

Tower Defense 1

Generated by Doxygen 1.9.3

| | |
|--|-----------|
| 1 Header files | 1 |
| 2 Source content | 3 |
| 3 Hierarchical Index | 5 |
| 3.1 Class Hierarchy | 5 |
| 4 Class Index | 7 |
| 4.1 Class List | 7 |
| 5 File Index | 9 |
| 5.1 File List | 9 |
| 6 Class Documentation | 11 |
| 6.1 BasicGoose Class Reference | 11 |
| 6.1.1 Detailed Description | 11 |
| 6.1.2 Constructor & Destructor Documentation | 11 |
| 6.1.2.1 BasicGoose() | 11 |
| 6.1.3 Member Function Documentation | 12 |
| 6.1.3.1 AcquireTarget | 12 |
| 6.1.3.2 AttackTarget() | 12 |
| 6.2 BuildIcon< T > Class Template Reference | 12 |
| 6.2.1 Detailed Description | 12 |
| 6.2.2 Constructor & Destructor Documentation | 13 |
| 6.2.2.1 BuildIcon() | 13 |
| 6.2.3 Member Function Documentation | 13 |
| 6.2.3.1 mousePressEvent() | 13 |
| 6.3 Bullet Class Reference | 14 |
| 6.3.1 Detailed Description | 14 |
| 6.3.2 Constructor & Destructor Documentation | 14 |
| 6.3.2.1 Bullet() | 14 |
| 6.3.3 Member Function Documentation | 15 |
| 6.3.3.1 GetDamage() | 15 |
| 6.3.3.2 GetDistanceTraveled() | 15 |
| 6.3.3.3 GetMaxRange() | 15 |
| 6.3.3.4 Move | 15 |
| 6.3.3.5 SetDistanceTraveled() | 15 |
| 6.3.3.6 SetMaxRange() | 15 |
| 6.3.4 Member Data Documentation | 16 |
| 6.3.4.1 bullet_damage_ | 16 |
| 6.3.4.2 distance_traveled_ | 16 |
| 6.3.4.3 max_range_ | 16 |
| 6.3.4.4 speed_ | 16 |
| 6.4 Cruiseship Class Reference | 16 |

| | |
|--|----|
| 6.4.1 Detailed Description | 17 |
| 6.4.2 Constructor & Destructor Documentation | 17 |
| 6.4.2.1 Cruiseship() | 17 |
| 6.4.3 Member Function Documentation | 17 |
| 6.4.3.1 CheckPoop() | 17 |
| 6.4.3.2 ChoosePath() | 17 |
| 6.4.3.3 Death() | 18 |
| 6.5 Dokaani Class Reference | 18 |
| 6.5.1 Detailed Description | 18 |
| 6.5.2 Constructor & Destructor Documentation | 18 |
| 6.5.2.1 Dokaani() | 18 |
| 6.5.3 Member Function Documentation | 19 |
| 6.5.3.1 ChoosePath() | 19 |
| 6.5.3.2 Death() | 19 |
| 6.5.3.3 MoveForward() | 19 |
| 6.6 Editor Class Reference | 19 |
| 6.6.1 Detailed Description | 20 |
| 6.6.2 Constructor & Destructor Documentation | 20 |
| 6.6.2.1 Editor() | 20 |
| 6.6.2.2 ~Editor() | 21 |
| 6.6.3 Member Function Documentation | 21 |
| 6.6.3.1 closeEvent() | 21 |
| 6.6.3.2 CreatePath() | 21 |
| 6.6.3.3 GetCursor() | 21 |
| 6.6.3.4 GetScene() | 21 |
| 6.6.3.5 mouseMoveEvent() | 21 |
| 6.6.3.6 mousePressEvent() | 21 |
| 6.6.3.7 ResetCursor() | 22 |
| 6.6.3.8 SavePathToFile | 22 |
| 6.6.3.9 SetCursor() | 22 |
| 6.6.3.10 TogglePathMode | 22 |
| 6.7 Enemy Class Reference | 22 |
| 6.7.1 Detailed Description | 23 |
| 6.7.2 Constructor & Destructor Documentation | 23 |
| 6.7.2.1 Enemy() | 23 |
| 6.7.2.2 ~Enemy() | 24 |
| 6.7.3 Member Function Documentation | 24 |
| 6.7.3.1 CheckPoop() | 24 |
| 6.7.3.2 ChoosePath() | 24 |
| 6.7.3.3 Death() | 24 |
| 6.7.3.4 DistanceLeft() | 25 |
| 6.7.3.5 GetDest() | 25 |

| | |
|--|----|
| 6.7.3.6 GetSpeed() | 25 |
| 6.7.3.7 MoveForward | 25 |
| 6.7.3.8 ReachDest() | 25 |
| 6.7.3.9 RotateToFacePoint() | 25 |
| 6.7.4 Member Data Documentation | 26 |
| 6.7.4.1 damage_ | 26 |
| 6.7.4.2 dest_ | 26 |
| 6.7.4.3 distance_left_ | 26 |
| 6.7.4.4 distance_traveled_ | 26 |
| 6.7.4.5 enemy_center_ | 26 |
| 6.7.4.6 enemy_hp_ | 26 |
| 6.7.4.7 game_ | 26 |
| 6.7.4.8 path_points_ | 27 |
| 6.7.4.9 point_index_ | 27 |
| 6.7.4.10 price_ | 27 |
| 6.7.4.11 speed_ | 27 |
| 6.7.4.12 timer_ | 27 |
| 6.8 Fyysikko Class Reference | 27 |
| 6.8.1 Detailed Description | 28 |
| 6.8.2 Constructor & Destructor Documentation | 28 |
| 6.8.2.1 Fyysikko() | 28 |
| 6.8.3 Member Function Documentation | 28 |
| 6.8.3.1 ChoosePath() | 28 |
| 6.9 Game Class Reference | 29 |
| 6.9.1 Detailed Description | 30 |
| 6.9.2 Constructor & Destructor Documentation | 30 |
| 6.9.2.1 Game() | 30 |
| 6.9.2.2 ~Game() | 30 |
| 6.9.3 Member Function Documentation | 30 |
| 6.9.3.1 ClearTowers | 30 |
| 6.9.3.2 closeEvent() | 30 |
| 6.9.3.3 CreatePaths() | 31 |
| 6.9.3.4 GameOver() | 31 |
| 6.9.3.5 GetBuild() | 31 |
| 6.9.3.6 GetCursor() | 31 |
| 6.9.3.7 GetHealthBar() | 31 |
| 6.9.3.8 GetMoney() | 31 |
| 6.9.3.9 GetPaths() | 31 |
| 6.9.3.10 GetScene() | 32 |
| 6.9.3.11 GetTowers() | 32 |
| 6.9.3.12 IsGameOver() | 32 |
| 6.9.3.13 mouseMoveEvent() | 32 |

| | |
|---|----|
| 6.9.3.14 mousePressEvent() | 32 |
| 6.9.3.15 PlayCruiseshipDiesSfx() | 32 |
| 6.9.3.16 PlayDokaaniDiesSfx() | 33 |
| 6.9.3.17 PlayEnemyDiesSfx() | 33 |
| 6.9.3.18 PlayHonkSfx() | 33 |
| 6.9.3.19 RemoveTower | 33 |
| 6.9.3.20 ResetCursor() | 33 |
| 6.9.3.21 SetBuild() | 33 |
| 6.9.3.22 SetCursor() | 33 |
| 6.9.3.23 SetMoney() | 34 |
| 6.9.3.24 SetPriceText() | 34 |
| 6.9.3.25 SpawnEnemy | 34 |
| 6.9.3.26 StartWave | 34 |
| 6.9.3.27 UpdateMoneyText() | 34 |
| 6.9.3.28 UpdateWaveText() | 34 |
| 6.9.3.29 UpgradeTower | 34 |
| 6.10 GoldenBullet Class Reference | 35 |
| 6.10.1 Detailed Description | 35 |
| 6.10.2 Constructor & Destructor Documentation | 35 |
| 6.10.2.1 GoldenBullet() | 35 |
| 6.11 Koneteekkari Class Reference | 35 |
| 6.11.1 Detailed Description | 36 |
| 6.11.2 Constructor & Destructor Documentation | 36 |
| 6.11.2.1 Koneteekkari() | 36 |
| 6.11.3 Member Function Documentation | 36 |
| 6.11.3.1 ChoosePath() | 36 |
| 6.12 Kylteri Class Reference | 37 |
| 6.12.1 Detailed Description | 37 |
| 6.12.2 Constructor & Destructor Documentation | 37 |
| 6.12.2.1 Kylteri() | 37 |
| 6.12.3 Member Function Documentation | 38 |
| 6.12.3.1 ChoosePath() | 38 |
| 6.13 MamaGoose Class Reference | 38 |
| 6.13.1 Detailed Description | 39 |
| 6.13.2 Constructor & Destructor Documentation | 39 |
| 6.13.2.1 MamaGoose() | 39 |
| 6.13.3 Member Function Documentation | 39 |
| 6.13.3.1 AcquireTarget | 39 |
| 6.13.3.2 AttackTarget() | 39 |
| 6.14 Menu Class Reference | 40 |
| 6.14.1 Detailed Description | 40 |
| 6.14.2 Constructor & Destructor Documentation | 40 |

| | |
|---|----|
| 6.14.2.1 Menu() | 40 |
| 6.14.3 Member Function Documentation | 41 |
| 6.14.3.1 closeEvent() | 41 |
| 6.14.3.2 GetScene() | 41 |
| 6.14.3.3 mousePressEvent() | 41 |
| 6.14.3.4 ReadPathsFromFile() | 41 |
| 6.14.3.5 StartCustom | 42 |
| 6.14.3.6 StartEditor | 42 |
| 6.14.3.7 StartLevel1 | 42 |
| 6.14.3.8 StartLevel2 | 42 |
| 6.14.3.9 StartLevel3 | 42 |
| 6.14.3.10 StartLevel4 | 42 |
| 6.14.3.11 StartLevel5 | 42 |
| 6.15 PlasmaBall Class Reference | 42 |
| 6.15.1 Detailed Description | 43 |
| 6.15.2 Constructor & Destructor Documentation | 43 |
| 6.15.2.1 PlasmaBall() | 43 |
| 6.16 Poop Class Reference | 43 |
| 6.16.1 Detailed Description | 43 |
| 6.16.2 Constructor & Destructor Documentation | 44 |
| 6.16.2.1 Poop() | 44 |
| 6.17 PooperGoose Class Reference | 44 |
| 6.17.1 Detailed Description | 44 |
| 6.17.2 Constructor & Destructor Documentation | 44 |
| 6.17.2.1 PooperGoose() | 44 |
| 6.17.3 Member Function Documentation | 45 |
| 6.17.3.1 AcquireTarget | 45 |
| 6.17.3.2 AttackTarget() | 45 |
| 6.18 ShotgunGoose Class Reference | 45 |
| 6.18.1 Detailed Description | 46 |
| 6.18.2 Constructor & Destructor Documentation | 46 |
| 6.18.2.1 ShotgunGoose() | 46 |
| 6.18.3 Member Function Documentation | 46 |
| 6.18.3.1 AcquireTarget | 46 |
| 6.18.3.2 AttackTarget() | 46 |
| 6.19 SniperGoose Class Reference | 47 |
| 6.19.1 Detailed Description | 47 |
| 6.19.2 Constructor & Destructor Documentation | 47 |
| 6.19.2.1 SniperGoose() | 47 |
| 6.19.3 Member Function Documentation | 48 |
| 6.19.3.1 AcquireTarget | 48 |
| 6.19.3.2 AttackTarget() | 48 |

| | |
|---|-----------|
| 6.20 Tower Class Reference | 48 |
| 6.20.1 Detailed Description | 49 |
| 6.20.2 Constructor & Destructor Documentation | 49 |
| 6.20.2.1 Tower() | 49 |
| 6.20.2.2 ~Tower() | 49 |
| 6.20.3 Member Function Documentation | 49 |
| 6.20.3.1 AcquireTarget | 50 |
| 6.20.3.2 AttackTarget() | 50 |
| 6.20.3.3 DistanceTo() | 50 |
| 6.20.3.4 GetAttackRadius() | 50 |
| 6.20.3.5 GetHeight() | 50 |
| 6.20.3.6 GetWidth() | 50 |
| 6.20.3.7 UpgradeAttackRadius() | 50 |
| 6.20.4 Member Data Documentation | 51 |
| 6.20.4.1 attack_area_ | 51 |
| 6.20.4.2 attack_dest_ | 51 |
| 6.20.4.3 attack_radius_ | 51 |
| 6.20.4.4 attack_speed_ | 51 |
| 6.20.4.5 has_target_ | 51 |
| 6.20.4.6 points_ | 51 |
| 6.20.4.7 scene_ | 52 |
| 6.20.4.8 tower_center_ | 52 |
| 6.20.4.9 tower_height_ | 52 |
| 6.20.4.10 tower_width_ | 52 |
| 7 File Documentation | 53 |
| 7.1 CMakeLists.txt File Reference | 53 |
| 7.2 include/basicgoose.hpp File Reference | 53 |
| 7.3 basicgoose.hpp | 53 |
| 7.4 include/buildicon.hpp File Reference | 53 |
| 7.5 buildicon.hpp | 54 |
| 7.6 include/bullet.hpp File Reference | 54 |
| 7.7 bullet.hpp | 55 |
| 7.8 include/cruiseship.hpp File Reference | 55 |
| 7.9 cruiseship.hpp | 55 |
| 7.10 include/dokaani.hpp File Reference | 56 |
| 7.11 dokaani.hpp | 56 |
| 7.12 include/editor.hpp File Reference | 56 |
| 7.13 editor.hpp | 57 |
| 7.14 include/enemy.hpp File Reference | 57 |
| 7.15 enemy.hpp | 58 |
| 7.16 include/fyysikko.hpp File Reference | 58 |

