# Software Requirements and Design Document

# For

# Group <5>

Version 1.0

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

## 1. Overview (5 points)

*Our system will be based around a website that has 3 main pages and a navigation bar at the top. One page being the central hub or "home" page with options that allow you to navigate to the 2 other pages. One page that holds one of our games being either tic tac toe or flappy plane. The other page will hold the other game. Both pages will have the ability/option to return back to the home page.*

*Give a general overview of the system in 1-2 paragraphs (similar to the one in the project proposal).*

## 2. Functional Requirements (10 points)

*List the **functional requirements** in sentences identified by numbers and for each requirement state if it is of high, medium, or low priority. Each functional requirement is something that the system shall do. Include all the details required such that there can be no misinterpretations of the requirements when read. Be very specific about what the system needs to do (not how, just what). You may provide a brief design rationale for any requirement which you feel requires explanation for how and/or why the requirement was derived.*

1. *Needs to be able to navigate between the different web pages. (HIGH)*
2. *(tic tac toe page) Needs to be able to play a game of tic tac toe. (HIGH)*
3. *(tic tac toe page) Needs to be able to choose different symbols for X and O (MEDIUM)*
4. *(tic tac toe page) Needs to be able to display the winner's score. (LOW)*
5. *(tic tac toe page) Needs to be able to display the number of wins. (LOW)*
6. *(tic tac toe page) Create a scoreboard to show the player's victory or defeat. (HIGH)*
7. *(tic tac toe page) Apply the changed symbol to the scoreboard. (MEDIUM)*
8. *(flappy plane page) Needs to be able to play a game of flappy plane. (HIGH)*
9. *(flappy plane page) Need to be able to choose between two play methods: mouse and keyboard. (HIGH)*
10. *(flappy plane page) Create a scoreboard and record the player's name and time. (MEDIUM)*
11. *(flappy plane page) Need to be able to rename a player. (LOW)*
12. *(flappy plane page) Need to be able to choose between restart, main menu, and scoreboard. (MEDIUM)*

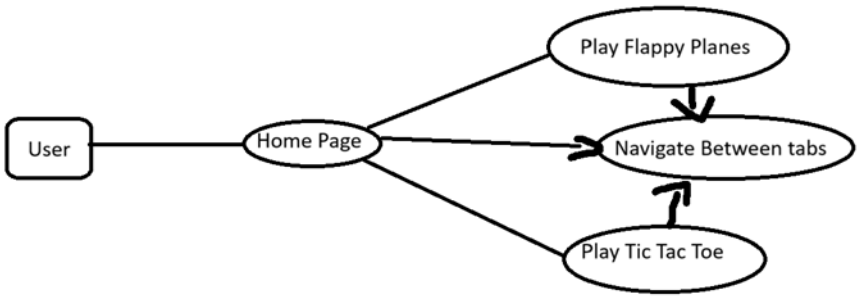
## 3. Non-functional Requirements (10 points)

The system is secure since it is served by HTTPS. The software is reliant on GitHub Pages. The transition between tabs is smooth. The quality of the webpage is basic and there is not much detail. The software is reliant on GitHub pages since that is the only free option, we had to host a live webpage. The games run smoothly and there are no problems with crashes.

## 4. Use Case Diagram (10 points)

*This section presents the **use case diagram** and the **textual descriptions** of the use cases for the system under development. The use case diagram should contain all the use cases and relationships between them needed to describe the functionality to be developed. If you discover new use cases between two increments, update the diagram for your future increments.*

***Textual descriptions of use cases**: For the first increment, the textual descriptions for the use cases are not required. However, the textual descriptions for all use cases discovered for your system are required for the second and third iterations.*




| Name | Load Homepage |
|---|---|
| Actor | User |
| Pre-condition | The User goes to the website url |
| Post Condition | The homepage is loaded |
| Maine Sequence Path | Actor: Goes to website<br>System: Loads homepage |
| Alternative Path | None |
| Alternative Path | None |
| Exception Path | User does not have internet, or GitHub is down will run into an error |
| Scenario | Actor: User opens website<br>System: Displays home page |

