**A**

**PROJECT REPORT**

**ON**

**Expense-tracker Website**

**SUBMITTED BY**

Hardik Kanajariya

Vora Chirag

**ACADEMIC YEAR 2024-25**

**M.S.C.IT & CA SEM - 2**

**UNDER THE GUIDANCE OF**

Ms. Reetu Tejvani

**SMT. C.Z.M GOSRANI B.C.A. COLLEGE**

**JAMNAGAR**

**SUBMITTED TO**



**SAURASHTRA UNIVERSITY**

**- RAJKOT**

**A**

**PROJECT REPORT**

**ON**

**Expense-tracker Website**

**SUBMITTED BY**

Hardik Kanajariya

Vora Chirag

**ACADEMIC YEAR 2024-25**

**M.S.C.I.T & CA SEM - 2**

**UNDER THE GUIDANCE OF**

Ms. Reetu Tejvani

**SMT. C.Z.M GOSRANI B.C.A. COLLEGE**

**JAMNAGAR**

**SUBMITTED TO**



**SAURASHTRA UNIVERSITY**

**- RAJKOT**

# **ABSTRACT**

The **Expense Tracker** project aims to provide a comprehensive solution for individuals and organizations to manage their expenses effectively. In today's fast-paced world, tracking expenses accurately is essential for financial planning and budgeting. This project offers a user-friendly interface coupled with powerful features to simplify expense management.

The Expense Tracker system allows users to record their expenses across various categories such as food, transportation, utilities, entertainment, etc. Users can input expenses manually or import them from bank statements or other financial sources. The system also supports recurring expenses and provides reminders for upcoming payments.

One of the key features of the Expense Tracker is its robust reporting capabilities. Users can generate detailed reports and visualizations to analyze their spending patterns over time. This insight helps users make informed decisions and identify areas where they can save money or optimize their expenses.

Security is paramount in the Expense Tracker system. It employs industry-standard encryption techniques to safeguard sensitive financial information. Additionally, user authentication and authorization mechanisms ensure that only authorized individuals can access and modify expense data.

Overall, the Expense Tracker project aims to empower users with the tools and insights they need to take control of their finances and achieve their financial goals.

# **ACKNOWLEDGEMENT**

We feel great pleasure in submitting this project report as a part of our MSCIT&CA Semester 2 curriculum. Practical study plays an important role.

For the successful completion of our project, we would especially like to thank our parents for their support and unconditional help. We would also like to thank our Project Guides Ms. Reetu Tejvani for their constant support and help in implementation of this project.

We are also thankful to our Principal Ma'am, Ms. Hetal G. Savla for all the facilities they provided throughout our semester and for encouraging us to take up this activity.

Lastly, we would also like to thank the faculties and staff members of Smt. C.Z.M. Gosrani College, Jamnagar

# **PROJECT PROFILE**

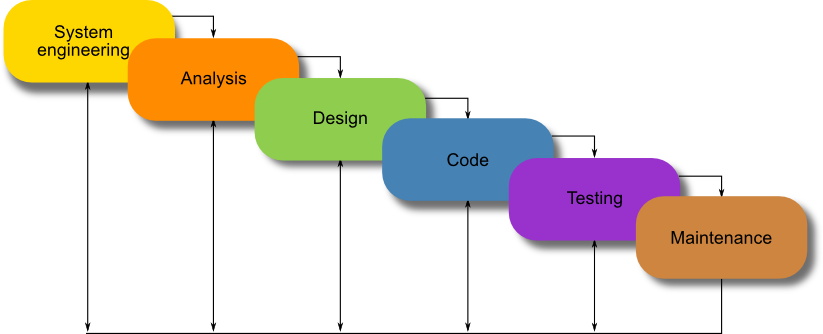
|  |  |  |
| --- | --- | --- |
| **STUDENT INFORMATION** | | |
| **Name**  Hardik Kanajariya  Vora Chirag | **Enrollment Numbers**  23SI032PG00359  23SI032PG00369 | |
| **PROJECT DETAILS** | | |
| **Project Title** | Expense-tracker | |
| **Duration** | 4 Months | |
| **Name of Project** | Expense-tracker | |
| **Platform** | React,  Laravel,  MySQL | |
| **Team Size** | 2 | |
| **GUIDE INFORMATION** | | |
| **Name of Guide** | | Ms. Reetu Tejvani |

# **INDEX**

|  |  |  |
| --- | --- | --- |
| Chapter | Title | Page No. |
| 1 | **Overview of the accepted SDLC Model** | 1 |
| 2 | **Requirement Gathering and Analysis** |  |
|  | 2.1 Organization Details | 2 |
| 2.2 Meetings | 2 |
| 2.3 Type of Project | 3 |
| 2.4 Method of collecting requirements | 3 |
| 3 | **System Requirement Specification** |  |
|  | 3.1 Introduction | 4 |
| 3.1.1 Purpose | 4 |
| 3.1.2 Scope | 4 |
| 3.1.3 Operating Environment | 4 |
| 3.1.4 User classes | 5 |
| 3.2 System Modules | 5 |
| 3.3 Feasible study | 6 |
| 3.3.1 Operational Feasibility | 5 |
| 3.3.2 Technical Feasibility | 6 |
| 3.3.3 Economical Feasibility | 6 |
| 3.3.4 Schedule Feasibility | 7 |
| 3.3.5 User interface | 7 |
| 3.4 Non-functional Requirements | 8 |
| 3.4.1 Performance Requirements | 8 |
| 3.4.2 Security Requirements | 8 |
| 4 | **System Analysis and Modeling** |  |
|  | 4.1 Normalization | 10 |
| 4.2 Data Dictionary | 11 |
| 4.3 Data Flow Diagram | 16 |
| 4.3.1 Context(0 level) Diagram | 17 |
| 4.3.2 1st Level Diagram | 18 |
| 4.4 E-R Diagram | 19 |
| 4.5 Use-case Diagram | 20 |
| 4.5 Gantt Chart | 21 |
| 5 | **Test Cases** | 22 |
| 6 | **Screenshots** | 23 |
| 7 | **Limitations and Future Enhancements** | 37 |
| 8 | **Conclusion** | 38 |
| 9 | **References and Bibliography** | 39 |

**CHAPTER 1**

**OVERVIEW OF THE ACCCEPTED SDLC MODEL**

****

**Waterfall Model**

* Each step of the SDLC life cycle is followed in a linear approach.
* Upon the completion of the step, the next step is started.
* Cannot go to the previous page.

**CHAPTER 2**

**REQUIREMENT GATHERING AND ANALYSIS**

1. **Organization details**
   * **Name of organization:**
     + For Expense-trecker
   * **Brief details of the organization:** 
     + Expense-trecker is a reputed company.
     + They have been operating since the year 2002.
2. **Meetings**
   * **Meeting with a Manager:**
     + **Reason to create this web application**

* The primary reason for developing the Expense Tracker web application is to assist individuals and organizations in managing their finances more effectively. By providing a centralized platform to track expenses, users can gain better control over their spending habits and make informed financial decisions.

.

* + - **Name of the Manager:**
      * Mr. Abhay Kachhdiya
    - **Requirements of the manager in their own words:**
      * Users should be able to add the expense, and income
      * User Can show our income and expense for date wise
      * Show Graph income and expense.
      * Admin can add Users

1. **Type of project**
   * The system will be in the form of application and accessible through the internet.
2. **Method of collecting requirements**
   * The System Requirements will be collect by interviewing the users of the proposed system.

**CHAPTER 3**

**SYSTEM REQUIREMENT SPECIFICATION**

* 1. **Introduction**
* Expense-trecker is a web application that allows the Users of Add their income and expense and show graph.
  + 1. **Purpose**
  + The purpose of this software Requirement specification document is to provide a detailed description of the functionalities of tourism.
    1. **Scope**
       - This system will be used by E-Commerce Companies and can be used by any common public.
    2. **Operating Environment**
* **Client-side requirements:**
  + To use the application, a computer with an internet connection is required
* **Server-side (hosting) requirements:**
  + Operating system: Linux hosting platform
  + Memory:4 GB or higher
  + Storage: 128GB GB or higher
  + Server Software: Apache 2.2+ Server, MySQL 5.x+ , PHP 5.2+
    1. **User Classes**
       - **Administrators**
         * Can add, update and remove user and other administrators.
       - **User**
         * Add income and expense.
  1. **System Modules**
     + **Authentication**
       - Users, and Admins can log in to the system by entering proper credentials.
     + **View income**
       - Available to Users.
       - Users can view the Product of any country.
     + **Add income**
       - Available to Users.
       - Users can view and purchase product product.

* + - **Update income.**
      * Available to Admin.
      * Create a new vendor and get this appropriate rights.
    - **Manage income**
      * Available for Administrator.
      * Administrators can add, update and remove Products.
    - **Add expense**
      * Available for Administrator.
      * Administrators can add, update and remove Vendor.
    - **Update income and expense both**
      * Available for Administrator.
      * Administrators can add, update and remove Users.
* **User interface Requirements**
* Stable internet connection
  1. **Feasibility study**

A feasibility study is a preliminary investigation of a proposed system to decide whether the system can run smoothly with organization.

* + 1. **Operational Feasibility:**

Operational feasibility is the measure of how well a proposed system solves the problems, and take advantage of the opportunities identified in the requirements analysis phase of the system development.

**3.3.2 Technical Feasibility:**

Technical feasibility determines whether the work for the project can done with existing equipment, software technology and available personal.

* + *Operating Environment*
  + *Android : 6+ Version*
  + *Minimum RAM : 1GB*
  + *Storage : 10GB*
* **Hardware Interface Requirements**
  + - * 512+ MB of RAM
      * Multimedia keyboard and Mouse
      * 1.0+ GHz Processor
      * Monitor
* **Software Interface Requirements**
  + - * Operating System: Microsoft Windows 7 or Above
      * Front End Tools: PHP, HTML, Bootstrap
      * Back End Tools: MySQL
      * Browser specification: Google Chrome
    1. **Economic Feasibility:**

Economic feasibility determines whether there are sufficient benefits in creating to make the cost acceptable, or is the cost of the system too high.

