Software Engineering Project:   Gaming Platform—Five in Row Reactive game

# Description:

The goal of our project was to build a real-time gaming platform. The goal of our project was to build a real-time gaming platform. On this platform, users can create different rooms, enter the room to start the battle game, or enter the room to watch the battle of other users. The first version was designed to be a Gomoku game (five in a row). The system supports maximum ten rooms at the same time, with two opponents in each room and up to five visitors in each room. Users can also chat in the game room.Different game categories will be added to the game platform in the future.

# Wireframes



# Feature List

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | | Feature | | | Owner | priority |
|  | Platform | 1. Display the welcome page of the gaming platform, introduce the rules of the game, and allow users to select the game to enter | | |  | P0 |
| 2.Users can create a room by entering a user name first. The game page is displayed after the room is created | | |  | P0 |
| 3 Show existing game rooms on the platform. Users can choose to enter the existing room. | | |  | P1 |
| 4. Allow users to visit existing rooms. Users entering the room can choose to play or just visit. | | |  | P2 |
|  | 5. Users can have real-time conversations in the room. | | |  | P2 |
| Game | 1. The user can place pieces anywhere on the board where there are no pieces. | | |  | P0 |
| 2 .The user must place the pieces in order and wait for the player to complete the move after other player performs the move. | | |  | P0 |
| 3. When the winning condition is reached, the system will give the winning prompt and end the game. | | |  | P1 |
| 4. Allows the user to take a step back | | |  | P2 |
| 5. Realize the man-machine battle of single user | | |  | P3 |
| 6.add more game into platform like black jack , chees etc. | | |  | P3 |
| 7. | | |  | P3 |
| 8. | | |  |  |
| 9. | | |  | P3 |
|  |  |  |  |  |  |  |

# Technical Considerations

* Node js
* Express
* Socket.io
* Angular

# Work Distribution

Same as the owner of the feature list