[Assignment Overview 2](#_Toc185490043)

[Database Design 3](#_Toc185490044)

[Database Design - Step 1: Tables and Normalization 3](#_Toc185490045)

[Tables Design 3](#_Toc185490047)

[1. User Table 3](#_Toc185490048)

[2. Expense Table 3](#_Toc185490049)

[3. Inventory Table 3](#_Toc185490050)

[4. Sales Table 3](#_Toc185490051)

[Normalization Explained (Up to 3rd Normal Form) 4](#_Toc185490052)

[1. 1NF (First Normal Form) 4](#_Toc185490053)

[2. 2NF (Second Normal Form) 4](#_Toc185490054)

[3. 3NF (Third Normal Form) 4](#_Toc185490055)

[4. ER Diagram 4](#_Toc185490056)

[5. Class Diagram 5](#_Toc185490057)

[Implementation 7](#_Toc185490058)

[Implementation of User Management 7](#_Toc185490059)

[1. Register New User 7](#_Toc185490060)

[2. Update User Information 7](#_Toc185490061)

[3. Delete User 8](#_Toc185490062)

[4. Daily Expense Menu 9](#_Toc185490063)

[5. Record Daily Expense 10](#_Toc185490064)

[6. View Daily Expense 11](#_Toc185490065)

[7. Add Sales 12](#_Toc185490066)

[8. View Sales 13](#_Toc185490067)

[9. Overall Report 14](#_Toc185490068)

# **Assignment Overview**

This project aims to develop an expense tracking system using a multi-tier architecture. The system will include a SQLite database, a data access layer (DAL) for database interaction, a business logic layer (BLL) for core functionality, and a presentation layer (PL) for user interaction. Additionally, security, optimization, and data integrity will be prioritized.

# **Database Design**

## **Database Design - Step 1: Tables and Normalization**

### **Tables Desig**

### **Tables Design**

In this assignment, I have created several tables to efficiently manage user data, expenses, inventory, and sales transactions. Each table is designed to store specific types of data and is structured to ensure data integrity and ease of access.

### **User Table**

The Users table stores information about individual users. Each user is uniquely identified by a user\_id, which serves as the primary key. The username, password, and email fields are stored to provide a complete profile for each user. The password field is encrypted to ensure security, and the email ensures that each user can be contacted or identified uniquely.

### **Expense Table**

The Expenses table tracks all expenses recorded by users. It includes an expense\_id as the primary key to uniquely identify each expense. Additionally, the user\_id is a foreign key that links expenses to specific users. The attributes date, amount, category, and description are included to capture comprehensive details about each expense. The inclusion of categories allows users to classify expenses efficiently, and the detailed description provides further context for each transaction.

### **Inventory Table**

The Inventory table manages items that a business or individual may sell or use. Each inventory item has a unique item\_id (primary key). The item\_name, quantity, and cost fields are included to manage stock levels and the cost associated with each item. These attributes ensure that inventory data is maintained accurately for business operations.

### **Sales Table**

The Sales table records sales transactions, linking them to specific users through the user\_id foreign key. The primary key sale\_id ensures that each sale transaction is distinct. Other fields such as date, amount, and items\_sold provide a comprehensive overview of sales activities. Items\_sold specifically captures the items involved in each transaction, ensuring that the sales history is detailed and informative.

## **Normalization Explained (Up to 3rd Normal Form)**

### **1NF (First Normal Form)**

In the first normal form, all tables must contain atomic values, meaning that each attribute (column) should hold indivisible, non-repeating values. This is essential to prevent redundancy and ensure consistency across the data. For example, the description field in the Expenses table should only store a single narrative without including multiple values or categories.

### **2NF (Second Normal Form)**

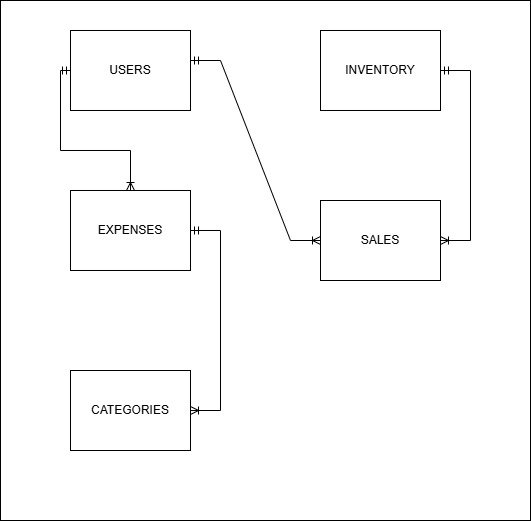
Second Normal Form builds on the principles of 1NF by ensuring that all non-key attributes (fields) are fully functionally dependent on the primary key. In simpler terms, for each table, every non-primary key attribute should depend solely on the primary key. In the Expenses table, the fields date, amount, category, and description are dependent entirely on the combination of expense\_id and user\_id, ensuring no partial dependencies are present.

