

Computer Graphics Project

AEI Tower

Authors:

Malika Uskembayeva

Adam Gembala

Robert Lotawiec

Field of study:

Informatics sem. VI

Academic year:

2021/2022

1. Description of the project

AEI Tower is a Icy Tower remake inspired by the day-to-day reality of students of the [Faculty of Automatic Control, Electronics And Computer Science](#). The game consists in controlling a character representing a student of the AEI faculty as he/she makes his/her way to the top of the Mage Tower by jumping on successive stairs and advancing to successive floors of the faculty. The character is controlled with the keyboard. It is a platform game set in a tower, where the player's goal is to jump from one "floor" to the next and go as high as possible without falling and plunging off the screen.

2. Task analysis

At the beginning of semester project assumed following features:

- Gameplay divided into levels – each with it's own background and platform texture.
- Randomly generated challenges for player.
- Animations of the player and surroundings.
- Generating platforms should be performed while player progresses through the game.
- Score system – player should collect ECTS points.

In the final version we managed to implement almost all functionalities.

Player levels up after collecting 10 ECTS on each floor. Game form time to time spawns mystery chest for the player, which contains random reward or surprise. Animation of the player has two states – idle and jumping. There is constant amount of the platforms in the game, when platform leaves camera view it is destroyed and new platform is generated. Initial concept of score system was extended by player lifes – during game player collects not only ECTS points but also materials from previous year, that counted as player lifes.

3. External specification

Instruction: [Link](#)

After launching program game displays main menu.



When player clicks start button game begins on the base platform.



Character can be controlled using following keys:

- W – jump
- A – move left
- D – move right

Description of UI:



In order to complete the game player has to jump on platforms. Each platform is worth 1 ECTS. Player can collect ECTS as he first time lands on platform or when platform is destroyed after falling of the screen.



As it was mentioned before player can also collect materials from the previous year.



It is also possible to collect mystery chests:



After opening chest player can encounter random events:

- Finding materials on the internet – Adds one material.
- Professor changed questions – Subtracts one material.
- Lucky exam – Adds 5 ECTS points.
- Empty chest – Does nothing.

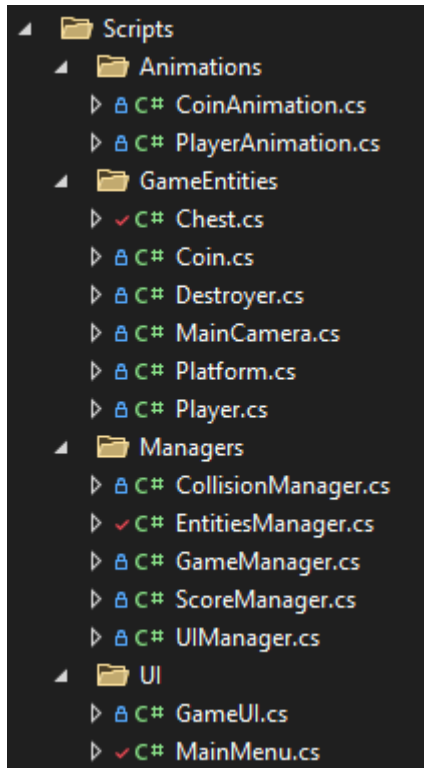
When player wins/looses one of the screens is displayed:



4. Internal Specification

Source code: [GitHub Repository](#)

Scripts structure

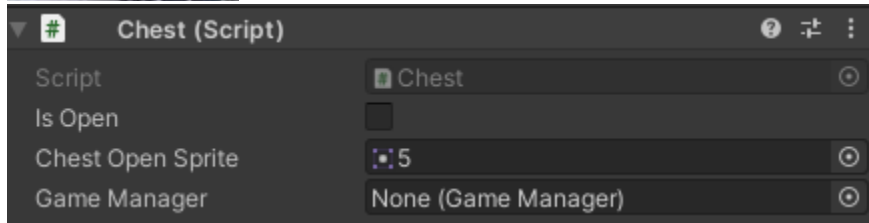


Scripts are divided into categories:

- Animations
Scripts that animate game entities.
- Game entities
Logic of game entities such as movement and collisions.
- Managers
Controllers of certain parts of the game.
- UI
User interface logic.

Game entities

- Chest



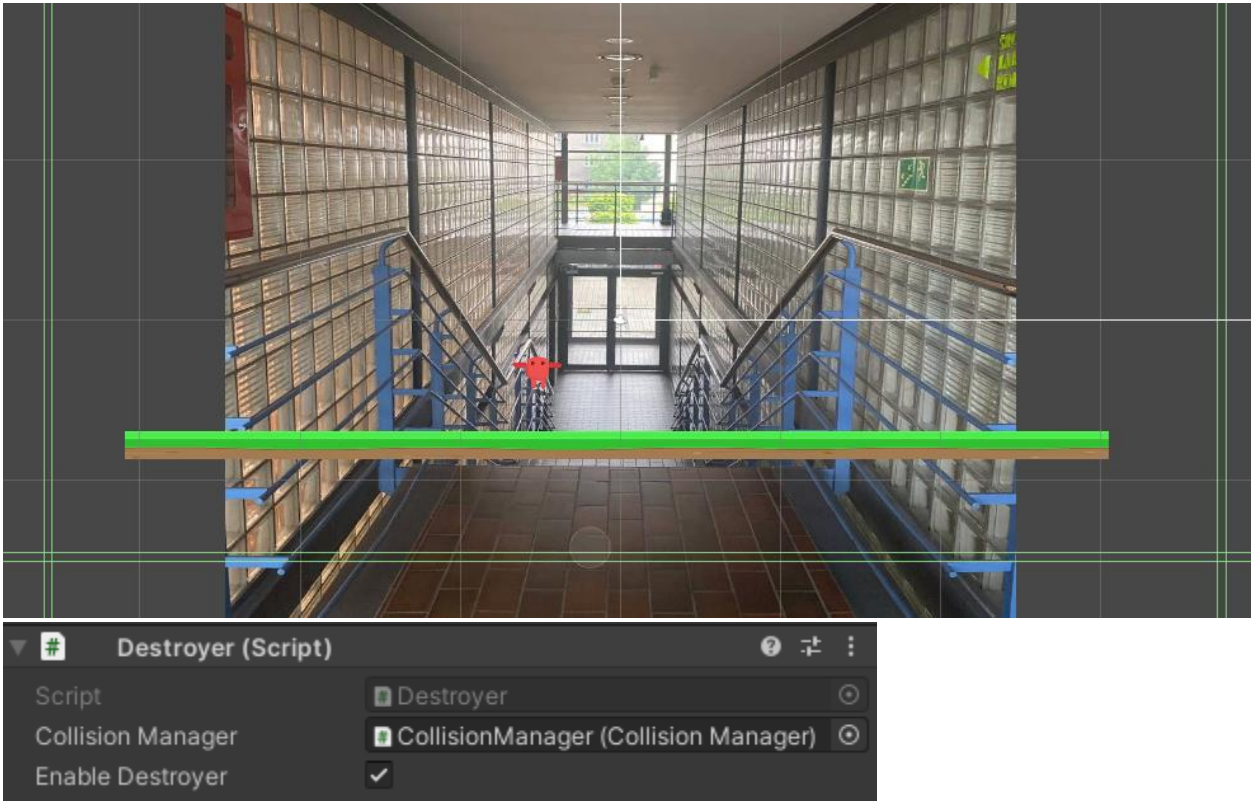
Chest can have two states – opened and closed. When spawned, chest is by default closed, when player collides with chest it opens and randomly generates event as it was described in external specification.

