Planning:

Overview:

Our game is a simple baseball simulation game. One player is the pitcher and the other is the batter. The pitcher presses space to throw the ball and the batter must time his swing correctly by using B to hit the back back. If they miss, they lose and if they hit the ball, they get a home run and win.

Objective Statement

The question we want to answer with our prototype is the following: How can we use the unity physics engine to create a quick and fast gameplay experience.

Design Rationale:

We envisioned a fun, quick game that utilizes an almost quick-time event in its core gameplay. From our research we knew that the initial baseball hit during a baseball match is fast and quick. We want to make it fairly realistic with just enough arcade elements to make it easy to understand for the player.

Our Team and Roles:

Neil McCuaig - Writer
Jake Devos - Programmer

Metrics Research and References:

Baseball:
Weighs 5 ounces
9 inches in circumference

Wooden Bat:
42 inches long
2 and ¾ inches in diameter

Aluminium Bat:
33 inches long
2 and ¼ inches in diameter

Mechanics:

The main mechanics of our game involve timing a button press correctly in order to get a home run to win the game. We wanted to mix a timing mechanic with the unity physics engine in order to get a quick and effective fast paced game.

Controls:

Space to have the pitcher begin to throw the ball.
B to hit with the bat.

Interface:

The interface will be fairly simple. Some text prompting the player on what button to press and a charge meter that will slowly fill as the player spams the charging button.

Resources:

https://entertainment.howstuffworks.com/physics-of-baseball6.htm
https://www.measuringknowhow.com/baseball-bat-dimensions/

Development Documentation:

We decided to make a sports game, since most sports games are physics based to match with their real life equivalent. We specifically opted for baseball since one of us had experience playing baseball and would be able to at least roughly judge the feel of it.

We had an issue with imparting force onto the bat, which was solved by a coding suggestion we got during class.

We ran into another issue where the assets and textures were not properly being committed to github, leaving Jake with the only fully functional copy of the game.

Questions for our testers:

Question 1: What sorts of changes could you incorporate to improve the physics based experience?

Answer 1:

Question 2: Did the way the ball and the bat interact feel right to you? Did it impart an appropriate amount of force and react to that force correctly?

Answer 2:

Question 3: Do you think adding different bat types and making the player try to pick which direction they think the pitcher is swinging would be more interesting? Or do you think we should focus on refining the current bat and ball?

Answer 3:

Distance tracker
Ball cam