

Web Based Minesweeper Game

UNDERGRADUATE PROJECT

Submitted in partial fulfillment of the requirements of software
development project II for the degree of
B.Sc. Eng. in CSE

UNDER SUPERVISION OF:

Md. Mahbubur Rahman

Assistant Professor, Dept. of CSE

Bangladesh University of Business & Technology



**Bangladesh University of Business &
Technology**

Dhaka-1216, 09 June 2024

Table of Contents

Group Member List:	3
Declaration of Authorship.....	4
Certificate.....	5
Dedication	6
Abstract	7
Acknowledgments.....	8
Chapter 1: Introduction	9
1.1 Problem Statement	9
1.2 Project Aim & Objectives	9
1.3 Motivations	9
Chapter 2: Literature Review	11
2.1 Modern Minesweeper Implementations.....	11
2.2 User Account & High Score Management.....	11
Chapter 3: Methodologies	12
3.1 Agile Development	12
3.2 User-Centered Design	12
3.2 Continues Integration/Continuous Deployment (CI/CD)	12
Chapter 4: Technical Stack Overview	14
4.1 Frontend	14
4.2 Backend.....	14
4.3 Database.....	14
4.4 Libraries & Tools	14
4.5 Functional Requirements	14
4.6 Non-Functional Requirements	15
4.7 Design & User Interface	15
4.8 API Design.....	17
4.9 Database Design.....	17

4.10 Game Mechanics.....	18
4.11 High Score System.....	19
4.12 User Account Management.....	19
Chapter 5: Future Enhancements	20
Chapter 6: Conclusion	21

Group Member List:

1. Asifur Rahman
ID:22234103353
Intake: 50 Section: 07 Semester: Spring,2024

2. Ferdouse Hassan Nowrin
ID:22234103237
Intake: 50 Section: 07 Semester: Spring,2024

3. Tamanna Hossain Badhon
ID:22234103253
Intake: 50 Section: 07 Semester: Spring,2024

4. Sourav Baidda
ID:22234103265
Intake: 50 Section: 07 Semester: Spring,2024

Declaration of Authorship

From our team, We, Asifur Rahman, Ferdouse Hassan Nowrin, Tamanna Hossain Badhon, and Sourav Baidida announced that this project titled, “Web-based Minesweeper Game” and the work presented in it are our own.

We, hereby declare that this submission is entirely our work, in our own words, and that all sources used in researching it are absolutely acknowledged and all quotations acknowledged and all quotations properly identified. We are alert that this internet-based project of ours’ published in digital form can be favorable for everyone through using the internet. It has not been submitted in whole by us to obtain any other credit or grade. We understand the ethical implications of our research and this work meets the requirements of the Faculty of Computer Science Engineering.

Signature of Developers

Asifur Rahman (ID:22234103353)

Ferdouse Hassan Nowrin (ID:22234103237)

Tamanna Hossain Badhon (ID:22234103253)

Sourav Baidida (ID:22234103265)

Certificate

This is to certify that the project called, "Web Based Minesweeper Game" and submitted by Asifur Rahman(ID:22234103353), Ferdouse Hassan Nowrin(ID:22234103253), Tamanna Hossain Badhon(ID: 22234103253) and Sourav Baidia (ID:22234103265) in partial fulfillment of the requirement of form the work done by them under my supervision.

Md. Mahbubur Rahman

Assistant Professor Department of CSE

Bangladesh University of Business and Technology (BUBT)

Dedication

Dedicated to our esteemed former faculty of CSE 121, **Khan Md. Hasib**, in recognition of his engaging lectures, unwavering support, warm friendliness, and exceptional mentorship left a lasting impact on us all.

We Miss You, Sir.

Abstract

The Web-Based Minesweeper Game Project aims to recreate the classic Minesweeper experience as a responsive web application using modern web development technologies. This comprehensive report describes the motivations, objectives, system requirements, and technical stack used to develop the game. The project employs a modular architecture with React for the frontend, Node.js for the backend, and PostgreSQL as the database. The implementation details cover gameplay mechanics, high-score systems, and user account management. By leveraging RESTful APIs, the backend enables seamless gameplay analytics and statistics tracking. This report concludes with potential future enhancements, including new difficulty levels, achievements, and a multiplayer mode.