So signifies cost benefit analysis and savings on the behalf of the cost benefit analysis.

* + 1. **Schedule Feasibility:**

How long will it take to get the technical expertise?

We may have the technology, but that doesn`t mean we have the skill required to properly apply that technology.

* + 1. **User Interface**
* In case of user is not registered. He/She can enter the details and create an account. Once the account is created user can ‘login’. If the username or password is incorrect then an error message appears.
  1. **Non-functional Requirements**
     1. **Performance Requirements**
* Overall System should be fast and error-free.
* It should have built-in error checking and correction facilities.
* The system should be able to handle large amounts of data.
  + 1. **Security Requirements**
* Users must Have Proper permission to perform tasks.
* Administrators and Hotel users must provide a Password when logging in.
* Administrator has more rights than the user.

**SYSTEM ANALYSIS AND MODELING**

**4.1 Normalization**

Normalizaion to organizing the columns

(Attributes) and tables (relations) of a relational database to

Reduce data redundancy and improve data integrity.

It is not possible to define here All Normalization That’s why it is show in reference book.

**4.2 Data Dictionary**

A data dictionary contains metadata i.e data about the database. The data dictionary is very important as it contains information such as what is in the database, who is allow to access it, where the database physically stored, etc. The users of the database normally do not interact with the data dictionary, it is only handle by the database administrators.

**Database Name:expensetraker**

1.Table Name:- currencies

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constrain |
| **id** | Integer | Primarykey |
| **currency\_code** | Varchar |  |
| **currency\_name** | Varchar |  |
| **country** | Text |  |
| **created\_at** | Timestamp |  |
| **updated\_at** | Timestamp |  |

2. Table Name:- failed\_jobs

|  |  |  |
| --- | --- | --- |
| Column Name | DataType | Constrain |
| **id** | Bigint | Primary key |
| **uuid** | Varchar | Foreign key |
| **connection** | Text |  |
| **queue** | Text |  |
| **payload** | Longtext |  |
| **exception** | Longtext |  |
| **failed\_at** | Timestamp |  |

3. Table name:- income\_expenses

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constrain |
| **id** | Inetger | Primarykey |
| **category\_id** | Inetger | Foreignkey |
| **currency\_id** | Integer | Foreignkey |
| **source** | Varchar |  |
| **spent\_on** | Varchar |  |
| **remarks** | Text |  |
| **amount** | Double |  |
| **transaction\_date** | Datetime |  |
| **transaction\_type** | text |  |
| **created\_by** | Integer |  |
| **updated\_by** | Integer |  |
| **created\_at** | Timestamp |  |
| **updated\_at** | Timestamp |  |

4. Table Name:- migrations

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Constrain |
| **id** | Integer | Primarykey |
| **migration** | Varchar |  |
| **batch** | Integer |  |

5. Table Name:- password\_reset\_tokens

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Constrain |
| **email** | Varchar | Primarykay |
| **token** | Varchar |  |
| **created\_at** | Timestamp |  |

6. Table Name:- personal\_access\_tokens

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Constrain |
| **id** | integer | Primarykey |
| **tokenable\_type** | Varchar | Unique key |
| **tokenable\_id** | Integer | Foreignkey |
| **name** | Varchar |  |
| **token** | Varchar | Uniquekey |
| **abilities** | Text |  |
| **last\_used\_at** | Timestamp |  |
| **expires\_at** | Timestamp |  |
| **created\_at** | Timestamp |  |
| **updated\_at** | Timestamp |  |

7. Table Name:- transaction\_categories

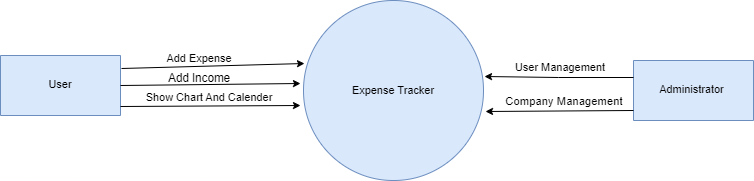
|  |  |  |
| --- | --- | --- |
| Column name | Data type | Constrain |
| **id** | Integer | Primarykey |
| **category\_name** | Varchar |  |
| **category\_type** | Text |  |
| **created\_by** | Integer |  |
| **updated\_by** | Integer |  |
| **created\_at** | Timestamp |  |
| **updated\_at** | Timestamp |  |

8. Table Name: - users

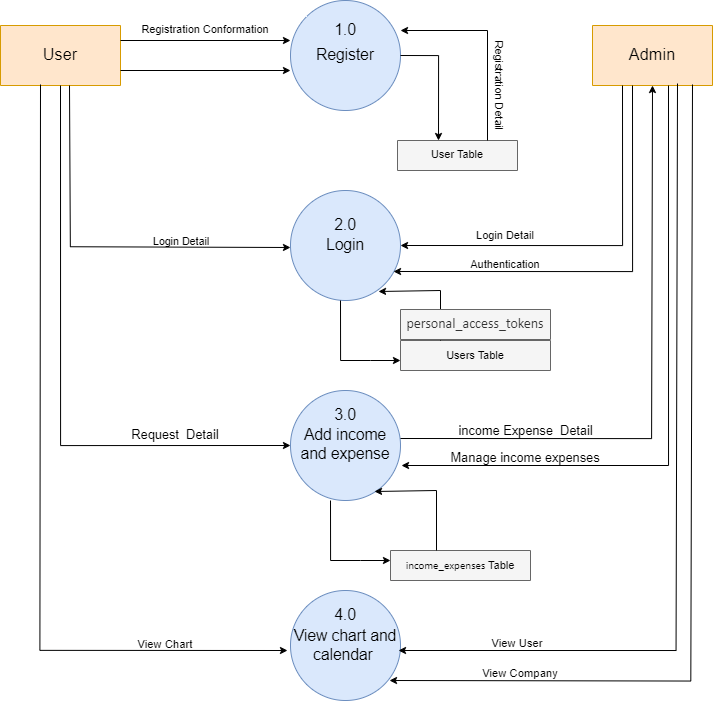
|  |  |  |
| --- | --- | --- |
| Column name | Data type | Constrain |
| **id** | Inetger | Primary key |
| **name** | Varchar |  |
| **email** | Varchar | Uniquekey |
| **password** | Varchar |  |
| **currency\_id** | Integer | Foreign key |
| **is\_admin** | Boolean |  |
| **remember\_token** | Varchar |  |
| **created\_at** | Timestamp |  |
| **updated\_at** | Timestamp |  |

