

Space Invaders

Generated by Doxygen 1.8.4

Sun Jan 12 2014 22:40:13

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Controller	??
Factory	??
Model	??
View	??

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Model::Entity	??
Model::MobileEntity	??
Model::Bullet	??
Model::AlienBullet	??
Model::PlayerBullet	??
Model::ShootingEntity	??
Model::Alien	??
Model::BonusAlien	??
Model::RegularAlien	??
Model::RegularAlienBot	??
Model::RegularAlienMid	??
Model::RegularAlienTop	??
Model::PlayerShip	??
Model::Shield	??
Factory::EntityFactory	??
Factory::AlienBulletFactory	??
Factory::BonusAlienFactory	??
Factory::PlayerBulletFactory	??
Factory::PlayerShipFactory	??
Factory::RegularAlienBotFactory	??
Factory::RegularAlienMidFactory	??
Factory::RegularAlienTopFactory	??
Factory::ShieldFactory	??
Observer	??
Controller::SpaceInvadersController	??
View::SpaceInvadersSFML	??
Model::SpaceInvadersModel	??
View::Textures	??

Chapter 3

Class Index

3.1 Class List

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

Model::Alien	??
Model::AlienBullet	??
Factory::AlienBulletFactory	??
Model::BonusAlien	??
Factory::BonusAlienFactory	??
Model::Bullet	??
Model::Entity	??
Factory::EntityFactory	??
Model::MobileEntity	??
Observer	??
Model::PlayerBullet	??
Factory::PlayerBulletFactory	??
Model::PlayerShip	??
Factory::PlayerShipFactory	??
Model::RegularAlien	??
Model::RegularAlienBot	??
Factory::RegularAlienBotFactory	??
Model::RegularAlienMid	??
Factory::RegularAlienMidFactory	??
Model::RegularAlienTop	??
Factory::RegularAlienTopFactory	??
Model::Shield	??
Factory::ShieldFactory	??
Model::ShootingEntity	??
Controller::SpaceInvadersController	??
Model::SpaceInvadersModel	??
View::SpaceInvadersSFML	??
View::Textures	??

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

/home/jakob/Dropbox/UA/SpacInvaders/src/main.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Controller/spacInvadersController.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Controller/spacInvadersController.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/spacInvadersModel.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/spacInvadersModel.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/alien.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/alienBullet.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/alienBullet.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/bonusAlien.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/bonusAlien.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/bullet.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/entity.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/entity.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/mobileEntity.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/mobileEntity.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/playerBullet.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/playerBullet.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/playerShip.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/playerShip.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/regularAlien.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/regularAlien.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/regularAlienTypes.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/shield.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Entities/shootingEntity.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Factories/abstractEntityFactory.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Factories/concreteEntityFactories.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Model/Factories/concreteEntityFactories.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Other/common.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/Other/observer.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/View/spacInvadersView.cpp	??
/home/jakob/Dropbox/UA/SpacInvaders/src/View/spacInvadersView.h	??
/home/jakob/Dropbox/UA/SpacInvaders/src/View/textures.h	??

Chapter 5

Namespace Documentation

5.1 Controller Namespace Reference

Classes

- class [SpaceInvadersController](#)

5.2 Factory Namespace Reference

Classes

- class [EntityFactory](#)
- class [AlienBulletFactory](#)
- class [BonusAlienFactory](#)
- class [PlayerBulletFactory](#)
- class [PlayerShipFactory](#)
- class [RegularAlienBotFactory](#)
- class [RegularAlienMidFactory](#)
- class [RegularAlienTopFactory](#)
- class [ShieldFactory](#)

5.3 Model Namespace Reference

Classes

- class [Alien](#)
- class [AlienBullet](#)
- class [BonusAlien](#)
- class [Bullet](#)
- class [Entity](#)
- class [MobileEntity](#)
- class [PlayerBullet](#)
- class [PlayerShip](#)
- class [RegularAlien](#)
- class [RegularAlienTop](#)
- class [RegularAlienMid](#)
- class [RegularAlienBot](#)

- class [Shield](#)
- class [ShootingEntity](#)
- class [SpaceInvadersModel](#)

5.4 View Namespace Reference

Classes

- class [SpaceInvadersSFML](#)
- struct [Textures](#)

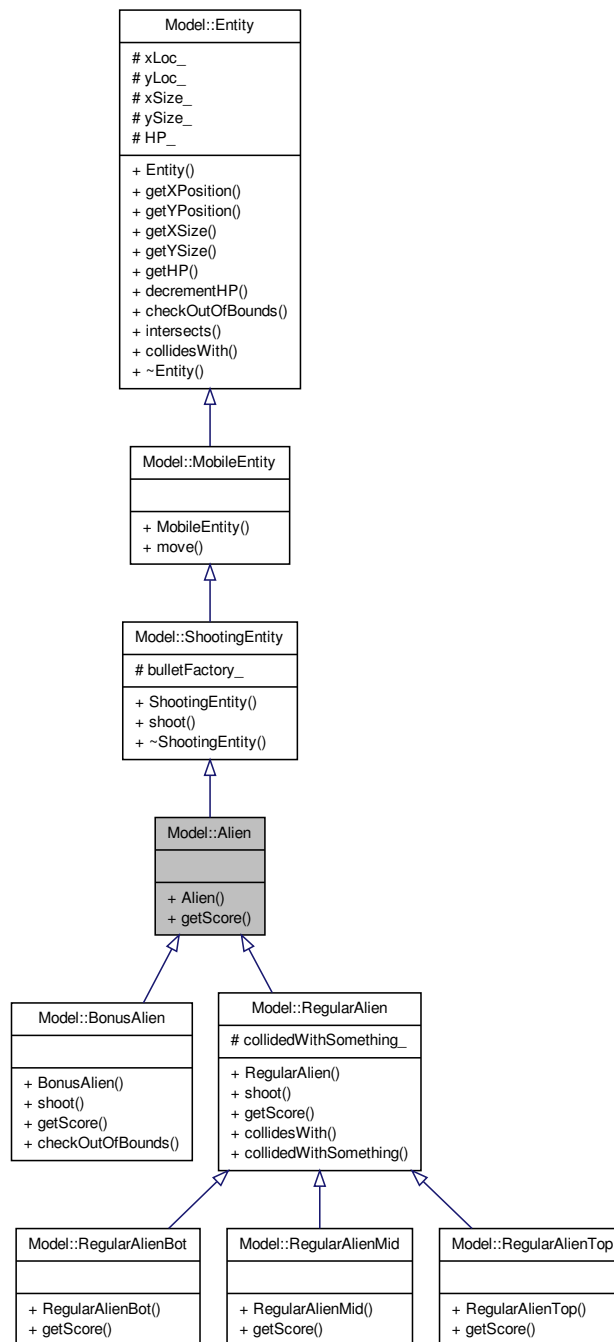
Chapter 6

Class Documentation

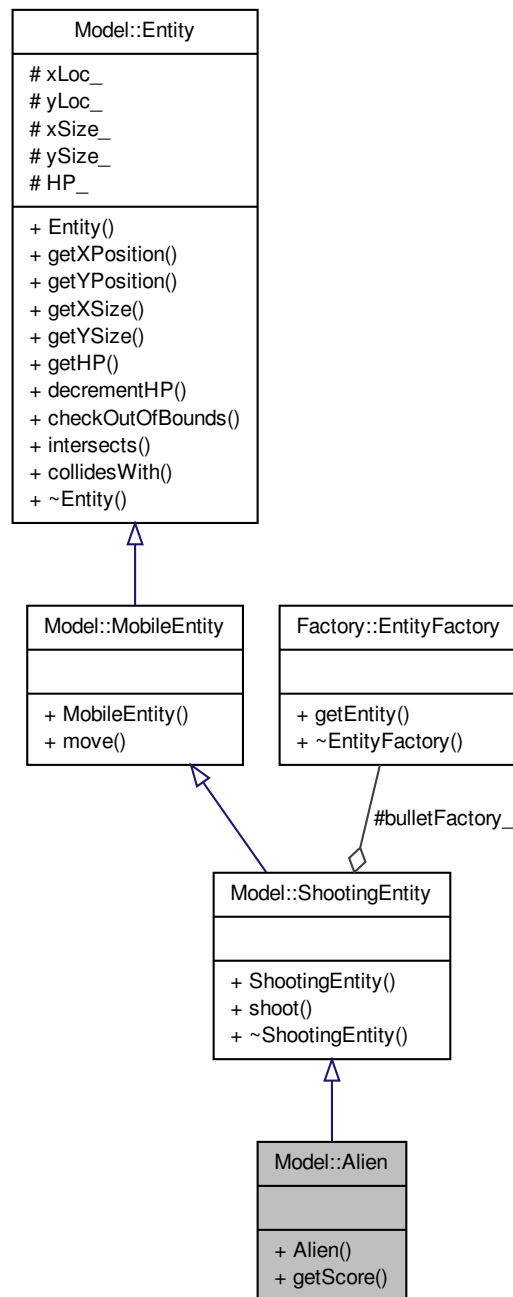
6.1 Model::Alien Class Reference

```
#include <alien.h>
```

Inheritance diagram for Model::Alien:



Collaboration diagram for Model::Alien:



Public Member Functions

- **Alien** (int xLoc, int yLoc, int xSize, int ySize, int HP, **Factory::EntityFactory** *fact)
Constructor.
- virtual int **getScore** ()=0
Gets the score value of the Alien.

Additional Inherited Members

6.1.1 Detailed Description

Abstract class representing any alien

6.1.2 Constructor & Destructor Documentation

6.1.2.1 `Model::Alien::Alien (int xLoc, int yLoc, int xSize, int ySize, int HP, Factory::EntityFactory * fact)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of Alien
<i>yLoc</i>	Vertical location of top of Alien
<i>xSize</i>	Horizontal size of Alien
<i>ySize</i>	Vertical size of Alien
<i>HP</i>	Starting HP
<i>fact</i>	Factory used to create Bullets when shooting

6.1.3 Member Function Documentation

6.1.3.1 `virtual int Model::Alien::getScore ()` `[pure virtual]`

Gets the score value of the [Alien](#).

Returns

the score

Implemented in [Model::RegularAlienBot](#), [Model::RegularAlienMid](#), [Model::BonusAlien](#), [Model::RegularAlien](#), and [Model::RegularAlienTop](#).

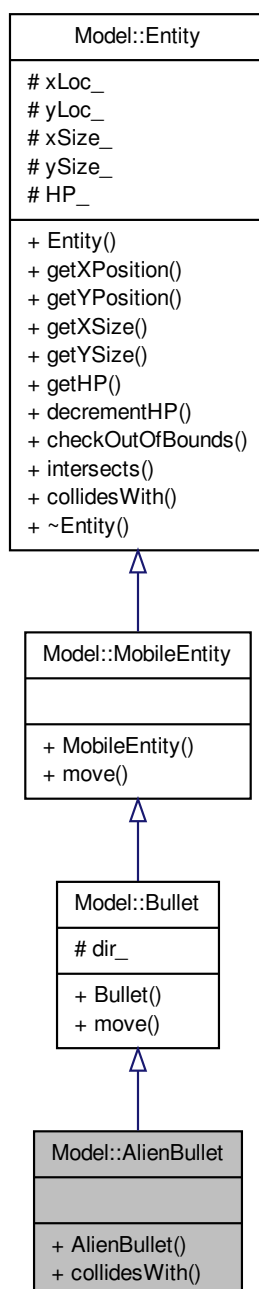
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alien.h`

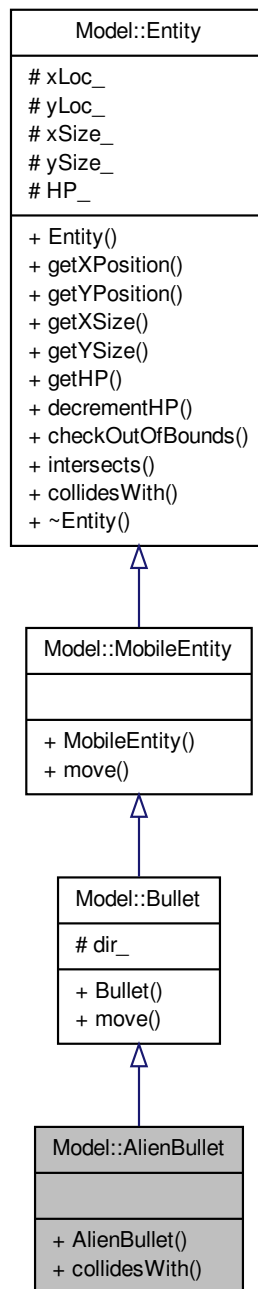
6.2 Model::AlienBullet Class Reference

```
#include <alienBullet.h>
```

Inheritance diagram for Model::AlienBullet:



Collaboration diagram for Model::AlienBullet:



Public Member Functions

- [AlienBullet](#) (int xLoc, int yLoc, int relativeSize)

Constructor.

- bool [collidesWith](#) ([Entity](#) *that)

Checks if [AlienBullet](#) collides with another.

Additional Inherited Members

6.2.1 Detailed Description

Class representing bullet shot by a [RegularAlien](#)

6.2.2 Constructor & Destructor Documentation

6.2.2.1 `Model::AlienBullet::AlienBullet (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of AlienBullet
<i>yLoc</i>	Vertical location of top of AlienBullet
<i>relativeSize</i>	size of the longest side

6.2.3 Member Function Documentation

6.2.3.1 `bool Model::AlienBullet::collidesWith (Entity * that)` `[virtual]`

Checks if [AlienBullet](#) collides with another.

Parameters

<i>that</i>	The other Entity
-------------	----------------------------------

Returns

True if [AlienBullet](#) and [Entity](#) collide

Reimplemented from [Model::Entity](#).

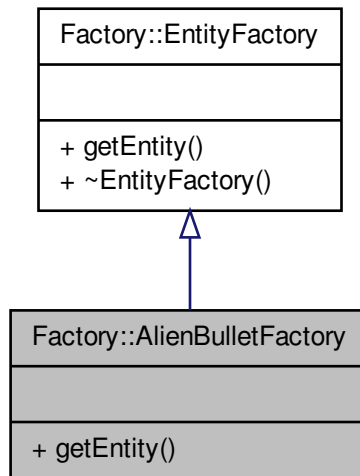
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alienBullet.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alienBullet.cpp`

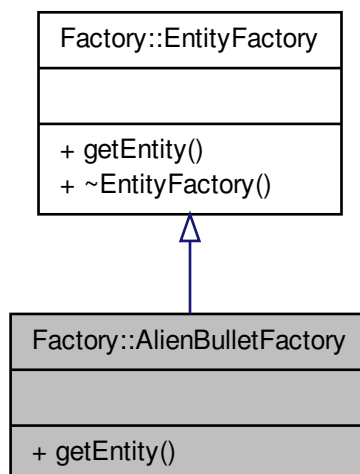
6.3 Factory::AlienBulletFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::AlienBulletFactory:



Collaboration diagram for Factory::AlienBulletFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an AlienBullet.

6.3.1 Detailed Description

Concrete [Factory](#) generating AlienBullets

6.3.2 Member Function Documentation

6.3.2.1 `Model::Entity * Factory::AlienBulletFactory::getEntity (int xLoc, int yLoc, int relativeSize)` `[virtual]`

Generates an AlienBullet.

Parameters

<i>xLoc</i>	Horizontal location of the AlienBullet
<i>yLoc</i>	Vertical location of the AlienBullet
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated AlienBullet

Implements [Factory::EntityFactory](#).

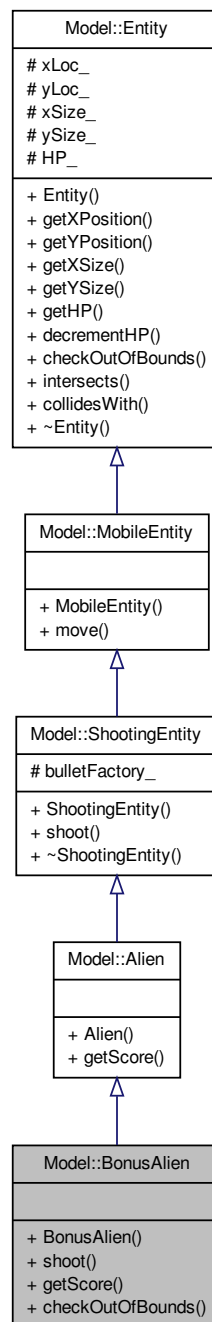
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp`

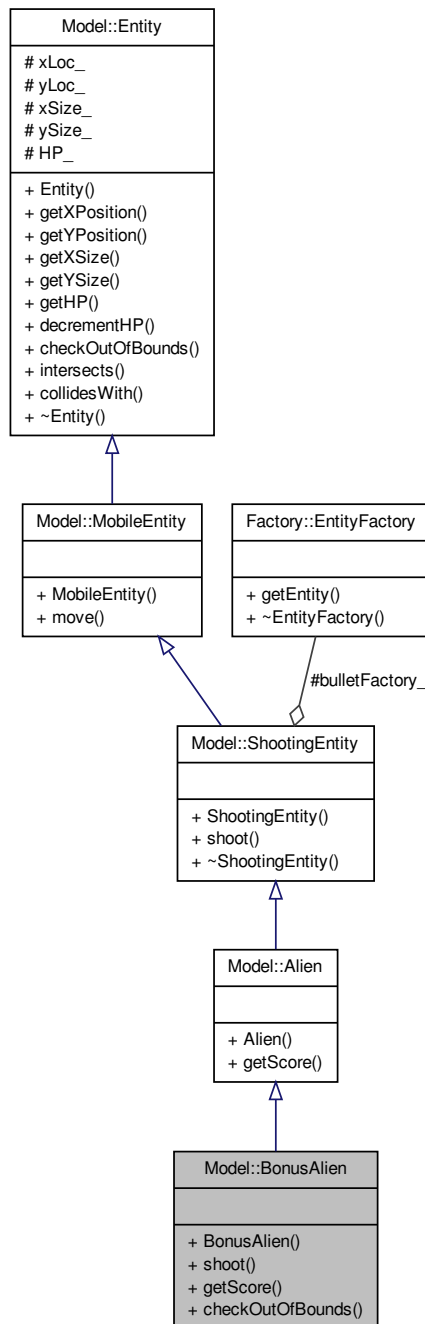
6.4 Model::BonusAlien Class Reference

```
#include <bonusAlien.h>
```

Inheritance diagram for Model::BonusAlien:



Collaboration diagram for Model::BonusAlien:



Public Member Functions

- `BonusAlien` (int xLoc, int yLoc, int relativeSize)
Constructor.
- virtual `Bullet * shoot ()`
Disables BonusAlien from shooting.
- virtual int `getScore ()`

Gets the score value of [BonusAlien](#).

- virtual bool [checkOutOfBounds](#) (int xReso, int yReso)

Checks if [BonusAlien](#) is PARTIALLY within bounds.

Additional Inherited Members

6.4.1 Detailed Description

Class representing the bonus alien appearing periodically at the top of the screen in classic Space Invaders

6.4.2 Constructor & Destructor Documentation

6.4.2.1 `Model::BonusAlien::BonusAlien (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of BonusAlien
<i>yLoc</i>	Vertical location of top of BonusAlien
<i>relativeSize</i>	size of the longest side

6.4.3 Member Function Documentation

6.4.3.1 `bool Model::BonusAlien::checkOutOfBounds (int xReso, int yReso)` `[virtual]`

Checks if [BonusAlien](#) is PARTIALLY within bounds.

Parameters

<i>xReso</i>	Maximum horizontal allowed location (minimum is 0)
<i>yReso</i>	Maximum vertical allowed location (minimum is 0)

Returns

true if not partially within bounds

Reimplemented from [Model::Entity](#).

6.4.3.2 `int Model::BonusAlien::getScore ()` `[virtual]`

Gets the score value of [BonusAlien](#).

Returns

100

Implements [Model::Alien](#).

6.4.3.3 `Bullet * Model::BonusAlien::shoot ()` `[virtual]`

Disables [BonusAlien](#) from shooting.

Returns

nullptr

Implements [Model::ShootingEntity](#).

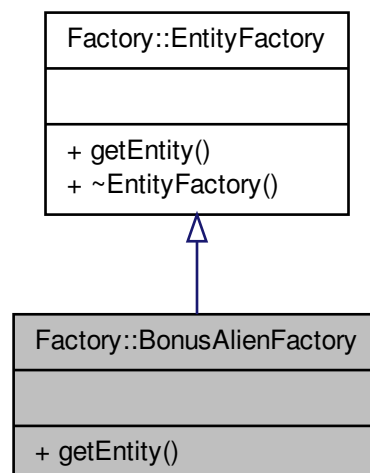
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bonusAlien.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bonusAlien.cpp](#)

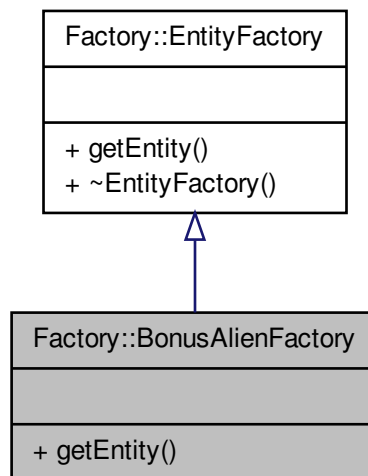
6.5 Factory::BonusAlienFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::BonusAlienFactory:



Collaboration diagram for Factory::BonusAlienFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an BonusAlien.

6.5.1 Detailed Description

Concrete [Factory](#) generating BonusAliens

6.5.2 Member Function Documentation

6.5.2.1 [Model::Entity](#) * [Factory::BonusAlienFactory::getEntity](#) (int xLoc, int yLoc, int relativeSize) [virtual]

Generates an BonusAlien.

Parameters

<i>xLoc</i>	Horizontal location of the BonusAlien
<i>yLoc</i>	Vertical location of the BonusAlien
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated BonusAlien

Implements [Factory::EntityFactory](#).

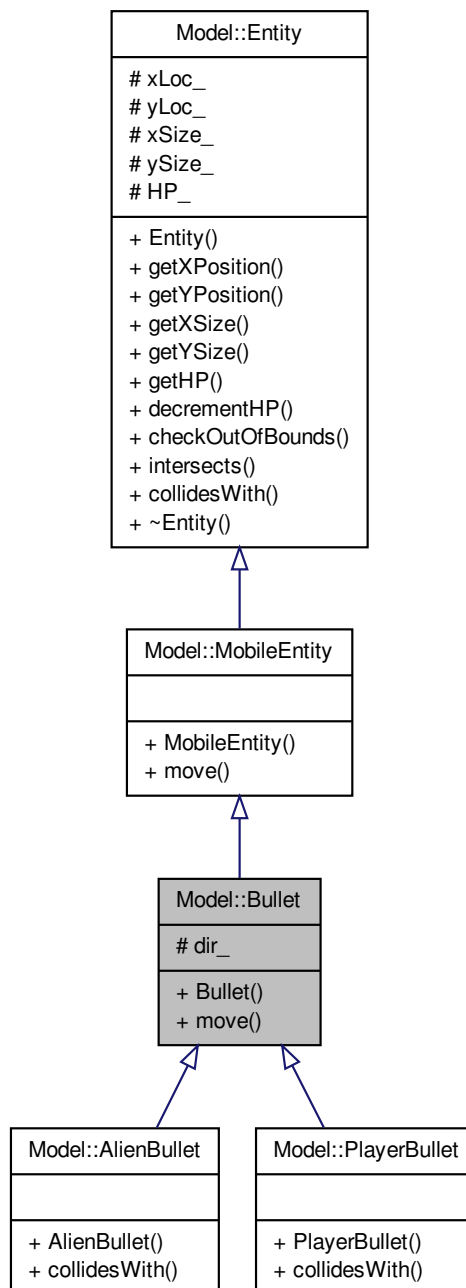
The documentation for this class was generated from the following files:

- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/[concreteEntityFactories.h](#)
- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/[concreteEntityFactories.cpp](#)

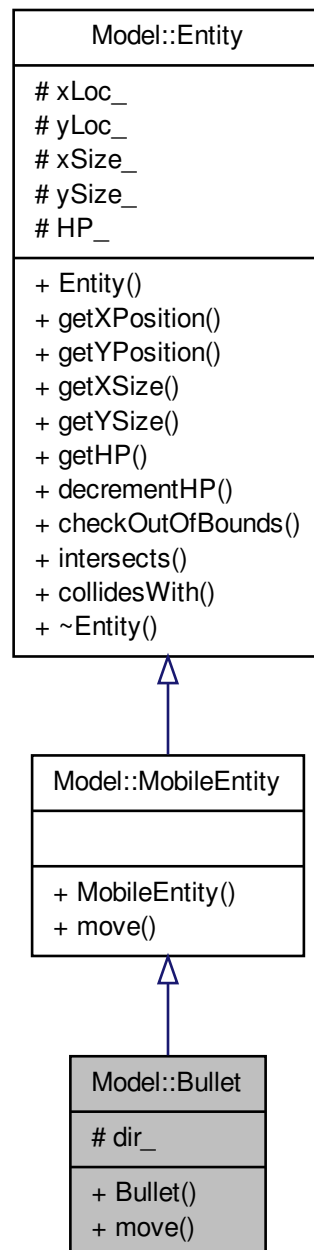
6.6 Model::Bullet Class Reference

```
#include <bullet.h>
```

Inheritance diagram for Model::Bullet:



Collaboration diagram for Model::Bullet:



Public Member Functions

- `Bullet` (int xLoc, int yLoc, int xSize, int ySize, int HP, `Direction` dir)

Constructor.

