CIS-344 GROUP PROJECT Project Title: Personal Finance Tracker

Moses Colon

Quinn Collazo

Chad Garcia

Project Plan

Project Scope: This application allows users to manage their personal finances by tracking income, expenses, budgets, and savings goals. It offers an intuitive interface to monitor spending habits, generate reports, and plan financially.

Objectives:

* Provide an easy-to-use platform for individuals to record and categorize financial transactions.
* Enable users to create and manage budgets.
* Track income and expenses over time.
* Visualize financial health with reports or charts.
* Emphasize data security and optimized database performance.

Initial Design Plans:

* Frontend: HTML/CSS/JavaScript for forms, tables, and UI interactions.
* Backend: PHP will handle all processing (form submission, DB interaction).
* Database: MySQL will store users, categories, transactions, and budgets.
* Use Group chat (WhatsApp) for task organization for team communication.

Technical Details

Development Process Overview

Environment Setup

* + Installed XAMPP to run Apache, PHP, and MySQL locally.
  + Created a MySQL database using phpMyAdmin.
  + Project files structured with clear folders: assets/, pages/, includes/, sql/.

Feature Breakdown

User Authentication

* + Users can register and log in.
  + PHP sessions manage user state.
  + Passwords hashed using password\_hash() for security.

Transaction Management

* + Users can add income or expense entries.
  + Each transaction is categorized (e.g., Food, Rent, Salary).
  + Validations on input types and values using both JS and PHP.

Budget Planning

* + Users can set monthly budgets per category.
  + The app shows alerts when spending exceeds the set budget.
  + A budget table is shown dynamically for each month.

Financial Summary

* + PHP and SQL aggregate data for:
    - Total income
    - Total expenses
    - Net balance

Backend Logic & Code Highlights

Add Transaction (PHP)

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$type = $\_POST['type']; // income or expense

$amount = floatval($\_POST['amount']);

$category = $\_POST['category'];

$date = $\_POST['date'];

$user\_id = $\_SESSION['user\_id'];

// Sanitize and insert

$stmt = $conn->prepare("INSERT INTO transactions (user\_id, type, amount, category, date) VALUES (?, ?, ?, ?, ?)");

$stmt->bind\_param("isdss", $user\_id, $type, $amount, $category, $date);

$stmt->execute();

}

Security Features

Inputs sanitized using htmlspecialchars() and prepared statements.

* + CSRF protection through session tokens (for future enhancement).
  + No user can access another user's data — enforced through user\_id.

Team Tools

* + Communication: Discord
  + Project Management: Trello board with labels (To-Do, In Progress, Done)
  + Version Control: Git + GitHub for commits, branching, and merging

Database Design

Overview:

The MySQL database is designed to efficiently store and retrieve user financial data such as income, expenses, budgets, and categories. It is normalized to reduce redundancy and supports optimized queries using SQL joins.

Entities and Relationships

users

* Stores user login details.
* Each user has a unique ID used in all related tables.

| Field | Type | Description |
| --- | --- | --- |
| user\_id | INT (PK, AI) | Unique identifier |
| username | VARCHAR | User’s name |
| email | VARCHAR | Email address |
| password | VARCHAR | Hashed password |

Transactions

* Records both income and expenses.
* Related to the user and the category.

| Field | Type | Description |
| --- | --- | --- |
| transaction\_id | INT (PK, AI) | Unique transaction ID |
| user\_id | INT (FK) | References users table |
| type | ENUM | 'income' or 'expense' |
| amount | DECIMAL | Transaction amount |
| category\_id | INT (FK) | References categories |
| date | DATE | Transaction date |

Categories

* Predefined or user-created categories for transactions.

| Field | Type | Description |
| --- | --- | --- |
| category\_id | INT (PK, AI) | Unique category ID |
| name | VARCHAR | Category name |
| type | ENUM | 'income' or 'expense' |

Budgets

* Allows users to set spending limits per category per month.

| Field | Type | Description |
| --- | --- | --- |
| budget\_id | INT (PK, AI) | Unique budget ID |
| user\_id | INT (FK) | References users |
| category\_id | INT (FK) | References categories |
| amount | DECIMAL | Budgeted amount |
| month | VARCHAR | E.g., '2025-04' for April 2025 |

Use of Joins

The app uses INNER JOINS to retrieve relational data across tables. Example:

View All Transactions with Category Names

SELECT t.transaction\_id, t.amount, c.name AS category, t.date

FROM transactions t

JOIN categories c ON t.category\_id = c.category\_id

WHERE t.user\_id = ?;

Monthly Budget vs. Spending

SELECT b.amount AS budgeted, SUM(t.amount) AS spent

FROM budgets b

JOIN transactions t ON b.category\_id = t.category\_id

WHERE b.user\_id = ? AND b.month = '2025-04';

These joins ensure accurate, efficient reporting of a user’s financial activity and adherence to set budgets.

User Guide: How to Use the Personal Finance Tracker

This guide walks users through the app’s features with step-by-step instructions.

Register and Log In

* Go to the homepage.
* Click Register and fill in your:
  + Username
  + Email
  + Password
* After registering, return to the login page and enter your credentials.

Add a Transaction

* From the dashboard, click “Add Transaction”
* Choose:
  + Type: Income or Expense
  + Amount
  + Category (e.g., Groceries, Rent)
  + Date
* Click Submit. Your transaction will appear in the list below.

