Game Logic: The game consists of three levels. To complete each level, players must balance the seesaw.

Game Design: For the project, we chose to create a website, utilizing HTML and CSS for our game design, and JavaScript for game logic.

A screenshot of a computer

Description automatically generated

Project Steps and Problems We Faced:

We began by designing a seesaw made of two rectangles. We included input boxes for force and distance for user inputs. When we applied the same torque on both the left and right sides, the seesaw balanced correctly.

Initially, the force values were limited to positive numbers, while we allowed the distance values to range freely. However, we later adjusted the distance range to a minimum of -5 and a maximum of 5. This change was made to add purpose to the game, as using infinite values made it challenging to design levels and draw the seesaw. In our design, negative values for distance apply to the left side of the seesaw, and positive values apply to the right side. For example, if we input a torque of 50 Nm (calculated as 50N \* 1 radius) on the right side, we would need an equivalent torque on the left side to maintain balance.

Additionally, we created a background for our project using Microsoft Paint and uploaded it to our HTML and JavaScript files.

A pixelated image of a pyramid in a field

Description automatically generated

Initially, we planned to draw two children—one boy and one girl—on either side of the seesaw, with their weights varying based on user input for force. However, we reconsidered this idea because the maximum force input (300N) is equivalent to approximately 30.6 kg. This weight would not make sense if we illustrated an overweight child, so we decided to replace the characters with fruit images. This change allowed for more reasonable weight representations.

We sourced fruit images from opengameart.com, uploading 5-6 fruit images as PNG files, which we incorporated into our HTML and JavaScript files.

A screenshot of a computer

Description automatically generated

A screenshot of a phone

Description automatically generated

Level System Explanation:

The game features three levels, each with the objective of balancing the seesaw. For example, in the first level, we start with a banana and an apple on the right side of the seesaw, and the player’s goal is to achieve balance.

Initially, we allowed players to input both force and radius values. However, we quickly realized that this made it too easy to progress through the levels, as users could pass simply by entering force values without considering distance. For instance, if we calculated 110 Nm of torque on the left side, a player could input 110N for force and -1 for radius to balance the seesaw, thereby skipping the crucial torque calculation. To address this, we decided to modify the requirements: in Level 1, players can choose the force while the distance is set at a constant of -3, forcing them to compute the torque. In Level 2, we set a constant value for force and allowed players to determine the distance, enhancing the game's complexity.

A screenshot of a game

Description automatically generated

Website URL:

Initially, we considered purchasing a domain for our project. However, upon discovering the high costs involved, we decided against it to avoid additional expenses, especially since we were already paying for the course. Instead, we opted to use a GitHub URL. One of our group members created a repository, uploaded our files, and published the project publicly, allowing us to obtain a free URL from GitHub.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated