CST8002 Programming Language Research Project

Assignment 3 Report

Student Name: Jagseer Singh Brar

Date: April 17, 2025

Course: Programming Language Research Project

Course code:

Instructor: Duncan Anderson

Contents

[Purpose 3](#_Toc195848467)

[Timeline 3](#_Toc195848468)

[Status 4](#_Toc195848469)

[Testing 5](#_Toc195848470)

[Traceability Matrix 5](#_Toc195848471)

[Conceptual Design 7](#_Toc195848472)

[User Interface Updates 7](#_Toc195848473)

[System Architecture 14](#_Toc195848474)

[Technologies and Frameworks 15](#_Toc195848475)

[References 15](#_Toc195848476)

# Purpose

Tupike is a comprehensive online Recipe Organizing and Sharing App that helps home cooks and food enthusiasts create, manage, and discover recipes. The platform offers users the ability to:

* Create, store, and manage personal recipes with detailed information
* Upload recipe images with step-by-step cooking directions
* Organize recipes into searchable collections
* Discover recipes shared by other users through search functionality
* Save favorite recipes for quick access
* Maintain a personalized digital recipe book

The primary objectives of Tupike are providing an intuitive recipe management system, streamlined recipe creation and editing, social recipe sharing, powerful search capabilities, and a responsive design that works across devices.

# Timeline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Task | Planned Start | Planned End | Actual Status |
| Research & Planning | Requirements Analysis | Jan 16, 2025 | Feb 5, 2025 | Completed |
| System Design | UI Wireframing & Database Schema | Feb 6, 2025 | Feb 15, 2025 | Completed |
| Backend Development | API Development (Node.js, Express, PostgreSQL) | Feb 16, 2025 | Mar 10, 2025 | Completed |
| Frontend Development | EJS Template & Frontend Integration | Mar 11, 2025 | Mar 29, 2025 | Completed |
| Search & Filter Implementation | Search API and UI Development | Mar 30, 2025 | Apr 5, 2025 | Completed |
| Recipe Detail Implementation | Detail View and Navigation | Apr 6, 2025 | Apr 10, 2025 | Completed |
| Testing & Debugging | Functional & Unit Testing | Apr 11, 2025 | Apr 16, 2025 | In Progress |
| Finalization | Hosting & Documentation | Apr 17, 2025 | Apr 20, 2025 | Upcoming |

## Status

The project has made significant progress since the previous report, with the successful implementation of all planned core functionality:

Key Achievements

* Search and Filter System: Implemented robust search functionality allowing users to find recipes by title, description, or username, with filter tabs for "All Recipes," "My Recipes," and "Favorites."
* Recipe Detail View: Created a dedicated recipe detail page that provides comprehensive information about each recipe, including ingredients, cooking procedures, and author information.
* Unified Navigation: Integrated search capabilities into the global navigation, providing consistent access to search functionality across the application.
* Responsive Design: Enhanced the UI to ensure optimal display on various device sizes, including responsive layout for search results and recipe details.
* User Experience Improvements: Added proper handling for empty search results and intuitive navigation between search results and recipe details.

Challenges Addressed

* Search Query Complexity: Successfully designed SQL queries to handle multiple search criteria with proper parameter handling.
* User Interface Consistency: Maintained consistent styling and behavior across all pages while adding new functionality.
* Database Performance: Optimized search queries to maintain performance with potentially large recipe collections.

# Testing

The testing approach for Tupike combines manual testing, integration testing, and component testing to ensure comprehensive validation of functionality and user experience. Key testing areas include:

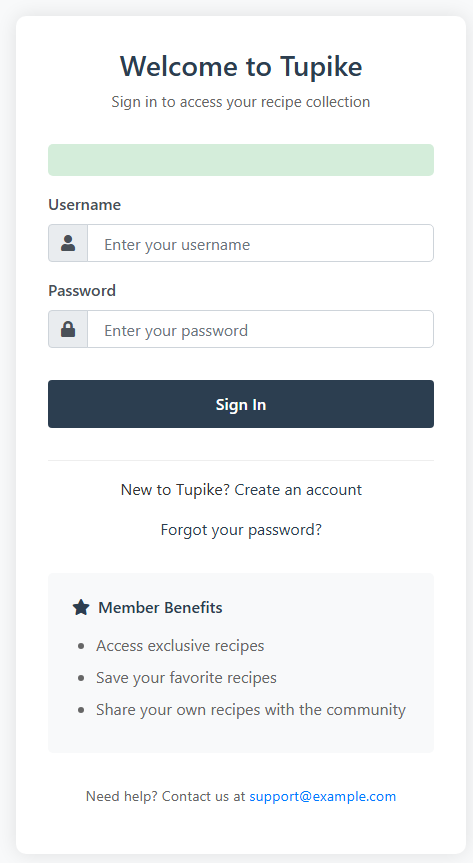
* User interface and interaction testing
* Backend functionality validation
* Database query testing
* Session management verification
* Responsive design confirmation