**4.3 Data Flow Diagram**

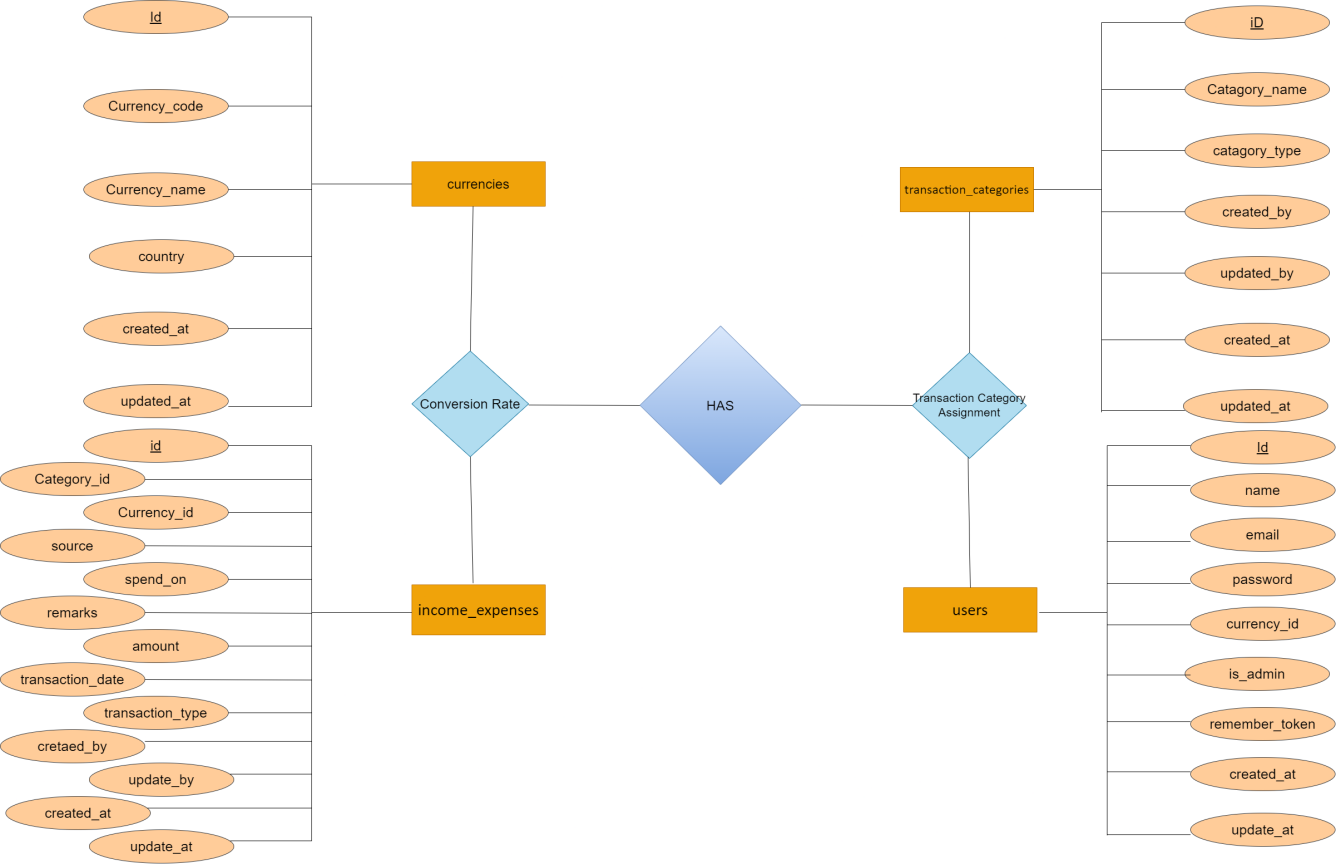
**4.3.1 Context level Diagram**



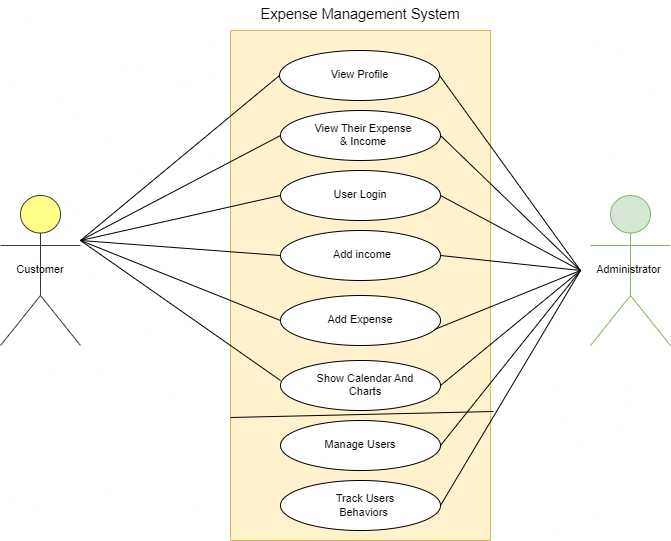
**4.3.2 1st Level Diagram**



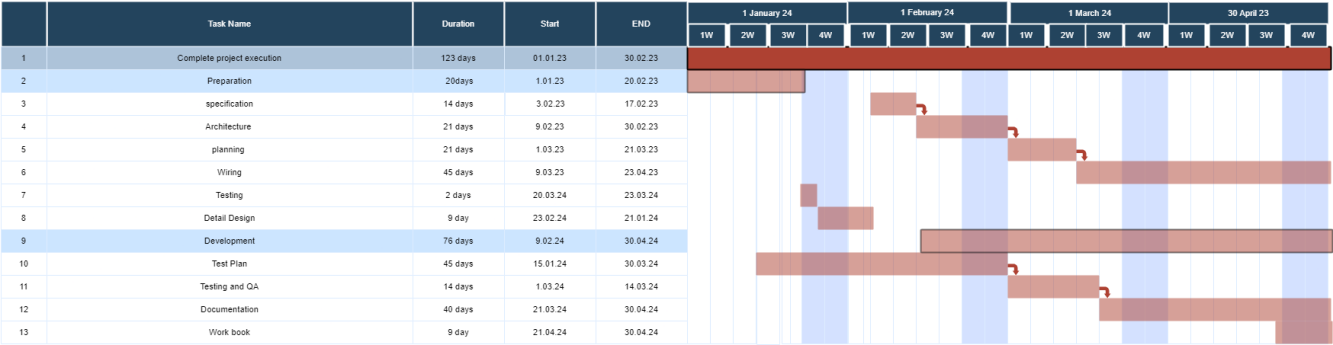
**4.3 E-R Diagram**



**4.5 Use-case Diagram**



**4.5 Gantt Chart**



**5-TSET CASES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Login Module | | | | | |
| TEST CASE No. 1 | | | TEST CASE DESIGN DATE :  1/june/2023 | | |
| TEST TITLE : Login Module Test | | | TEST CASE EXECUTION DATE :  1/june/2023 | | |
| DESCRIPTION : In this test case Login module would be tested | | | | | |
| PRECONDITION : Login Activity must be working | | | | | |
| STEP  No. | TEST STEP | TEST DATA | EXPECTED RESULT | ACTUAL RESULT | STATUS |
| 1 | Navigate to Login Page | Login Activity | Login Activity is  visible | Same As Expected | PASS |
| 2 | Enter Email | darsan@gmail | Email is accepted | Invalid Email | FAIL |
| 3 | Enter Password | User123 | Password is accepted | Same As Expected | PASS |
| 4 | Click Login Button | Button Click | Check Credentials | Same As Expected | PASS |

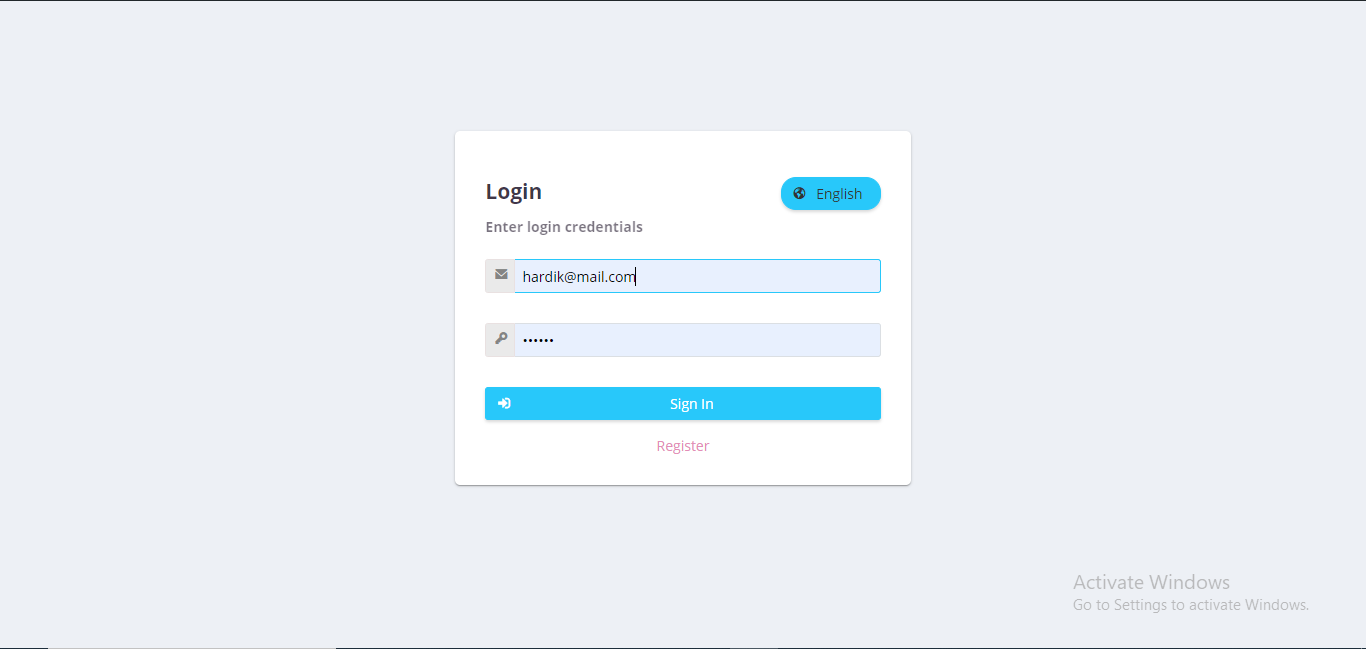
\* Note: - Email has not accepted in above test case due to invalid format.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signup Module | | | | | |
| TEST CASE No. 1 | | | TEST CASE DESIGN DATE :  5/june/2023 | | |
| TEST TITLE : Signup Module Test | | | TEST CASE EXECUTION DATE :  5/june/2023 | | |
| DESCRIPTION : In this test case Signup Activity would be tested | | | | | |
| PRECONDITION : Signup Activity must be working | | | | | |
| STEP  No. | TEST STEP | TEST DATA | EXPECTED RESULT | ACTUAL RESULT | STATUS |
| 1 | Navigate to Sign up activity | Signup Activity | Signup Activity is  visible | Same As Expected | PASS |
| 2 | Enter All Fields | Data | Check Data Integrity | Same As Expected | PASS |
| 3 | Click Sign up Button | Button Click | Signup done  With all Validation | Same As Expected | PASS |

