**Written Report**

In the initial phase of our project, we focused on creating the Python window, using all the necessary codes to configure parameters such as the size, color, screen caption, and implementing the “pygame.QUIT” event in order to close the pygame window. Then, the “pygame.draw.rect” command allowed us to draw rectangular paddles which represent Player 1 and Player 2’s rackets. We strategically positioned each player's paddle in the middle of their respective sides, with Player 1 on the left and Player 2 on the right, and programmed their movements accordingly.

As the game’s development progressed, we encountered a significant challenge: implementing an artificial intelligence system to enable single-player gameplay against the CPU. This task proved to be a complex addition for Eric and myself. Thus, we decided to designing a multiplayer experience rather than a single-player one.

Once the mobility of the rackets was fully functional, we introduced a movable ball, with its starting position defined by a variable called “center”. In order for implement collision, the “colliderect” command had to be introduced. This command allowed us to add physics to the game, such that when the ball clashes with the walls or the rackets, it will bounce opposite to how to ball was going (done by setting the value of the ball direction to negative). Innovatively, we programmed the ball to increase its speed by 0.05 times every time it collides with a paddle, and in the case that one player scores, the ball returns to its original speed.

The scoring system was another crucial aspect of our game. A point is awarded whenever the ball crosses either baseline. After scoring, the ball reappears in front of the conceding player’s racket. For gameplay simplification, the ball is served automatically instead of adding a manual mouse-click serving option. With the scoring system working, we added a scoreboard, which displays each player’s score and changes every time a goal is conceded. The game was programmed to display a winner screen for 6 seconds when a player reaches 7 goals, after which it returns the player to the main menu.

Small features were incorporated to enhance the game. For instance, a menu was added to allow players choose their preferred difficulty, each affecting the ball’s speed differently. Moreover, we included different music tracks that plays randomly whenever the game difficulty is chosen. We also added a function to the ‘M’ key, which allows the player to restart the game instantaneously. As a more complex mechanic, we introduced the 'smash shot' technique. This swift, powerful move is triggered when the ball makes contact with the top or bottom corner of the paddle. If executed incorrectly, the ball will go inside your own baseline, resulting in a point for the opponent. On the other hand, if the technique is done accurately, it results in a significant boost of the ball’s speed towards the opposing player.

Reflecting on this project, we recognize the value of advanced planning to minimize errors. Given another opportunity, we would explore the possibility of integrating an AI component. Ultimately, our objective would be to continually improve the game with more features, further enhancing the playing experience.