| | |
|--|----|
| 7.17 fyysikko.hpp | 58 |
| 7.18 include/game.hpp File Reference | 59 |
| 7.19 game.hpp | 59 |
| 7.20 include/goldenbullet.hpp File Reference | 60 |
| 7.21 goldenbullet.hpp | 61 |
| 7.22 include/koneteekkari.hpp File Reference | 61 |
| 7.23 koneteekkari.hpp | 61 |
| 7.24 include/kylteri.hpp File Reference | 61 |
| 7.25 kylteri.hpp | 62 |
| 7.26 include/mamagoose.hpp File Reference | 62 |
| 7.27 mamagoose.hpp | 62 |
| 7.28 include/menu.hpp File Reference | 62 |
| 7.29 menu.hpp | 63 |
| 7.30 include/plasmaball.hpp File Reference | 63 |
| 7.31 plasmaball.hpp | 64 |
| 7.32 include/poop.hpp File Reference | 64 |
| 7.33 poop.hpp | 64 |
| 7.34 include/poopergoose.hpp File Reference | 64 |
| 7.35 poopergoose.hpp | 65 |
| 7.36 include/readme.md File Reference | 65 |
| 7.37 src/readme.md File Reference | 65 |
| 7.38 include/shotgungoose.hpp File Reference | 65 |
| 7.39 shotgungoose.hpp | 65 |
| 7.40 include/snipergoose.hpp File Reference | 66 |
| 7.41 snipergoose.hpp | 66 |
| 7.42 include/tower.hpp File Reference | 66 |
| 7.43 tower.hpp | 67 |
| 7.44 spelling.txt File Reference | 67 |
| 7.45 src/basicgoose.cpp File Reference | 67 |
| 7.46 src/bullet.cpp File Reference | 67 |
| 7.47 src/cruiseship.cpp File Reference | 68 |
| 7.48 src/dokaani.cpp File Reference | 68 |
| 7.49 src/editor.cpp File Reference | 68 |
| 7.49.1 Macro Definition Documentation | 68 |
| 7.49.1.1 WINDOW_HEIGHT | 68 |
| 7.49.1.2 WINDOW_WIDTH | 69 |
| 7.50 src/enemy.cpp File Reference | 69 |
| 7.51 src/fyysikko.cpp File Reference | 69 |
| 7.52 src/game.cpp File Reference | 69 |
| 7.52.1 Macro Definition Documentation | 70 |
| 7.52.1.1 WINDOW_HEIGHT | 70 |
| 7.52.1.2 WINDOW_WIDTH | 70 |

| | |
|--|-----------|
| 7.53 src/goldenbullet.cpp File Reference | 70 |
| 7.54 src/koneteekkari.cpp File Reference | 70 |
| 7.55 src/kylteri.cpp File Reference | 70 |
| 7.56 src/main.cpp File Reference | 70 |
| 7.56.1 Function Documentation | 71 |
| 7.56.1.1 main() | 71 |
| 7.57 src/mamagoose.cpp File Reference | 71 |
| 7.58 src/menu.cpp File Reference | 71 |
| 7.58.1 Macro Definition Documentation | 72 |
| 7.58.1.1 WINDOW_HEIGHT | 72 |
| 7.58.1.2 WINDOW_WIDTH | 72 |
| 7.59 src/plasmaball.cpp File Reference | 72 |
| 7.60 src/poop.cpp File Reference | 72 |
| 7.61 src/poopergoose.cpp File Reference | 72 |
| 7.62 src/shotgungoose.cpp File Reference | 73 |
| 7.63 src/snipergoose.cpp File Reference | 73 |
| 7.64 src/tower.cpp File Reference | 73 |
| Index | 75 |

Chapter 1

Header files

This folder should contain only .hpp files of your implementation. The .cpp files are in a separate directory `src`.

A short summary of files in this directory.

Menu, Game and Level editor \ `menu.hpp` (Includes `Game` and `Editor`) \ `game.hpp` \ `editor.hpp`

Towers \ `tower.hpp` (*Abstract base class* `Tower`) \ `basicgoose.hpp` (Inherits `Tower`) \ `poopergoose.hpp` (Inherits `Tower`) \ `shotgungoose.hpp` (Inherits `Tower`) \ `snipergoose.hpp` (Inherits `Tower`) \ `mamagoose.hpp` (Inherits `Tower`)

****Build Icon for Towers**** \ `buildicon.hpp` (*Template class* `BuildIcon` is implemented in .hpp)

Enemies \ `enemy.hpp` (*Abstract base class* `Enemy`) \ `fyysikko.hpp` (Inherits `Enemy`) \ `kylteri.hpp` (Inherits `Enemy`) \ `koneteekkari.hpp` (Inherits `Enemy`) \ `dokaani.hpp` (Inherits `Enemy`) \ `cruiseship.hpp` (Inherits `Enemy`)

Bullets and Projectiles \ `bullet.hpp` (*Base class* `Bullet`) \ `goldenbullet.hpp` (Inherits `Bullet`) \ `plasmaball.hpp` (Inherits `Bullet`) \ `poop.hpp` (Inherits `Bullet`)

Chapter 2

Source content

This folder should contain only .cpp files of your implementation. The .hpp files are in a separate directory include.

A short summary of files in this directory.

Menu, Game and Level editor \ [menu.cpp](#) (Includes [Game](#) and [Editor](#)) \ [game.cpp](#) \ [editor.cpp](#)

Towers \ [tower.cpp](#) (*Abstract base class* [Tower](#)) \ [basicgoose.cpp](#) (Inherits [Tower](#)) \ [poopergoose.cpp](#) (Inherits [Tower](#)) \ [shotgungoose.cpp](#) (Inherits [Tower](#)) \ [snipergoose.cpp](#) (Inherits [Tower](#)) \ [mamagoose.cpp](#) (Inherits [Tower](#))

****Build Icon for Towers**** \ --- (*Template class* [BuildIcon](#) is implemented in .hpp)

Enemies \ [enemy.cpp](#) (*Abstract base class* [Enemy](#)) \ [fyysikko.cpp](#) (Inherits [Enemy](#)) \ [kylteri.cpp](#) (Inherits [Enemy](#)) \ [koneteekkari.cpp](#) (Inherits [Enemy](#)) \ [dokaani.cpp](#) (Inherits [Enemy](#)) \ [cruiseship.cpp](#) (Inherits [Enemy](#))

Bullets and Projectiles \ [bullet.cpp](#) (*Base class* [Bullet](#)) \ [goldenbullet.cpp](#) (Inherits [Bullet](#)) \ [plasmaball.cpp](#) (Inherits [Bullet](#)) \ [poop.cpp](#) (Inherits [Bullet](#))

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

| | |
|--------------------------|----|
| QGraphicsPixmapItem | |
| BuildIcon< T > | 12 |
| Bullet | 14 |
| GoldenBullet | 35 |
| PlasmaBall | 42 |
| Poop | 43 |
| Enemy | 22 |
| Cruiseship | 16 |
| Dokaani | 18 |
| Fyysikko | 27 |
| Koneteekkari | 35 |
| Kylteri | 37 |
| Tower | 48 |
| BasicGoose | 11 |
| MamaGoose | 38 |
| PooperGoose | 44 |
| ShotgunGoose | 45 |
| SniperGoose | 47 |
| QGraphicsView | |
| Editor | 19 |
| Game | 29 |
| Menu | 40 |
| QObject | |
| Bullet | 14 |
| Enemy | 22 |
| Tower | 48 |

Chapter 4

Class Index

4.1 Class List

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

| | | |
|--------------------------------------|---|----|
| BasicGoose | A basic Tower that shoots basic Bullet objects | 11 |
| BuildIcon< T > | A template class for building Tower objects in a Game | 12 |
| Bullet | A base class for all projectiles/bullets that Tower class object shoot | 14 |
| Cruiseship | Cruiseship is a strong enemy which spawns new enemies when it's killed | 16 |
| Dokaani | Dokaani is the head of the enemies which takes a lot of damage but also has nice yield | 18 |
| Editor | A level editor that can be used to create custom levels | 19 |
| Enemy | Enemy is an abstract class that all enemies inherit | 22 |
| Fyysikko | Fyysikko is a smart enemy that chooses the optimized with the shortest length | 27 |
| Game | A single game of tower defense | 29 |
| GoldenBullet | A GoldenBullet is a Bullet that is fast and has high damage | 35 |
| Koneteekkari | Koneteekkari is the basic Enemy that chooses its path at random | 35 |
| Kylteri | Kylteri is a quick enemy but always chooses the path with the closest Tower | 37 |
| MamaGoose | A powerful Tower that shoots PlasmaBall objects with high frequency | 38 |
| Menu | Menu is a class for the main menu of the game | 40 |
| PlasmaBall | A PlasmaBall is a Bullet that MamaGoose uses | 42 |
| Poop | A Poop is a Bullet that slows down enemies | 43 |
| PooperGoose | A close-range Tower that slows enemies with Poop | 44 |
| ShotgunGoose | A close-range Tower that shoots multiple Bullet objects | 45 |

| | |
|--|----|
| SniperGoose | |
| A long-range Tower that shoots powerful GoldenBullet objects | 47 |
| Tower | |
| Tower is an abstract class that all towers inherit | 48 |

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

| | |
|--------------------------|----|
| include/basicgoose.hpp | 53 |
| include/buildicon.hpp | 53 |
| include/bullet.hpp | 54 |
| include/cruiseship.hpp | 55 |
| include/dokaani.hpp | 56 |
| include/editor.hpp | 56 |
| include/enemy.hpp | 57 |
| include/fyysikko.hpp | 58 |
| include/game.hpp | 59 |
| include/goldenbullet.hpp | 60 |
| include/koneteekkari.hpp | 61 |
| include/kylteri.hpp | 61 |
| include/mamagoose.hpp | 62 |
| include/menu.hpp | 62 |
| include/plasmaball.hpp | 63 |
| include/poop.hpp | 64 |
| include/poopergoose.hpp | 64 |
| include/shotgungoose.hpp | 65 |
| include/snipergoose.hpp | 66 |
| include/tower.hpp | 66 |
| src/basicgoose.cpp | 67 |
| src/bullet.cpp | 67 |
| src/cruiseship.cpp | 68 |
| src/dokaani.cpp | 68 |
| src/editor.cpp | 68 |
| src/enemy.cpp | 69 |
| src/fyysikko.cpp | 69 |
| src/game.cpp | 69 |
| src/goldenbullet.cpp | 70 |
| src/koneteekkari.cpp | 70 |
| src/kylteri.cpp | 70 |
| src/main.cpp | 70 |
| src/mamagoose.cpp | 71 |
| src/menu.cpp | 71 |
| src/plasmaball.cpp | 72 |

| | |
|---|----|
| src/ poop.cpp | 72 |
| src/ poopergoose.cpp | 72 |
| src/ shotgungoose.cpp | 73 |
| src/ snipergoose.cpp | 73 |
| src/ tower.cpp | 73 |

Chapter 6

Class Documentation

6.1 BasicGoose Class Reference

A basic [Tower](#) that shoots basic [Bullet](#) objects.

```
#include <basicgoose.hpp>
```

Inherits [Tower](#).

Public Slots

- void [AcquireTarget](#) ()
Targets the closest [Enemy](#) within its attack area.

Public Member Functions

- [BasicGoose](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a new [BasicGoose](#) object.
- void [AttackTarget](#) ()
Attacks the current target with a single [Bullet](#).

Additional Inherited Members

6.1.1 Detailed Description

A basic [Tower](#) that shoots basic [Bullet](#) objects.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 BasicGoose()

```
BasicGoose::BasicGoose (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [BasicGoose](#) object.

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.1.3 Member Function Documentation

6.1.3.1 AcquireTarget

```
void BasicGoose::AcquireTarget ( ) [slot]
```

Targets the closest [Enemy](#) within its attack area.

6.1.3.2 AttackTarget()

```
void BasicGoose::AttackTarget ( ) [virtual]
```

Attacks the current target with a single [Bullet](#).

Implements [Tower](#).

6.2 BuildIcon< T > Class Template Reference

A template class for building [Tower](#) objects in a [Game](#).

```
#include <buildicon.hpp>
```

Inherits QGraphicsPixmapItem.

Public Member Functions

- [BuildIcon](#) (QString imagepath, int price, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [BuildIcon](#) object.
- void [mousePressEvent](#) (QGraphicsSceneMouseEvent *event)
Sets the cursor in [Game](#) to the [Tower](#) image and sets the [Game](#) to Build mode.

6.2.1 Detailed Description

```
template<class T>
class BuildIcon< T >
```

A template class for building [Tower](#) objects in a [Game](#).

Template classes' implementation and definition cannot simply be separated into header and source files. This is solved using inline definitions for the class methods in this header.

Template Parameters

| | |
|----------|---|
| <i>T</i> | is a Tower subclass for which we want to create the BuildIcon |
|----------|---|

6.2.2 Constructor & Destructor Documentation

6.2.2.1 BuildIcon()

```
template<class T >
BuildIcon< T >::BuildIcon (
    QString imagepath,
    int price,
    Game * game,
    QGraphicsItem * parent = 0 ) [inline]
```

Construct a new [BuildIcon](#) object.

Template Parameters

| | |
|----------|---|
| <i>T</i> | is a Tower subclass for which we want to create the BuildIcon |
|----------|---|

Parameters

| | |
|------------------|---|
| <i>imagepath</i> | a QString file path to the wanted image |
| <i>price</i> | the price that will be associated with the Tower |
| <i>game</i> | the Game where this BuildIcon is used |
| <i>parent</i> | optional QGraphicsItem parent |

6.2.3 Member Function Documentation

6.2.3.1 mousePressEvent()

```
template<class T >
void BuildIcon< T >::mousePressEvent (
    QGraphicsSceneMouseEvent * event ) [inline]
```

Sets the cursor in [Game](#) to the [Tower](#) image and sets the [Game](#) to Build mode.

Template Parameters

| | |
|----------|---|
| <i>T</i> | is a Tower subclass for which we want to create the BuildIcon |
|----------|---|

Parameters

| | |
|--------------|---|
| <i>event</i> | a QGraphicsSceneMouseEvent that is passed as a parameter for GUI events |
|--------------|---|

6.3 Bullet Class Reference

A base class for all projectiles/bullets that [Tower](#) class object shoot.

