# Minesweeper Project Writeup

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

Making the first click always safe turned out to be surprisingly challenging. I had to completely rethink how the board generates mines to ensure the player never hits a mine on their first move while keeping the game random and fair. Managing the game state with Context API was also tricky, especially when implementing the recursive reveal of empty cells. The flag system needed careful consideration to handle both right-clicks and shift-clicks correctly. Another challenge was ensuring cell states updated properly without causing unnecessary re-renders, particularly during the cascade effect of revealing multiple empty cells.

## Given more time, what additional features would you make?

I would love to add a timer to track game duration and potentially save high scores locally. A game save feature using localStorage would be useful for longer games at higher difficulties. Different visual themes would make the game more personalized - perhaps including a dark mode and some colorful variations. The mobile experience could definitely be improved with better touch controls and responsive design. I'd also add some subtle animations for revealing cells and explosions to make the game feel more dynamic. For new players, a hint system could help them learn strategies without getting frustrated.

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

I assumed players would be familiar with basic Minesweeper rules and would have access to both mouse and keyboard controls. The game was primarily designed for desktop use with modern browsers, though it works on mobile too. I focused on functionality over performance optimization since the game scale wasn't too demanding. I also assumed users would prefer clear, simple visuals over elaborate graphics, keeping with the classic Minesweeper style.

## How long did this assignment take to complete?

The project took approximately 12 hours to complete. Initial setup and board implementation took about 3 hours. The core game logic, including mine placement and cell revelation, required 4 hours to get right. UI design and styling took 2 hours. The final 3 hours were spent on bug fixes, testing, and polishing the user experience. The safe-first-click feature required several iterations to implement correctly.

## Feedback on this challenge

I really enjoyed working on this project as it provided great practice with React Context and state management. The requirements were clear while still leaving room for creative solutions. Implementing a classic game was fun and engaging, offering a good mix of UI and logic challenges. The project helped strengthen my understanding of React's capabilities.

Some suggestions for improvement: it would be helpful to have performance benchmarks as targets. Additional optional features for extra credit could make the project more challenging for advanced students. Some guidance on testing strategies would also be valuable, especially for the recursive elements of the game.

Overall, it was a challenging but rewarding project that helped improve my React skills while creating something entertaining.