**SCREENSHOT**

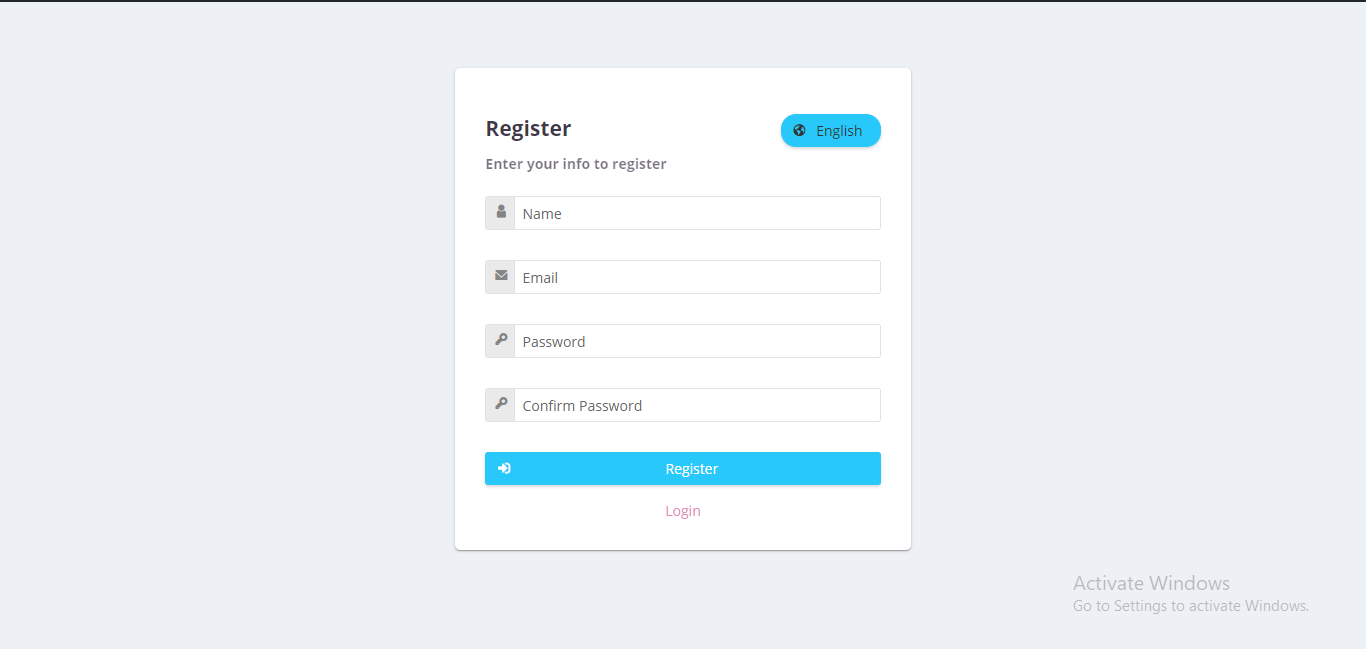
* **User side**

1. **Login Page**

****

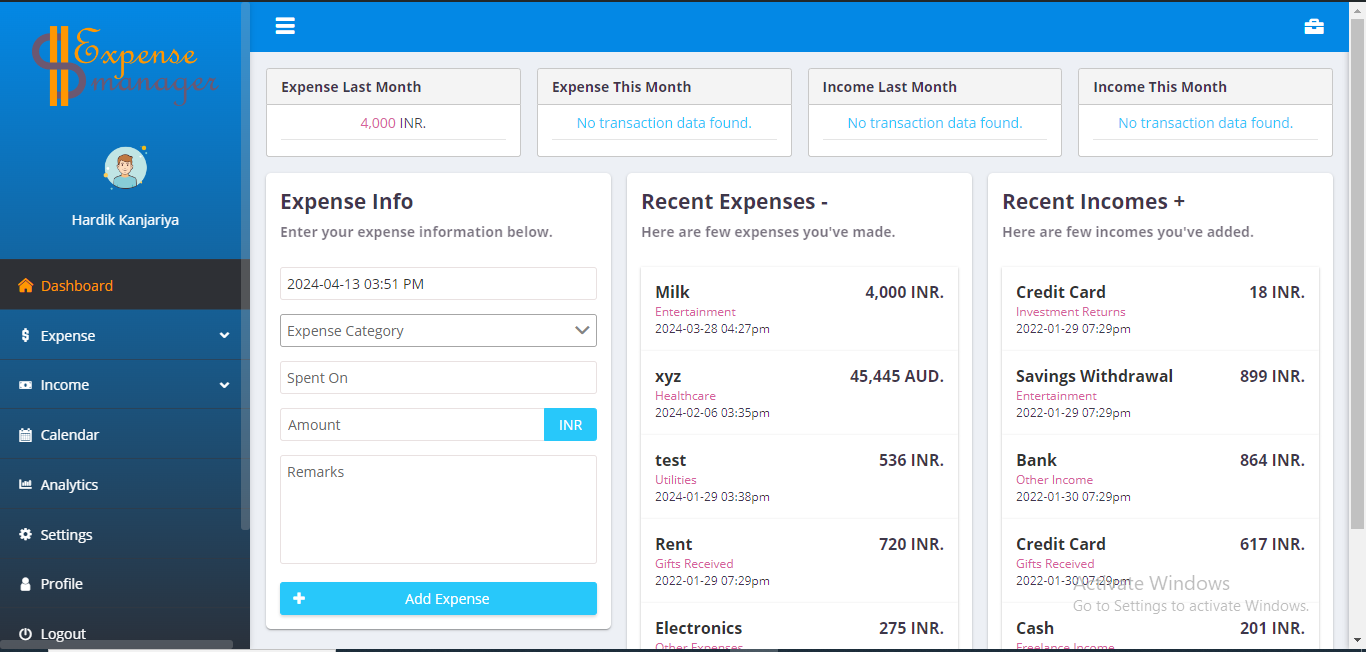
This is a login screen where users can log in with appropriate login details.

1. **Register Page**

****

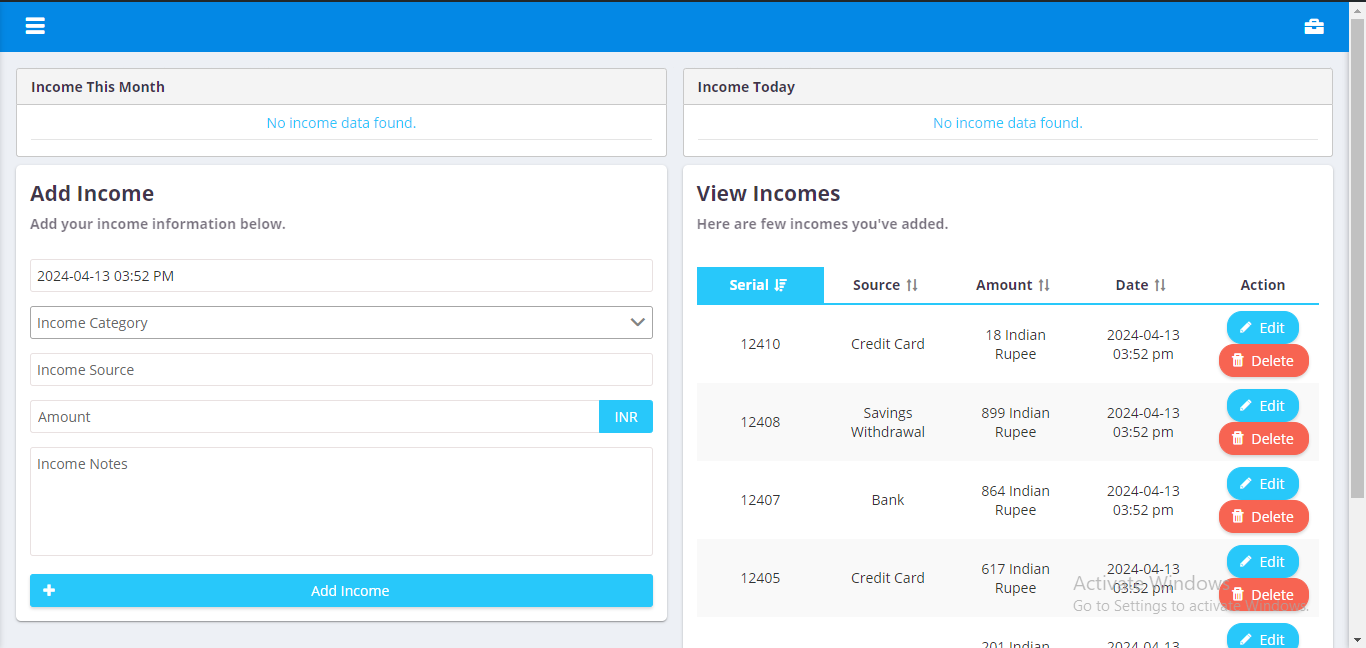
This is Register Page User Can Register.

1. **Dashboard**

****

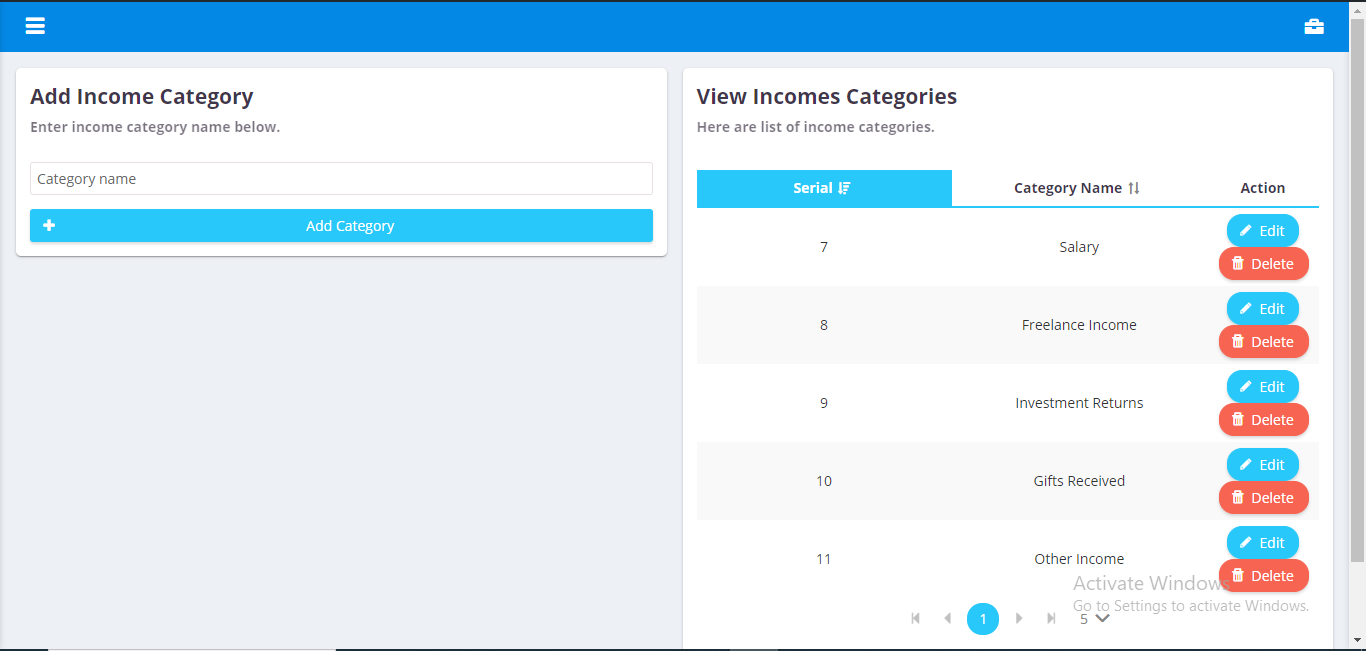
This page is UserDesgboard.

