

COMPUTER GRAPHICS PROJECT REPORT

Berke Derin Berktaş

Beyza Çavuşoğlu

Osman Yasal

Table of Contents

1. About the Game	3
2. Subject	4
3. UI Design & Mechanics	7

About the Game

For our project we have created a game that is a spinoff of a very common eighties game called space invaders. It is a thrilling, action-packed game with stunning modern graphics and exciting new features. The game is set in Turkey, a country steeped in rich history and culture, and we incorporated five famous locations that showcase Turkey's diverse landscape. These locations include the legendary "Maiden's Tower" located in the middle of the Bosphorus, the "Fairy Chimneys" of Cappadocia, the breathtaking "Pamukkale Travertines", the mysterious "Basilica Cistern" of Istanbul, and the majestic "Mount Nemrut". We certainly hope that our game will not only provide players with hours of entertainment but also inspire them to explore and appreciate the beauty of Turkey's cultural and historical landmarks. In our game, we have tried to incorporate various unique factors that in some cases worked and in some cases did not. For instance, we have incorporated a background change feature where if the player obtains a certain score, the background changes. We have successfully completed this task. We have also tried to create an AI player for it to play the game instead of the user, but unfortunately, we were not successful in incorporating this extra task. Since the original game has always been a 2-d game, we have not changed the trend and hence we have created our game on a 2-d plane and our main camera is always in orthogonal view. This allows the user to see the upcoming alien waves and act accordingly on the 2-d plane with the background. We obtained the orthographic 2-d view by setting $z = 0$. We have added filters to the background in order to simulate an ambient lighting environment. We have used several types of animations for different instances and cases. For instance, in the case of a collision between the user's spaceship and any alien object, or a collision between any alien object and a laser blast, a collision figure animation occurs and a collision sound is played. We have used 2 different avatars for the aliens in order to provide variability for a better view experience. For creating and simulating the collisions, we have used numerous free assets from the Unity Asset Store. Then we added collision scripts to our user spaceship and every alien object. Since we have tried many avatars, they can be seen in our assets as well, but after a long while of thinking we chose these figures:



Figure 1: Laser Beam



Figure 2: Collision Animation

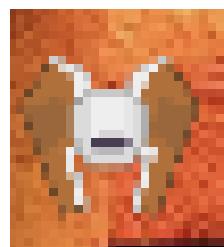


Figure 3: Alien 1



Figure 4: Alien 2



Figure 5: Our Spaceship Avatar

Subject, Mechanics and Gameplay

As the player, your goal is to keep your health above zero and obtain a score as high as possible. The fact is that the alien waves that are constantly

spawning wave by wave are trying to kill you by colliding with you. Whereas you are trying to dodge them with the arrow or wasd keys in order to move out the way and achieve your objective of keeping yourself alive. Additionally, by killing this aliens with your linear laser beams that you can shoot by pressing the space bar or clicking on the screen, you can gain points. Every alien is killed by a single collision with the laser beam that the user shoots.

In the figures below, you can see random screenshots from the game.

