**Section 1: Metadata**

*to be filled by the student*

**1.1. Project Information** to be filled by the student

| Title: **FinTrack: A Comprehensive Accounting Management System** | |
| --- | --- |
| Section: L2 | Instructor: Mohsin Nagaria |

**1.2. Student(s) Information**

| Name: Daniyal Areshia | ID: d07605 |
| --- | --- |
| Section: L2 | Batch: 2025 |

| Name: Huzaifah Tariq Ahmed | ID: ha07151 |
| --- | --- |
| Section: L2 | Batch: 2025 |

| Name: Samiya Ali Zaidi | ID: sa07171 |
| --- | --- |
| Section: L2 | Batch: 2025 |

**Submission guideline:** Save your project proposal as a pdf file and rename as Project Proposal\_L1\_ProposedTitle where L1 is to be replaced with your section

**Section 2: The Project**

*to be filled by the student*

**2.1. Project Description:** *Please provide a brief introduction of the project including its scope.*

FinTrack is an advanced database-driven accounting system designed to streamline financial management processes for businesses of all sizes. The system provides a robust platform for managing financial transactions, generating accurate reports, and ensuring compliance with accounting standards. Built using a relational database, FinTrack enables secure storage, retrieval, and manipulation of financial data, offering users a reliable tool to monitor and manage their accounts.

FinTrack is an ideal solution for businesses seeking to enhance their accounting processes, improve financial oversight, and ensure the accuracy of their financial data.

**2.2 Functional Requirements**

*This section describes each function/feature provided by your system. These functions are logically grouped into modules based on their purposes. The users in your system must be categorized such as client, customer or administrator etc. These users will be accessing the database with the level of access that they are authorized with.*

**Module # 1 - Automated Transaction Recording Module**

This module handles the automatic recording, categorization, and updating of financial transactions.

**Feature 1a: Transaction Entry**

Description:

The system automatically records financial transactions such as sales, purchases, payments, and receipts.

User Access:

1) Administrator: Can view, edit, and delete all transactions.

2) Accountant: Can create and edit transactions but cannot delete them.

3) Auditor: Can view all transactions.

**Feature 1b: Transaction Categorization**

Description: Automatically categorizes transactions into predefined categories (e.g., revenue, expenses) based on rules set by the Administrator.

User Access:

1) Administrator: Can set and modify categorization rules.

2) Accountant: Can view and apply categorization rules.

3) Auditor: Can view categorized transactions.

**Feature 1c: Account Balances Update**

Description:

The system updates account balances after each transaction is recorded.

User Access:

1) Administrator: Can view all account balances and manually adjust them if necessary.

2) Accountant: Can view account balances but cannot adjust them.

3) Auditor: Can view account balances and audit changes.

**Module # 2 - Financial Reporting Module**

This module generates various financial reports for analysis and decision-making.

**Feature 2a: Cash Flow Statement Generation**

Description: Generates a cash flow statement detailing the inflows and outflows of cash.

User Access:

1) Administrator: Can generate, customize, and save cash flow statements.

2) Accountant: Can generate and save cash flow statements.

3) Auditor: Can generate and view cash flow statements for auditing purposes.

**Module # 3 - User Access Control Module**

This module manages user roles, access levels, and security protocols.

**Feature 3a: Role-Based Access Control**

Description: The system assigns specific roles (Administrator, Accountant, Auditor) to users, defining their access level and permissions.

User Access:

1) Administrator: Can create, modify, and delete user roles and assign permissions.

2) Accountant: No access to role management.

3) Auditor: Can view user roles and permissions for audit purposes.

**Additional Features (Tentative)**

**Feature 4a: Authentication and Authorization**

Description: The system requires users to log in with a username and password. Authorization is based on their role.

User Access:

1) Administrator: Can view login activity logs and modify authentication settings.

2) Accountant: Can log in and perform assigned tasks.

3) Auditor: Can log in and view audit logs related to user access.

