Personal Expense Tracker Application
PROJECT REPORT

Submitted by

SURYA PRASATH J   722819106101
SRI RAGAVENDRA BALAJI P 722819106094
HARISH RAMANAN A   722819106303
SARAN KARTHIK C   722819106088

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SRI ESHWAR COLLEGE OF ENGINEERING

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

INTRODUCTION

1.1 Project Overview:

Personal Expense Tracker is a daily expense management system which is specially designed for non- salaried and salaried personnel for keeping track of their daily expenditure with easy and effective way through computerized system which tends to eliminate manual paper works. It will also manage records in systematic way and user can access the stored data conveniently. We have tried to design the project in such way that user may not have any difficulty in using this application without much effort. This software can be really used by end user who has stable internet . The language that we use to develop this system is flask using python and IBM Db2 for database.

1.2 Purpose:

Expense Tracker is an Application which can help the user to keep track of their Expenses. Nowadays, people can do various things by using a mobile and so, they can also use it for Budgeting and planning their expense in the mobile instead of doing it manually. For this purpose ,an application can be developed to satisfy the needs of the customer. This application can help the user to keep track of their expenses in an organized way and to maintain a proper balance between expenditure and savings.

The idea of developing this project for user convenience. Because whenever they make expenses immediately, they add in the application. Some of the concerns maintaining a personal expense is a BIG problem, in daily expenses many times we don't know where the money goes. Some of the conventional methods used to tackle this problem in normal circumstances are like making use of a sticky notes by common users, Proficient people deals with this kind of problems by using spreadsheets to record expense and using a ledger to maintains the large amounts data by especially by expert people. Such that app is capable of recording the expenditure and giving broad view with easy to use the user interface and this application is intelligent enough to shows the history of expenses.

LITERATURE SURVEY

2.1 Existing Problem:

(i) Miriam Thomas et al., proposed an Expense Tracker System which works based on the Least Square Algorithm which is a statistical procedure to find the best fit for a data point by minimizing offsets. In this, they have proposed an application which allows the user to maintain a Digital Automated Diary. The User is required to register on the system to get an user id and login password which they will use to keep track of their expenses.

(ii) Gomathy et al., proposed a system which has an Expense Tracker with few more features like Weekly Budget planner to keep track of expenses, UPI linkup to keep track of online transactions and an Automated message alert will be generated when the user crosses their budget limit, Wishlist, Rewards, Weekly and Monthly Analysis in the form of a pie chart.

(iii) Aman Garg et al., has proposed a system which has eliminated sticky notes and spreadsheets that cause confusion and data inconsistency problems. Their System has features like Categorization of expenditures and Report generation and Comparison of prices.

2.2 References:

(i) International Journal for Research in Applied Science & Engineering Technology (IJRASET) Expense Tracker
Aman Garg , Mukul Goel , Sagar Mittal , Mr. Shekhar Singh

(ii) EXPENDITURE MANAGEMENT SYSTEM Dr. C.K.Gomathy, G. Nikhitha , H. Sri Lasya Dr. V. Geetha
https://www.researchgate.net/publication/360620084

(iii) https://ijirt.org/master/publishedpaper/IJIRT150860_PAPER.pdf

2.3 Problem Statements:

(i)The User Needs to manage their Budget and keep track of their expenses in an more effective manner.

(ii)   The User may use Traditional way of manual diary, but it will be more time-consuming and confusing for the user

(iii)   The User can also use spreadsheets to do that, but it will lead to more confusion and data inconsistency.

(iv)   An Effective of economy management is needed for the users in form of either mobile or web application.

IDEATION & PROPOSED SOLUTION

3.1 Empathy map canvas:
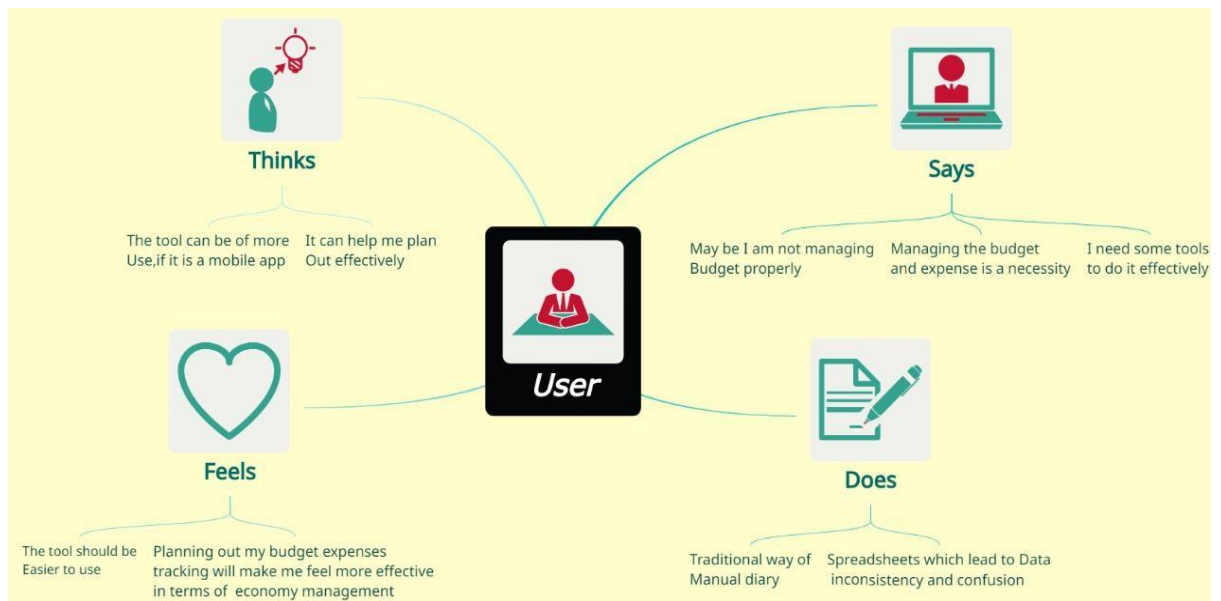
3.2 Ideation & Brainstorming:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving.

Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Template

Brainstorm
& ideaprioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

⏱ 10 minutes

Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM
How might we track personal expenses?

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.	Encourage wild ideas.
Defer judgment.	Listen to others.
Go for volume.	If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go.
In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Kavinaya N

> Day to Day reminder about account balance via mail

Afra Thahseen J

> If the user spent high ,then send mail

Abdul Wazeere Nihaal KM

> If they logged in to the web app, we can send some alert

Jayasri PS

> Sending mail when exceeds the limit

> If they logged in to the web app, we can send some alert

> Frequent monitoring on balance by user.

> Sending mail when exceeds the limit

> If the user spent high ,then send mail

> Sending mail after exceeds the limit is useless

> Day to Day remainder about account balance via mail

Step-3: Idea Prioritization

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Importance

If each of these

tasks could get done without any difficulty or cost, which would have the most positive impact?

> If the user spent high ,then send mail

> Day to Day remainder about account balance via mail

> if they logged in to the web app, we can send some alert

Feasibility

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	• To digitalize the records of the expenses of the user. • Generally people are not aware of some of the expenses and end up in debt.
2.	Idea / Solution description	• An application is designed to monitor the income and manage the expenses. • The income and the expense can be represented in graphical manner
3.	Novelty / Uniqueness	• An alert can be sent to user if the expense is reached the threshold limit. • A notification will be send to the user if the user didn't enter the data for particular period of time
4.	Social Impact / Customer Satisfaction	• The user can keep track of the expense and manage the money accordingly. • The users can refer the expense and balance amount anytime they want by their phone.
5.	Business Model (Revenue Model)	• The user can keep track of the expense and manage the money accordingly. • The users can refer the expense and balance amount anytime they want by their phone.

| 6. | Scalability of the Solution | • Our Application can handle a large number of Users |
| | | • The scalability can be easily achieved since we using cloud technology. |

3.4 Proposed Solution fit:

PROBLEM-SOLUTION FIT

Define CS, fit into CC

1. CUSTOMER SEGMENT(S)

• Working Individuals
• Students
• Budget conscious consumers

6. CUSTOMER CONSTRAINTS

• Internet Access
• Device (Smartphone) to access the application
• Data Privacy
• Cost of existing applications
• Trust

5. AVAILABLE SOLUTIONS

• Expense Diary or Excel sheet

PROS : Have to make a note daily which helps to be constantly aware

CONS : Inconvenient, takes a lot of time

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS

• To keep track of money lent or borrowed
• To keep track of daily transactions
• Alert when a threshold limit is reached

9. PROBLEM ROOT CAUSE

• Reckless spendings
• Indecisive about the finances
• Procrastination
• Difficult to maintain a note of daily spendings (Traditional methods like diary)

7. BEHAVIOUR

• Make a note of the expenses on a regular basis.
• Completely reduce spendings or spend all of the savings
• Make use of online tools to interpret monthly expense patterns

Identify strong TR & EM

3. TRIGGERS
• Excessive spending
• No money in case of emergency

4. EMOTIONS

BEFORE
• Anxious
• Confused
• Fear

AFTER
• Confident
• Composed
• Calm

10. YOUR SOLUTION

Creating an application to manage the expenses of an individual in an efficient and manageable manner, as compared to traditional methods

8. CHANNELS OF BEHAVIOUR

ONLINE
Maintain excel sheets and use visualizing tools

OFFLINE
Maintain an expense diary

REQUIREMENT ANALYSIS

4.1 Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Application Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User monthly expense tentative data	Data to be registered in the app
FR-4	User monthly income data	Data to be registered in the app
FR-5	Alert/ Notification	Alert through E-mail Alert through SMS
FR-6	User Budget Plan	Planning and Tracking of user expense vs budget limit

4.2 Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Effectiveness, efficiency and overall satisfaction of the user while interacting with our application.
NFR-2	**Security**	Authentication, authorization, encryption of the application.
NFR-3	**Reliability**	Probability of failure-free operations in a specified environment for a specified time.
NFR-4	**Performance**	How the application is functioning and how responsive the application is to the end-users.
NFR-5	**Availability**	Without near 100% availability, application reliability and the user satisfaction will affect the solution.
NFR-6	**Scalability**	Capacity of the application to handle growth, especially in handling more users.