```
#include <bullet.hpp>
```

Inherits QObject, and QGraphicsPixmapItem.

Inherited by [GoldenBullet](#), [PlasmaBall](#), and [Poop](#).

Public Slots

- void [Move](#) ()

Public Member Functions

- [Bullet](#) (QGraphicsItem *parent=0)
Construct a new [Bullet](#) object.
- double [GetMaxRange](#) ()
- void [SetMaxRange](#) (double range)
- double [GetDistanceTraveled](#) ()
- void [SetDistanceTraveled](#) (double dist)
- int [GetDamage](#) ()

Protected Attributes

- double [speed_](#)
- double [max_range_](#)
- double [distance_traveled_](#)
- int [bullet_damage_](#)

6.3.1 Detailed Description

A base class for all projectiles/bullets that [Tower](#) class object shoot.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 Bullet()

```
Bullet::Bullet (
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [Bullet](#) object.

Parameters

| | |
|---------------|---|
| <i>parent</i> | Optional QGraphicsItem pointer, default value = 0 |
|---------------|---|

6.3.3 Member Function Documentation

6.3.3.1 GetDamage()

```
int Bullet::GetDamage ( )
```

6.3.3.2 GetDistanceTraveled()

```
double Bullet::GetDistanceTraveled ( )
```

6.3.3.3 GetMaxRange()

```
double Bullet::GetMaxRange ( )
```

6.3.3.4 Move

```
void Bullet::Move ( ) [slot]
```

6.3.3.5 SetDistanceTraveled()

```
void Bullet::SetDistanceTraveled (
    double dist )
```

6.3.3.6 SetMaxRange()

```
void Bullet::SetMaxRange (
    double range )
```

6.3.4 Member Data Documentation

6.3.4.1 bullet_damage_

```
int Bullet::bullet_damage_ [protected]
```

6.3.4.2 distance_traveled_

```
double Bullet::distance_traveled_ [protected]
```

6.3.4.3 max_range_

```
double Bullet::max_range_ [protected]
```

6.3.4.4 speed_

```
double Bullet::speed_ [protected]
```

6.4 Cruiseship Class Reference

[Cruiseship](#) is a strong enemy which spawns new enemies when it's killed.

```
#include <cruiseship.hpp>
```

Inherits [Enemy](#).

Public Member Functions

- [Cruiseship](#) (QList< QList< QPointF > > paths, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [Cruiseship](#) object.
- QList< QPointF > [ChoosePath](#) (QList< QList< QPointF > > paths)
Chooses the path which starts the furthers from any [Tower](#).
- void [Death](#) ()
When cruiseship dies, spawn new enemies, update player money, and play the sound effects.
- void [CheckPoop](#) ()
If enemy gets hit by poop, reduce its volume by 0.5 units.

Additional Inherited Members

6.4.1 Detailed Description

[Cruiseship](#) is a strong enemy which spawns new enemies when it's killed.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 Cruiseship()

```
Cruiseship::Cruiseship (
    QList< QList< QPointF > > paths,
    Game * game,
    QGraphicsItem * parent = 0 )
```

Construct a new [Cruiseship](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | List of different paths on the map |
| <i>game</i> | The Game where the enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.4.3 Member Function Documentation

6.4.3.1 CheckPoop()

```
void Cruiseship::CheckPoop ( ) [virtual]
```

If enemy gets hit by poop, reduce its volume by 0.5 units.

Reimplemented from [Enemy](#).

6.4.3.2 ChoosePath()

```
QList< QPointF > Cruiseship::ChoosePath (
    QList< QList< QPointF > > paths ) [virtual]
```

Chooses the path which starts the furthers from any [Tower](#).

Parameters

| | |
|--------------|------------------------------------|
| <i>paths</i> | List of different paths on the map |
|--------------|------------------------------------|

Returns

QList<QPointF>

Implements [Enemy](#).

6.4.3.3 Death()

```
void Cruiseship::Death ( ) [virtual]
```

When cruiseship dies, spawn new enemies, update player money, and play the sound effects.

Reimplemented from [Enemy](#).

6.5 Dokaani Class Reference

[Dokaani](#) is the head of the enemies which takes a lot of damage but also has nice yield.

```
#include <dokaani.hpp>
```

Inherits [Enemy](#).

Public Member Functions

- [Dokaani](#) (QList< QList< QPointF > > paths, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [Dokaani](#) object.
- QList< QPointF > [ChoosePath](#) (QList< QList< QPointF > > paths)
Chooses the path with the longest length.
- void [MoveForward](#) ()
- void [Death](#) ()

Additional Inherited Members

6.5.1 Detailed Description

[Dokaani](#) is the head of the enemies which takes a lot of damage but also has nice yield.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 Dokaani()

```
Dokaani::Dokaani (
    QList< QList< QPointF > > paths,
    Game * game,
    QGraphicsItem * parent = 0 )
```

Construct a new [Dokaani](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | List of different paths on the map |
| <i>game</i> | The Game where the enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.5.3 Member Function Documentation

6.5.3.1 ChoosePath()

```
QList< QPointF > Dokaani::ChoosePath (
    QList< QList< QPointF > > paths ) [virtual]
```

Chooses the path with the longest length.

Parameters

| | |
|--------------|------------------------------------|
| <i>paths</i> | List of different paths on the map |
|--------------|------------------------------------|

Returns

QList<QPointF>

Implements [Enemy](#).

6.5.3.2 Death()

```
void Dokaani::Death ( ) [virtual]
```

Reimplemented from [Enemy](#).

6.5.3.3 MoveForward()

```
void Dokaani::MoveForward ( )
```

6.6 Editor Class Reference

A level editor that can be used to create custom levels.

```
#include <editor.hpp>
```

Inherits QGraphicsView.

Public Slots

- void [SavePathToFile](#) ()
Saves the paths to a .dat file using a serialized binary format, which can be read by the [Menu](#) class.
- void [TogglePathMode](#) ()
Toggling in_path_mode_ enables the creation of path points which will be visualized by the [CreatePath](#) method.

Public Member Functions

- [Editor](#) (QWidget *parent=0)
Construct a new [Editor](#) object.
- [~Editor](#) ()
- void [SetCursor](#) (QString filename)
- void [mouseMoveEvent](#) (QMouseEvent *event)
- void [mousePressEvent](#) (QMouseEvent *event)
- void [closeEvent](#) (QCloseEvent *event)
- QGraphicsScene * [GetScene](#) ()
- QGraphicsPixmapItem * [GetCursor](#) ()
- void [ResetCursor](#) ()
- void [CreatePath](#) ()
Visualizes the most recent path with a (almost) unique color.

6.6.1 Detailed Description

A level editor that can be used to create custom levels.

Levels are saved in a serialized binary format which can be read by the [Menu](#) class.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 Editor()

```
Editor::Editor (
    QWidget * parent = 0 ) [explicit]
```

Construct a new [Editor](#) object.

Parameters

| | |
|---------------|---|
| <i>parent</i> | will be Menu in this case |
|---------------|---|

6.6.2.2 ~Editor()

```
Editor::~~Editor ( )
```

6.6.3 Member Function Documentation

6.6.3.1 closeEvent()

```
void Editor::closeEvent (
    QCloseEvent * event )
```

6.6.3.2 CreatePath()

```
void Editor::CreatePath ( )
```

Visualizes the most recent path with a (almost) unique color.

6.6.3.3 GetCursor()

```
QGraphicsPixmapItem * Editor::GetCursor ( )
```

6.6.3.4 GetScene()

```
QGraphicsScene * Editor::GetScene ( )
```

6.6.3.5 mouseMoveEvent()

```
void Editor::mouseMoveEvent (
    QMouseEvent * event )
```

6.6.3.6 mousePressEvent()

```
void Editor::mousePressEvent (
    QMouseEvent * event )
```

6.6.3.7 ResetCursor()

```
void Editor::ResetCursor ( )
```

6.6.3.8 SavePathToFile

```
void Editor::SavePathToFile ( ) [slot]
```

Saves the paths to a .dat file using a serialized binary format, which can be read by the [Menu](#) class.

See also

[Menu::ReadPathsFromFile\(const QString& filename\)](#)

6.6.3.9 SetCursor()

```
void Editor::SetCursor (
    QString filename )
```

6.6.3.10 TogglePathMode

```
void Editor::TogglePathMode ( ) [slot]
```

Toggling `in_path_mode_` enables the creation of path points which will be visualized by the `CreatePath` method.

6.7 Enemy Class Reference

[Enemy](#) is an abstract class that all enemies inherit.

```
#include <enemy.hpp>
```

Inherits `QObject`, and `QGraphicsPixmapItem`.

Inherited by [Cruiseship](#), [Dokaani](#), [Fyysikko](#), [Koneteekkari](#), and [Kylteri](#).

Public Slots

- void [MoveForward](#) ()
Check if [Enemy](#) gets hit by [Poop](#) or [Bullet](#). Rotate to face the next point and check if it has reached its destination.

Public Member Functions

- [Enemy](#) ([Game](#) *game, [QGraphicsItem](#) *parent=0)
Construct a new [Enemy](#) object (Abstract class)
- [~Enemy](#) ()
- void [RotateToFacePoint](#) ([QPointF](#) p)
- [QPointF](#) [GetDest](#) ()
- virtual [QList](#)< [QPointF](#) > [ChoosePath](#) ([QList](#)< [QList](#)< [QPointF](#) > > paths)=0
- void [ReachDest](#) ()
If [Enemy](#) has reached its destination, reduce player health and/or start GameOver.
- double [GetSpeed](#) ()
- virtual void [Death](#) ()
- virtual void [CheckPoop](#) ()
If enemy gets hit by poop, reduce its volume by 0.5 units.
- double [DistanceLeft](#) ()
Returns how much distance the [Enemy](#) has left to travel.

Protected Attributes

- [QTimer](#) * [timer_](#)
- [QList](#)< [QPointF](#) > [path_points_](#)
- [QPointF](#) [dest_](#)
- [QPointF](#) [enemy_center_](#)
- int [damage_](#)
- double [speed_](#)
- int [point_index_](#)
- [Game](#) * [game_](#)
- int [price_](#)
- int [enemy_hp_](#)
- double [distance_traveled_](#)
- double [distance_left_](#)

6.7.1 Detailed Description

[Enemy](#) is an abstract class that all enemies inherit.

Chooses its path on the map according to the virtual [ChoosePath](#) method. Has a virtual void function [Death](#) when enemy has reached its destination. All enemies move with the public slot [MoveForward\(\)](#).

6.7.2 Constructor & Destructor Documentation

6.7.2.1 Enemy()

```
Enemy::Enemy (
    Game * game,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [Enemy](#) object (Abstract class)

Creates a timer for [Enemy](#) that calls the [MoveForward\(\)](#) slot.

Parameters

| | |
|---------------|--|
| <i>game</i> | The Game where the Enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.7.2.2 ~Enemy()

```
Enemy::~Enemy ( )
```

6.7.3 Member Function Documentation

6.7.3.1 CheckPoop()

```
void Enemy::CheckPoop ( ) [virtual]
```

If enemy gets hit by poop, reduce its volume by 0.5 units.

Reimplemented in [Cruiseship](#).

6.7.3.2 ChoosePath()

```
virtual QList< QPointF > Enemy::ChoosePath (
    QList< QList< QPointF > > paths ) [pure virtual]
```

Implemented in [Cruiseship](#), [Dokaani](#), [Fyysikko](#), [Koneteekkari](#), and [Kylteri](#).

6.7.3.3 Death()

```
void Enemy::Death ( ) [virtual]
```

Reimplemented in [Cruiseship](#), and [Dokaani](#).

6.7.3.4 DistanceLeft()

```
double Enemy::DistanceLeft ( )
```

Returns how much distance the [Enemy](#) has left to travel.

Returns

double

6.7.3.5 GetDest()

```
QPointF Enemy::GetDest ( )
```

6.7.3.6 GetSpeed()

```
double Enemy::GetSpeed ( )
```

6.7.3.7 MoveForward

```
void Enemy::MoveForward ( ) [slot]
```

Check if [Enemy](#) gets hit by [Poop](#) or [Bullet](#). Rotate to face the next point and check if it has reached its destination.

6.7.3.8 ReachDest()

```
void Enemy::ReachDest ( )
```

If [Enemy](#) has reached its destination, reduce player health and/or start GameOver.

6.7.3.9 RotateToFacePoint()

```
void Enemy::RotateToFacePoint (
    QPointF p )
```

6.7.4 Member Data Documentation

6.7.4.1 damage_

```
int Enemy::damage_ [protected]
```

6.7.4.2 dest_

```
QPointF Enemy::dest_ [protected]
```

6.7.4.3 distance_left_

```
double Enemy::distance_left_ [protected]
```

6.7.4.4 distance_traveled_

```
double Enemy::distance_traveled_ [protected]
```

6.7.4.5 enemy_center_

```
QPointF Enemy::enemy_center_ [protected]
```

6.7.4.6 enemy_hp_

```
int Enemy::enemy_hp_ [protected]
```

6.7.4.7 game_

```
Game* Enemy::game_ [protected]
```

6.7.4.8 path_points_

```
QList<QPointF> Enemy::path_points_ [protected]
```

6.7.4.9 point_index_

```
int Enemy::point_index_ [protected]
```

6.7.4.10 price_

```
int Enemy::price_ [protected]
```

6.7.4.11 speed_

```
double Enemy::speed_ [protected]
```

6.7.4.12 timer_

```
QTimer* Enemy::timer_ [protected]
```

6.8 Fyysikko Class Reference

[Fyysikko](#) is a smart enemy that chooses the optimized with the shortest length.