- Coin



Coin represents materials from previous year, which are counted as player lifes. When player collides with coin it calls score manager to add number of points assigned to it and calls entities manager to remove it from the game.

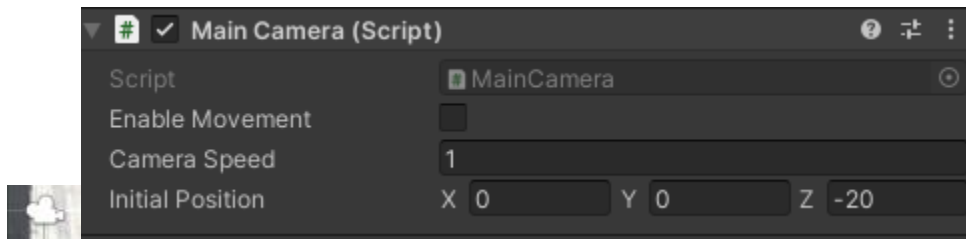
- Destroyer



Destroyer is an empty object, which consists of box collider. It's job is to detect objects that are out of the range of the game area, and as its name suggests destroy them. Destroyer depending on colliding object type can before destroying object take some specific action:

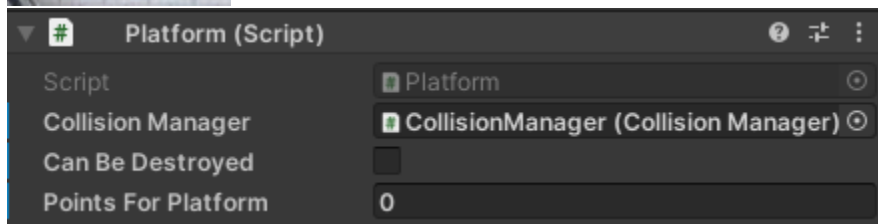
- Player – calls score manager to decrease player's lives (which sometimes end the game).
- Platform – if platform has still assigned points to it, destroyer calls score manager to add this points to overall score and then destroys that platform.
- Coin – Just destroys it.
- Chest – Just destroys it.

- Main camera



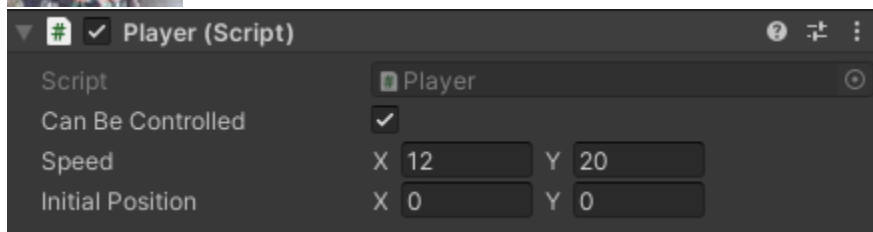
Main camera moves the game view up with certain speed.

- Platform



Platform is an object that detects only collisions that come from above. When player steps on platform it calls score manager to add ECTS point assigned to it.

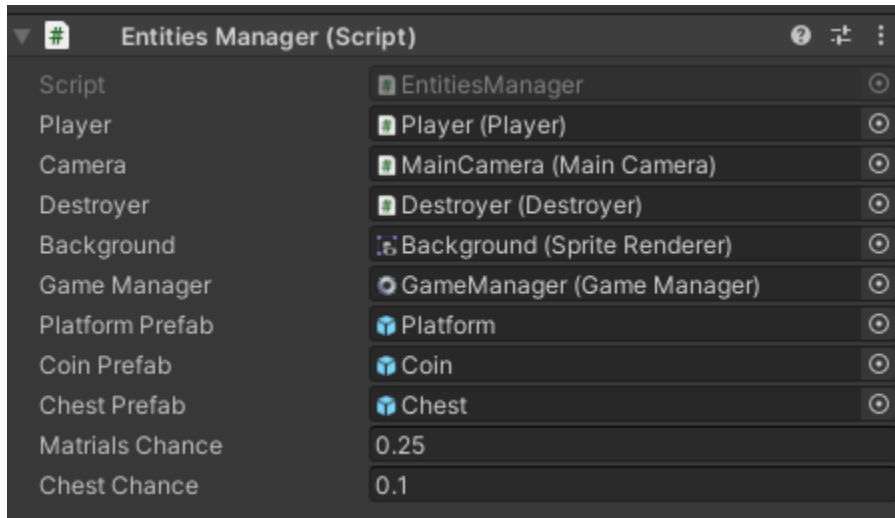
- Player



Player is a main character in the game. It can be controlled in order to progress through the game.

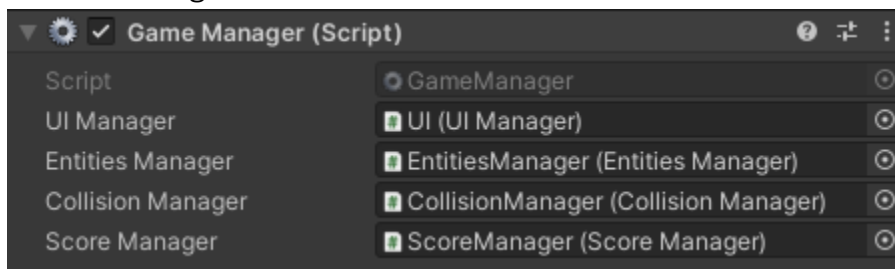
Managers

- Collision manager
Contains logic of collisions between various object types.
- Entities manager



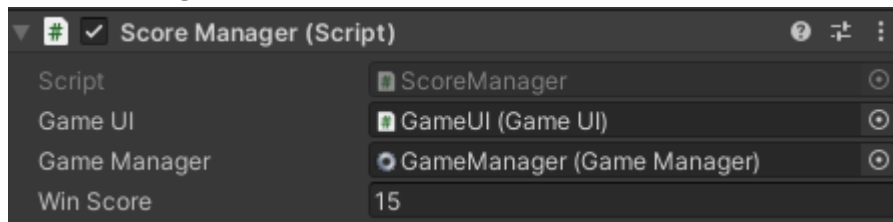
Contains references to various types of game entities and prefabs. Controls lifetime of every object generated during game – can spawn and destroy platforms, chests and coins.

