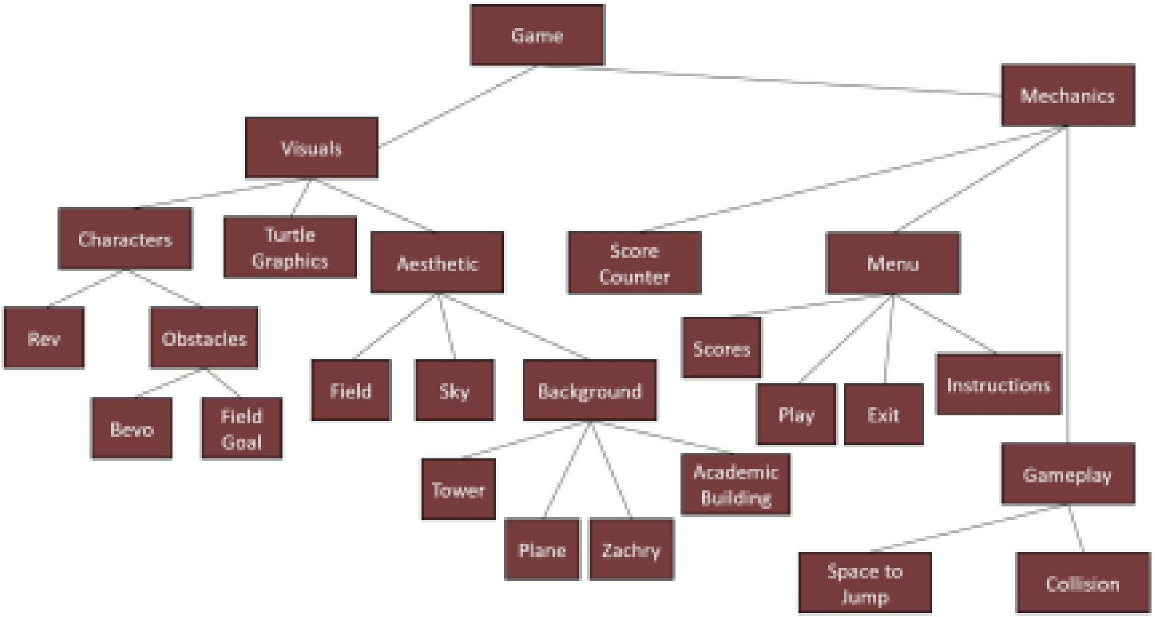
Rev Run

David Lloyd, Harry Oclarit, Austin Thames

Hierarchy



Reasoning

We decided to create Rev Run, as opposed to one of the pre-existing options given for games, to challenge ourselves in making a more complex game than most other groups. We were drawn to the many moving parts of making a fast-paced game, as many different features result in a more challenging design process. Furthermore, we started by attempting to make Minesweeper but could not figure out how to code the beginning board break.

Instructions

Begin by typing “pip install ursina” in the terminal and “pip install turtle” after that. Download the “Reveille_Run.zip, unzip the file, open the folder, then go to “Reveille_Run.py” and click the run button. From then on, instructions for the game itself can be found by clicking the “instructions” button within the game. To check recent scores, click the “scores” button, to quit early, click the “exit” button, and to play the game, click the “play” button!

In the program, variables with Entities() are the objects that move in the game, ground(background), reveille(reveille character), plane(plane entity on the back), etc. The first part of the program includes the turtle graphics that draw “REV RUN.” The functions under the turtle graphics part include the buttons and texts for the main menu. The update function is the backbone of the game, allowing for changes in the entities while the program is running. The two collision entities, goal_post, and Bevo, move to the left in the game and if Reveille collides with any of these entities, the game stops and the score is saved in “highscore.txt.”