FINGRO

Arshdeep Singh(S224750073)
Emmalu joseph(S224791713)
Shrikesh(S224404506)
Umar Khan(S223744692)

Software Requirements Specification (SRS) for FinGro

Project Name: FinGro (Finance & Grocery Planner)

1. Introduction

1.1 Purpose

The purpose of the FinGro web application is to assist users in managing their grocery planning and financial tracking efficiently. By providing an integrated platform for meal planning, grocery list generation, expense categorization, and savings monitoring, FinGro empowers users to make informed financial and lifestyle decisions.

1.2 Scope

FinGro serves as a comprehensive web-based solution for both meal and financial planning. It supports the following key functionalities:

- **Grocery Planning:** Allows users to plan their meals and generate consolidated grocery lists based on family size and selected dishes.
- **Expense Tracking:** Enables users to track daily, weekly, and monthly expenses, categorized for easy review.
- **Savings Insights:** Users can set financial goals and visualize their progress toward achieving them.

FinGro is designed to operate on desktop and mobile web browsers and supports various user groups, including individuals, students, and families.

1.3 Definitions and Abbreviations

- **SRS:** Software Requirements Specification
- **UI:** User Interface
- **API:** Application Programming Interface
- **GDPR:** General Data Protection Regulation (data privacy regulation)
- **MongoDB:** A NoSQL database used for data storage

2. System Overview

FinGro functions as a standalone web application built using modern web technologies. The front end is developed using HTML, CSS, JavaScript, and Materialize CSS, while the back end runs on Node.js and MongoDB for secure and scalable data storage.

3. Functional Requirements

3.1 Grocery Planning Module

Description: Allows users to plan meals and generate weekly grocery lists.

Requirements:

- Users can add or remove dishes for each meal of the week (breakfast, lunch, dinner).
- The system calculates grocery quantities based on serving sizes.
- The module fetches recipes and ingredients from external APIs if required.

User Story:
"As a user, I want to plan my meals and generate a comprehensive grocery list for the week to simplify my shopping."

3.2 Expense Tracker Module

Description: Tracks user expenses and categorizes them for budgeting and analysis.

Requirements:

- Users can add expenses with descriptions and categories (e.g., groceries, rent, utilities).
- The system generates summaries of daily, weekly, and monthly expenses.
- Visual reports (e.g., bar charts, pie charts) display expense breakdowns using external charting libraries.

User Story:
"As a user, I want to track my monthly expenses so that I can monitor my spending and plan my budget accordingly."

3.3 Savings Goals Module

Description: Provides users with tools to set and monitor financial goals.

Requirements:

- Users can set savings goals with specific amounts and timelines.
- The module displays progress toward each goal through graphical insights.
- Notifications alert users when they are close to meeting or missing their goal targets.

4. Non-Functional Requirements (NFR)

- **Performance:** All user actions (e.g., adding expenses, generating grocery lists) should complete within 3 seconds.
- **Security:** All user data must be encrypted both in transit (HTTPS) and at rest (MongoDB).
- **Scalability:** The system should handle up to 100,000 concurrent users.
- **Usability:** The UI must be intuitive and easy to use, even for non-technical users.
- **Reliability:** The application should maintain 99.9% uptime, ensuring availability.

5. System Architecture

5.1 High-Level Overview

- **Frontend:** Developed using HTML, CSS, JavaScript, and Materialize CSS for responsive styling.
- **Backend:** Built using Node.js for server-side logic and MongoDB for database storage.

5.2 Data Flow

- **Inputs:** User-provided data, such as meal preferences and expense details.
- **Processes:** Data aggregation, API calls for recipes, and financial calculations.
- **Outputs:** Consolidated grocery lists and visualized expense summaries.

6. External Interfaces

- **User Interface:** Web application accessible via browser with navigation options for meal planning and expense tracking.
- **API Integration:** External APIs for fetching recipes and ingredient data.
- **Communication Interface:** HTTP/HTTPS protocols for client-server communication.

7. Constraints and Assumptions

- **Assumptions:**
    - Users have a stable internet connection while using the app.
    - MongoDB is used as the primary database for storage.
- **Constraints:**
    - The web application will support only English language for the initial release.

8. User Stories and Use Cases

8.1 Grocery Planner Use Case

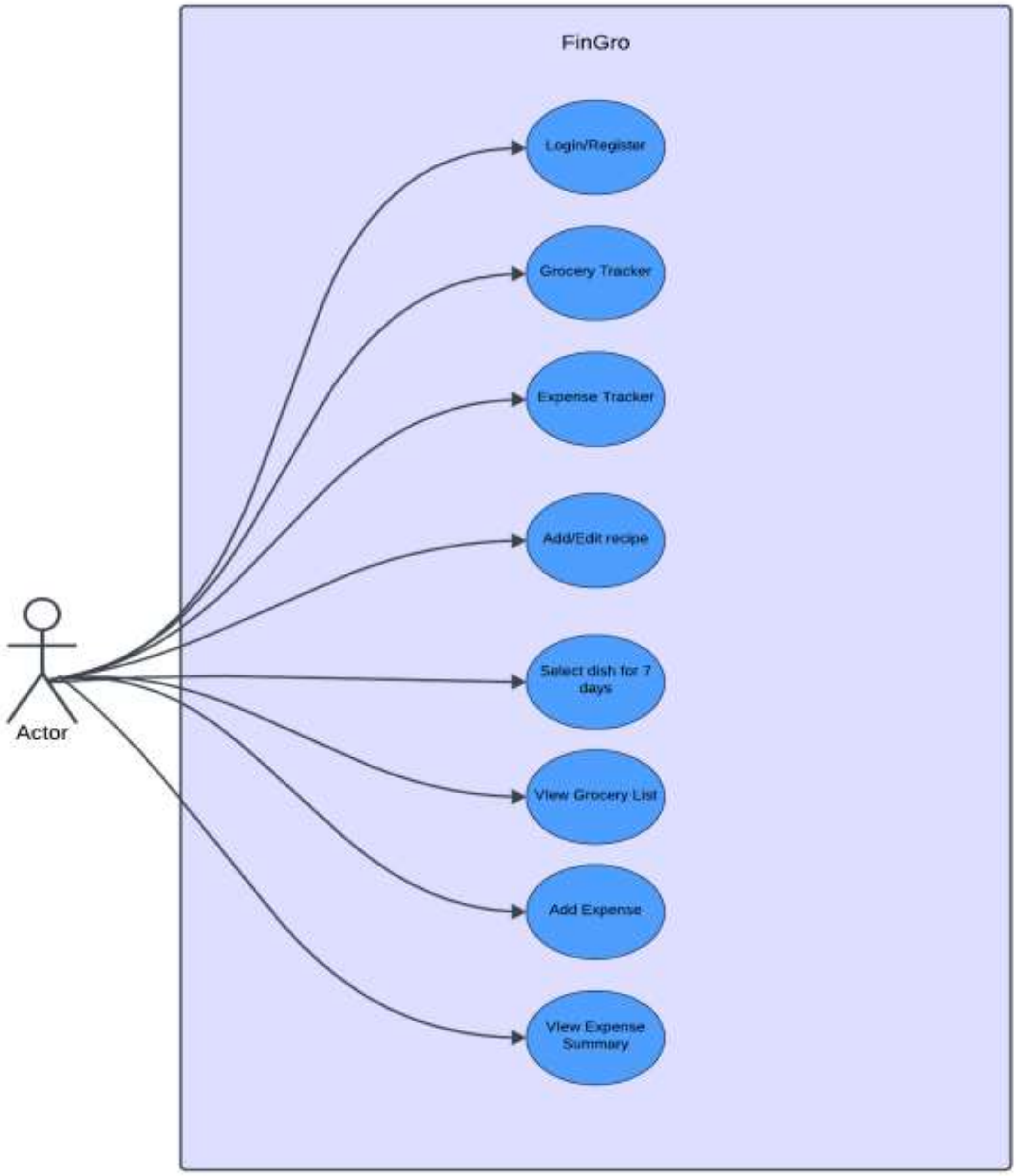
Scenario: A user plans meals for the week and generates a grocery list.
Steps:

1. Log in and select the "Grocery Planner" section.
2. Add dishes for each meal of the week.
3. The system generates a grocery list with ingredient quantities.

8.2 Expense Tracker Use Case

Scenario: A user tracks expenses for the month.
Steps:

1. Log in and navigate to the "Expense Tracker" tab.
2. Add expenses and categorize them (e.g., "Groceries" or "Transportation").
3. The system displays a summary report of total expenses per category.



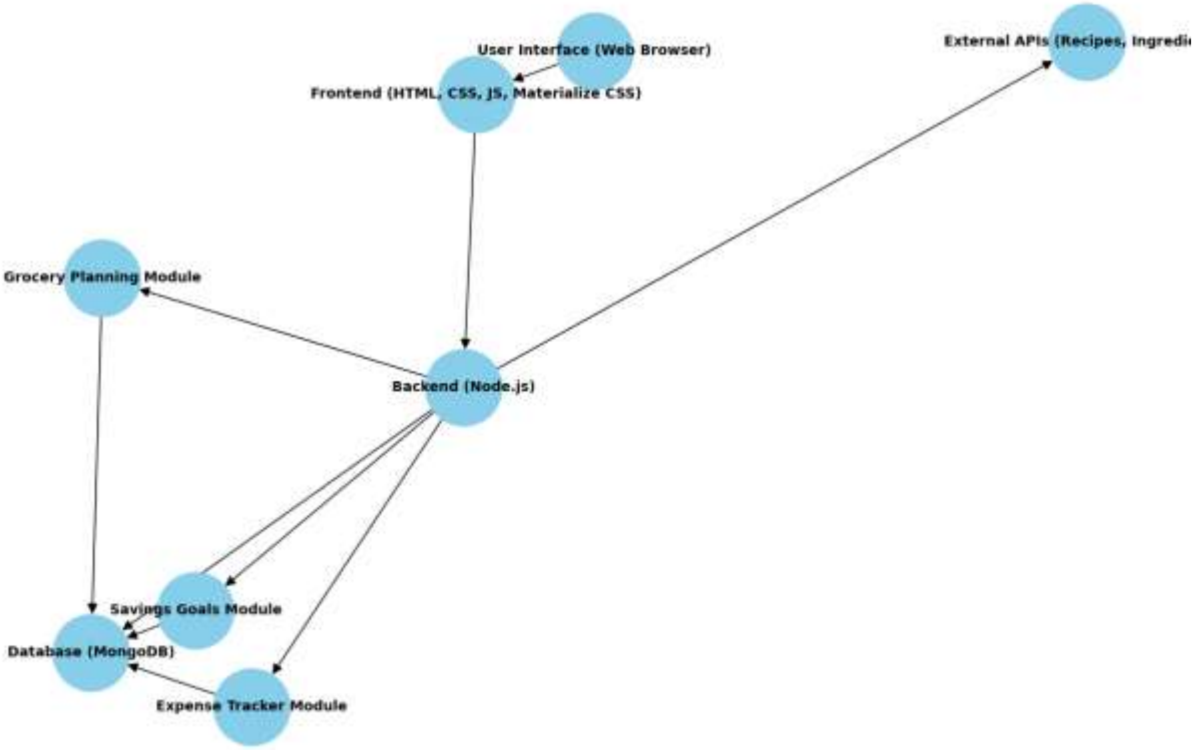
9. Work Breakdown Structure (WBS)

- **Frontend Development:** Implement the UI for meal planning and expense tracking.
- **Backend Development:** Create REST APIs, connect MongoDB, and implement user authentication.
- **Testing:** Conduct usability tests and load tests for performance evaluation.
- **Deployment:** Finalize and host the web app on a cloud platform (e.g., AWS).

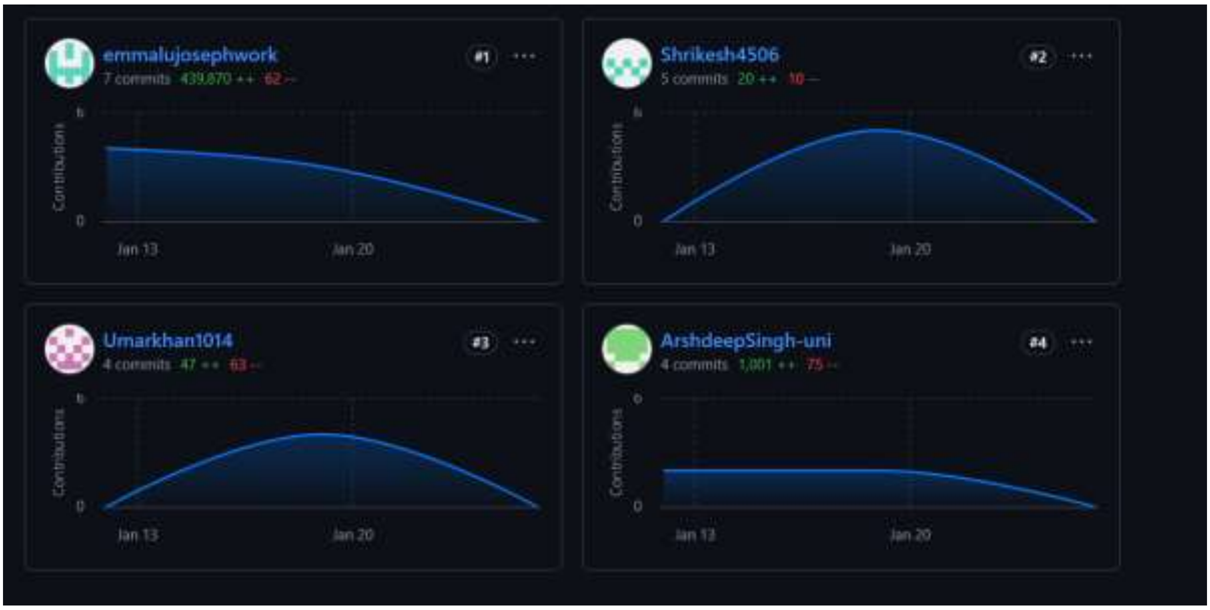
10. Approval and Sign-Off

This document must be reviewed and signed off by all stakeholders, including team members, project managers, and product owners, before proceeding with development.