### **3NF (Third Normal Form)**

Third Normal Form addresses transitive dependencies. Once a table is in 2NF, it must be ensured that all non-key attributes are not only dependent on the primary key but also independent of other intermediate attributes. In the Sales table, for example, items\_sold is only dependent on the sale\_id and not on user\_id or other related fields. This design eliminates unnecessary dependencies, reducing redundancy and simplifying the data model.

### **ER Diagram**

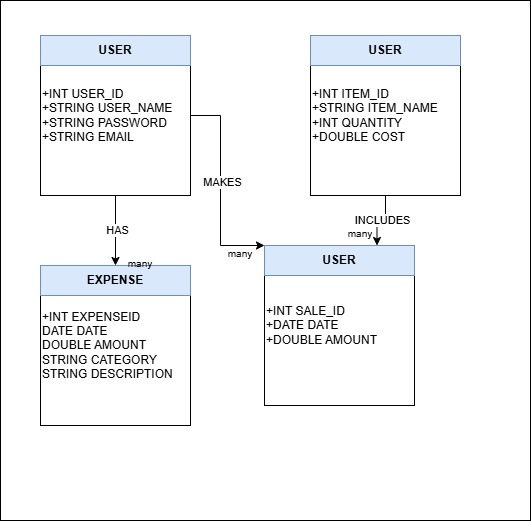
The Entity-Relationship (ER) Diagram is a conceptual representation of the data and its relationships for the expense tracking system. Below is a diagram of each component and their relationships.



**ER Diagram of Expense Tracking**

### **Class Diagram**

The Class Diagram is a structural diagram that provides a blueprint for the system's implementation. It outlines the system's key classes, their attributes, methods (operations), and the relationships between them. Below is a diagram for the expense tracking system.