**Feature 4b: Income Statement Generation**

Description: Generates an income statement summarizing revenues, costs, and expenses over a specific period.

User Access:

1) Administrator: Can generate, customize, and save income statements.

2) Accountant: Can generate and save income statements.

3) Auditor: Can generate and view income statements for auditing purposes.

**Feature 4c: Balance Sheet Generation**

Description: Generates a balance sheet showing the organization’s assets, liabilities, and equity.

User Access:

1) Administrator: Can generate, customize, and save balance sheets.

2) Accountant: Can generate and save balance sheets.

3) Auditor: Can generate and view balance sheets for auditing purposes.

**2.3. Planned Schedule:** *Kindly list the start/end dates and the timeline for the achievement of any intermediate milestones and the expected contribution to be made by the participant(s).*

| Proposal Submission - (10th September 2024) |
| --- |
| Entity Relationship Diagram - (Week 4 - 5) |
| Front end completion - (Week 6 - 7) |
| Database Building - (Week 8 - 11) |
| Front-End and Database Connectivity - (Week 11 - 13) |
| Final Debugging and Software Finalisation - (Week 14 - 15) |

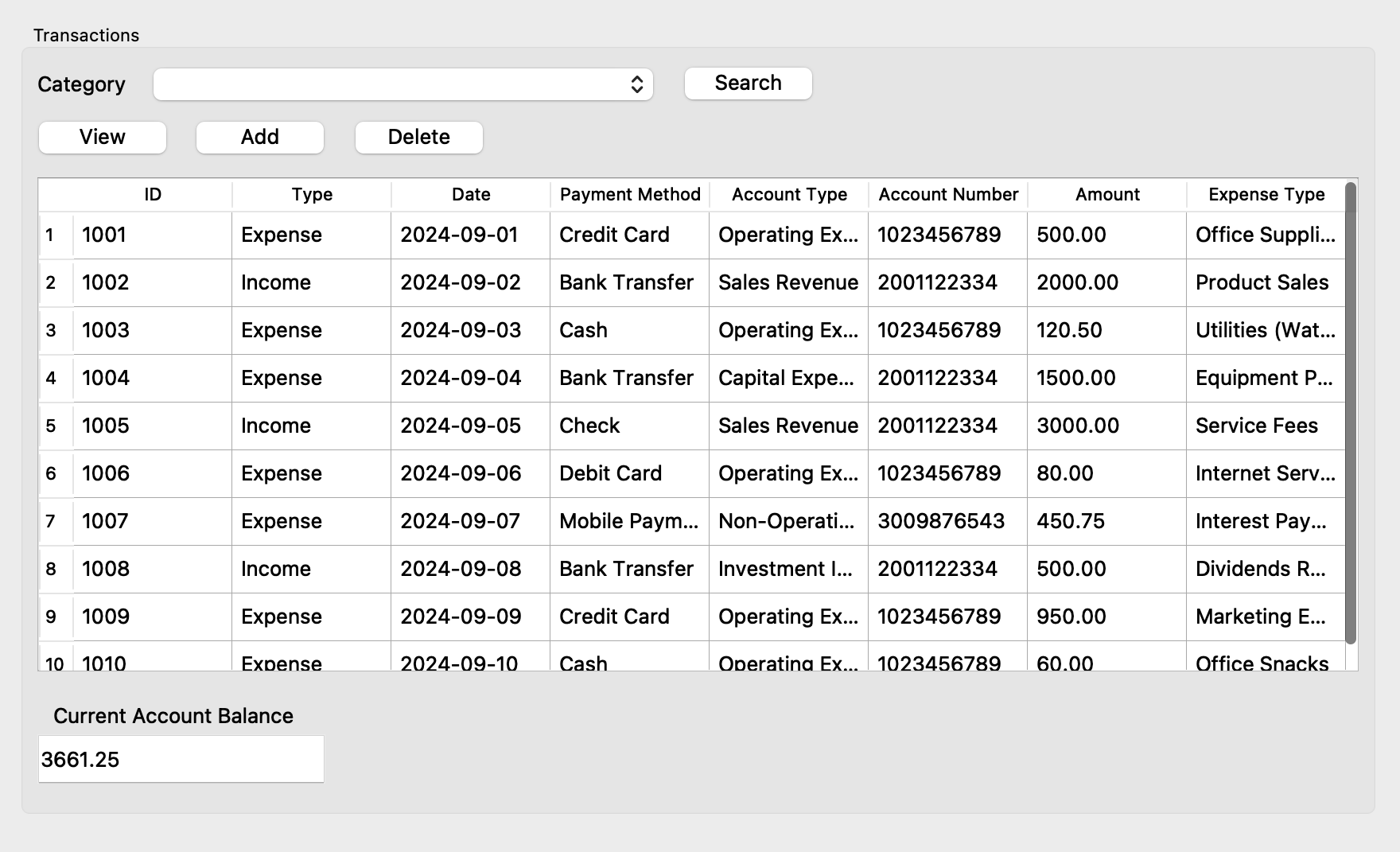
**2.4. Technology Stack:** *If you are utilizing any language or database besides PyQt and SQL Server, please complete this section; otherwise, leave it blank. Specify the programming language and database management system intended for constructing this application, as well as the application type (Desktop, Web, or Mobile).*

* DB Designer
* QT Designer
* Python

Application Type: Desktop Application

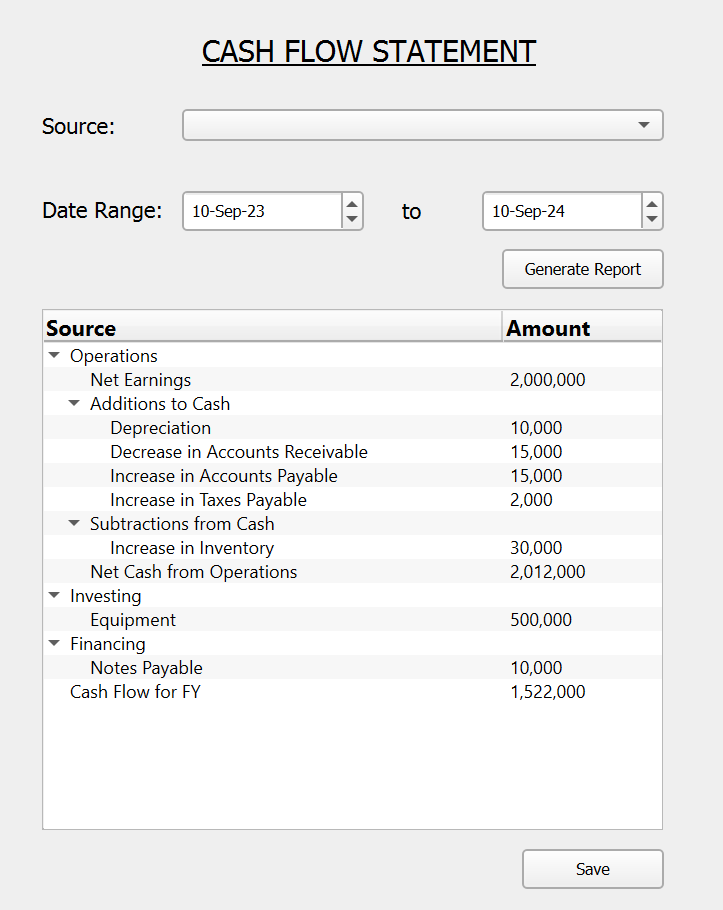
**2.5. Screens:** *Provide images of all application screens, showcasing clear input and corresponding outputs. Ensure each image includes a concise caption explaining user action and expected/observed output. You can create these screens using Qt Designer.*