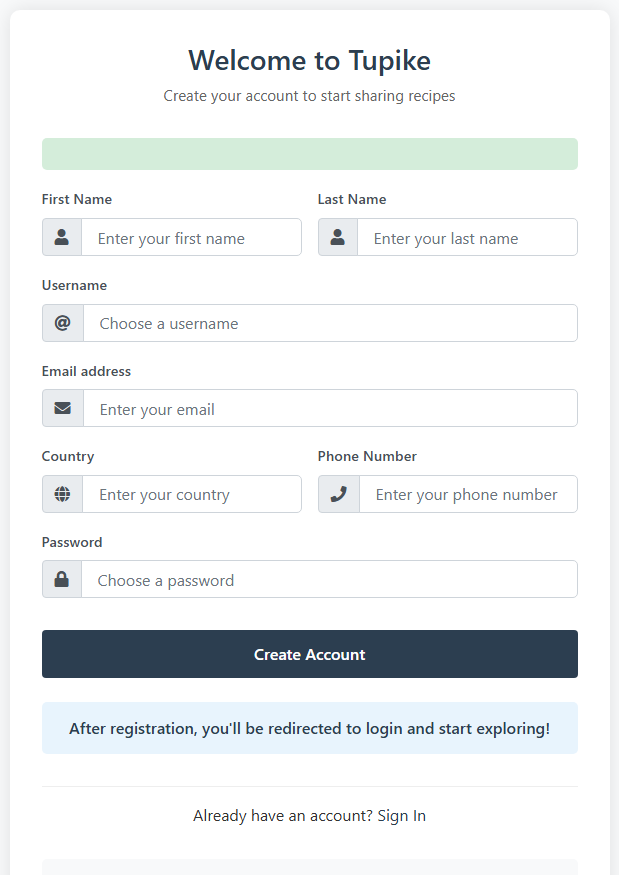
# Traceability Matrix

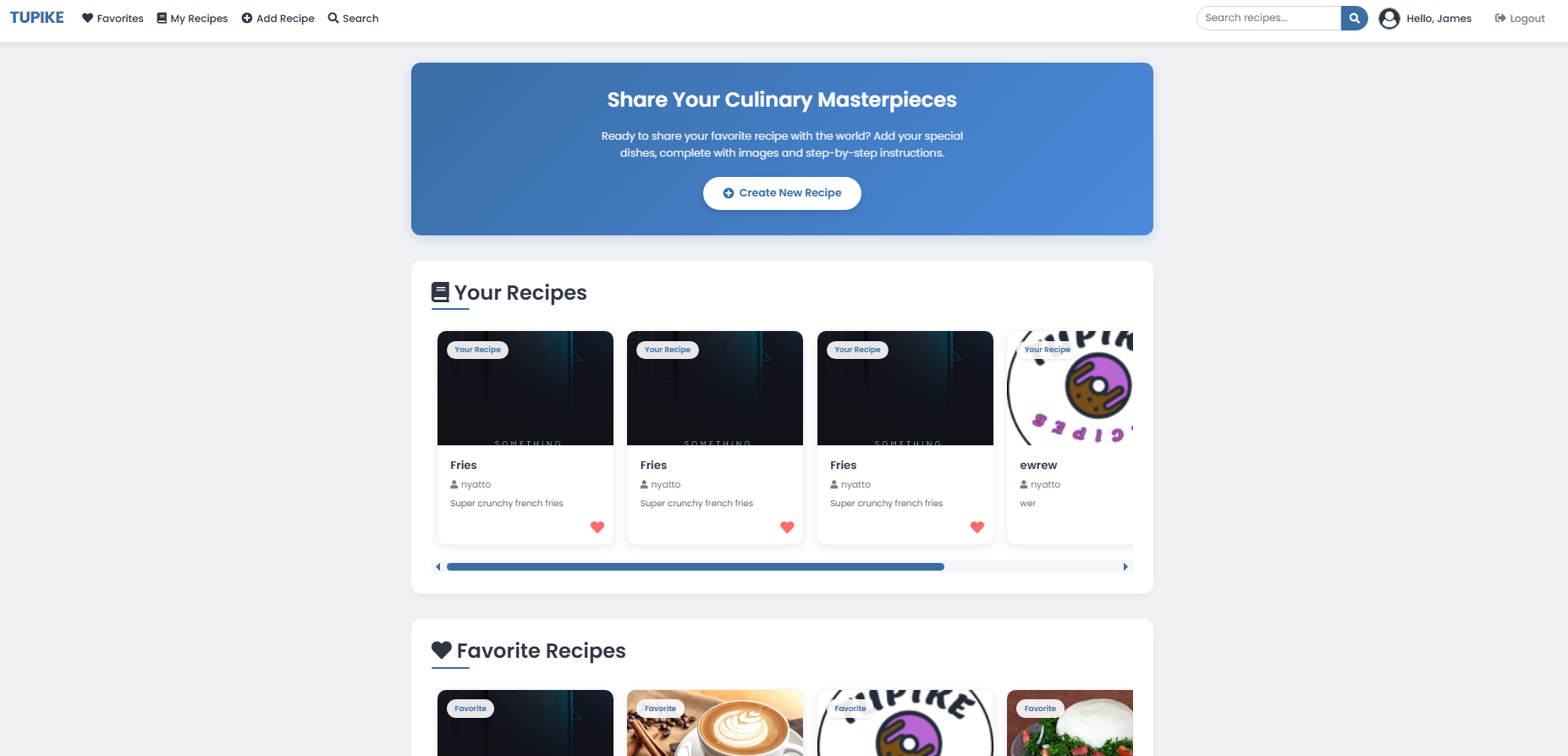
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req ID | Requirement Name | Category | Test Criteria | Expected Result | Status |
| TR-01 | User Registration | Technology | Register new user | User successfully added | Completed |
| TR-02 | User Login | Technology | Login with credentials | Session created, dashboard displayed | Completed |
| TR-03 | Recipe Creation | Business | Add a new recipe | Recipe successfully saved | Completed |
| TR-04 | Recipe Editing | Business | Modify existing recipe | Recipe updates saved | Completed |
| TR-05 | Image Upload | Technology | Upload recipe image | Image stored correctly | Completed |
| UR-01 | Search Functionality | User | Search recipes by title, description, username | Relevant recipes retrieved | Completed |
| UR-02 | Filter Functionality | User | Filter recipes by ownership and favorites | Filtered recipes displayed | Completed |
| UR-03 | Recipe Detail View | User | Click on recipe card | Complete recipe details displayed | Completed |
| BR-01 | Favorites System | Business | Add/remove favorite recipes | Favorites managed correctly | Completed |
| UR-04 | Responsive Design | User | Access site on different devices | UI adapts to screen size | Completed |

