X-Track

SYNOPSIS/Report

OF MINI PROJECT

BACHELOR OF COMPUTER APPLICATIONS

SUBMITTED BY

BCA 5th Semester

Shobhit Pandey

Batch Year –2023-26

Enrollment No. -U2346006

PROJECT GUIDE – Mr. Brijesh Rai

Centre of Computer Education & Training
Institute of Professional Studies
University of Allahabad, Prayagraj

Uttar Pradesh

Table of Contents

Introduction

In today's dynamic economic landscape, effective personal financial management has emerged as a critical life skill. For a significant segment of the Indian population, particularly the middle-class and dedicated competitive examination candidates, daily financial decisions directly affect their long-term stability and aim.

Many individuals still dependent on manual methods, mental accounting, or spreadsheets, all of which lead to inaccuracies, lack real-time insights, and fail to provide a wide picture of spending patterns. This factor often leads to unseen financial pressures, missed opportunities for savings, and a general disconnect from one's true financial standing. The "Expense Tracker" project is helping to bridge this gap, offering a modern, innovative, and effective solution.

Category of the project: OOPs-based

Problem Definition

The main problem addressed by this project starts from the problems that middle-class individuals and competitive exam aspirants face in maintaining a record of their everyday financial transactions. Without a clear view of where their money goes, it becomes challenging to:

- Identify unnecessary spending areas.
- Follow the declared budgets.
- Collect savings for future goals (e.g., higher education, home purchase, emergency fund).
- Understand the impact of small, seemingly unused expenditures.

Existing accounting software often becomes too complex or expensive for this work, while basic mobile notes or calculator apps lack the capabilities required for genuine financial details.

Motivation

For many people, keeping track of expenses can feel overwhelming and confusing, leading to stress and missed chances to save. That is the main reason for this project. We want to build a simple, easy-to-use tool that helps people see exactly where their money is going. Our goal is to make managing your money feel less like a chaos and more like a way to take control of your finances, helping you feel more confident and secure.

Objectives

- The aim of the "Expense Tracker" is to empower its target users with enhanced financial visibility and control.
- Provide functionality for predefined and custom expense categories, allowing users to efficiently classify their spending.

Target Audience Profile

- Middle-Class Families
- Competitive Examination Candidates

Anticipated Benefits

The successful deployment of the "Expense Tracker" is expected to deliver benefits to its users:

- Enhanced Financial Awareness
- Improved Budget Adherence
- Increased Savings Potential
- Reduced Financial Stress
- Simplicity and Accessibility

Requirement Analysis

Software Analysis

- **Frontend:** KivyMD module of Python

(KivyMD, chosen for its Pythonic nature, rich UI components, and cross-platform capabilities. The UI will be broken down into reusable components. Dedicated screens for Dashboard, Add Expense, Categories, Budgets, Reports, and Settings)

- **Backend:** Django module of Python

(Django will receive HTTP requests (GET, POST, PUT, DELETE) from the KivyMD frontend. Use of a Python MySQL connector library to interact with the MySQL database. Robust error handling for API requests)

- **Database:** MySQL (an open-source RDBMS, A MySQL database responsible for storing all application data)

- **Version Control:** Git will be used for source code management, hosted on platforms like GitHub or GitLab, to manage changes, track history, and enable collaborative development.

Study Design

The project will develop in an iterative and collaborative approach to software development that breaks projects into small, manageable increments called sprints, allowing for continuous feedback and adaptation to change.

Development will be broken down into a series of short sprints, with each sprint focusing on a specific set of features.

This iterative process will allow for continuous feedback and adaptation.

It prioritizes customer satisfaction through frequent delivery of working software, fostering a flexible, self-organizing team environment with strong communication and collaboration.

Type of Problem

The primary problem type is **Data Management**.

The core domain is **RDBMS / Data Science**.

While the initial version will not include complex machine learning, the foundation is laid for future enhancements, such as using regression or classification algorithms to predict future spending patterns or identify potential savings.

Algorithm(s) Developed or Used

Primary Algorithms for core functionality

- **CRUD**

1. **Purpose:** create, read, update, and delete the expenses of the user.
2. **Functionality:** It is the operation which takes the inputs of the user from the UI window and perform the appropriate operation according to the given request, which will be displayed in the database.