**4.** Income-details

****

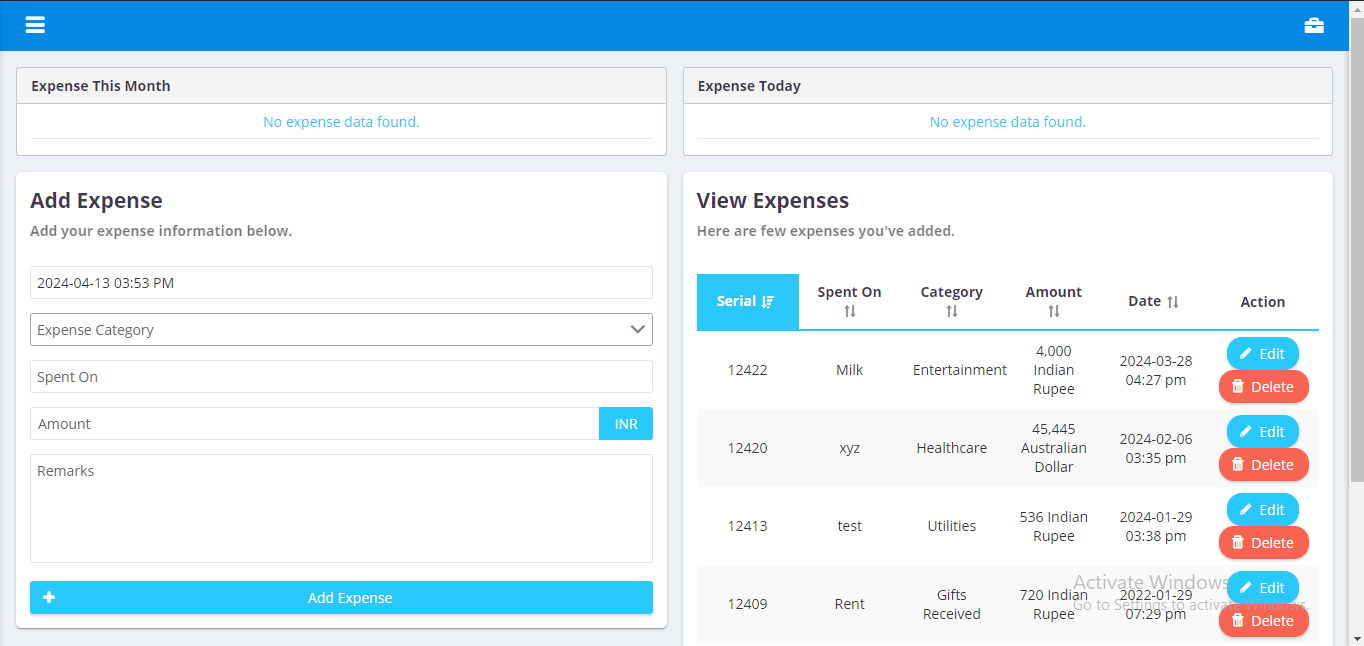
Uers Show various kinds of Income and their details

1. Income Catagory



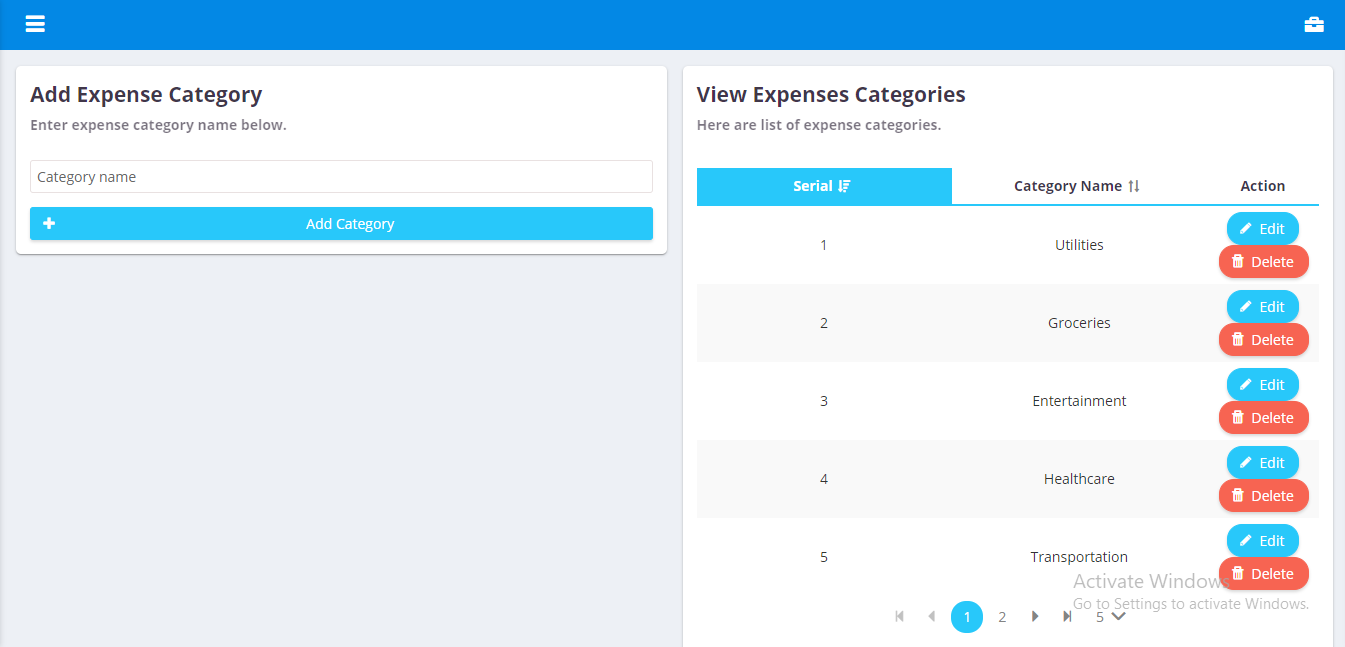
User Can add,update and delete income catagorys.

1. Expense Details



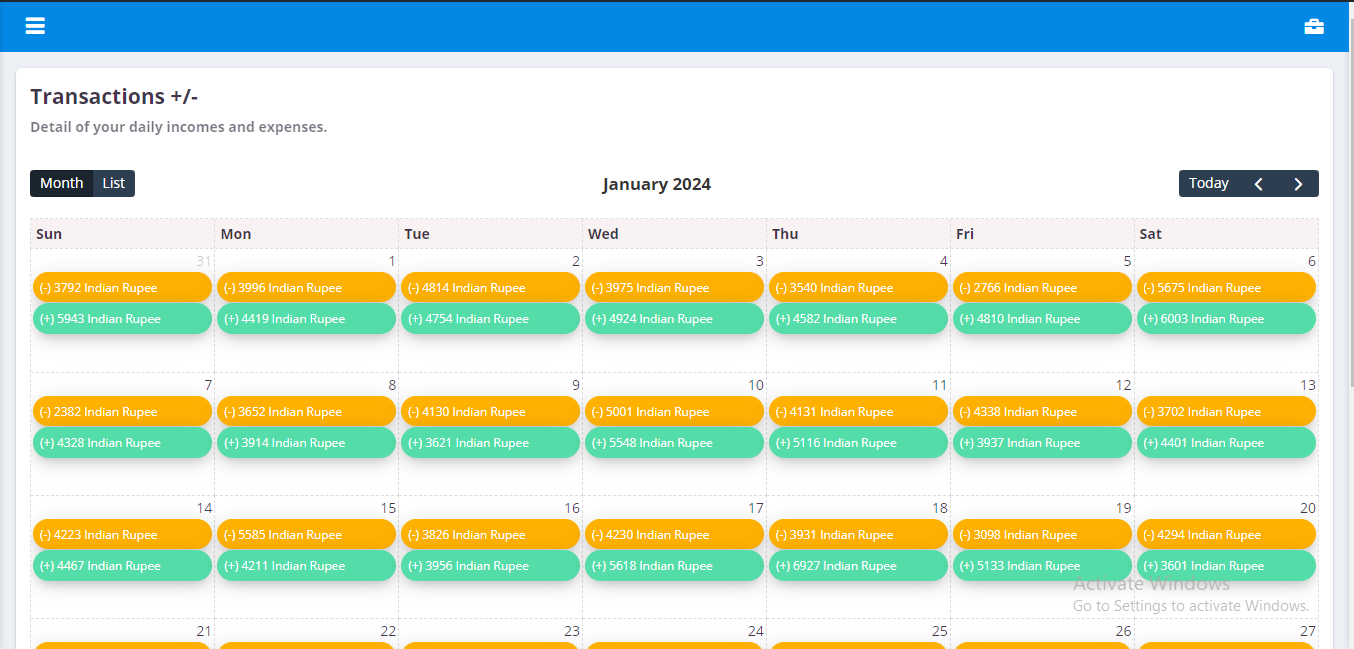
Users can show expense details and perfrom oprations like add , update and delete.