Create or Edit a Budget

* Navigate to the Budget section.
* Select a month and choose a category.
* Enter the maximum budget amount you want to set.
* Click Save.
* You’ll get an alert if spending in that category exceeds the set amount.

View Financial Summary

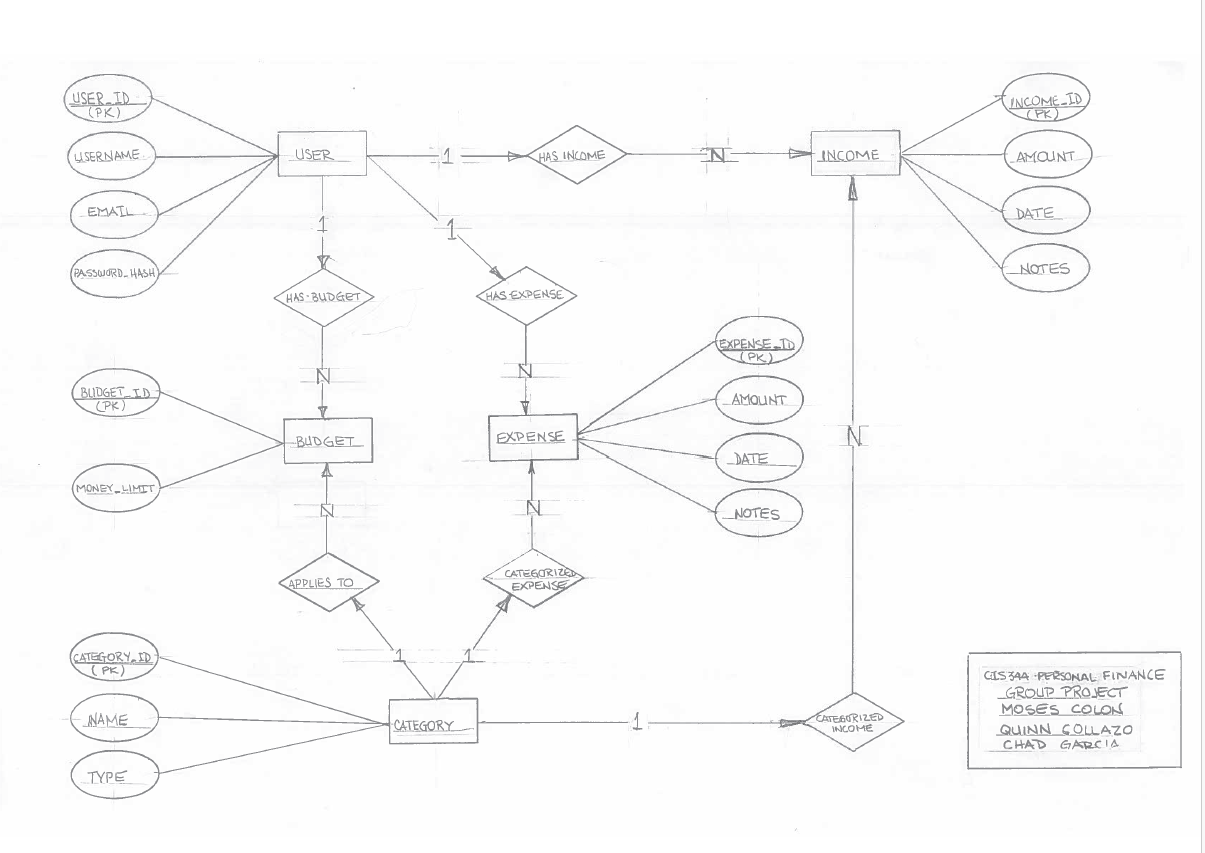
* Go to the Dashboard.
* You’ll see:
  + Total Income
  + Total Expenses
  + Net Balance
  + A pie chart showing spending by category (powered by Chart.js)

Log Out

* Click the Logout button on the top-right to end your session securely.

ER Diagram

Entity-Relationship (Chen-style) diagram for your Personal Finance Tracker.



Personal Finance Tracker:

A web-based application to help users manage income, expenses, and monthly budgets. Built for CIS 344 - Spring 2025 at Lehman College.

Features

- User registration and login

- Track income and expenses by category

- Set monthly budgets and monitor spending

- View financial summaries with charts

- Secure input validation and SQL joins for efficiency

Tech Stack

- \*\*Frontend\*\*: HTML, CSS, JavaScript

- \*\*Backend\*\*: PHP (server-side processing)

- \*\*Database\*\*: MySQL (relational schema, SQL joins)

- \*\*Tools\*\*: Trello for project management, GitHub for version control

Folder Structure

/finance-tracker/ │ ├── assets/ # Stylesheets, JS, images ├── includes/ # Header, footer, DB config, auth logic ├── pages/ # Core pages (dashboard, login, register, etc.) ├── sql/ # SQL setup scripts ├── index.php # Entry point └── README.md

How to Run the Application

1. \*\*Clone the repository\*\*

```bash

git clone https://github.com/your-username/finance-tracker.git

Set up the environment

* + Install XAMPP (or similar PHP server).
  + Start Apache and MySQL.
  + Import sql/setup.sql into phpMyAdmin.

1. Configure Database
   * Edit includes/db\_config.php to match your DB credentials.
2. Launch
   * Open your browser and navigate to http://localhost/finance-tracker/

Security Features

Input sanitization using htmlspecialchars() and prepared statements

* Passwords hashed using password\_hash()
* Session-based authentication

Contributors

* Quinn
* Moses
* Chad