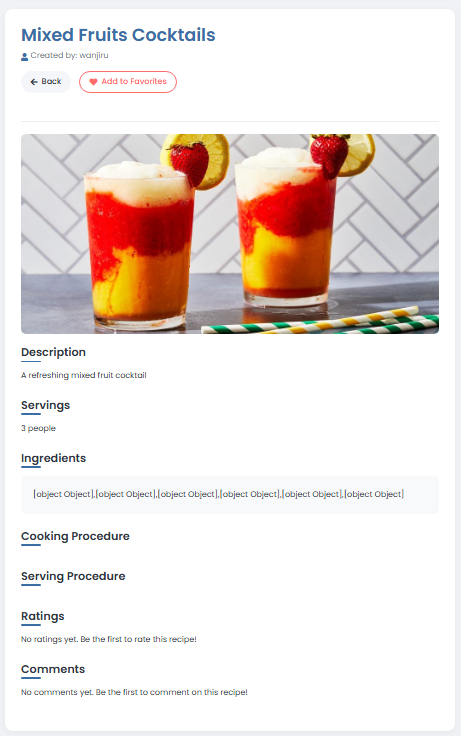
# Conceptual Design

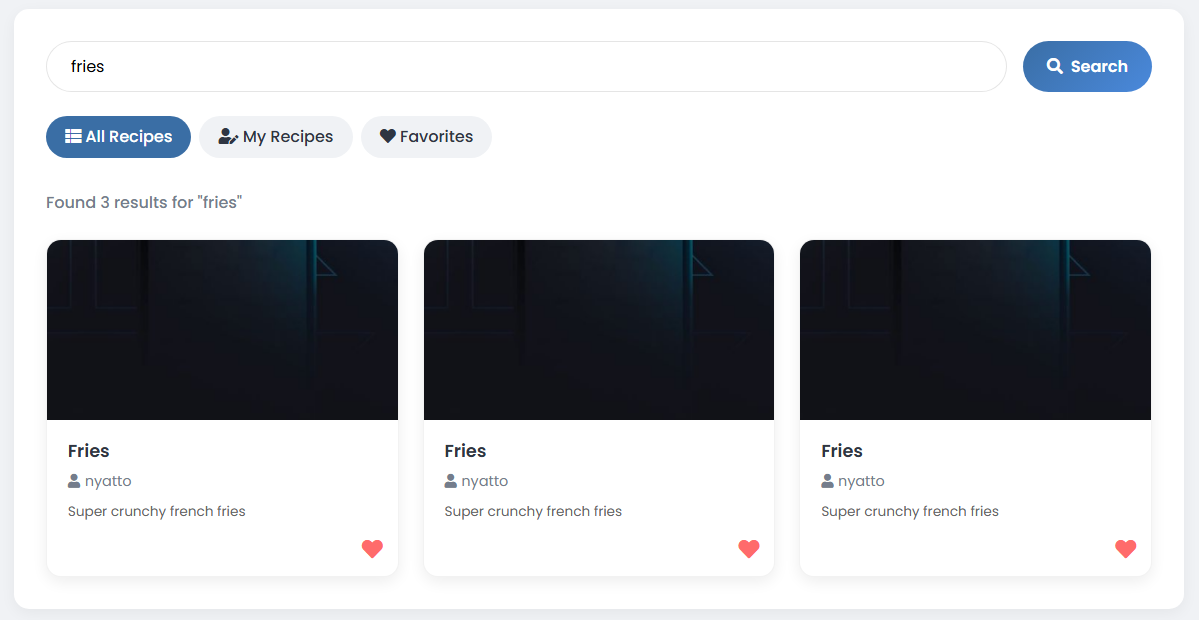
## User Interface Updates

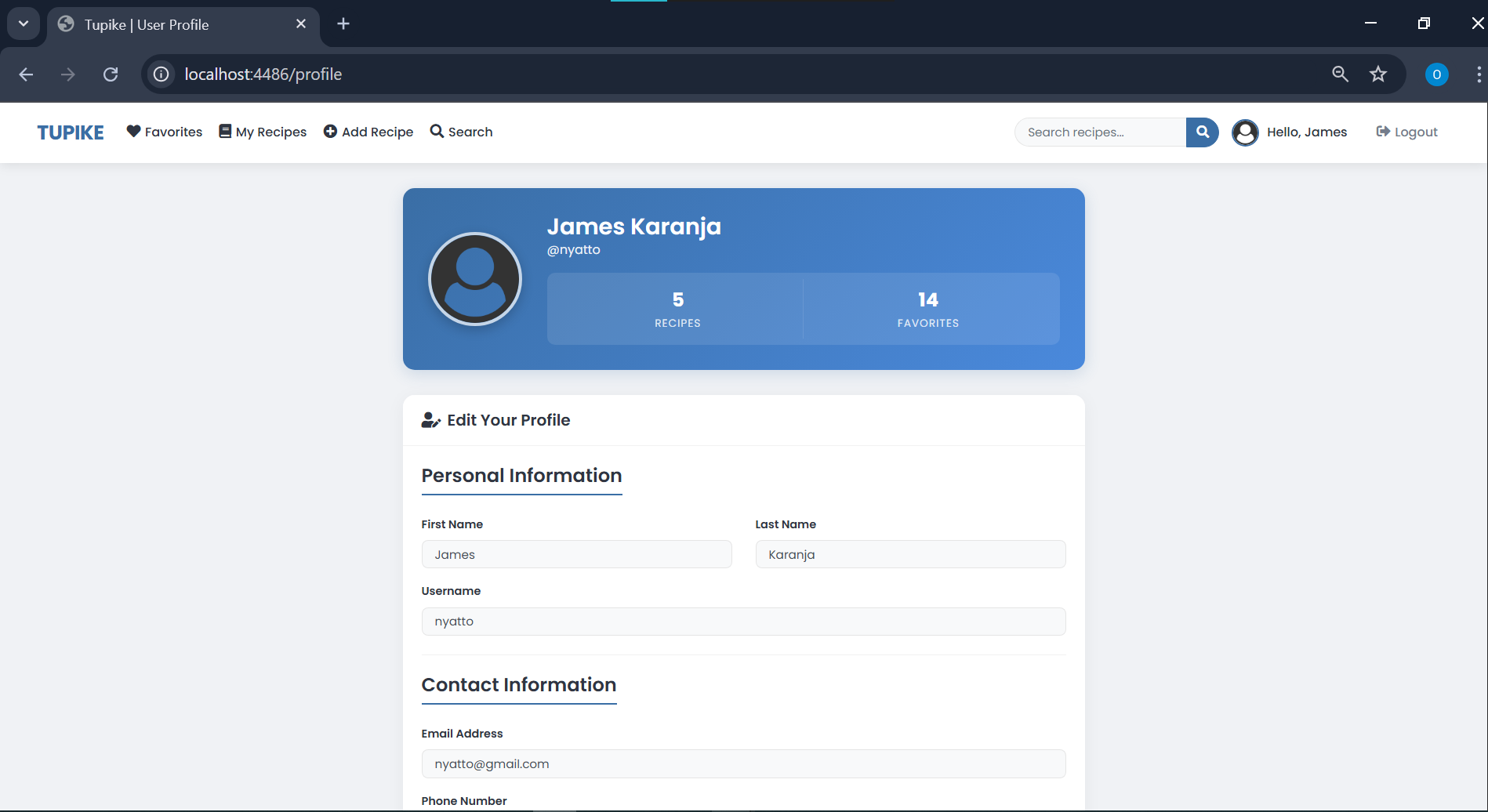












The user interface has been enhanced with:

Search Interface

The search page now provides:

* A prominent search bar for query input
* Filter tabs for categorizing recipes
* A responsive grid layout for search results
* Clear empty state messaging when no results are found

Recipe Detail View

The recipe detail view includes:

* Recipe header with title, author, and image
* Action buttons for favoriting and editing
* Well-formatted ingredients list and cooking procedures
* Responsive layout that adapts to screen sizes

Global Search Integration

The search functionality has been integrated into the global navigation across all pages, making it accessible throughout the application.

## System Architecture

The system follows the Model-View-Controller (MVC) pattern:

* Models: Database interaction for users, recipes, and favorites
* Views: EJS templates for rendering different application states
* Controllers: Route handlers processing requests and coordinating data flow

Search and Filter Implementation

The search and filter functionality has been implemented with:

1. Backend Implementation:

* Parameterized SQL queries with dynamic construction based on filters
* Join operations to retrieve related user and favorite information
* Proper error handling for database operations