```
#include <fyysikko.hpp>
```

Inherits [Enemy](#).

Public Member Functions

- [Fyysikko](#) (QList< QList< QPointF > > paths, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [Fyysikko](#) object.
- QList< QPointF > [ChoosePath](#) (QList< QList< QPointF > > paths)
Chooses the path with shortest length.

Additional Inherited Members

6.8.1 Detailed Description

[Fyysikko](#) is a smart enemy that chooses the optimized with the shortest length.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 Fyysikko()

```
Fyysikko::Fyysikko (
    QList< QList< QPointF > > paths,
    Game * game,
    QGraphicsItem * parent = 0 )
```

Construct a new [Fyysikko](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | List of different paths on the map |
| <i>game</i> | The Game where the enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.8.3 Member Function Documentation

6.8.3.1 ChoosePath()

```
QList< QPointF > Fyysikko::ChoosePath (
    QList< QList< QPointF > > paths ) [virtual]
```

Chooses the path with shortest length.

Parameters

| | |
|--------------|------------------------------------|
| <i>paths</i> | List of different paths on the map |
|--------------|------------------------------------|

Returns

QList<QPointF>

Implements [Enemy](#).

6.9 Game Class Reference

A single game of tower defense.

```
#include <game.hpp>
```

Inherits QGraphicsView.

Public Slots

- void [SpawnEnemy](#) ()
The function that spawns new enemies to the game.
- void [ClearTowers](#) ()
- void [UpgradeTower](#) ()
- void [RemoveTower](#) ()
- void [StartWave](#) ()
The function that starts a new wave.

Public Member Functions

- [Game](#) (QList< QList< QPointF > > paths, QWidget *parent=0)
Construct a [Game](#) object.
- [~Game](#) ()
- void [SetCursor](#) (QString filename)
- void [mouseMoveEvent](#) (QMouseEvent *event)
- void [mousePressEvent](#) (QMouseEvent *event)
This function handles an incoming mousePressEvent.
- bool [IsGameOver](#) ()
- QProgressBar * [GetHealthBar](#) ()
- int [GetMoney](#) ()
- void [SetMoney](#) (int new_money)
- void [SetPriceText](#) (int icon_x, int icon_height, int price)
- void [UpdateMoneyText](#) ()
- void [UpdateWaveText](#) ()
- [Tower](#) * [GetBuild](#) ()
- QList< [Tower](#) * > [GetTowers](#) ()
- void [SetBuild](#) ([Tower](#) *new_build)
- QGraphicsScene * [GetScene](#) ()
- QGraphicsPixmapItem * [GetCursor](#) ()
- void [ResetCursor](#) ()
- QList< QList< QPointF > > [GetPaths](#) ()
- void [CreatePaths](#) ()
Draw a path to the scene.
- void [closeEvent](#) (QCloseEvent *event)
- void [PlayEnemyDiesSfx](#) ()
- void [PlayDokaaniDiesSfx](#) ()
- void [PlayCruiseshipDiesSfx](#) ()
- void [PlayHonkSfx](#) ()
- void [GameOver](#) ()
End the game if we run out of HPs.

6.9.1 Detailed Description

A single game of tower defense.

A single game is responsible for handling the towers and enemies. The path that will be used in the game is given as a parameter to game's constructor.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 Game()

```
Game::Game (
    QList< QList< QPointF > > paths,
    QWidget * parent = 0 ) [explicit]
```

Construct a [Game](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | the path(s) that the enemies can travel |
| <i>parent</i> | the parent will be the menu in this case |

6.9.2.2 ~Game()

```
Game::~Game ( )
```

6.9.3 Member Function Documentation

6.9.3.1 ClearTowers

```
void Game::ClearTowers ( ) [slot]
```

6.9.3.2 closeEvent()

```
void Game::closeEvent (
    QCloseEvent * event )
```


6.9.3.3 CreatePaths()

```
void Game::CreatePaths ( )
```

Draw a path to the scene.

The path is based on the paths_ variable, which gets its value from the games constructor

6.9.3.4 GameOver()

```
void Game::GameOver ( )
```

End the game if we run out of HPs.

6.9.3.5 GetBuild()

```
Tower * Game::GetBuild ( )
```

6.9.3.6 GetCursor()

```
QGraphicsPixmapItem * Game::GetCursor ( )
```

6.9.3.7 GetHealthBar()

```
QProgressBar * Game::GetHealthBar ( )
```

6.9.3.8 GetMoney()

```
int Game::GetMoney ( )
```

6.9.3.9 GetPaths()

```
QList< QList< QPointF > > Game::GetPaths ( )
```

6.9.3.10 GetScene()

```
QGraphicsScene * Game::GetScene ( )
```

6.9.3.11 GetTowers()

```
QList< Tower * > Game::GetTowers ( )
```

6.9.3.12 IsGameOver()

```
bool Game::IsGameOver ( )
```

6.9.3.13 mouseMoveEvent()

```
void Game::mouseMoveEvent (
    QMouseEvent * event )
```

6.9.3.14 mousePressEvent()

```
void Game::mousePressEvent (
    QMouseEvent * event )
```

This function handles an incoming mousePressEvent.

Given that the game is not over, the function will either create a new tower, upgrade an existing tower, or delete an existing tower

Parameters

| | |
|--------------|--|
| <i>event</i> | |
|--------------|--|

6.9.3.15 PlayCruiseshipDiesSfx()

```
void Game::PlayCruiseshipDiesSfx ( )
```

6.9.3.16 PlayDokaaniDiesSfx()

```
void Game::PlayDokaaniDiesSfx ( )
```

6.9.3.17 PlayEnemyDiesSfx()

```
void Game::PlayEnemyDiesSfx ( )
```

6.9.3.18 PlayHonkSfx()

```
void Game::PlayHonkSfx ( )
```

6.9.3.19 RemoveTower

```
void Game::RemoveTower ( ) [slot]
```

6.9.3.20 ResetCursor()

```
void Game::ResetCursor ( )
```

6.9.3.21 SetBuild()

```
void Game::SetBuild (
    Tower * new_build )
```

6.9.3.22 SetCursor()

```
void Game::SetCursor (
    QString filename )
```

6.9.3.23 SetMoney()

```
void Game::SetMoney (
    int new_money )
```

6.9.3.24 SetPriceText()

```
void Game::SetPriceText (
    int icon_x,
    int icon_height,
    int price )
```

6.9.3.25 SpawnEnemy

```
void Game::SpawnEnemy ( ) [slot]
```

The function that spawns new enemies to the game.

Based on the current value of the wave_ variable, different types of enemies will be spawned.

6.9.3.26 StartWave

```
void Game::StartWave ( ) [slot]
```

The function that starts a new wave.

The variable wave_ is incremented at the start of each wave, which results in different, and harder waves.

6.9.3.27 UpdateMoneyText()

```
void Game::UpdateMoneyText ( )
```

6.9.3.28 UpdateWaveText()

```
void Game::UpdateWaveText ( )
```

6.9.3.29 UpgradeTower

```
void Game::UpgradeTower ( ) [slot]
```

6.10 GoldenBullet Class Reference

A [GoldenBullet](#) is a [Bullet](#) that is fast and has high damage.

```
#include <goldenbullet.hpp>
```

Inherits [Bullet](#).

Public Member Functions

- [GoldenBullet](#) (QGraphicsItem *parent=0)
Construct a new Golden Golden [Bullet](#) object.

Additional Inherited Members

6.10.1 Detailed Description

A [GoldenBullet](#) is a [Bullet](#) that is fast and has high damage.

6.10.2 Constructor & Destructor Documentation

6.10.2.1 GoldenBullet()

```
GoldenBullet::GoldenBullet (
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new Golden Golden [Bullet](#) object.

Parameters

| | |
|---------------|---|
| <i>parent</i> | Optional QGraphicsItem pointer, default value = 0 |
|---------------|---|

6.11 Koneteekkari Class Reference

[Koneteekkari](#) is the basic [Enemy](#) that chooses its path at random.

```
#include <koneteekkari.hpp>
```

Inherits [Enemy](#).

Public Member Functions

- [Koneteekkari](#) (QList< QList< QPointF > > paths, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [Koneteekkari](#) object.
- QList< QPointF > [ChoosePath](#) (QList< QList< QPointF > > paths)
Chooses the path at random.

Additional Inherited Members

6.11.1 Detailed Description

[Koneteekkari](#) is the basic [Enemy](#) that chooses its path at random.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 Koneteekkari()

```
Koneteekkari::Koneteekkari (
    QList< QList< QPointF > > paths,
    Game * game,
    QGraphicsItem * parent = 0 )
```

Construct a new [Koneteekkari](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | List of different paths on the map |
| <i>game</i> | The Game where the enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.11.3 Member Function Documentation

6.11.3.1 ChoosePath()

```
QList< QPointF > Koneteekkari::ChoosePath (
    QList< QList< QPointF > > paths ) [virtual]
```

Chooses the path at random.

Parameters

| | |
|--------------|------------------------------------|
| <i>paths</i> | List of different paths on the map |
|--------------|------------------------------------|

Returns

QList<QPointF>

Implements [Enemy](#).

6.12 Kylteri Class Reference

[Kylteri](#) is a quick enemy but always chooses the path with the closest [Tower](#).

```
#include <kylteri.hpp>
```

Inherits [Enemy](#).

Public Member Functions

- [Kylteri](#) (QList< QList< QPointF > > paths, [Game](#) *game, QGraphicsItem *parent=0)
Construct a new [Kylteri](#) object.
- QList< QPointF > [ChoosePath](#) (QList< QList< QPointF > > paths)
Chooses the path which starts closest to a [Tower](#).

Additional Inherited Members

6.12.1 Detailed Description

[Kylteri](#) is a quick enemy but always chooses the path with the closest [Tower](#).

6.12.2 Constructor & Destructor Documentation

6.12.2.1 Kylteri()

```
Kylteri::Kylteri (
    QList< QList< QPointF > > paths,
    Game * game,
    QGraphicsItem * parent = 0 )
```

Construct a new [Kylteri](#) object.

Parameters

| | |
|---------------|--|
| <i>paths</i> | List of different paths on the map |
| <i>game</i> | The Game where the enemy is created in |
| <i>parent</i> | Optional QGraphicsItem pointer parent, default to 0 |

6.12.3 Member Function Documentation

6.12.3.1 ChoosePath()

```
QList< QPointF > Kylteri::ChoosePath (
    QList< QList< QPointF > > paths ) [virtual]
```

Chooses the path which starts closest to a [Tower](#).

Parameters

| | |
|--------------|------------------------------------|
| <i>paths</i> | List of different paths on the map |
|--------------|------------------------------------|

Returns

QList<QPointF>

Implements [Enemy](#).

6.13 MamaGoose Class Reference

A powerful [Tower](#) that shoots [PlasmaBall](#) objects with high frequency.

```
#include <mamagoose.hpp>
```

Inherits [Tower](#).

Public Slots

- void [AcquireTarget](#) ()
Targets the [Enemy](#) closest to its destination (highest priority)

Public Member Functions

- [MamaGoose](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a new [MamaGoose](#) object.
- void [AttackTarget](#) ()
Attacks the current target with a single [PlasmaBall](#).

Additional Inherited Members

6.13.1 Detailed Description

A powerful [Tower](#) that shoots [PlasmaBall](#) objects with high frequency.

6.13.2 Constructor & Destructor Documentation

6.13.2.1 MamaGoose()

```
MamaGoose::MamaGoose (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [MamaGoose](#) object.

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.13.3 Member Function Documentation

6.13.3.1 AcquireTarget

```
void MamaGoose::AcquireTarget ( ) [slot]
```

Targets the [Enemy](#) closest to its destination (highest priority)

6.13.3.2 AttackTarget()

```
void MamaGoose::AttackTarget ( ) [virtual]
```

Attacks the current target with a single [PlasmaBall](#).

Implements [Tower](#).

6.14 Menu Class Reference

[Menu](#) is a class for the main menu of the game.