- void `move` (int dist)

Move the `Bullet` in its direction.

Protected Attributes

- [Direction dir_](#)

6.6.1 Detailed Description

Class representing any kind of bullet in Space Invaders

6.6.2 Constructor & Destructor Documentation

6.6.2.1 `Model::Bullet::Bullet (int xLoc, int yLoc, int xSize, int ySize, int HP, Direction dir)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of Bullet
<i>yLoc</i>	Vertical location of top of Bullet
<i>xSize</i>	Horizontal size of Bullet
<i>ySize</i>	Vertical size of Bullet ;
<i>HP</i>	Starting HP
<i>dir</i>	Direction in which the Bullet is to travel

6.6.3 Member Function Documentation

6.6.3.1 `void Model::Bullet::move (int dist)` `[inline]`

Move the [Bullet](#) in its direction.

Parameters

<i>dist</i>	The distance the Bullet should move
-------------	---

6.6.4 Member Data Documentation

6.6.4.1 `Direction Model::Bullet::dir_` `[protected]`

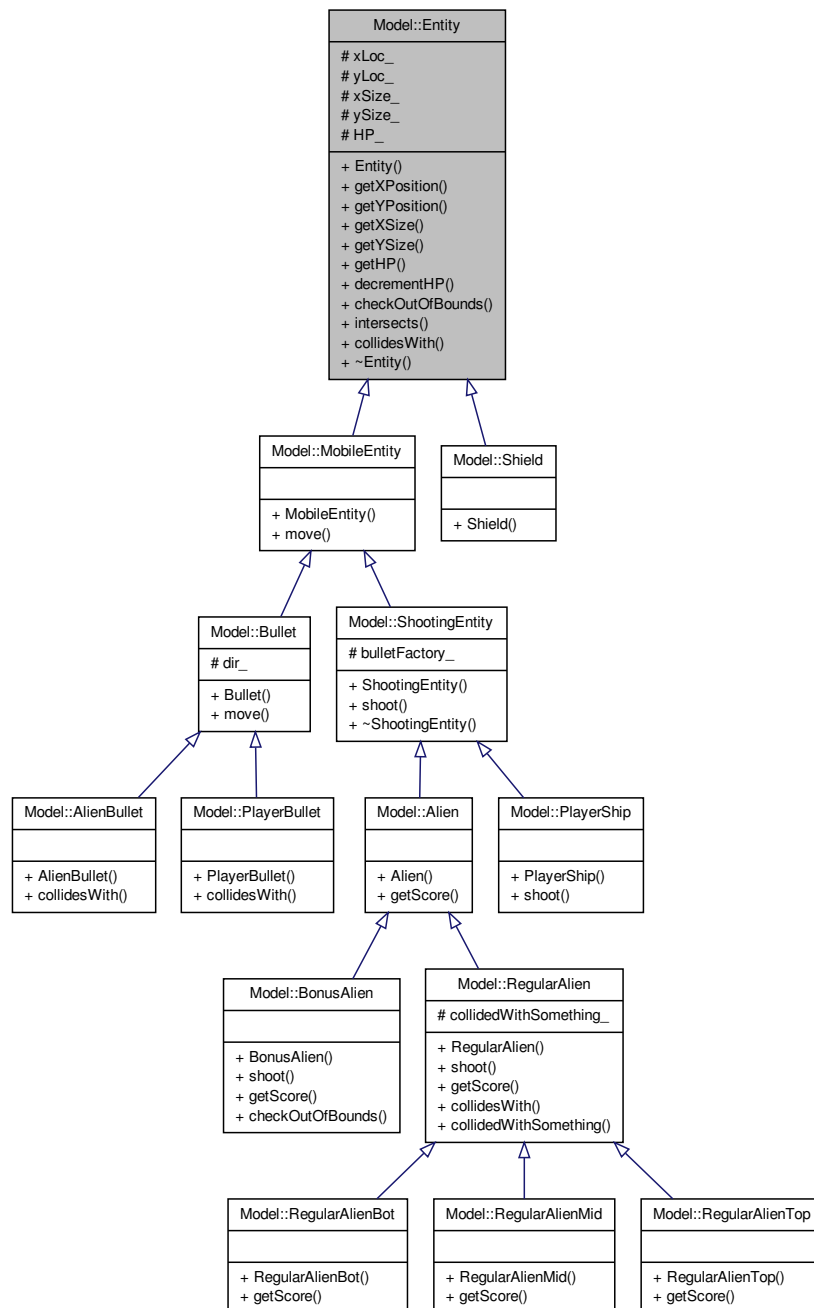
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bullet.h`

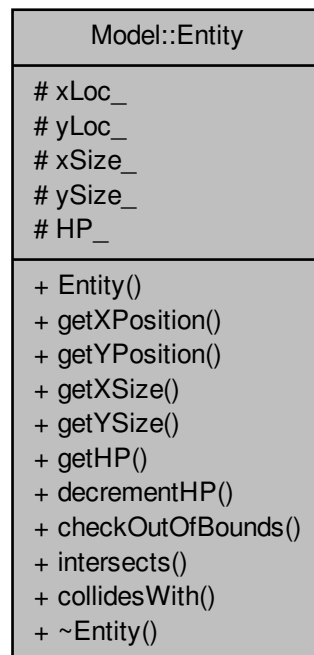
6.7 Model::Entity Class Reference

```
#include <entity.h>
```

Inheritance diagram for Model::Entity:



Collaboration diagram for Model::Entity:



Public Member Functions

- [Entity](#) (int xLoc, int yLoc, int xSize, int ySize, int HP)
Constructor.
- int [getXPosition](#) () const
Gets horizontal position.
- int [getYPosition](#) () const
Gets vertical position.
- int [getXSize](#) () const
Gets horizontal size.
- int [getYSize](#) () const
Gets vertical size.
- int [getHP](#) () const
Gets current HP.
- void [decrementHP](#) ()
Decrements HP by one.
- virtual bool [checkOutOfBounds](#) (int xReso, int yReso)
Checks if the [Entity](#) is still FULLY within the provided bounds.
- bool [intersects](#) (const [Entity](#) *that)
Checks if this [Entity](#) intersects with another.
- virtual bool [collidesWith](#) ([Entity](#) *that)
Checks if [Entity](#) collides with another.
- virtual [~Entity](#) ()
Destructor.

Protected Attributes

- int [xLoc_](#)
- int [yLoc_](#)
- int [xSize_](#)
- int [ySize_](#)
- int [HP_](#)

6.7.1 Detailed Description

Abstract base class representing all entities in the game

6.7.2 Constructor & Destructor Documentation

6.7.2.1 `Model::Entity::Entity (int xLoc, int yLoc, int xSize, int ySize, int HP)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of Entity
<i>yLoc</i>	Vertical location of top of Entity
<i>xSize</i>	Horizontal size of Entity
<i>ySize</i>	Vertical size of Entity
<i>HP</i>	Starting HP

6.7.2.2 `virtual Model::Entity::~Entity ()` `[inline]`, `[virtual]`

Destructor.

6.7.3 Member Function Documentation

6.7.3.1 `bool Model::Entity::checkOutOfBounds (int xReso, int yReso)` `[virtual]`

Checks if the [Entity](#) is still FULLY within the provided bounds.

Parameters

<i>xReso</i>	Maximum horizontal allowed location (minimum is 0)
<i>yReso</i>	Maximum vertical allowed location (minimum is 0)

Returns

true if not fully within bounds

Reimplemented in [Model::BonusAlien](#).

6.7.3.2 `bool Model::Entity::collidesWith (Entity * that)` `[virtual]`

Checks if [Entity](#) collides with another.

Note that intersecting Entities do not necessarily collide

Parameters

<i>that</i>	The other Entity
-------------	----------------------------------

Returns

True if Entities collide

Reimplemented in [Model::RegularAlien](#), [Model::PlayerBullet](#), and [Model::AlienBullet](#).

6.7.3.3 void Model::Entity::decrementHP ()

Decrements HP by one.

Note that this assumes HP is >0!

6.7.3.4 int Model::Entity::getHP () const

Gets current HP.

Returns

Current HP

6.7.3.5 int Model::Entity::getXPosition () const

Gets horizontal position.

Returns

Horizontal position

6.7.3.6 int Model::Entity::getXSize () const

Gets horizontal size.

Returns

Horizontal size

6.7.3.7 int Model::Entity::getYPosition () const

Gets vertical position.

Returns

Vertical position

6.7.3.8 int Model::Entity::getYSize () const

Gets vertical size.

Returns

Vertical size

6.7.3.9 `bool Model::Entity::intersects (const Entity * that)`

Checks if this [Entity](#) intersects with another.

Parameters

<i>that</i>	The other Entity
-------------	----------------------------------

Returns

True if they intersect

6.7.4 Member Data Documentation

6.7.4.1 int Model::Entity::HP_ [protected]

[Entity](#) should be considered destroyed with 0HP, though this class does not enforce this

6.7.4.2 int Model::Entity::xLoc_ [protected]

Position of [Entity](#) is stored with these four (unitless) values xLoc_ and yLoc_ are coordinates of the top left corner xSize_ and ySize_ are the horizontal and vertical sizes Note that this means every [Entity](#) is rectangular

6.7.4.3 int Model::Entity::xSize_ [protected]

6.7.4.4 int Model::Entity::yLoc_ [protected]

6.7.4.5 int Model::Entity::ySize_ [protected]

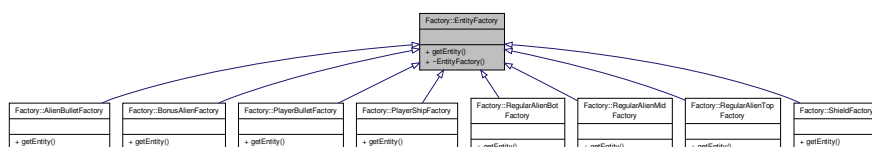
The documentation for this class was generated from the following files:

- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/[entity.h](#)
- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/[entity.cpp](#)

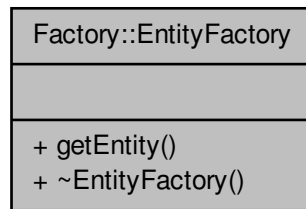
6.8 Factory::EntityFactory Class Reference

```
#include <abstractEntityFactory.h>
```

Inheritance diagram for Factory::EntityFactory:



Collaboration diagram for Factory::EntityFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)=0
Generates an Entity.
- virtual [~EntityFactory](#) ()
Destructor.

6.8.1 Detailed Description

Abstract [Factory](#) for Entities

6.8.2 Constructor & Destructor Documentation

6.8.2.1 virtual Factory::EntityFactory::~~EntityFactory () [inline],[virtual]

Destructor.

6.8.3 Member Function Documentation

6.8.3.1 virtual [Model::Entity](#)* [Factory::EntityFactory::getEntity](#) (int *xLoc*, int *yLoc*, int *relativeSize*) [pure virtual]

Generates an Entity.

Parameters

<i>xLoc</i>	Horizontal location of the Entity
<i>yLoc</i>	Vertical location of the Entity
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated Entity

Implemented in [Factory::ShieldFactory](#), [Factory::RegularAlienTopFactory](#), [Factory::RegularAlienMidFactory](#), [Factory::RegularAlienBotFactory](#), [Factory::PlayerShipFactory](#), [Factory::PlayerBulletFactory](#), [Factory::BonusAlienFactory](#), and [Factory::AlienBulletFactory](#).

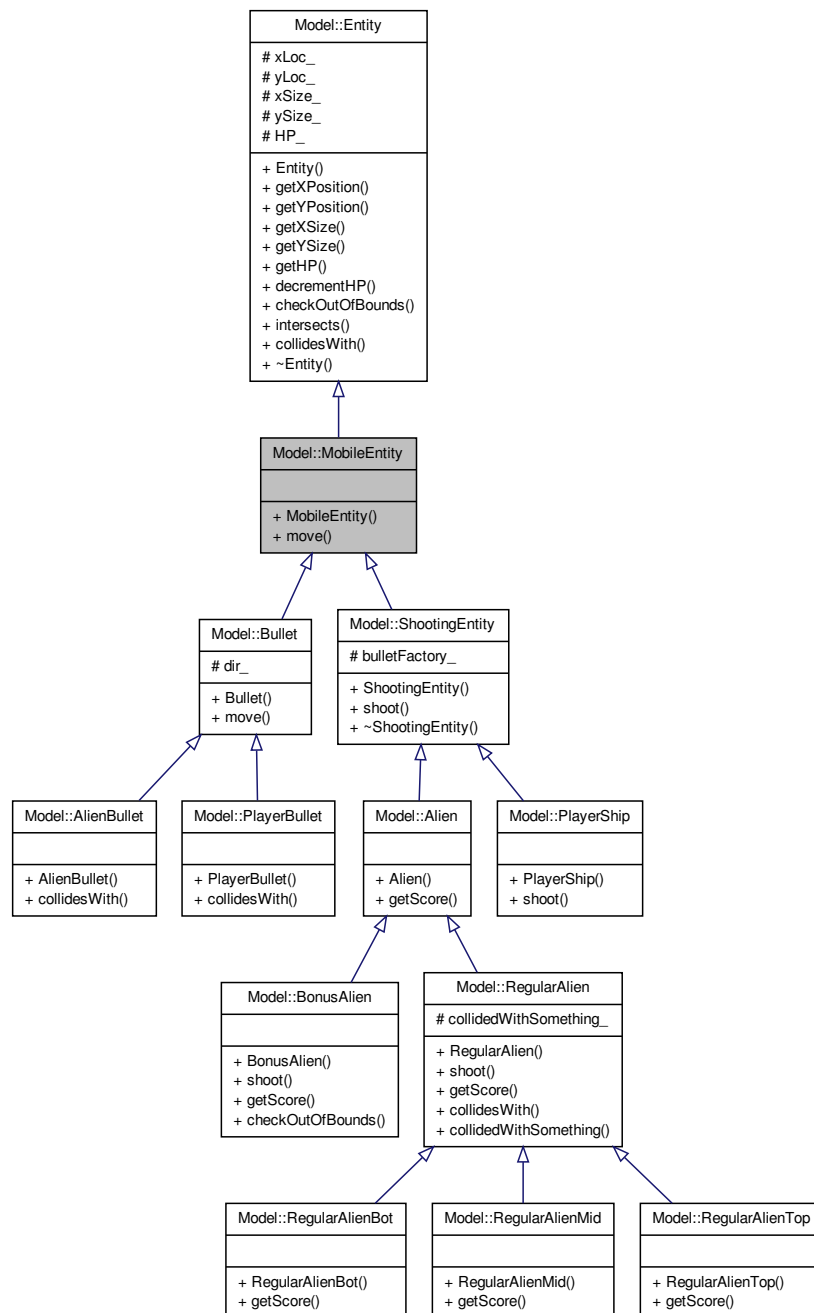
The documentation for this class was generated from the following file:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/abstractEntityFactory.h](#)

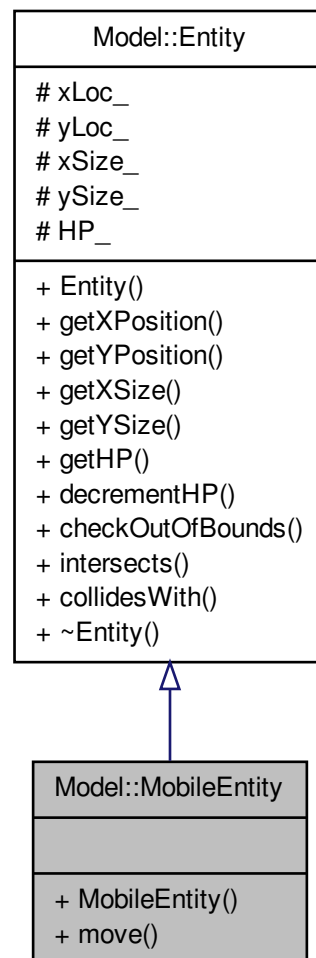
6.9 Model::MobileEntity Class Reference

```
#include <mobileEntity.h>
```

Inheritance diagram for Model::MobileEntity:



Collaboration diagram for Model::MobileEntity:



Public Member Functions

- [MobileEntity](#) (int xLoc, int yLoc, int xSize, int ySize, int HP)
Constructor.
- virtual void [move](#) ([Direction](#) dir, int distance)
Move the [MobileEntity](#).

Additional Inherited Members

6.9.1 Detailed Description

Abstract class representing mobile (==with non-fixed position) entities in the game

6.9.2 Constructor & Destructor Documentation

6.9.2.1 Model::MobileEntity::MobileEntity (int *xLoc*, int *yLoc*, int *xSize*, int *ySize*, int *HP*) [inline]

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of MobileEntity
<i>yLoc</i>	Vertical location of top of MobileEntity
<i>xSize</i>	Horizontal size of MobileEntity
<i>ySize</i>	Vertical size of MobileEntity
<i>HP</i>	Starting HP

6.9.3 Member Function Documentation

6.9.3.1 void Model::MobileEntity::move ([Direction](#) *dir*, int *distance*) [virtual]

Move the [MobileEntity](#).

Parameters

<i>dir</i>	The direction in which the MobileEntity should be moving
<i>distance</i>	The distance the MobileEntity should move

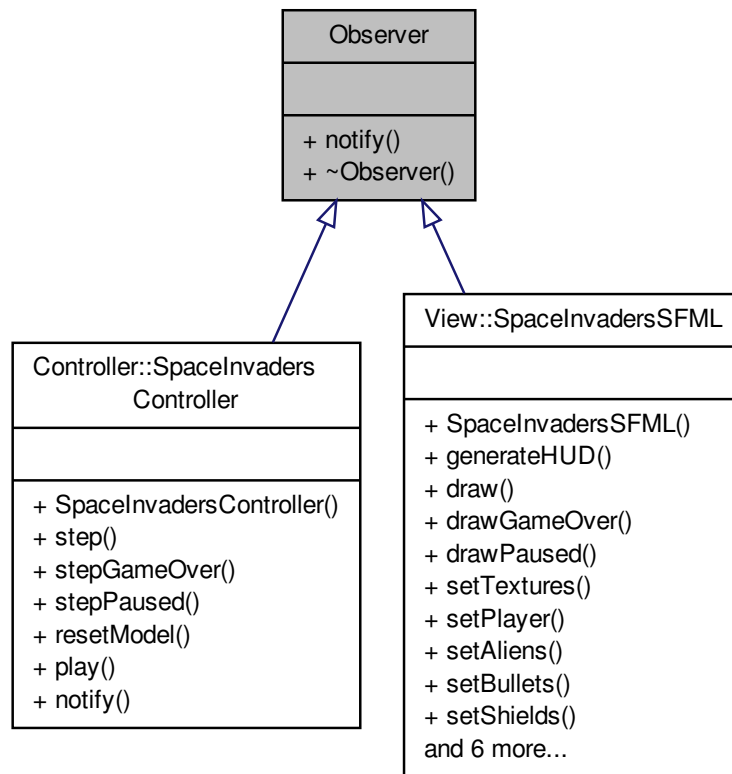
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/mobileEntity.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/mobileEntity.cpp](#)

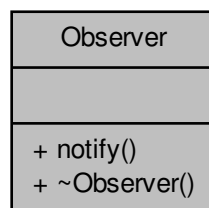
6.10 Observer Class Reference

```
#include <observer.h>
```

Inheritance diagram for Observer:



Collaboration diagram for Observer:



Public Member Functions

- virtual void `notify` (bool gameOver, bool winner)=0
Handles a notification sent by Space Invaders model.
- virtual `~Observer` ()

Destructor.

6.10.1 Detailed Description

Abstract base class for any class that observes the Space Invaders model

6.10.2 Constructor & Destructor Documentation

6.10.2.1 `virtual Observer::~~Observer () [inline],[virtual]`

Destructor.

6.10.3 Member Function Documentation

6.10.3.1 `virtual void Observer::notify (bool gameOver, bool winner) [pure virtual]`

Handles a notification sent by Space Invaders model.

Parameters

<i>gameOver</i>	True if the game is currently over
<i>winner</i>	True if player just won a level

Implemented in [View::SpaceInvadersSFML](#), and [Controller::SpaceInvadersController](#).

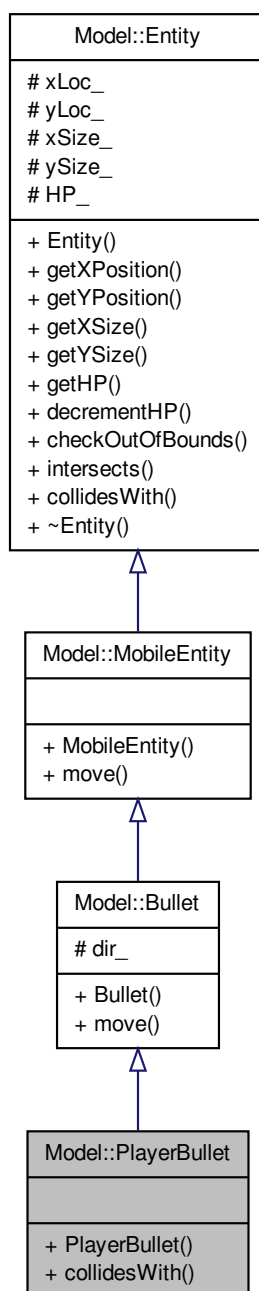
The documentation for this class was generated from the following file:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Other/observer.h](#)

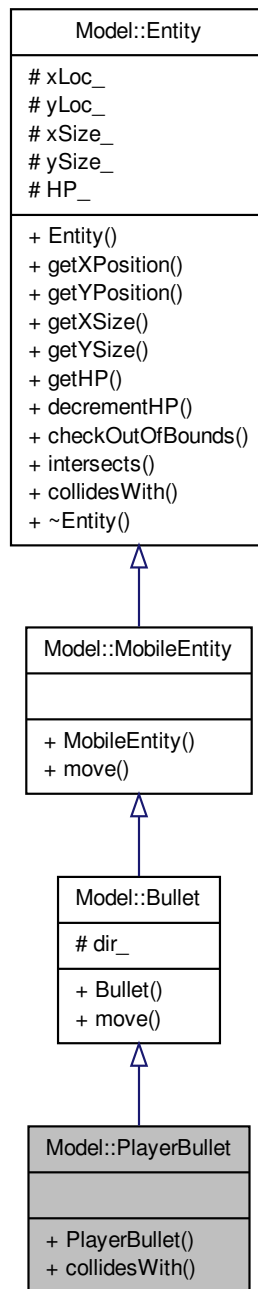
6.11 Model::PlayerBullet Class Reference

```
#include <playerBullet.h>
```

Inheritance diagram for Model::PlayerBullet:



Collaboration diagram for Model::PlayerBullet:



Public Member Functions

- [PlayerBullet](#) (int xLoc, int yLoc, int relativeSize)
Constructor.
- bool [collidesWith](#) ([Entity](#) *that)
Checks if [PlayerBullet](#) collides with another [Entity](#).

