Software Implementation and Testing Document

For

Group <5>

Version 1.0

Authors:
Jordan Bradley
Bradley Lashley
Trevor Cooley
Jeongyeon Kim
Shashank Dontireddy

1. Programming Languages (5 points)

We used JavaScript, CSS and HTML. Our Tic-Tac-Toe used Javascript while our Flappy Planes used HTML. We used them for each of their parts because we were familiar with them and they were easier to import.

2. Platforms, APIs, Databases, and other technologies used (5 points)

In this project we used GitHub Pages to make a live website for us. Also used vscode live server extension to run locally.

3. Execution-based Functional Testing (10 points)

We have a live website, so we tested to see how our games were working or how the games were looking after updating the game logic and game background. We tried to test every case such as using both keyboard and mouse for flappy planes to see if it was functional. We checked to see if the symbol change of Tic Tac Toe was working properly and played some games to check if the scoreboard worked as intended.

4. Execution-based Non-Functional Testing (10 points)

To test the efficiency of our project we all opened the live webpage and tried cycling through the tabs to see if it was quick. We also tried accessing the webpage during different times of the day to see the reliability of the webpage during all parts of the day. We also tested the games to see if they are playable and work as intended. Play tested the game as well as stress tested it by playing the game with a lot of other chrome tabs open which utilized much of the computers RAM to see if the game was still efficient under suboptimal conditions.

5. Non-Execution-based Testing (10 points)

For non-execution-based testing, before any code was merged into main, we would make a pull request and from there, the new code to be merged would be reviewed and tested by another group member other than the author of the code before it was completely merged into the main.