**Module # 1. Automated Transaction Recording Module:**

****

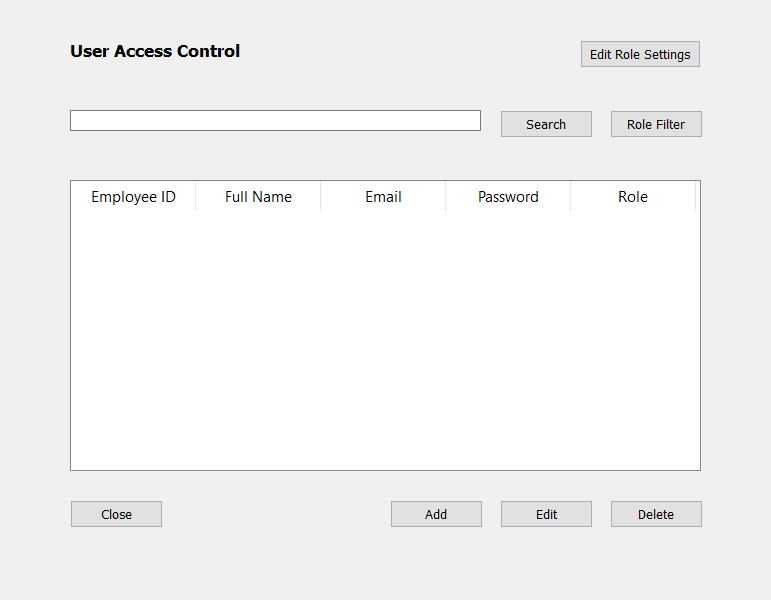
The user can view, add, or delete transactions by interacting with the buttons. The table displays the transaction details such as ID, type, date, and amount. The current account balance, calculated from all transactions, is shown below the table. The user can filter transactions by category or search for specific records.

**Module # 2. Financial Reporting Module:**

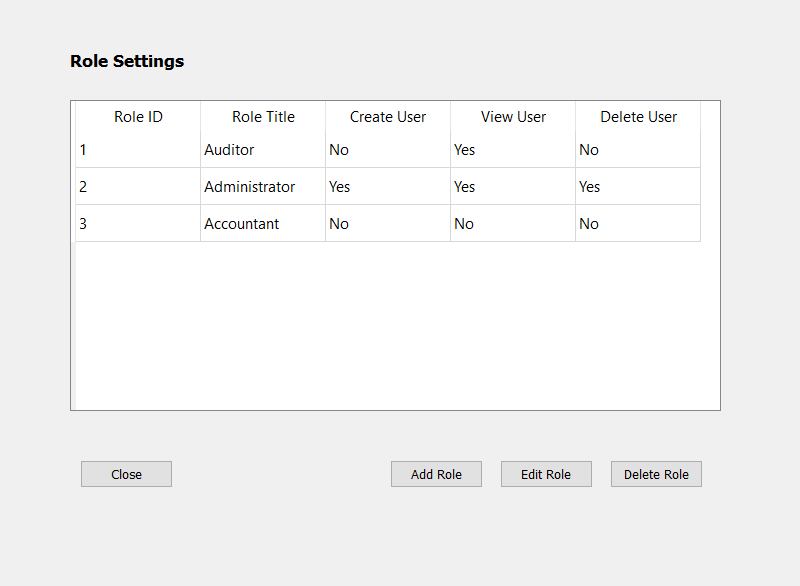


Using the above screen, our users can generate and save detailed cash flow statements within certain dates/fiscal years. Additionally, the user (admin) will have the option to “customize” the statement based on the source of the cash flow using the category box with the text ‘source’.

**Module # 3 - User Access Control Module**



Above screen will be used to *add*, *delete*, *search* and *edit* the users. One can search the employee either through their name or can list all the employees of a specific role through *role filter* button.



Above screen will be used to *add*, *edit* and *delete* the roles. It will also be used to manage and limit the role access.