```
#include <menu.hpp>
```

Inherits QGraphicsView.

Public Slots

- void [StartLevel1](#) ()
- void [StartLevel2](#) ()
- void [StartLevel3](#) ()
- void [StartLevel4](#) ()
- void [StartLevel5](#) ()
- void [StartCustom](#) ()
- void [StartEditor](#) ()

Public Member Functions

- [Menu](#) ()
Construct a [Menu](#) object.
- void [mousePressEvent](#) (QMouseEvent *event)
- QGraphicsScene * [GetScene](#) ()
- QList< QList< QPointF > > [ReadPathsFromFile](#) (const QString &filename)
Convert the path read from a .dat file to a QList<QList< QPointF>> value.
- void [closeEvent](#) (QCloseEvent *event)

6.14.1 Detailed Description

[Menu](#) is a class for the main menu of the game.

The menu is responsible for opening editors and instances of Games. The menu reads the paths for the levels from .dat files

6.14.2 Constructor & Destructor Documentation

6.14.2.1 Menu()

```
Menu::Menu ( )
```

Construct a [Menu](#) object.

6.14.3 Member Function Documentation

6.14.3.1 closeEvent()

```
void Menu::closeEvent (
    QCloseEvent * event )
```

6.14.3.2 GetScene()

```
QGraphicsScene * Menu::GetScene ( )
```

6.14.3.3 mousePressEvent()

```
void Menu::mousePressEvent (
    QMouseEvent * event )
```

6.14.3.4 ReadPathsFromFile()

```
QList< QList< QPointF > > Menu::ReadPathsFromFile (
    const QString & filename )
```

Convert the path read from a .dat file to a QList<QList<QPointF>> value.

Parameters

| | |
|-----------------|--|
| <i>filename</i> | |
|-----------------|--|

Returns

QList<QList<QPointF>> This will be used by the [Game](#) to construct the path

See also

[Editor::SavePathToFile\(\)](#)

6.14.3.5 StartCustom

```
void Menu::StartCustom ( ) [slot]
```

6.14.3.6 StartEditor

```
void Menu::StartEditor ( ) [slot]
```

6.14.3.7 StartLevel1

```
void Menu::StartLevel1 ( ) [slot]
```

6.14.3.8 StartLevel2

```
void Menu::StartLevel2 ( ) [slot]
```

6.14.3.9 StartLevel3

```
void Menu::StartLevel3 ( ) [slot]
```

6.14.3.10 StartLevel4

```
void Menu::StartLevel4 ( ) [slot]
```

6.14.3.11 StartLevel5

```
void Menu::StartLevel5 ( ) [slot]
```

6.15 PlasmaBall Class Reference

A [PlasmaBall](#) is a [Bullet](#) that [MamaGoose](#) uses.

```
#include <plasmaball.hpp>
```

Inherits [Bullet](#).

Public Member Functions

- [PlasmaBall](#) (QGraphicsItem *parent=0)
Construct a new Plasma Ball object.

Additional Inherited Members

6.15.1 Detailed Description

A [PlasmaBall](#) is a [Bullet](#) that [MamaGoose](#) uses.

6.15.2 Constructor & Destructor Documentation

6.15.2.1 PlasmaBall()

```
PlasmaBall::PlasmaBall (
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new Plasma Ball object.

Parameters

| | |
|---------------|---|
| <i>parent</i> | Optional QGraphicsItem pointer, default value = 0 |
|---------------|---|

6.16 Poop Class Reference

A [Poop](#) is a [Bullet](#) that slows down enemies.

```
#include <poop.hpp>
```

Inherits [Bullet](#).

Public Member Functions

- [Poop](#) (QGraphicsItem *parent=0)
Construct a new [Poop](#) object.

Additional Inherited Members

6.16.1 Detailed Description

A [Poop](#) is a [Bullet](#) that slows down enemies.

6.16.2 Constructor & Destructor Documentation

6.16.2.1 Poop()

```
Poop::Poop (
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [Poop](#) object.

Parameters

| | |
|---------------|---|
| <i>parent</i> | Optional QGraphicsItem pointer, default value = 0 |
|---------------|---|

6.17 PooperGoose Class Reference

A close-range [Tower](#) that slows enemies with [Poop](#).

```
#include <poopergoose.hpp>
```

Inherits [Tower](#).

Public Slots

- void [AcquireTarget](#) ()
Targets the closest [Enemy](#) within its attack area.

Public Member Functions

- [PooperGoose](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a new [PooperGoose](#) object.
- void [AttackTarget](#) ()
Attacks the current target with a single [Poop](#) that slows.

Additional Inherited Members

6.17.1 Detailed Description

A close-range [Tower](#) that slows enemies with [Poop](#).

6.17.2 Constructor & Destructor Documentation

6.17.2.1 PooperGoose()

```
PooperGoose::PooperGoose (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [PooperGoose](#) object.

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.17.3 Member Function Documentation

6.17.3.1 AcquireTarget

```
void PooperGoose::AcquireTarget ( ) [slot]
```

Targets the closest [Enemy](#) within its attack area.

6.17.3.2 AttackTarget()

```
void PooperGoose::AttackTarget ( ) [virtual]
```

Attacks the current target with a single [Poop](#) that slows.

Implements [Tower](#).

6.18 ShotgunGoose Class Reference

A close-range [Tower](#) that shoots multiple [Bullet](#) objects.

```
#include <shotgungoose.hpp>
```

Inherits [Tower](#).

Public Slots

- void [AcquireTarget](#) ()
Targets the closest [Enemy](#) within its attack area.

Public Member Functions

- [ShotgunGoose](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a new [ShotgunGoose](#) object.
- void [AttackTarget](#) ()
Attacks the current target with five [Bullet](#) objects.

Additional Inherited Members

6.18.1 Detailed Description

A close-range [Tower](#) that shoots multiple [Bullet](#) objects.

6.18.2 Constructor & Destructor Documentation

6.18.2.1 ShotgunGoose()

```
ShotgunGoose::ShotgunGoose (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [ShotgunGoose](#) object.

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.18.3 Member Function Documentation

6.18.3.1 AcquireTarget

```
void ShotgunGoose::AcquireTarget ( ) [slot]
```

Targets the closest [Enemy](#) within its attack area.

6.18.3.2 AttackTarget()

```
void ShotgunGoose::AttackTarget ( ) [virtual]
```

Attacks the current target with five [Bullet](#) objects.

Implements [Tower](#).

6.19 SniperGoose Class Reference

A long-range [Tower](#) that shoots powerful [GoldenBullet](#) objects.

```
#include <snipergoose.hpp>
```

Inherits [Tower](#).

Public Slots

- virtual void [AcquireTarget](#) ()
Targets the [Enemy](#) closest to its destination (highest priority)

Public Member Functions

- [SniperGoose](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a [SniperGoose](#) object.
- virtual void [AttackTarget](#) ()

Additional Inherited Members

6.19.1 Detailed Description

A long-range [Tower](#) that shoots powerful [GoldenBullet](#) objects.

6.19.2 Constructor & Destructor Documentation

6.19.2.1 SniperGoose()

```
SniperGoose::SniperGoose (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a [SniperGoose](#) object.

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.19.3 Member Function Documentation

6.19.3.1 AcquireTarget

```
void SniperGoose::AcquireTarget ( ) [virtual], [slot]
```

Targets the [Enemy](#) closest to its destination (highest priority)

6.19.3.2 AttackTarget()

```
void SniperGoose::AttackTarget ( ) [virtual]
```

Implements [Tower](#).

6.20 Tower Class Reference

[Tower](#) is an abstract class that all towers inherit.

```
#include <tower.hpp>
```

Inherits QObject, and QGraphicsPixmapItem.

Inherited by [BasicGoose](#), [MamaGoose](#), [PooperGoose](#), [ShotgunGoose](#), and [SniperGoose](#).

Public Slots

- virtual void [AcquireTarget](#) ()=0

Public Member Functions

- [Tower](#) (QGraphicsScene *scene, QGraphicsItem *parent=0)
Construct a new [Tower](#) object (Abstract class).
- [~Tower](#) ()
- unsigned int [GetAttackRadius](#) ()
- void [UpgradeAttackRadius](#) (unsigned int new_radius)
Increases the attack radius of the tower and updates the attack area accordingly.
- double [DistanceTo](#) (QGraphicsItem *item)
- virtual void [AttackTarget](#) ()=0
- int [GetWidth](#) ()
- int [GetHeight](#) ()

Protected Attributes

- QGraphicsScene * [scene_](#)
- QVector< QPointF > [points_](#)
- QPointF [tower_center_](#)
- int [tower_width_](#)
- int [tower_height_](#)
- QGraphicsPolygonItem * [attack_area_](#)
- unsigned int [attack_radius_](#)
- unsigned int [attack_speed_](#)
- QPointF [attack_dest_](#)
- bool [has_target_](#)

6.20.1 Detailed Description

[Tower](#) is an abstract class that all towers inherit.

Towers attack enemies in their attack range according to the pure virtual [AttackTarget\(\)](#). The pure virtual slot [AcquireTarget\(\)](#) is used to choose the [Enemy](#) which is attacked.

6.20.2 Constructor & Destructor Documentation

6.20.2.1 Tower()

```
Tower::Tower (
    QGraphicsScene * scene,
    QGraphicsItem * parent = 0 ) [explicit]
```

Construct a new [Tower](#) object (Abstract class).

Parameters

| | |
|---------------|---|
| <i>scene</i> | the Scene where the Tower is rendered |
| <i>parent</i> | optional QGraphicsItem parent |

6.20.2.2 ~Tower()

```
Tower::~~Tower ( )
```

6.20.3 Member Function Documentation

6.20.3.1 AcquireTarget

```
virtual void Tower::AcquireTarget ( ) [pure virtual], [slot]
```

6.20.3.2 AttackTarget()

```
virtual void Tower::AttackTarget ( ) [pure virtual]
```

Implemented in [BasicGoose](#), [MamaGoose](#), [PooperGoose](#), [ShotgunGoose](#), and [SniperGoose](#).

6.20.3.3 DistanceTo()

```
double Tower::DistanceTo (
    QGraphicsItem * item )
```

6.20.3.4 GetAttackRadius()

```
unsigned int Tower::GetAttackRadius ( )
```

6.20.3.5 GetHeight()

```
int Tower::GetHeight ( )
```

6.20.3.6 GetWidth()

```
int Tower::GetWidth ( )
```

6.20.3.7 UpgradeAttackRadius()

```
void Tower::UpgradeAttackRadius (
    unsigned int new_radius )
```

Increases the attack radius of the tower and updates the attack area accordingly.

Parameters

| | |
|-------------------|--------------------------------------|
| <i>new_radius</i> | the attack radius we want to achieve |
|-------------------|--------------------------------------|

6.20.4 Member Data Documentation

6.20.4.1 attack_area_

QGraphicsPolygonItem* Tower::attack_area_ [protected]

6.20.4.2 attack_dest_

QPointF Tower::attack_dest_ [protected]

6.20.4.3 attack_radius_

unsigned int Tower::attack_radius_ [protected]

6.20.4.4 attack_speed_

unsigned int Tower::attack_speed_ [protected]

6.20.4.5 has_target_

bool Tower::has_target_ [protected]

6.20.4.6 points_

QVector<QPointF> Tower::points_ [protected]

6.20.4.7 scene_

QGraphicsScene* Tower::scene_ [protected]

6.20.4.8 tower_center_

QPointF Tower::tower_center_ [protected]

6.20.4.9 tower_height_

int Tower::tower_height_ [protected]

6.20.4.10 tower_width_

int Tower::tower_width_ [protected]

Chapter 7

File Documentation

7.1 CMakeLists.txt File Reference

7.2 include/basicgoose.hpp File Reference

```
#include "tower.hpp"
```

Classes

- class [BasicGoose](#)
A basic [Tower](#) that shoots basic [Bullet](#) objects.

7.3 basicgoose.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_BASICGOOSE_HPP_
2 #define INCLUDE_BASICGOOSE_HPP_
3
4 #include "tower.hpp"
5
6 class BasicGoose : public Tower {
7     Q_OBJECT
8
9 public:
10     explicit BasicGoose(QGraphicsScene* scene, QGraphicsItem* parent = 0);
11     void AttackTarget();
12     public slots:
13     void AcquireTarget();
14 };
15
16 #endif // INCLUDE_BASICGOOSE_HPP_
```

7.4 include/buildicon.hpp File Reference

```
#include <QGraphicsPixmapItem>
#include <QGraphicsSceneMouseEvent>
#include <QString>
#include "game.hpp"
#include "tower.hpp"
```

Classes

- class [BuildIcon< T >](#)

A template class for building [Tower](#) objects in a [Game](#).

7.5 buildicon.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_BUILDICON_HPP_
2 #define INCLUDE_BUILDICON_HPP_
3
4 #include <QGraphicsPixmapItem>
5 #include <QGraphicsSceneMouseEvent>
6 #include <QString>
7
8 #include "game.hpp"
9 #include "tower.hpp"
10
11 template <class T>
12 class BuildIcon: public QGraphicsPixmapItem {
13 public:
14     BuildIcon(QString imagepath, int price, Game* game, QGraphicsItem* parent = 0);
15     void mousePressEvent(QGraphicsSceneMouseEvent* event);
16 private:
17     Game* game_;
18     QString imagepath_;
19     int price_;
20 };
21
22 template <class T>
23 inline BuildIcon<T>::BuildIcon(QString imagepath, int price, Game* game, QGraphicsItem* parent) :
24     QGraphicsPixmapItem(parent) {
25     imagepath_ = imagepath;
26     QPixmap p = QPixmap(imagepath_);
27     p = p.scaled(100, 100, Qt::KeepAspectRatio);
28     setPixmap(p); // Set size for the goose
29     setOffset(-p.width() / 2, -p.height() / 2); // Centering
30     game_ = game;
31     price_ = price; // Set price
32 }
33
34 template <class T>
35 inline void BuildIcon<T>::mousePressEvent(QGraphicsSceneMouseEvent* event) {
36     if (!(game_>GetBuild()) && game_>GetMoney() >= price_ && !(game_>IsGameOver())) {
37         // Create a new tower to the build_pointer
38         T* newGoose = new T(game_>GetScene());
39         game_>SetBuild(newGoose);
40         game_>SetCursor(QString(imagepath_));
41         // Player buys the tower which takes money
42         game_>SetMoney(game_>GetMoney() - price_);
43         game_>UpdateMoneyText();
44     }
45 }
46
47 #endif // INCLUDE_BUILDICON_HPP_
```

7.6 include/bullet.hpp File Reference

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

Classes

- class [Bullet](#)

A base class for all projectiles/bullets that [Tower](#) class object shoot.