Application Diagram


User Interface (Web Browser)
Frontend (HTML, CSS, JS, Materialize CSS)
External APIs (Recipes, Ingredi
Grocery Planning Module
Backend (Node.js)
Savings Goals Module
Database (MongoDB)
Expense Tracker Module

GitHub


emmalujosephwork #1
7 commits 439,870 ++ 62 --

Shrikesh4506 #2
5 commits 20 ++ 10 --

Umarkhan1014 #3
4 commits 47 ++ 63 --

ArshdeepSingh-uni #4
4 commits 1,001 ++ 75 --

Appendix 1: Sprint 1 Review

During Sprint 1, the team reviewed several key aspects of the project's progress and addressed critical questions related to the application's status, data connections, and functionality. Below is a summary of the discussions and outcomes:

1. What is the status of our application?

- The application's core structure has been established.
- Initial setup, including foundational frameworks and environments, is complete.
- Some basic UI components have been implemented.
- Identified a few technical challenges related to feature integration.

2. Data Connection

- The database schema was finalized and integrated.
- Basic CRUD operations were successfully tested.
- Connectivity between the backend and the database is functional but requires optimization.
- Established plans for securing data access and implementing authentication protocols.

3. Functionality Done by the Group

- The team collectively implemented the initial features, including user login, API endpoints, and basic UI components.
- Collaborative efforts focused on setting up core modules and ensuring smooth data flow between the frontend and backend.
- Progress was made on the grocery planning feature and its integration with the application.

4. How the Grocery Plan Works

- The grocery planning module allows users to:
  - Input dietary preferences and restrictions.
  - Generate a weekly grocery list based on pre-defined meal plans.
  - Modify and save grocery lists for future use.
- Backend logic for generating grocery plans is under development, with basic functionality in place.
- Frontend components for displaying and managing grocery plans are partially complete.

5. Expense Tracker

- The expense tracker module enables users to:
  - Record daily expenses with details such as category, amount, and date.
  - Visualize spending patterns through graphical summaries.
  - Set monthly budgets and receive alerts when nearing limits.
- The backend for storing and retrieving expense data is fully functional.
- Initial UI components for expense entry and budget management have been implemented.
- Integration with the database for real-time updates is complete, with further optimization planned.

Next Steps

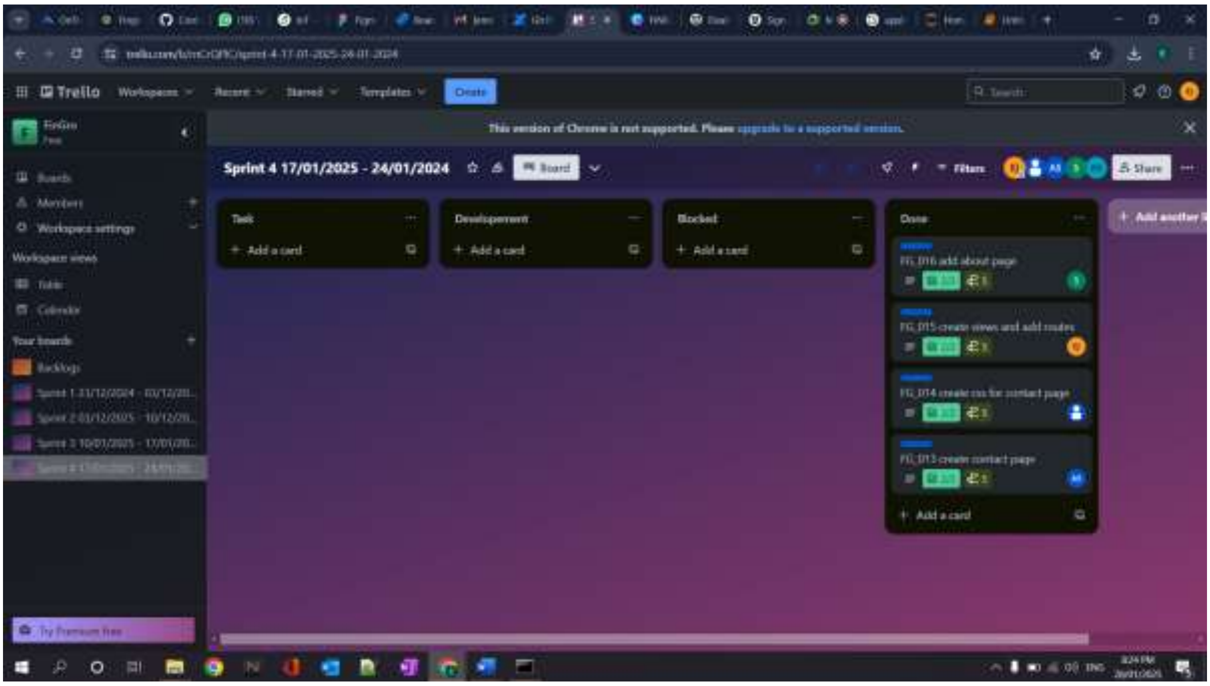
- Resolve technical challenges in feature integration.
- Enhance data connection performance and security.
- Complete remaining frontend components for the grocery planning and expense tracker modules.
- Conduct team-wide testing of implemented features.

This appendix captures the current state of Sprint 1 and provides a roadmap for addressing pending tasks and improvements.

Trello Screen Shots

Trello Link : https://trello.com/b/CHfdtBBS

1. The overall Trello board structure used for managing tasks during the project.



This version of Chrome is not supported. Please upgrade to a supported version.

