High Scores

High Scores

Approach

From the Main Menu we'll implement a button to display 'High Scores'. As before, the HighScoreScene will be loaded 'additively' to the MainMenuScene
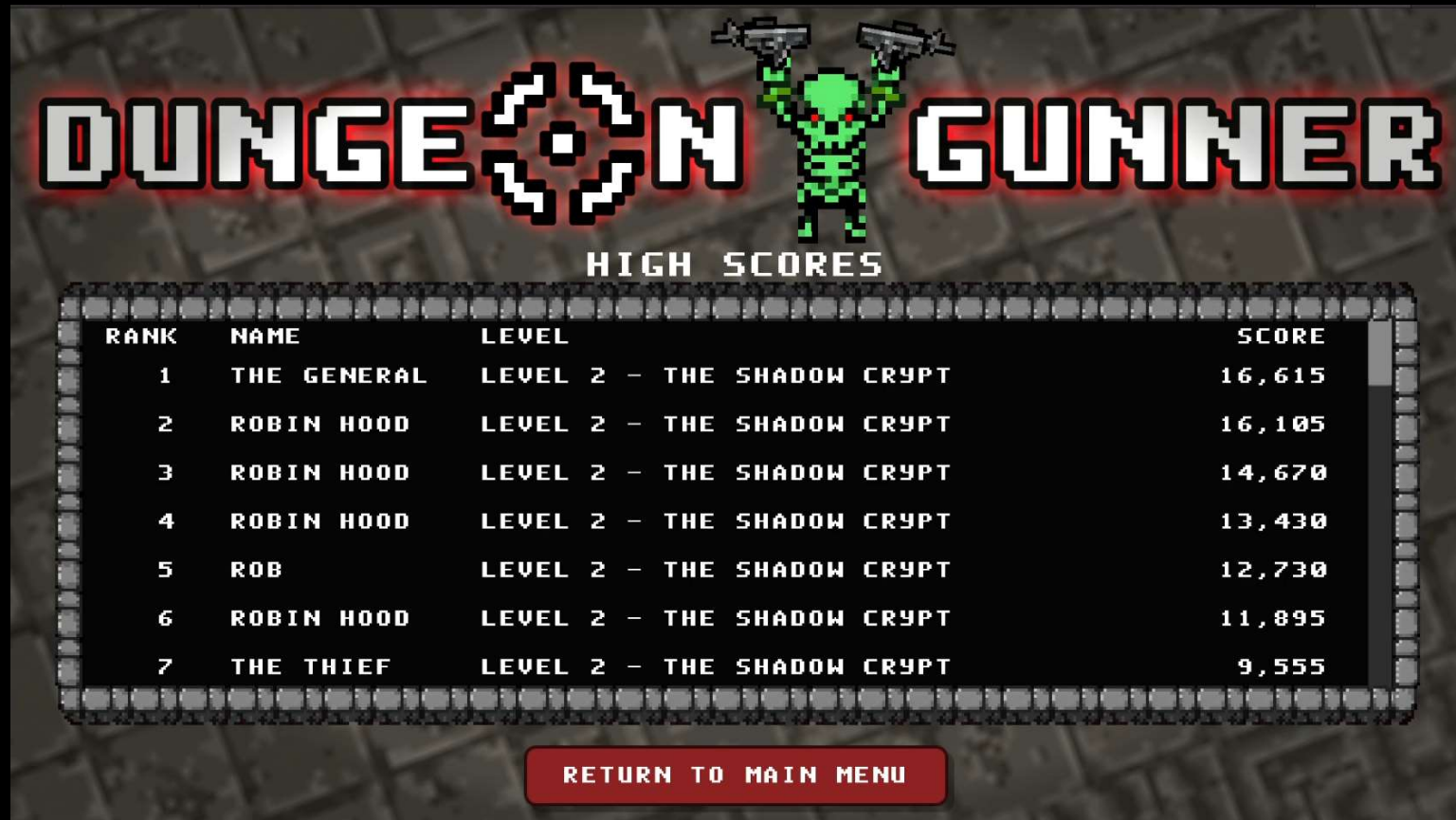
High Scores

