Jug Run: A Student's Journey
Isabel Kraus, Kobe Darko, Hank Alderoty, Ignacio Bernaldez
COSC 435 Final Project Proposal

Statement:

We will be developing a game that depicts the often done, very difficult, run from the Jug to the home of our hero through the cold streets of Hamilton. There will be a main menu / title screen before the Townhouses. Our Student will run past important Hamilton landmarks, avoiding snow banks and cruisers. Concurrently, the user will eat slices and pick up jackets to recharge their constantly diminishing health bar. As the player progresses, the speed and frequency of events will increase.

Our scene takes place on Broad St. The street will be 3D, cartoon, and minimalist representation of Hamilton's most famous street. The road will be surrounded by 3D buildings and trees, creating an immersive environment for the user. Our main focus is to develop an immersive game that depicts a consistent issue many Colgate University students face. Using a variety of graphics, animations, and gameplay features, the user will experience and navigate game begins, allowing the user to change the difficulty of the game, the ability to pause the game, and an explanation of how the game works. To win the game, the user must control the hero and navigate them to the townhouses, their home.

Sketches and Drawings

Overhead view of the game. From this specific section, the cruiser (pictured in the top left section of the road), is coming towards the Student. Also pictured are snow banks (the layered "splotches" on the ground") and pizza slices. There are also 4 sided shapes on either side of the sidewalk. These represent the homes found on Broad St.
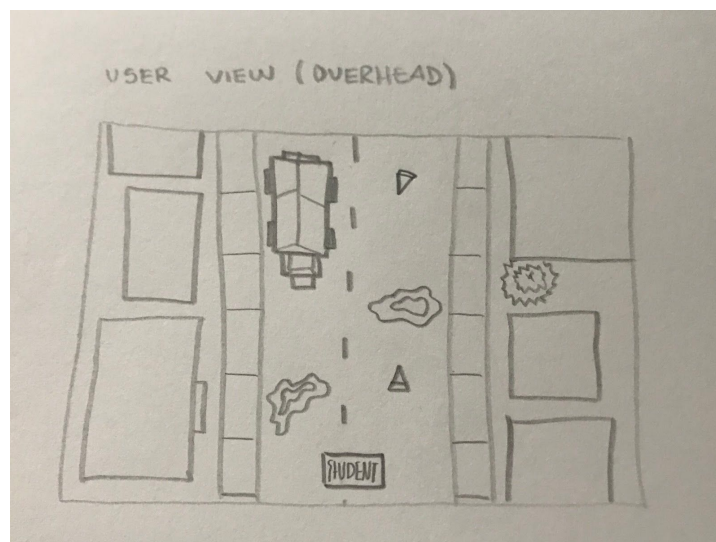
This is a 3D view of the pizza and jackets the user will be able to pick up, subsequently increasing their health.
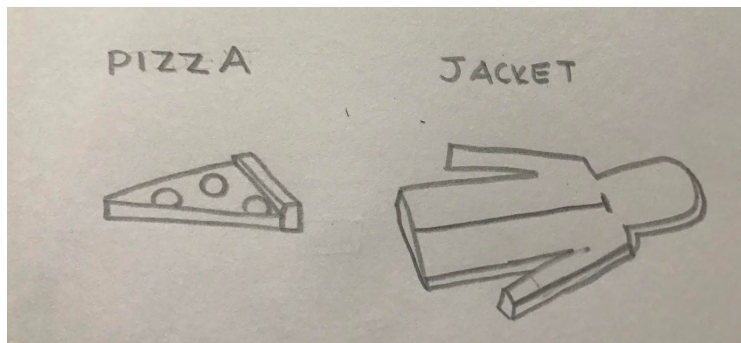
Our cruiser, roughly drawn below, will be one of the main collision objects the user must avoid. The headlights will be turned on and shining light.
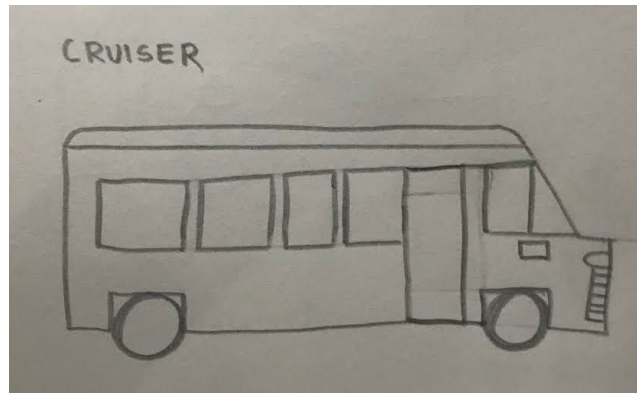
This is our student, seen from behind.