- Game manager



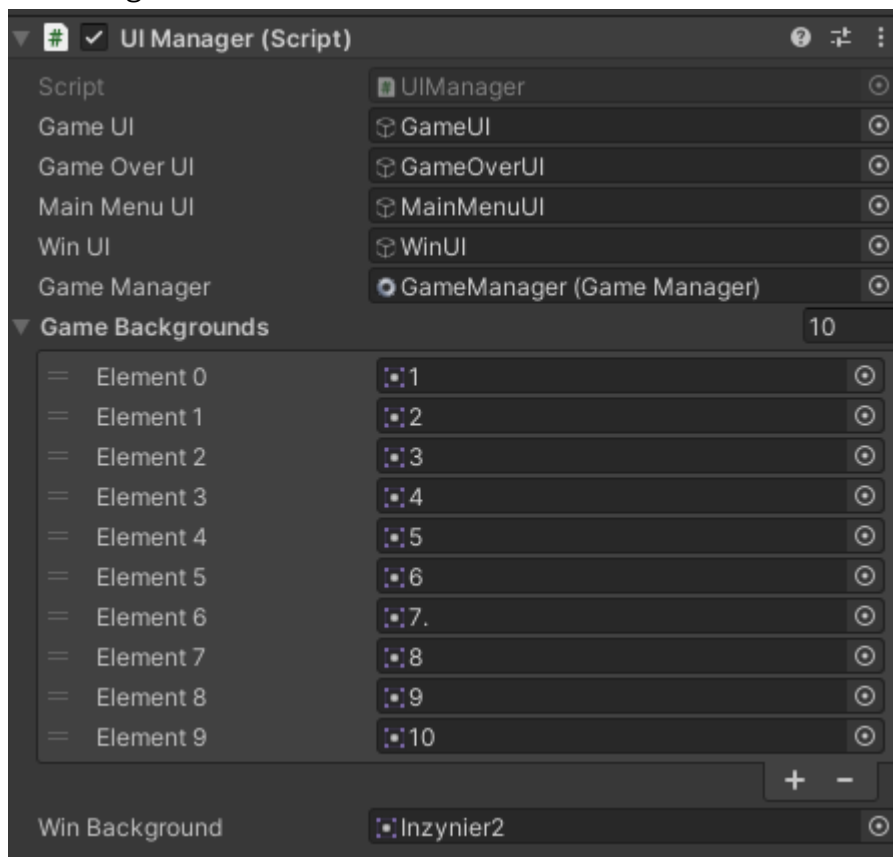
Contains logic of events such as new game, game over or win. Game manager has references to all other managers, therefore can reset the state of the game.

- Score manager



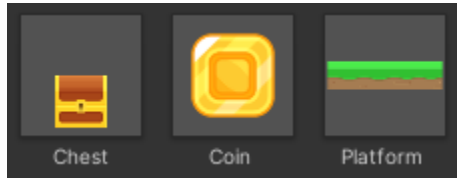
Controls player score and lifes. Score manager can also decide if player completed the game.

- UI manager



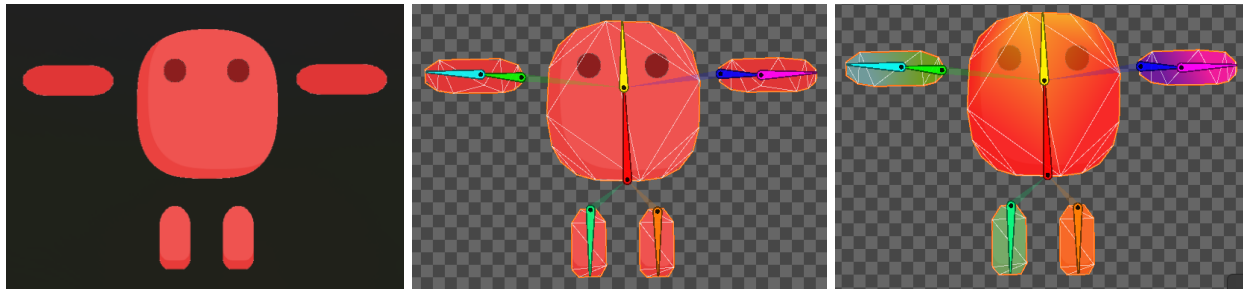
UI manager changes currently active UI depending on game state. It can also control background as player progresses through the game and sets the win background when score manager reports that game was completed. Manager has a queue of messages to be displayed on the screen.

Prefabs



Entities that are generated during game are using prefabs as a templates for new objects.

Player sprite



Player character moves using bone animation. Character sprite was created from free assets available at Unity Hub¹.

¹ <https://assetstore.unity.com/packages/2d/environments/free-platform-game-assets-85838>

5. Testing and running

Game speed bug

During the early development of our game (before beta version) we observed a strange behavior of camera speed depending on the computer game was run on. The mystery was that the problem wasn't in performance of our code – other elements of the game were smooth on every machine. The most insane case was that when one of our team members launched the game on his laptop without power plugged in, the camera was barely moving, but after plugging it to the wall socket camera speed was enormous. Bug was hard to debug, but we managed to solve the problem by multiplying the camera speed by `Time.deltaTime` in `MainCamera` update method. We learned by experience that period of calling update method for game object depends on machine and that some laptops use their GPU only when they are plugged into wall socket.

6. Conclusions

In the final version of the project our team managed to implement almost all functionalities that were assumed at the beginning of the semester. Our team at the beginning of the semester decided to use Jira to track the progress of the project, but because development speed was vastly influenced by the fact that our team was learning Unity engine as the project grew, our team decided in first two weeks that we will resign from this idea. In the end we've learned a lot from development of this project, not only about Unity itself, but also about preparation of the assets and the design of the game.

AEI Tower

Generated by Doxygen 1.9.4

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 Chest Class Reference	7
4.1.1 Detailed Description	7
4.1.2 Member Function Documentation	7
4.1.2.1 Open()	8
4.1.3 Member Data Documentation	8
4.1.3.1 ChestOpenSprite	8
4.1.3.2 GameManager	8
4.1.3.3 IsOpen	8
4.2 Coin Class Reference	8
4.2.1 Detailed Description	9
4.2.2 Member Data Documentation	9
4.2.2.1 CollisionManager	9
4.3 CollisionManager Class Reference	9
4.3.1 Detailed Description	9
4.3.2 Member Function Documentation	10
4.3.2.1 OnChestCollisionWithDestroyer()	10
4.3.2.2 OnCoinCollisionWithDestroyer()	10
4.3.2.3 OnPlatformCollisionWithDestroyer()	10
4.3.2.4 OnPlayerCollisionWithChest()	10
4.3.2.5 OnPlayerCollisionWithCoin()	10
4.3.2.6 OnPlayerCollisionWithDestroyer()	11
4.3.2.7 OnPlayerCollisionWithPlatform()	11
4.3.3 Member Data Documentation	11
4.3.3.1 GameManager	11
4.4 Destroyer Class Reference	11
4.4.1 Detailed Description	12
4.4.2 Member Data Documentation	12
4.4.2.1 CollisionManager	12
4.4.2.2 EnableDestroyer	12
4.5 EntitiesManager Class Reference	12
4.5.1 Detailed Description	13
4.5.2 Member Function Documentation	13
4.5.2.1 RemoveChest()	13