Scroll View

The high scores will be displayed in a Unity 'Scroll View' with a scroll bar
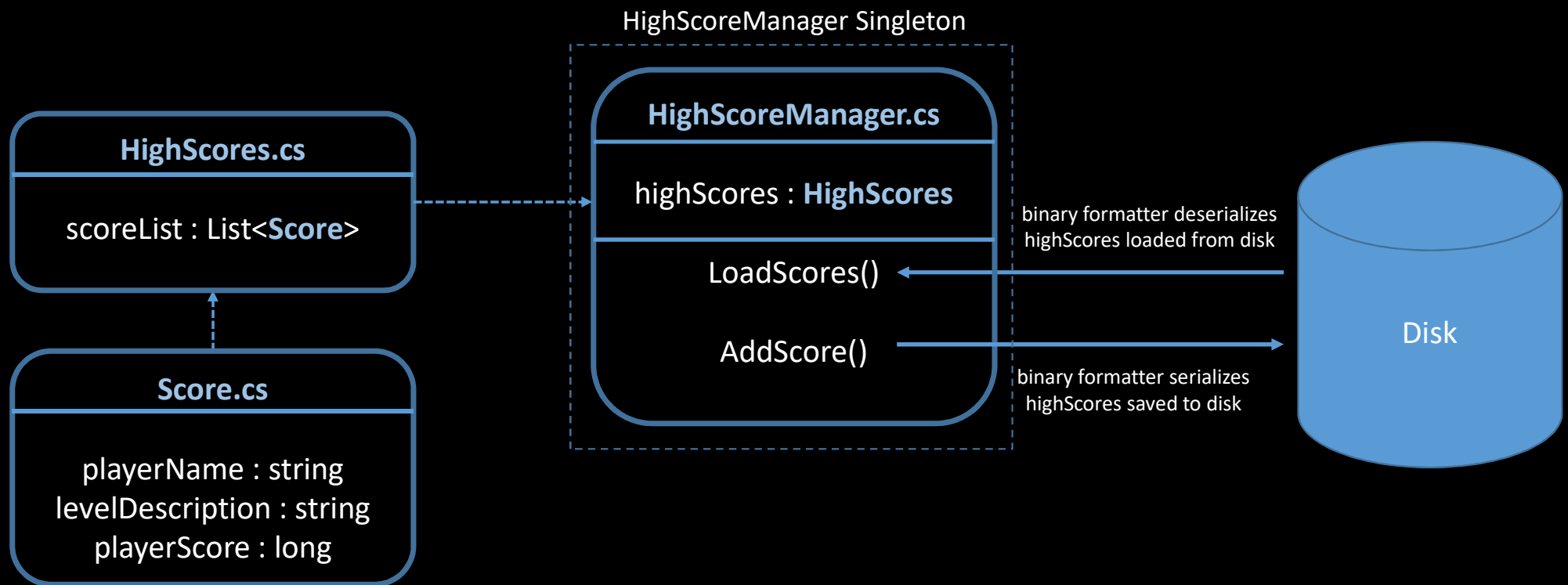
High Scores
Scroll View

The number of high scores to save will be configurable in the Settings class.

High Scores

Saving & Loading High Scores - Class Structure Overview

HighScoreManager Singleton

HighScores.cs

scoreList : List<**Score**>

Score.cs

playerName : string
levelDescription : string
playerScore : long

HighScoreManager.cs

highScores : **HighScores**

LoadScores()

AddScore()

binary formatter deserializes
highScores loaded from disk

binary formatter serializes
highScores saved to disk

Disk

High Scores

Score.cs

```
[System.Serializable]
6 references
public class Score
{

    public string playerName;
    public string levelDescription;
    public long playerScore;

}
```

We'll create a new 'Score' class, which will be serializable. This will hold the player name, the game level they reached, and the player score.

High Scores

HighScores.cs

```csharp
[System.Serializable]
5 references
public class HighScores
{
    public List<Score> scoreList = new List<Score>();
}
```

The HighScores class will be serializable, and contain a member variable called 'scoreList', which will be a List of objects of type Score.

We'll use this list in the HighScoreManager class to keep track of high scores and save/load them to/from disk.

High Scores

HighScoreManager.cs

The HighScoreManager will be a Singleton in the scene.