Additional Inherited Members

6.11.1 Detailed Description

Class representing bullet shot by player's ship

6.11.2 Constructor & Destructor Documentation

6.11.2.1 `Model::PlayerBullet::PlayerBullet (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of PlayerBullet
<i>yLoc</i>	Vertical location of top of PlayerBullet
<i>relativeSize</i>	size of the longest side

6.11.3 Member Function Documentation

6.11.3.1 `bool Model::PlayerBullet::collidesWith (Entity * that)` `[virtual]`

Checks if [PlayerBullet](#) collides with another [Entity](#).

Note that intersecting Entities do not necessarily collide

Parameters

<i>that</i>	The other Entity
-------------	----------------------------------

Returns

True if [PlayerBullet](#) and [Entity](#) collide

Reimplemented from [Model::Entity](#).

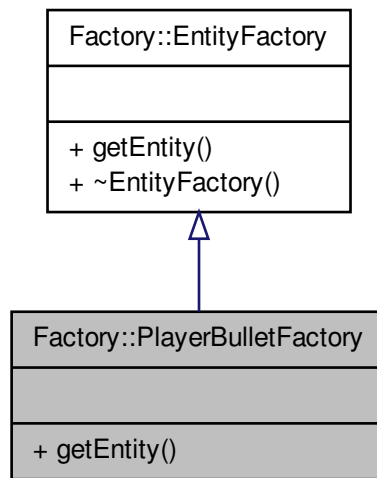
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerBullet.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerBullet.cpp](#)

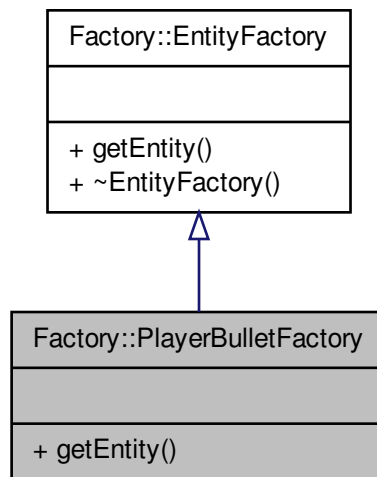
6.12 Factory::PlayerBulletFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::PlayerBulletFactory:



Collaboration diagram for Factory::PlayerBulletFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)

Generates an PlayerBullet.

6.12.1 Detailed Description

Concrete [Factory](#) generating PlayerBullets

6.12.2 Member Function Documentation

6.12.2.1 `Model::Entity * Factory::PlayerBulletFactory::getEntity (int xLoc, int yLoc, int relativeSize)` `[virtual]`

Generates an PlayerBullet.

Parameters

<i>xLoc</i>	Horizontal location of the PlayerBullet
<i>yLoc</i>	Vertical location of the PlayerBullet
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated PlayerBullet

Implements [Factory::EntityFactory](#).

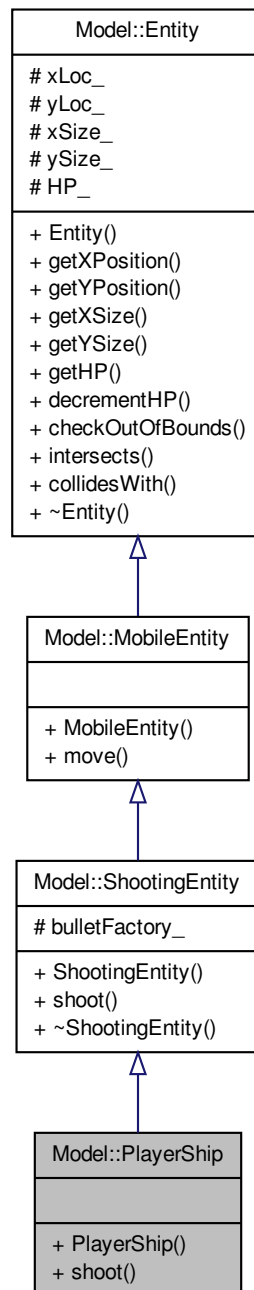
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp`

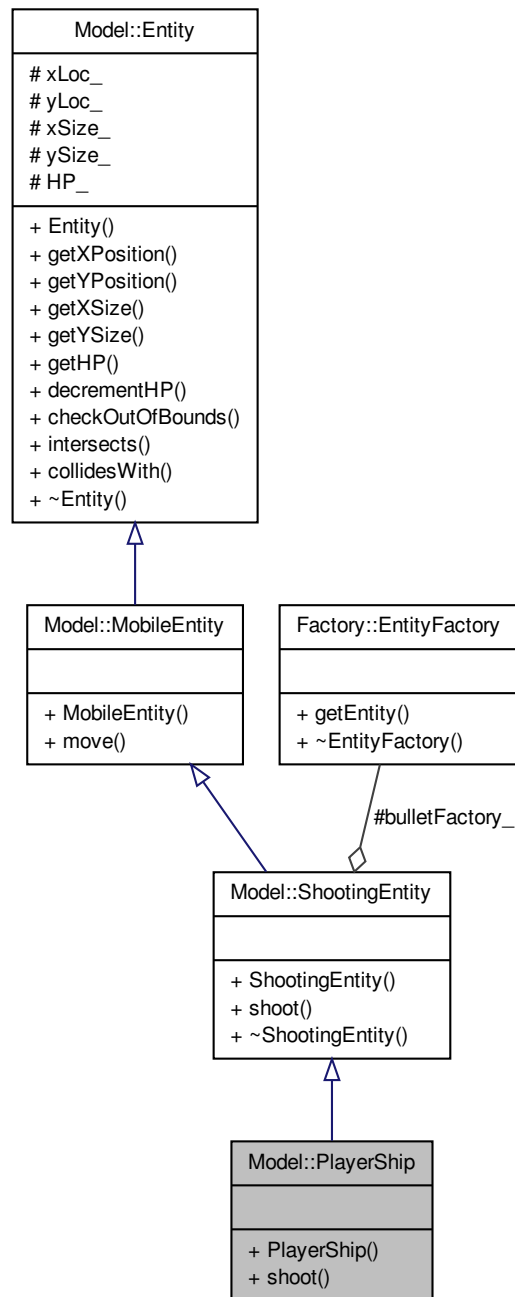
6.13 Model::PlayerShip Class Reference

```
#include <playerShip.h>
```

Inheritance diagram for Model::PlayerShip:



Collaboration diagram for Model::PlayerShip:



Public Member Functions

- [PlayerShip](#) (int xLoc, int yLoc, int relativeSize)

Constructor.

- [Bullet * shoot](#) ()

PlayerShip fires one Bullet.

Additional Inherited Members

6.13.1 Detailed Description

Class representing the controllable ship at the bottom of the screen in classic Space Invaders

6.13.2 Constructor & Destructor Documentation

6.13.2.1 `Model::PlayerShip::PlayerShip (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of PlayerShip
<i>yLoc</i>	Vertical location of top of PlayerShip
<i>relativeSize</i>	size of the longest side
<i>HP</i>	Starting HP

6.13.3 Member Function Documentation

6.13.3.1 `Bullet * Model::PlayerShip::shoot ()` `[virtual]`

[PlayerShip](#) fires one [Bullet](#).

Returns

pointer to the fired [Bullet](#)

Implements [Model::ShootingEntity](#).

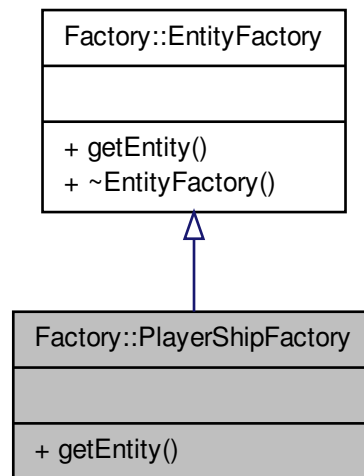
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerShip.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerShip.cpp`

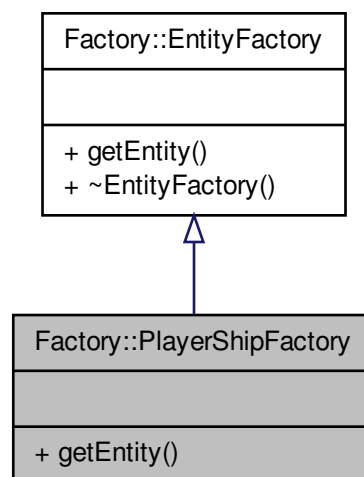
6.14 `Factory::PlayerShipFactory` Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::PlayerShipFactory:



Collaboration diagram for Factory::PlayerShipFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)

Generates an PlayerShip.

6.14.1 Detailed Description

Concrete [Factory](#) generating PlayerShips

6.14.2 Member Function Documentation

6.14.2.1 `Model::Entity * Factory::PlayerShipFactory::getEntity (int xLoc, int yLoc, int relativeSize)` `[virtual]`

Generates an PlayerShip.

Parameters

<i>xLoc</i>	Horizontal location of the PlayerShip
<i>yLoc</i>	Vertical location of the PlayerShip
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated PlayerShip

Implements [Factory::EntityFactory](#).

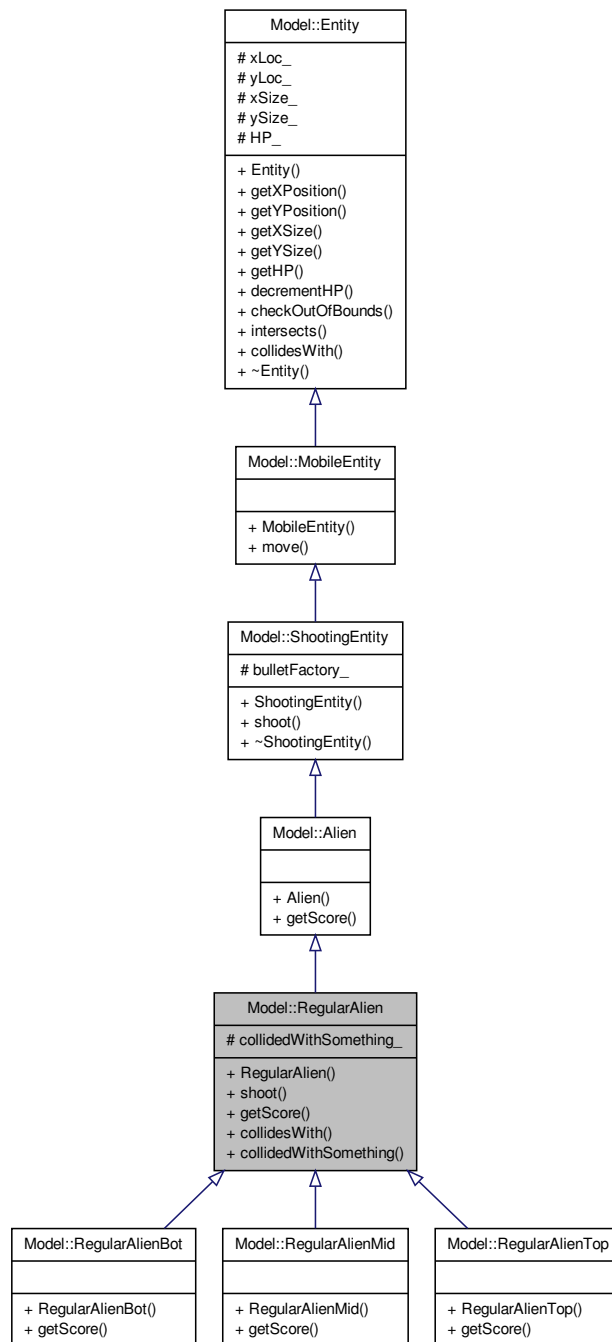
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp`

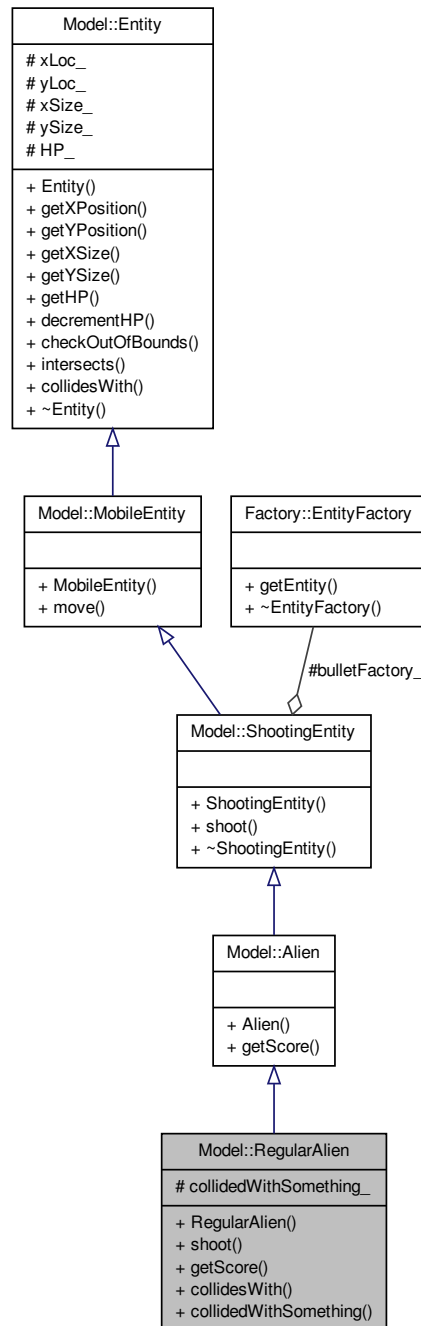
6.15 Model::RegularAlien Class Reference

```
#include <regularAlien.h>
```

Inheritance diagram for Model::RegularAlien:



Collaboration diagram for Model::RegularAlien:



Public Member Functions

- `RegularAlien` (int xLoc, int yLoc, int relativeSize)
Constructor.
- virtual `Bullet * shoot` ()
RegularAlien fires one Bullet.
- virtual int `getScore` ()=0

Gets the score value of the [RegularAlien](#).

- virtual bool [collidesWith](#) ([Entity](#) *entity)

Checks if [RegularAlien](#) collides with another [Entity](#).

- virtual bool [collidedWithSomething](#) () const

: checks if [RegularAlien](#) has collided with something other than a [Bullet](#)

Protected Attributes

- bool [collidedWithSomething_](#)

6.15.1 Detailed Description

Class representing the regular, classic Space Invaders alien

6.15.2 Constructor & Destructor Documentation

6.15.2.1 `Model::RegularAlien::RegularAlien (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of RegularAlien
<i>yLoc</i>	Vertical location of top of RegularAlien
<i>relativeSize</i>	size of the longest side

6.15.3 Member Function Documentation

6.15.3.1 `bool Model::RegularAlien::collidedWithSomething () const` `[virtual]`

: checks if [RegularAlien](#) has collided with something other than a [Bullet](#)

Returns

True if it did

6.15.3.2 `bool Model::RegularAlien::collidesWith (Entity * entity)` `[virtual]`

Checks if [RegularAlien](#) collides with another [Entity](#).

Note that intersecting Entities do not necessarily collide

Parameters

<i>that</i>	The other Entity
-------------	----------------------------------

Returns

True if [RegularAlien](#) and [Entity](#) collide

Reimplemented from [Model::Entity](#).

6.15.3.3 `virtual int Model::RegularAlien::getScore () [pure virtual]`

Gets the score value of the [RegularAlien](#).

Returns

the score

Implements [Model::Alien](#).

Implemented in [Model::RegularAlienBot](#), [Model::RegularAlienMid](#), and [Model::RegularAlienTop](#).

6.15.3.4 `Bullet * Model::RegularAlien::shoot () [virtual]`

[RegularAlien](#) fires one [Bullet](#).

Returns

pointer to the fired [Bullet](#)

Implements [Model::ShootingEntity](#).

6.15.4 Member Data Documentation

6.15.4.1 `bool Model::RegularAlien::collidedWithSomething_ [protected]`

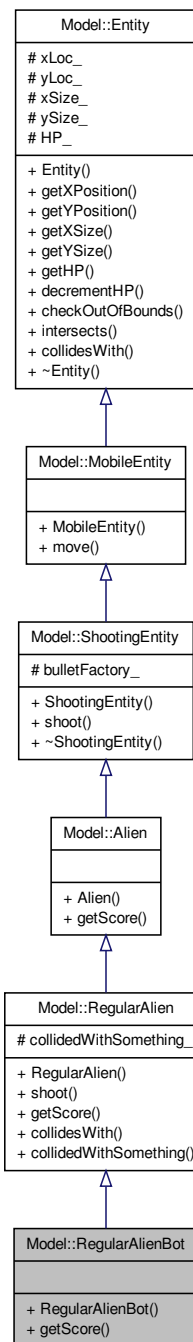
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlien.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlien.cpp](#)

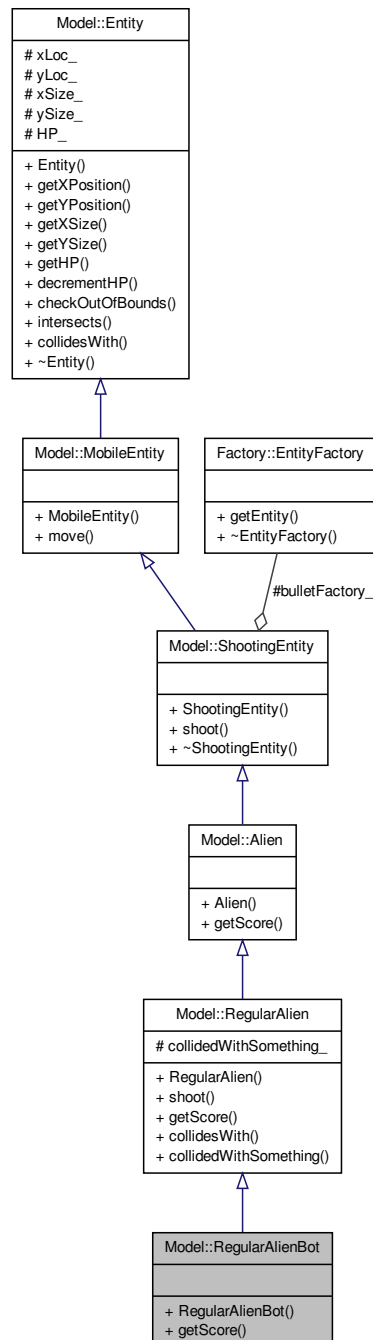
6.16 Model::RegularAlienBot Class Reference

```
#include <regularAlienTypes.h>
```

Inheritance diagram for Model::RegularAlienBot:



Collaboration diagram for Model::RegularAlienBot:



Public Member Functions

- [RegularAlienBot](#) (int xLoc, int yLoc, int relativeSize)

Constructor.

- int [getScore](#) ()

Gets the score value of the [RegularAlienBot](#).

Additional Inherited Members

6.16.1 Detailed Description

Class representing aliens in the two bottom rows in classic Space Invaders

6.16.2 Constructor & Destructor Documentation

6.16.2.1 Model::RegularAlienBot::RegularAlienBot (int *xLoc*, int *yLoc*, int *relativeSize*) `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of RegularAlienBot
<i>yLoc</i>	Vertical location of top of RegularAlienBot
<i>relativeSize</i>	size of the longest side

6.16.3 Member Function Documentation

6.16.3.1 int Model::RegularAlienBot::getScore () `[inline], [virtual]`

Gets the score value of the [RegularAlienBot](#).

Returns

10

Implements [Model::RegularAlien](#).

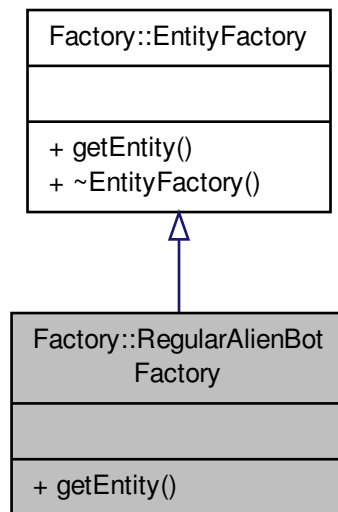
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlienTypes.h`

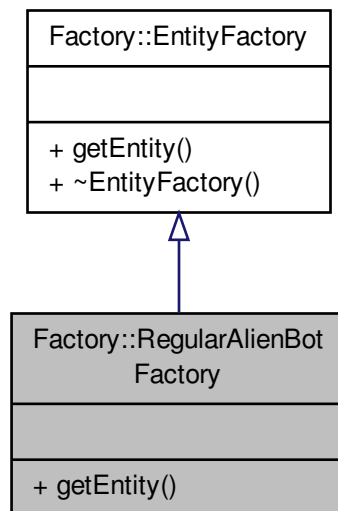
6.17 Factory::RegularAlienBotFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::RegularAlienBotFactory:



Collaboration diagram for Factory::RegularAlienBotFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an RegularAlienBot.

6.17.1 Detailed Description

Concrete [Factory](#) generating RegularAlienBots

6.17.2 Member Function Documentation

6.17.2.1 Model::Entity * Factory::RegularAlienBotFactory::getEntity (int *xLoc*, int *yLoc*, int *relativeSize*) [virtual]

Generates an RegularAlienBot.

Parameters

<i>xLoc</i>	Horizontal location of the RegularAlienBot
<i>yLoc</i>	Vertical location of the RegularAlienBot
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated RegularAlienBot

Implements [Factory::EntityFactory](#).

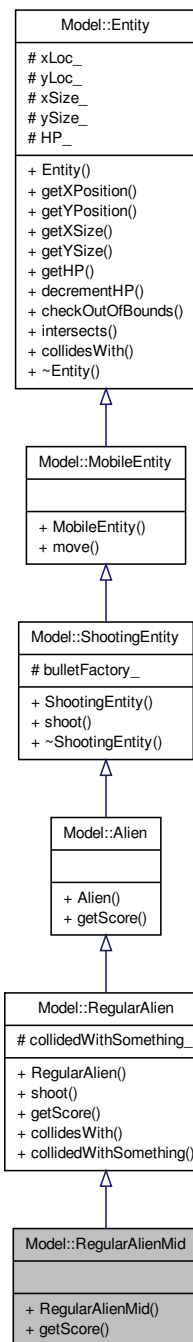
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp](#)

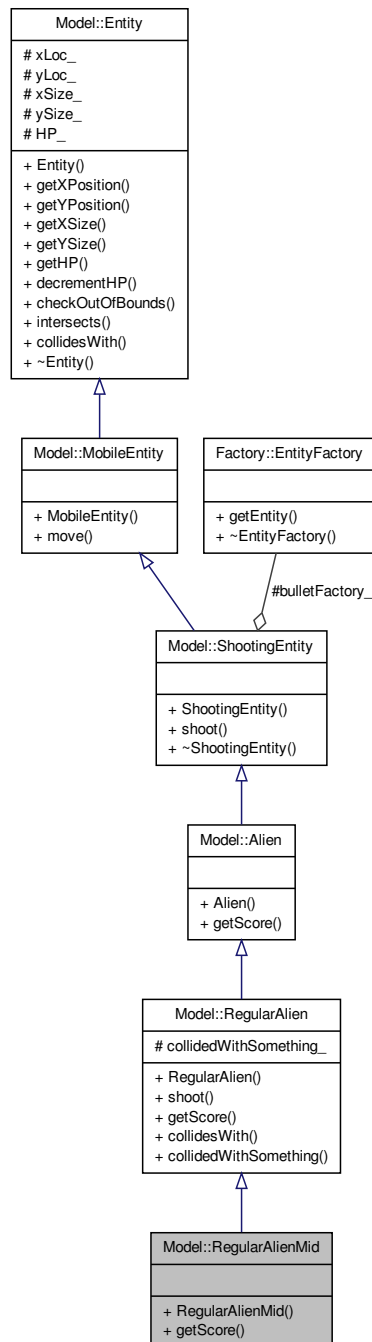
6.18 Model::RegularAlienMid Class Reference

```
#include <regularAlienTypes.h>
```

Inheritance diagram for Model::RegularAlienMid:



Collaboration diagram for Model::RegularAlienMid:



Public Member Functions

- [RegularAlienMid](#) (int xLoc, int yLoc, int relativeSize)

Constructor.

