Project Report

FRONTEND DEVELOPMENT INTERSHIP

NAME – MINI GUPTA
INTERNSHIP DURATION – 3 MONTH
COMPANY NAME – UNIFIED MENTOR

Tic-Tac-Toe Game

Project Details

- Project Type: Frontend Development
- Contribution: Individual
- Frontend Technologies: HTML, CSS, JavaScript
- Development IDE: Visual Studio Code(VS Code)

Project Summary

- Developed a fully interactive, browser-based Tic-Tac-Toe game implementing core game logic, responsive design, and dynamic user interaction using HTML5, CSS3, and vanilla JavaScript.
- Two-player (X/O) turn-based interaction with visual turn indicators.
- Draw/tie detection with appropriate messaging.
- Clean, intuitive 3×3 game grid with consistent visual hierarchy.
- Event-driven architecture with efficient DOM manipulation.
- Modular JavaScript code separating game logic, UI updates, and event handling

  CSS animations for player moves and game transitions.

Project Objective

This project involves the design and development of a fully responsive, browser-based Tic-Tac-Toe game to provide a polished and intuitive gaming experience. The core objective is to implement a robust, user-centric interface that ensures seamless interaction across all modern devices and screen sizes. By leveraging fundamental web technologies—HTML5 for structure, CSS3 for responsive styling and visual feedback, and vanilla JavaScript for dynamic behavior—the application will deliver a classic two-player experience with modern interactive elements.

Technical Architecture

- **Frontend Structure:** Semantic HTML5 for accessible game board layout and interface components
- **Styling & UI:** CSS3 with Flexbox/Grid for responsive design, hover effects, and smooth visual transitions
- **Game Logic:** Pure JavaScript implementing player turns, win-condition validation, and game state management

Tools Used

- **Core:** HTML5, CSS3, JavaScript
- **Version Control:** Git, GitHub
- **Deployment/Hosting:** GitHub
- **Code Editor:** Visual Studio Code(VS Code)

Development & Coding

- Wrote clean, semantic HTML5 for structure.
- Styled with modular, responsive CSS3 using Flexbox .
- Implemented all logic and interactivity using vanilla JavaScript (ES6).

Technical Deep Dive: The Core Stack

- HTML5: The foundation. I focused on using semantic tags like <nav> ,<main> ,<section> , and <article> to improve SEO and accessibility.
- CSS3: The visual design. I heavily utilized modern layout modules like 'Flexbox' and 'CSS Grid' to build responsive designs, along with transitions and animations for a premium user experience.
- JavaScript (ES6): The engine. I used it to bring the static pages to life through 'DOM Manipulation, 'Event Handling, and using the ; Local Storage API' for data persistence

Working

- **Initialization:** A 3×3 game board loads with all cells empty
- **Player Turns:** Players alternate clicking cells (X starts, then O)
- **Move Validation:** Each click places a mark if the cell is empty
- **Win Check:** After each move, the game checks for three-in-a-row
- **Game End:** Ends when a player wins or all cells are filled (draw)
- **Reset:** Players can restart for a new game

Output

- **3×3 Game Grid**: Clear, styled cells display X/O markers
- **Turn Indicator**: Shows "Player X's Turn" or "Player O's Turn"
- **Winning Highlight**: Winning line visually emphasized (color/animation)
- **Status Display**: Announces winner ("X Wins!"/"O Wins!") or "Game Draw!"
- **Interactive Controls**: Reset button to restart the game
- **Valid Move**: Clicked cell shows X/O symbol
- **Game End**: Disables further moves after win/draw
- **Instant Reset**: Board clears completely for new game

GitHub Link: https://github.com/mg-0521/Third-Repo

Thank You & Q/A

Thank you for your time and attention.

I am eager to apply the skills and passion I've developed during this internship to a full-time role.

Open to Questions*