4.5.2.2 RemoveCoin()	13
4.5.2.3 RemovePlatform()	13
4.5.2.4 Reset()	14
4.5.2.5 SpawnPlatform()	14
4.5.3 Member Data Documentation	14
4.5.3.1 Background	14
4.5.3.2 Camera	14
4.5.3.3 ChestChance	14
4.5.3.4 ChestPrefab	14
4.5.3.5 Chests	14
4.5.3.6 CoinPrefab	15
4.5.3.7 Coins	15
4.5.3.8 Destroyer	15
4.5.3.9 GameManager	15
4.5.3.10 MatrialsChance	15
4.5.3.11 PlatformPrefab	15
4.5.3.12 Platforms	15
4.5.3.13 Player	16
4.6 GameManager Class Reference	16
4.6.1 Detailed Description	16
4.6.2 Member Function Documentation	16
4.6.2.1 OnGameOver()	16
4.6.2.2 OnMainMenu()	17
4.6.2.3 OnNewGame()	17
4.6.2.4 OnWin()	17
4.6.3 Member Data Documentation	17
4.6.3.1 CollisionManager	17
4.6.3.2 EntitiesManager	17
4.6.3.3 ScoreManager	17
4.6.3.4 UIManager	17
4.7 GameUI Class Reference	18
4.7.1 Detailed Description	18
4.7.2 Member Function Documentation	18
4.7.2.1 DisplayMaterials()	18
4.7.2.2 DisplayMessage()	18
4.7.2.3 DisplayPoints()	19
4.7.3 Member Data Documentation	19
4.7.3.1 _currentMessageLifeTime	19
4.7.3.2 EctsPointsLabel	19
4.7.3.3 MaterialsLabel	19
4.7.3.4 MessageLifeTimeSeconds	19
4.7.3.5 MessagesLabel	19

4.8 MainCamera Class Reference	20
4.8.1 Detailed Description	20
4.8.2 Member Function Documentation	20
4.8.2.1 Reset()	20
4.8.3 Member Data Documentation	20
4.8.3.1 CameraSpeed	20
4.8.3.2 EnableMovement	21
4.8.3.3 InitialPosition	21
4.9 MainMenu Class Reference	21
4.9.1 Detailed Description	21
4.9.2 Member Function Documentation	21
4.9.2.1 OnExitClick()	22
4.9.2.2 OnMainMenuClick()	22
4.9.2.3 OnNewGameClick()	22
4.9.3 Member Data Documentation	22
4.9.3.1 GameManager	22
4.10 Platform Class Reference	22
4.10.1 Detailed Description	23
4.10.2 Member Data Documentation	23
4.10.2.1 CanBeDestroyed	23
4.10.2.2 CollisionManager	23
4.10.2.3 PointsForPlatform	23
4.11 Player Class Reference	23
4.11.1 Detailed Description	24
4.11.2 Member Function Documentation	24
4.11.2.1 Reset()	24
4.11.2.2 RunForrestRun()	24
4.11.3 Member Data Documentation	24
4.11.3.1 CanBeControlled	24
4.11.3.2 InitialPosition	24
4.11.3.3 Speed	25
4.12 PlayerAnimation Class Reference	25
4.12.1 Member Function Documentation	25
4.12.1.1 SetAnimation()	25
4.13 ScoreManager Class Reference	25
4.13.1 Detailed Description	26
4.13.2 Member Function Documentation	26
4.13.2.1 AddMaterials()	26
4.13.2.2 AddPoints()	26
4.13.2.3 Reset()	26
4.13.3 Member Data Documentation	27
4.13.3.1 GameManager	27

4.13.3.2 GameUI	27
4.13.3.3 WinScore	27
4.13.4 Property Documentation	27
4.13.4.1 Materials	27
4.13.4.2 Points	27
4.14 UIManager Class Reference	28
4.14.1 Detailed Description	28
4.14.2 Member Function Documentation	28
4.14.2.1 DisplayMessage()	28
4.14.2.2 LevelUp()	28
4.14.2.3 Reset()	29
4.14.2.4 SetView()	29
4.14.3 Member Data Documentation	29
4.14.3.1 GameBackgrounds	29
4.14.3.2 GameManager	29
4.14.3.3 GameOverUI	29
4.14.3.4 GameUI	29
4.14.3.5 MainMenuUI	29
4.14.3.6 WinBackground	30
4.14.3.7 WinUI	30
5 File Documentation	31
5.1 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Animations/PlayerAnimation.cs File Reference	31
5.1.1 Enumeration Type Documentation	31
5.1.1.1 EnumPlayerAnimation	31
5.2 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Chest.cs File Reference	31
5.3 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Coin.cs File Reference	32
5.4 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Destroyer.cs File Reference	32
5.5 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/MainCamera.cs File Reference	32
5.6 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Platform.cs File Reference	32
5.7 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Player.cs File Reference	32
5.8 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/CollisionManager.cs File Reference	32
5.9 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/EntitiesManager.cs File Reference	33
5.9.1 Typedef Documentation	33
5.9.1.1 Random	33
5.10 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/GameManager.cs File Reference	33
5.11 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/ScoreManager.cs File Reference	33
5.12 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/UIManager.cs File Reference	33
5.12.1 Enumeration Type Documentation	34
5.12.1.1 EnumUIView	34
5.13 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/GameUI.cs File Reference	34
5.14 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/MainMenu.cs File Reference	34

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

MonoBehaviour	
Chest	7
Coin	8
CollisionManager	9
Destroyer	11
EntitiesManager	12
GameManager	16
GameUI	18
MainCamera	20
MainMenu	21
Platform	22
Player	23
PlayerAnimation	25
ScoreManager	25
UIManager	28

Chapter 2

Class Index

2.1 Class List

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

Chest	7
Coin	8
CollisionManager	9
Destroyer	11
EntitiesManager	12
GameManager	16
GameUI	18
MainCamera	20
MainMenu	21
Platform	22
Player	23
PlayerAnimation	25
ScoreManager	25
UIManager	28

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

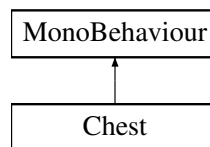
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Animations/PlayerAnimation.cs	31
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Chest.cs	31
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Coin.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Destroyer.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/MainCamera.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Platform.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Player.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/CollisionManager.cs	32
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/EntitiesManager.cs	33
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/GameManager.cs	33
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/ScoreManager.cs	33
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/UIManager.cs	33
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/GameUI.cs	34
D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/MainMenu.cs	34

Chapter 4

Class Documentation

4.1 Chest Class Reference

Inheritance diagram for Chest:



Public Member Functions

- void [Open](#) ()

Public Attributes

- bool [IsOpen](#) = false
- Sprite [ChestOpenSprite](#)
- [GameManager](#) [GameManager](#)

4.1.1 Detailed Description

[Chest](#) class

4.1.2 Member Function Documentation

4.1.2.1 Open()

```
void Chest.Open ( )
```

Opens chest.

4.1.3 Member Data Documentation

4.1.3.1 ChestOpenSprite

```
Sprite Chest.ChestOpenSprite
```

Sprite of opened chest

4.1.3.2 GameManager

```
GameManager Chest.GameManager
```

Game manager reference

4.1.3.3 IsOpen

```
bool Chest.IsOpen = false
```

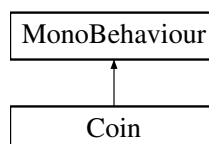
Is chest opened

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/[Chest.cs](#)

4.2 Coin Class Reference

Inheritance diagram for Coin:



Public Attributes

- [CollisionManager](#) [CollisionManager](#)

4.2.1 Detailed Description

[Coin](#) class

4.2.2 Member Data Documentation

4.2.2.1 CollisionManager

[CollisionManager](#) `Coin.CollisionManager`

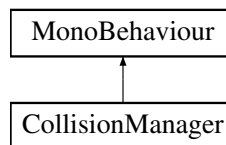
Reference to collision manager

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

- `D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Coin.cs`

4.3 CollisionManager Class Reference

Inheritance diagram for CollisionManager:



Public Member Functions

- void [OnPlayerCollisionWithPlatform](#) ([Platform](#) platform, [Player](#) player, Collision2D collision)
- void [OnPlayerCollisionWithDestroyer](#) ([Destroyer](#) destroyer, [Player](#) player)
- void [OnPlayerCollisionWithCoin](#) ([Coin](#) coin, [Player](#) player)
- void [OnPlayerCollisionWithChest](#) ([Chest](#) chest, [Player](#) player)
- void [OnPlatformCollisionWithDestroyer](#) ([Destroyer](#) destroyer, [Platform](#) platform)
- void [OnCoinCollisionWithDestroyer](#) ([Destroyer](#) destroyer, [Coin](#) coin)
- void [OnChestCollisionWithDestroyer](#) ([Destroyer](#) destroyer, [Chest](#) chest)

Public Attributes

- [GameManager](#) `GameManager`

4.3.1 Detailed Description

Collision manager class

4.3.2 Member Function Documentation

4.3.2.1 OnChestCollisionWithDestroyer()

```
void CollisionManager.OnChestCollisionWithDestroyer (
    Destroyer destroyer,
    Chest chest )
```

[Chest](#) collision with destroyer logic

4.3.2.2 OnCoinCollisionWithDestroyer()

```
void CollisionManager.OnCoinCollisionWithDestroyer (
    Destroyer destroyer,
    Coin coin )
```

[Coin](#) collision with destroyer logic

4.3.2.3 OnPlatformCollisionWithDestroyer()

```
void CollisionManager.OnPlatformCollisionWithDestroyer (
    Destroyer destroyer,
    Platform platform )
```

[Platform](#) collision with destroyer logic

4.3.2.4 OnPlayerCollisionWithChest()

```
void CollisionManager.OnPlayerCollisionWithChest (
    Chest chest,
    Player player )
```

[Player](#) collision with chest logic

4.3.2.5 OnPlayerCollisionWithCoin()

```
void CollisionManager.OnPlayerCollisionWithCoin (
    Coin coin,
    Player player )
```

[Player](#) collision with coin logic

4.3.2.6 OnPlayerCollisionWithDestroyer()

```
void CollisionManager.OnPlayerCollisionWithDestroyer (
    Destroyer destroyer,
    Player player )
```

[Player](#) collision with destroyer logic

4.3.2.7 OnPlayerCollisionWithPlatform()

```
void CollisionManager.OnPlayerCollisionWithPlatform (
    Platform platform,
    Player player,
    Collision2D collision )
```

[Player](#) collision with platform logic

4.3.3 Member Data Documentation

4.3.3.1 GameManager

[GameManager](#) CollisionManager.GameManager

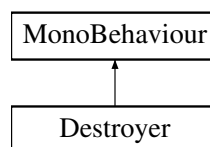
Reference to game manager

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/[CollisionManager.cs](#)

4.4 Destroyer Class Reference

Inheritance diagram for Destroyer:



Public Attributes

- [CollisionManager](#) [CollisionManager](#)
- bool [EnableDestroyer](#) = true

4.4.1 Detailed Description

[Destroyer](#) class

4.4.2 Member Data Documentation

4.4.2.1 CollisionManager

[CollisionManager](#) `Destroyer.CollisionManager`

Reference to collision manager

4.4.2.2 EnableDestroyer

```
bool Destroyer.EnableDestroyer = true
```

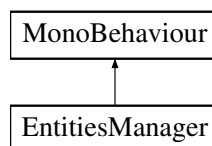
Enables destroyer interactions

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

- `D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Destroyer.cs`

4.5 EntitiesManager Class Reference

Inheritance diagram for EntitiesManager:



Public Member Functions

- void [Reset](#) ()
- void [RemovePlatform](#) (GameObject platform)
- void [RemoveCoin](#) (GameObject coin)
- void [RemoveChest](#) (GameObject chest)
- void [SpawnPlatform](#) ()

Public Attributes

- [Player](#) `Player`
- [MainCamera](#) `Camera`
- [Destroyer](#) `Destroyer`
- `SpriteRenderer` `Background`
- [GameManager](#) `GameManager`
- `ICollection< GameObject >` `Platforms` = `new List<GameObject>()`
- `ICollection< GameObject >` `Coins` = `new List<GameObject>()`
- `ICollection< GameObject >` `Chests` = `new List<GameObject>()`
- `GameObject` `PlatformPrefab`
- `GameObject` `CoinPrefab`
- `GameObject` `ChestPrefab`
- `float` `MatrialsChance` = `0.25f`
- `float` `ChestChance` = `0.10f`

4.5.1 Detailed Description

Entities manager class

4.5.2 Member Function Documentation

4.5.2.1 RemoveChest()

```
void EntityManager.RemoveChest (
    GameObject chest )
```

Remove chest

4.5.2.2 RemoveCoin()

```
void EntityManager.RemoveCoin (
    GameObject coin )
```

Remove coin

4.5.2.3 RemovePlatform()

```
void EntityManager.RemovePlatform (
    GameObject platform )
```

Remove platform

4.5.2.4 Reset()

```
void EntitiesManager.Reset ( )
```

Reset game state

4.5.2.5 SpawnPlatform()

```
void EntitiesManager.SpawnPlatform ( )
```

Spawn platform

4.5.3 Member Data Documentation

4.5.3.1 Background

```
SpriteRenderer EntitiesManager.Background
```

Reference to backgorund

4.5.3.2 Camera

```
MainCamera EntitiesManager.Camera
```

Reference to camera

4.5.3.3 ChestChance

```
float EntitiesManager.ChestChance = 0.10f
```

Chance of spawning chest. Effective range 0-1

4.5.3.4 ChestPrefab

```
GameObject EntitiesManager.ChestPrefab
```

Reference to chest prefab

4.5.3.5 Chests

```
ICollection<GameObject> EntitiesManager.Chests = new List<GameObject>()
```

Collection of generated chests

4.5.3.6 CoinPrefab

```
GameObject EntityManager.CoinPrefab
```

Reference to coin prefab

4.5.3.7 Coins

```
ICollection<GameObject> EntityManager.Coins = new List<GameObject>()
```

Collection of generated coins

4.5.3.8 Destroyer

```
Destroyer EntityManager.Destroyer
```

Reference to destroyer

4.5.3.9 GameManager

```
GameManager EntityManager.GameManager
```

Reference to game manager

4.5.3.10 MatrialsChance

```
float EntityManager.MatrialsChance = 0.25f
```

Chance of spawning materials. Effective range 0-1

4.5.3.11 PlatformPrefab

```
GameObject EntityManager.PlatformPrefab
```

Reference to player prefab

4.5.3.12 Platforms

```
ICollection<GameObject> EntityManager.Platforms = new List<GameObject>()
```

Collection of generated platforms

4.5.3.13 Player

`Player` `EntitiesManager.Player`

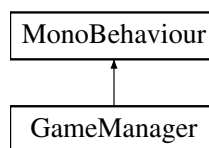
Reference to player

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

- `D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/EntitiesManager.cs`

4.6 GameManager Class Reference

Inheritance diagram for GameManager:



Public Member Functions

- void `OnMainMenu` ()
- void `OnNewGame` ()
- void `OnWin` ()
- void `OnGameOver` ()

Public Attributes

- `UIManager` `UIManager`
- `EntitiesManager` `EntitiesManager`
- `CollisionManager` `CollisionManager`
- `ScoreManager` `ScoreManager`