- int [getScore](#) ()

Gets the score value of the RegularAlienMid.

Additional Inherited Members

6.18.1 Detailed Description

Class representing aliens in the second and third row from the top in classic Space Invaders

6.18.2 Constructor & Destructor Documentation

6.18.2.1 `Model::RegularAlienMid::RegularAlienMid (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of RegularAlienMid
<i>yLoc</i>	Vertical location of top of RegularAlienMid
<i>relativeSize</i>	size of the longest side

6.18.3 Member Function Documentation

6.18.3.1 `int Model::RegularAlienMid::getScore ()` `[inline]`, `[virtual]`

Gets the score value of the RegularAlienMud.

Returns

20

Implements [Model::RegularAlien](#).

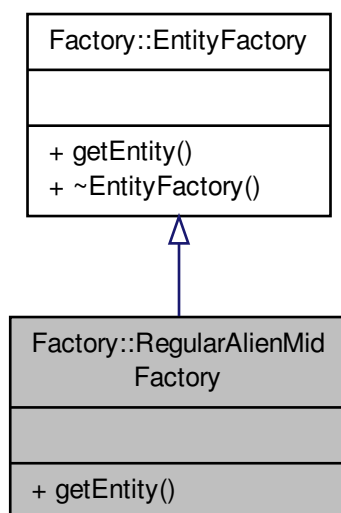
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlienTypes.h`

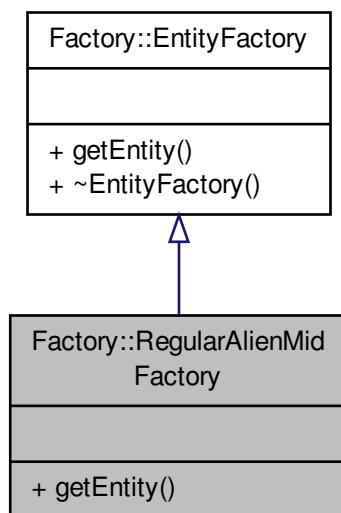
6.19 Factory::RegularAlienMidFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::RegularAlienMidFactory:



Collaboration diagram for Factory::RegularAlienMidFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an RegularAlienMid.

6.19.1 Detailed Description

Concrete [Factory](#) generating RegularAlienMids

6.19.2 Member Function Documentation

6.19.2.1 `Model::Entity * Factory::RegularAlienMidFactory::getEntity (int xLoc, int yLoc, int relativeSize)` `[virtual]`

Generates an RegularAlienMid.

Parameters

<i>xLoc</i>	Horizontal location of the RegularAlienMid
<i>yLoc</i>	Vertical location of the RegularAlienMid
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated RegularAlienMid

Implements [Factory::EntityFactory](#).

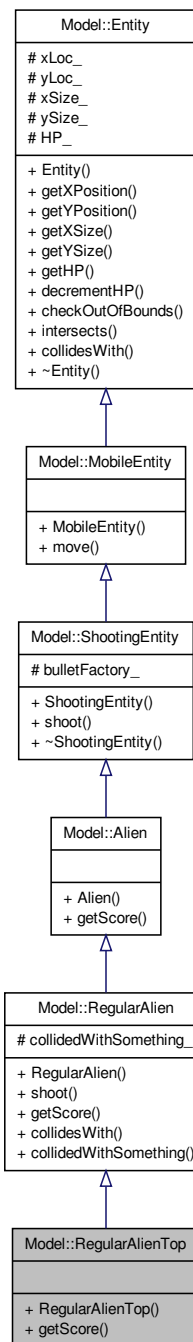
The documentation for this class was generated from the following files:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h`
- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp`

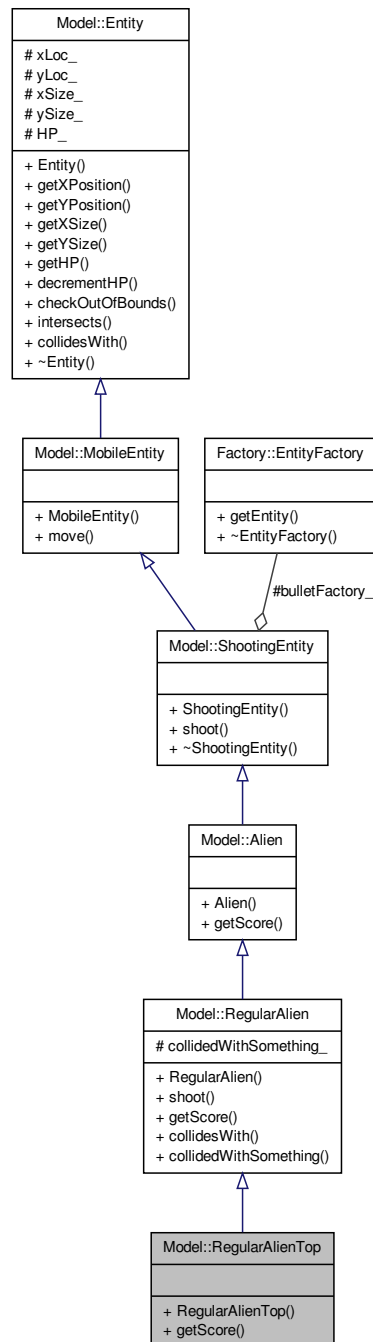
6.20 Model::RegularAlienTop Class Reference

```
#include <regularAlienTypes.h>
```

Inheritance diagram for Model::RegularAlienTop:



Collaboration diagram for Model::RegularAlienTop:



Public Member Functions

- [RegularAlienTop](#) (int xLoc, int yLoc, int relativeSize)

Constructor.

- int [getScore](#) ()

Gets the score value of the [RegularAlienTop](#).

Additional Inherited Members

6.20.1 Detailed Description

Class representing aliens in the top row in classic Space Invaders

6.20.2 Constructor & Destructor Documentation

6.20.2.1 `Model::RegularAlienTop::RegularAlienTop (int xLoc, int yLoc, int relativeSize)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of RegularAlienTop
<i>yLoc</i>	Vertical location of top of RegularAlienTop
<i>relativeSize</i>	size of the longest side

6.20.3 Member Function Documentation

6.20.3.1 `int Model::RegularAlienTop::getScore ()` `[inline]`, `[virtual]`

Gets the score value of the [RegularAlienTop](#).

Returns

40

Implements [Model::RegularAlien](#).

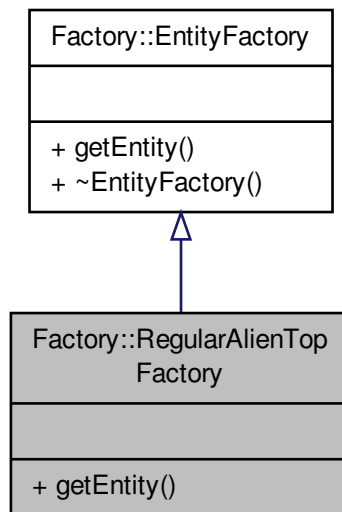
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlienTypes.h`

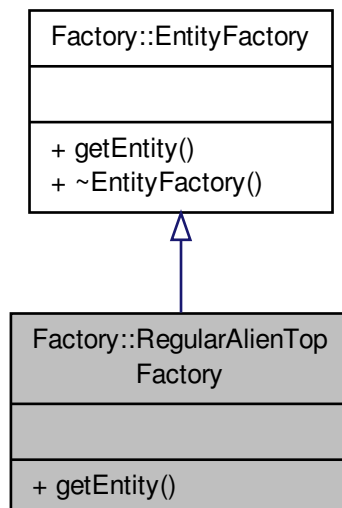
6.21 Factory::RegularAlienTopFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::RegularAlienTopFactory:



Collaboration diagram for Factory::RegularAlienTopFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an RegularAlienTop.

6.21.1 Detailed Description

Concrete [Factory](#) generating RegularAlienTops

6.21.2 Member Function Documentation

6.21.2.1 Model::Entity * Factory::RegularAlienTopFactory::getEntity (int *xLoc*, int *yLoc*, int *relativeSize*) [virtual]

Generates an RegularAlienTop.

Parameters

<i>xLoc</i>	Horizontal location of the RegularAlienTop
<i>yLoc</i>	Vertical location of the RegularAlienTop
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated RegularAlienTop

Implements [Factory::EntityFactory](#).

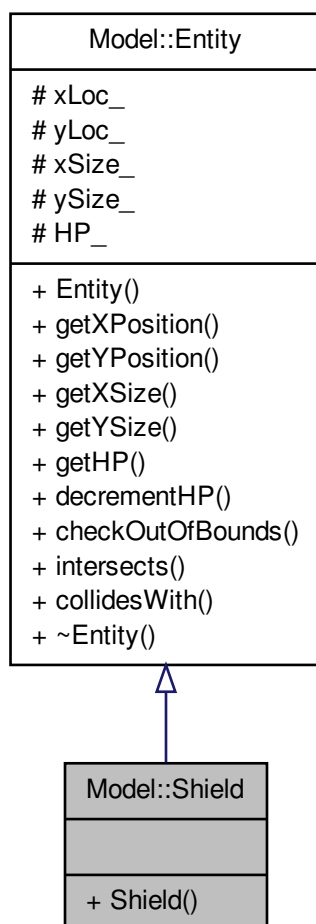
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp](#)

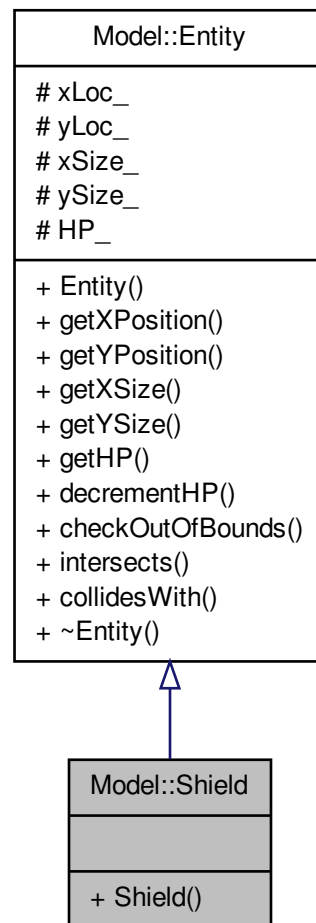
6.22 Model::Shield Class Reference

```
#include <shield.h>
```

Inheritance diagram for Model::Shield:



Collaboration diagram for Model::Shield:



Public Member Functions

- [Shield](#) (int xLoc, int yLoc, int relativeSize)
Constructor.

Additional Inherited Members

6.22.1 Detailed Description

Class representing a shield

6.22.2 Constructor & Destructor Documentation

6.22.2.1 Model::Shield::Shield (int xLoc, int yLoc, int relativeSize) [inline]

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of Shield
<i>yLoc</i>	Vertical location of top of Shield
<i>relativeSize</i>	size of the longest side
<i>maxHP</i>	Starting HP

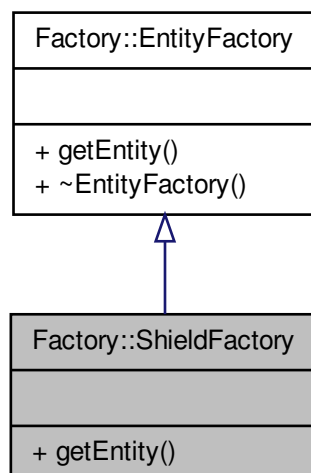
The documentation for this class was generated from the following file:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/shield.h](#)

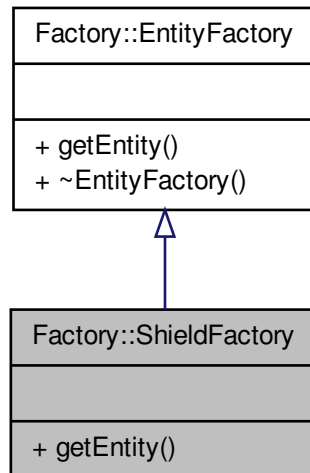
6.23 Factory::ShieldFactory Class Reference

```
#include <concreteEntityFactories.h>
```

Inheritance diagram for Factory::ShieldFactory:



Collaboration diagram for Factory::ShieldFactory:



Public Member Functions

- virtual [Model::Entity](#) * [getEntity](#) (int xLoc, int yLoc, int relativeSize)
Generates an Shield.

6.23.1 Detailed Description

Concrete [Factory](#) generating Shields

6.23.2 Member Function Documentation

6.23.2.1 [Model::Entity](#) * [Factory::ShieldFactory::getEntity](#) (int xLoc, int yLoc, int relativeSize) [virtual]

Generates an Shield.

Parameters

<i>xLoc</i>	Horizontal location of the Shield
<i>yLoc</i>	Vertical location of the Shield
<i>relativeSize</i>	Size of the longest side

Returns

Pointer to the generated Shield

Implements [Factory::EntityFactory](#).

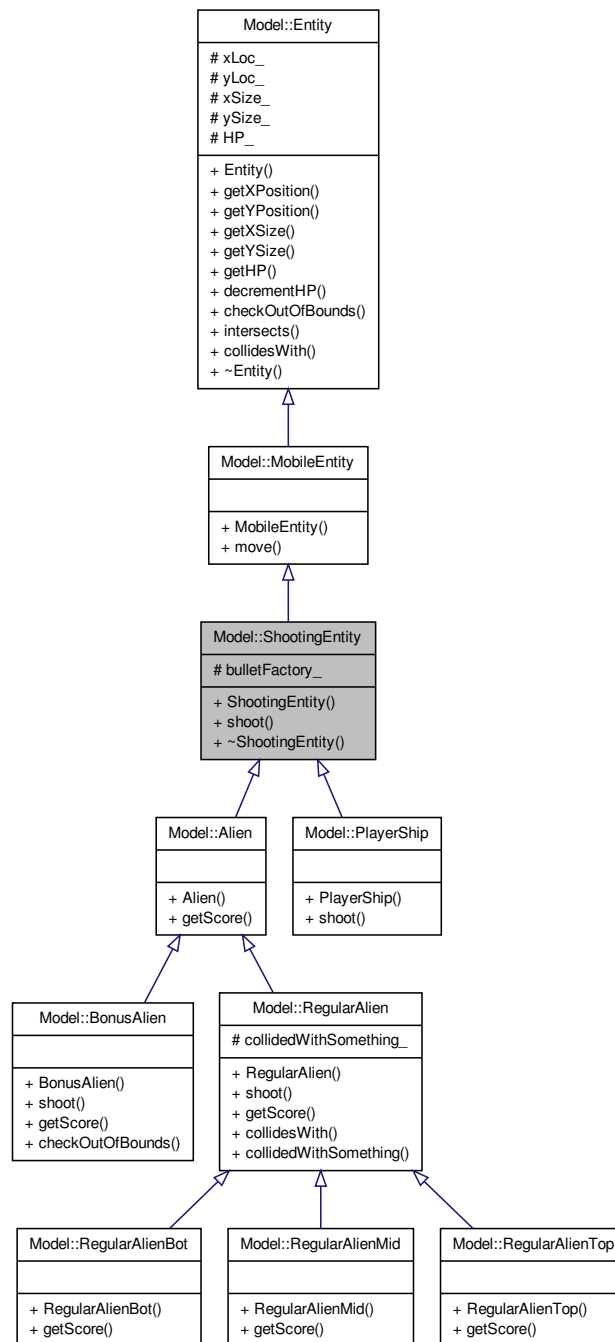
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp](#)

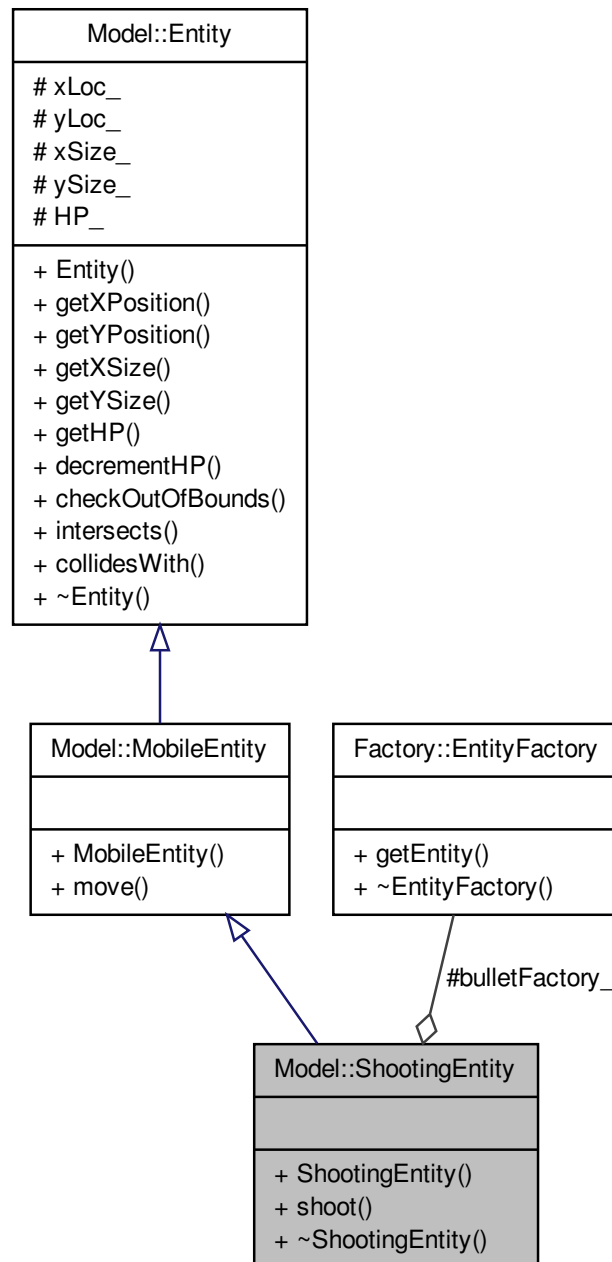
6.24 Model::ShootingEntity Class Reference

```
#include <shootingEntity.h>
```

Inheritance diagram for Model::ShootingEntity:



Collaboration diagram for Model::ShootingEntity:



Public Member Functions

- [ShootingEntity](#) (int xLoc, int yLoc, int xSize, int ySize, int HP, [Factory::EntityFactory](#) *fact)
Constructor.
- virtual [Bullet](#) * [shoot](#) ()=0
[ShootingEntity](#) fires one [Bullet](#).
- virtual [~ShootingEntity](#) ()

Destructor.

Protected Attributes

- [Factory::EntityFactory](#) * [bulletFactory_](#)

6.24.1 Detailed Description

Class representing shooting (==able to create Bullets) entities in the game

6.24.2 Constructor & Destructor Documentation

6.24.2.1 `Model::ShootingEntity::ShootingEntity (int xLoc, int yLoc, int xSize, int ySize, int HP, Factory::EntityFactory * fact)` `[inline]`

Constructor.

Parameters

<i>xLoc</i>	Horizontal location of left side of Entity
<i>yLoc</i>	Vertical location of top of Entity
<i>xSize</i>	Horizontal size of Entity
<i>ySize</i>	Vertical size of Entity
<i>HP</i>	Starting HP
<i>fact</i>	Factory used to create Bullets when shooting

6.24.2.2 `virtual Model::ShootingEntity::~~ShootingEntity ()` `[inline]`, `[virtual]`

Destructor.

6.24.3 Member Function Documentation

6.24.3.1 `virtual Bullet* Model::ShootingEntity::shoot ()` `[pure virtual]`

[ShootingEntity](#) fires one [Bullet](#).

Returns

pointer to the fired [Bullet](#)

Implemented in [Model::PlayerShip](#), [Model::BonusAlien](#), and [Model::RegularAlien](#).

6.24.4 Member Data Documentation

6.24.4.1 `Factory::EntityFactory* Model::ShootingEntity::bulletFactory_` `[protected]`

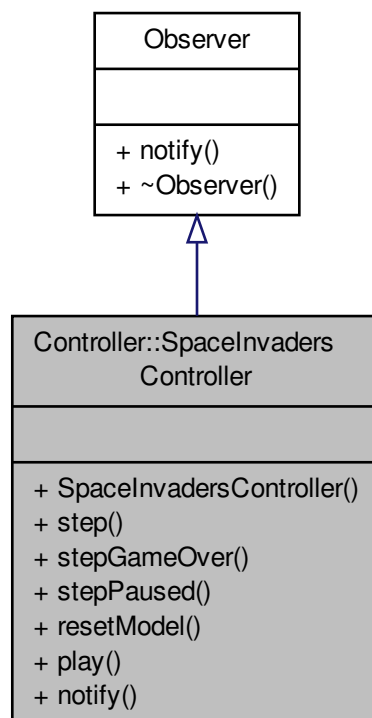
The documentation for this class was generated from the following file:

- `/home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/shootingEntity.h`

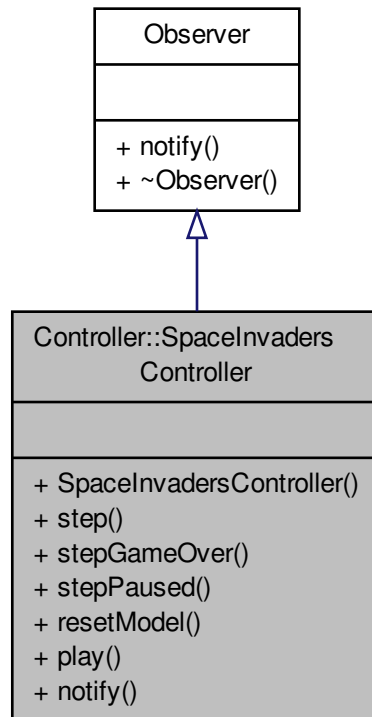
6.25 Controller::SpaceInvadersController Class Reference

```
#include <spaceInvadersController.h>
```

Inheritance diagram for Controller::SpaceInvadersController:



Collaboration diagram for Controller::SpaceInvadersController:



Public Member Functions

- [SpaceInvadersController](#) ()
Constructor.
- bool [step](#) ()
Performs one step of the game.
- bool [stepGameOver](#) ()
Performs one step of the game while game is over.
- bool [stepPaused](#) ()
Performs one step of the game while game is paused.
- void [resetModel](#) ()
Generates a new [Model](#) when game is about to restart.
- void [play](#) ()
Play the game.
- void [notify](#) (bool gameOver, bool winner)
Handles notifications sent by observee (the [Model](#))

6.25.1 Detailed Description

Class responsible for controlling both the [Model](#) and [View](#) of a Space Invaders game

6.25.2 Constructor & Destructor Documentation

6.25.2.1 Controller::SpaceInvadersController::SpaceInvadersController ()

Constructor.

6.25.3 Member Function Documentation

6.25.3.1 void Controller::SpaceInvadersController::notify (bool *gameOver*, bool *winner*) [virtual]

Handles notifications sent by observee (the [Model](#))

Parameters

<i>gameOver</i>	True if game is currently over
<i>winner</i>	True if player just won a level

Implements [Observer](#).

6.25.3.2 void Controller::SpaceInvadersController::play ()

Play the game.

6.25.3.3 void Controller::SpaceInvadersController::resetModel ()

Generates a new [Model](#) when game is about to restart.

6.25.3.4 bool Controller::SpaceInvadersController::step ()

Performs one step of the game.

Returns

True if game is still running

6.25.3.5 bool Controller::SpaceInvadersController::stepGameOver ()

Performs one step of the game while game is over.

