is a single ""
- maxPixels should be 0; because everything should always move it should always shoot, if shootTime == 0; just shoot EVERYTHING because lower swarms (that are not the root) decide for them selves anyway leaf nodes from the root will always be shooting
- Image will be some default e.g., redInvader
- Type is as above, default is red

Else, just fill out the swarm info and positions

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

- swarm.h
- swarm.cpp

5.44 game::SwarmInfo Class Reference

Public Member Functions

- **SwarmInfo** (QString type, QList< QPair< int, int >> positions, QStringList move, int shoot)

Public Attributes

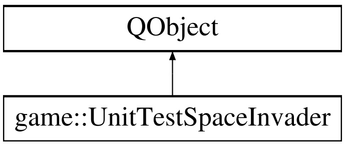
- QPixmap **swarmImage**
- QString **type**
- QList< QPair< int, int > > **positions**
- QStringList **move**
- int **shoot**

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

- swarminfo.h
- swarminfo.cpp

5.45 game::UnitTestSpaceInvader Class Reference

Inheritance diagram for game::UnitTestSpaceInvader:



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

- unittestspaceinvader.h
- unittestspaceinvader.cpp

Index