Sprint 4 17/01/2025 - 24/01/2024

Task

Developement

Blocked

Done

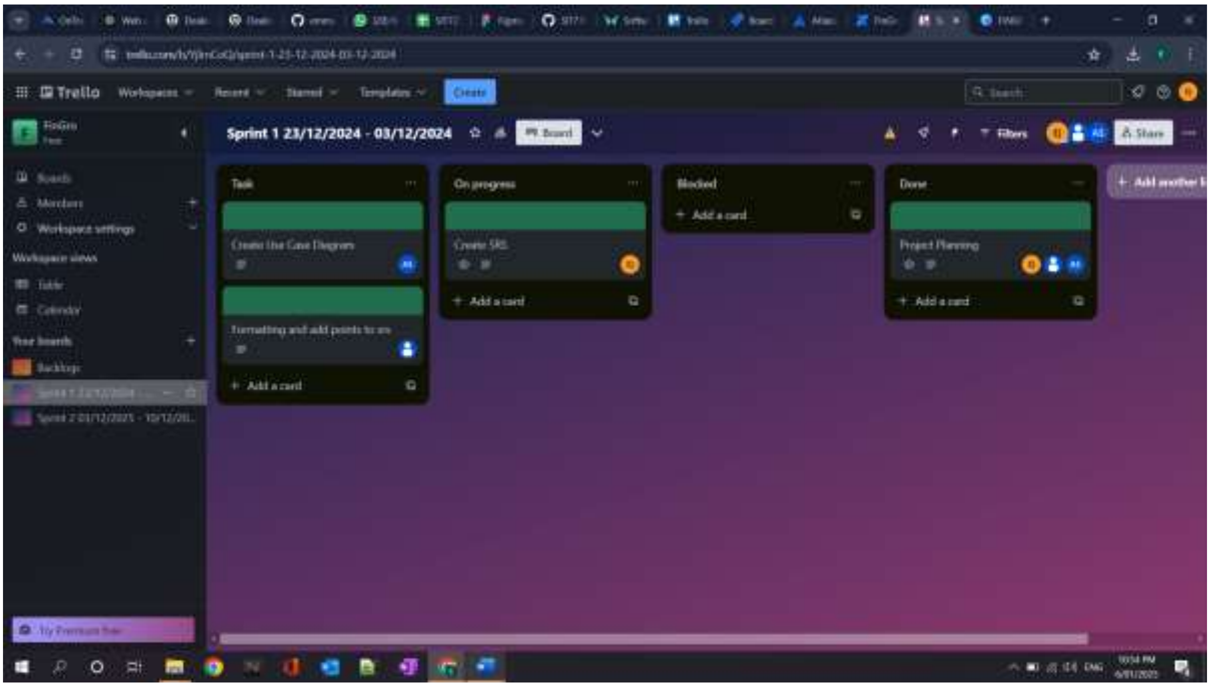
FG_D16 add about page

FG_D15 create views and add routes

FG_D14 create css for contact page

FG_D13 create contact page

2. In Each Sprint We have 4 stages for a task.



Sprint 1 23/12/2024 - 03/12/2024

Task

Create Use Case Diagram

Formatting and add points to srs

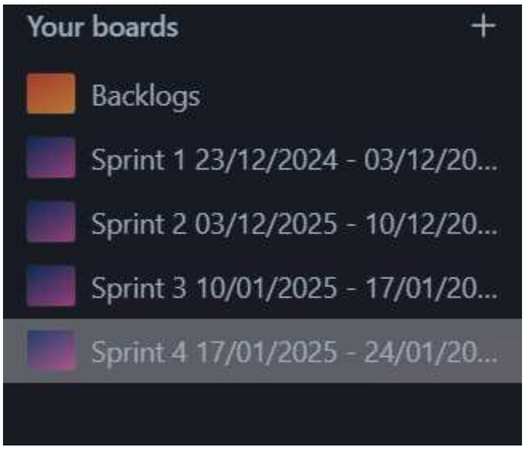
On progress

Create SRS

Blocked

Done

Project Planning



Your boards +

Backlogs

Sprint 1 23/12/2024 - 03/12/20...

Sprint 2 03/12/2025 - 10/12/20...

Sprint 3 10/01/2025 - 17/01/20...

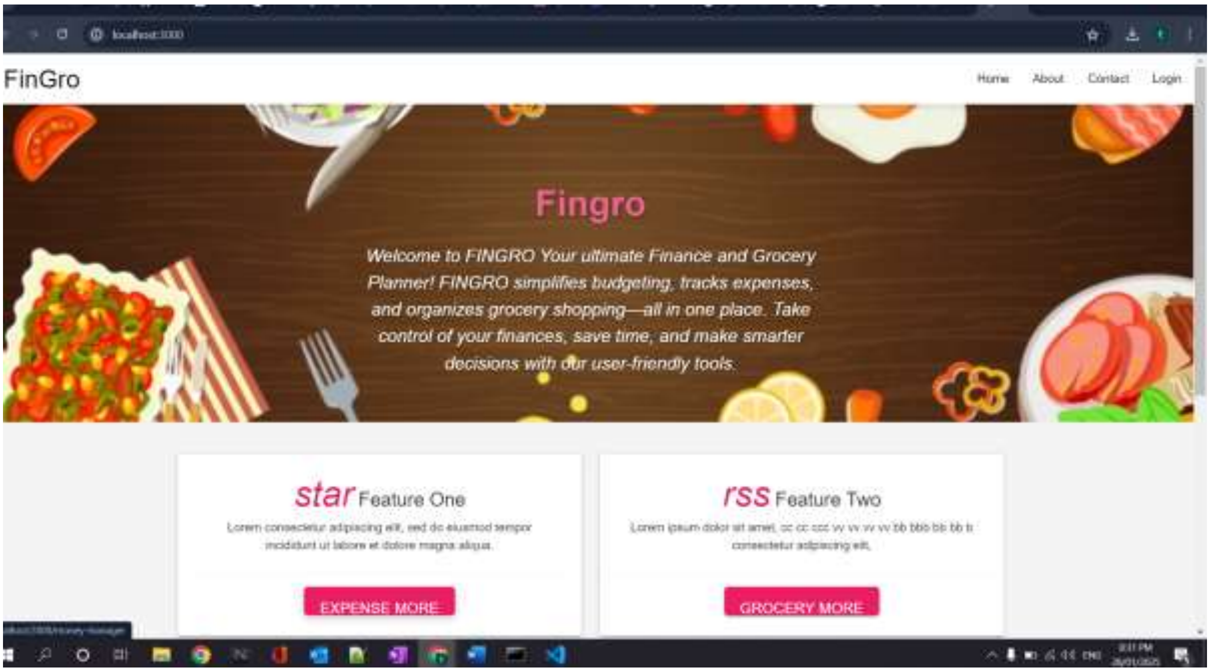
Sprint 4 17/01/2025 - 24/01/20...

Team Tasks Breakdown

Team Members and Their Contributions:

1. **Emmalu Joseph (S224791713):**
   - Completed the basic structure for the login, signup, and home pages.
   - Added the started and grocery planning functionality.
2. **Arshdeep Singh (S224750073):**
   - Developed the content for the Contact and About pages.
   - Started the smart expense tracker functionality.
3. **Umar Khan (S223744692):**
   - Designed and implemented CSS for the Home page.
   - Handled specific tasks related to the Login page.
4. **Shrikesh (S224404506):**
   - Designed and implemented CSS for the About Us and Signup pages.
   - Fixed various issues and completed miscellaneous tasks across the project.