The HighScoreManager has an 'AddScore' method to add the player score to the scoreList. The scoreList is constrained to the maximum number of high scores defined in Settings.
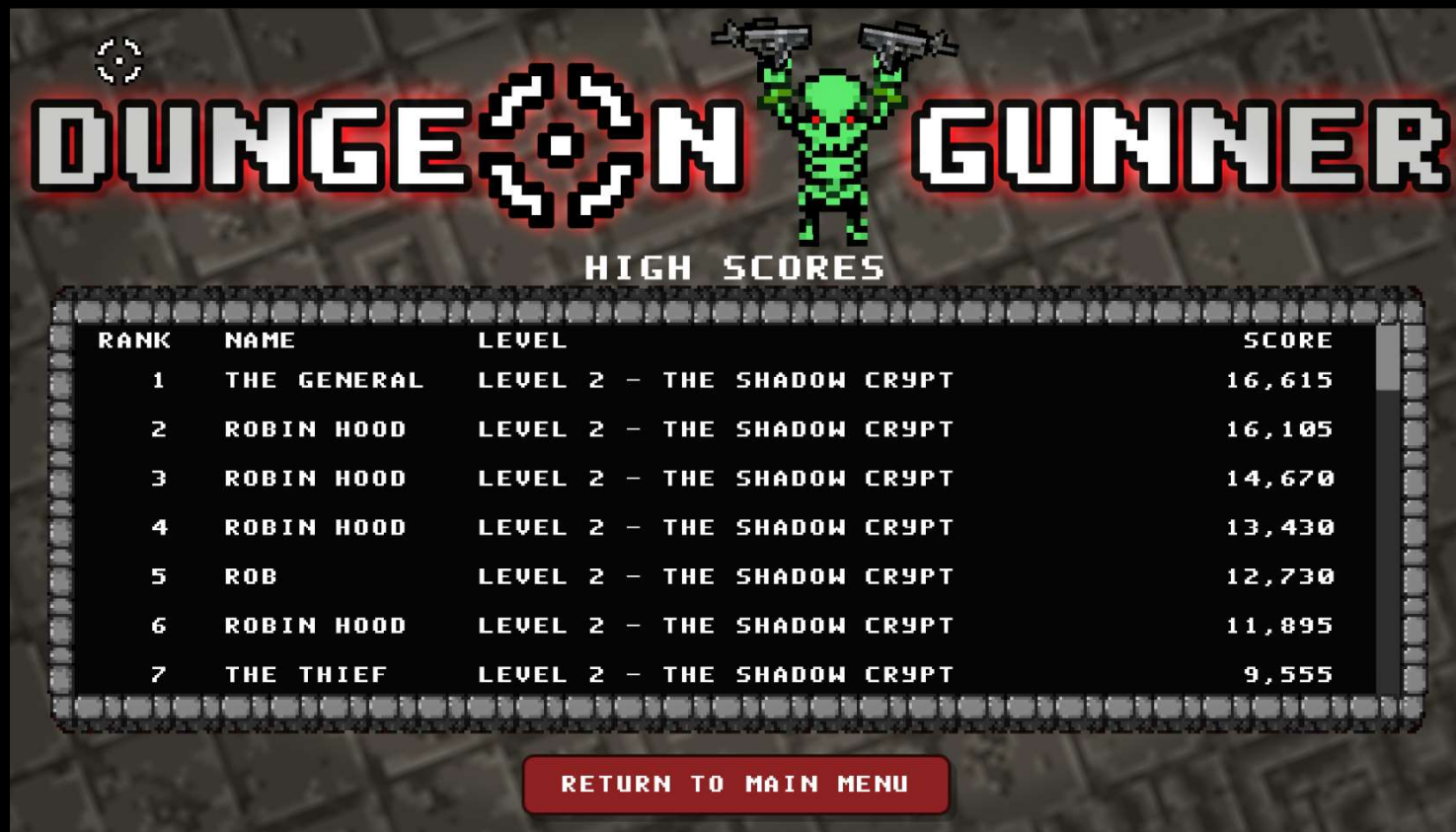
Every time a score is added, the highScores list is serialized and saved to disk, using a C# BinaryFormatter class.

Every time the HighScoreScene is loaded, the saved high scores are deserialized and loaded back into the highScores list.