PROJECT DESIGN

5.1 Data Flow Diagram:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.
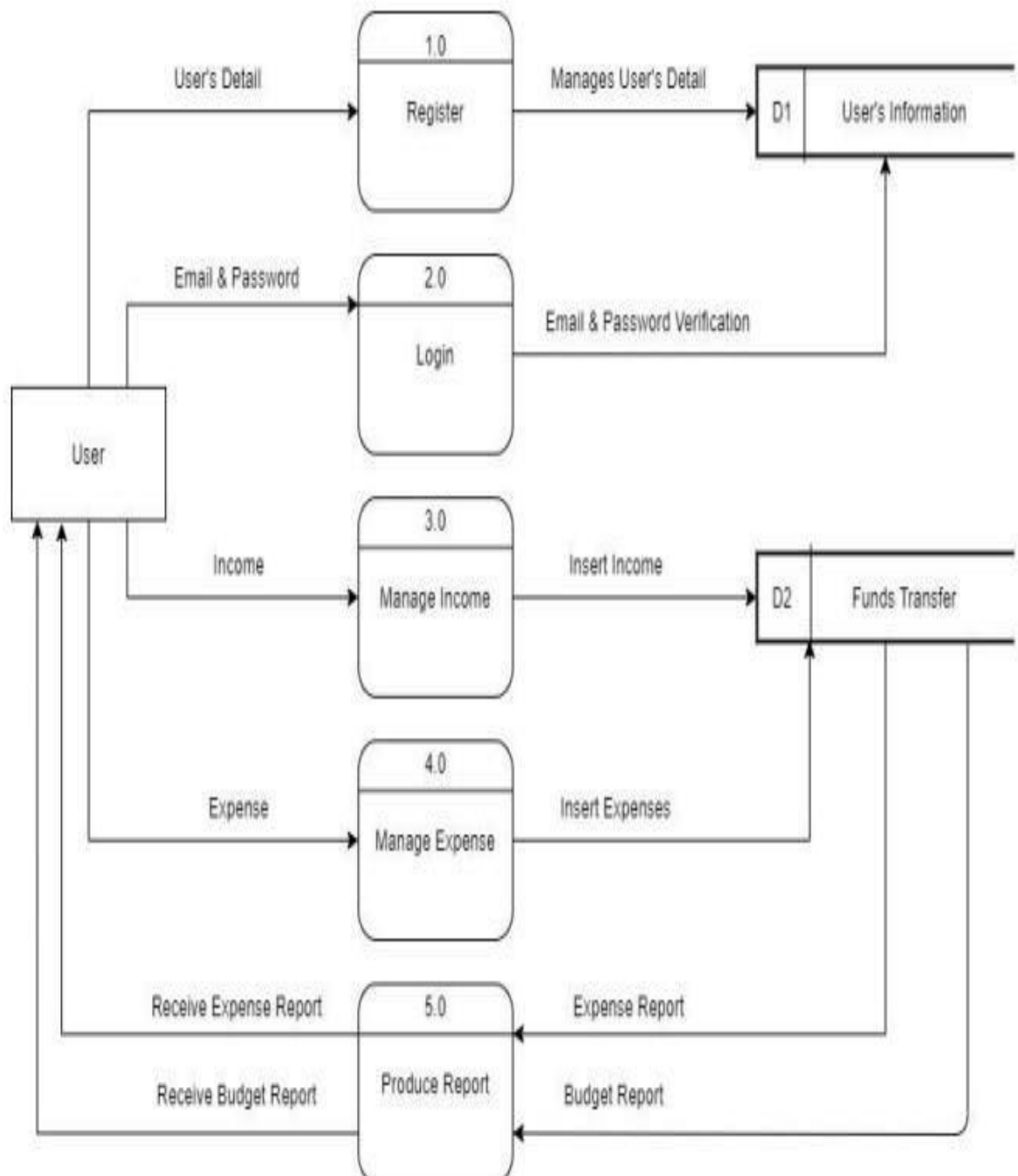
5.2 Solution & Technical Architecture:
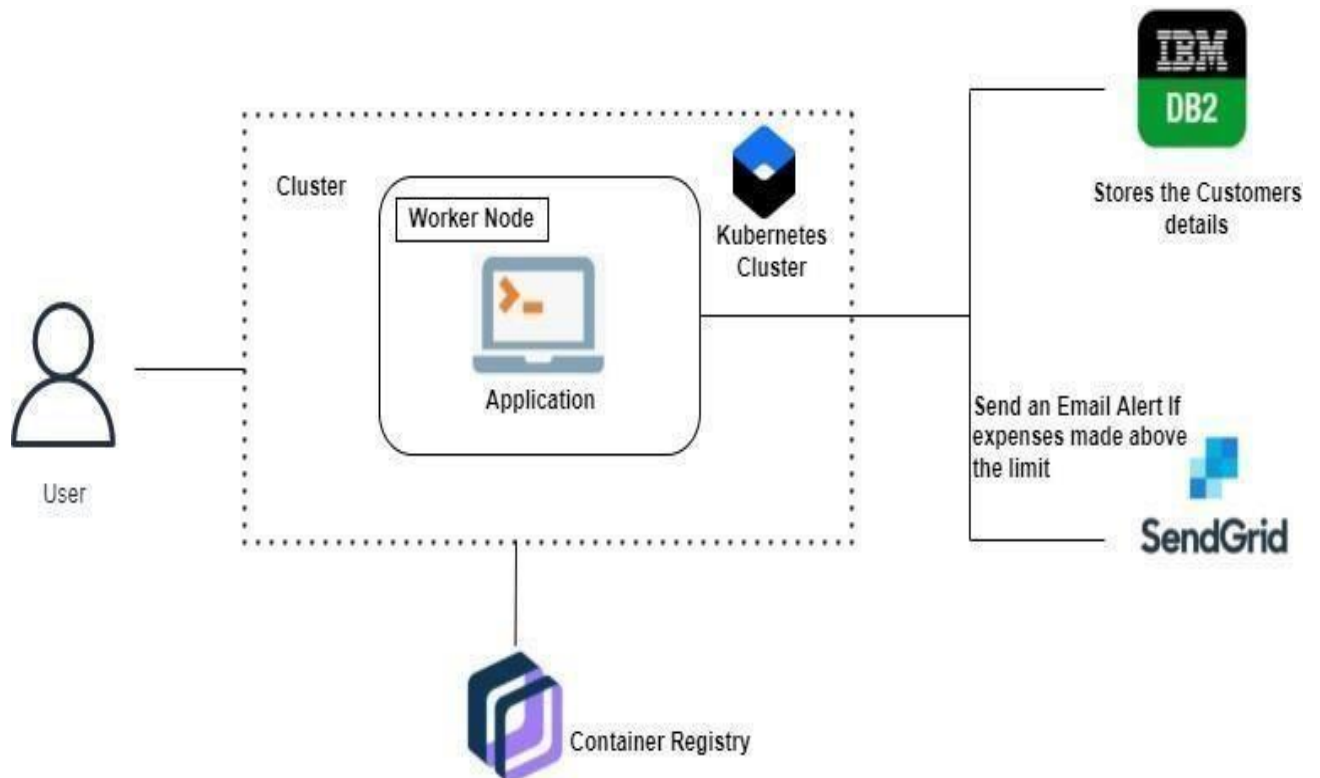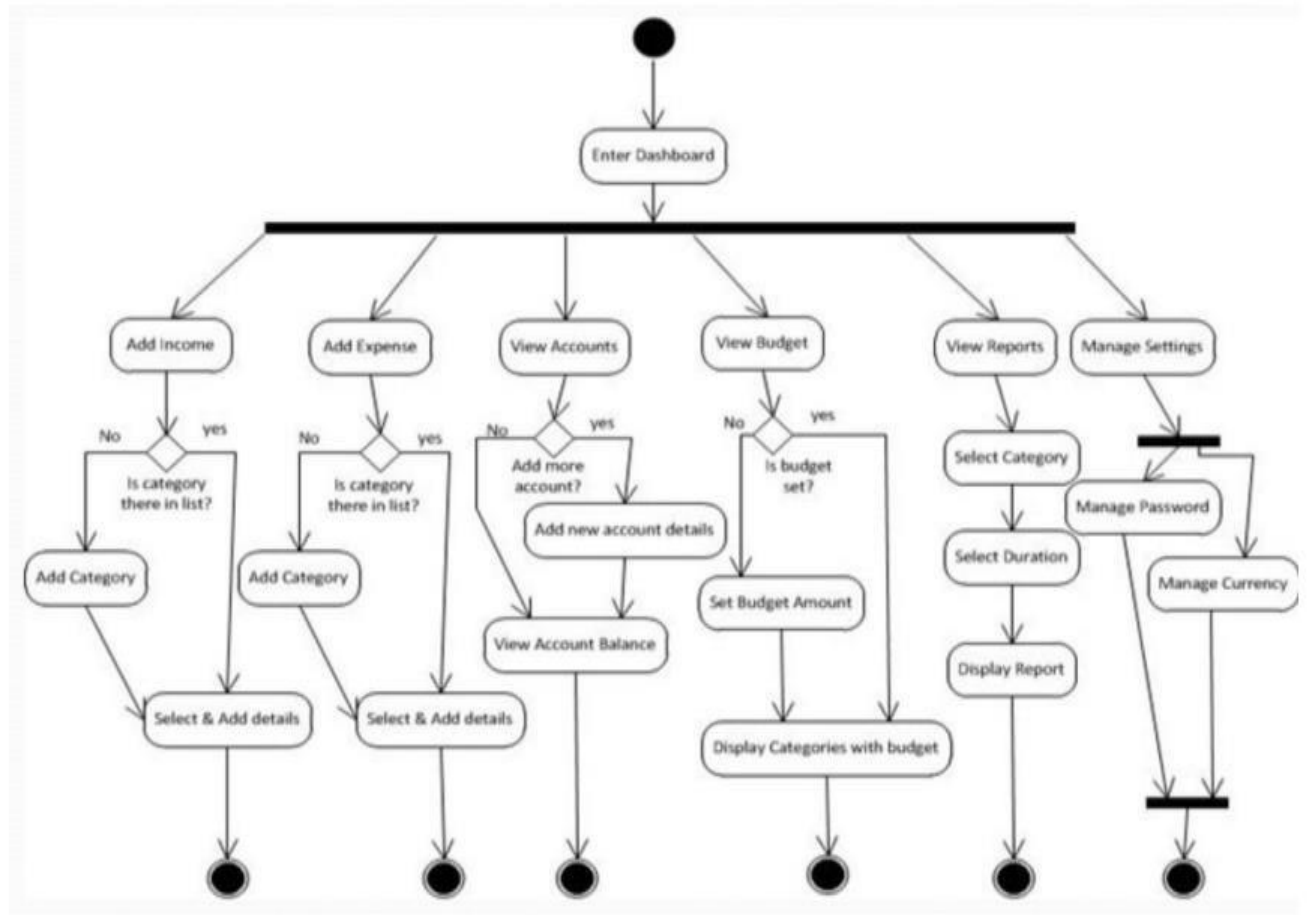
5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user & web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	
	Login	USN - 4	As a user, I can log into the application by entering email & password	I can access the application	High	
	Dashboard	USN - 5	As a user I can enter my income and expenditure details.	I can view my daily expenses	High	
Customer Care Executive		USN – 6	As a customer care executive I can solve the log in issues and other issues of the application.	I can provide support or solution at any time 24*7	Medium	
Administrator	Application	USN - 7	As a administrator I can upgrade or update the application.	I can fix the bug which arises for the customers and users of the application	Medium	