1. Expense Catagory



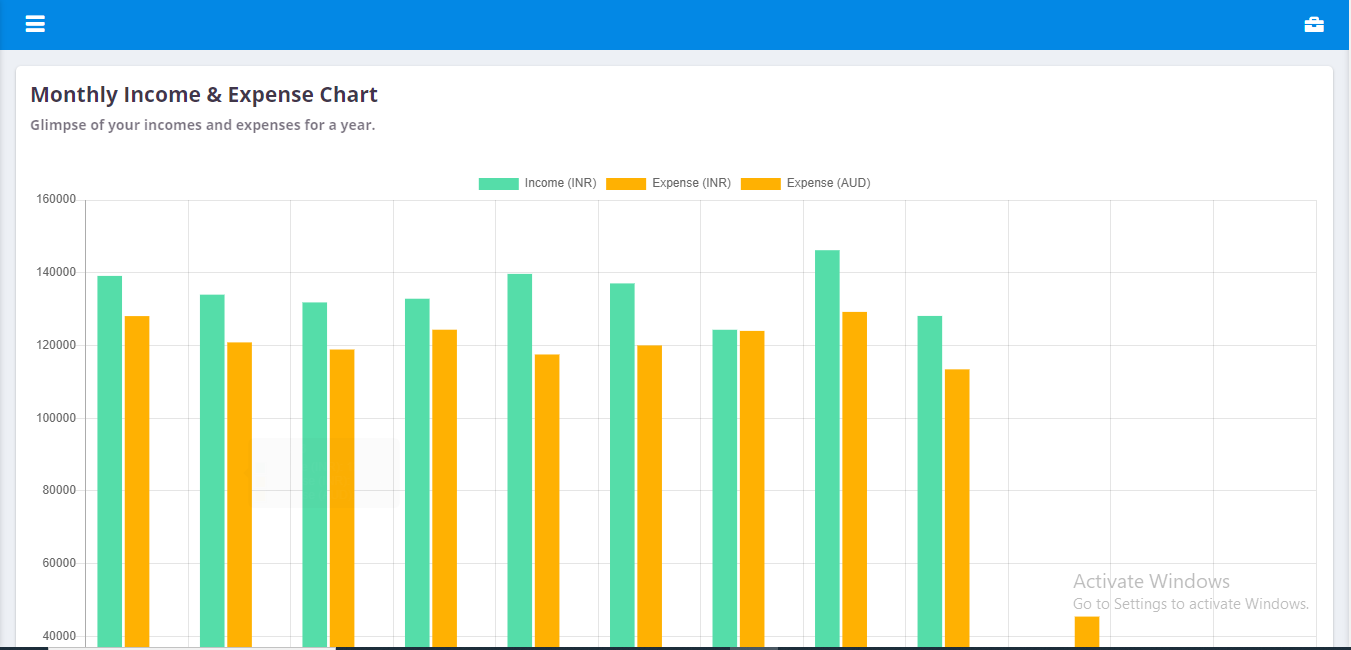
User Add,update and delete various types of catagories.

1. Calender



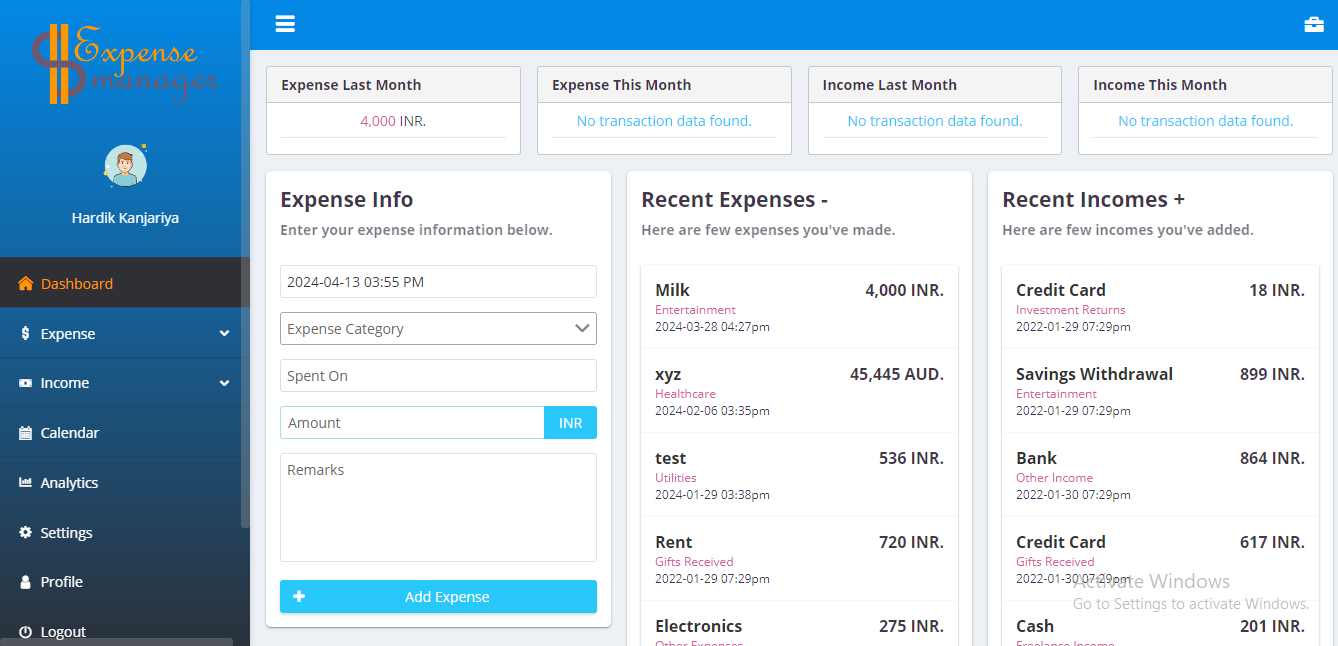
User can show date wise his income and expenses.

1. Charts



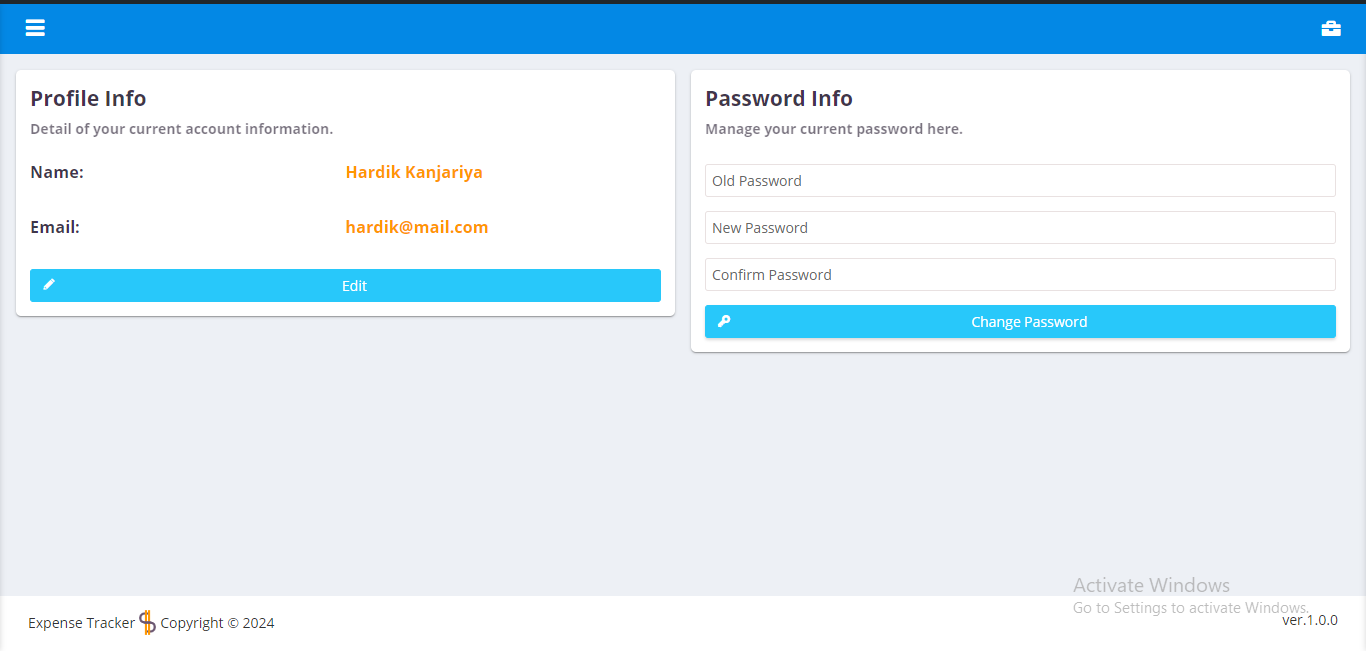
This is represent a chart in income and expense. User an view his income and expense vie chart.

10.Drawer



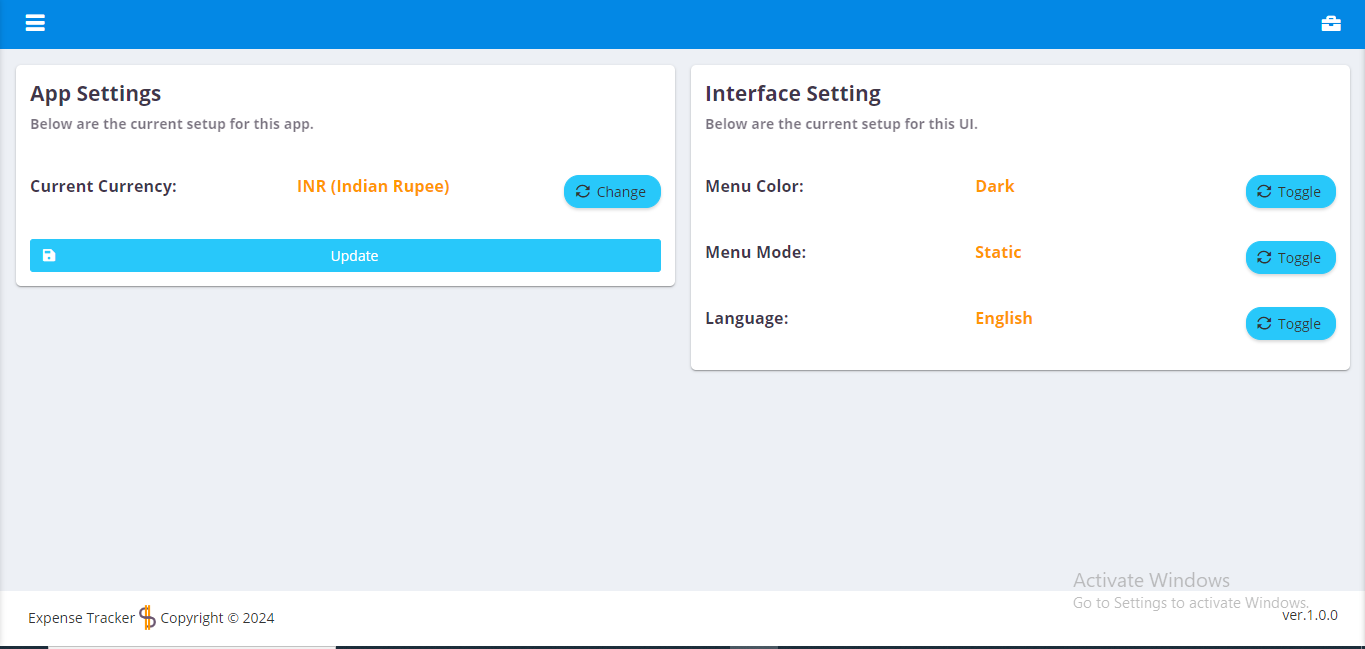
This page is display Drawer and his total items.