Technical Outline:
1. **Buildings/Houses:** We will create a function that generates a building of random size and color. The function will be called repeatedly to simulate the varying buildings and houses along Broad Street. We are thinking of making random buildings line the right-hand side of the street, so that we can include trees and Taylor Lake on the left. As the player moves down Broad Street, there will be houses on the right of varying color and size, and the left will have trees up until a certain point where you will see the lake.
2. **Health Bar:** The health bar start completely filled and drain at a constant rate. A full health bar will look like a filled rectangle and an empty health bar will look like the outline of a rectangle. As the health drains the filled rectangle will become shorter and the outlined rectangle will become taller.
3. **Snowfall:** We will use the particle aspect of Three.js to model snowflakes to help simulate snowfall and winter. They will fall across the screen from top to bottom.
4. **Jacket and Slices:** We will have jackets and slices scattered along Broad Street that the player can collect to keep up health. We will create the jackets by using rectangles and ovals, and the slices will be made with triangles, rectangles, and circles (see sketches). A function will randomly place them along Broad Street. Starting at the beginning section, the function will generate slices and jackets more frequently, and as you move down Broad St, they won't appear as often.
5. **Cruiser:** This will be modelled with a cube and circles for the wheels and we will model its animation off of the cart from lab 6. The students can collide with cruiser from any angle, so they must avoid them at all times. This means our collison function will have to include all sides of the cruiser.
6. **Snowbanks:** The snow banks will act in a similar fashion as the jackets, but in the inverse way. The user can collide with them from any angle an experience a loss of health.
7. **Road:** Along with the functions that randomly generate cruisers and snowbanks, we will need to have a farther distance function that will make sure that these objects are not spawned too close to each other, allowing for the player to be able to actually navigate through them. We will need to have something similar for the slices and jackets so that they do not spawn too frequently, making it too easy for the user. Changing the difficulty of the game will help dictate the frequency of the jackets and slices.
8. **Collision:** The student, jackets, slices, cruisers, and snowbanks, will all have collision boxes around the models that will trigger when the student is healed or hit. Similarly, there will be bounds on the left and right side of Broad Street, so that the student can't run off of the street.

Objectives:

1. Users will play as **The Student** (Ignacio)

a. The Student will not be seen from the front making modeling easier

b. The Student has to be able to stand still, run forward, and jump side-to-side

2. The Student is running along a **road and sidewalks** (Isabel)
   a. The road has two lanes and a sidewalk on either side
   b. 4 lanes total for The Student to run on
   c. This object has to be designed so that it can be repeated and look like one flowing road
   d. We may use animation to mimic a moving road - we can have a dotted yellow line running down the center of the road, so yellow rectangles are moving down the screen to look like The Student is running down the road

3. The Student's health will be tracked on the **Health Bar** (Hank)
   a. The Health Bar starts fully filled and drains at a constant rate
   b. If the Health Bar completely drains the game is over the player has lost

4. While health is always draining, if The Student picks up a **slice** or **jacket** then their health bar will be pushed up a certain amount (Kobe)
   a. Slices will have simple designs and be placed in any line
   b. The jackets will be random colors
   c. When The Student picks up a jacket the jacket they are wearing will change to that color

5. In the same fashion, if The Student hits a **snow pile** or **cruiser** then their health will be pushed down a certain amount (Ignacio)
   a. Snow piles will have simple designs and will only be placed on sidewalks
   b. The cruisers will only drive down the road lanes - when you move down Broad St, we may make it so that the cruisers change from one lane to another, but in beginning sections they will only drive down one lane
   c. We will use collision boxes to manage the collision of the student with snow piles or cruisers

6. In the game The Student will be running by randomly generated **houses** (Isabel)

7. In the game The Student will run by **Taylor Lake** (Hank)
   a. We will model Taylor Lake on the left-hand side of the street, so as to provide a landmark for the player, indicating how far along Broad Street they are

8. During the game there will be **snow falling** (Kobe)
   a. The actual snow is going to be a simple model, likely small white circles with alpha blending to make some look more transparent than others

Reaching Objectives:

If we have time…

1. Jackets won't push Health up, but instead **slow down the rate that health is draining**
2. We want to make **two Students**, one male and one female so users can choose

3. We will try to map out the amount of time / progress the student has made by showing them a small map of their distance to the end, which will be placed in the corner of the screen somewhere.
4. Design one or two of the more distinctive locations on Broad Street in detail, like Gamma Phi's little blue house, the Creative Arts House, the fraternity houses, etc.

Bibliography:
https://gamedevelopment.tutsplus.com/tutorials/creating-a-simple-3d-endless-runner-game-using-three-js--cms-29157

https://threejsfundamentals.org/threejs/lessons/threejs-game.html

http://www.bryanjones.us/article/basic-threejs-game-tutorial-part-5-collision-detection

https://www.evermade.fi/pure-three-js-hud/

https://threejs.org/examples/

https://github.com/wanfungchui/Boxy-Run