High Scores
Displaying The High Score UI

High Scores

Displaying The High Score UI

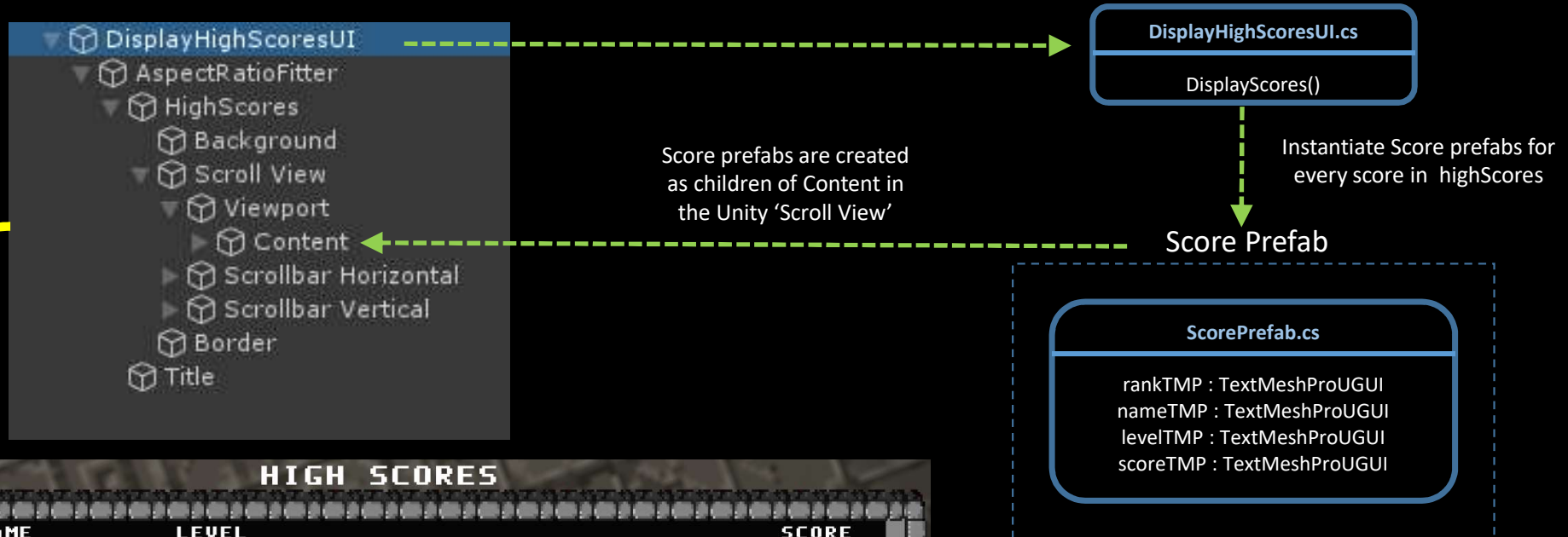|			
The High Score UI uses a Unity 'Scroll View' to display the high scores in a scrollable list	→	The DisplayHighScoresUI class instantiates 'Score' prefabs as children of the Content area of the 'Scroll View'	→
The Scroll View has a 'Viewport' mask so that only the prefabs that are within the mask are visible, and we use a border to frame the content.	→	Scrollbars in the Scroll View allow all the content prefabs to be scrolled through	

High Scores

High Score UI - Class & Prefab Structure Overview

DisplayHighScoresUI
- AspectRatioFitter
  - HighScores
    - Background
    - Scroll View
      - Viewport
        - Content
      - Scrollbar Horizontal
      - Scrollbar Vertical
    - Border
- Title

DisplayHighScoresUI.cs

DisplayScores()

Score prefabs are created as children of Content in the Unity 'Scroll View'

Instantiate Score prefabs for every score in highScores

Score Prefab

ScorePrefab.cs

rankTMP : TextMeshProUGUI
nameTMP : TextMeshProUGUI
levelTMP : TextMeshProUGUI
scoreTMP : TextMeshProUGUI

HIGH SCORES

RANK	NAME	LEVEL	SCORE
1	THE GENERAL	LEVEL 2 – THE SHADOW CRYPT	16,615
2	ROBIN HOOD	LEVEL 2 – THE SHADOW CRYPT	16,105
3	ROBIN HOOD	LEVEL 2 – THE SHADOW CRYPT	14,670
4	ROBIN HOOD	LEVEL 2 – THE SHADOW CRYPT	13,430
5	ROB	LEVEL 2 – THE SHADOW CRYPT	12,730
6	ROBIN HOOD	LEVEL 2 – THE SHADOW CRYPT	11,895
7	THE THIEF	LEVEL 2 – THE SHADOW CRYPT	9,555

High Scores