1. Profile



User can handle our Profile his is able to change password and also his name and email address.

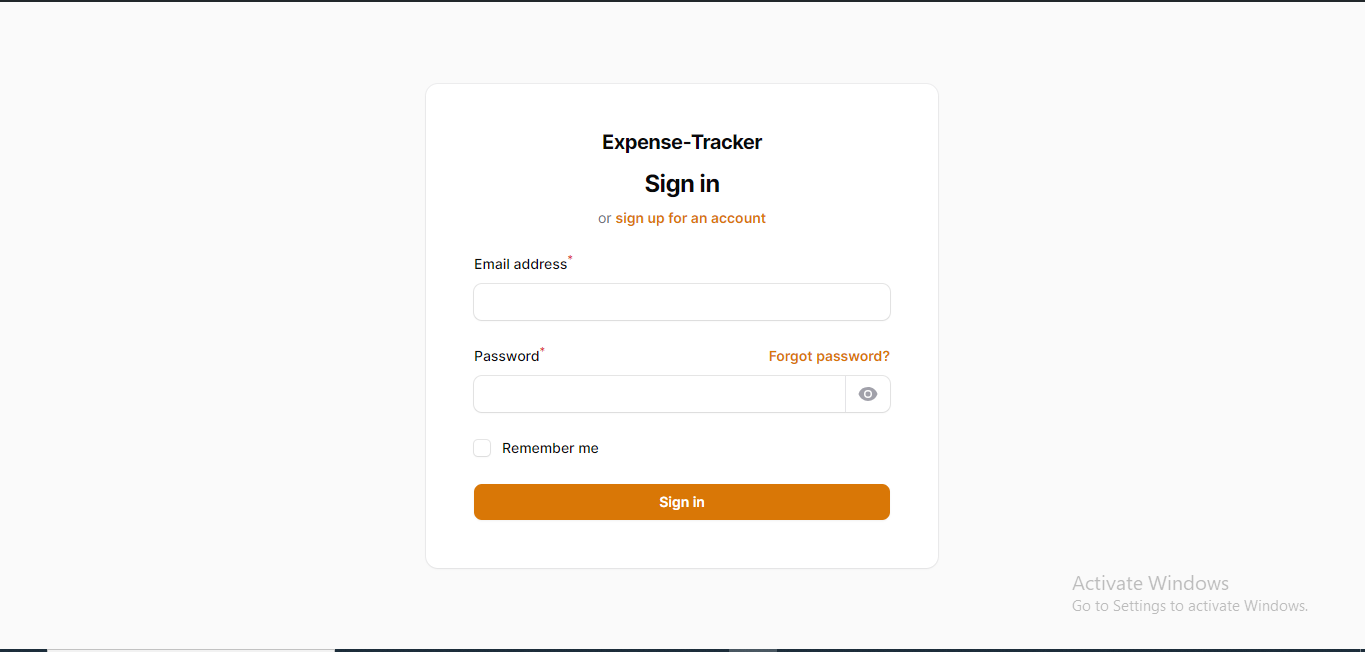
1. Setting



User can authorized to change Language Mamu mode and Manu color.

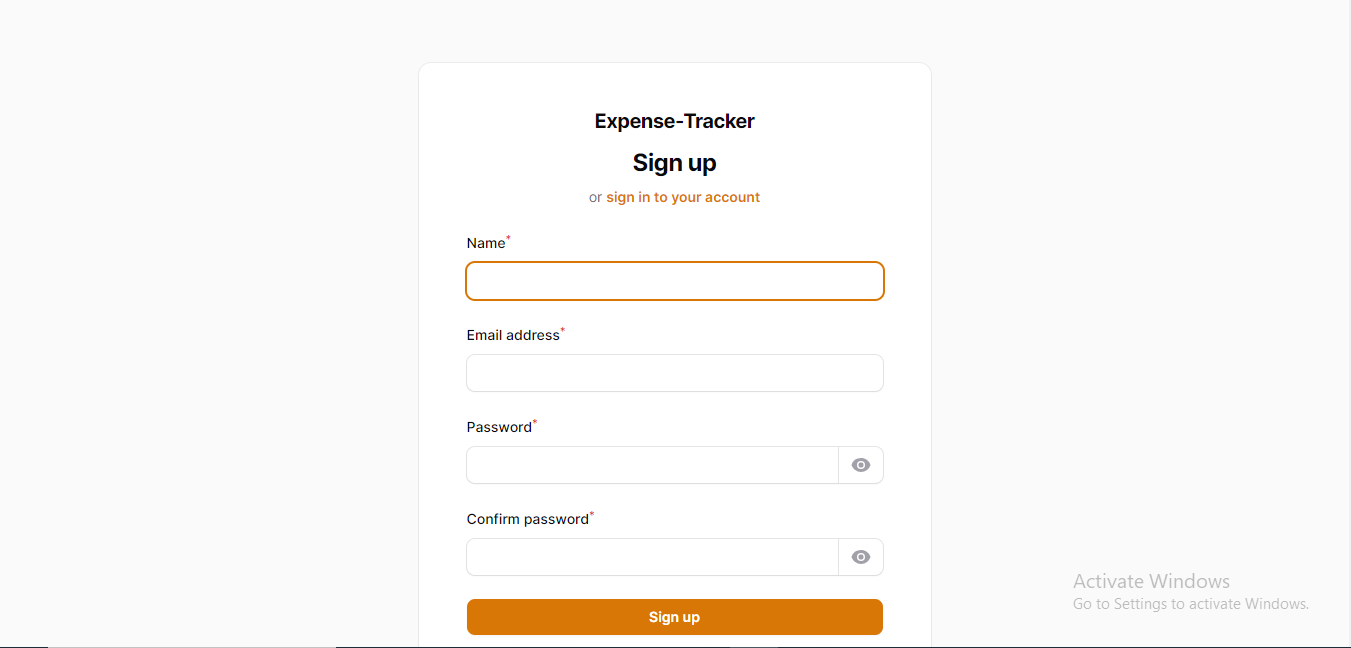
* **Vendor**

1.Login-Page



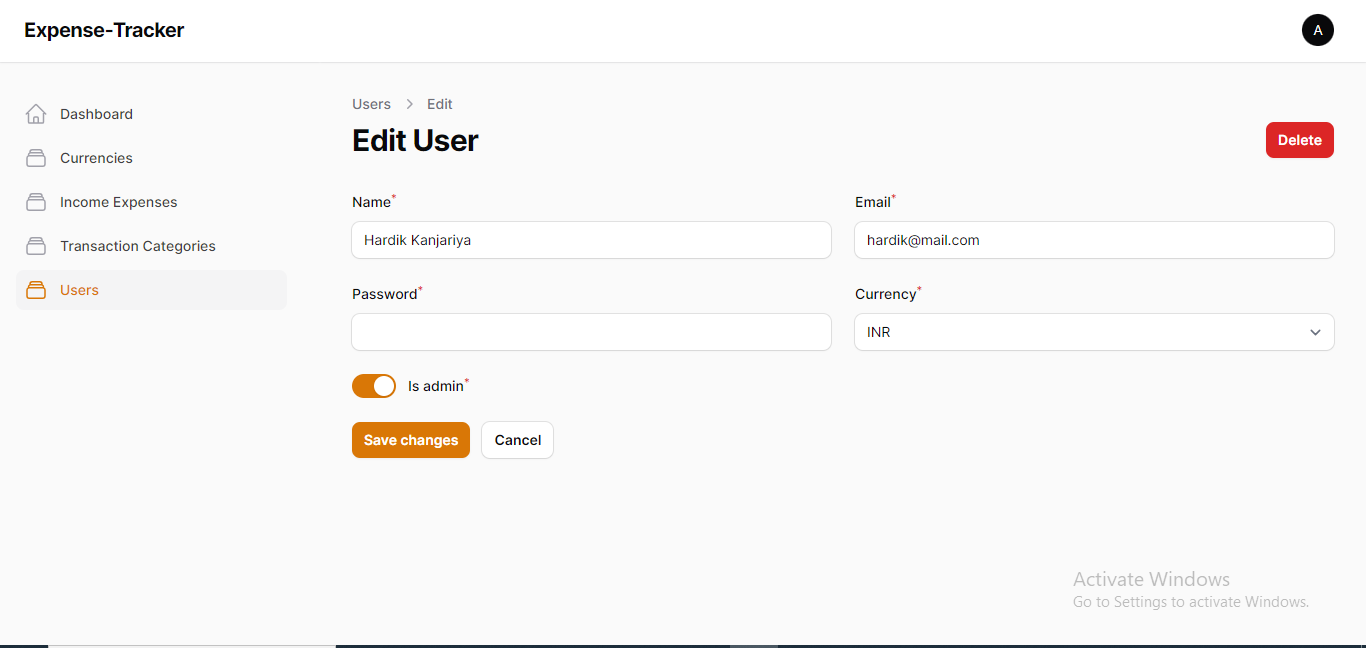
This is a login screen where users can log in with appropriate login details.

