

Battleship Game - Angular Application

Overview

This project is an Angular-based implementation of the classic Battleship game. It allows players to play against an AI or another player in a turn-based strategy game. The application features drag-and-drop ship placement, visual feedback for hits and misses, and a simple AI for single-player mode.

Features

- **Game Modes:**
 - Player vs AI
 - Player vs Player
- **Drag-and-Drop Ship Placement:** Players can drag ships onto their board to set up their fleet.
- **Turn-Based Gameplay:** Players take turns attacking each other's boards.
- **Visual Feedback:** Hits and misses are displayed on the board.
- **AI Opponent:** A basic AI randomly selects cells to attack.

How to Play

Start the Game

1. Choose the game mode:
 - **Player vs Player:** Click the "Start Player vs Player" button.
 - **Player vs AI:** Click the "Start Player vs AI" button.
2. For AI mode, the opponent's board is automatically populated with ships.

Place Ships

1. Drag and drop ships from the ship container onto your board.
2. The application validates the placement to ensure ships do not overlap or go out of bounds.

Take Turns

1. **Player's Turn:**
 - Click on a cell in the opponent's board to attack.
 - Hits are marked with an 'X', and misses are left blank.
2. **AI's Turn (Player vs AI):**
 - The AI randomly selects a cell on the player's board to attack.
3. **Opponent's Turn (Player vs Player):**
 - The second player clicks on a cell in the first player's board to attack.

Winning the Game

- The game ends when all ships on one board are hit.
- The winner is declared based on whose ships remain.

Reset the Game

- Click the "Reset Game" button to clear the boards and start over.

Project Structure

The project is organized as follows:

```
angular.json
karma.conf.js
package.json
README.md
src/
  app/
    app.component.ts
    app.component.html
    app.component.scss
    game-board/
      game-board.component.ts
      game-board.component.html
      game-board.component.scss
  services/
    board.service.ts
  environments/
    environment.ts
    environment.prod.ts
```

Key Files

- **app.component.ts**: Main component handling game logic and state.
- **game-board.component.ts**: Component for rendering the game board and handling user interactions.
- **board.service.ts**: Service for managing board state, ship placement, and validation.
- **environment.ts**: Configuration for development and production environments.

Development

Prerequisites

- Node.js
- Angular CLI

Setup

1. Clone the repository.
2. Install dependencies:

```
npm install
```

Run the Application

Start the development server:

```
ng serve
```

Access the application at <http://localhost:4200> (<http://localhost:4200>).

Run Tests

Run unit tests using Karma:

```
ng test
```

Future Enhancements

- Add difficulty levels for AI.
- Implement a leaderboard for Player vs Player mode.
- Enhance UI/UX with animations and sound effects.

License

This project is licensed under the MIT License.