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION:

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	2	High	Sri Ragavendra Balaji P
		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Harish Ramanan A
	Login	USN-3	As a user, I can log into the application by entering email & password	1	High	Surya Prasath J

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Dashboard	USN-4	Logging in takes to the dashboard for the logged user.	2	High	Saran Karthick

	Bug fixes, routine checks and improvisation by everyone in the team *Intended bugs only					
Sprint 2	Workspace	USN-1	Workspace for personal expense tracking	2	High	Harish Ramanan A
	Charts	USN-2	Creating various graphs and statistics of customer's data	1	Medium	Surya Prasath J
	Connecting to IBM DB2	USN-3	Linking database with dashboard	2	High	Saran Karthick
		USN-4	Making dashboard interactive with JS	2	High	Sri Ragavendra Balaji P
Sprint-3		USN-1	Wrapping up the server side works of frontend	1	Medium	Surya Prasath J
	Watson Assistant	USN-2	Creating Chatbot for expense tracking and for clarifying user's query	1	Medium	Saran Karthick
	SendGrid	USN-3	Using SendGrid to send mail to the user about their expenses	1	Low	Sri Ragavendra Balaji P
		USN-4	Integrating both frontend and backend	2	High	Harish Ramanan A

			Bug fixes, routine checks and improvisation by everyone in the team *Intended bugs only			
Sprint-4	Docker	USN-1	Creating image of website using docker/	2	High	Saran Karthick
	Cloud Registry	USN-2	Uploading docker image to IBM Cloud registry	2	High	Sri Ragavendra Balaji P
	Kubernetes	USN-3	Create container using the docker image and hosting the site	2	High	Harish Ramanan A
	Exposing	USN-4	Exposing IP/Ports for the site	2	High	Surya Prasath J

6.2 Sprint Delivery:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \text{Sprint Duration} / \text{Velocity} = 20/6 = 3.33$$
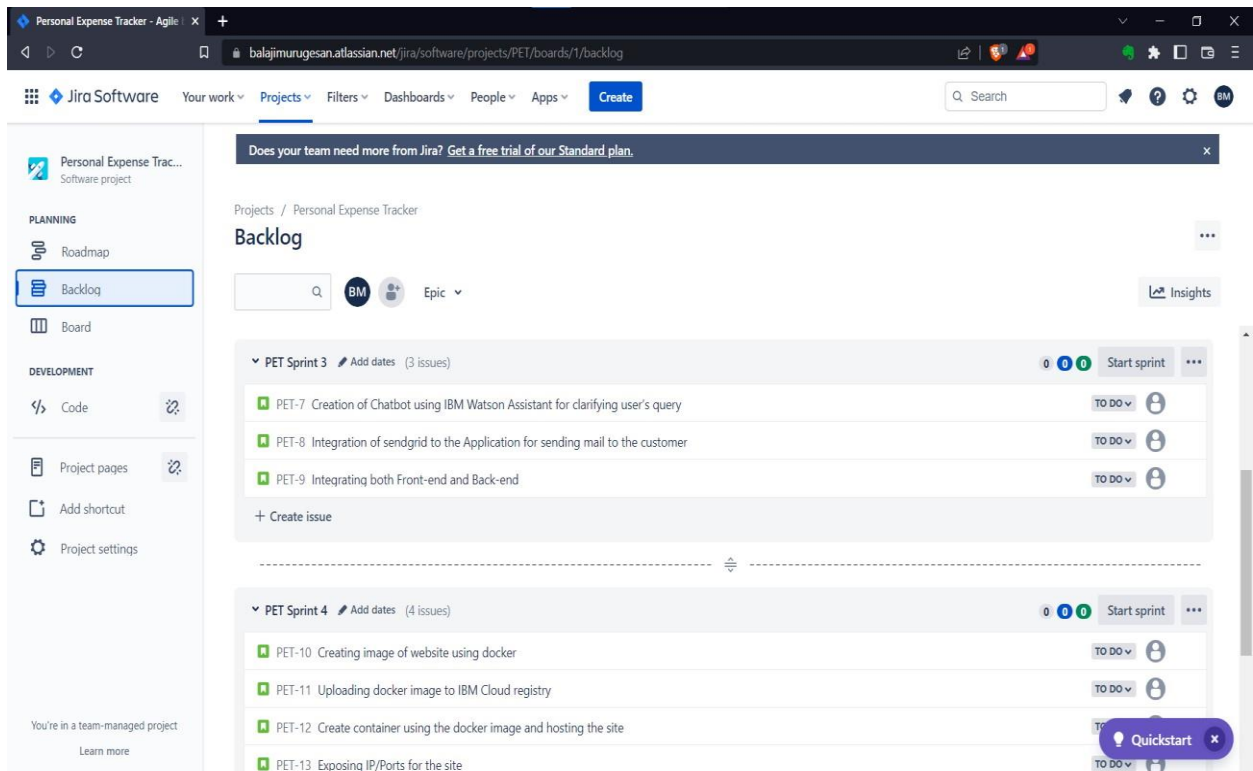
6.3 Report from JIRA:

CODING & SOLUTIONING

7.1 Feature 1:

Python

- Python is a widely-used, interpreted, object-oriented, and high-level programming language with dynamic semantics, used for generalpurpose programming. It's everywhere, and people use numerous Python-powered devices on a daily basis, whether they realize it or not.

- Python was created by Guido van Rossum, and first released on February 20, 1991.

- Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Smalltalk, and Unix shell and other scripting languages.

- Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL)

- It is easy to learn – the time needed to learn Python is shorter than for many other languages; this means that it's possible to start the actual programming fast

- It is easy to use for writing new software – it's often possible to write code faster when using Python.
- It is easy to obtain, install and deploy – Python is free, open and multiplatform; not all languages can boast that.
- Programming skills prepare you for careers in almost any industry and are required if you want to continue to more advanced and higherpaying software development and engineering roles.

- Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

7.2 Feature 2:

Flask

- **Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.

- It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself.

- Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

7.3 Database Scheme

IBM Db2

- DB2 is a database product from IBM.

- It is a Relational Database Management System (RDBMS). DB2 is designed to store, analyze and retrieve the data efficiently.

- DB2 product is extended with the support of Object-Oriented features and non-relational structures with XML.

○ Provide a massively parallel processing (MPP) architecture Exploits Hive, HBase and Apache Spark concurrently for best-in-class analytic capabilities.

○ Provides low latency support for ad-hoc and complex queries, high performance, and federation capabilities Understands dialects from other vendors and various products from Oracle, IBM® Db2® and IBM Netezza® Enables advanced row and column security.

Kubernetes

○ **Kubernetes** is also known as **'k8s'.**
○ **Kubernetes** is an extensible, portable, and open-source platform designed by **Google** in **2014**.
○ It is mainly used to automate the deployment, scaling, and operations of the container based applications across the cluster of nodes.
○ Kubernetes helps to manage containerised applications in various types of physical, virtual, and cloud environments.
○ Google Kubernetes is a highly flexible container tool to consistently deliver complex applications running on clusters of hundreds to thousands of individual servers ○ Kubernetes is the Linux kernel which is used for distributed systems.
○ It helps you to be abstract the underlying hardware of the nodes(servers) and offers a consistent interface for applications that consume the shared pool of resources.

8 TESTING:

8.1 Test case

○ The purpose of testing is to discover errors.
○ Testing is the process of trying to discover every conceivable fault or weakness in awork product.
○ It provides a way to check the functional it your components, subassemblies, assemblies and/or a finished product
○ It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectation and does not fail in an unacceptable manner.
○ There are various types of test. Each test type addresses a specific testing requirement

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup displayed or not	http://127.0.0.1:5000/signup	Login/Signup popup should display	Working as expected	Pass	Steps are not clear to follow	yes		customer
LoginPage_TC_OO2	UI	Home Page	Verify the UI elements in Login/Signup popup	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link	http://127.0.0.1:5000/signin	Application show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Working as expected	Pass	Steps are not clear to follow	yes		customer
LoginPage_TC_OO3	Functional	Home page	Verify user is able to log into application with Valid credentials	1.Enter URL(https://shopenzer.com/) and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: thamizhchelvan@gmail.com password: Thamizhchelvan123	User should navigate to user account homepage	Working as expected	Pass	Steps are not clear to follow	yes		customer
LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials	1.Enter URL(https://shopenzer.com/) and click go 2.Click on My Account dropdown button 3.Enter InValid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: thamizhchelvan@gmail l password: Thamizhchelvan123	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass	Steps are not clear to follow	yes		customer

8.2 User Acceptance Testing:

1.Purpose of Document: The purpose of this document is to briefly explain the test coverage and open issues of the Personal Expense Tracker Application project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis: This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Sub total
By Design	8	4	2	3	17
Duplicate	1	0	2	1	4
External	2	3	0	1	6
Fixed	10	2	5	18	35

Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	3	2	1	6
Totals	21	12	13	25	71

3.Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	50	0	0	50
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	10	0	0	10
Final Report Output	6	0	0	6
Version Control	3	0	0	3