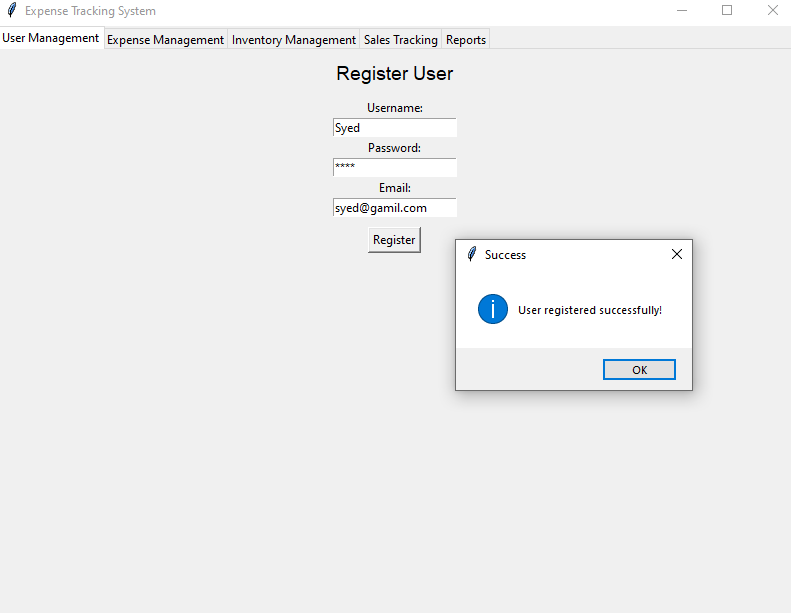
**Class Diagram of Expense Tracking**

# **Implementation**

## **Implementation of User Management**

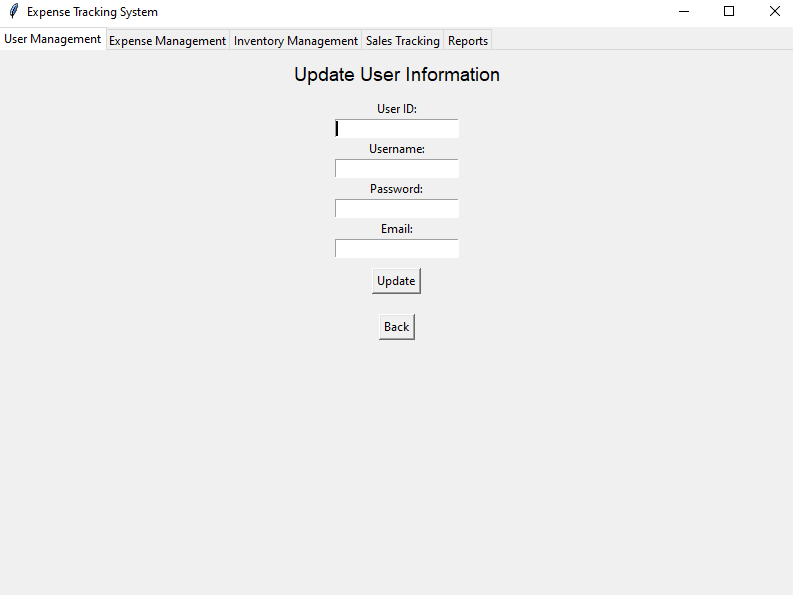
### **Register New User**

In the image below, the user registration process is illustrated.



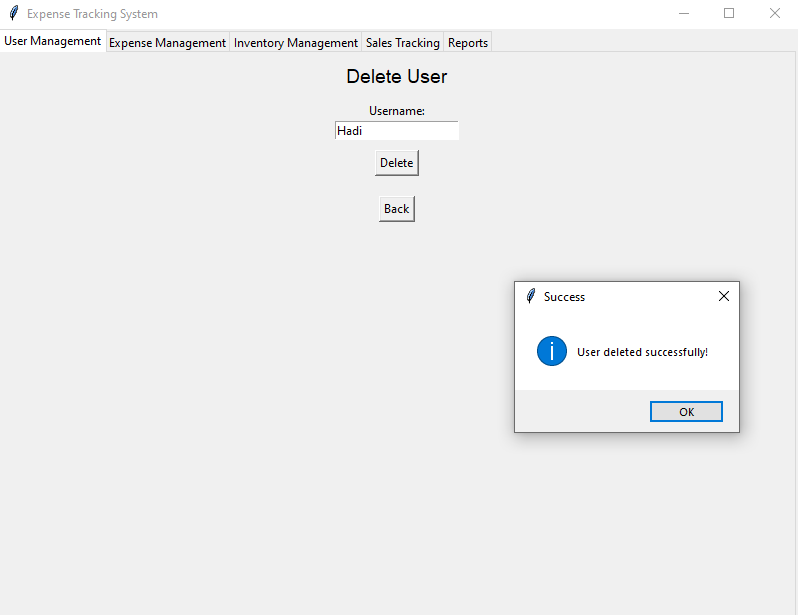
### **Update User Information**

In the below image, User information can be update by using User\_ID.



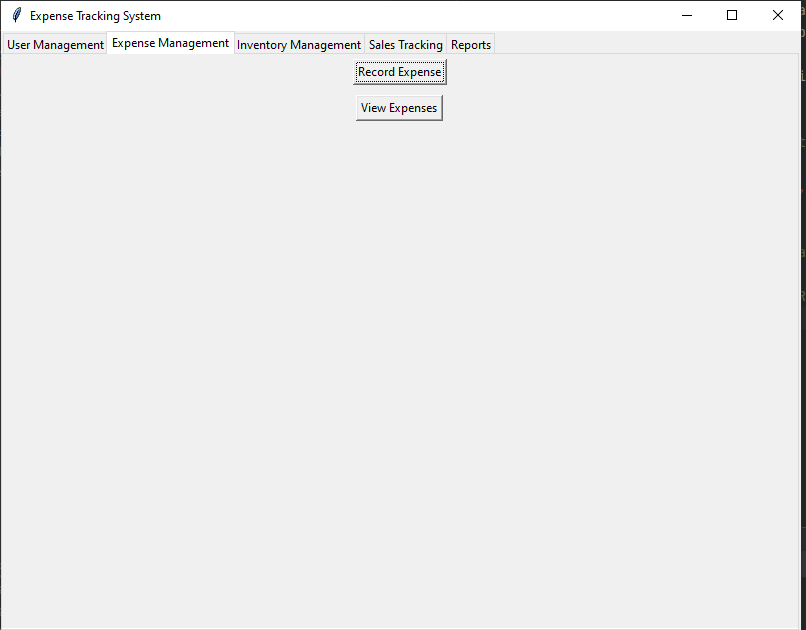
### **Delete User**

In the below image, User can b deleted by using UserName.