- **Data aggregation algorithms**
    1. **Purpose:** It will be used for calculating sums, averages, and displaying totals for the expenses.
    2. **Functionality:** It is an automated algorithm which calculate the sums, averages, and totals of the expenses, saving, amount, etc entered by the user.
- **Sorting and filtering**
    1. **Purpose:** This algorithm will be used to organize transaction lists.
    2. **Functionality:** This algorithm mainly helps in arranging all the details of the transactions performed by the user entered in the application and sort them according to the required criteria.

Secondary Algorithms

- **Password Hashing Algorithm**
    1. **Purpose:** Provide security to the password generates by the user.
    2. **Functionality:** It manages the password of the users by providing them a secured key which will be different for every user to identify them uniquely and restrict their access from the unknown user.
- **User Authentication Algorithm**

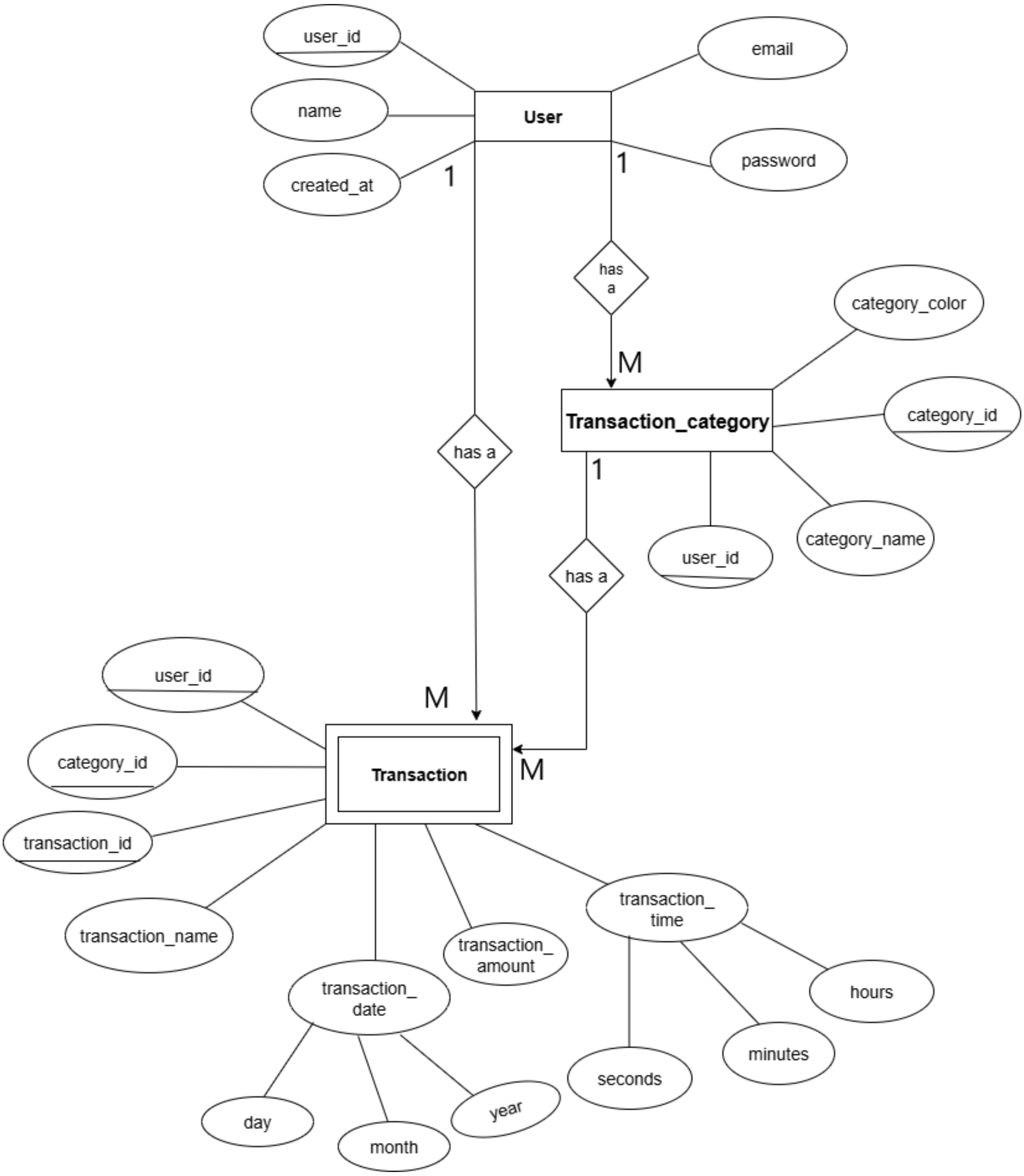
1. **Purpose:** Check that the logged in user is correct or not.
2. **Functionality:** This will validate that the user who has logged in will be the authenticated user by matching the username and password with the database.