7.7 bullet.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_BULLET_HPP_
2 #define INCLUDE_BULLET_HPP_
3
4 #include <QGraphicsPixmapItem>
5 #include <QObject>
10 class Bullet : public QObject, public QGraphicsPixmapItem {
11     Q_OBJECT // Macro needed to work with signals and slots (and it needs to inherit from QObject)
12
13 public:
14     explicit Bullet(QGraphicsItem* parent = 0);
15     double GetMaxRange();
16     void SetMaxRange(double range);
17     double GetDistanceTraveled();
18     void SetDistanceTraveled(double dist);
19     int GetDamage();
20 public slots:
21     void Move();
22
23 protected:
24     double speed_;
25     double max_range_;
26     double distance_traveled_;
27     int bullet_damage_;
28 };
29
30
31 #endif // INCLUDE_BULLET_HPP_

```

7.8 include/cruiseship.hpp File Reference

```

#include <QList>
#include <QPointF>
#include "game.hpp"
#include "enemy.hpp"

```

Classes

- class [Cruiseship](#)

Cruiseship is a strong enemy which spawns new enemies when it's killed.

7.9 cruiseship.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_CRUISESHIP_HPP_
2 #define INCLUDE_CRUISESHIP_HPP_
3
4 #include <QList>
5 #include <QPointF>
6
7 #include "game.hpp"
8 #include "enemy.hpp"
9
10
15 class Cruiseship : public Enemy {
16     Q_OBJECT
17
18 public:
19     Cruiseship(QList<QList<QPointF>> paths, Game* game, QGraphicsItem* parent = 0);
20     QList<QPointF> ChoosePath(QList<QList<QPointF>> paths);
21     void Death();
22     void CheckPoop();
23 };
24
25 #endif // INCLUDE_CRUISESHIP_HPP_

```

7.10 include/dokaani.hpp File Reference

```
#include "enemy.hpp"
```

Classes

- class [Dokaani](#)

[Dokaani](#) is the head of the enemies which takes a lot of damage but also has nice yield.

7.11 dokaani.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_DOKAANI_HPP_
2 #define INCLUDE_DOKAANI_HPP_
3
4 #include "enemy.hpp"
5
10 class Dokaani : public Enemy {
11     Q_OBJECT
12
13 public:
14     Dokaani(QList<QList<QPointF>> paths, Game* game, QGraphicsItem* parent = 0);
15     QList<QPointF> ChoosePath(QList<QList<QPointF>> paths);
16     void MoveForward();
17     void Death();
18 };
19
20
21
22 #endif // INCLUDE_DOKAANI_HPP_
```

7.12 include/editor.hpp File Reference

```
#include <QGraphicsView>
#include <QGraphicsScene>
#include <QMouseEvent>
#include <QGraphicsItem>
#include <QString>
#include <QList>
#include <QPointF>
#include "editor.hpp"
```

Classes

- class [Editor](#)

A level editor that can be used to create custom levels.

7.13 editor.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_EDITOR_HPP_
2 #define INCLUDE_EDITOR_HPP_
3
4 #include <QGraphicsView>
5 #include <QGraphicsScene>
6 #include <QMouseEvent>
7 #include <QGraphicsItem>
8 #include <QString>
9 #include <QList>
10 #include <QPointF>
11
12 #include "editor.hpp"
13
20 class Editor : public QGraphicsView {
21     Q_OBJECT
22
23 public:
24     // member functions
25     explicit Editor(QWidget* parent = 0);
26     ~Editor();
27
28     void SetCursor(QString filename);
29     void mouseMoveEvent(QMouseEvent* event);
30     void mousePressEvent(QMouseEvent* event);
31
32     void closeEvent(QCloseEvent *event);
33     QGraphicsScene* GetScene();
34     QGraphicsPixmapItem* GetCursor();
35     void ResetCursor();
36     void CreatePath();
37
38 public slots:
39     void SavePathToFile();
40     void TogglePathMode();
41
42 private:
43     QGraphicsScene* scene_;
44     QGraphicsPixmapItem* cursor_ = nullptr;
45     QList<QList<QPointF>> paths_;
46     int path_index_ = -1;
47     bool in_path_mode_ = false;
48 };
49
50
51 #endif // INCLUDE_EDITOR_HPP_

```

7.14 include/enemy.hpp File Reference

```

#include <QGraphicsPixmapItem>
#include <QObject>
#include <QList>
#include <QPointF>
#include <QProgressBar>
#include "game.hpp"

```

Classes

- class [Enemy](#)

[Enemy](#) is an abstract class that all enemies inherit.

7.15 enemy.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_ENEMY_HPP_
2 #define INCLUDE_ENEMY_HPP_
3
4 #include <QGraphicsPixmapItem>
5 #include <QObject>
6 #include <QList>
7 #include <QPointF>
8 #include <QProgressBar>
9
10 #include "game.hpp"
11
12
13 class Enemy : public QObject, public QGraphicsPixmapItem {
14     Q_OBJECT
15
16 public:
17     explicit Enemy(Game* game, QGraphicsItem* parent = 0);
18     ~Enemy();
19     void RotateToFacePoint(QPointF p);
20     QPointF GetDest();
21     virtual QList<QPointF> ChoosePath(QList<QList<QPointF>> paths) = 0;
22     void ReachDest();
23     double GetSpeed();
24     virtual void Death();
25     virtual void CheckPoop();
26     double DistanceLeft();
27
28 public slots:
29     void MoveForward();
30
31 protected:
32     QTimer* timer_;
33     QList<QPointF> path_points_;
34     QPointF dest_;
35     QPointF enemy_center_;
36     int damage_;
37     double speed_;
38     int point_index_;
39     Game* game_;
40     int price_;
41     int enemy_hp_;
42     double distance_traveled_;
43     double distance_left_;
44 };
45
46 #endif // INCLUDE_ENEMY_HPP_

```

7.16 include/fyysikko.hpp File Reference

```
#include "enemy.hpp"
```

Classes

- class [Fyysikko](#)

Fyysikko is a smart enemy that chooses the optimized with the shortest length.

7.17 fyysikko.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_FYYSIKKO_HPP_
2 #define INCLUDE_FYYSIKKO_HPP_
3
4 #include "enemy.hpp"
5

```

```

10 class Fyysikko : public Enemy {
11     Q_OBJECT
12
13 public:
14     Fyysikko(QList<QList<QPointF> paths, Game* game, QGraphicsItem* parent = 0);
15     QList<QPointF> ChoosePath(QList<QList<QPointF> paths);
16 private:
17 };
18
19
20
21 #endif // INCLUDE_FYYSIKKO_HPP_

```

7.18 include/game.hpp File Reference

```

#include <QGraphicsView>
#include <QGraphicsScene>
#include <QMouseEvent>
#include <QGraphicsItem>
#include <QString>
#include <QList>
#include <QPointF>
#include <QTimer>
#include <QLineEdit>
#include <QProgressBar>
#include <QPushButton>
#include <QSoundEffect>
#include "tower.hpp"

```

Classes

- class [Game](#)

A single game of tower defense.

7.19 game.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_GAME_HPP_
2 #define INCLUDE_GAME_HPP_
3
4 #include <QGraphicsView>
5 #include <QGraphicsScene>
6 #include <QMouseEvent>
7 #include <QGraphicsItem>
8 #include <QString>
9 #include <QList>
10 #include <QPointF>
11 #include <QTimer>
12 #include <QLineEdit>
13 #include <QProgressBar>
14 #include <QPushButton>
15 #include <QSoundEffect>
16
17 #include "tower.hpp"
18
26 class Game : public QGraphicsView {
27     Q_OBJECT
28 public:
29     // member functions
30     explicit Game(QList<QList<QPointF> paths, QWidget* parent = 0);
31     ~Game();
32
33     void SetCursor(QString filename);

```

```

34     void mouseMoveEvent(QMouseEvent* event);
35     void mousePressEvent(QMouseEvent* event);
36
37     bool IsGameOver();
38     QProgressBar* GetHealthBar();
39     int GetMoney();
40     void SetMoney(int new_money);
41     void SetPriceText(int icon_x, int icon_height, int price);
42     void UpdateMoneyText();
43     void UpdateWaveText();
44     Tower* GetBuild();
45     QList<Tower*> GetTowers();
46     void SetBuild(Tower* new_build);
47     QGraphicsScene* GetScene();
48     QGraphicsPixmapItem* GetCursor();
49     void ResetCursor();
50
51     QList<QList<QPointF>> GetPaths();
52     void CreatePaths();
53     void closeEvent(QCloseEvent *event);
54
55     void PlayEnemyDiesSfx();
56     void PlayDokaaniDiesSfx();
57     void PlayCruiseshipDiesSfx();
58     void PlayHonkSfx();
59     void GameOver();
60
61 public slots:
62     void SpawnEnemy();
63     void ClearTowers();
64     void UpgradeTower();
65     void RemoveTower();
66     void StartWave();
67
68 private:
69     QGraphicsScene* scene_;
70     Tower* build_;
71     QGraphicsPixmapItem* cursor_;
72     QTimer* enemy_spawn_timer_;
73     int no_of_enemies_;
74     QList<QList<QPointF>> paths_;
75     QList<Tower*> towers_;
76     QProgressBar* health_bar_;
77     int money_;
78     int wave_;
79     QLineEdit* money_text_;
80     QLineEdit* wave_text_;
81     Tower* closest_tower_;
82     QPushButton* upgrade_button_;
83     QPushButton* delete_button_;
84     QPushButton* start_button_;
85     QPushButton* clear_button_;
86     QGraphicsRectItem* selected_tower_rect_;
87     bool game_over_;
88     bool wave_in_progress_;
89
90     QSoundEffect enemy_dies_sfx_;
91     QSoundEffect cruiship_dies_sfx_;
92     QSoundEffect dokaani_dies_sfx_;
93     QSoundEffect honk_sfx_;
94     QSoundEffect chaching_sfx_;
95     QSoundEffect game_over_sfx_;
96 };
97
98
99 #endif // INCLUDE_GAME_HPP_

```

7.20 include/goldenbullet.hpp File Reference

```
#include "bullet.hpp"
```

Classes

- class [GoldenBullet](#)

A [GoldenBullet](#) is a [Bullet](#) that is fast and has high damage.

7.21 goldenbullet.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_GOLDENBULLET_HPP_
2 #define INCLUDE_GOLDENBULLET_HPP_
3
4 #include "bullet.hpp"
5
10 class GoldenBullet : public Bullet {
11 public:
12     explicit GoldenBullet(QGraphicsItem* parent = 0);
13 };
14
15
16 #endif // INCLUDE_GOLDENBULLET_HPP_
```

7.22 include/koneteekkari.hpp File Reference

```
#include "enemy.hpp"
```

Classes

- class [Koneteekkari](#)
Koneteekkari is the basic [Enemy](#) that chooses its path at random.

7.23 koneteekkari.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_KONETEEKKARI_HPP_
2 #define INCLUDE_KONETEEKKARI_HPP_
3
4 #include "enemy.hpp"
5
10 class Koneteekkari : public Enemy {
11     Q_OBJECT
12
13 public:
14     Koneteekkari(QList<QList<QPointF>> paths, Game* game, QGraphicsItem* parent = 0);
15     QList<QPointF> ChoosePath(QList<QList<QPointF>> paths);
16 };
17
18
19
20 #endif // INCLUDE_KONETEEKKARI_HPP_
```

7.24 include/kylteri.hpp File Reference

```
#include "enemy.hpp"
```

Classes

- class [Kylteri](#)
Kylteri is a quick enemy but always chooses the path with the closest [Tower](#).

7.25 kylteri.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_KYLTERI_HPP_
2 #define INCLUDE_KYLTERI_HPP_
3
4 #include "enemy.hpp"
5
10 class Kylteri : public Enemy {
11     Q_OBJECT
12
13 public:
14     Kylteri(QList<QList<QPointF> paths, Game* game, QGraphicsItem* parent = 0);
15     QList<QPointF> ChoosePath(QList<QList<QPointF> paths);
16 };
17
18
19 #endif // INCLUDE_KYLTERI_HPP_
```

7.26 include/mamagoose.hpp File Reference

```
#include "tower.hpp"
```

Classes

- class [MamaGoose](#)

A powerful [Tower](#) that shoots [PlasmaBall](#) objects with high frequency.

