A Project Report On

Tic Tac Toe - Multiplayer

Submitted for the CS5102 Project in NERIST

By

Mritunjay Saha (D/18/CS/002)

Under the guidance of

Mr. Yogendra Mohan

(Asst. Professor, Dept. of Computer Science and Engineering)

Department of Computer Science and Engineering

North Eastern Regional Institute of Science & Technology

(Deemed to be University)

Nirjuli-791109, Arunachal Pradesh

Table of Contents

List of figures

I. Introduction

Tic Tac Toe - Multiplayer is a web application that will allow two users to play against each other instead of playing against a computer. The users will be playing in a virtual room. The virtual room will be created by one user and a game code will be generated. The first user has to share game code with the second user. The second user will have to join using the game code. Once both the users are in the same room they will be able to play the game.

II. Why is this particular project chosen?

I've chosen this particular project because I was looking for ways to improve the Tic Tac Toe game. It only allowed the option to play against the computer. And 90% of the time the user will win the game. I wanted to make a Tic Tac Toe game that will allow two humans to play against each other. At the same time, I wanted to understand how to get two devices running a web application to communicate with each other.

III. Objective and Scope

Implement a web application that will allow two users to play against each other at a time. We will implement this using web sockets. The web sockets will enable us to communicate with the server. The server will act as the common point of information exchange. The current moves of the user-1 will be sent to the server. The server will receive this information and display the move to user-2. Similarly, the moves of user-2 will be shown to user-1.

We will create rooms for the users. The other user can join the game using the game code. The server will keep track of the moves of both the users to check the status of the game, whether there is a winner or the game ended in a draw.

IV. Process Description

When the first user starts the game, an event is sent to the server. This event calls the function to generate the game code. The first user is added to a room using the game code. This game code is sent back to the frontend. The second user has to use this game code to join the game. When the second user uses this game code to join the game then he is added to the same room as that of the first user.

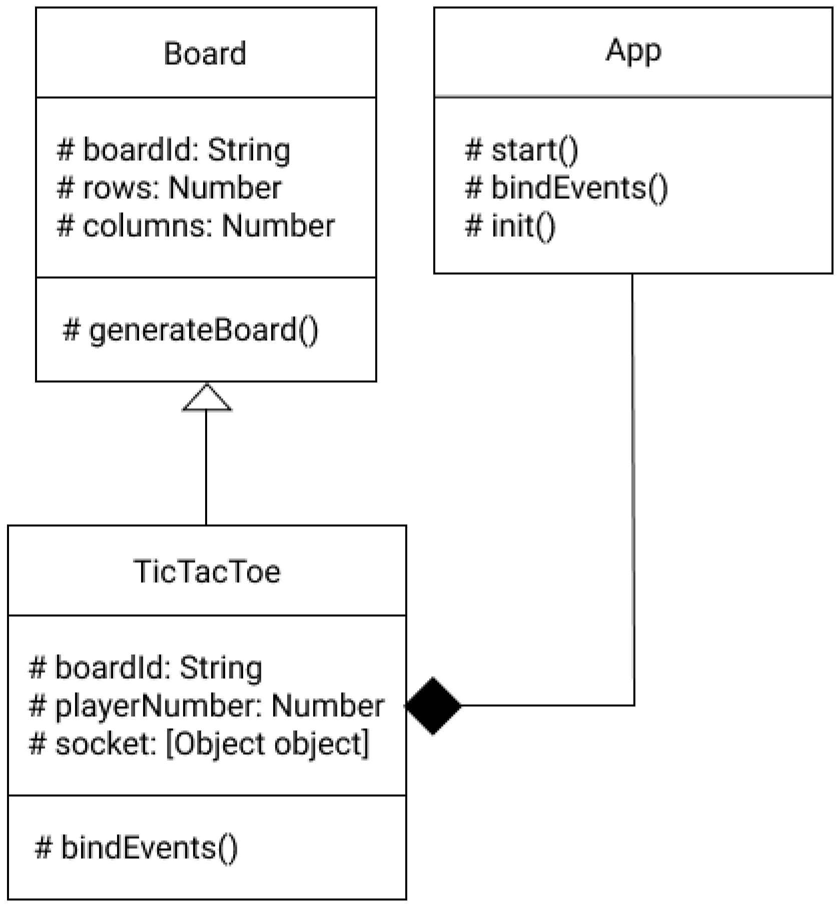

Fig1. Class Diagram of the frontend portion

After both the users have joined the same room they will be able to play the game. When a user clicks on a cell, the value of that cell will be sent to the server. The server will keep track of the moves. After every move, the server will check if there is a winner, or the game ends in a draw. Depending on the result of the game an event will be sent to the frontend which will notify users the result of the game.