Webpage Screenshots



Fingro

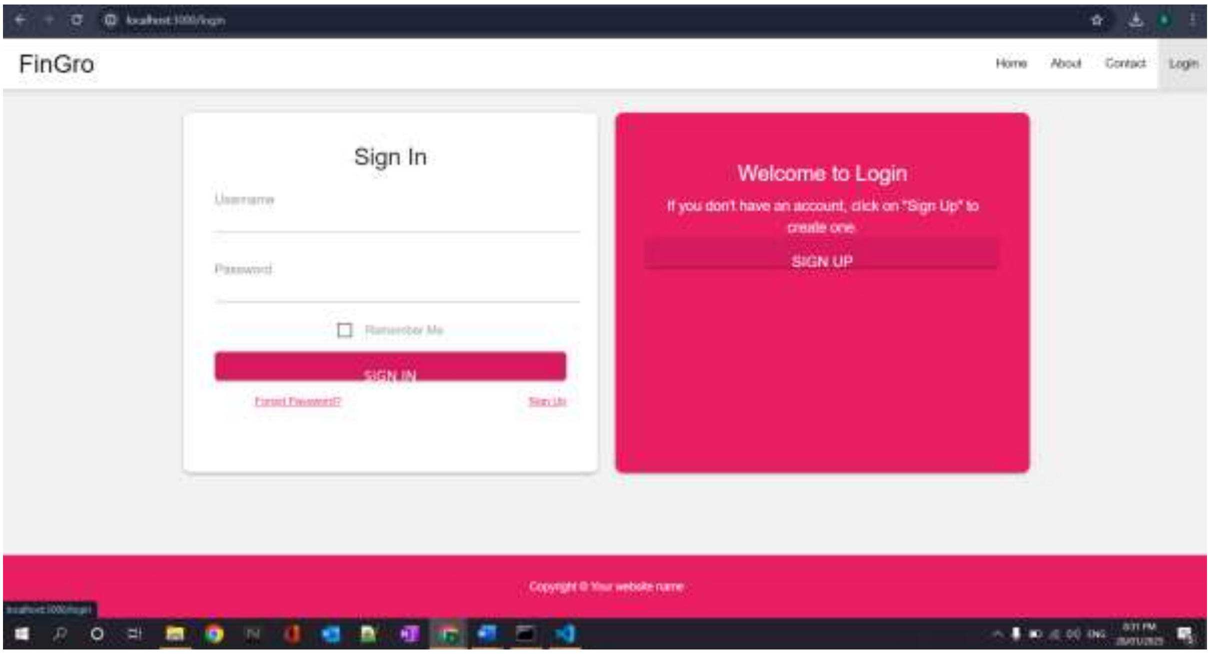
Welcome to FINGRO Your ultimate Finance and Grocery Planner! FINGRO simplifies budgeting, tracks expenses, and organizes grocery shopping—all in one place. Take control of your finances, save time, and make smarter decisions with our user-friendly tools.

star Feature One

Lorem consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

rss Feature Two

Lorem ipsum dolor sit amet, cc cc ccc vv vv vv vv bb bbb bb bb b consectetur adipiscing elit.



Sign In

Username

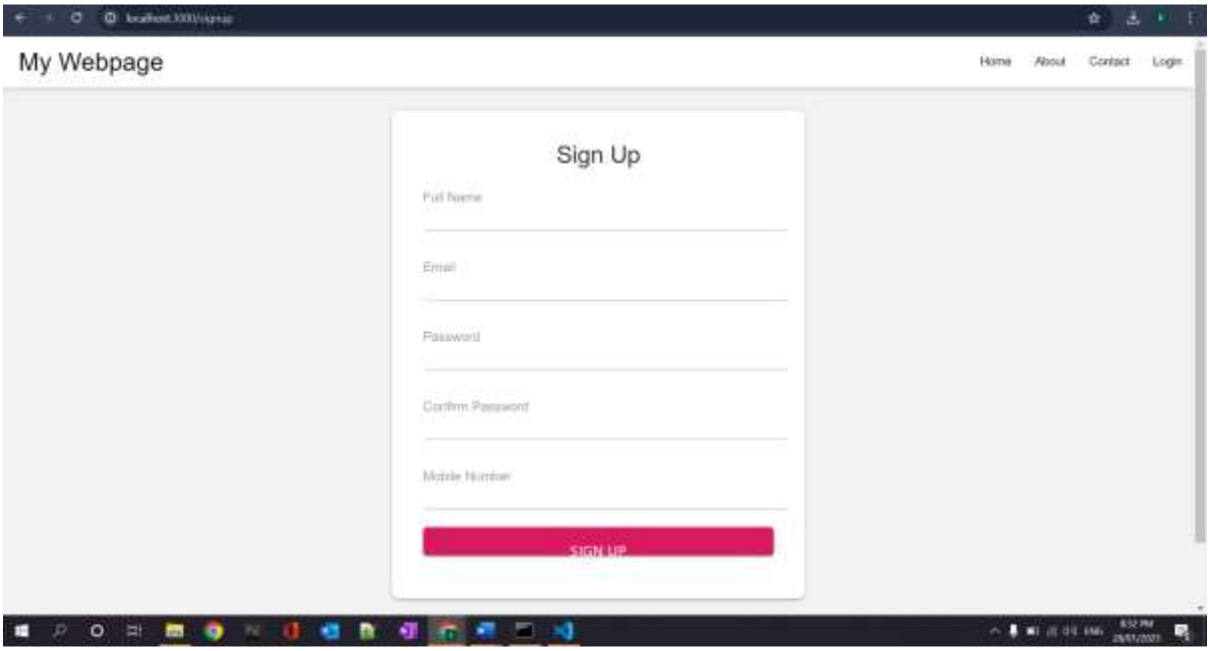
Password

□ Remember Me

Welcome to Login

If you don't have an account, click on "Sign Up" to create one.

