Emrys Sledding Down the Ski Hill
COSC435 Final Project Proposal
Group Members: Asad Jamil, Zoila Rodriguez, Leslie Subaldo, and Jingxian Wu

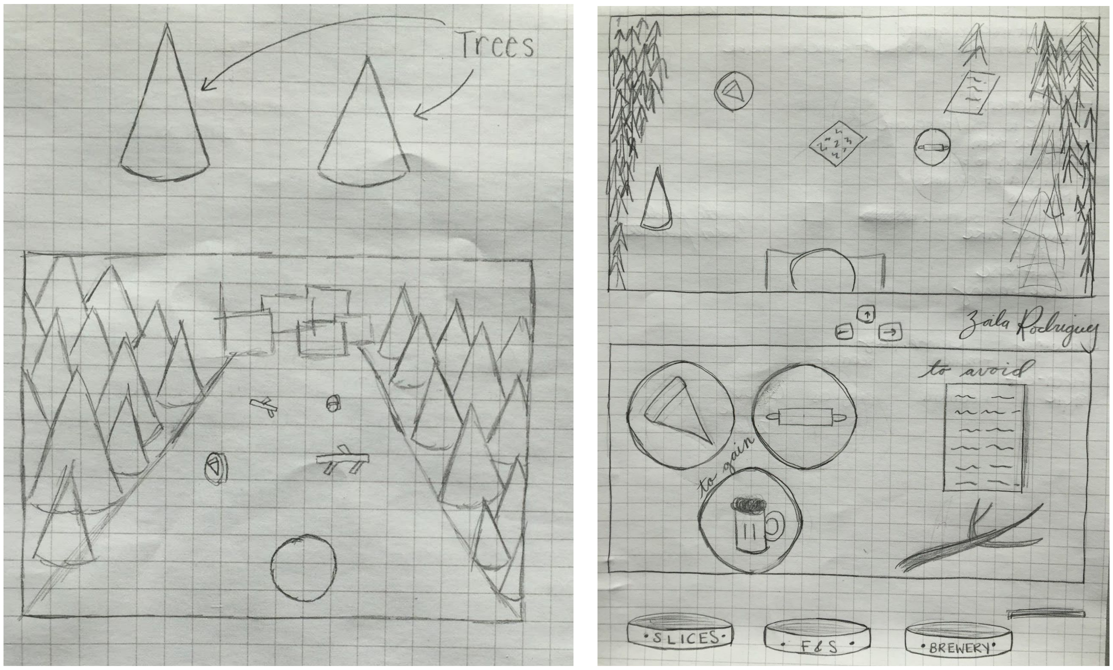
Introductory Statement

Our goal is to design and implement a game in which the user is able to control Emrys, President Casey's dog, on a sled down the ski hill for a certain amount of time. The objective is to collect food tokens while avoiding obstacles such as branches, papers, and rocks. The game ends when the player collides with a certain number of obstacles or when time runs out.

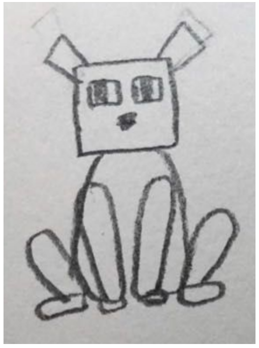
The scene is able to be viewed from both first-person and third-person perspectives. When the game starts, Emrys will start in the center and immediately start sledding down the ski hill. The trees are going to be on both sides of the ski hill, creating a boundary for Emrys' movement. In front of Emrys will be the various tokens that the user can collect and the branches that the user can avoid. These objects will be randomly spread out. Buildings of campus, e.g. Huntington Gym, will be rendered at the "end" of the hill to show campus in the distance.

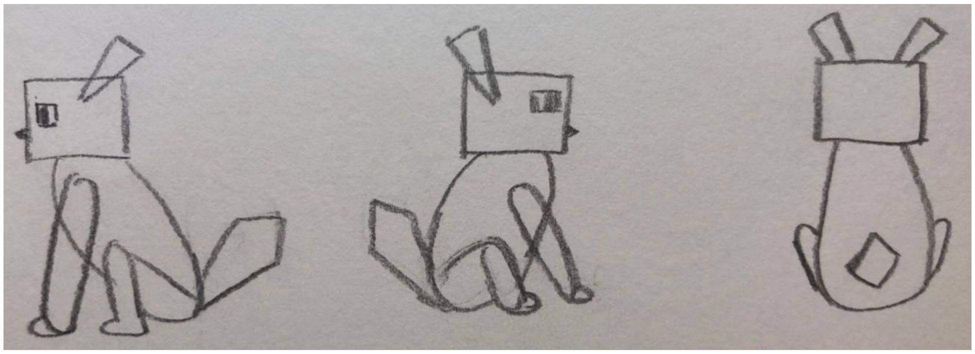
The key user action in this game is to control Emrys as he sleds down the hill. The user can use the left and right arrow keys to move him left and right, the up and down arrow keys to speed up and slow down the sled speed, and the spacebar to jump over obstacles.

Sketches and Drawings



(Left) General overview of scene, detail of trees; (Right) detail of tokens and obstacles





(Above left) rough sketch of Emrys from the front (modeled after heroes demo)
(Above right) more sketches of Emrys from sides and back

Technical Outline

1. **Ski Hill**: We will model the ski hill as an endless plane[9] so we can keep moving the sled down the hill endlessly. We will also apply a snow texture on the hill.
2. **Emrys and the sled**: We will create an Object3D to hold models of both Emrys and a sled. Emrys will be modeled based off of the cat hero in the heroes demo[1] with appropriate animations, and the sled will be a rectangular box.
3. **Trees**: We will model the trees similar to the ones shown in class i.e. composed of simple geometric figures like cylinders and cones. We will first model trees on a horizontal surface, then change the code to model trees on a sloped plane. They will be randomly generated on the sides of the ski hill with a control in overlap and changing density.
4. **Tokens**: We will model them as textured thin cylinders, with a different texture for each type of token. They will appear standing up, i.e. not laid down on the ski hill. They will rotate clockwise along their vertical axis.
5. **Branches, rocks, and papers**: We will model them using an Object3D composed of simple geometric figures such as cylinders, cones, and planes.
6. **Buildings**: We will model a building from campus, namely Huntington Gymnasium, to place in the background. We will incorporate bump mapping to make it have rocky textures so that it appears more realistic.
7. **Skybox**: We will use blue sky with clouds as our skybox for the scene
8. **Particles**: We will incorporate particle effects in the scene to have a snowing[2] effect. We will also have particle effects for collisions[3] with tokens and branches.
9. **Shadows/lighting/fog**: We will use Three.js shadow map features[4] to handle shadows for all objects on the hill, including Emrys and the sled, trees, tokens, and branches. Shadows will be generated from a light source that moves with the sled around the hill.
10. **Moving Emrys and the sled**: We will allow the user to move Emrys and the sled to the left and right with the ArrowLeft and ArrowRight keys, respectively, up to where the trees are. We will also allow the user to speed up and slow down the sled with the ArrowUp and ArrowDown keys, respectively. Finally, if time allows, we will allow the user to make the sled jump with Spacebar.

11. **Physics engine/Collisions**: We will incorporate Physijs[5] to handle the physics of the sled's movement, particularly jumps, and any collisions with obstacles.
12. **Animations**: Tokens will rotate clockwise along their vertical axis and the Emrys will move on the sled in the same direction as the user moves the sled.
13. **GUI**: A graphical user interface will be used for the HUD of the game, displaying the player's score as well as the number of lives and time remaining. It will also include buttons to toggle features such as particle effects[6] and shadows on/off[7] and change camera perspectives.
14. **Toon Shading**: Apply toon shading using Three.js features[8] on Emrys and the sled to achieve a more game-like/cartoonish look.

Objectives

1. Model main objects i.e. ski hill, Emrys and the sled, tokens, branches, papers, rocks, trees, and building according to schematic diagrams using Three.js geometry:
2. Use winter colors for the ski hill. Randomly generate trees at the side of the ski hill with a control in overlap and changing density.
3. Include blue sky skybox with clouds for a full experience of the environment.
4. Implement a shadow map to make Emrys and the sled, tokens, branches, and trees i.e. all objects on the hill cast shadows for a more immersive scene.
5. Implement a particle effect to show snow falling from the sky, contributing to the aesthetic of the snowy ski hill.
6. Incorporate a physics engine, potentially Physijs, to realistically handle collisions and jumps.
7. Implement different particle effects when the ball collides with tokens and branches to notify the player of the (type of) collision.
8. Implement animations for the tokens and Emrys to make the scene more dynamic. The tokens will slowly rotate in the y-axis in place. Emrys will also move left and right on the sled when the user uses the left and right arrow keys.
9. Implement a menu through a GUI that will contain buttons to enable/disable snow particle effects and shadows and to toggle between first-person and third-person perspectives.
10. Implement toon shading for Emrys and the sled to make them look more cartoonish/game-like.

Stretch Objectives

1. Incorporating tree generators to make more realistic trees.
2. Creating a skybox using images taken from the ski hill.
3. Implementing spline curves to create more realistic branches.

Bibliography

1. https://codepen.io/Yakudoo/pen/oXJYxy

2. http://jsfiddle.net/3drjwj2n/
3. https://aerotwist.com/tutorials/creating-particles-with-three-js/
4. https://aerotwist.com/tutorials/creating-particles-with-three-js/
5. http://chandlerprall.github.io/Physijs/
6. https://github.com/mrdoob/three.js/issues/1690
7. https://github.com/mrdoob/three.js/issues/2466
8. https://threejs.org/docs/#api/materials/MeshToonMaterial
9. https://github.com/tapio/jetpursuit/blob/a78a4efd6ce5ea4f2cd41b74c23b1e803fa22095/js/world.js#L70-L73