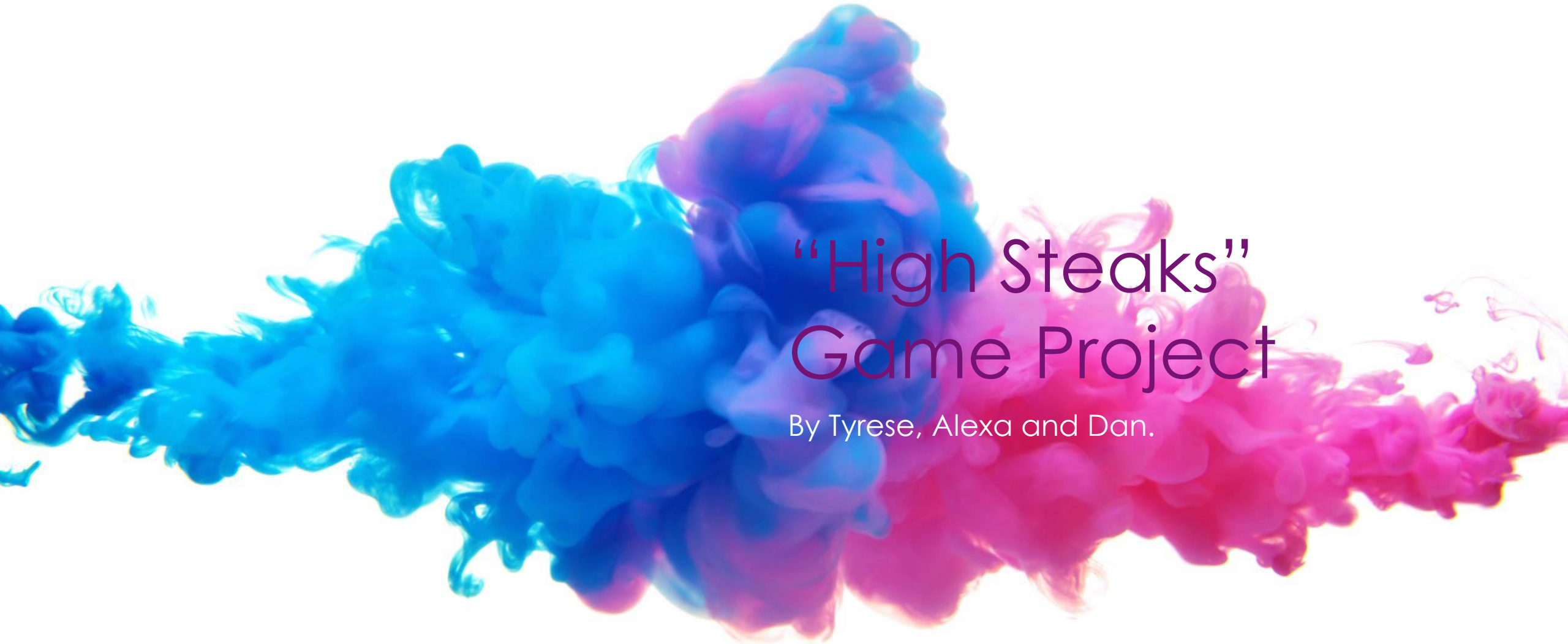


“High Steaks” Game Project

By Tyrese, Alexa and Dan.

Storyline of “High Steaks”:

- Story: A farmer receives a high volume of orders for the meat he sells over Christmas. Help him with preparing the delicacy he's known for - foie gras! Force feed the animals with steaks till they drop - and take a breather to enjoy the golden apple the local beauty throws you... Merry Christmas!
- Main Character: The main character is named Cluck Bacon, a middle aged farmer who opened a business to support himself in a declining economy. However, he was not expecting the amount of orders that came in and developed a new way of processing the animals for his products. A beautiful woman – Goldie Fawn - who lives nearby is enamoured by his genius and offers him some of the precious, high quality golden apples from her orchard.
- Objective: Feed the running animals as fast as you can! Use the golden apples to recover your health and stamina.

Design considerations:

- Preparation: in order to create our game we had to learn how to work in Unity and learn the C# programming language.

- Scripts identified and why: in our case, each script and what they do have different uses. An example would be:
- “Destroy out of bounds” to ensure all objects beyond the game bounds would be destroyed to improve game performance.
- “Detect collisions”: collision script for the animals created so we can modify the values and have the animals destroy upon colliding with the steaks.
- “Player controller”: created to allow the player to have their health increase, shoot and create bounds.