Testing strategy used

A multi-layered testing strategy will be employed:

- **Unit Testing:** To ensure individual functions and components work as expected.
- **Integration Testing:** To verify that different modules (e.g., frontend to backend, backend to database) work together seamlessly.
- **User Acceptance Testing (UAT):** The final application will be tested by a small group of users to validate that it meets the stated objectives and is intuitive to use.

ER Diagram


user_id
name
created_at
User
email
password
1
1
has a
M
has a
Transaction_category
category_color
category_id
1
user_id
category_name
has a
M
M
user_id
category_id
transaction_id
transaction_name
Transaction
transaction_date
transaction_amount
transaction_time
day
month
year
seconds
minutes
hours

Structure

Data Structures

The project will use the following data structures to store information, managed through a **relational database**:

- **User Profile:** Stores user-specific data such as user_id, username, email, password_hash, and creation_date.
- **Categories:** Stores predefined and custom expense categories like category_id, category_name, and user_id (for custom categories).
- **Transactions:** The core data structure for storing each expense or income record. Fields will include transaction_id, user_id, amount, date, category_id, description, and type (e.g., 'expense' or 'income').

Modules Descriptions

- **Module 1:** User Management (Timeline: 2 weeks)
- **Module 2:** Transaction Management (Timeline: 3 weeks)
- **Module 3:** Transaction Category (Timeline: 3 weeks)
- **Module 4:** Expenses Management (Timeline: 2 weeks)
- **Module 5:** Testing & Debugging (Timeline: 1 weeks)
- **Module 6:** Final Deployment & User Testing (Timeline: 1 weeks)

Testing Methods Used

- **Unit Testing**:(1 week)
- **Integration testing**:(1 week)
- **User Acceptance testing:**(1 week)

Milestone

S.No.	Project Activity	Estimated Timeline/Start & End Date
1.	Synopsis	30-08-2025 & 22-09-2025
2.	Frontend	19-09-2025 & 31-10-2025
3.	Backend	01-11-2025 & 15-11-2025
4.	Project Report	30-11-2025 & 02-12-2025

Meeting With Supervisor

Date of the meet	**Mode**	**Comments of the Supervisor**	**Signature of the Supervisor**
25-08-2025	Offline	Prepare Synopsis	
15-09-2025	Offline	Synopsis Guidance	
17-09-2025	Offline	Synopsis Correction	
13-10-2025	Offline	Synopsis Presentation Discussion	

Hardware Requirements & Platform

The project will be a desktop-based application with **client-server architecture**.

- **Development Environment (Tools & Platforms):**
  - **Front-End: KivyMD module of Python** for building a dynamic user interface.
  - **Back-End: Django** framework of Python will be used for the server-side logic.
  - **Database: MySQL** will be used for data storage due to their scalability and reliability.
  - **Version Control: Git** will be used for version control, with **GitHub** as the remote repository.

Hardware Requirements

The hardware requirements are not very complex. It can be executed on Desktop and Laptops as there is an installer available for the installation which supports modern operating system of running Python 3 with KivyMD and Django framework and MySQL for the database. There are not any specialized hardware requirements.

Bibliography & References

References:

1. https://www.udemy.com/course/python-and-django-for-beginners/learn/lecture/21742454#overview
2. https://www.youtube.com/watch?v=Z0hj1_OGCz8
3. https://www.youtube.com/playlist?list=PLhTjy8cBISEoQQLZ9IBlVlr4WjVoStmy-
4. https://www.youtube.com/watch?v=JxzZxdht-XY&list=WL&index=26
5. https://www.freshbooks.com/hub/accounting/expense-tracking?srsltid=AfmBOorGA095d47pxWzMYMeJUf5BtwfM0d7DdfvT7MTbSVQ--98cqgJJ
6. https://chatgpt.com/
7. https://gemini.google.com/app