

# Project Plan - ExpenseTracker

**Project Name:** Expense Tracker

**Team Members:** Venkateswarlu Nagineni, Quan Nguyen, Pradeep Periyasamy, Amesh Shrestha

**Project Duration:** 10/3/2025 - 11/3/2025

**Repository:** <https://github.com/CSCE-Project-2-Group-3/Project-2>

**Project Board:** <https://github.com/orgs/CSCE-Project-2-Group-3/projects/1>

---

## Initial Setup

1. Create a **GitHub repository**.
2. Create a **GitHub Project**.
3. Define **user requirements** in GitHub (Issues, Projects, or equivalent) and assign them to team members.

## Sprint Workflow

- Each member should contribute regularly (at least every other day).
  - Each sprint focused on a subset of goals.
  - Completed goals were checked off.
  - Held stand-ups daily (or every other day) to coordinate work. Questions regarding what we did, what we planned on working on next, what help we needed were raised
  - Each member contributed regularly throughout the project.
- 

## Potential Goals / Features

- **Authentication**
  - Implemented Traditional User Login & Logout System.
  - Implemented Google OAuth Login & Logout.
  - Implemented "Remember Me" Feature.
  - Implemented Delete, Update, and Forgot Password features.
  - Implemented Email Delivery for Password Reset.
- **Core Application**
  - All signup/login, expense, group, dashboard, backup-restore, and messaging pages implemented
  - Expense creation, editing, and deletion
  - Group management and membership
  - Expense categorization
  - Personal and group expense views
  - Data backup and restoration
- **Data & APIs**
  - Expense and Category models.

- CRUD operations and Excel upload.
  - Backup and Restore system.
  - **UI/UX**
    - Clean, styled forms and pages across the entire application.
    - Modern messaging app interface.
    - Flash messages for user feedback.
  - **Documentation & Testing**
    - Create and maintain a README (critical for onboarding).
    - Document codebase thoroughly.
    - Write tests for all new functions/classes (e.g., handling missing users, invalid data).
- 

## Sprint 1

Implement Traditional User Login System

Implement User Logout Functionality

Implement Google OAuth Login

Implement Google OAuth Logout

Implement Remember Me Feature to Stay Logged-In for a Period of Time

Implement Delete Account Features

Implement Update Account Features

Implement Forgot Your Password Feature

Implement Email Delivery for Password Reset Feature

[Expense Management] Create Expense and Category model

[Expense Management] Implement CRUD operations for expenses

[Expense Management] Add expense categorization feature

[Expense Management] Write unit tests for Expense model and CRUD functionality

[Expense Management] Implement Excel upload feature using Pandas

[Expense Management] Write integration tests with RSpec for Excel upload and expense

[Expense Management] Write Cucumber feature tests for expense creation, editing, deletion, and Excel upload

[Expense Management] Ability to add new categories

[Expense Management] Ability to edit existing expenses

[Expense Management] Ability to delete expenses

Create Group for Shared Expenses

Invite Users to a Group

Split Expense Among Group Members

View Total Group Expense

Comment on a Group Expense

View All Comments in a Group Expense Thread

Personal expense summary

Summary of my group expenses

Expenses filtered by date

Expenses filtered by category

Recent expenses on dashboard

Recent group expenses  
Expense dashboard

---

## **Sprint 2**

Implement Private Direct Messaging Button Between Two Users Regarding an Expense via Expense Page

Implement Private Direct Message System Between Two Users

Fix: Restore config/development.rb and Remove Misplaced Button in Landing Page

Implement a Conversation Page That Divides Contactable Users Based on Groups

Implement Multiple Expense References in Messages

Implement Group-Based Conversation System with Multi-Expense Quoting for Context

high-level AI summary of my spending habits

AI analysis of that specific widget

Flash messages (like "Success" or "Error")

Landing page

(Sign Up, Login, Edit Profile) to be clean, styled, and easy to use

Expenses pages to be clean and well-styled

Groups pages to be clean and well-styled

Conversations to look and feel like a modern messaging app

New Category form to be a simple, styled form

Implement Backup Functionality

Implement Restore Functionality

Implement Backup & Restore User Interface

Implement Backup and Restore Data System

Handle Undefined Routes with Proper Authentication

Implement Edit Feature for Group-Based Expenses

Implement Date Autofill Feature to Expense Forms

Implement Route Handling, Expense Editing Fixes, and Date Autofill Feature