Final Prototype Changes/Goals

Overall Stack Ranking

1. Balance movement speed and player “floatiness”
2. Addressing how long players will be in the air
3. Design the level and tutorial based around the movement and ability balancing
4. AI characters in the playable level; ball-shooting machines in the tutorial
5. Bug Hunting and Fixing

Level Design

1. Address player movement speed, dash time, and in general how long we want players to spend time in the air.

- Teach dash jumping in the level
- Spline movement set up around the map
- Modify it to fit faster movement?
- Moving platforms?
- Tutorial level to teach player the movement mechanics
    - Hit a switch to open a door with the dodgeball.
    - Gap the player must clear with an arc jump, dash, etc.
    - Door from tutorial to level closes behind the player.
- Plan for getting hit from above.

Artwork/Animation

- Dashing
- ~~Jumping~~
- ~~Falling~~
- ~~Superhero landing~~
- Extra strafing animations?

UI Elements

- ~~Finishing Control Screen Layout with ReadMe information~~
- ~~Enemy Kill Counter (for Gameplay Screen)~~
- UI Pop-Up Animations for when the player gets hit or when the player hits an enemy

Programming

STACK RANKING

1. Cannons that shoot balls at the player.

2. Enemy shooting and lockon
3. Enemy shooting and lockon
4. Check out player mass and see if that changes anything
5. FOV Slider
6. Player notified when locked on to.
7. Look at player gravity.

Able to hit enemies from under the floor.
Fix ball levitator ball position
Fix invert control settings

- Cannons that shoot balls at the player.
    - Player notified when the cannon locks on?
- Address how the player is determined as 'grounded.'
    - Playtesting revealed that many times players think they can jump since they're on the ground, but they are not technically grounded.
    - This occurs especially with jump pads.
- Camera sensitivity slider.
- Right now, x is inverted by default, y is not. Make sure y is inverted by default. Go into code and change values.
- Address quickly left clicking a ball and having it go straight left.
- Address character able to stick to the wall.
- Increase player movement.
- Look into putting the dodgeball as a child of the right arm joint.
- Add slippery physics material to the walls so the player won't stick.
- Shadow for the player so ppl know where the character will land.
- Mess with adding more weight to the ball, maybe jump pads take weight into account??
    - Doug said that the ball went really high but he didn't.
- Address angle of rotation to enemy when locked on. If angle is too high, no more lock on.
- Address air control with jump pads
    - Remove it completely since the player can jump to reset velocity??
- Address ball velocity - make sure it reaches its target with enough force.

GOAL FOR THE GAME: Kill the enemies as fast as you can. Moving targets, maybe some machines that throw balls at you.

Stretch Goals

- Animation for picking up the ball
- Settings for left handed throwing and right handed throwing.
- Lock on enemy sound.

- Crosshair graphic on top of enemy when locked on.
- Blocking mechanic - block a dodgeball with a dodgeball

Doc for Laying out how the multiplayer would work (for explanation in Capstone)