Jonathan Brewer
Matthew Gmitro
Charles Gong
Terence Kidd

Ac!d Milestone 2.1

- **Overall Design**

  We will be building off of the room idea for project one with 6 walls surrounding the player who is positioned at the back of the room. The player will control a paddle that can translate along the x and y directions and can pan and rotate. The front wall that the player is facing will be divided up into 9 equal squares that are goals. A random square will be chosen as the goal and the player must aim the paddle for the ball to hit this square. When hit, the score will increase, the ball will get a boost to combat gravity, and a new random goal will be selected. If the user misses, the score will decrease by a small amount and they will not get a velocity boost to the ball. The game ends and the game menu will show if they miss the ball and it goes behind their paddle or if the ball falls due to gravity. We will use a HUD to indicate score, a GUI for a game menu with sound settings, and sounds will be played on ball collisions and goal scoring.

  Current racquetball properties (subject to change):
  Racquetball Mass: 0.864 kg
  Coefficient of Restitution: between 0.82 and 0.85

- **Software Architecture**

  Game loop:

  Will keep track of the ball's position, reacting when the ball has made contact with the paddle, walls, or goal area, triggering sound, and updating the score and effect of gravity accordingly.

  Rendering loop:

  A camera will be centred on the paddle's position facing the back wall. Will show the current score and highlight the current goal area on the back wall and draw the updated positions of the ball and paddle.

  Simulation loop:

  Will control physics simulation and calculate the ball's position.

  Object Structure:

  We will divide up different game objects into their own classes in a hierarchy (such as the Room containing the ScoreWall and the Ball) to keep our code modular. The different classes will be Room, Wall, ScoreWall, Ball, and Paddle (subject to change).

Jonathan Brewer
Matthew Gmitro
Charles Gong
Terence Kidd

- **Division of Labor**
  Scene setup: 1 day
  Physics: 1 week
  GUI/HUD: 3 days
  Sound: 2 days
  User Input: 1 week

  We will mostly be meeting up and working on all of the parts together.
  Meeting times:
  M - W: 3pm to Undefined
  Tu - Th: 12:30pm to Undefined

  We will use github for source control and to allow us to work in parallel on different aspects of the code. Different features will be split up into different classes to avoid problems when merging code.