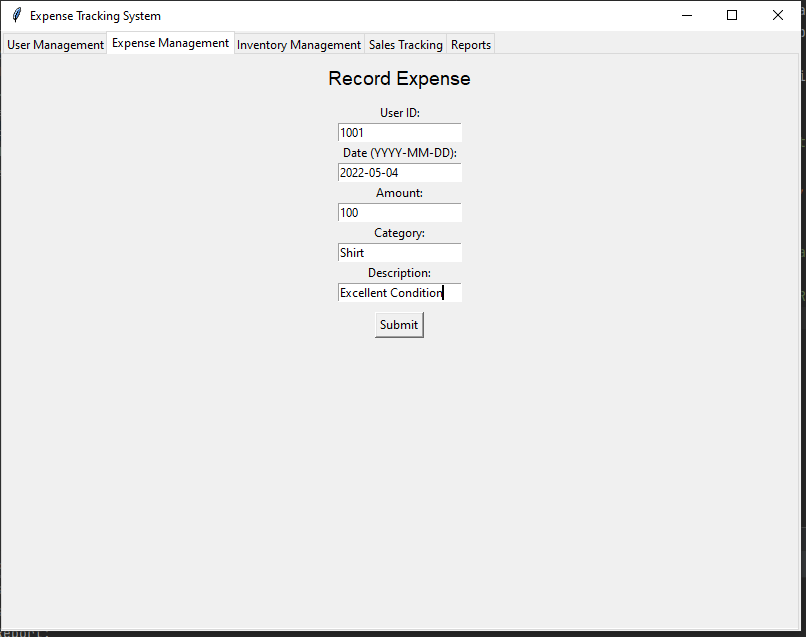
### **Daily Expense Menu**

In the image below, the expense menu is displayed. Here, users can record their daily expenses and also view their expense history.



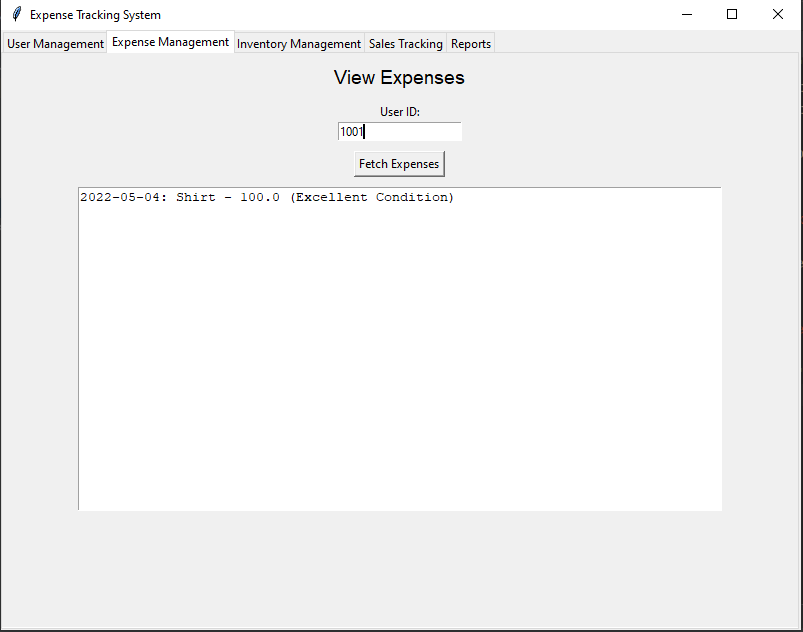
### **Record Daily Expense**

In the image below, daily expense records have been entered, and this data is securely stored in the backend using SQL. The system ensures that each entry is properly validated before saving, allowing users to maintain an organized and accurate record of their expenses. These records can be retrieved, analyzed, and managed efficiently, providing valuable insights into daily spending habits