User-1 selects a cell, the cell is sent to the server. The server then updates the selected cell of user-1 in user-2's device. Similarly, user-2's move is shown in user-1's device.

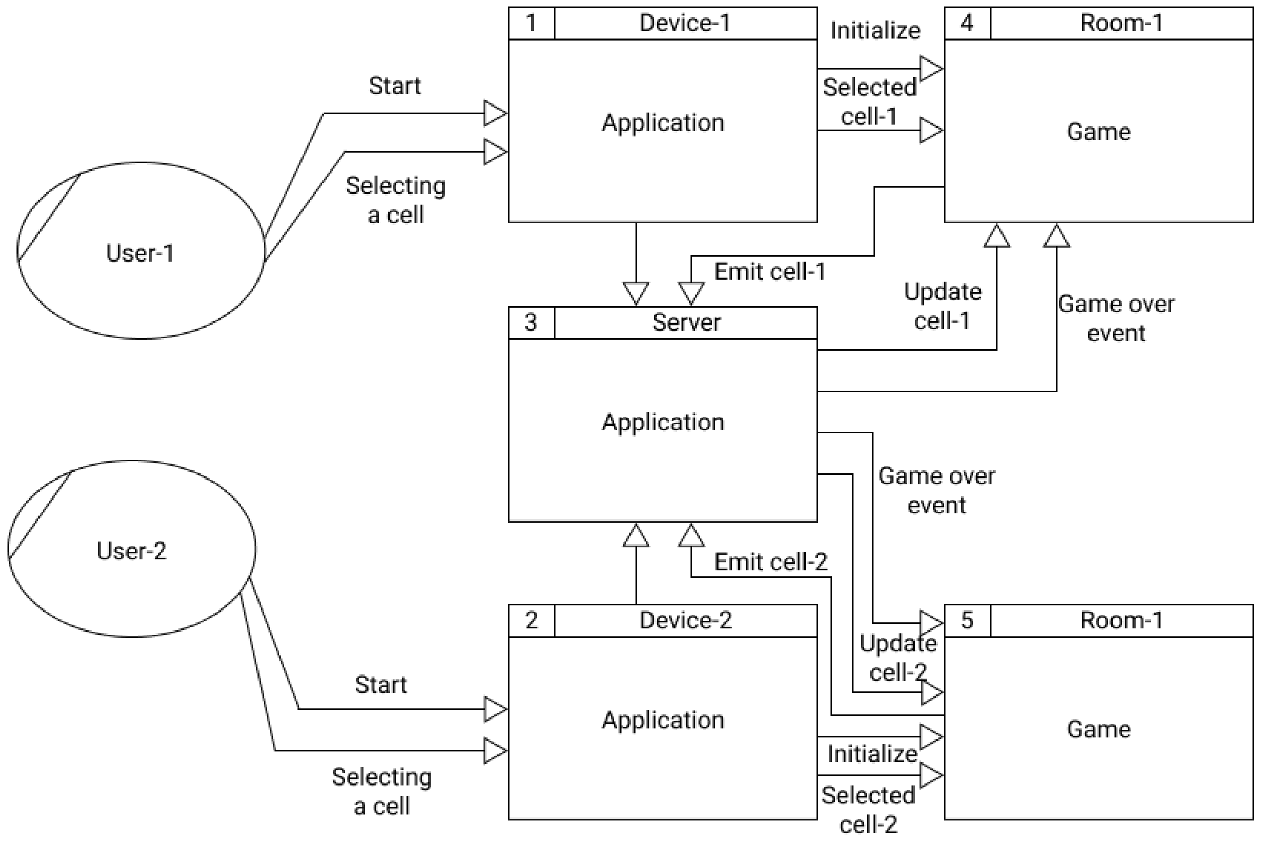

Fig 2: Data flow diagram

V. Resources and Limitations

The tools/software required to fulfil this project can be divided into two sections:

1. Frontend.
2. Backend.
3. Module bundlers.
4. Hosting Services.

Frontend: For the frontend, we will use JavaScript, Sass, HTML and Socket.io.
Backend: For the backend, we will use JavaScript, Node.js, Socket.io and express.js
Module bundlers: Webpack.
Hosting Services: The frontend will be hosted on Netlify and the backend will be hosted on Heroku.

VI. Conclusion

The home screen of the web application will give the user to create a new game or join a game. To join a game the user needs to use a game code to join a specific room. To create a new room the user can create a new game.

The application can be extended to have a login/signup page and a database. If we have a database then we keep track of the number of games played and the result of those games.