Returns

True if game is still running

6.25.3.6 bool Controller::SpaceInvadersController::stepPaused ()

Performs one step of the game while game is paused.

Returns

True if game is still running

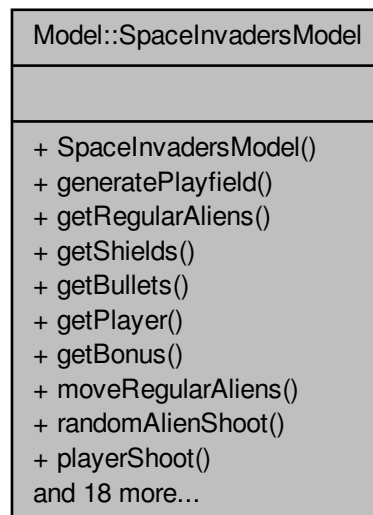
The documentation for this class was generated from the following files:

- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Controller/spaceInvadersController.h](#)
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/Controller/spaceInvadersController.cpp](#)

6.26 Model::SpaceInvadersModel Class Reference

```
#include <spaceInvadersModel.h>
```

Collaboration diagram for Model::SpaceInvadersModel:



Public Member Functions

- [SpaceInvadersModel](#) (int xSize, int ySize)
Constructor.
- void [generatePlayfield](#) ()
Generates a fresh playfield for when a next level is started.
- std::vector< [RegularAlien](#) * > & [getRegularAliens](#) ()
gets vector containing pointers to all RegularAliens
- std::vector< [Shield](#) * > & [getShields](#) ()
gets vector containing pointers to all Shields
- std::vector< [Bullet](#) * > & [getBullets](#) ()
gets vector containing pointers to all Bullets
- [PlayerShip](#) * [getPlayer](#) ()
gets pointer to PlayerShip
- [BonusAlien](#) * [getBonus](#) ()
gets pointer to BonusAlien
- void [moveRegularAliens](#) ([Direction](#) dir, int distance)
Moves all of the RegularAliens.
- void [randomAlienShoot](#) ()
Makes one random RegularAlien (bottom RegularAlien in his column) shoot one AlienBullet.
- void [playerShoot](#) ()
makes the PlayerShip shoot one PlayerBullet
- void [handleAllCollisions](#) ()
Checks if any Entities collided with each other.

- void [levelOver](#) (bool won)
Tears down the current playfield and sets up a new one if necessary.
- int [getScore](#) () const
Gets the current score.
- int [getLives](#) () const
Gets the current remaining extra lives.
- int [getLevel](#) () const
Gets the current level.
- int [getFramesPerAlienMove](#) () const
Gets how many frames it takes for Aliens to move once.
- [RegularAlien](#) * [getLeftmostRegularAlien](#) () const
Determines (one of) the RegularAlien(s) that is in the leftmost column.
- [RegularAlien](#) * [getRightmostRegularAlien](#) () const
Determines (one of) the RegularAlien(s) that is in the rightmost column.
- void [setPlayerMovements](#) (bool moveLeft, bool moveRight, bool shoot)
Sets what the [PlayerShip](#) should to in the next step.
- bool [areAliensAlive](#) () const
Checks whether there are any living RegularAliens left.
- void [stepPlayer](#) ()
Handles all PlayerShip-specific actions in one step of the game.
- void [stepRegularAliens](#) ()
Handles all RegularAlien-specific actions in one step of the game.
- void [stepBonusAlien](#) ()
Handles of all [BonusAlien](#) actions (generating, destroying, moving)
- void [step](#) ()
Performs one step of the ongoing game.
- std::vector< [Observer](#) * >::iterator [findObserver](#) ([Observer](#) *obs)
Searches for given [Observer](#) in vector of Observers.
- bool [registerObserver](#) ([Observer](#) *obs)
Add an observer to the vector of observers.
- bool [unregisterObserver](#) ([Observer](#) *obs)
- void [notifyObservers](#) ()
Notifies all observers if something changed.

6.26.1 Detailed Description

Class representing the full SpaceInvaders model

6.26.2 Constructor & Destructor Documentation

6.26.2.1 Model::SpaceInvadersModel::SpaceInvadersModel (int xSize, int ySize)

Constructor.

Parameters

<i>xSize</i>	Horizontal size of the playfield
<i>ySize</i>	Vertical size of the playfield

6.26.3 Member Function Documentation

6.26.3.1 `bool Model::SpaceInvadersModel::areAliensAlive () const`

Checks whether there are any living RegularAliens left.

Returns

True if any [RegularAlien](#) alive

6.26.3.2 `std::vector< Observer * >::iterator Model::SpaceInvadersModel::findObserver (Observer * obs)`

Searches for given [Observer](#) in vector of Observers.

Parameters

<i>obs</i>	Observer* to look for
------------	-----------------------

Returns

iterator to given Observer* if found, iterator to end of [Observer](#) vector if not found

6.26.3.3 `void Model::SpaceInvadersModel::generatePlayfield ()`

Generates a fresh playfield for when a next level is started.

6.26.3.4 `BonusAlien * Model::SpaceInvadersModel::getBonus ()`

gets pointer to [BonusAlien](#)

Returns

pointer to [BonusAlien](#)

6.26.3.5 `std::vector< Bullet * > & Model::SpaceInvadersModel::getBullets ()`

gets vector containing pointers to all Bullets

Returns

vector of all [Bullet](#) pointers

6.26.3.6 `int Model::SpaceInvadersModel::getFramesPerAlienMove () const`

Gets how many frames it takes for Aliens to move once.

Frames it takes for aliens to move once

6.26.3.7 RegularAlien * Model::SpaceInvadersModel::getLeftmostRegularAlien () const

Determines (one of) the RegularAlien(s) that is in the leftmost column.

Returns

Pointer to leftmost [RegularAlien](#)

6.26.3.8 int Model::SpaceInvadersModel::getLevel () const

Gets the current level.

Returns

The current level

6.26.3.9 int Model::SpaceInvadersModel::getLives () const

Gets the current remaining extra lives.

Returns

The current remaining extra lives

6.26.3.10 PlayerShip * Model::SpaceInvadersModel::getPlayer ()

gets pointer to [PlayerShip](#)

Returns

pointer to [PlayerShip](#)

6.26.3.11 std::vector< RegularAlien * > & Model::SpaceInvadersModel::getRegularAliens ()

gets vector containing pointers to all RegularAliens

Returns

vector of all [RegularAlien](#) pointers

6.26.3.12 RegularAlien * Model::SpaceInvadersModel::getRightmostRegularAlien () const

Determines (one of) the RegularAlien(s) that is in the rightmost column.

Returns

Pointer to rightmost [RegularAlien](#)

6.26.3.13 int Model::SpaceInvadersModel::getScore () const

Gets the current score.

Returns

The current score

6.26.3.14 `std::vector< Shield * > & Model::SpaceInvadersModel::getShields ()`

gets vector containing pointers to all Shields

Returns

vector of all [Shield](#) pointers

6.26.3.15 `void Model::SpaceInvadersModel::handleAllCollisions ()`

Checks if any Entities collided with each other.

6.26.3.16 `void Model::SpaceInvadersModel::levelOver (bool won)`

Tears down the current playfield and sets up a new one if necessary.

Parameters

<i>won</i>	Incicates whether the player won this level. If true, a new playfield will be generated
------------	---

6.26.3.17 `void Model::SpaceInvadersModel::moveRegularAliens (Direction dir, int distance)`

Moves all of the RegularAliens.

Parameters

<i>dir</i>	Current direction of the RegularAliens
<i>distance</i>	The distance they should be moved over

6.26.3.18 `void Model::SpaceInvadersModel::notifyObservers ()`

Notifies all observers if something changed.

6.26.3.19 `void Model::SpaceInvadersModel::playerShoot ()`

makes the [PlayerShip](#) shoot one [PlayerBullet](#)

6.26.3.20 `void Model::SpaceInvadersModel::randomAlienShoot ()`

Makes one random [RegularAlien](#) (bottom [RegularAlien](#) in his column) shoot one [AlienBullet](#).

6.26.3.21 `bool Model::SpaceInvadersModel::registerObserver (Observer * obs)`

Add an observer to the vector of observers.

Returns

True if added, otherwise false

6.26.3.22 `void Model::SpaceInvadersModel::setPlayerMovements (bool moveLeft, bool moveRight, bool shoot)`

Sets what the [PlayerShip](#) should to in the next step.

Parameters

<i>moveLeft</i>	True if PlayerShip should move left
<i>moveRight</i>	True if PlayerShip should move right
<i>shoot</i>	True if player should fire one PlayerBullet

6.26.3.23 void Model::SpaceInvadersModel::step ()

Performs one step of the ongoing game.

6.26.3.24 void Model::SpaceInvadersModel::stepBonusAlien ()

Handles of all [BonusAlien](#) actions (generating, destroying, moving)

6.26.3.25 void Model::SpaceInvadersModel::stepPlayer ()

Handles all PlayerShip-specific actions in one step of the game.

6.26.3.26 void Model::SpaceInvadersModel::stepRegularAliens ()

Handles all RegularAlien-specific actions in one step of the game.

6.26.3.27 bool Model::SpaceInvadersModel::unregisterObserver (Observer * obs)

an observer from the vector of observers

Returns

True if added, otherwise false

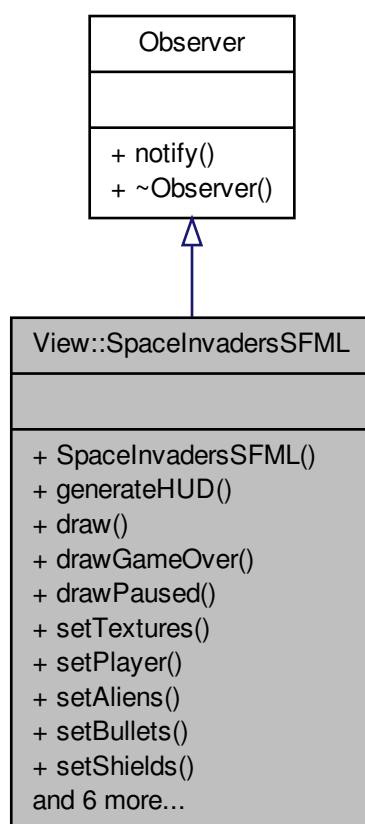
The documentation for this class was generated from the following files:

- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/[spaceInvadersModel.h](#)
- /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/[spaceInvadersModel.cpp](#)

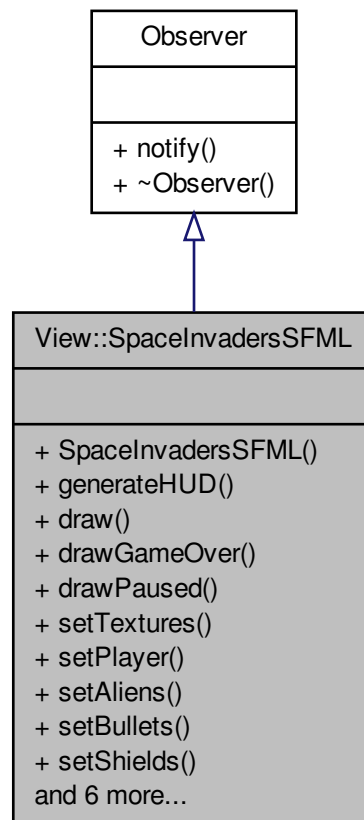
6.27 View::SpaceInvadersSFML Class Reference

```
#include <spaceInvadersView.h>
```

Inheritance diagram for View::SpaceInvadersSFML:



Collaboration diagram for View::SpaceInvadersSFML:



Public Member Functions

- [SpaceInvadersSFML](#) (int xReso, int yReso, [Model::SpaceInvadersModel](#) *SI)
Constructor.
- void [generateHUD](#) ()
Generates the HUD for the current state of the game.
- bool [draw](#) ()
Draws the current state of the game.
- bool [drawGameOver](#) ()
Draws the current state of a game currently not in progress.
- bool [drawPaused](#) ()
Draws the current state of a game currently paused (but in progress)
- void [setTextures](#) ()
Loads all textures.
- void [setPlayer](#) ()
Sets the Sprite for the PlayerShip.
- void [setAliens](#) ()
Sets the Sprites for the RegularAliens.
- void [setBullets](#) ()

- Sets the Sprites for the Bullets.*
- void [setShields](#) ()
- Sets the Sprites for the Shields.*
- void [setBonus](#) ()
- Sets the Sprite for the BonusAlien.*
- void [generateShape](#) ([Model::Entity](#) *entity)
- : Generates a square shape of the given Entity Should only be used in case of missing Texture*
- void [setModel](#) ([Model::SpaceInvadersModel](#) *SI)
- Sets new [Model](#) pointer.*
- void [closeWindow](#) ()
- Closes the window.*
- bool [isViewOpen](#) () const
- Checks if the [View](#) (in this [View](#): the SFML window) is currently open.*
- virtual void [notify](#) (bool gameOver, bool winner)
- Handles notifications sent by observee (the [Model](#))*

6.27.1 Detailed Description

Class taking care of the visual representation of a Space Invaders game through the SFML library

6.27.2 Constructor & Destructor Documentation

6.27.2.1 [View::SpaceInvadersSFML::SpaceInvadersSFML](#) (int *xReso*, int *yReso*, [Model::SpaceInvadersModel](#) * *SI*)

Constructor.

Parameters

<i>xReso</i>	Horizontal resolution of the playfield
<i>yReso</i>	Vertical resolution of the playfield
<i>SI</i>	Pointer to the Model of the game being represented

6.27.3 Member Function Documentation

6.27.3.1 void [View::SpaceInvadersSFML::closeWindow](#) ()

Closes the window.

6.27.3.2 bool [View::SpaceInvadersSFML::draw](#) ()

Draws the current state of the game.

Returns

True if window still open

6.27.3.3 bool [View::SpaceInvadersSFML::drawGameOver](#) ()

Draws the current state of a game currently not in progress.

Returns

True if window still open

6.27.3.4 `bool View::SpaceInvadersSFML::drawPaused ()`

Draws the current state of a game currently paused (but in progress)

Returns

True if window still open

6.27.3.5 `void View::SpaceInvadersSFML::generateHUD ()`

Generates the HUD for the current state of the game.

6.27.3.6 `void View::SpaceInvadersSFML::generateShape (Model::Entity * entity)`

: Generates a square shape of the given Entity Should only be used in case of missing Texture

Parameters

<i>entity</i>	Pointer to the Entity of which shape should be generated
---------------	--

6.27.3.7 `bool View::SpaceInvadersSFML::isViewOpen () const`

Checks if the [View](#) (in this [View](#): the SFML window) is currently open.

Returns

True if window is open

6.27.3.8 `void View::SpaceInvadersSFML::notify (bool gameOver, bool winner)` `[virtual]`

Handles notifications sent by observee (the [Model](#))

Parameters

<i>gameOver</i>	True if game is currently over
<i>winner</i>	True if player just won a level

Implements [Observer](#).

6.27.3.9 `void View::SpaceInvadersSFML::setAliens ()`

Sets the Sprites for the RegularAliens.

6.27.3.10 `void View::SpaceInvadersSFML::setBonus ()`

Sets the Sprite for the BonusAlien.

6.27.3.11 `void View::SpaceInvadersSFML::setBullets ()`

Sets the Sprites for the Bullets.

6.27.3.12 `void View::SpaceInvadersSFML::setModel (Model::SpaceInvadersModel * SI)`

Sets new [Model](#) pointer.

6.27.3.13 void View::SpacInvadersSFML::setPlayer ()

Sets the Sprite for the PlayerShip.

6.27.3.14 void View::SpacInvadersSFML::setShields ()

Sets the Sprites for the Shields.

6.27.3.15 void View::SpacInvadersSFML::setTextures ()

Loads all textures.

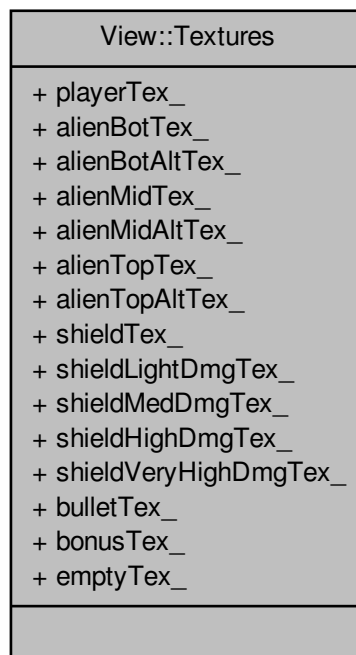
The documentation for this class was generated from the following files:

- /home/jakob/Dropbox/UA/SpacInvaders/src/View/[spacInvadersView.h](#)
- /home/jakob/Dropbox/UA/SpacInvaders/src/View/[spacInvadersView.cpp](#)

6.28 View::Textures Struct Reference

```
#include <textures.h>
```

Collaboration diagram for View::Textures:



Public Attributes

- sf::Texture [playerTex_](#)

- sf::Texture [alienBotTex_](#)
- sf::Texture [alienBotAltTex_](#)
- sf::Texture [alienMidTex_](#)
- sf::Texture [alienMidAltTex_](#)
- sf::Texture [alienTopTex_](#)
- sf::Texture [alienTopAltTex_](#)
- sf::Texture [shieldTex_](#)
- sf::Texture [shieldLightDmgTex_](#)
- sf::Texture [shieldMedDmgTex_](#)
- sf::Texture [shieldHighDmgTex_](#)
- sf::Texture [shieldVeryHighDmgTex_](#)
- sf::Texture [bulletTex_](#)
- sf::Texture [bonusTex_](#)
- sf::Texture [emptyTex_](#)

6.28.1 Detailed Description

Holds all of the [Textures](#) used by the SFML view

6.28.2 Member Data Documentation

6.28.2.1 sf::Texture View::Textures::alienBotAltTex_

6.28.2.2 sf::Texture View::Textures::alienBotTex_

6.28.2.3 sf::Texture View::Textures::alienMidAltTex_

6.28.2.4 sf::Texture View::Textures::alienMidTex_

6.28.2.5 sf::Texture View::Textures::alienTopAltTex_

6.28.2.6 sf::Texture View::Textures::alienTopTex_

6.28.2.7 sf::Texture View::Textures::bonusTex_

6.28.2.8 sf::Texture View::Textures::bulletTex_

6.28.2.9 sf::Texture View::Textures::emptyTex_

6.28.2.10 sf::Texture View::Textures::playerTex_

6.28.2.11 sf::Texture View::Textures::shieldHighDmgTex_

6.28.2.12 sf::Texture View::Textures::shieldLightDmgTex_

6.28.2.13 sf::Texture View::Textures::shieldMedDmgTex_

6.28.2.14 sf::Texture View::Textures::shieldTex_

6.28.2.15 sf::Texture View::Textures::shieldVeryHighDmgTex_

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

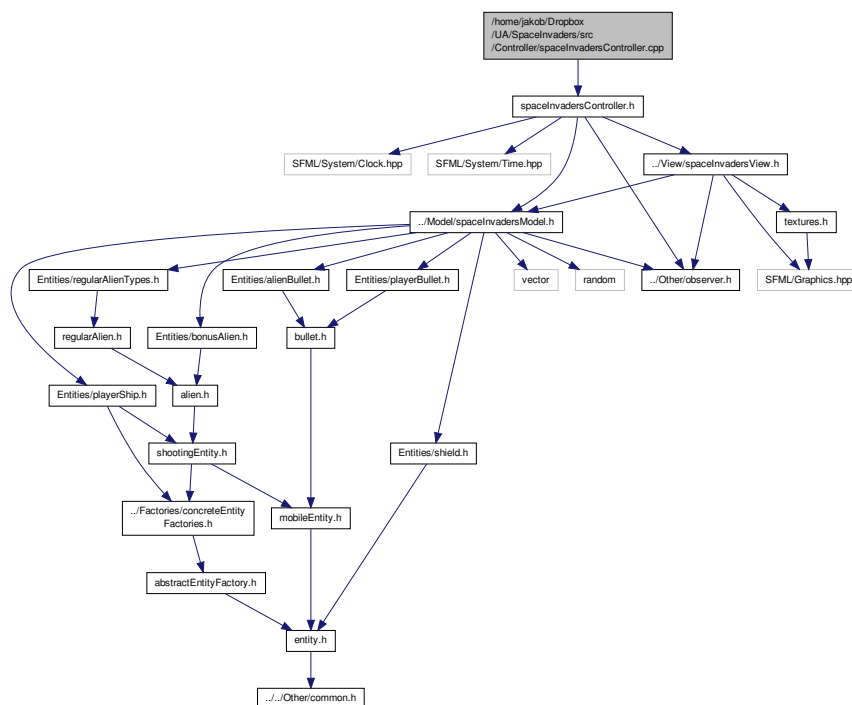
- [/home/jakob/Dropbox/UA/SpaceInvaders/src/View/textures.h](#)

Chapter 7

File Documentation

7.1 /home/jakob/Dropbox/UA/SpaceInvaders/src/Controller/spaceInvadersController.cpp File Reference

#include "spaceInvadersController.h"
Include dependency graph for spaceInvadersController.cpp:



Namespaces

- [Controller](#)

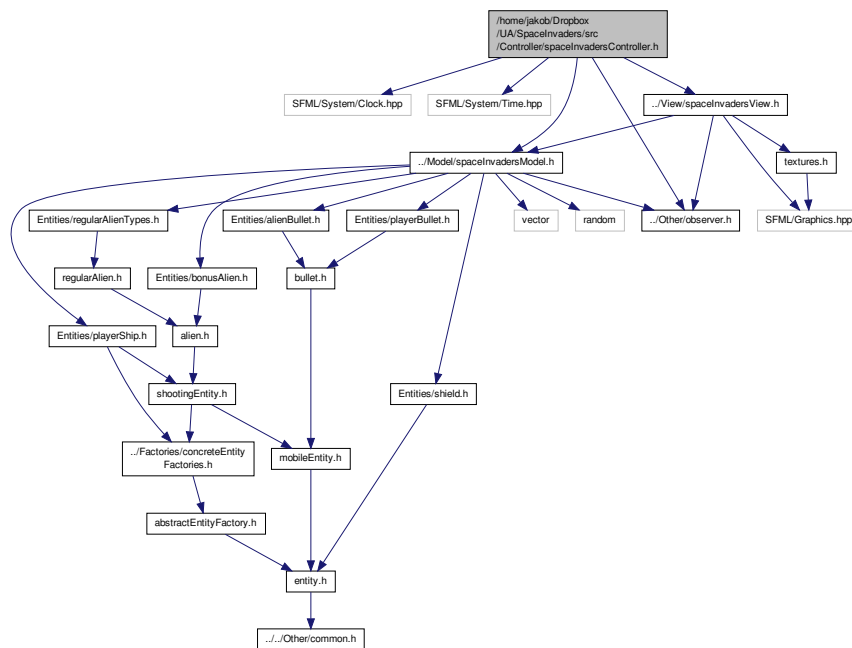
Constant Groups

- [Controller](#)

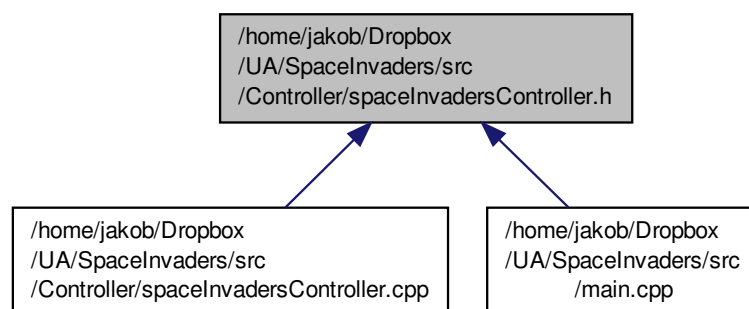
7.2 /home/jakob/Dropbox/UA/SpaceInvaders/src/Controller/spacInvadersController.h File Reference

```
#include <SFML/System/Clock.hpp>
#include <SFML/System/Time.hpp>
#include "../Model/spaceInvadersModel.h"
#include "../View/spaceInvadersView.h"
#include "../Other/observer.h"
```