2.Register-Page

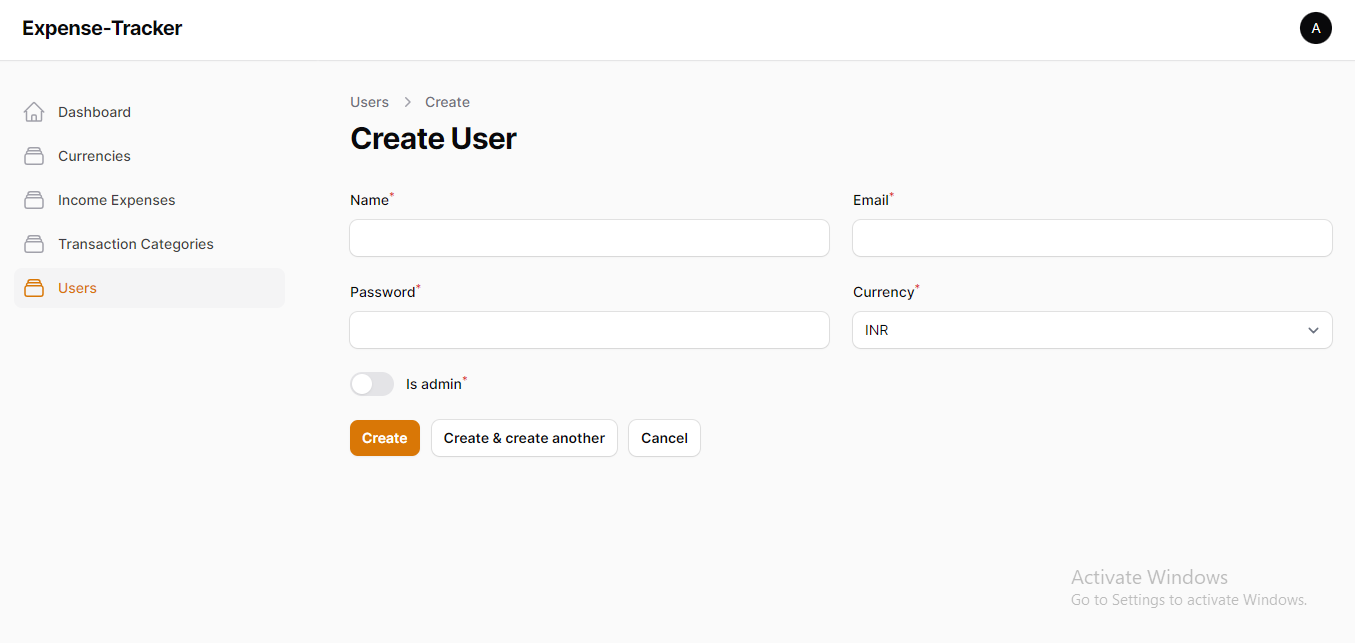


This is Register Page User Can Register.

3. Manage User Page

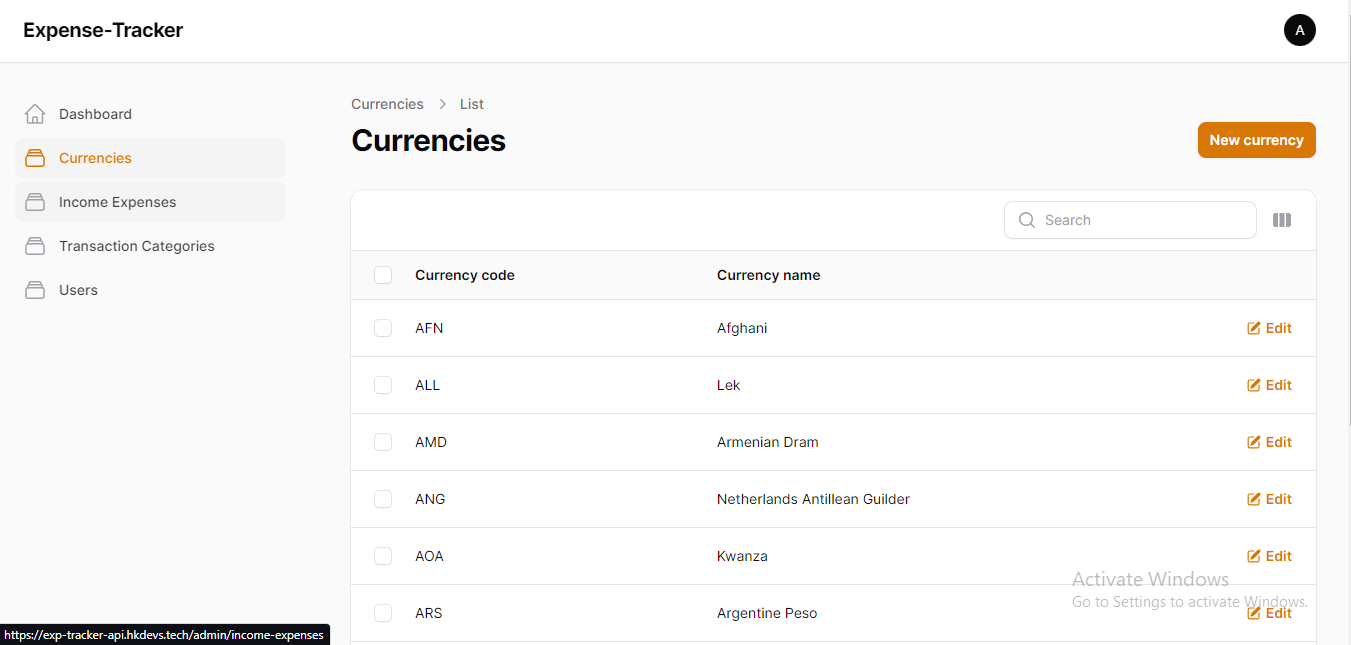
  
  
Admins can also manage users, such as updating user details and deleting users.

4. Create-User



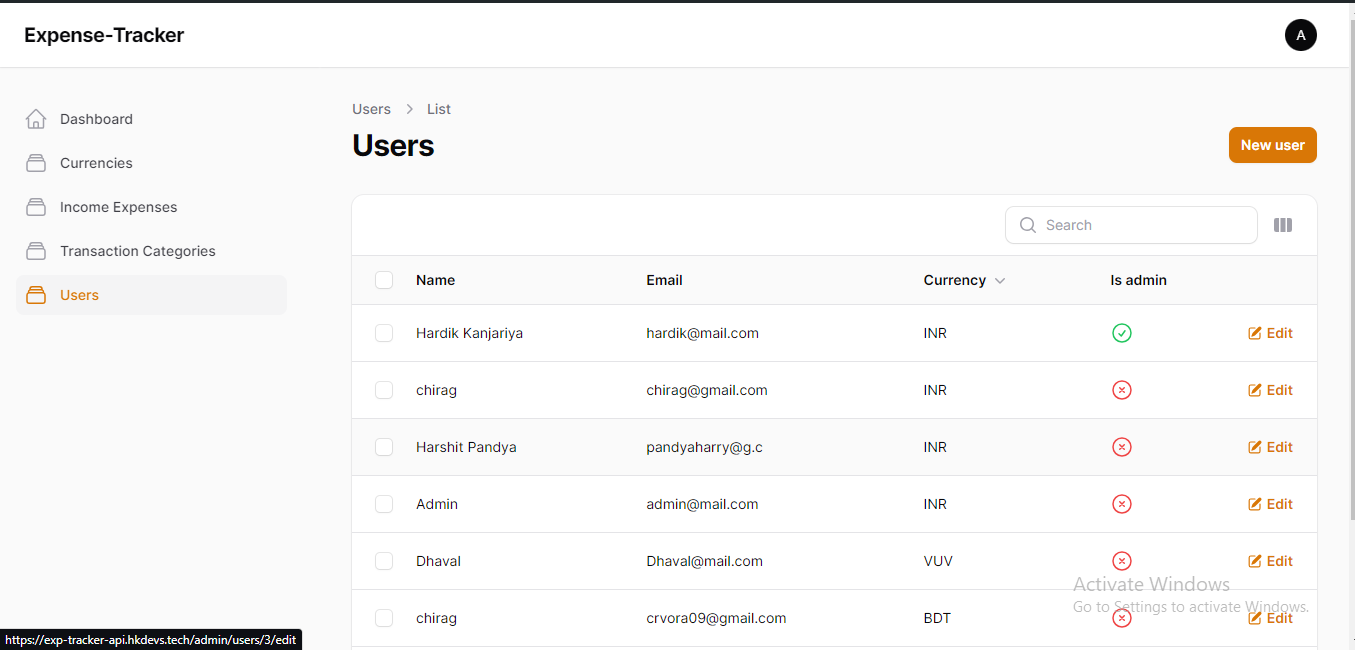
Admin can also Add-New User in User list.

5. Manage Currencies



Admin can Add,update and delete currencies.

1. View All User



Admins can view all users and manage them.

**7-Limitations and Future Enhancements**

**7.1 LIMITATIONS: -**

* User can not Provide Budget Management.
* User can not get Reminders and Alerts.
* Data Backup and Sync are not available.

**7.2 FUTURE ENHANCEMENTS: -**

* User can Provide Budget Management.
* User can get a Reminders and Alerts.
* We can provideData Backup and Sync process.

**8-Conclusion**

* the expense tracker project aims to provide users with a comprehensive tool to effectively manage their finances. Through features such as expense and income tracking, budget management, currency conversion, and insightful reports, users can gain better control over their financial activities and make informed decisions.

**9-References and Bibliography**

* ***<https://stackoverflow.com/>***
* ***<https://www.w3schools.com/>***
* ***<https://getbootstrap.com/>***
* ***<https://www.tutorialrepublic.com/>***

**BOOK**

* **ReactJS Notes for Professional, GoalKicker, Website ebook,**
* **Codevolution. “ReactJS Tutorial for Beginners.” YouTube, YouTube, www.youtube.com/playlist?list=PLC3y8-rFHvwgg3vaYJgHGnModB54rxOk3.**