7.27 mamagoose.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_MAMAGOOSE_HPP_
2 #define INCLUDE_MAMAGOOSE_HPP_
3
4 #include "tower.hpp"
5
10 class MamaGoose : public Tower {
11     Q_OBJECT
12
13 public:
14     explicit MamaGoose(QGraphicsScene* scene, QGraphicsItem* parent = 0);
15     void AttackTarget();
16 public slots:
17     void AcquireTarget();
18 };
19
20
21
22 #endif // INCLUDE_MAMAGOOSE_HPP_
```

7.28 include/menu.hpp File Reference

```
#include <QGraphicsView>
#include <QGraphicsScene>
#include <QMouseEvent>
#include <QGraphicsItem>
#include <QString>
#include <QList>
```



```
#include <QPointF>
#include <QTimer>
#include <QLineEdit>
#include <QProgressBar>
#include "game.hpp"
#include "editor.hpp"
```

Classes

- class [Menu](#)

[Menu](#) is a class for the main menu of the game.

7.29 menu.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_MENU_HPP_
2 #define INCLUDE_MENU_HPP_
3
4 #include <QGraphicsView>
5 #include <QGraphicsScene>
6 #include <QMouseEvent>
7 #include <QGraphicsItem>
8 #include <QString>
9 #include <QList>
10 #include <QPointF>
11 #include <QTimer>
12 #include <QLineEdit>
13 #include <QProgressBar>
14
15 #include "game.hpp"
16 #include "editor.hpp"
17
24 class Menu : public QGraphicsView {
25     Q_OBJECT
26
27 public:
28     // member functions
29     Menu();
30     void mousePressEvent(QMouseEvent* event);
31     QGraphicsScene* GetScene();
32     QList<QList<QPointF>> ReadPathsFromFile(const QString& filename);
33     void closeEvent(QCloseEvent *event);
34
35 public slots:
36     void StartLevel1();
37     void StartLevel2();
38     void StartLevel3();
39     void StartLevel4();
40     void StartLevel5();
41     void StartCustom();
42     void StartEditor();
43
44 private:
45     QGraphicsScene* scene_;
46     QList<Game*> active_games_;
47     QList<Editor*> active_editors_;
48     QLineEdit* bottom_text_;
49 };
50
51 #endif // INCLUDE_MENU_HPP_
```

7.30 include/plasmaball.hpp File Reference

```
#include "bullet.hpp"
```

Classes

- class [PlasmaBall](#)

A *PlasmaBall* is a *Bullet* that *MamaGoose* uses.

7.31 plasmaball.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_PLASMABALL_HPP_
2 #define INCLUDE_PLASMABALL_HPP_
3
4 #include "bullet.hpp"
5
10 class PlasmaBall : public Bullet {
11 public:
12     explicit PlasmaBall(QGraphicsItem* parent = 0);
13 };
14
15
16 #endif // INCLUDE_PLASMABALL_HPP_
```

7.32 include/poop.hpp File Reference

```
#include "bullet.hpp"
```

Classes

- class [Poop](#)

A *Poop* is a *Bullet* that slows down enemies.

7.33 poop.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_POOP_HPP_
2 #define INCLUDE_POOP_HPP_
3
4 #include "bullet.hpp"
5
10 class Poop : public Bullet {
11 public:
12     explicit Poop(QGraphicsItem* parent = 0);
13 };
14
15 #endif // INCLUDE_POOP_HPP_
```

7.34 include/poopergoose.hpp File Reference

```
#include "tower.hpp"
```

Classes

- class [PooperGoose](#)
A close-range [Tower](#) that slows enemies with [Poop](#).

7.35 poopergoose.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_POOPERGOOSE_HPP_
2 #define INCLUDE_POOPERGOOSE_HPP_
3
4 #include "tower.hpp"
5
10 class PooperGoose : public Tower {
11     Q_OBJECT
12
13 public:
14     explicit PooperGoose(QGraphicsScene* scene, QGraphicsItem* parent = 0);
15     void AttackTarget();
16 public slots:
17     void AcquireTarget();
18 };
19
20
21 #endif // INCLUDE_POOPERGOOSE_HPP_
```

7.36 include/readme.md File Reference

7.37 src/readme.md File Reference

7.38 include/shotgungoose.hpp File Reference

```
#include "tower.hpp"
```

Classes

- class [ShotgunGoose](#)
A close-range [Tower](#) that shoots multiple [Bullet](#) objects.

7.39 shotgungoose.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_SHOTGUNGOOSE_HPP_
2 #define INCLUDE_SHOTGUNGOOSE_HPP_
3
4 #include "tower.hpp"
5
10 class ShotgunGoose : public Tower {
11     Q_OBJECT
12
13 public:
14     explicit ShotgunGoose(QGraphicsScene* scene, QGraphicsItem* parent = 0);
15     void AttackTarget();
16 public slots:
17     void AcquireTarget();
18 };
19
20
21
22 #endif // INCLUDE_SHOTGUNGOOSE_HPP_
```

7.40 include/snipergoose.hpp File Reference

```
#include "tower.hpp"
```

Classes

- class [SniperGoose](#)
A long-range [Tower](#) that shoots powerful [GoldenBullet](#) objects.

7.41 snipergoose.hpp

[Go to the documentation of this file.](#)

```
1 #ifndef INCLUDE_SNIPERGOOSE_HPP_
2 #define INCLUDE_SNIPERGOOSE_HPP_
3
4 #include "tower.hpp"
5
10 class SniperGoose : public Tower {
11     Q_OBJECT
12
13 public:
14     explicit SniperGoose(QGraphicsScene* scene, QGraphicsItem* parent = 0);
15     virtual void AttackTarget();
16 public slots:
17     virtual void AcquireTarget();
18 };
19
20 #endif // INCLUDE_SNIPERGOOSE_HPP_
```

7.42 include/tower.hpp File Reference

```
#include <QGraphicsPixmapItem>
#include <QGraphicsPolygonItem>
#include <QGraphicsItem>
#include <QVector>
#include <QPointF>
#include <QPolygonF>
#include <QObject>
#include <QGraphicsScene>
#include <QGraphicsSceneMouseEvent>
```

Classes

- class [Tower](#)
[Tower](#) is an abstract class that all towers inherit.

7.43 tower.hpp

[Go to the documentation of this file.](#)

```

1 #ifndef INCLUDE_TOWER_HPP_
2 #define INCLUDE_TOWER_HPP_
3
4 #include <QGraphicsPixmapItem>
5 #include <QGraphicsPolygonItem>
6 #include <QGraphicsItem>
7 #include <QVector>
8 #include <QPointF>
9 #include <QPolygonF>
10 #include <QObject>
11 #include <QGraphicsScene>
12 #include <QGraphicsSceneMouseEvent>
13
21 class Tower : public QObject, public QGraphicsPixmapItem {
22     Q_OBJECT
23
24 public:
25     explicit Tower(QGraphicsScene* scene, QGraphicsItem* parent = 0);
26     ~Tower();
27
28     unsigned int GetAttackRadius();
29     void UpgradeAttackRadius(unsigned int new_radius);
30     double DistanceTo(QGraphicsItem* item);
31     virtual void AttackTarget() = 0;
32     int GetWidth();
33     int GetHeight();
34
35 public slots:
36     virtual void AcquireTarget() = 0;
37
38 protected:
39     QGraphicsScene* scene_;
40     QVector<QPointF> points_;
41     QPointF tower_center_;
42     int tower_width_;
43     int tower_height_;
44     QGraphicsPolygonItem* attack_area_;
45     unsigned int attack_radius_;
46     unsigned int attack_speed_;
47     QPointF attack_dest_;
48     bool has_target_;
49 };
50
51 #endif // INCLUDE_TOWER_HPP_

```

7.44 spelling.txt File Reference

7.45 src/basicgoose.cpp File Reference

```

#include "basicgoose.hpp"
#include <QTimer>
#include <QList>
#include "bullet.hpp"
#include "enemy.hpp"

```

7.46 src/bullet.cpp File Reference

```

#include "bullet.hpp"
#include <QPixmap>
#include <QTimer>
#include <qmath.h>

```

7.47 src/cruiseship.cpp File Reference

```
#include "cruiseship.hpp"
#include <qmath.h>
#include "koneteekkari.hpp"
#include "kylteri.hpp"
#include "fyysikko.hpp"
#include "bullet.hpp"
```

7.48 src/dokaani.cpp File Reference

```
#include "dokaani.hpp"
#include <qmath.h>
#include "bullet.hpp"
```

7.49 src/editor.cpp File Reference

```
#include "editor.hpp"
#include <QGraphicsRectItem>
#include <QGraphicsPixmapItem>
#include <QGraphicsLineItem>
#include <QPixmap>
#include <QLineF>
#include <QBrush>
#include <QPen>
#include <QPushButton>
#include <QLineEdit>
#include <QFileDialog>
```

Macros

- `#define WINDOW_WIDTH 1000`
- `#define WINDOW_HEIGHT 700`

7.49.1 Macro Definition Documentation

7.49.1.1 WINDOW_HEIGHT

```
#define WINDOW_HEIGHT 700
```

7.49.1.2 WINDOW_WIDTH

```
#define WINDOW_WIDTH 1000
```

7.50 src/enemy.cpp File Reference

```
#include <enemy.hpp>
#include <QPixmap>
#include <QTimer>
#include <QString>
#include <qmath.h>
#include <bullet.hpp>
#include <poop.hpp>
```

7.51 src/fyysikko.cpp File Reference

```
#include "fyysikko.hpp"
```

7.52 src/game.cpp File Reference

```
#include "game.hpp"
#include <QGraphicsRectItem>
#include <QGraphicsPixmapItem>
#include <QGraphicsLineItem>
#include <QPixmap>
#include <QLineF>
#include <QBrush>
#include <QPen>
#include <QPushButton>
#include <QProgressBar>
#include <QLineEdit>
#include <QPalette>
#include "tower.hpp"
#include "bullet.hpp"
#include "enemy.hpp"
#include "buildicon.hpp"
#include "mamagoose.hpp"
#include "snipergoose.hpp"
#include "basicgoose.hpp"
#include "poopergoose.hpp"
#include "shotgungoose.hpp"
#include "cruiseship.hpp"
#include "fyysikko.hpp"
#include "kylteri.hpp"
#include "koneteekkari.hpp"
#include "dokaani.hpp"
```

Macros

- `#define WINDOW_WIDTH 1000`
- `#define WINDOW_HEIGHT 700`

7.52.1 Macro Definition Documentation

7.52.1.1 WINDOW_HEIGHT

```
#define WINDOW_HEIGHT 700
```

7.52.1.2 WINDOW_WIDTH

```
#define WINDOW_WIDTH 1000
```

7.53 src/goldenbullet.cpp File Reference

```
#include "goldenbullet.hpp"
```

7.54 src/koneteekkari.cpp File Reference

```
#include "koneteekkari.hpp"  
#include <QRandomGenerator>
```

7.55 src/kylteri.cpp File Reference

```
#include "kylteri.hpp"
```

7.56 src/main.cpp File Reference

```
#include <QApplication>  
#include "menu.hpp"
```


Functions

- `int main (int argc, char *argv[])`
The main function of the program.

7.56.1 Function Documentation

7.56.1.1 main()

```
int main (
    int argc,
    char * argv[ ] )
```

The main function of the program.

The function is responsible for creating the main menu of the game.

Parameters

| | |
|-------------|--|
| <i>argc</i> | |
| <i>argv</i> | |

Returns

int

7.57 src/mamagoose.cpp File Reference

```
#include "mamagoose.hpp"
#include <QTimer>
#include <QList>
#include "bullet.hpp"
#include "enemy.hpp"
#include "plasmaball.hpp"
```

7.58 src/menu.cpp File Reference

```
#include "menu.hpp"
#include <QGraphicsRectItem>
#include <QGraphicsPixmapItem>
#include <QGraphicsLineItem>
#include <QPixmap>
#include <QLineF>
#include <QBrush>
```

```
#include <QPen>
#include <QPushButton>
#include <QProgressBar>
#include <QLineEdit>
#include "game.hpp"
#include "editor.hpp"
```

Macros

- `#define WINDOW_WIDTH 400`
- `#define WINDOW_HEIGHT 300`

7.58.1 Macro Definition Documentation

7.58.1.1 WINDOW_HEIGHT

```
#define WINDOW_HEIGHT 300
```

7.58.1.2 WINDOW_WIDTH

```
#define WINDOW_WIDTH 400
```

7.59 src/plasmaball.cpp File Reference

```
#include "plasmaball.hpp"
```

7.60 src/poop.cpp File Reference

```
#include "poop.hpp"
```

7.61 src/poopergoose.cpp File Reference

```
#include "poopergoose.hpp"
#include <QTimer>
#include <QList>
#include "poop.hpp"
#include "enemy.hpp"
```

7.62 src/shotgungoose.cpp File Reference

```
#include "shotgungoose.hpp"  
#include <QTimer>  
#include <QList>  
#include "bullet.hpp"  
#include "enemy.hpp"
```

7.63 src/snipergoose.cpp File Reference

```
#include "snipergoose.hpp"  
#include <QTimer>  
#include <QList>  
#include <qmath.h>  
#include "mamagoose.hpp"  
#include "goldenbullet.hpp"  
#include "enemy.hpp"
```

7.64 src/tower.cpp File Reference

```
#include "tower.hpp"  
#include <QPixmap>  
#include <QLabel>  
#include <QString>  
#include <QTimer>  
#include <QList>  
#include <QPushButton>  
#include <QGraphicsRectItem>  
#include <QBrush>  
#include <QPen>  
#include "bullet.hpp"  
#include "enemy.hpp"
```