4.6.1 Detailed Description

Game manager class

4.6.2 Member Function Documentation

4.6.2.1 OnGameOver()

`void GameManager.OnGameOver ()`

`Player` lose logic

4.6.2.2 OnMainMenu()

```
void GameManager.OnMainMenu ( )
```

Open main menu

4.6.2.3 OnNewGame()

```
void GameManager.OnNewGame ( )
```

Start new game

4.6.2.4 OnWin()

```
void GameManager.OnWin ( )
```

[Player](#) victory logic

4.6.3 Member Data Documentation

4.6.3.1 CollisionManager

```
CollisionManager GameManager.CollisionManager
```

Reference to collision manager

4.6.3.2 EntitiesManager

```
EntitiesManager GameManager.EntitiesManager
```

Reference to entities manager

4.6.3.3 ScoreManager

```
ScoreManager GameManager.ScoreManager
```

Reference to score manager

4.6.3.4 UIManager

```
UIManager GameManager.UIManager
```

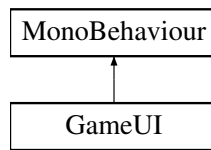
Reference to UI manager

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/[GameManager.cs](#)

4.7 GameUI Class Reference

Inheritance diagram for GameUI:



Public Member Functions

- void [DisplayPoints](#) (int points)
- void [DisplayMaterials](#) (int materials)
- void [DisplayMessage](#) (string message)

Public Attributes

- TMP_Text [EctsPointsLabel](#)
- TMP_Text [MaterialsLabel](#)
- TMP_Text [MessagesLabel](#)
- float [MessageLifeTimeSeconds](#) = 3
- float [_currentMessageLifeTime](#) = 0

4.7.1 Detailed Description

Game UI class

4.7.2 Member Function Documentation

4.7.2.1 DisplayMaterials()

```
void GameUI.DisplayMaterials (  
    int materials )
```

Update materials label

4.7.2.2 DisplayMessage()

```
void GameUI.DisplayMessage (  
    string message )
```

Enqueue message to display

4.7.2.3 DisplayPoints()

```
void GameUI.DisplayPoints (
    int points )
```

Update points label

4.7.3 Member Data Documentation

4.7.3.1 _currentMessageLifeTime

```
float GameUI._currentMessageLifeTime = 0
```

Time of displaying of current message

4.7.3.2 EctsPointsLabel

```
TMP_Text GameUI.EctsPointsLabel
```

Reference to points label

4.7.3.3 MaterialsLabel

```
TMP_Text GameUI.MaterialsLabel
```

Reference to materials label

4.7.3.4 MessageLifeTimeSeconds

```
float GameUI.MessageLifeTimeSeconds = 3
```

Time of displaying of single message

4.7.3.5 MessagesLabel

```
TMP_Text GameUI.MessagesLabel
```

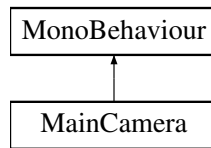
Reference to messages label

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/UI/[GameUI.cs](#)

4.8 MainCamera Class Reference

Inheritance diagram for MainCamera:



Public Member Functions

- void [Reset](#) ()

Public Attributes

- bool [EnableMovement](#) = false
- float [CameraSpeed](#) = 0.01f
- Vector3 [InitialPosition](#) = new Vector3(0f, 0f, -20f)

4.8.1 Detailed Description

Main camera class

4.8.2 Member Function Documentation

4.8.2.1 Reset()

```
void MainCamera.Reset ( )
```

4.8.3 Member Data Documentation

4.8.3.1 CameraSpeed

```
float MainCamera.CameraSpeed = 0.01f
```

Speed of the camera

4.8.3.2 EnableMovement

```
bool MainCamera.EnableMovement = false
```

Enables movement of the camera

4.8.3.3 InitialPosition

```
Vector3 MainCamera.InitialPosition = new Vector3(0f, 0f, -20f)
```

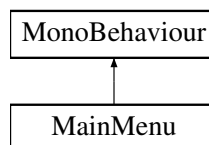
Initial position of the camera

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/[MainCamera.cs](#)

4.9 MainMenu Class Reference

Inheritance diagram for MainMenu:



Public Member Functions

- void [OnNewGameClick](#) ()
- void [OnMainMenuClick](#) ()
- void [OnExitClick](#) ()

Public Attributes

- [GameManager](#) [GameManager](#)

4.9.1 Detailed Description

Main menu class

4.9.2 Member Function Documentation

4.9.2.1 OnExitClick()

```
void MainMenu.OnExitClick ( )
```

When exit button was pressed

4.9.2.2 OnMainMenuClick()

```
void MainMenu.OnMainMenuClick ( )
```

When main menu button was pressed

4.9.2.3 OnNewGameClick()

```
void MainMenu.OnNewGameClick ( )
```

When new game button was pressed

4.9.3 Member Data Documentation

4.9.3.1 GameManager

```
GameManager MainMenu.GameManager
```

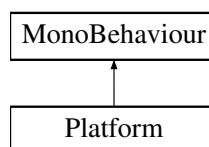
Reference to game manager

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

- D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/UI/[MainMenu.cs](#)

4.10 Platform Class Reference

Inheritance diagram for Platform:



Public Attributes

- [CollisionManager](#) CollisionManager
- bool [CanBeDestroyed](#) = true
- int [PointsForPlatform](#) = 1

4.10.1 Detailed Description

[Platform](#) class

4.10.2 Member Data Documentation

4.10.2.1 CanBeDestroyed

```
bool Platform.CanBeDestroyed = true
```

Can platform be destroyed

4.10.2.2 CollisionManager

```
CollisionManager Platform.CollisionManager
```

Reference to collision manager

4.10.2.3 PointsForPlatform

```
int Platform.PointsForPlatform = 1
```

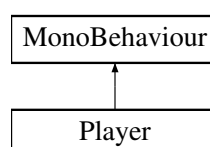
Points assigned to platform

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

- D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/[Platform.cs](#)

4.11 Player Class Reference

Inheritance diagram for Player:



Public Member Functions

- void [RunForrestRun](#) ()
- void [Reset](#) ()

Public Attributes

- bool `CanBeControlled` = true
- Vector2 `Speed` = new Vector2(15, 35)
- Vector2 `InitialPosition` = new Vector2(4.1638f, -1.91f)

4.11.1 Detailed Description

`Player` class

4.11.2 Member Function Documentation

4.11.2.1 Reset()

```
void Player.Reset ( )
```

Reset player

4.11.2.2 RunForrestRun()

```
void Player.RunForrestRun ( )
```

Bounce player up

4.11.3 Member Data Documentation

4.11.3.1 CanBeControlled

```
bool Player.CanBeControlled = true
```

Enables movement logic

4.11.3.2 InitialPosition

```
Vector2 Player.InitialPosition = new Vector2(4.1638f, -1.91f)
```

Initial position of the player

4.11.3.3 Speed

```
Vector2 Player.Speed = new Vector2(15, 35)
```

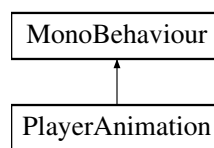
Speed of the player

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

- D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/[Player.cs](#)

4.12 PlayerAnimation Class Reference

Inheritance diagram for PlayerAnimation:



Public Member Functions

- void [SetAnimation](#) ([EnumPlayerAnimation](#) animation)

4.12.1 Member Function Documentation

4.12.1.1 SetAnimation()

```
void PlayerAnimation.SetAnimation (
    EnumPlayerAnimation animation )
```

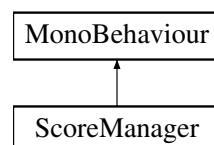
Sets current animation of the player.