RESULT
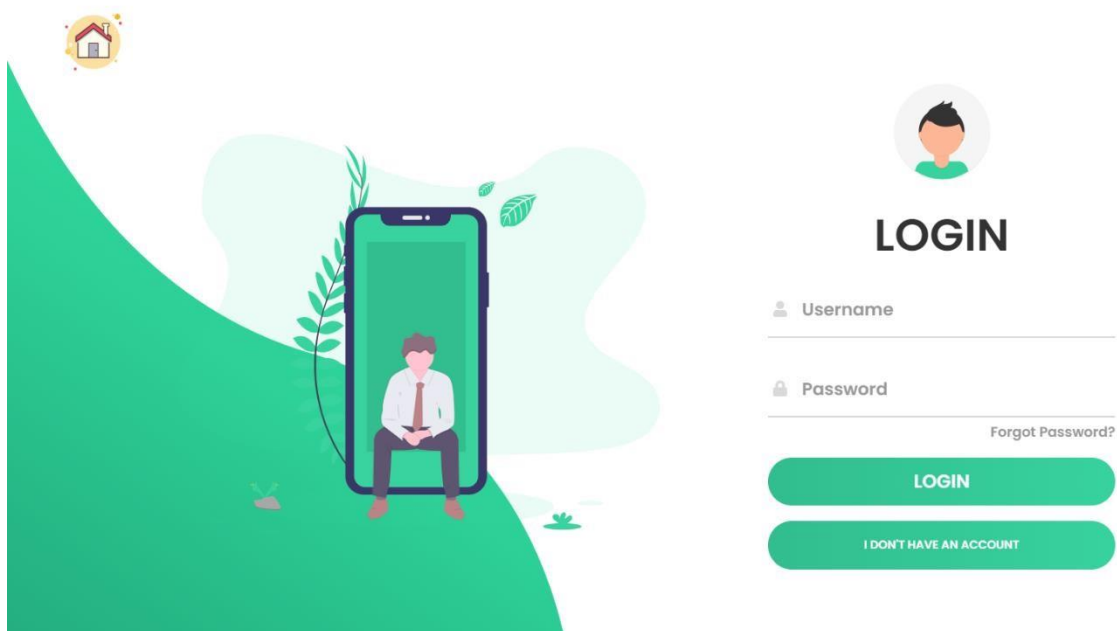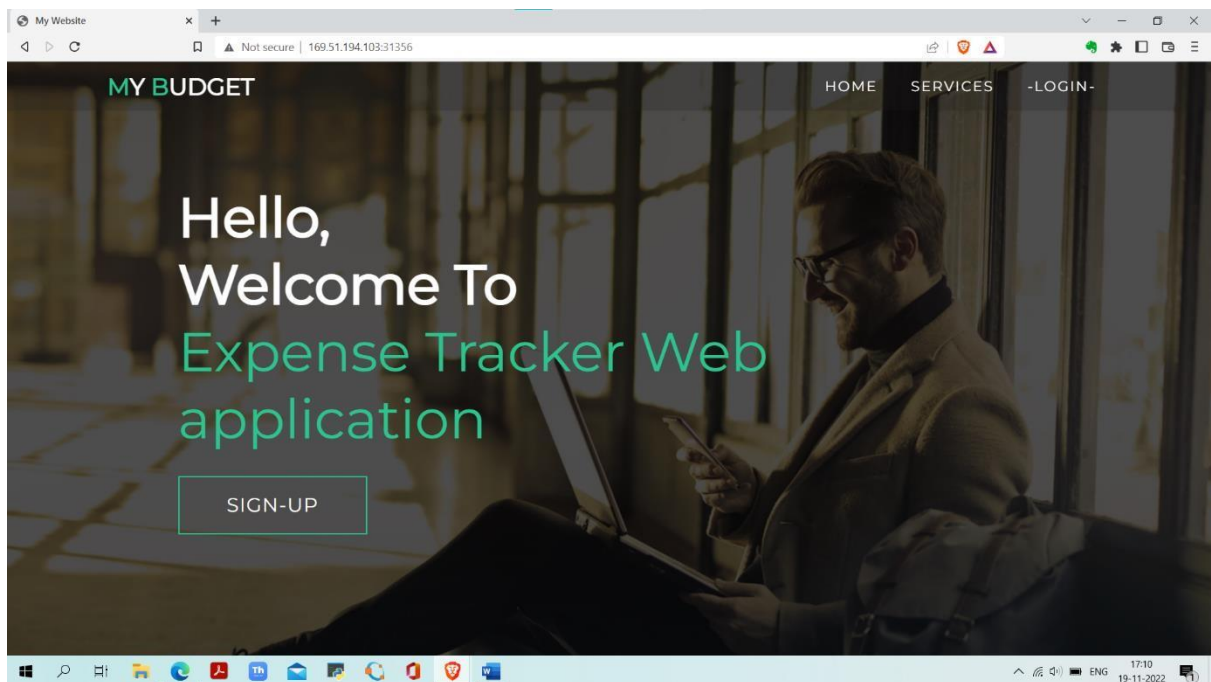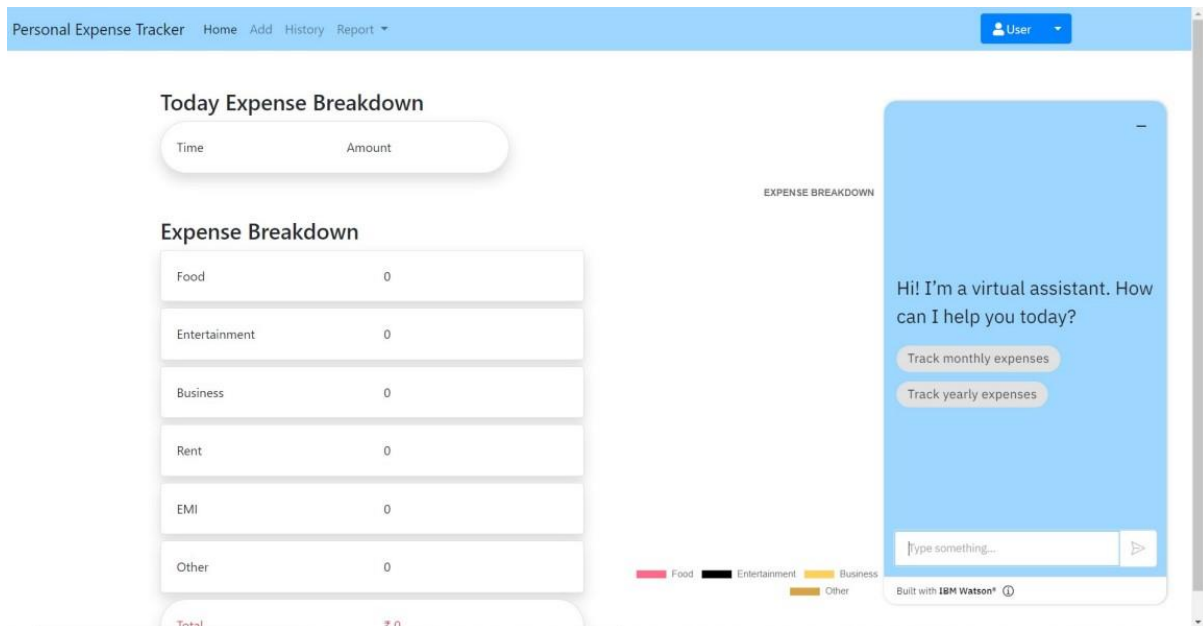
10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:

• Track your expenses anywhere, anytime.

• Seamlessly manage your money and budget without any financial paperwork.

• Access, submit, and approve invoices irrespective of time and location.

• Avoid data loss by scanning your tickets and bills and saving in the app.

DISADVANTAGES:

• Difficulty finding a budgeting method that works for you.

• Budgeting takes time and effort.

• You may think it's too rigid.

11. CONCLUSION :

The project personal expensive tracker has been successfully implemented by using python, flask, html/css/java script and the database created by using ibm db2 and also successfully executed and implemented.

12. FUTURE SCOPE:

1)   It will have various options to keep record (for example Food, Travelling Fuel, Salary).

2)   Automatically it will keep on sending notifications for our daily expenditure.

3)   In today's busy and expensive life, we are in a great rush to make moneys, but at the end of the month we broke off.

 As we are unknowingly spending money on title and unwanted things. So, we have come over with the plan to follow our profit.

13. APPENDIX:

SOURCE CODE:

```html
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initialscale=1.0">

  <title>Sign-up</title>

  <link href="..\static\css\signup.css" rel="stylesheet">

  <script src="https://kit.fontawesome.com/a81368914c.js"></script>
```

```html
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.cs s"
integrity="sha384-
    Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFA
W/dAiS6JXm" crossorigin="anonymous">
    </head>
    <body>
    <!-- container -->
    <div class="container" >
      <!--sign-up-box-container--->
      <div class="sign-up">
        <div id="png">
          <a href="/" title="HOME"><img style="width:55px; height:55px ; "
src="..\static\ images\home-page.png"></a>
        </div>
        <div class="frm">
          <!--heading-->
          <form action="/register" method="post">
            <div class="msg">{{ msg }}</div>
            <h1 class="heading">Hello,Friend</h1>

            <!--name-box-->
            <div class="text">
```

```html
            <img height="20px" src="..\static\images\user.png" />

            <input placeholder="Name" type="text" name="username"/>

        </div>

        <!--Email-box-->

        <div class="text">

            <img height="12px" src="..\static\images/email.png" />

            <input placeholder="Example@email.com" type="email"
name="email"/>

        </div>



        <!--Password-box-->

        <div class="text">

            <img height="20px" src="..\static\images\password.png" />

            <input placeholder="Password" type="password"
name="password"/>

        </div>



        <!--trems-->

        <div class="terms">

            <input class="check" type="checkbox" required/>

            <p class="conditions">I read and agree to <a href="#">Terms
&amp; Conditions</ a></p>

        </div>
```

```
<!--button-->

<div class="toop">

    <button type="submit" class="btn btn-primary" >CREATE

ACCOUNT</button><br><br>


    <!--sign-in-->

    <p class="conditions" id="p3">Already have an account <a

href="/signin">Sign in</a></p>

    </div>

    </form>

    </div>

    </div>

    </body>

</html>
```

GITHUB & PROJECT DEMO LINK:

Github: https://github.com/IBM-EPBL/IBM-Project-4507-1658733659.git

Project demo video link:

https://drive.google.com/file/d/1SpFtASUL0EPZmm6FOr5HgGUIklwh5svC/view?usp=sharing