Acknowledgments

First of all, we are thankful and expressing our gratefulness to Almighty Allah who offers us His divine blessing, patience, mental and physical strength to complete this project work. We are deeply indebted to our project supervisor, **MD Mahbubur Rahman**, Assistant Professor, Department of Computer Science and Engineering (CSE), Bangladesh University of Business and Technology (BUBT). His scholarly guidance, important suggestions, work for going through our drafts and correcting them, and generating courage from the beginning to the end of the project work has made the completion of this report possible. A very special gratitude goes out to all our friends for their support and help to implement our work. The discussions with them on various topics of our works have been very helpful for us to enrich our knowledge and conception regarding the work. Last but not least; we are highly grateful to our parents and family members for supporting us spiritually throughout writing this report and our life in general.

Chapter 1: Introduction

Minesweeper is a classic logic puzzle game that challenges players to reveal safe cells without triggering hidden mines. This report details the development of a web-based version of Minesweeper. The project integrates a seamless user experience with robust backend services for user account management, high-score tracking, and gameplay statistics.

1.1 Problem Statement

While Minesweeper is a widely recognized game, most implementations are desktop-based, lacking a modern, accessible, web-based version that provides seamless gameplay across devices. Moreover, current versions often lack comprehensive player analytics, high score tracking, and an engaging user interface.

1.2 Project Aim & Objectives

Aim: To develop a fully functional web-based Minesweeper game with user account management, high score tracking, and a modern, responsive interface.

Objectives:

- Create a user-friendly web-based version of Minesweeper.
- Implement user account management with registration and login features.
- Track and display high scores for different difficulty levels.
- Provide comprehensive gameplay statistics for individual users.
- Ensure responsiveness and compatibility across devices.

1.3 Motivations

1. **Nostalgia:** Minesweeper holds a nostalgic value for many users who wish to relive the classic experience.
2. **Modern Technology:** Utilize modern web technologies like React and Node.js to create an updated version.
3. **Learning Opportunity:** This project provides an opportunity to apply frontend and backend development skills in a comprehensive, real-world application.

4. **Analytics & Personalization:** Adding user analytics and high scores enhances player engagement and motivation.

Chapter 2: Literature Review

The original Minesweeper game, introduced by Microsoft in the early 1990s, was designed to teach users how to effectively use the right and left mouse buttons. The core gameplay involves revealing safe cells and flagging mines based on numerical clues.

2.1 Modern Minesweeper Implementations

1. **Windows Minesweeper:** The classic game, bundled with Windows operating systems, has been improved over time with better graphics and more levels.
2. **Minesweeper Clone (Java):** Open-source Java versions provide similar gameplay with different UI implementations.
3. **Minesweeper X:** A popular third-party clone for Windows, adding customizable board sizes and themes

2.2 User Account & High Score Management

User account systems have become integral to modern gaming, allowing players to save progress and compete globally. High score systems provide competitive elements, encouraging users to improve their performance.

Chapter 3: Methodologies

In developing the web-based Minesweeper game, several methodologies were employed to ensure a structured, efficient, and high-quality project delivery. These methodologies include Agile development practices, user-centered design, and continuous integration/continuous deployment (CI/CD).

3.1 Agile Development

We adopted an Agile methodology to manage the project's development lifecycle. This approach facilitated iterative progress, allowing for flexibility and rapid response to changing requirements. Key Agile practices included:

- **Sprint Planning:** The project was divided into multiple sprints, each lasting two weeks. During sprint planning meetings, tasks were prioritized and assigned to team members based on their skills and availability.
- **Daily Standups:** Short daily meetings were held to discuss progress, identify roadblocks, and coordinate efforts. This ensured team alignment and prompt issue resolution.
- **Sprint Reviews and Retrospectives:** At the end of each sprint, we conducted reviews to demonstrate completed work and gather feedback. Retrospectives were held to reflect on the sprint process and identify areas for improvement.

3.2 User-Centered Design

To create a user-friendly and engaging Minesweeper game, we focused on user-centered design principles. This involved:

- **User Research:** Surveys and interviews were conducted to understand user preferences and expectations for the Minesweeper game. This information guided the design and development process.
- **Prototyping and Usability Testing:** Wireframes and prototypes were created using Figma and Wireframe.cc. Usability testing sessions with potential users provided insights into the interface's effectiveness, leading to iterative improvements.
- **Accessibility Considerations:** Ensuring the game was accessible to a broad audience, we incorporated features like adjustable difficulty levels and clear visual indicators.