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

- D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Animations/[PlayerAnimation.cs](#)

4.13 ScoreManager Class Reference

Inheritance diagram for ScoreManager:



Public Member Functions

- void [AddPoints](#) (int points)
- void [AddMaterials](#) (int materials)
- void [Reset](#) ()

Public Attributes

- [GameUI](#) [GameUI](#)
- [GameManager](#) [GameManager](#)
- int [WinScore](#) = 20

Properties

- int [Points](#) [get]
- int [Materials](#) [get]

4.13.1 Detailed Description

Score manager class

4.13.2 Member Function Documentation

4.13.2.1 AddMaterials()

```
void ScoreManager.AddMaterials (  
    int materials )
```

Increase player materials

4.13.2.2 AddPoints()

```
void ScoreManager.AddPoints (  
    int points )
```

Increase player score

4.13.2.3 Reset()

```
void ScoreManager.Reset ( )
```

Reset state

4.13.3 Member Data Documentation

4.13.3.1 GameManager

[GameManager](#) `ScoreManager.GameManager`

Reference to game manager

4.13.3.2 GameUI

[GameUI](#) `ScoreManager.GameUI`

Reference to UI

4.13.3.3 WinScore

`int ScoreManager.WinScore = 20`

Points required to win the game

4.13.4 Property Documentation

4.13.4.1 Materials

`int ScoreManager.Materials [get]`

Getter of player materials

4.13.4.2 Points

`int ScoreManager.Points [get]`

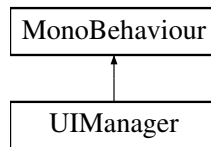
Getter of player points

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

- `D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/ScoreManager.cs`

4.14 UIManager Class Reference

Inheritance diagram for UIManager:



Public Member Functions

- void [Reset](#) ()
- void [LevelUp](#) ()
- void [SetView](#) ([EnumUIView](#) uiView)
- void [DisplayMessage](#) (string message)

Public Attributes

- GameObject [GameUI](#)
- GameObject [GameOverUI](#)
- GameObject [MainMenuUI](#)
- GameObject [WinUI](#)
- [GameManager](#) [GameManager](#)
- List< Sprite > [GameBackgrounds](#) = new List<Sprite>()
- Sprite [WinBackground](#)

4.14.1 Detailed Description

UI manager class

4.14.2 Member Function Documentation

4.14.2.1 DisplayMessage()

```
void UIManager.DisplayMessage (  
    string message )
```

Display message

4.14.2.2 LevelUp()

```
void UIManager.LevelUp ( )
```

Update UI on level up

4.14.2.3 Reset()

```
void UIManager.Reset ( )
```

Reset state

4.14.2.4 SetView()

```
void UIManager.SetView (
    EnumUIView uiView )
```

Change game view

4.14.3 Member Data Documentation

4.14.3.1 GameBackgrounds

```
List<Sprite> UIManager.GameBackgrounds = new List<Sprite>()
```

Collection of game backgrounds

4.14.3.2 GameManager

```
GameManager UIManager.GameManager
```

Reference to game manager

4.14.3.3 GameOverUI

```
GameObject UIManager.GameOverUI
```

Reference to game over view

4.14.3.4 GameUI

```
GameObject UIManager.GameUI
```

Reference to UI

4.14.3.5 MainMenuUI

```
GameObject UIManager.MainMenuUI
```

Reference to main menu view

4.14.3.6 WinBackground

`Sprite UIManager.WinBackground`

Reference to victory background

4.14.3.7 WinUI

`GameObject UIManager.WinUI`

Reference to win view

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

- `D:/Projects/Polis/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/`[UIManager.cs](#)

Chapter 5

File Documentation

5.1 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Animations/↔ PlayerAnimation.cs File Reference

Classes

- class [PlayerAnimation](#)

Enumerations

- enum [EnumPlayerAnimation](#) { [Idle](#) , [Jump](#) }

5.1.1 Enumeration Type Documentation

5.1.1.1 EnumPlayerAnimation

enum [EnumPlayerAnimation](#)

Enumerator

Idle	
Jump	

5.2 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↔ Entities/Chest.cs File Reference

Classes

- class [Chest](#)

5.3 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↵ Entities/Coin.cs File Reference

Classes

- class [Coin](#)

5.4 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↵ Entities/Destroyer.cs File Reference

Classes

- class [Destroyer](#)

5.5 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↵ Entities/MainCamera.cs File Reference

Classes

- class [MainCamera](#)

5.6 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↵ Entities/Platform.cs File Reference

Classes

- class [Platform](#)

5.7 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Game↵ Entities/Player.cs File Reference

Classes

- class [Player](#)

5.8 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/↵ CollisionManager.cs File Reference

Classes

- class [CollisionManager](#)

5.9 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/EntitiesManager.cs File Reference ↵

Classes

- class [EntitiesManager](#)

Typedefs

- using [Random](#) = UnityEngine.Random

5.9.1 Typedef Documentation

5.9.1.1 Random

```
using Random = UnityEngine.Random
```

5.10 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/GameManager.cs File Reference ↵

Classes

- class [GameManager](#)

5.11 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/ScoreManager.cs File Reference ↵

Classes

- class [ScoreManager](#)

5.12 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/UIManager.cs File Reference ↵

Classes

- class [UIManager](#)

Enumerations

- enum [EnumUIView](#) { [Game](#) , [GameOver](#) , [MainMenu](#) , [Win](#) }

5.12.1 Enumeration Type Documentation

5.12.1.1 EnumUIView

enum [EnumUIView](#)

Enum of game views

Enumerator

Game	
GameOver	
MainMenu	
Win	

5.13 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/GameUI.cs File Reference

Classes

- class [GameUI](#)

5.14 D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/MainMenu.cs File Reference

Classes

- class [MainMenu](#)

Index

`_currentMessageLifeTime`
 GameUI, [19](#)

AddMaterials
 ScoreManager, [26](#)
AddPoints
 ScoreManager, [26](#)

Background
 EntitiesManager, [14](#)

Camera
 EntitiesManager, [14](#)
CameraSpeed
 MainCamera, [20](#)

CanBeControlled
 Player, [24](#)

CanBeDestroyed
 Platform, [23](#)

Chest, [7](#)
 ChestOpenSprite, [8](#)
 GameManager, [8](#)
 IsOpen, [8](#)
 Open, [7](#)

ChestChance
 EntitiesManager, [14](#)

ChestOpenSprite
 Chest, [8](#)

ChestPrefab
 EntitiesManager, [14](#)

Chests
 EntitiesManager, [14](#)

Coin, [8](#)
 CollisionManager, [9](#)

CoinPrefab
 EntitiesManager, [14](#)

Coins
 EntitiesManager, [15](#)