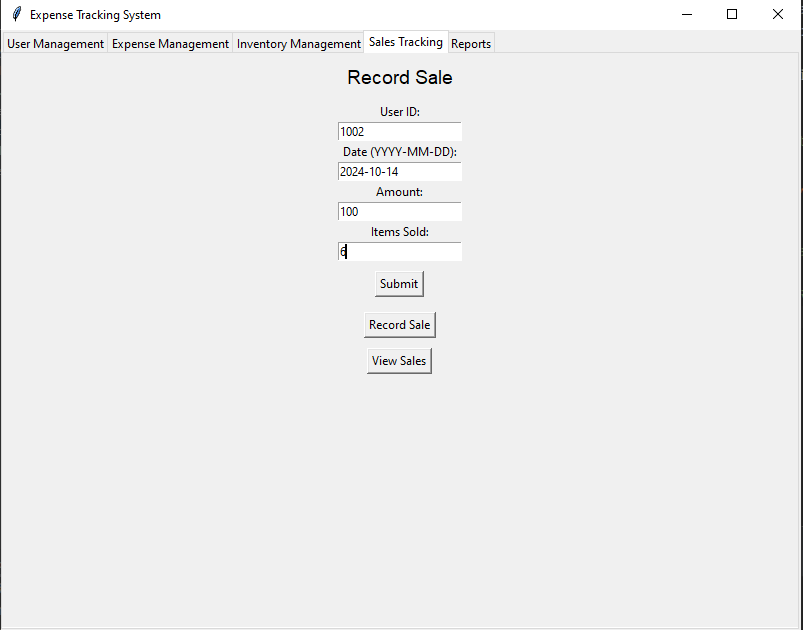
### **View Daily Expense**

In the image below, the daily expense records are displayed, showing a detailed view of the user's spending history. This feature allows users to review their expenses with ease, providing a clear breakdown of all recorded transactions. The data is fetched from the backend using SQL, ensuring accurate and up-to-date information. Users can utilize this view to monitor their spending patterns, identify trends, and manage their finances effectively.



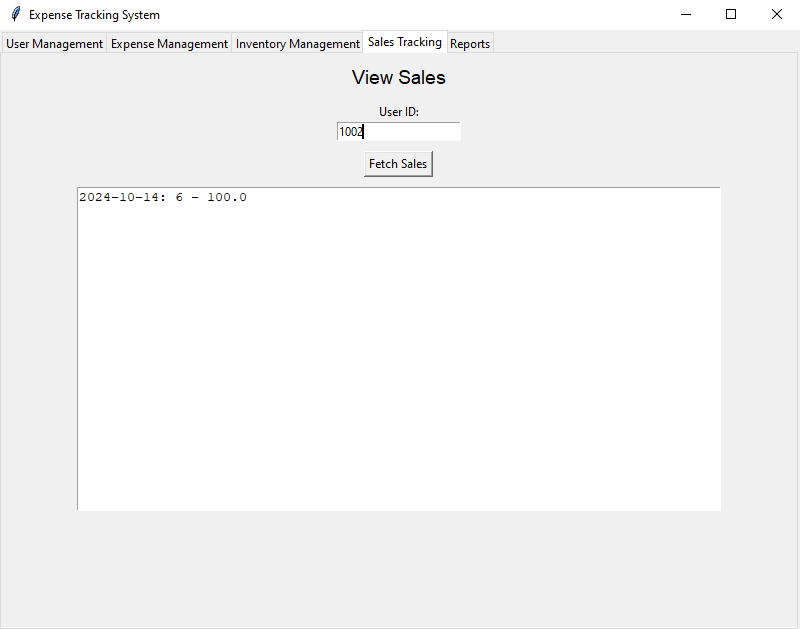
### **Add Sales**

In the image below, the sales entry feature is displayed, allowing users to add new sales records. This functionality enables users to input details such as the date, item, quantity, price, and total amount. Once entered, the data is securely stored in the backend using SQL, ensuring it is accurately recorded and readily available for future reference. This feature helps users keep track of their sales activities and maintain an organized record for analysis and reporting purposes.



### **View Sales**

In the image below, the sales records are displayed, providing users with a detailed view of all recorded sales transactions. This feature fetches data from the backend using SQL, ensuring accurate and up-to-date information is presented. Users can review key details such as the date, item, quantity, price, and total amount for each sale. This organized view helps users monitor their sales performance, analyze trends, and make informed business decisions.



### **Overall Report**

In the image below, the overall report is displayed, offering a comprehensive summary of all recorded data, including expenses and sales. This feature consolidates information from the backend using SQL, ensuring accurate and real-time insights. Users can view key metrics such as total expenses, total sales, profit margins, and other relevant statistics. The overall report helps users analyze financial performance, identify trends, and make data-driven decisions to improve efficiency and profitability.