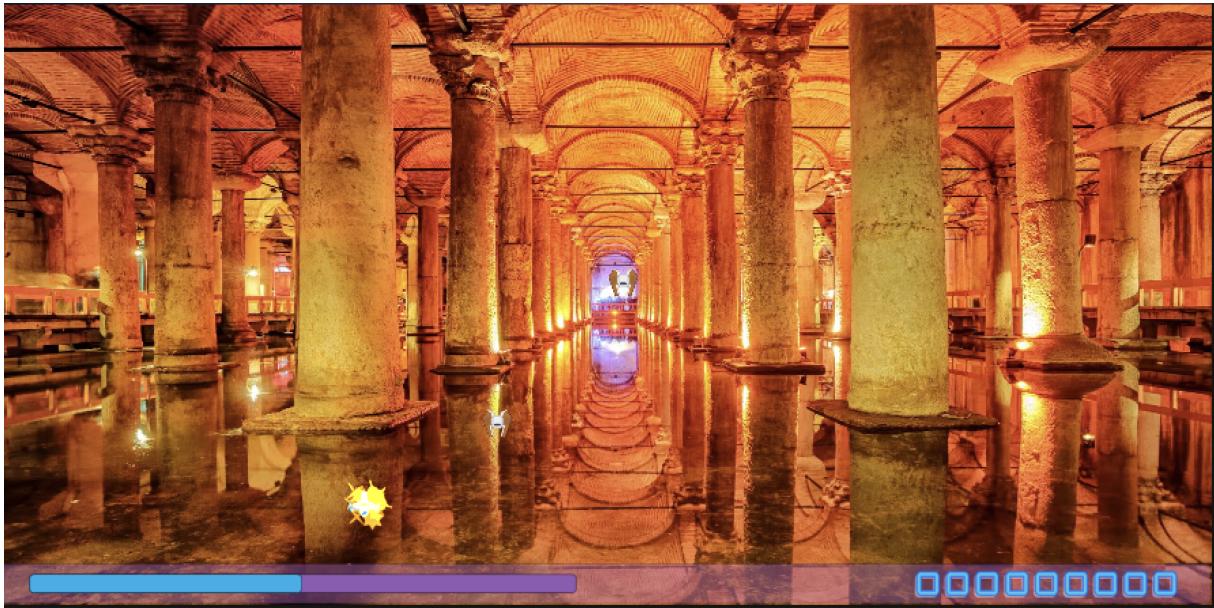


Figure 6: Random Mid-Game Screenshot

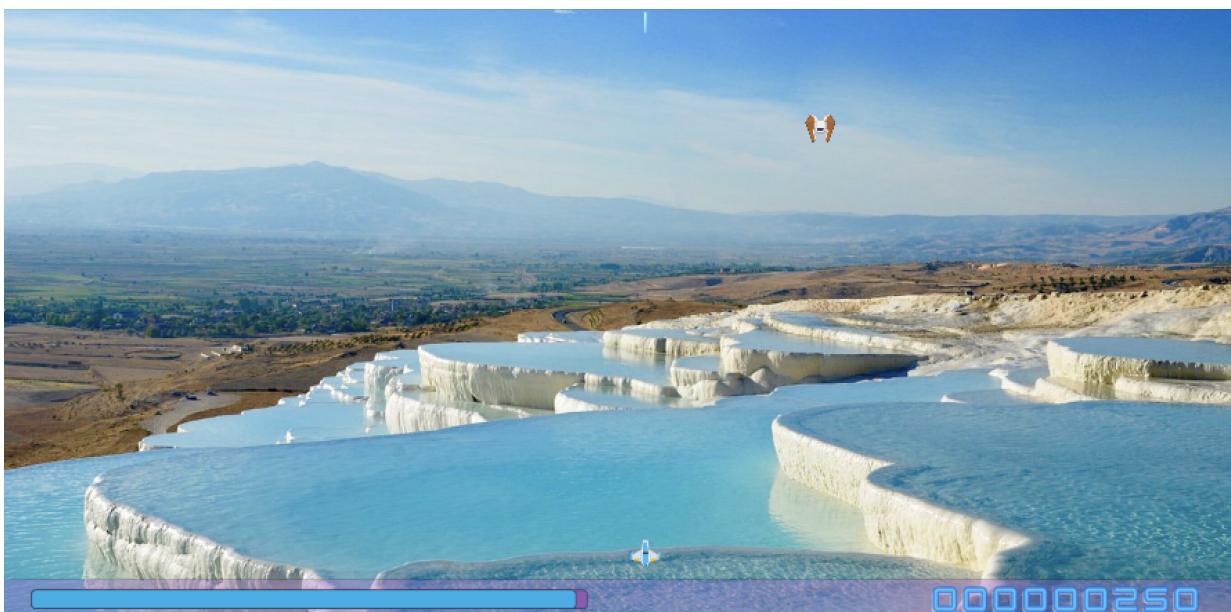


Figure 7: Random Mid-Game Screenshot

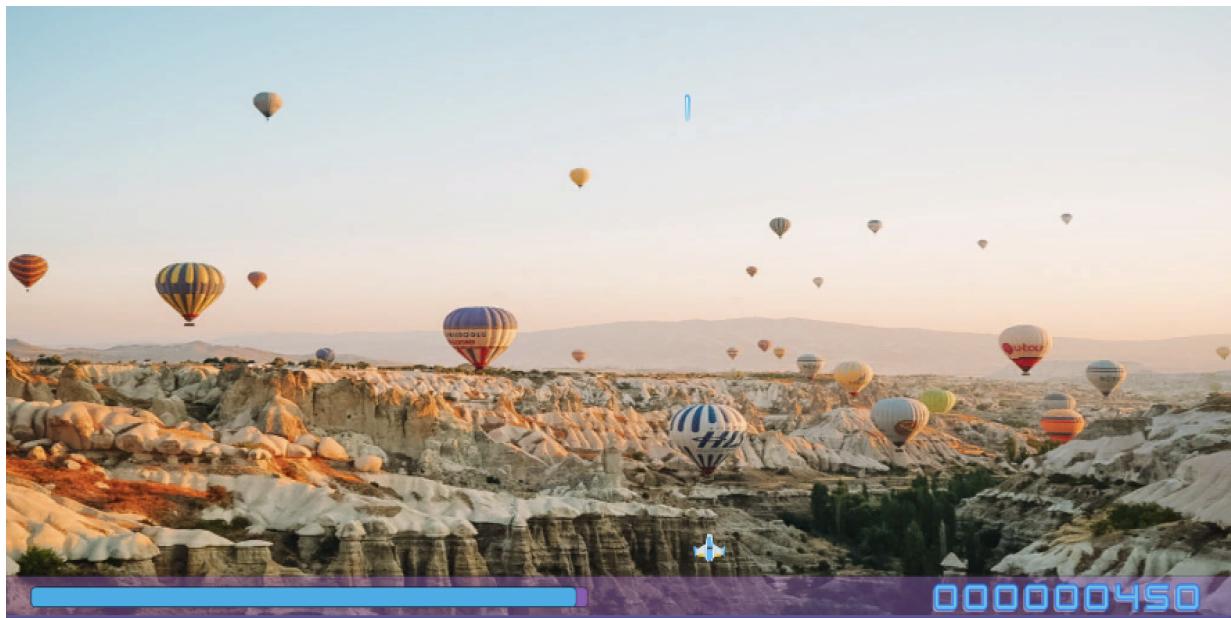


Figure 8: Random Mid-Game Screenshot



Figure 9: Random Mid-Game Screenshot



Figure 10: Random Mid-Game Screenshot

UI Design

There are mainly 2 components of UI that are constantly displayed below the game screen regardless of the background scene, or the level. One of them is the health bar, as seen in the figure below. Here, we have done it in a way that with a single collision between any of the aliens and the user, exactly half of the user's health is gone and once another collision happens, the bar is set to zero and hence the scene changes to the game over screen. We have exactly 3 scenes, the main gameplay scene, the game over scene and the main menu scene. A user can navigate to the gameplay screen from the main menu screen by clicking on the first button . Then once the game ends by the user dying, the scene is changed to the game over screen, where the user can either replay the game or navigate back to the main menu by clicking the corresponding buttons and can also display his or her most recent score. Another UI component that we have is the score button. For the score button, which can be seen in the below figure, we start with 0 and