Include dependency graph for spacInvadersController.h:



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



Classes

- class [Controller::SpaceInvadersController](#)

Namespaces

- [Controller](#)

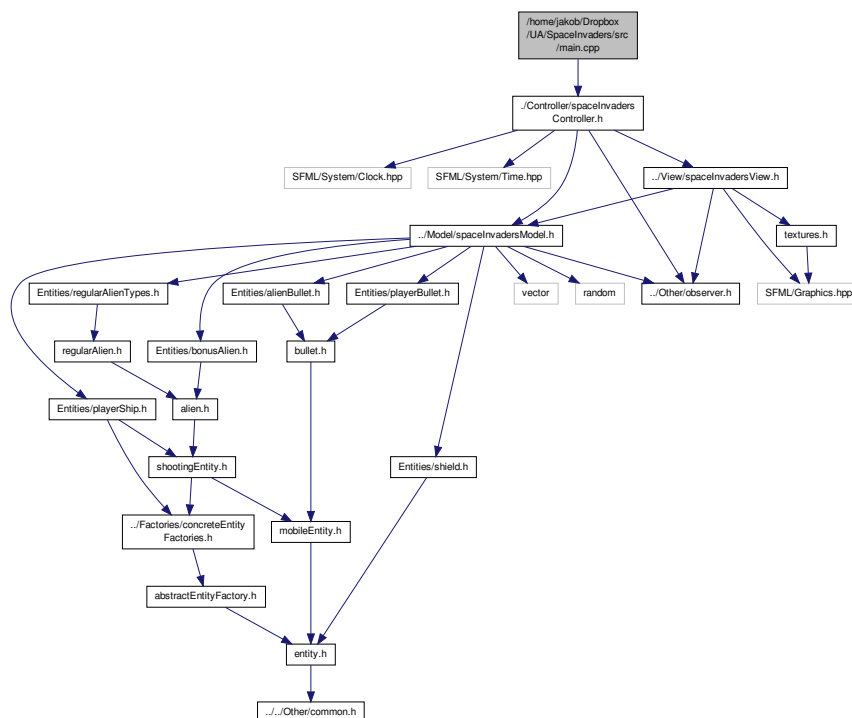
Constant Groups

- [Controller](#)

7.3 /home/jakob/Dropbox/UA/SpaceInvaders/src/main.cpp File Reference

```
#include "../Controller/spaceInvadersController.h"
```

Include dependency graph for main.cpp:



Functions

- `int main ()`

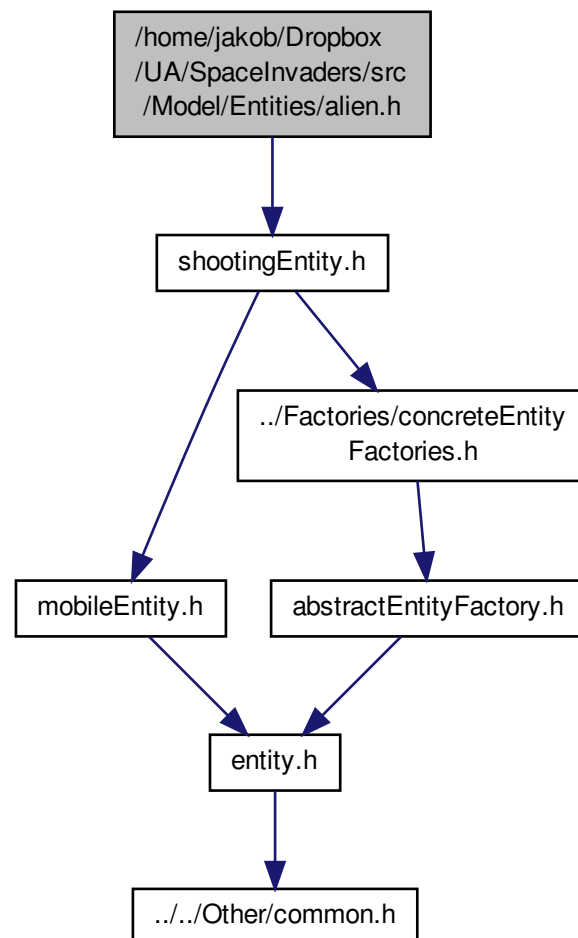
7.3.1 Function Documentation

7.3.1.1 `int main ()`

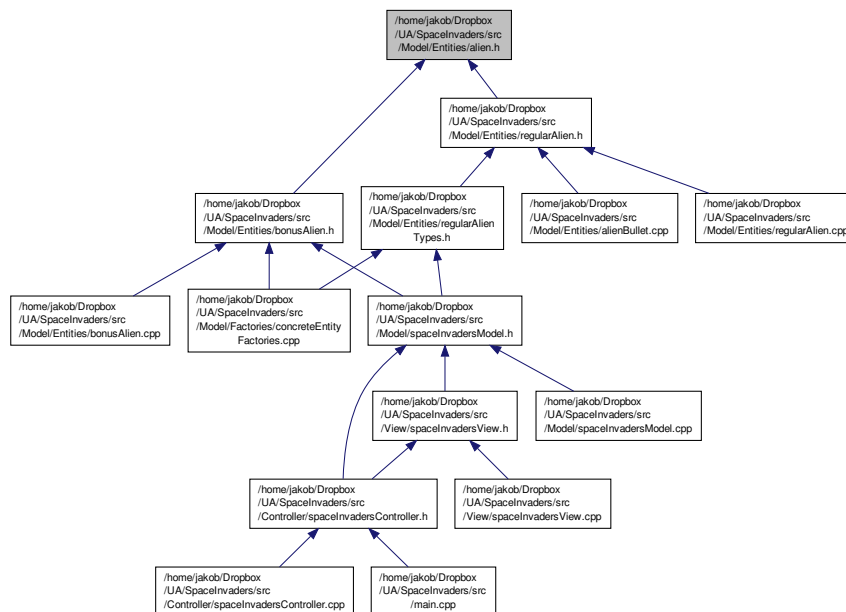
7.4 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alien.h File Reference

```
#include "shootingEntity.h"
```

Include dependency graph for alien.h:



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



Classes

- class [Model::Alien](#)

Namespaces

- [Model](#)

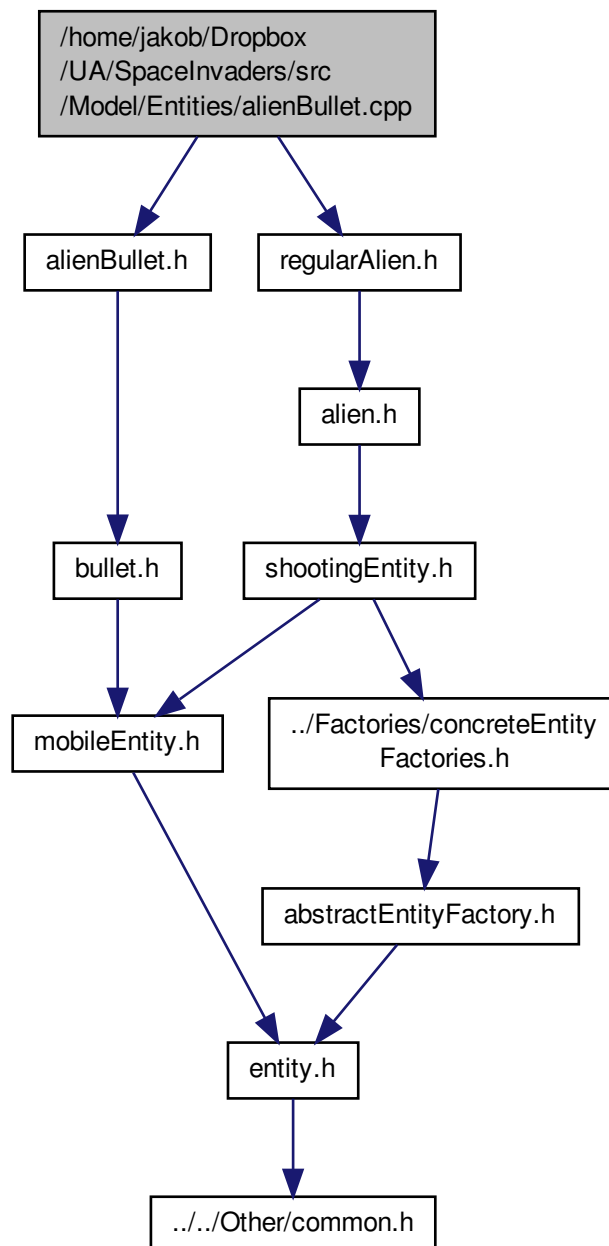
Constant Groups

- [Model](#)

7.5 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alienBullet.cpp File Reference

```
#include "alienBullet.h"
#include "regularAlien.h"
```

Include dependency graph for alienBullet.cpp:



Namespaces

- [Model](#)

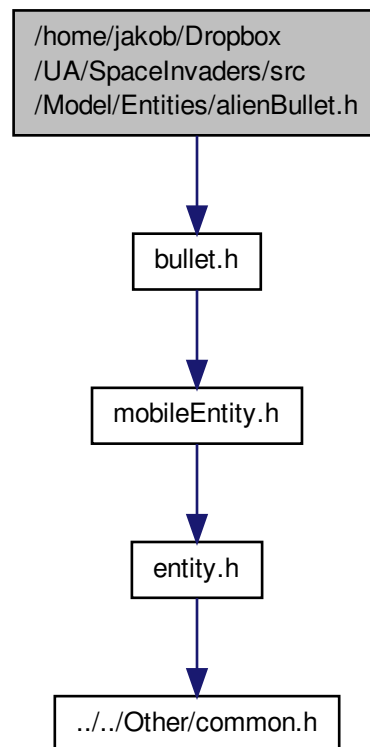
Constant Groups

- [Model](#)

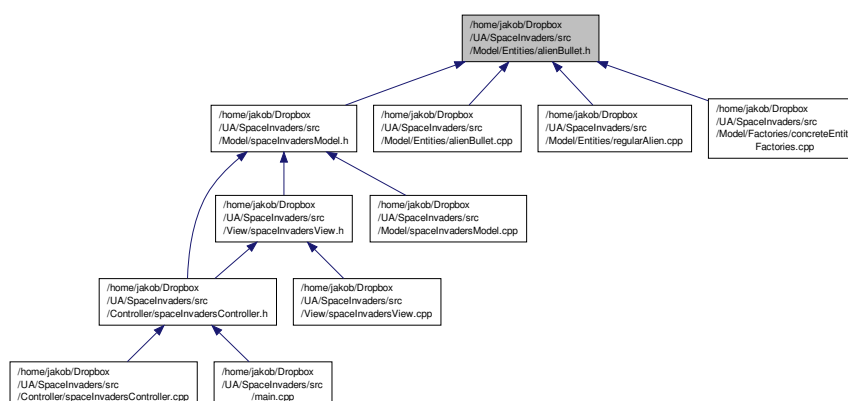
7.6 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/alienBullet.h File Reference

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

Include dependency graph for alienBullet.h:



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



Classes

- class [Model::AlienBullet](#)

Namespaces

- [Model](#)

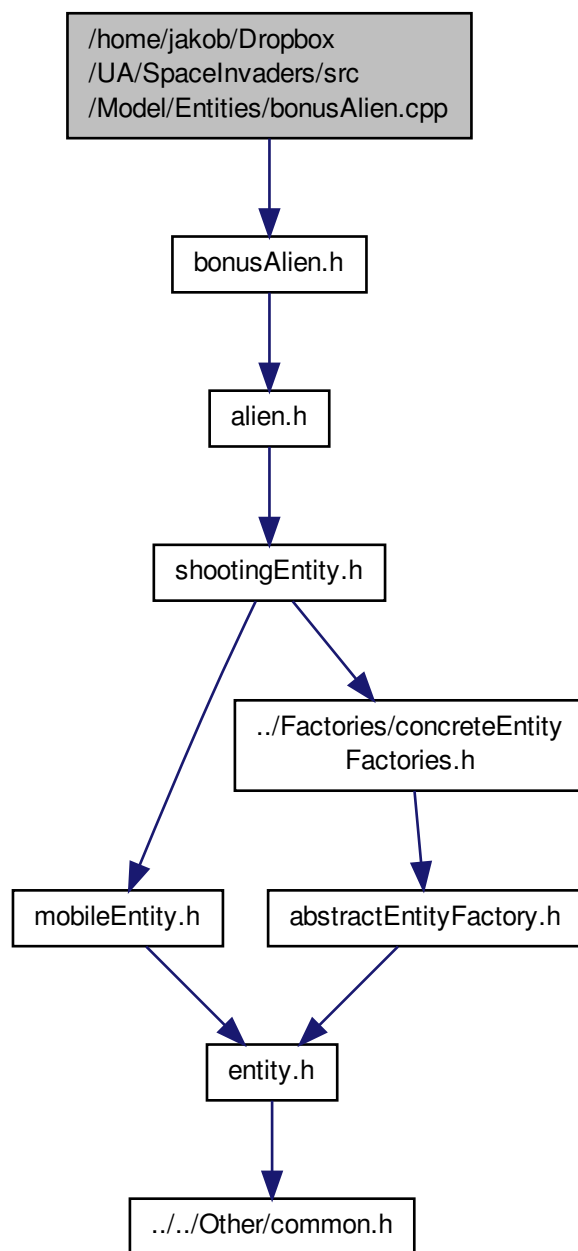
Constant Groups

- [Model](#)

7.7 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bonusAlien.cpp File Reference

```
#include "bonusAlien.h"
```


Include dependency graph for bonusAlien.cpp:



Namespaces

- [Model](#)

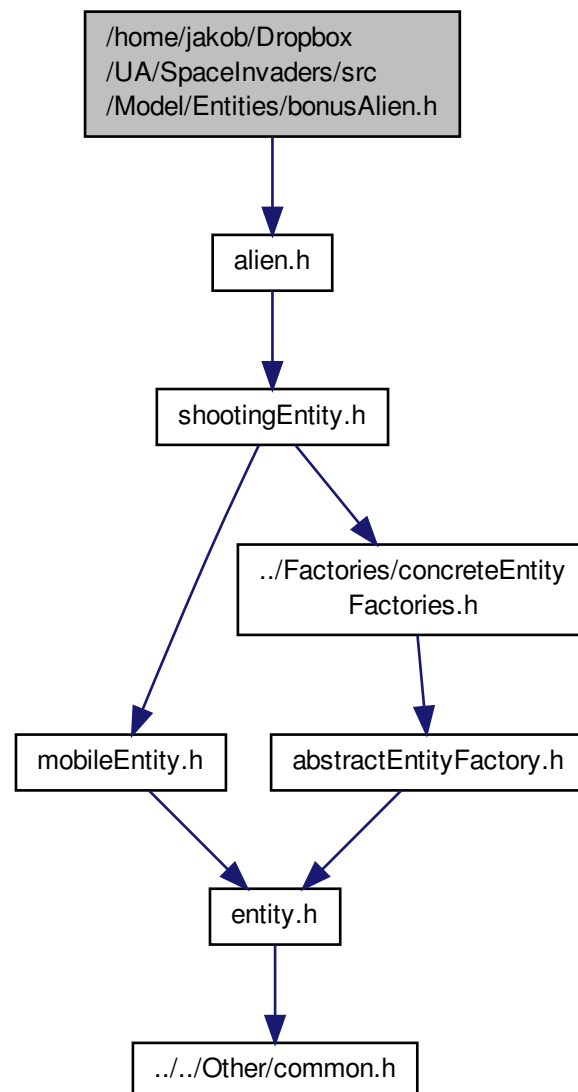
Constant Groups

- [Model](#)

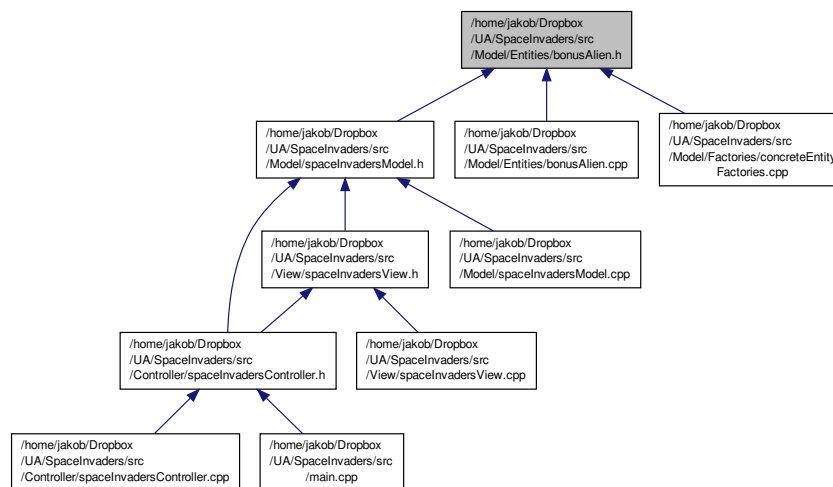
7.8 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bonusAlien.h File Reference

```
#include "alien.h"
```

Include dependency graph for bonusAlien.h:



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



Classes

- class [Model::BonusAlien](#)

Namespaces

- [Model](#)

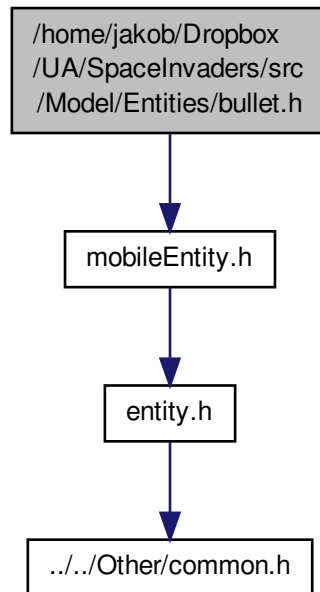
Constant Groups

- [Model](#)

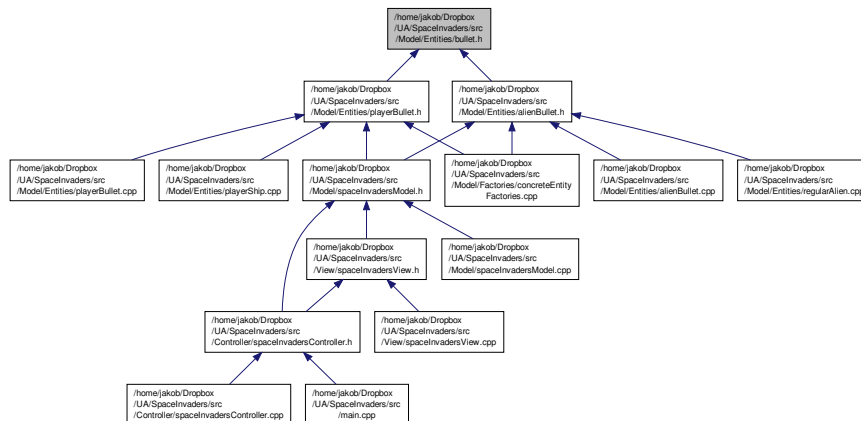
7.9 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/bullet.h File Reference

```
#include "mobileEntity.h"
```

Include dependency graph for `bullet.h`:



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



Classes

- class [Model::Bullet](#)

Namespaces

- [Model](#)

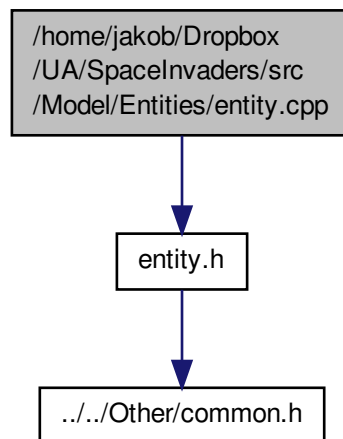
Constant Groups

- [Model](#)

7.10 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/entity.cpp File Reference

```
#include "entity.h"
```

Include dependency graph for entity.cpp:



Namespaces

- [Model](#)

Constant Groups

- [Model](#)

7.11 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/entity.h File Reference

```
#include "../../Other/common.h"
```

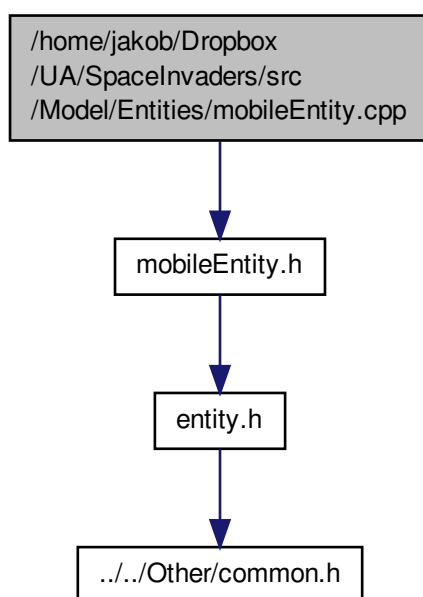

Constant Groups

- [Model](#)

7.12 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/mobileEntity.cpp File Reference

```
#include "mobileEntity.h"
```

Include dependency graph for mobileEntity.cpp:



Namespaces

- [Model](#)

Constant Groups

- [Model](#)

7.13 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/mobileEntity.h File Reference

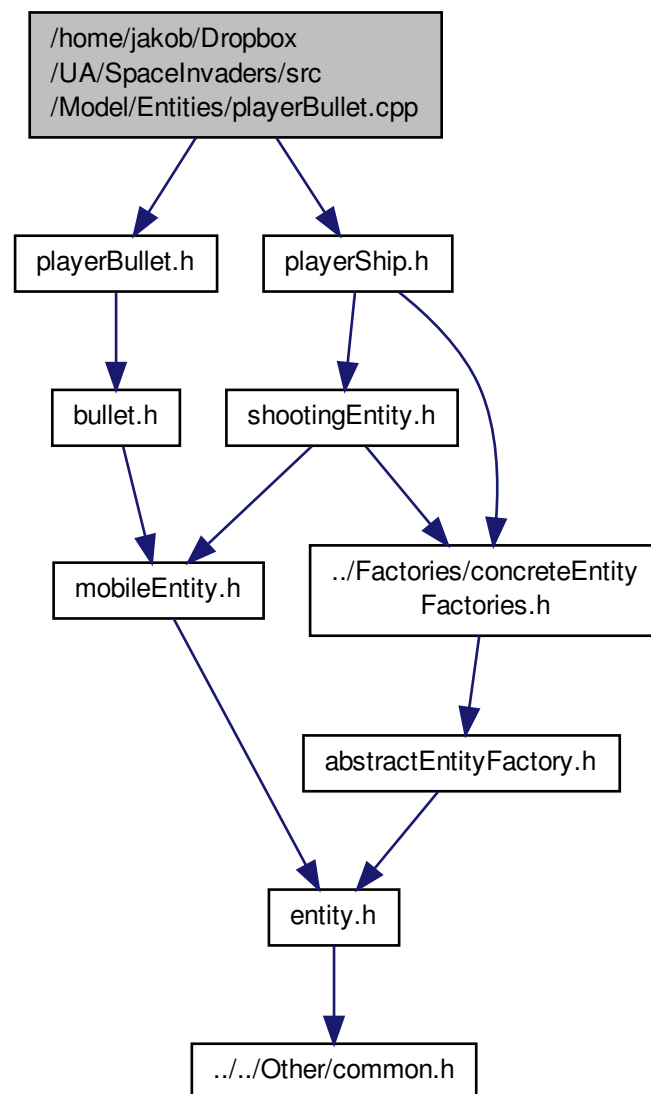
```
#include "entity.h"
```


Constant Groups

- [Model](#)

7.14 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerBullet.cpp File Reference

```
#include "playerBullet.h"  
#include "playerShip.h"  
Include dependency graph for playerBullet.cpp:
```



Namespaces

- [Model](#)