Copyright © Your website name



localhost:3000/signup

My Webpage

Sign Up

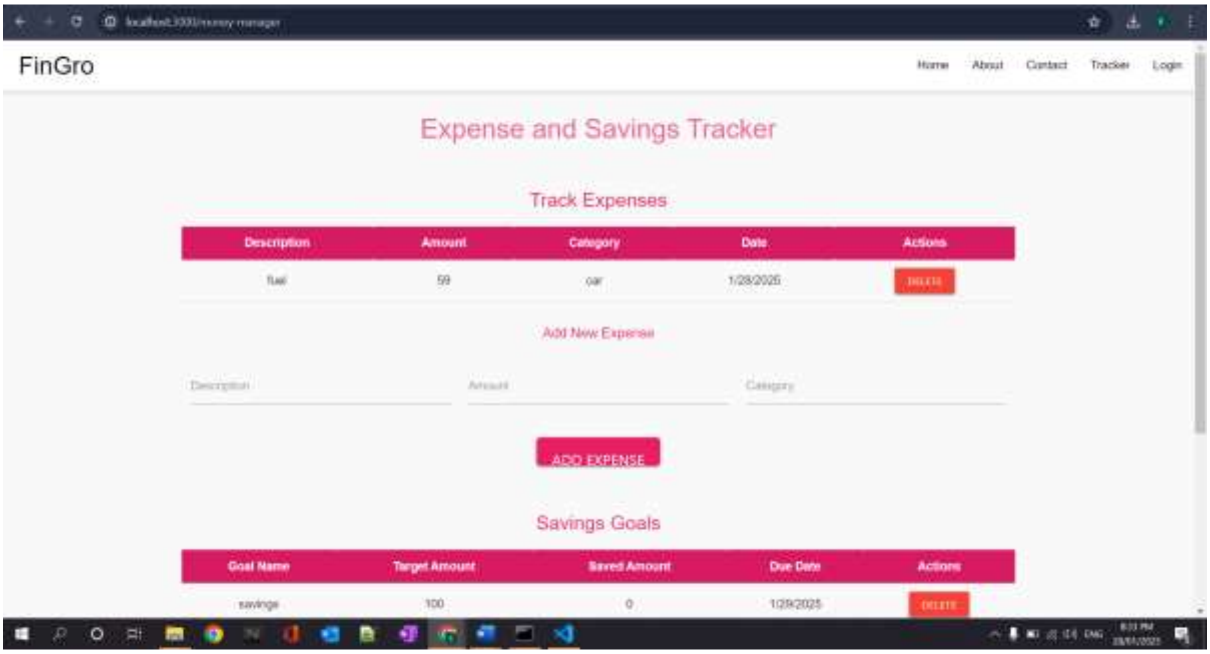
Full Name

Email

Password

Confirm Password

Mobile Number



localhost:3000/money-manager

FinGro

Expense and Savings Tracker

Track Expenses

Description	Amount	Category	Date	Actions
fuel	59	car	1/28/2025	DELETE

Add New Expense

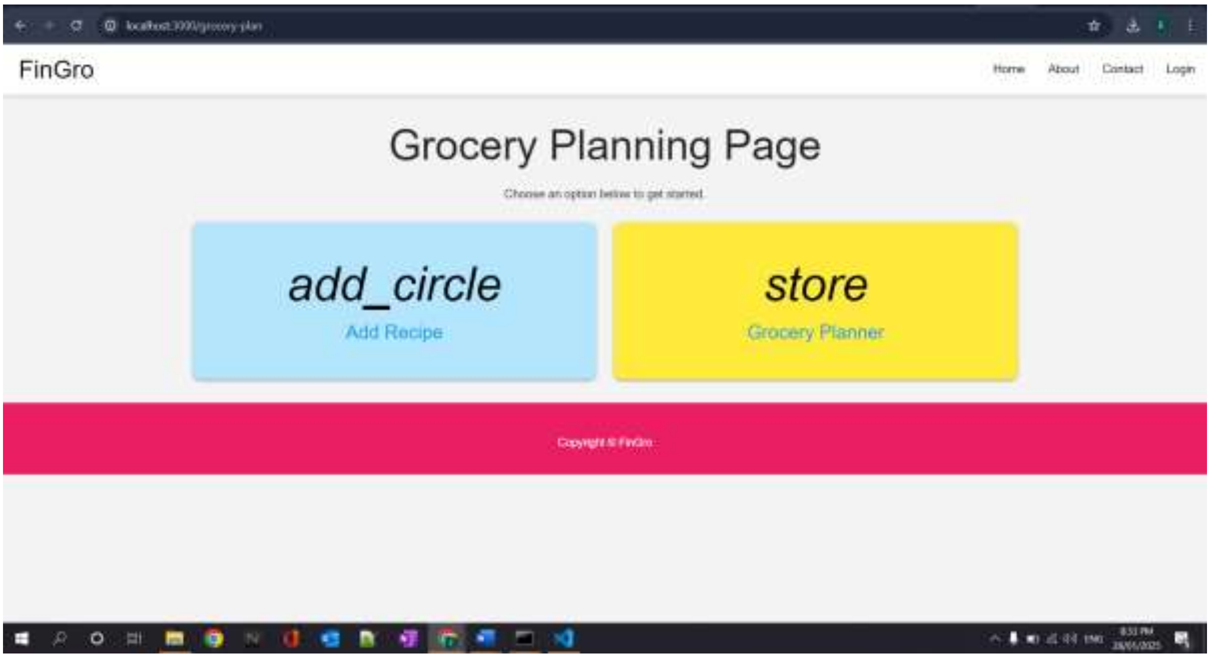
Description

Amount

Category

Savings Goals

Goal Name	Target Amount	Saved Amount	Due Date	Actions
savings	100	0	1/29/2025	DELETE



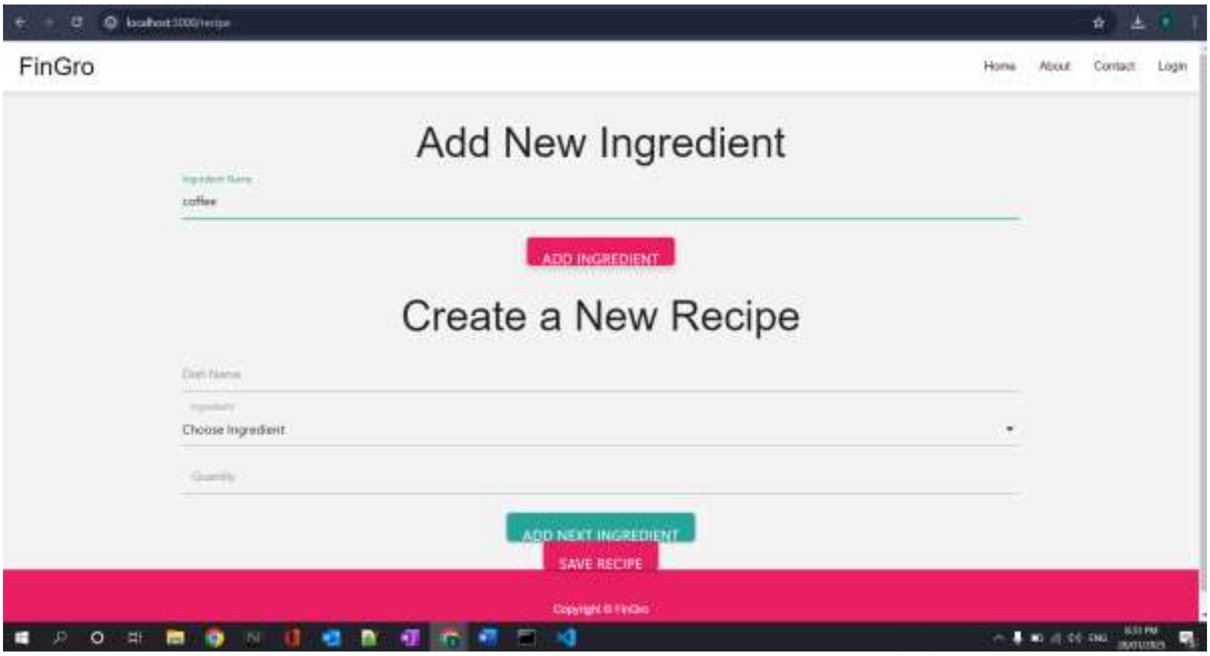
localhost:3000/grocery-plan

FinGro

Grocery Planning Page

Choose an option below to get started.

