Writeup

## What were some challenges you faced while making this app?

The primary challenges included managing real-time state synchronization between player and AI boards to ensure turn consistency, which required careful handling of game logic and event flows. Implementing drag-and-drop functionality proved tricky due to ship rotation mechanics and collision detection, demanding precise grid alignment calculations and DOM updates. Designing the AI’s targeting system required balancing randomness with fairness to avoid repetitive patterns, while responsive styling demanded meticulous adjustments to maintain usability across mobile and desktop screens.

## Given more time, what additional features, functional or design changes would you make?

With additional time, I’d introduce an interactive tutorial to onboard new players unfamiliar with Battleship mechanics. Visually, 3D effects for sinking ships and smoother animations would enhance feedback. Functionally, I’d expand the AI’s strategy beyond random targeting (e.g., implementing a "hunt" mode) and add local storage for game persistence.

##What assumptions did you make while working on this assignment?

The project assumed players already understood Battleship rules, avoiding redundant tutorial work. The AI’s simplicity—relying on random valid attacks—was deemed sufficient for the scope, prioritizing basic functionality over advanced behavioral logic. Styling focused on clean, functional layouts rather than elaborate visuals, and local storage was omitted unless explicitly required, as the MVP emphasized real-time play over persistence.

## How long did this assignment take to complete?

This assignment took approximately **30 hours**, spanning core logic implementation (ship placement, turn systems, AI), UI/UX polish (drag-and-drop, responsive grids), and iterative debugging to resolve state mismatches and edge cases.