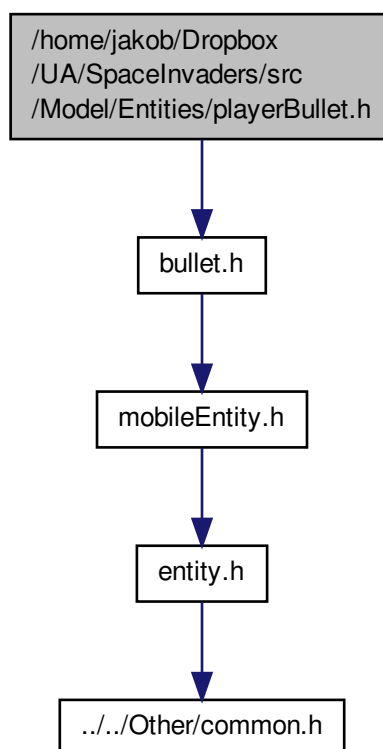
Constant Groups

- [Model](#)

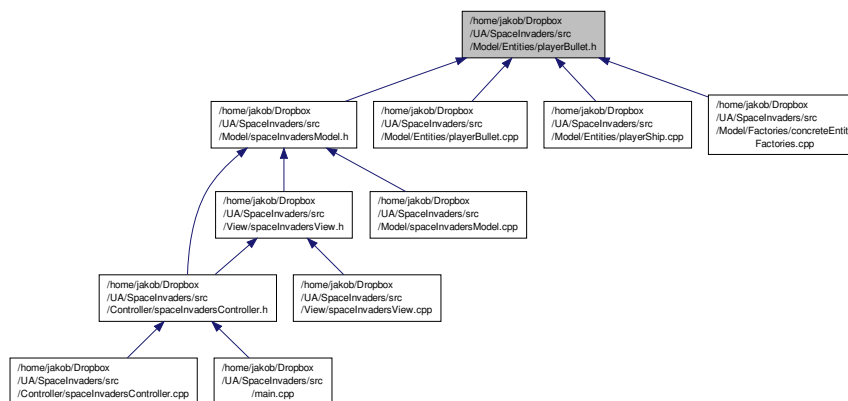
7.15 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerBullet.h File Reference

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

Include dependency graph for playerBullet.h:



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



Classes

- class [Model::PlayerBullet](#)

Namespaces

- [Model](#)

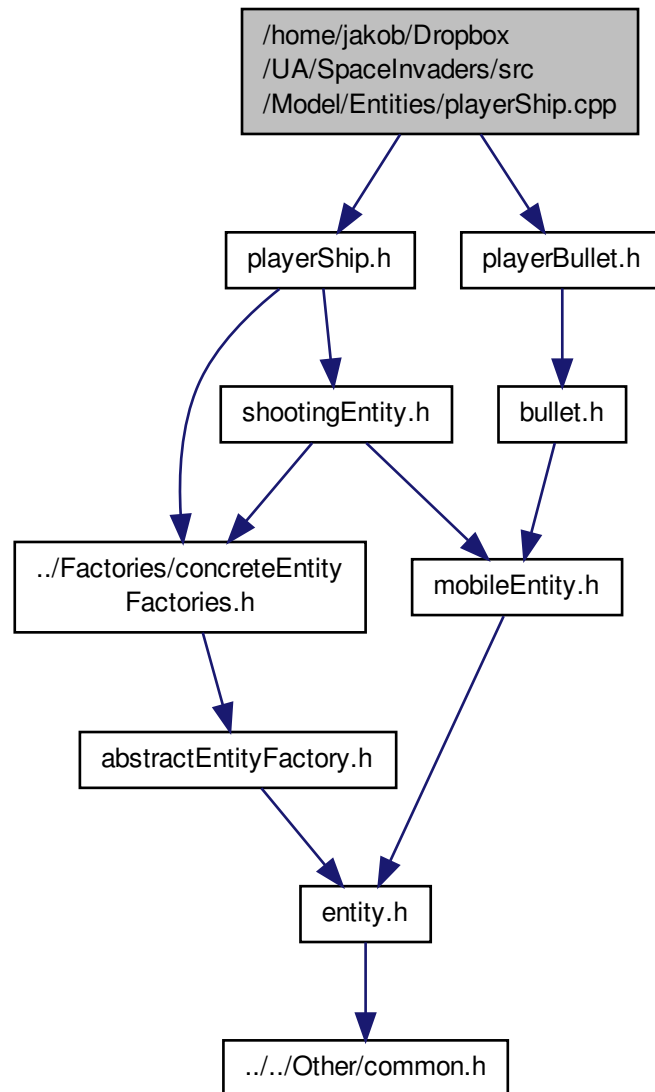
Constant Groups

- [Model](#)

7.16 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerShip.cpp File Reference

```
#include "playerShip.h"
#include "playerBullet.h"
```

Include dependency graph for playerShip.cpp:



Namespaces

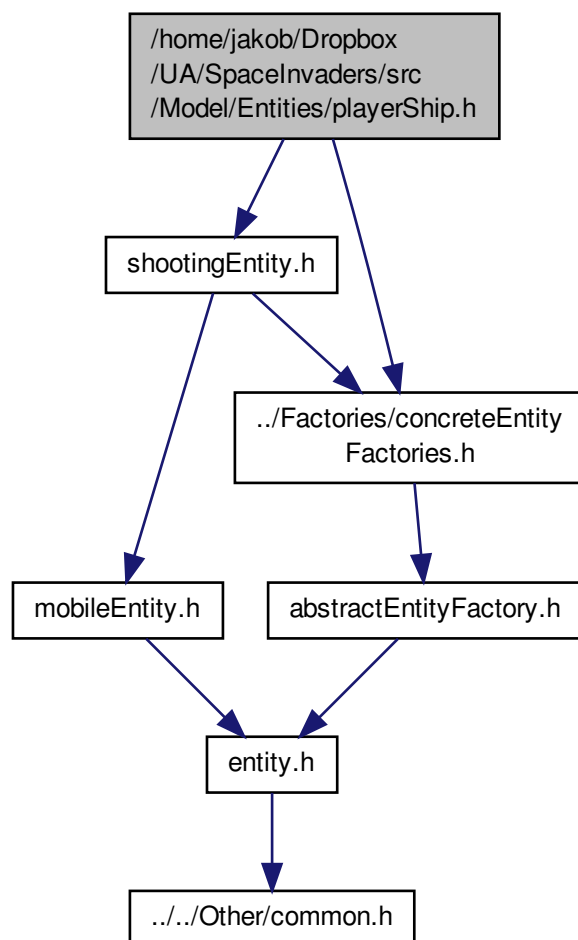
- [Model](#)

Constant Groups

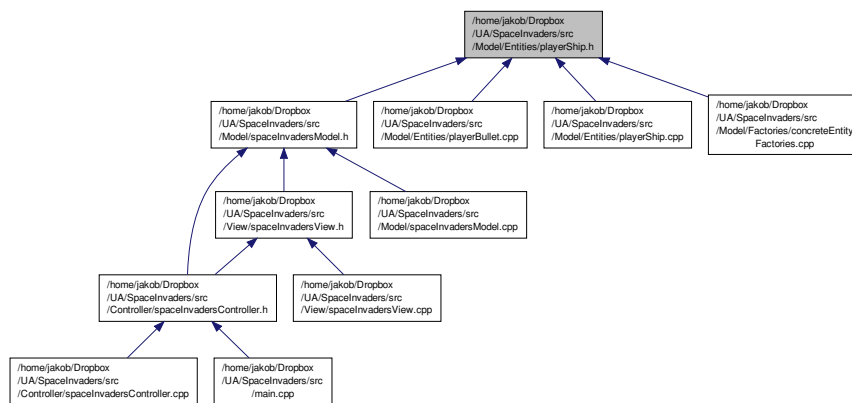
- [Model](#)

7.17 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/playerShip.h File Reference

```
#include "shootingEntity.h"  
#include "../Factories/concreteEntityFactories.h"  
Include dependency graph for playerShip.h:
```



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



Classes

- class [Model::PlayerShip](#)

Namespaces

- [Model](#)

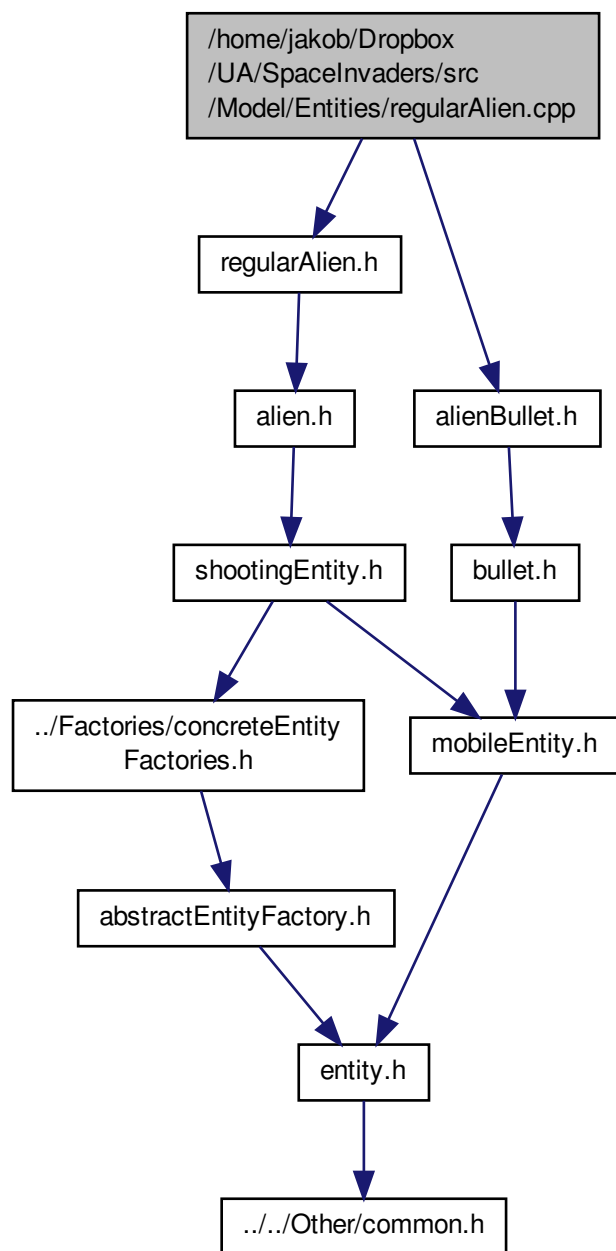
Constant Groups

- [Model](#)

7.18 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlien.cpp File Reference

```
#include "regularAlien.h"
#include "alienBullet.h"
```

Include dependency graph for regularAlien.cpp:



Namespaces

- [Model](#)

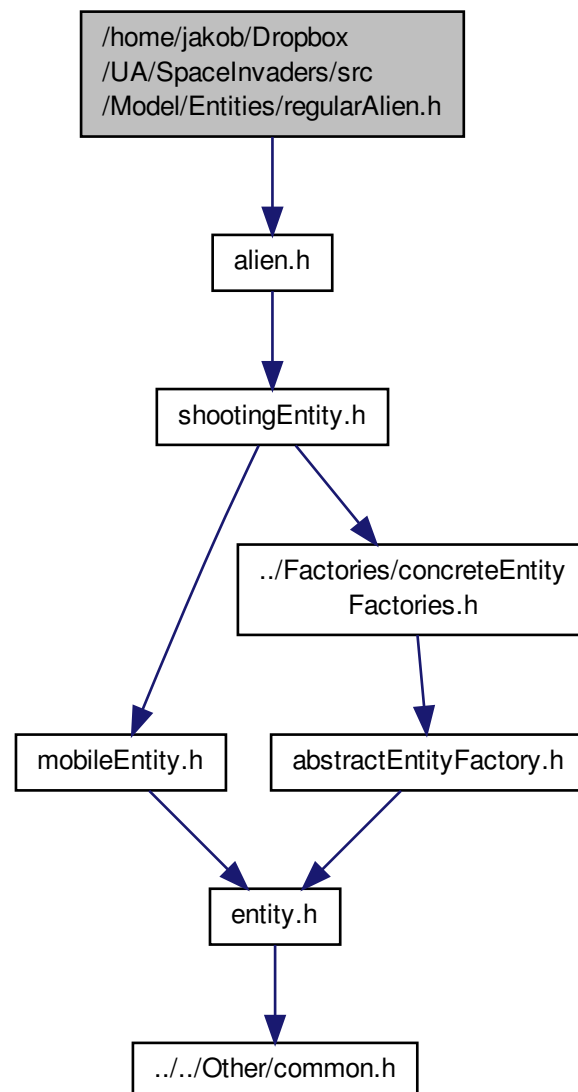
Constant Groups

- [Model](#)

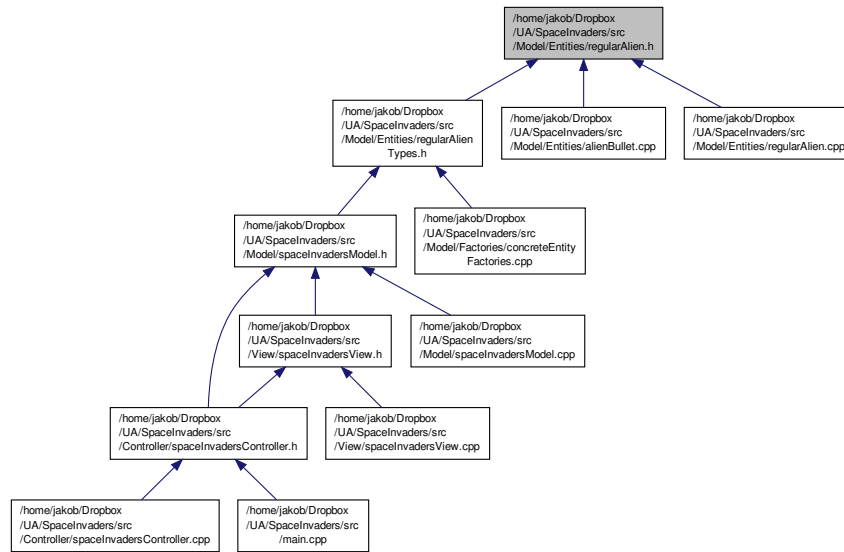
7.19 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlien.h File Reference

```
#include "alien.h"
```

Include dependency graph for regularAlien.h:



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



Classes

- class [Model::RegularAlien](#)

Namespaces

- [Model](#)

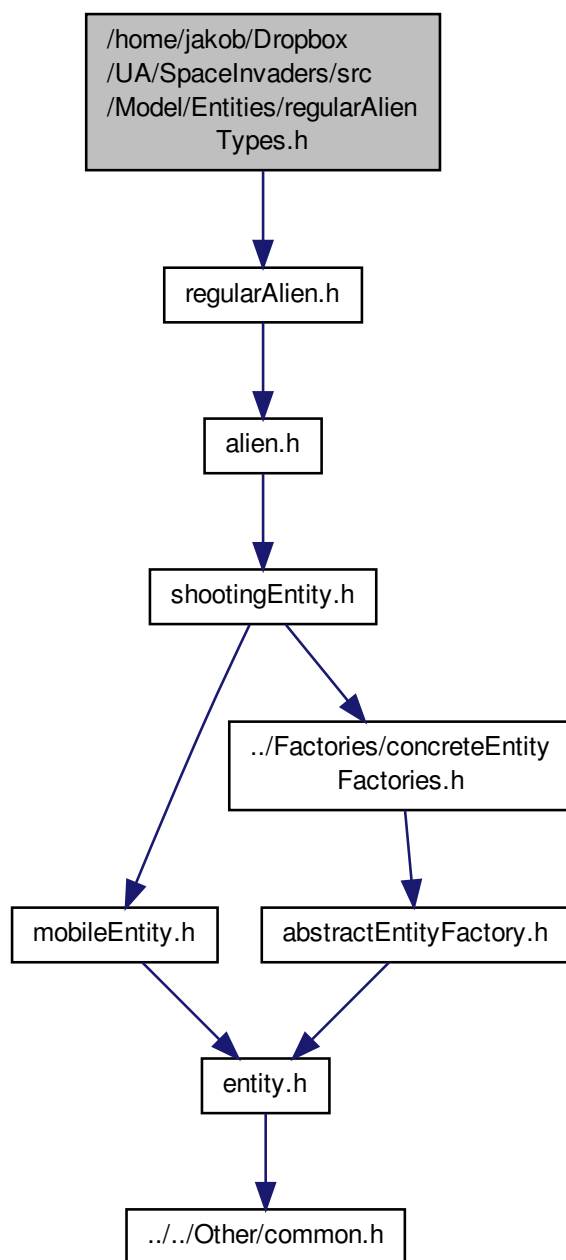
Constant Groups

- [Model](#)

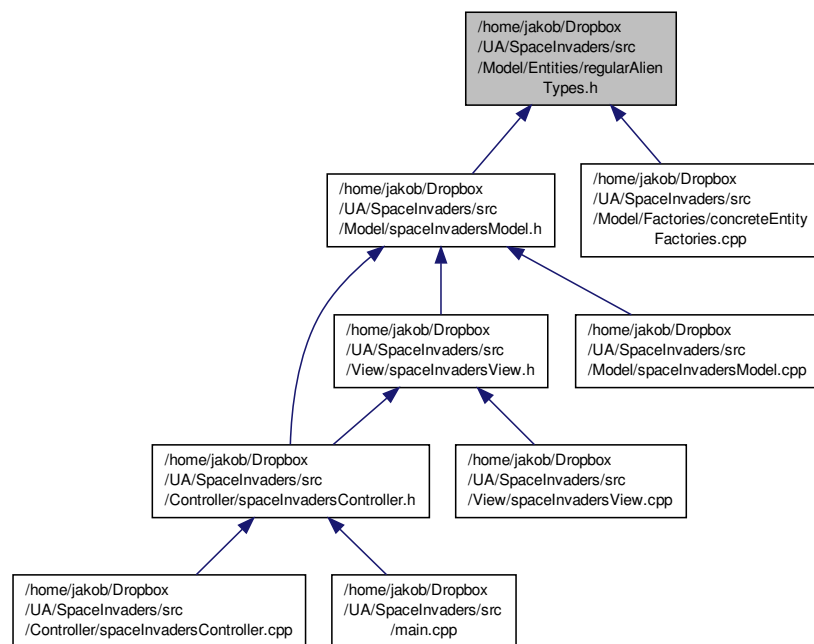
7.20 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/regularAlienTypes.h File Reference

```
#include "regularAlien.h"
```

Include dependency graph for regularAlienTypes.h:



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



Classes

- class [Model::RegularAlienTop](#)
- class [Model::RegularAlienMid](#)
- class [Model::RegularAlienBot](#)

Namespaces

- [Model](#)

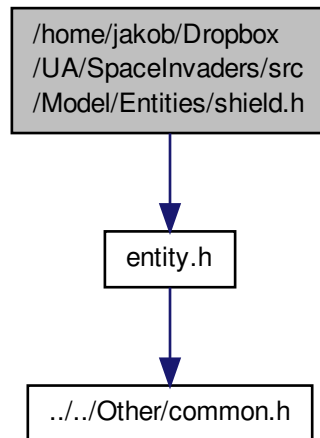
Constant Groups

- [Model](#)

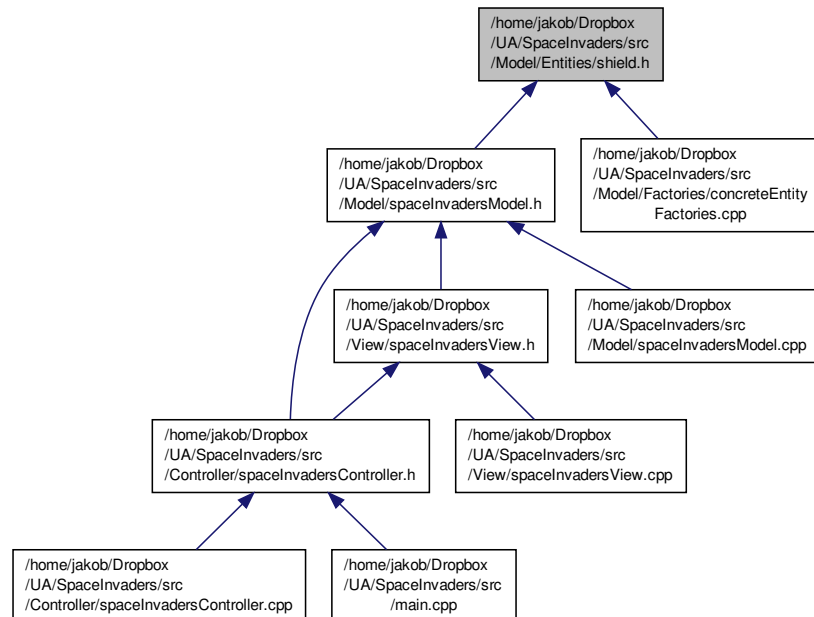
7.21 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/shield.h File Reference

```
#include "entity.h"
```

Include dependency graph for shield.h:



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



Classes

- class `Model::Shield`

Namespaces

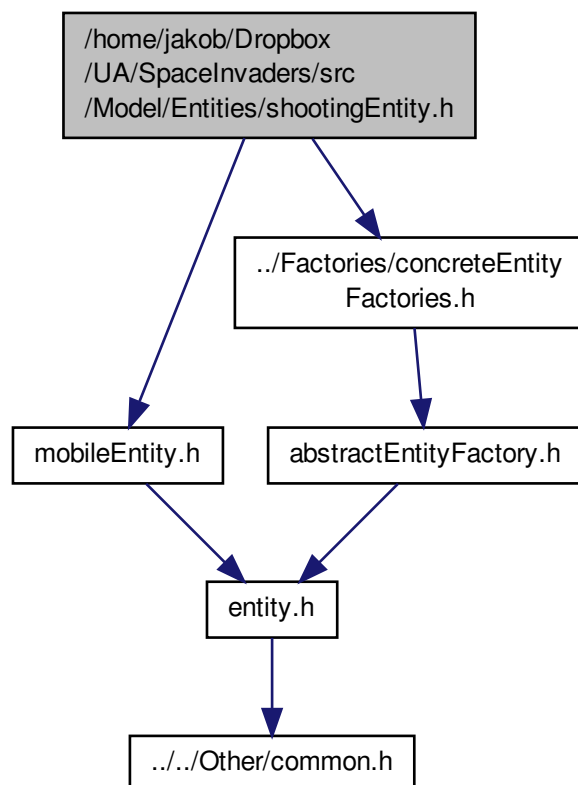
- [Model](#)

Constant Groups

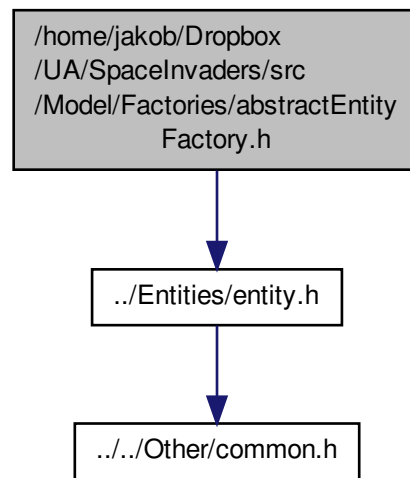
- [Model](#)

7.22 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Entities/shootingEntity.h File Reference