2. Frontend Implementation:

* Form submission with GET method for bookmarkable searches
* Dynamic filter tab highlighting
* Responsive grid layout for search results
* Interactive recipe cards with event handling

## Technologies and Frameworks

* Backend: Node.js, Express.js, PostgreSQL
* Frontend: EJS Templates, Bootstrap, Font Awesome, JavaScript
* Authentication: bcrypt, express-session
* File Handling: Multer

# References

Desai, J. (2023, December 27). Web application vs desktop application: pros and cons. Positiwise. https://positiwise.com/blog/web-application-vs-desktop-application-pros-and-cons

Joshi, M. (2023, May 11). Angular vs React vs Vue: Core Differences | BrowserStack. BrowserStack. https://www.browserstack.com/guide/angular-vs-react-vs-vue

Udhayaseelan. (2024, January 1). Task Management App -Solving a UX design challenge - Udhayaseelan - Medium. Medium. https://medium.com/@udhayaseelan941/task-management-app-solving-a-ux-design-challenge-4fda61d189bf

StackHawk. (2023, February 12). Secure your web App: 10 key improvements. StackHawk. https://www.stackhawk.com/blog/web-application-security-checklist-10-improvements/

MDN Web Docs. (2023). Using the Fetch API. Mozilla. https://developer.mozilla.org/en-US/docs/Web/API/Fetch\_API/Using\_Fetch

Express.js. (2023). Express API reference. Express.js. https://expressjs.com/en/4x/api.html

Bootstrap. (2023). Grid system. Bootstrap. https://getbootstrap.com/docs/5.0/layout/grid/