Index

- ~Editor
 - Editor, [20](#)
- ~Enemy
 - Enemy, [24](#)
- ~Game
 - Game, [30](#)
- ~Tower
 - Tower, [49](#)
- AcquireTarget
 - BasicGoose, [12](#)
 - MamaGoose, [39](#)
 - PooperGoose, [45](#)
 - ShotgunGoose, [46](#)
 - SniperGoose, [48](#)
 - Tower, [49](#)
- attack_area_
 - Tower, [51](#)
- attack_dest_
 - Tower, [51](#)
- attack_radius_
 - Tower, [51](#)
- attack_speed_
 - Tower, [51](#)
- AttackTarget
 - BasicGoose, [12](#)
 - MamaGoose, [39](#)
 - PooperGoose, [45](#)
 - ShotgunGoose, [46](#)
 - SniperGoose, [48](#)
 - Tower, [50](#)
- BasicGoose, [11](#)
 - AcquireTarget, [12](#)
 - AttackTarget, [12](#)
 - BasicGoose, [11](#)
- BuildIcon
 - BuildIcon< T >, [13](#)
- BuildIcon< T >, [12](#)
 - BuildIcon, [13](#)
 - mousePressEvent, [13](#)
- Bullet, [14](#)
 - Bullet, [14](#)
 - bullet_damage_, [16](#)
 - distance_traveled_, [16](#)
 - GetDamage, [15](#)
 - GetDistanceTraveled, [15](#)
 - GetMaxRange, [15](#)
 - max_range_, [16](#)
 - Move, [15](#)
 - SetDistanceTraveled, [15](#)
 - SetMaxRange, [15](#)
 - speed_, [16](#)
 - bullet_damage_
 - Bullet, [16](#)
- CheckPoop
 - Cruiseship, [17](#)
 - Enemy, [24](#)
- ChoosePath
 - Cruiseship, [17](#)
 - Dokaani, [19](#)
 - Enemy, [24](#)
 - Fyysikko, [28](#)
 - Koneteekkari, [36](#)
 - Kylteri, [38](#)
- ClearTowers
 - Game, [30](#)
- closeEvent
 - Editor, [21](#)
 - Game, [30](#)
 - Menu, [41](#)
- CMakeLists.txt, [53](#)
- CreatePath
 - Editor, [21](#)
- CreatePaths
 - Game, [30](#)
- Cruiseship, [16](#)
 - CheckPoop, [17](#)
 - ChoosePath, [17](#)
 - Cruiseship, [17](#)
 - Death, [18](#)
- damage_
 - Enemy, [26](#)
- Death
 - Cruiseship, [18](#)
 - Dokaani, [19](#)
 - Enemy, [24](#)
- dest_
 - Enemy, [26](#)
- distance_left_
 - Enemy, [26](#)
- distance_traveled_
 - Bullet, [16](#)
 - Enemy, [26](#)
- DistanceLeft
 - Enemy, [24](#)
- DistanceTo
 - Tower, [50](#)

- Dokaani, [18](#)
 - ChoosePath, [19](#)
 - Death, [19](#)
 - Dokaani, [18](#)
 - MoveForward, [19](#)
- Editor, [19](#)
 - ~Editor, [20](#)
 - closeEvent, [21](#)
 - CreatePath, [21](#)
 - Editor, [20](#)
 - GetCursor, [21](#)
 - GetScene, [21](#)
 - mouseMoveEvent, [21](#)
 - mousePressEvent, [21](#)
 - ResetCursor, [21](#)
 - SavePathToFile, [22](#)
 - SetCursor, [22](#)
 - TogglePathMode, [22](#)
- editor.cpp
 - WINDOW_HEIGHT, [68](#)
 - WINDOW_WIDTH, [68](#)
- Enemy, [22](#)
 - ~Enemy, [24](#)
 - CheckPoop, [24](#)
 - ChoosePath, [24](#)
 - damage_, [26](#)
 - Death, [24](#)
 - dest_, [26](#)
 - distance_left_, [26](#)
 - distance_traveled_, [26](#)
 - DistanceLeft, [24](#)
 - Enemy, [23](#)
 - enemy_center_, [26](#)
 - enemy_hp_, [26](#)
 - game_, [26](#)
 - GetDest, [25](#)
 - GetSpeed, [25](#)
 - MoveForward, [25](#)
 - path_points_, [26](#)
 - point_index_, [27](#)
 - price_, [27](#)
 - ReachDest, [25](#)
 - RotateToFacePoint, [25](#)
 - speed_, [27](#)
 - timer_, [27](#)
- enemy_center_
 - Enemy, [26](#)
- enemy_hp_
 - Enemy, [26](#)
- Fyysikko, [27](#)
 - ChoosePath, [28](#)
 - Fyysikko, [28](#)
- Game, [29](#)
 - ~Game, [30](#)
 - ClearTowers, [30](#)
 - closeEvent, [30](#)
 - CreatePaths, [30](#)
 - Game, [30](#)
 - GameOver, [31](#)
 - GetBuild, [31](#)
 - GetCursor, [31](#)
 - GetHealthBar, [31](#)
 - GetMoney, [31](#)
 - GetPaths, [31](#)
 - GetScene, [31](#)
 - GetTowers, [32](#)
 - IsGameOver, [32](#)
 - mouseMoveEvent, [32](#)
 - mousePressEvent, [32](#)
 - PlayCruiseshipDiesSfx, [32](#)
 - PlayDokaaniDiesSfx, [32](#)
 - PlayEnemyDiesSfx, [33](#)
 - PlayHonkSfx, [33](#)
 - RemoveTower, [33](#)
 - ResetCursor, [33](#)
 - SetBuild, [33](#)
 - SetCursor, [33](#)
 - SetMoney, [33](#)
 - SetPriceText, [34](#)
 - SpawnEnemy, [34](#)
 - StartWave, [34](#)
 - UpdateMoneyText, [34](#)
 - UpdateWaveText, [34](#)
 - UpgradeTower, [34](#)
- game.cpp
 - WINDOW_HEIGHT, [70](#)
 - WINDOW_WIDTH, [70](#)
- game_
 - Enemy, [26](#)
- GameOver
 - Game, [31](#)
- GetAttackRadius
 - Tower, [50](#)
- GetBuild
 - Game, [31](#)
- GetCursor
 - Editor, [21](#)
 - Game, [31](#)
- GetDamage
 - Bullet, [15](#)
- GetDest
 - Enemy, [25](#)
- GetDistanceTraveled
 - Bullet, [15](#)
- GetHealthBar
 - Game, [31](#)
- GetHeight
 - Tower, [50](#)
- GetMaxRange
 - Bullet, [15](#)
- GetMoney
 - Game, [31](#)
- GetPaths
 - Game, [31](#)

- GetScene
 - Editor, [21](#)
 - Game, [31](#)
 - Menu, [41](#)
- GetSpeed
 - Enemy, [25](#)
- GetTowers
 - Game, [32](#)
- GetWidth
 - Tower, [50](#)
- GoldenBullet, [35](#)
 - GoldenBullet, [35](#)
- has_target_
 - Tower, [51](#)
- include/basicgoose.hpp, [53](#)
- include/buildicon.hpp, [53](#), [54](#)
- include/bullet.hpp, [54](#), [55](#)
- include/cruiseship.hpp, [55](#)
- include/dokaani.hpp, [56](#)
- include/editor.hpp, [56](#), [57](#)
- include/enemy.hpp, [57](#), [58](#)
- include/fyysikko.hpp, [58](#)
- include/game.hpp, [59](#)
- include/goldenbullet.hpp, [60](#), [61](#)
- include/koneteekkari.hpp, [61](#)
- include/kylteri.hpp, [61](#), [62](#)
- include/mamagoose.hpp, [62](#)
- include/menu.hpp, [62](#), [63](#)
- include/plasmaball.hpp, [63](#), [64](#)
- include/poop.hpp, [64](#)
- include/poopergoose.hpp, [64](#), [65](#)
- include/readme.md, [65](#)
- include/shotgungoose.hpp, [65](#)
- include/snipergoose.hpp, [66](#)
- include/tower.hpp, [66](#), [67](#)
- IsGameOver
 - Game, [32](#)
- Koneteekkari, [35](#)
 - ChoosePath, [36](#)
 - Koneteekkari, [36](#)
- Kylteri, [37](#)
 - ChoosePath, [38](#)
 - Kylteri, [37](#)
- main
 - main.cpp, [71](#)
- main.cpp
 - main, [71](#)
- MamaGoose, [38](#)
 - AcquireTarget, [39](#)
 - AttackTarget, [39](#)
 - MamaGoose, [39](#)
- max_range_
 - Bullet, [16](#)
- Menu, [40](#)
 - closeEvent, [41](#)
 - GetScene, [41](#)
 - Menu, [40](#)
 - mousePressEvent, [41](#)
 - ReadPathsFromFile, [41](#)
 - StartCustom, [41](#)
 - StartEditor, [42](#)
 - StartLevel1, [42](#)
 - StartLevel2, [42](#)
 - StartLevel3, [42](#)
 - StartLevel4, [42](#)
 - StartLevel5, [42](#)
- menu.cpp
 - WINDOW_HEIGHT, [72](#)
 - WINDOW_WIDTH, [72](#)
- mouseMoveEvent
 - Editor, [21](#)
 - Game, [32](#)
- mousePressEvent
 - BuildIcon< T >, [13](#)
 - Editor, [21](#)
 - Game, [32](#)
 - Menu, [41](#)
- Move
 - Bullet, [15](#)
- MoveForward
 - Dokaani, [19](#)
 - Enemy, [25](#)
- path_points_
 - Enemy, [26](#)
- PlasmaBall, [42](#)
 - PlasmaBall, [43](#)
- PlayCruiseshipDiesSfx
 - Game, [32](#)
- PlayDokaaniDiesSfx
 - Game, [32](#)
- PlayEnemyDiesSfx
 - Game, [33](#)
- PlayHonkSfx
 - Game, [33](#)
- point_index_
 - Enemy, [27](#)
- points_
 - Tower, [51](#)
- Poop, [43](#)
 - Poop, [44](#)
- PooperGoose, [44](#)
 - AcquireTarget, [45](#)
 - AttackTarget, [45](#)
 - PooperGoose, [44](#)
- price_
 - Enemy, [27](#)
- ReachDest
 - Enemy, [25](#)
- ReadPathsFromFile
 - Menu, [41](#)
- RemoveTower
 - Game, [33](#)

- ResetCursor
 - Editor, [21](#)
 - Game, [33](#)
- RotateToFacePoint
 - Enemy, [25](#)
- SavePathToFile
 - Editor, [22](#)
- scene_
 - Tower, [51](#)
- SetBuild
 - Game, [33](#)
- SetCursor
 - Editor, [22](#)
 - Game, [33](#)
- SetDistanceTraveled
 - Bullet, [15](#)
- SetMaxRange
 - Bullet, [15](#)
- SetMoney
 - Game, [33](#)
- SetPriceText
 - Game, [34](#)
- ShotgunGoose, [45](#)
 - AcquireTarget, [46](#)
 - AttackTarget, [46](#)
 - ShotgunGoose, [46](#)
- SniperGoose, [47](#)
 - AcquireTarget, [48](#)
 - AttackTarget, [48](#)
 - SniperGoose, [47](#)
- SpawnEnemy
 - Game, [34](#)
- speed_
 - Bullet, [16](#)
 - Enemy, [27](#)
- spelling.txt, [67](#)
- src/basicgoose.cpp, [67](#)
- src/bullet.cpp, [67](#)
- src/cruiseship.cpp, [68](#)
- src/dokaani.cpp, [68](#)
- src/editor.cpp, [68](#)
- src/enemy.cpp, [69](#)
- src/fyysikko.cpp, [69](#)
- src/game.cpp, [69](#)
- src/goldenbullet.cpp, [70](#)
- src/koneteekkari.cpp, [70](#)
- src/kylteri.cpp, [70](#)
- src/main.cpp, [70](#)
- src/mamagoose.cpp, [71](#)
- src/menu.cpp, [71](#)
- src/plasmaball.cpp, [72](#)
- src/poop.cpp, [72](#)
- src/poopergoose.cpp, [72](#)
- src/readme.md, [65](#)
- src/shotgungoose.cpp, [73](#)
- src/snipergoose.cpp, [73](#)
- src/tower.cpp, [73](#)
- StartCustom
 - Menu, [41](#)
- StartEditor
 - Menu, [42](#)
- StartLevel1
 - Menu, [42](#)
- StartLevel2
 - Menu, [42](#)
- StartLevel3
 - Menu, [42](#)
- StartLevel4
 - Menu, [42](#)
- StartLevel5
 - Menu, [42](#)
- StartWave
 - Game, [34](#)
- timer_
 - Enemy, [27](#)
- TogglePathMode
 - Editor, [22](#)
- Tower, [48](#)
 - ~Tower, [49](#)
 - AcquireTarget, [49](#)
 - attack_area_, [51](#)
 - attack_dest_, [51](#)
 - attack_radius_, [51](#)
 - attack_speed_, [51](#)
 - AttackTarget, [50](#)
 - DistanceTo, [50](#)
 - GetAttackRadius, [50](#)
 - GetHeight, [50](#)
 - GetWidth, [50](#)
 - has_target_, [51](#)
 - points_, [51](#)
 - scene_, [51](#)
 - Tower, [49](#)
 - tower_center_, [52](#)
 - tower_height_, [52](#)
 - tower_width_, [52](#)
 - UpgradeAttackRadius, [50](#)
- tower_center_
 - Tower, [52](#)
- tower_height_
 - Tower, [52](#)
- tower_width_
 - Tower, [52](#)
- UpdateMoneyText
 - Game, [34](#)
- UpdateWaveText
 - Game, [34](#)
- UpgradeAttackRadius
 - Tower, [50](#)
- UpgradeTower
 - Game, [34](#)
- WINDOW_HEIGHT
 - editor.cpp, [68](#)
 - game.cpp, [70](#)

 menu.cpp, [72](#)
WINDOW_WIDTH
 editor.cpp, [68](#)
 game.cpp, [70](#)
 menu.cpp, [72](#)