3.2 Continues Integration/Continuous Deployment (CI/CD)

To maintain code quality and streamline the deployment process, we implemented a CI/CD pipeline. This included:

- **Version Control:** Using Git for version control, we managed code changes efficiently and facilitated collaboration among team members.
- **Automated Testing:** Unit and integration tests were written using Jest and React Testing Library. These tests were automatically run on each code commit to detect and address issues early.
- **Automated Deployment:** A CI/CD tool (e.g., GitHub Actions) was configured to automate the build and deployment process. This ensured that any code changes were quickly and reliably deployed to the staging and production environments.

By employing these methodologies, we were able to deliver a robust, user-friendly, and scalable web-based Minesweeper game within the project timeline. These practices not only enhanced our development process but also ensured that the final product met user needs and maintained high-quality standards.

Chapter 4: Technical Stack Overview

4.1 Frontend

- **HTML5/CSS3/JavaScript:** For basic structure, styling, and scripting.
- **React:** Component-based JavaScript library for building the UI.
- **Material-UI:** UI component library for styling and theming.
- **Figma & Wireframe.cc:** Design tools for UI/UX wireframing.

4.2 Backend

- **Node.js:** JavaScript runtime for server-side development.
- **Express.js:** Web framework for creating RESTful APIs.
- **Knex.js & Objection.js:** SQL query builder and ORM for database interactions.

4.3 Database

- **PostgreSQL:** Relational database for storing user data and scores.

4.4 Libraries & Tools

- **Twemoji:** Emoji rendering library for a richer UI experience.
- **Redux:** State management library for efficient data flow.
- **Jest & React Testing Library:** Testing frameworks for robust unit and integration testing.
- **Git:** Version Control

4.5 Functional Requirements

1. **Game Mechanics:**
 - Players should be able to reveal cells and flag suspected mines.
 - Game mechanics should mirror the original Minesweeper game.
2. **User Accounts:**

- Users can register, log in, and manage their accounts.
3. **High Score Tracking:**
 - Track high scores across different difficulty levels.
 4. **Game Analytics:**
 - Track and display individual player statistics, including average and fastest completion times.

4.6 Non-Functional Requirements

1. **Performance:**
 - Responsive UI and efficient API responses.
2. **Security:**
 - Secure user authentication using JWT tokens.
 - Passwords should be hashed using bcrypt.
3. **Scalability:**
 - Support multiple simultaneous users and large data sets.
4. **Compatibility:**
 - Cross-browser and cross-device compatibility.

4.7 Design & User Interface

The design focuses on providing a clean, intuitive user interface. The gameplay area is divided into two main sections:

1. **Left Panel:**
 - Minesweeper grid
 - Game status indicators (flags counter, timer)

- Instructions and game result messages



Figure 4.7.1: Left Panel of the Web-Based Minesweeper

2. Right Panel:

- High score list for the selected difficulty
- User account management and stats

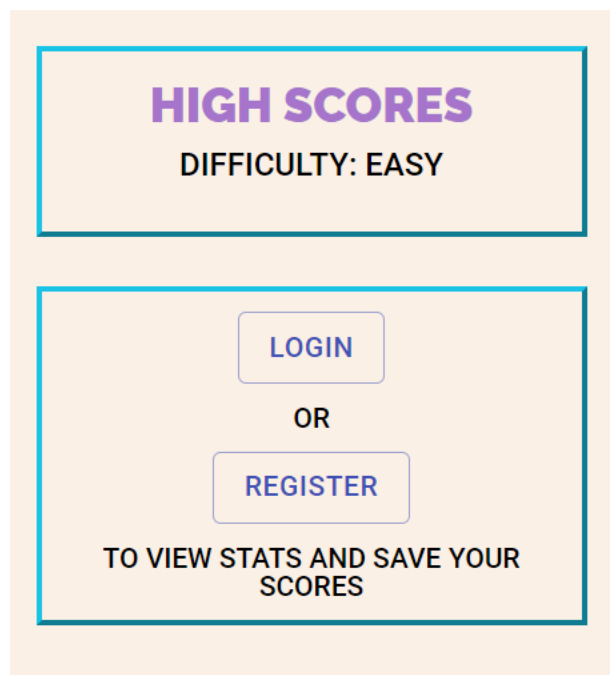


Figure 4.7.2: Right Panel of the Web-Based Minesweeper

4.8 API Design

The backend exposes RESTful endpoints for handling scores, high scores, and user accounts. Key routes include:

- **Scores Routes (/scores)**
 - **POST /scores:** Submit a new score.
 - **GET /scores/user:** Retrieve scores for the logged-in user.
- **High Scores Routes (/highscores)**
 - **GET /highscores/:difficulty:** Get high scores for a specific difficulty.
- **Users Routes (/users)**
 - **POST /users:** Register a new user.
 - **GET /users/scores:** Retrieve scores for the logged-in user.
- **Login Routes (/login)**
 - **POST /login:** Authenticate and log in the user.

4.9 Database Design

The database schema utilizes three primary tables:

Users Table

- **id:** Unique identifier (UUID)
- **username:** Unique username
- **password:** Hashed password
- **display_name:** Display name

Scores Table

- **id:** Unique identifier (UUID)
- **user_id:** References the **Users** table
- **difficulty:** Game difficulty level
- **time:** Completion time

High Scores Table

- **id:** Unique identifier (UUID)
- **user_id:** References the **Users** table

- **difficulty:** Game difficulty level
- **time:** Best completion time

Database Schema Diagram:

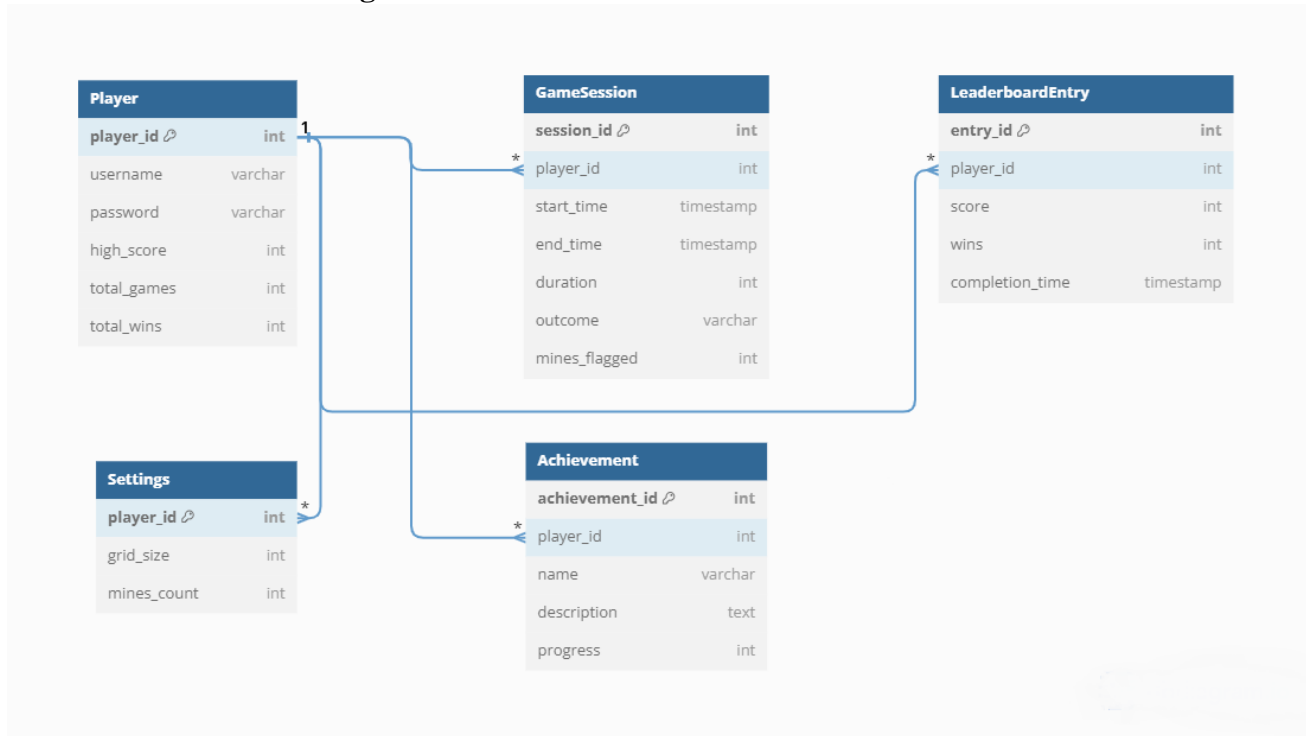


Figure 4.9.1: Database Schema Diagram

4.10 Game Mechanics

The gameplay revolves around revealing safe cells while flagging cells with hidden mines. The game state is updated based on the player's progress.

Game State Management

- **New:** Initial game state.
- **Won:** All safe cells revealed.