then each time an alien is killed, the score goes up by 50. We have 2 scripts that allow us to spawn waves of aliens in a periodic manner, which are called WaveconfigSO and Shooter. During the gameplay, we have set up the level/background change so that in every 100 points it is changed, but we remain in the same gameplay scene regardless. In order to implement the background change, we have added if conditions to the modifyfunction of the Scorekeeper script. In the beginning of the gameplay, all of the backgrounds are active, but stacked on top of each other, and the script makes them disappear one after another, depending on the constantly increasing score. In the gameplay scene, we have added individual gameobjects for the

player, canvas(for the pink banner of the 2 UI components), scorekeeper for the scorekeeper UI, enemyspawner for spawning new enemies in a periodic wave fashion, audioplayers for the background music and the sound effects that play during a collision and when a new laser beam is shot. Then, lastly various objects for the different backgrounds. We have also created an ObjectPoolManager object that recycles all of the dead alien objects to new aliens so that there is no new waste of space and computation, which is a very nice feature to have.



Figure 11: Health Bar

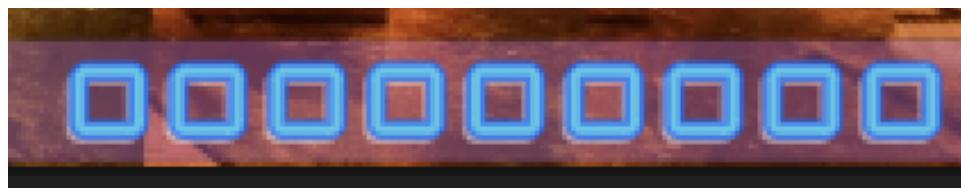


Figure 12: Score Bar