**IT6034 – Game Development**

Project Assessment Task 3: Game Testing Log

GitHub: <https://github.com/jwri211/IT6034GameProject>

John Wright, 20210556

*Approach:*

Individual playtesting is the most common testing approach. Throughout the development, any time a feature was not working as intended it was fixed before moving on to implementation of the next feature. In this manner, the building blocks of the game were solid foundations upon which to add further details. Nearer the end of development, I had other students play a version to get some feedback.

*Individual tests:*

These are some examples of problems I encountered during development:

|  |  |
| --- | --- |
| Problem: | Solution |
| Asteroids that moved off screen were lagging the game | Set the asteroids to detect when they were sufficiently outside of the screen and removed them from the scene using queue.free() |
| When firing, lasers were instantly hitting the ship and spinning off in random directions. | Displace the spawn position of lasers so they appear just in front of the ship, change the collision mask so they do not collide with player ship. |
| Lasers that fired above a certain rate would bounce off asteroids in an unrealistic fashion | Lowered the value that increased rate of fire, so lasers wouldn’t fire as quick but would not have random bouncing effect. |
| When a green power up was collected, the fire rate was not changing back to normal after two seconds | Created a timer that starts when the power up is collected, and uses a signal to indicate when the fire rate should change back. |
| Player ship could move off screen. | Created and enforced margins based on the screen size the ship would not be able to go past. |
| Player Ship appeared above UI elements | Changed the ‘Z Index’ of the UI elements to appear on top |
| Level was not changing when the goal score was reached | The method ending the level was infinitely looping when called in the process function. Introduced a Boolean switch to make sure the function was only called once. |
| When player was destroyed, they could still collide with asteroids and fire lasers while the screen displayed the ‘Game Over’ text | On player destruction, rendered the ship invisible, turned off their physics collision layer and used the prebuilt ability to jam their guns to remove the ability to fire. |
| Sound effects tied to the laser hitting an asteroid were not playing when they should, despite being correctly coded to play at the right time. | Re parented the sound to the level rather than the laser. The laser instance was being deleted upon hitting the asteroid, meaning the sound was only being played for a single frame. |

*Feedback from others:*

When all the game mechanics were implemented, I had other students play a version and the feedback was as follows

|  |  |
| --- | --- |
| Problem: | Solution |
| Game controls are not obvious and trying to use keys to move the ship gives no response. | Create an introductory screen just after main menu, explaining mouse-based control system. |
| Game difficulty is too high | Decreased the score required to pass levels 1 and to from 10 and 15 to 5 and 10 respectively. |
| Game sounds are too loud, uneven, and only come from the left. | Configured different sounds to be more even by changing decibel levels and repositioned all sounds to the center of the viewport. |
| Game speed is too high | Reduced both physical speed and spawn rate of asteroids in both levels. |
| Goal score is not clear | Placed the label for the current score and the goal at the top, center of the screen to make it more obvious. |