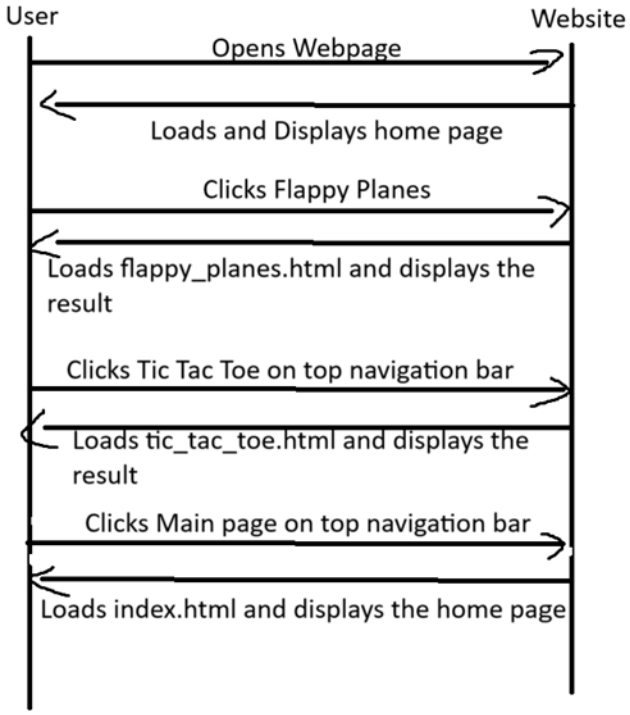
| Name | Navigate between Tabs |
|---|---|
| Actor | User |
| Pre-condition | The User loaded onto one of the pages on the website |
| Post Condition | The new tab clicked is loaded |
| Maine Sequence Path | Actor: Clicks the tab at top or image on home page<br>System: Loads the tab that was clicked |
| Alternative Path | None |
| Alternative Path | None |
| Exception Path | User does not have internet, or GitHub is down will run into an error |
| Scenario | Actor: User clicks flappy planes at the top<br>System: Loads and displays flappy planes |

| | |
|---|---|
| Name | Load and Play Flappy planes |
| Actor | User |
| Pre-condition | The User loaded onto one of the pages on the website and clicks flappy planes at the top navigation bar or on home page. |
| Post Condition | The Flappy planes page is loaded, and the game logic is loaded and playable |
| Maine Sequence Path | Actor: Clicks the flappy planes at the top or on home page<br>System: Loads the flappy planes html file which then loads the game logic on game.js |
| Alternative Path | None |
| Alternative Path | None |
| Exception Path | User does not have internet, or GitHub is down will run into an error |
| Scenario | Actor: User clicks flappy planes at the top<br>System: Loads and displays flappy planes and its game logic |

| | |
|---|---|
| Name | Load and Play Tic Tac Toe |
| Actor | User |
| Pre-condition | The User loaded onto one of the pages on the website and clicks Tic Tac Toe at the top navigation bar or on home page. |
| Post Condition | The Tic Tac Toe page is loaded, and the game logic is loaded and is semi playable |
| Maine Sequence Path | Actor: Clicks the Tic Tac Toe at the top or on home page<br>System: Loads the Tic Tac Toe html file which then loads the game logic on Tic Tac Toe script.js |
| Alternative Path | None |
| Alternative Path | None |
| Exception Path | User does not have internet, or GitHub is down will run into an error |
| Scenario | Actor: User clicks Tic Tac Toe at the top |

| | |
|---|---|
| | System: Loads and displays Tic Tac Toe and its game logic |

## 5. Class Diagram and/or Sequence Diagrams (15 points)




## 6. Operating Environment (5 points)

This software will operate in any web browser such as Google Chrome or Safari and will work on Windows, MacOs, and linux. The website will also be served over HTTPS which is a secure connection and protects user data. The software is reliant on GitHub Pages if there is a problem with GitHub pages the software will not work. The software can also be ran locally through the use of vscode extension live server/five server.

## 7. Assumptions and Dependencies (5 points)

*List any assumed factors (as opposed to known facts) that could affect the requirements stated in this document. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.*

There could be a miss fired code that could corrupt the whole thing. The options for the tic tac toe could glitch out and cause the check win system to bug out makes it impossible to win or draw. Some web browsers could have different software issues and problems when running it. Playability could be affected by individual screen dimensions such as a

phone having a worse view of the game they're playing in the browser than a laptop/computer. The flappy planes implementation is made for computer display and is not playable on mobile. The games cannot work on some computers but work on others. Another issue is optimization of the game so that even if someone is using a lot of their RAM on other programs or tabs the game might not run smoothly.