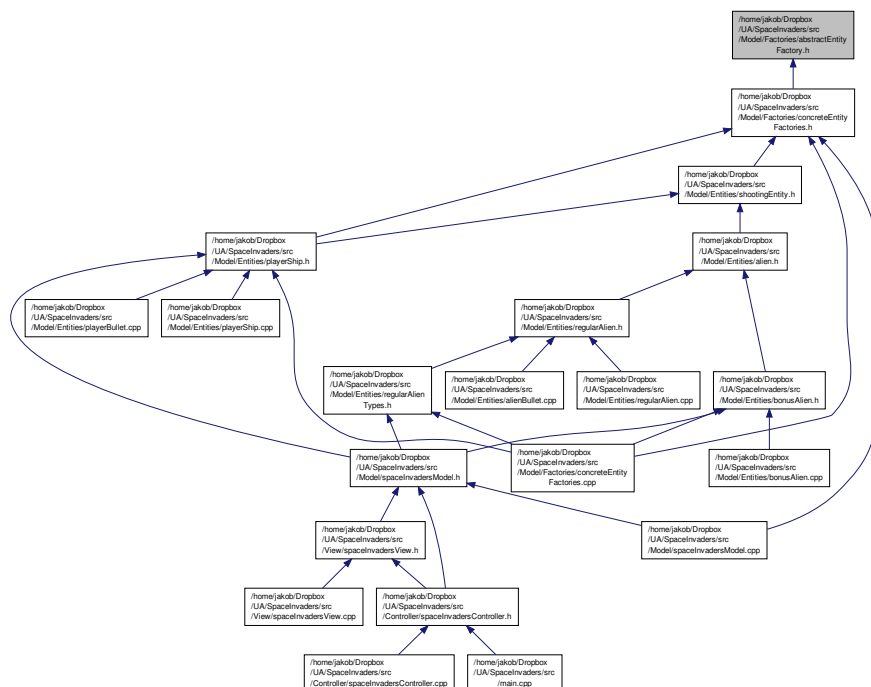
```
#include "mobileEntity.h"  
#include "../Factories/concreteEntityFactories.h"  
Include dependency graph for shootingEntity.h:
```



Include dependency graph for abstractEntityFactory.h:



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



Classes

- class [Factory::EntityFactory](#)

Namespaces

- [Factory](#)

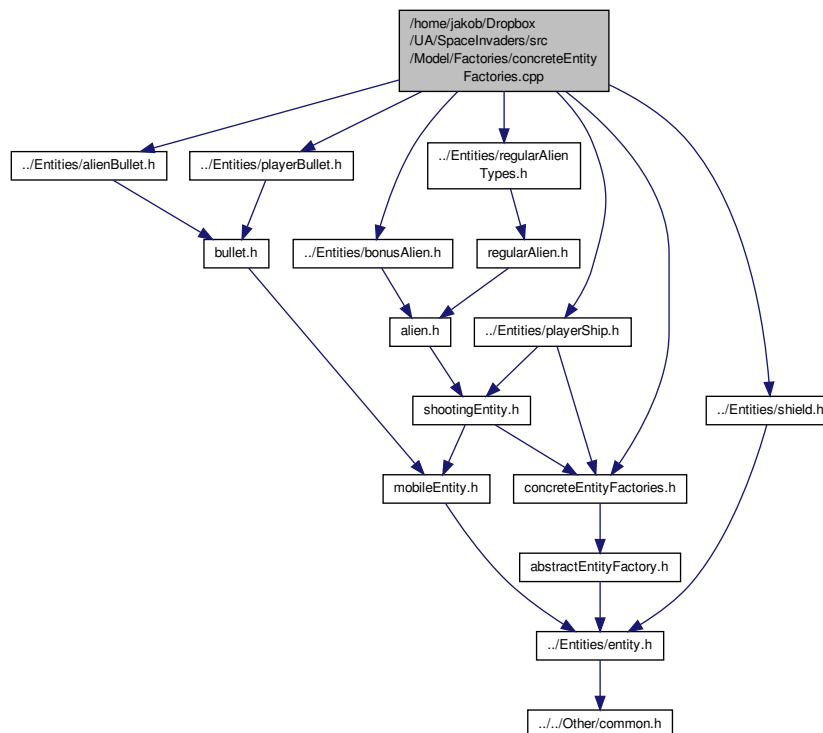
Constant Groups

- [Factory](#)

7.24 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.cpp File Reference

```
#include "concreteEntityFactories.h"
#include "../Entities/alienBullet.h"
#include "../Entities/bonusAlien.h"
#include "../Entities/playerBullet.h"
#include "../Entities/playerShip.h"
#include "../Entities/regularAlienTypes.h"
#include "../Entities/shield.h"
```

Include dependency graph for concreteEntityFactories.cpp:



Namespaces

- [Factory](#)

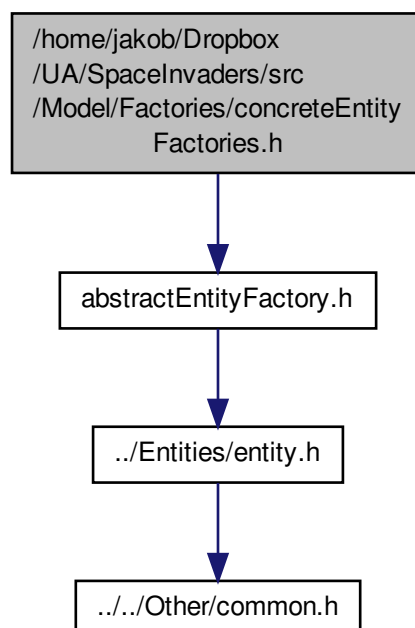
Constant Groups

- [Factory](#)

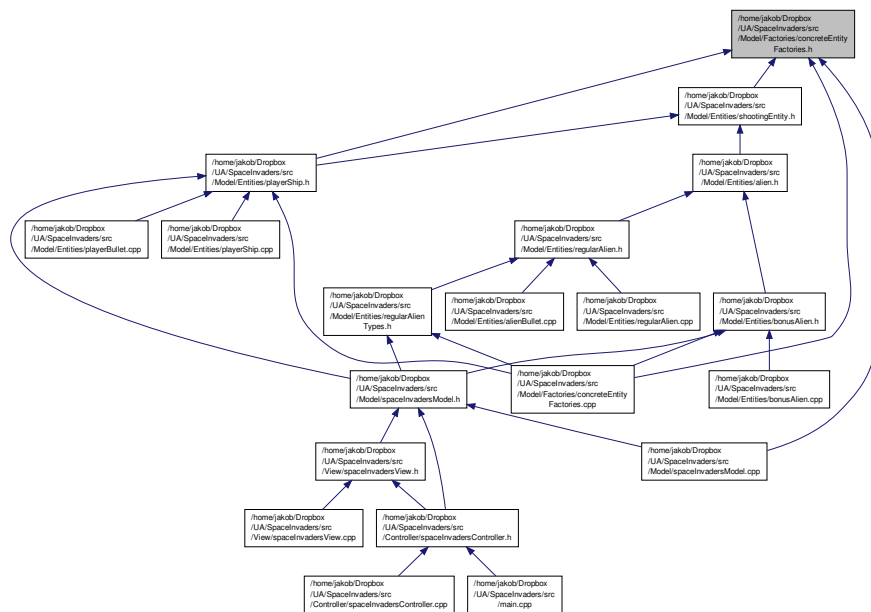
7.25 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/Factories/concreteEntityFactories.h File Reference

```
#include "abstractEntityFactory.h"
```

Include dependency graph for concreteEntityFactories.h:



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



Classes

- class [Factory::AlienBulletFactory](#)
- class [Factory::BonusAlienFactory](#)
- class [Factory::PlayerBulletFactory](#)
- class [Factory::PlayerShipFactory](#)
- class [Factory::RegularAlienBotFactory](#)
- class [Factory::RegularAlienMidFactory](#)
- class [Factory::RegularAlienTopFactory](#)
- class [Factory::ShieldFactory](#)

Namespaces

- [Factory](#)

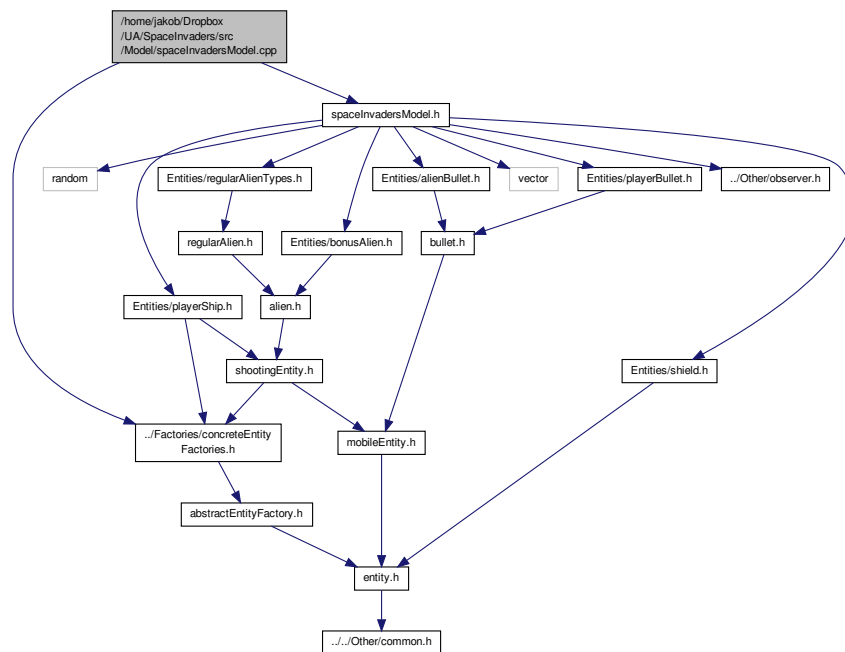
Constant Groups

- [Factory](#)

7.26 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/spacInvadersModel.cpp File Reference

```
#include "spaceInvadersModel.h"
#include "../Factories/concreteEntityFactories.h"
```

Include dependency graph for spaceInvadersModel.cpp:



Namespaces

- [Model](#)

Constant Groups

- [Model](#)

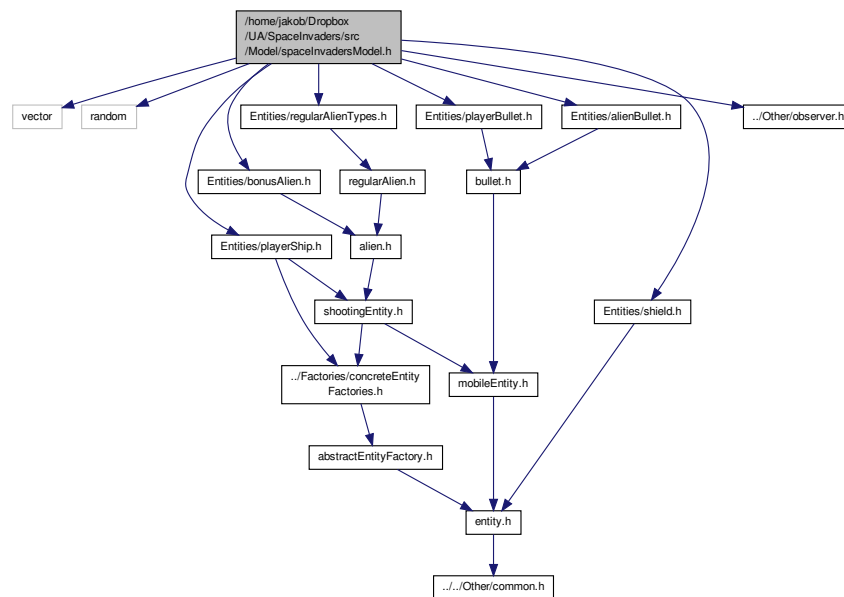
7.27 /home/jakob/Dropbox/UA/SpaceInvaders/src/Model/spaceInvadersModel.h File Reference

```

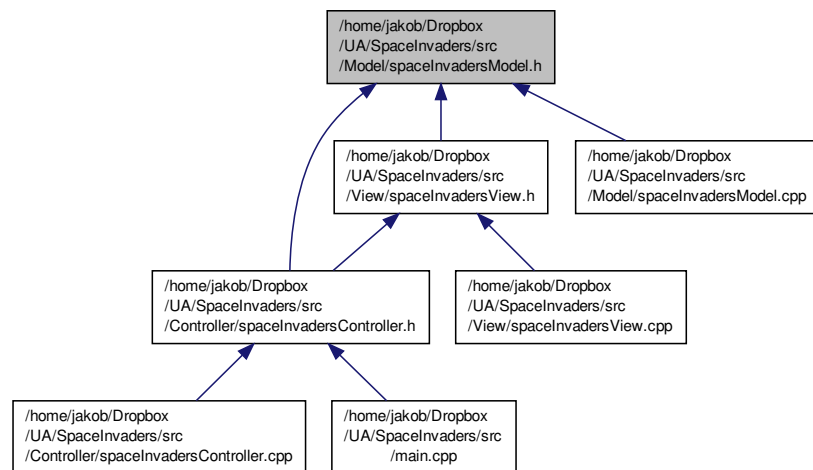
#include <vector>
#include <random>
#include "Entities/regularAlienTypes.h"
#include "Entities/playerShip.h"
#include "Entities/shield.h"
#include "Entities/alienBullet.h"
#include "Entities/playerBullet.h"
#include "Entities/bonusAlien.h"
#include "../Other/observer.h"

```

Include dependency graph for spacelInvadersModel.h:



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



Classes

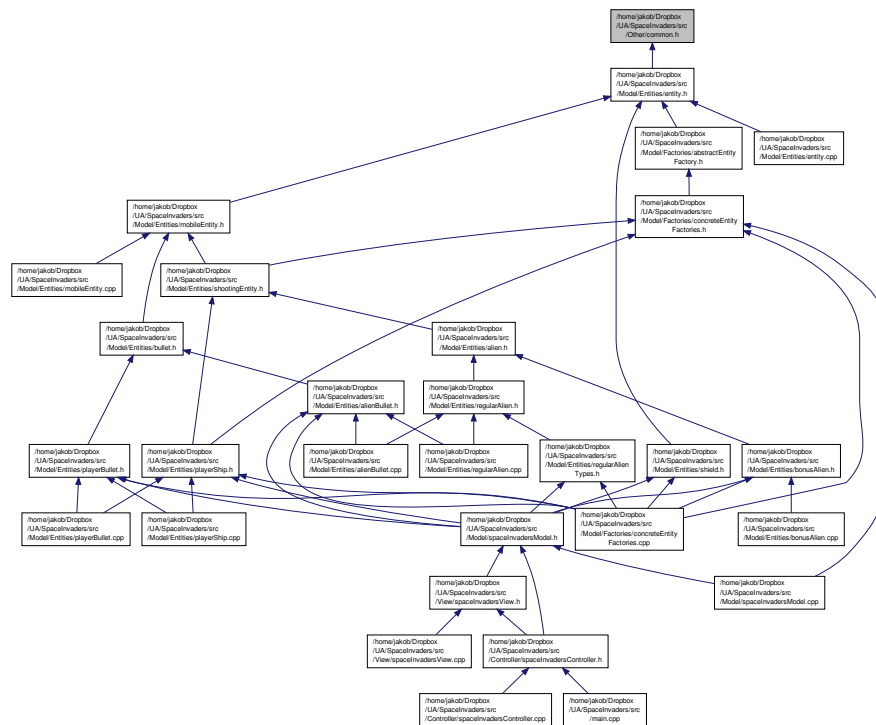
- class [Model::SpaceInvadersModel](#)

Namespaces

- [Model](#)

- Model

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

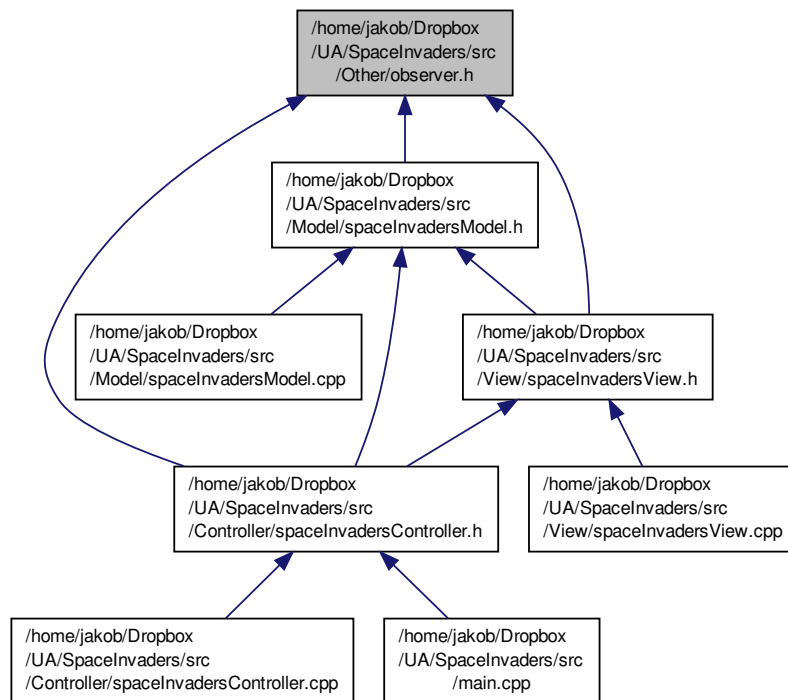


- enum **Direction** { L, R, U, D }

L
R
U
D

7.29 /home/jakob/Dropbox/UA/SpaceInvaders/src/Other/observer.h File Reference

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



Classes

- class [Observer](#)

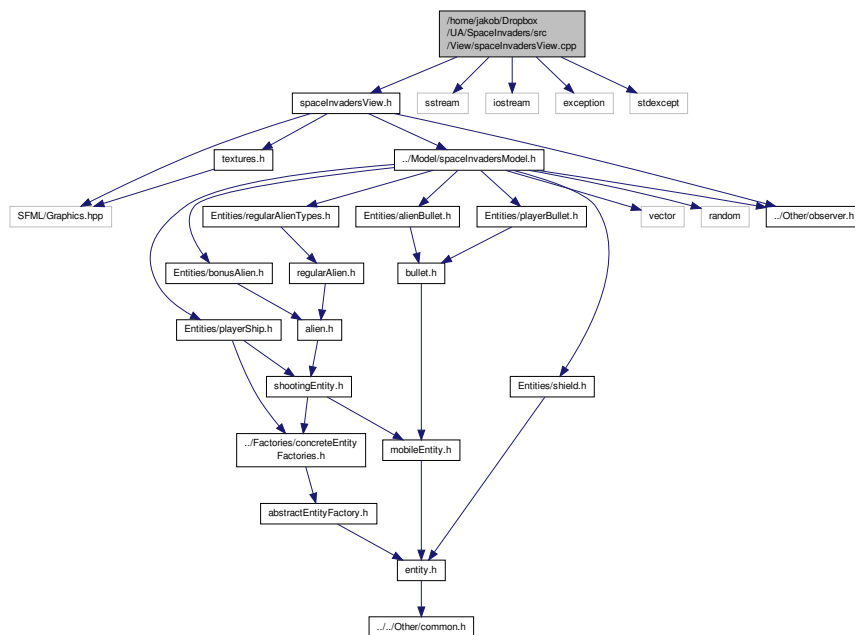
7.30 /home/jakob/Dropbox/UA/SpaceInvaders/src/View/spaceInvadersView.cpp File Reference

```

#include "spaceInvadersView.h"
#include <sstream>
#include <iostream>
#include <exception>
#include <stdexcept>

```

Include dependency graph for spacInvadersView.cpp:



Namespaces

- [View](#)

Constant Groups

- [View](#)

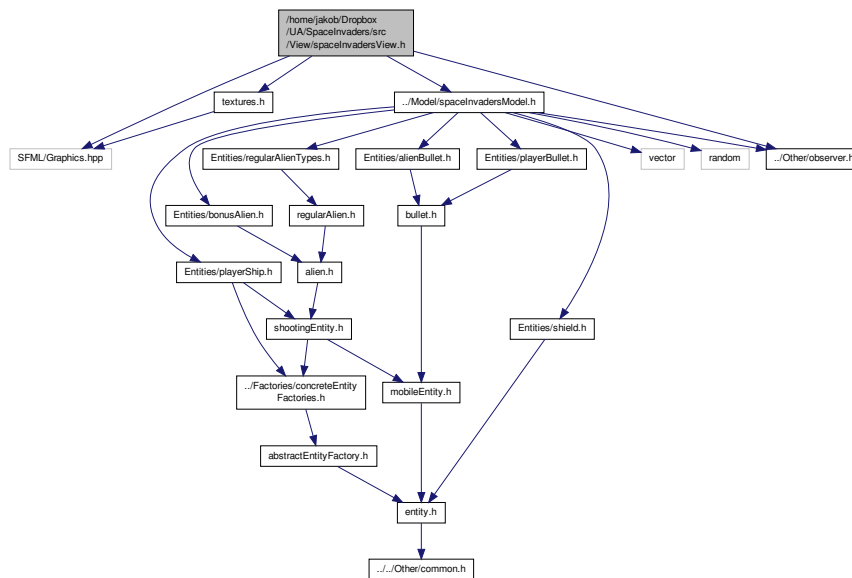
7.31 /home/jakob/Dropbox/UA/SpaceInvaders/src/View/spacInvadersView.h File Reference

```

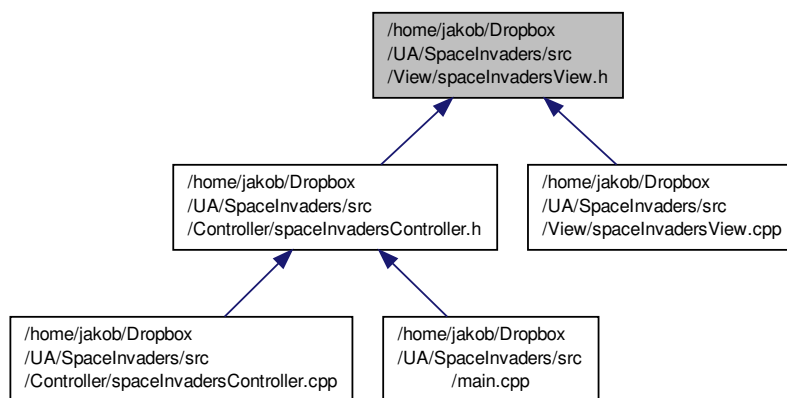
#include <SFML/Graphics.hpp>
#include "../Model/spacInvadersModel.h"
#include "textures.h"
#include "../Other/observer.h"

```

Include dependency graph for spacelInvadersView.h:



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



Classes

- class [View::SpaceInvadersSFML](#)

Namespaces

- [View](#)

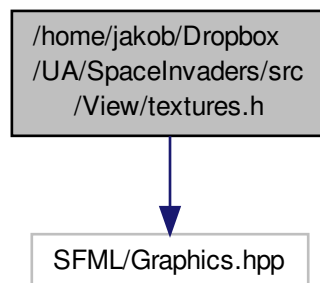
Constant Groups

- [View](#)

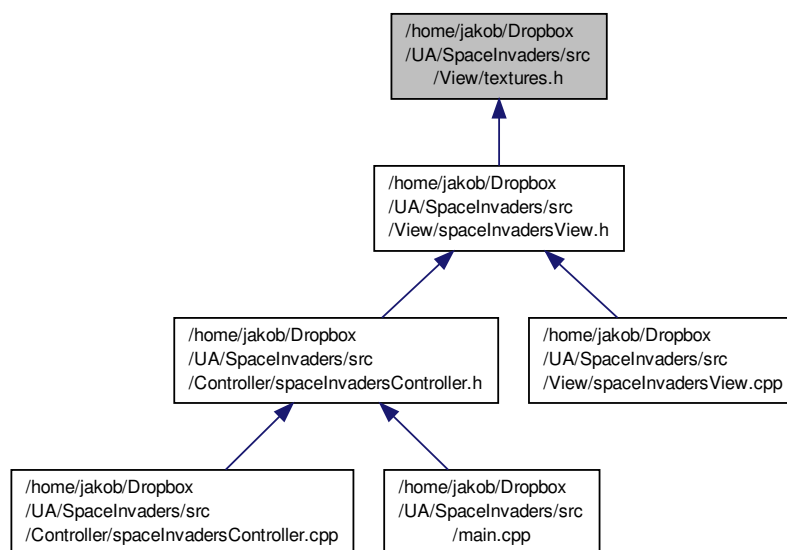
7.32 /home/jakob/Dropbox/UA/SpaceInvaders/src/View/textures.h File Reference

```
#include <SFML/Graphics.hpp>
```

Include dependency graph for textures.h:



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



Classes

- struct [View::Textures](#)

Namespaces

- [View](#)

Constant Groups

- [View](#)