

# Brushed Steel Consulting

Carlos Valcarcel and ChatGPT

## Bomb Grid Game Requirements Document

### Overview

This document outlines the functional requirements for a web-based implementation of the classic Minesweeper game. The game will be HTML-based with a JavaScript backend.

### Game Mechanics and Behavior

- **Grid Initialization**
  - The game starts with a grid of covered cells.
  - Grid sizes can vary based on difficulty level (e.g., 9x9, 16x16).
- **Mine Placement**
  - A predetermined number of mines are randomly placed on the grid.
  - No mine will be placed on the first clicked cell.
- **Gameplay**
  - Left-clicking a cell uncovers it.
  - Right-clicking a cell toggles a flag on it to mark suspected mines.
  - The first click in any game will never uncover a mine.
- **Cell States**
  - Each cell can be in one of three states: covered, uncovered, or flagged.
  - Uncovered cells display either a number (indicating the count of adjacent mines) or a mine.
- **Uncovering Cells**
  - Uncovering a mine ends the game.
  - Uncovering a cell without adjacent mines automatically uncovers adjacent cells recursively.
  - Uncovering a cell with adjacent mines displays the number of adjacent mines.
- **Flagging Mines**
  - Players can flag cells they suspect contain mines.
  - Flagged cells cannot be uncovered until they are unflagged.
  - The game displays a count of remaining unflagged mines.
- **Winning the Game**
  - The game is won when all non-mine cells are uncovered.
- **Losing the Game**
  - The game is lost if a mine is uncovered.
  - All mines are revealed when the game is lost.
- **Timer**
  - A timer starts when the first cell is clicked and stops when the game ends.

- **Restarting the Game**
  - Players can start a new game at any time.
  - Restarting the game resets the grid, mines, and timer.
- **Difficulty Levels**
  - The game offers different difficulty levels, affecting the grid size and number of mines.

## User Interface

- **Grid Display**
  - The grid is centrally displayed on the page.
  - Cells are visually distinct and indicate their state clearly (covered, uncovered, flagged).
- **Controls**
  - A reset button allows the player to start a new game.
  - Difficulty level can be selected from a dropdown menu or buttons.
- **Indicators**
  - A mine counter displays the number of mines yet to be flagged.
  - A timer displays the elapsed time since the start of the game.
- **End of Game**
  - Messages or visual cues indicate winning or losing the game.
  - Option to restart the game upon completion.

## Technical Requirements

- **Responsive Design**
  - The game should be playable on various devices and screen sizes.
- **Cross-Browser Compatibility**
  - The game should function correctly on major web browsers.
- **Accessibility**
  - The game should be accessible, including keyboard navigability and screen reader compatibility.

## Testing and Validation

- **Unit Testing**
  - Key functions (grid initialization, mine placement, cell uncovering, flagging) should be unit tested.
- **User Testing**
  - The game should be tested for usability and user experience across different devices and browsers.
- **Performance**
  - The game should perform smoothly without significant delays or resource usage.

# Maintenance and Support

- **Documentation**
  - Code should be well-documented for future maintenance and updates.
- **Error Handling**
  - The game should handle errors gracefully and provide user feedback where appropriate.
- **Updates and Patches**
  - The game should be designed to allow easy updates and bug fixes.