Copyright © FinGro



Add New Ingredient

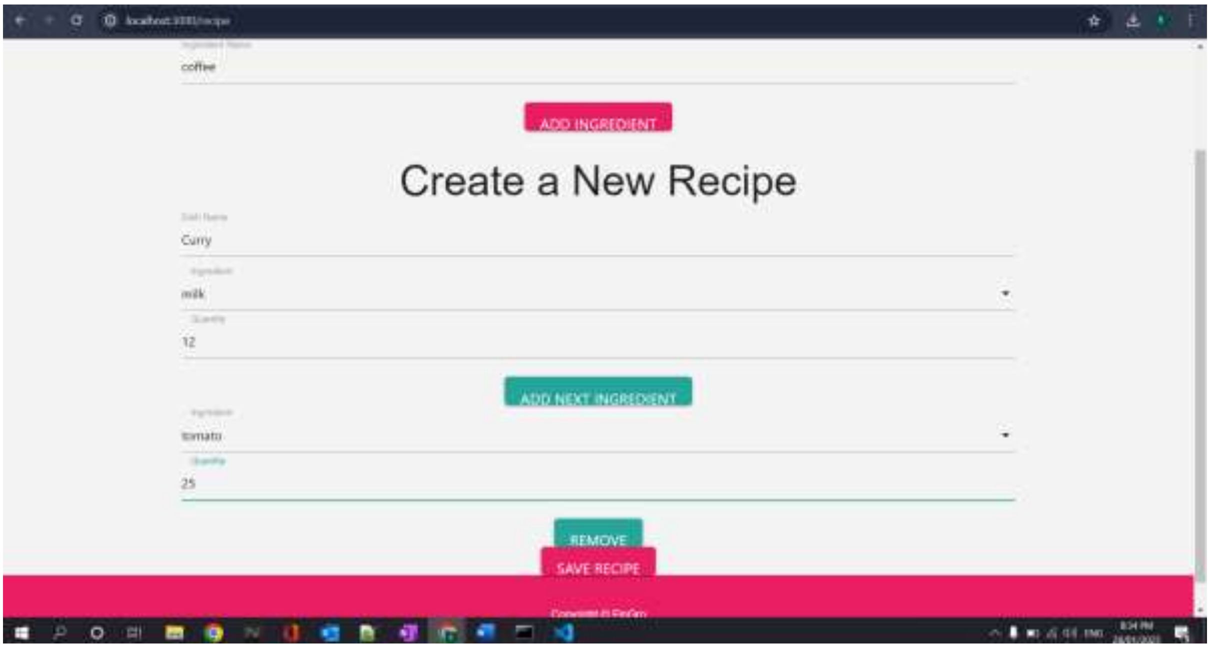
Ingredient Name: coffee

Create a New Recipe

Dish Name

Ingredient: Choose Ingredient

Quantity



Ingredient Name: coffee

Create a New Recipe

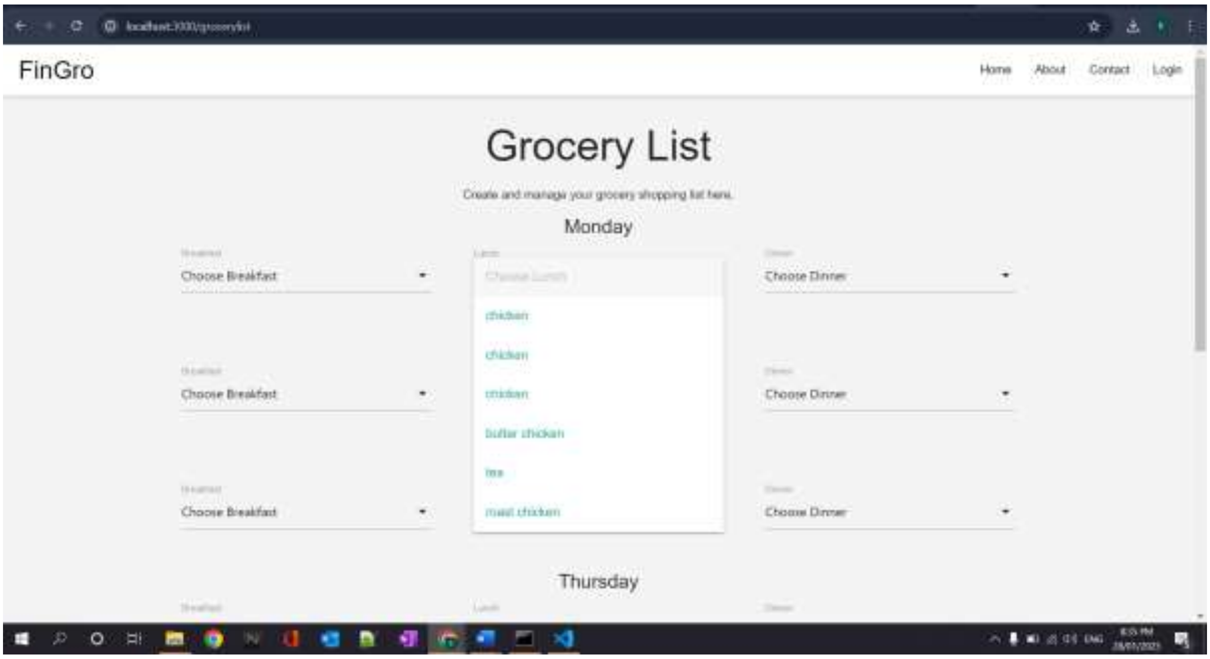
Dish Name: Curry

Ingredient: milk

Quantity: 12

Ingredient: tomato

Quantity: 25



Grocery List

Create and manage your grocery shopping list here.