CollisionManager, [9](#)
 Coin, [9](#)
 Destroyer, [12](#)
 GameManager, [11](#), [17](#)
 OnChestCollisionWithDestroyer, [10](#)
 OnCoinCollisionWithDestroyer, [10](#)
 OnPlatformCollisionWithDestroyer, [10](#)
 OnPlayerCollisionWithChest, [10](#)
 OnPlayerCollisionWithCoin, [10](#)
 OnPlayerCollisionWithDestroyer, [10](#)
 OnPlayerCollisionWithPlatform, [11](#)
 Platform, [23](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Animations/PlayerA
 [31](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Chest
 [31](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Coin.
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Destr
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/MainC
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Platfo
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/GameEntities/Playe
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/CollisionM
 [32](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/EntitiesM
 [33](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/GameMa
 [33](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/ScoreMa
 [33](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/Managers/UIManag
 [33](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/GameUI.cs,
 [34](#)

D:/Projects/Polsl/AEI-Tower/AEI-Tower/Assets/Scripts/UI/MainMenu.cs,
 [34](#)

Destroyer, [11](#)
 CollisionManager, [12](#)
 EnableDestroyer, [12](#)
 EntitiesManager, [15](#)

DisplayMaterials
 GameUI, [18](#)

DisplayMessage
 GameUI, [18](#)
 UIManager, [28](#)

DisplayPoints
 GameUI, [18](#)

EctsPointsLabel
 GameUI, [19](#)

EnableDestroyer
 Destroyer, [12](#)

EnableMovement
 MainCamera, [20](#)

EntitiesManager, [12](#)
 Background, [14](#)
 Camera, [14](#)
 ChestChance, [14](#)

- ChestPrefab, 14
- Chests, 14
- CoinPrefab, 14
- Coins, 15
- Destroyer, 15
- GameManager, 15, 17
- MatrialsChance, 15
- PlatformPrefab, 15
- Platforms, 15
- Player, 15
- RemoveChest, 13
- RemoveCoin, 13
- RemovePlatform, 13
- Reset, 13
- SpawnPlatform, 14
- EntitiesManager.cs
 - Random, 33
- EnumPlayerAnimation
 - PlayerAnimation.cs, 31
- EnumUIView
 - UIManager.cs, 34
- Game
 - UIManager.cs, 34
- GameBackgrounds
 - UIManager, 29
- GameManager, 16
 - Chest, 8
 - CollisionManager, 11, 17
 - EntitiesManager, 15, 17
 - MainMenu, 22
 - OnGameOver, 16
 - OnMainMenu, 16
 - OnNewGame, 17
 - OnWin, 17
 - ScoreManager, 17, 27
 - UIManager, 17, 29
- GameOver
 - UIManager.cs, 34
- GameOverUI
 - UIManager, 29
- GameUI, 18
 - _currentMessageLifeTime, 19
 - DisplayMaterials, 18
 - DisplayMessage, 18
 - DisplayPoints, 18
 - EctsPointsLabel, 19
 - MaterialsLabel, 19
 - MessageLifeTimeSeconds, 19
 - MessagesLabel, 19
 - ScoreManager, 27
 - UIManager, 29
- Idle
 - PlayerAnimation.cs, 31
- InitialPosition
 - MainCamera, 21
 - Player, 24
- IsOpen
 - Chest, 8
- Jump
 - PlayerAnimation.cs, 31
- LevelUp
 - UIManager, 28
- MainCamera, 20
 - CameraSpeed, 20
 - EnableMovement, 20
 - InitialPosition, 21
 - Reset, 20
- MainMenu, 21
 - GameManager, 22
 - OnExitClick, 21
 - OnMainMenuClick, 22
 - OnNewGameClick, 22
 - UIManager.cs, 34
- MainMenuUI
 - UIManager, 29
- Materials
 - ScoreManager, 27
- MaterialsLabel
 - GameUI, 19
- MatrialsChance
 - EntitiesManager, 15
- MessageLifeTimeSeconds
 - GameUI, 19
- MessagesLabel
 - GameUI, 19
- OnChestCollisionWithDestroyer
 - CollisionManager, 10
- OnCoinCollisionWithDestroyer
 - CollisionManager, 10
- OnExitClick
 - MainMenu, 21
- OnGameOver
 - GameManager, 16
- OnMainMenu
 - GameManager, 16
- OnMainMenuClick
 - MainMenu, 22
- OnNewGame
 - GameManager, 17
- OnNewGameClick
 - MainMenu, 22
- OnPlatformCollisionWithDestroyer
 - CollisionManager, 10
- OnPlayerCollisionWithChest
 - CollisionManager, 10
- OnPlayerCollisionWithCoin
 - CollisionManager, 10
- OnPlayerCollisionWithDestroyer
 - CollisionManager, 10
- OnPlayerCollisionWithPlatform
 - CollisionManager, 11
- OnWin

- GameManager, [17](#)
- Open
 - Chest, [7](#)
- Platform, [22](#)
 - CanBeDestroyed, [23](#)
 - CollisionManager, [23](#)
 - PointsForPlatform, [23](#)
- PlatformPrefab
 - EntitiesManager, [15](#)
- Platforms
 - EntitiesManager, [15](#)
- Player, [23](#)
 - CanBeControlled, [24](#)
 - EntitiesManager, [15](#)
 - InitialPosition, [24](#)
 - Reset, [24](#)
 - RunForrestRun, [24](#)
 - Speed, [24](#)
- PlayerAnimation, [25](#)
 - SetAnimation, [25](#)
- PlayerAnimation.cs
 - EnumPlayerAnimation, [31](#)
 - Idle, [31](#)
 - Jump, [31](#)
- Points
 - ScoreManager, [27](#)
- PointsForPlatform
 - Platform, [23](#)
- Random
 - EntitiesManager.cs, [33](#)
- RemoveChest
 - EntitiesManager, [13](#)
- RemoveCoin
 - EntitiesManager, [13](#)
- RemovePlatform
 - EntitiesManager, [13](#)
- Reset
 - EntitiesManager, [13](#)
 - MainCamera, [20](#)
 - Player, [24](#)
 - ScoreManager, [26](#)
 - UIManager, [28](#)
- RunForrestRun
 - Player, [24](#)
- ScoreManager, [25](#)
 - AddMaterials, [26](#)
 - AddPoints, [26](#)
 - GameManager, [17](#), [27](#)
 - GameUI, [27](#)
 - Materials, [27](#)
 - Points, [27](#)
 - Reset, [26](#)
 - WinScore, [27](#)
- SetAnimation
 - PlayerAnimation, [25](#)
- SetView
 - UIManager, [29](#)
- SpawnPlatform
 - EntitiesManager, [14](#)
- Speed
 - Player, [24](#)
- UIManager, [28](#)
 - DisplayMessage, [28](#)
 - GameBackgrounds, [29](#)
 - GameManager, [17](#), [29](#)
 - GameOverUI, [29](#)
 - GameUI, [29](#)
 - LevelUp, [28](#)
 - MainMenuUI, [29](#)
 - Reset, [28](#)
 - SetView, [29](#)
 - WinBackground, [29](#)
 - WinUI, [30](#)
- UIManager.cs
 - EnumUIView, [34](#)
 - Game, [34](#)
 - GameOver, [34](#)
 - MainMenu, [34](#)
 - Win, [34](#)
- Win
 - UIManager.cs, [34](#)
- WinBackground
 - UIManager, [29](#)
- WinScore
 - ScoreManager, [27](#)
- WinUI
 - UIManager, [30](#)