Scripts:

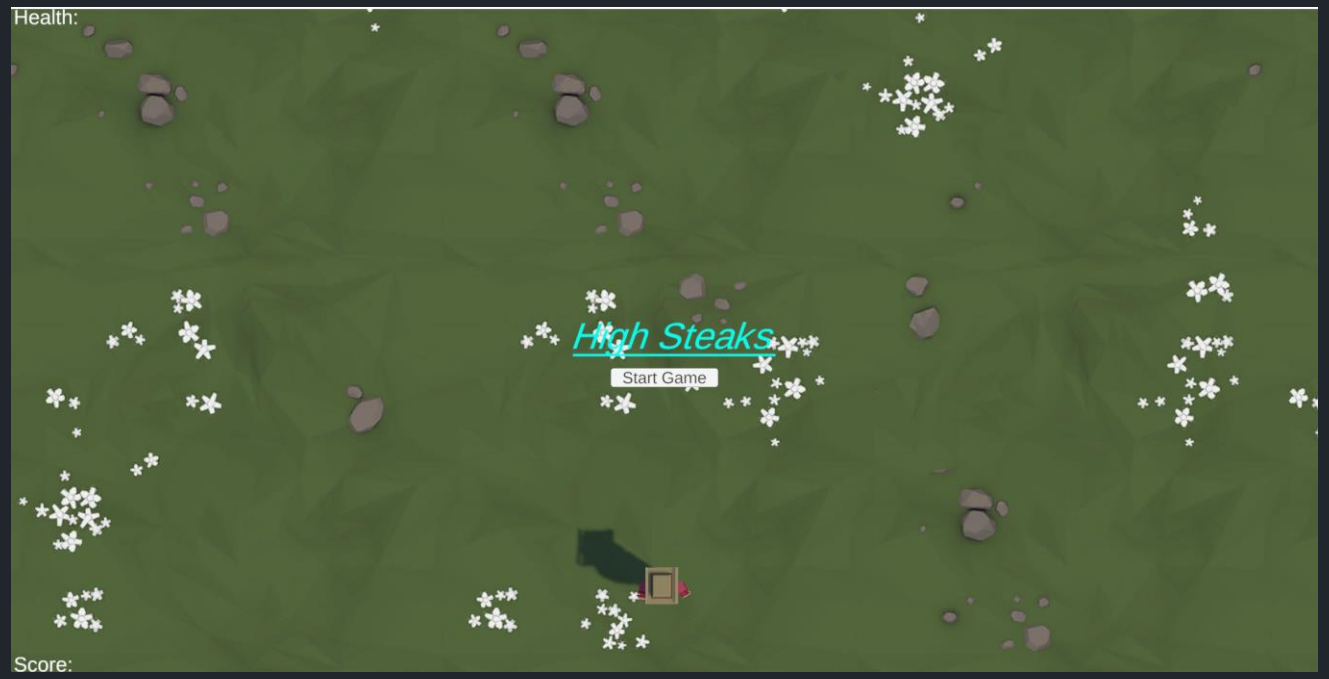
- “Destroy out of bounds”
- Detect collisions / detect collisions: apple
- Game manager / power up
- Game over
- Move forward / move forward animal new / move forward apple
- Player controller
- Start button

Development:

One of the issues we experienced was trying to find the most suitable software for our game idea. We eventually had to reevaluate, downsize and change our original game pitch to accommodate the change of software and the steep learning curve. Another issue we came across is that some of the tutorials we used were made for the older version of Unity and thus weren't particularly helpful, especially as we weren't experienced enough to adjust the older tutorials to work in the updated software. Some of the materials were difficult to find even if they were updated.

We overcame this by looking through comments and reverse engineering some of the scripts to make it all work. As for the game pitch, we decided to simplify to ensure it was workable.

Demonstration of game:



Evaluation:

Improvements for next time:

- An improvement would be to add a timer that stops once you reach the goal of 100 score.
- Another improvement would be add extra models for the projectiles such as carrots.
- Adding difficulty levels like increasing the spawn rate of animals or increasing their speed.
- A leaderboard for keeping the completion times of players in order provide a sense of competitiveness.
- levels that introduce new animals and movement patterns as well as powerups.

Lessons Learned:

- Time constraints: we learned to manage our time and production time to allow for the learning curve, scheduling (Christmas break) and to fit with the size of the project.
- Management: to ensure efficiency, we managed each other to make sure we were on schedule, everyone had a job to do and that we were communicating out progress and any issues we had.
- We learned that to achieve a specific collision detection we had to use an “if” statement.
- Ran into a syntax error while writing the “target Rigidbody” code – fixed by reviewing contents and adding a semicolon and removing an unnecessary bracket.
- While coding the health system, we forgot to add an “int” which stopped the game from running as intended until we reviewed another piece of code and fixed the mistake.