Monday

Breakfast: Choose Breakfast

Lunch: Choose Lunch

Dinner: Choose Dinner

Breakfast: Choose Breakfast

Dinner: Choose Dinner

Breakfast: Choose Breakfast

Dinner: Choose Dinner

Thursday

Breakfast

Lunch

Dinner

100 words reflection for each participant

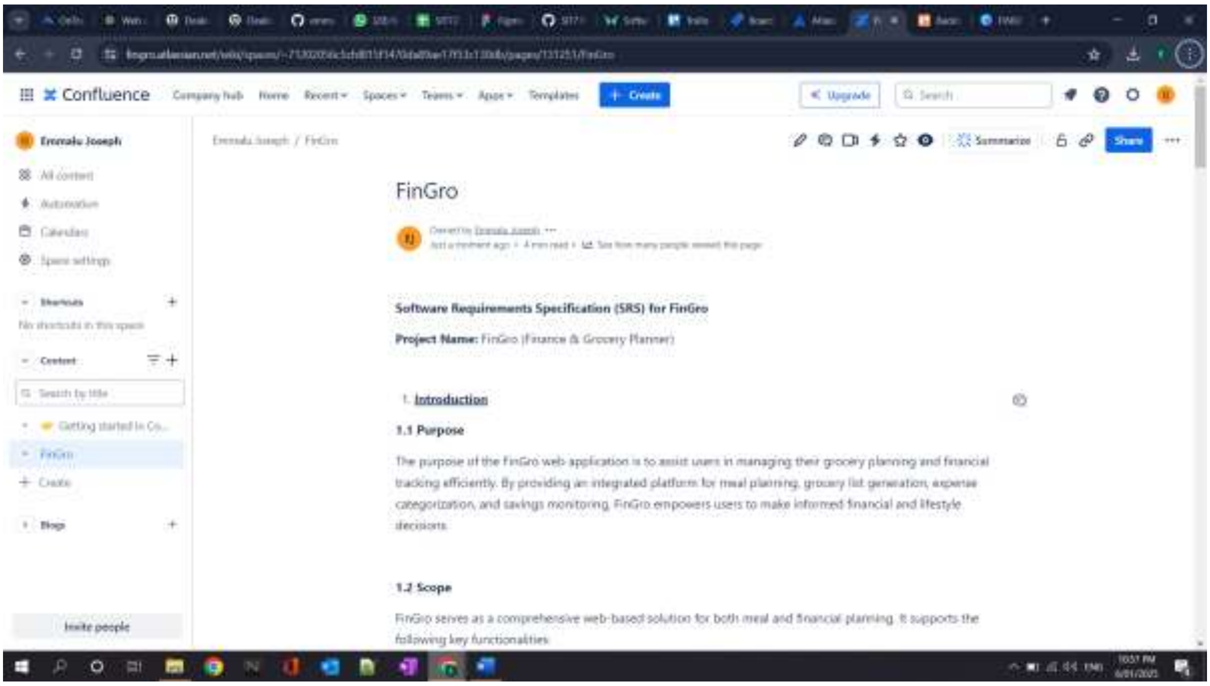
Reflection by Emmalu Joseph (S224791713): Working on the basic structure for the login, signup, and home pages allowed me to enhance my understanding of web development fundamentals. Adding the started and grocery planning functionality further honed my problem-solving skills and taught me the importance of user-centered design. Collaborating with the team was a rewarding experience, as I learned to balance independent work with group efforts. This project not only strengthened my technical skills but also boosted my confidence in handling complex requirements. It was an enriching journey that emphasized the value of planning and adaptability in achieving our goals.

Reflection by Arshdeep Singh (S224750073): Developing the Contact and About pages and initiating the smart expense tracker functionality was a challenging yet fulfilling experience. These tasks allowed me to dive deeper into designing features that cater to user needs while maintaining aesthetic appeal. I particularly enjoyed brainstorming with the team to ensure consistency across all functionalities. This project helped me improve my coding efficiency and ability to prioritize tasks. The collaborative environment fostered by the team encouraged me to share ideas openly and learn from others. Overall, this experience underscored the significance of teamwork, attention to detail, and continuous learning in achieving project success.

Reflection by Umar Khan (S223744692): Contributing CSS to the Home page and working on the Login page tasks offered me a great opportunity to focus on UI/UX design. I gained a deeper appreciation for the role of styling in enhancing user experience. The project also allowed me to refine my technical skills and problem-solving abilities when dealing with complex requirements. Collaborating with my teammates was an enlightening experience, as I learned new techniques and approaches from their work. This journey reinforced my belief in the power of collaboration and clear communication, both of which were crucial to the project's success.

Reflection by Shrikesh (S224404506): Designing and implementing CSS for the About Us and Signup pages, along with fixing various issues across the project, was a rewarding challenge. These tasks enhanced my technical expertise and adaptability, as I had to manage multiple priorities and ensure high-quality output. Working on bug fixes gave me insight into debugging processes and maintaining consistency across the project. The supportive and collaborative atmosphere of the team allowed me to learn from my peers while sharing my knowledge. This project was a valuable experience that underscored the importance of perseverance, attention to detail, and effective teamwork in delivering successful outcomes.

We are using Confluence page for documentation.



FinGro

Owned by Emmalu Joseph
Just a moment ago • 4 min read • See how many people viewed this page

Software Requirements Specification (SRS) for FinGro

Project Name: FinGro (Finance & Grocery Planner)

1. **Introduction**

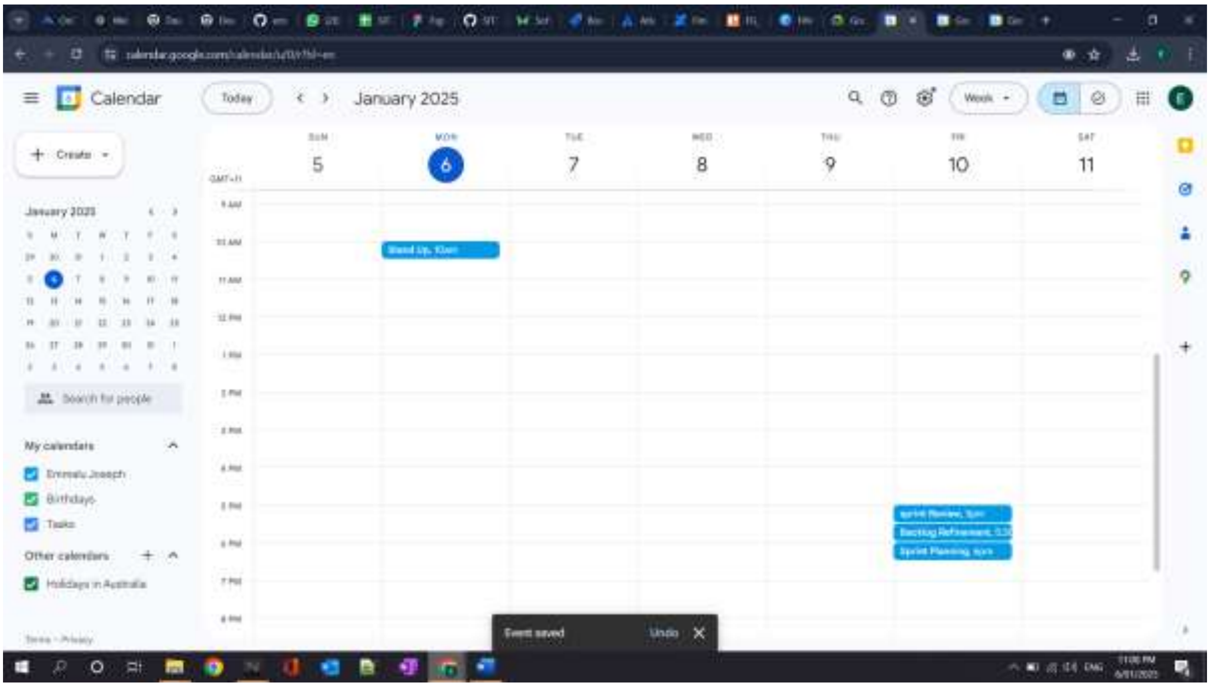
1.1 Purpose

The purpose of the FinGro web application is to assist users in managing their grocery planning and financial tracking efficiently. By providing an integrated platform for meal planning, grocery list generation, expense categorization, and savings monitoring, FinGro empowers users to make informed financial and lifestyle decisions.

1.2 Scope

FinGro serves as a comprehensive web-based solution for both meal and financial planning. It supports the following key functionalities

Meetings Scheduled on Each sprint



Reference:

Diceus. (n.d.). How to Write an Effective Software Requirements Specification (SRS)
Document. Retrieved from [https://diceus.com/custom-software-requirements-specification/].