- **Lost:** A mine was triggered.

Mine Detection Logic

- Mines are randomly placed on the board.
- Adjacent cells with numbers indicate the count of neighboring mines.

4.11 High Score System

The high score system ranks players based on the fastest completion times for each difficulty level. Players can:

- View the top scores for each level.
- Track their personal best times.

4.12 User Account Management

Registration & Login

- Users can register using a unique username, password, and display name.
- Upon successful registration or login, a JWT token is provided for authenticated requests.

User Statistics

- Fastest completion time.
- Average completion time.
- Total games won.

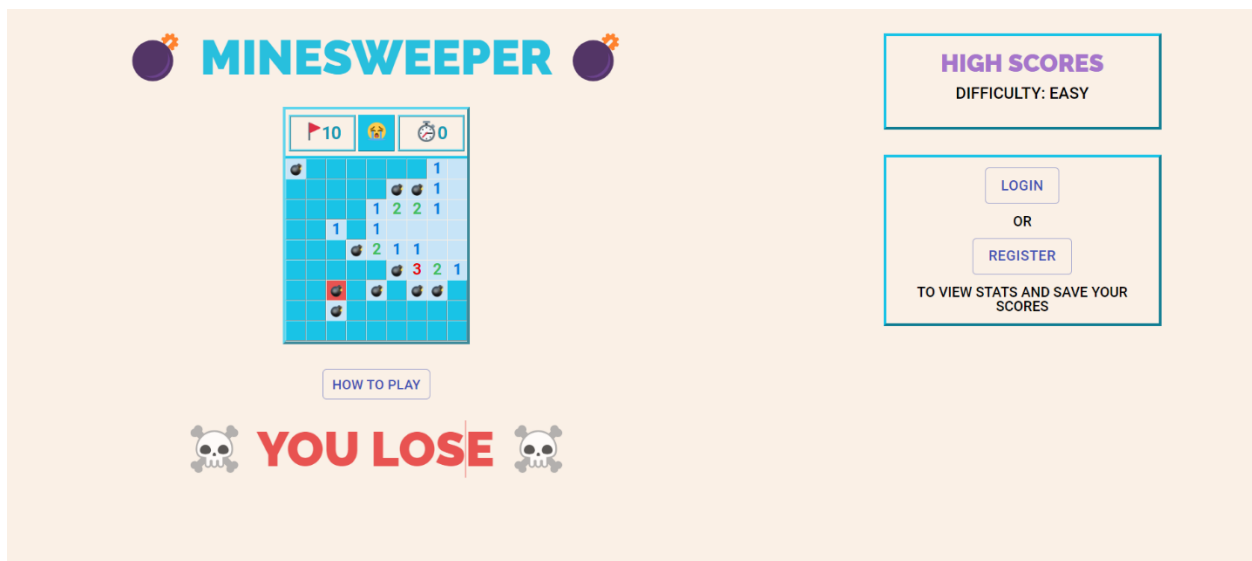


Figure 4.12.1: Web-Based Minesweeper

Chapter 5: Future Enhancements

1. **New Difficulty Levels:** Add medium and hard modes.
2. **Custom Board Sizes:** Allow users to set custom grid sizes and mine counts.
3. **Multiplayer Mode:** Real-time competition between players.
4. **Achievements & Rewards:** Unlock achievements based on gameplay milestones.

Chapter 6: Conclusion

The web-based Minesweeper project achieves the objective of recreating the classic game experience while adding modern features such as user accounts and high-score tracking. Its modular architecture enables easy maintenance and future feature additions.