RMK ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601 206, Gummidipoondi Taluk,
Thiruvallur District.

NALAIYA THIRAN PROJECT

PERSONAL EXPENSE TRACKER

DONE BY

TEAM ID: PNT2022TMID15893

VAYUGANDLA NEELIMA (111719104172)

VANKADARU LEELA SATHVIKA (111719104169)

SHAIK SAFEENA (111719104140)

S PRAGNASYA (111719104151)

CONTENTS

1. INTRODUCTION

1.1 Project Overview

The web application “Personal Expense Tracker” is developed to manage the daily expenses in a more efficient and manageable way. By using this application, we can reduce the manual calculations of the daily expenses and keep track of the expenditure. In this application, user can provide his income to calculate his total expenses per day and these results will be stored for each user. Sometimes we can’t remember where our money goes. And we can’t handle our cash flow.

For this problem, we need a solution that everyone can manage their expenses. So, we decided to find an easier way to get rid of this problem. So, our application attempts to free the user with as much as possible the burden of manual calculation and to keep the track of the expenditure. Instead of keeping a diary or a log of the expenses, this application enables the user to not just keep the control on the expenses but also to generate reports and analysis

With the help of this application, the user can manage their expenses on a daily, weekly, monthly, and yearly basis. Users can insert and delete transactions as well as can generate reports and analysis.

The graphical representation of the application along with the reminders and loan tracking is the main part of the system as it makes a full-fledged application of managing all the tasks.

1.2 Purpose

The main purpose of this application is to provide the customers with all the abilities to manage their personal expenses. For an average individual, it is very difficult to track expenses as it takes a long time because there are way too many expenses to track. It is not feasible to manage personal transactions in various accounts as it is not trackable because there are various payment options to consider. This makes it hard to save money because of high unmanageable expenditures. It is also difficult to calculate the taxes because needs statistical thinking and management of various assets such as vehicles, properties, salary etc. Users try to analyze expenses, but they can’t do it efficiently because it may involve a lot of data and graphical representations which makes it complicated. Users feel frustrated, confused, stressed etc. which makes them give up.

2. LITERATURE SURVEY

Project Title	Publisher	Advantages	Disadvantages
Expenditure Management System	Journal Of Engineering, Computing & Architecture	It provides reminders to assist user to track the project	Lacks in providing the savings analysis and records
Expense Tracker: A Smart Approach to Track	EasyChair Preprint	It provides the user to choose an expense category and add additional info such as a photo, a location, expense made.	The application is not user friendly and lacks in UI.
Expense Tracker	IJARSCT	The application they proposed allows users to keep up a digital automated diary.	Security provided by the application is low.
Application for Predictive Recommendation and Visualization of Personal Expenses	Darsh Shah, Sanay Shah, Ritik Savani	They offer shortcut and indicators to the consumer to go into the ordinary expenses from time to time	The tracking of loans and reminders is not provided
Daily Expense Tracker	Muskaan Sharma1, Ayush Bansal2	They used ReactJS to make it feasible for builders to broaden User Interfaces (UI) without difficulty via way of means of dividing it into diverse additives and additionally to develop fast systems	The analysis reports the expenses and savings is not generated.
A Review on Budget Estimator Android Application	Namita Jagtap, Priyanka Joshi, Aditya Kamble	This utility is meant to run on android tool specifically clever phone.	No specific usage of Chabot in the cloud or the usage of any helping service that would serve the expense customers

2.1 Existing problem

The main purpose of this application is to provide the customers with all the abilities to manage their personal expenses. In the existing problem they proposed an expense tracker that will maintain all the expenses record of users and manage them efficiently. The user can choose an expense category and provide additional information such as a photo, a location, and the amount of the expense, among other things. This will save the information to the local database. The user can examine and sort expenses on a weekly, monthly, or annual basis. By utilizing this, they reduced the quantity of manual calculations for their expenses and maintain track of their spending. The user can enter his income to compute his total daily expenses, and the data will be saved for each individual user. This tracker could be useful for people who frequently go on trips or to the theatre with their buddies. This tracker will make it easier for them to disburse the bill. This will show the graph in the chosen view.

To reduce manual calculations, they proposed an application. The application they proposed allows users to keep up a digital automated diary. Each user is going to be required to register on the system at registration time, the user is going to be provided id, which is able to be wont to maintain the record of every unique user. Expense Tracker application which can keep a track of IncomeExpense of a user on each day-to-day basis. the most effective organizations have the way of tracking and handling these reimbursements. This ideal practice guarantees that the expenses tracked are accurately and in an exceedingly timely manner. From a corporation perspective, timely settlements of those expenses when tracked well will boost employees' morale. Additional feature of Expense and income prediction helps to higher budget management

Every day in our lives we have a tendency to invest in distinct styles of things. A massive quantity may be remembered and additionally helps in the very last tallying of the costs and earnings. But the smaller costs pass unattended, and people small quantities eventually lead to a massive amount of cash being misplaced or spent. As a result, an overall comprehension of man or woman budget is getting progressively extensive because the regular human being’s disposable earnings has dwindled due to a converting monetary atmosphere. For the nice OCR results, a couple of schooling and checking out photographs could be required with guide efforts to accurate the textual content fetched from the receipts without a fix/established format. In assessment to the numerous gift apps to be had at the Play Store and AppStore for iOS, we described a few base parameters primarily based totally on their certain evaluation with the add-on of the prediction fashions to set it unique with inside the market.

2.2 References

[1]. Shahed Anzarus Sabab; Sadman Saumik Islam; Md. Jewel Rana; Monir Hossain (2017). eExpense: A Smart Approach to Track Everyday Expense

[2]. Hrithik Gupta, Anant Prakash Singh, Navneet Kumar and J. Angelin Blessy. (2012). Expense Tracker: A Smart Approach to Track Everyday Expense

[3]. Abishek Hagawane, Roshan Gopalghare, Prathmesh Isawe, Mrunal Aware. (2020). Daily Expense Tracker

[4]. Dr. B. Muthu Senthil1, Mummaneni Sravani2 , Konduru Theja Sree3 , M. P. Krithik Shri4 (2018), SPESE-Everyday Expenditure Tracker

[5]. "Daily Expense 3 - Apps on Google Play", [online] Available: https://play.google.com/store/apps/details?id = mic.app.gastosdiarios.

[6]. B. B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi)", Document Analysis and Recognition 1997. Proceedings of the Fourth International Conference on, vol. 2, pp. 1011-1015, 1997.

[7]. "AndroMoney (Expense Track) - Apps on Google Play", [online] Available: https://play.google.com/store/apps/details?id=com.kpmoney.android.

[8]. B. B. Chaudhuri and U. Pal, "An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi)", Document Analysis and Recognition 1997.

[9]. Thanapal, P., Patel, M., Lokesh Raj, T., & Satheesh Kumar, J. (2015). Income and Expense Tracker. Indian Journal Of Science And Technology, 8(S2), 118-122.

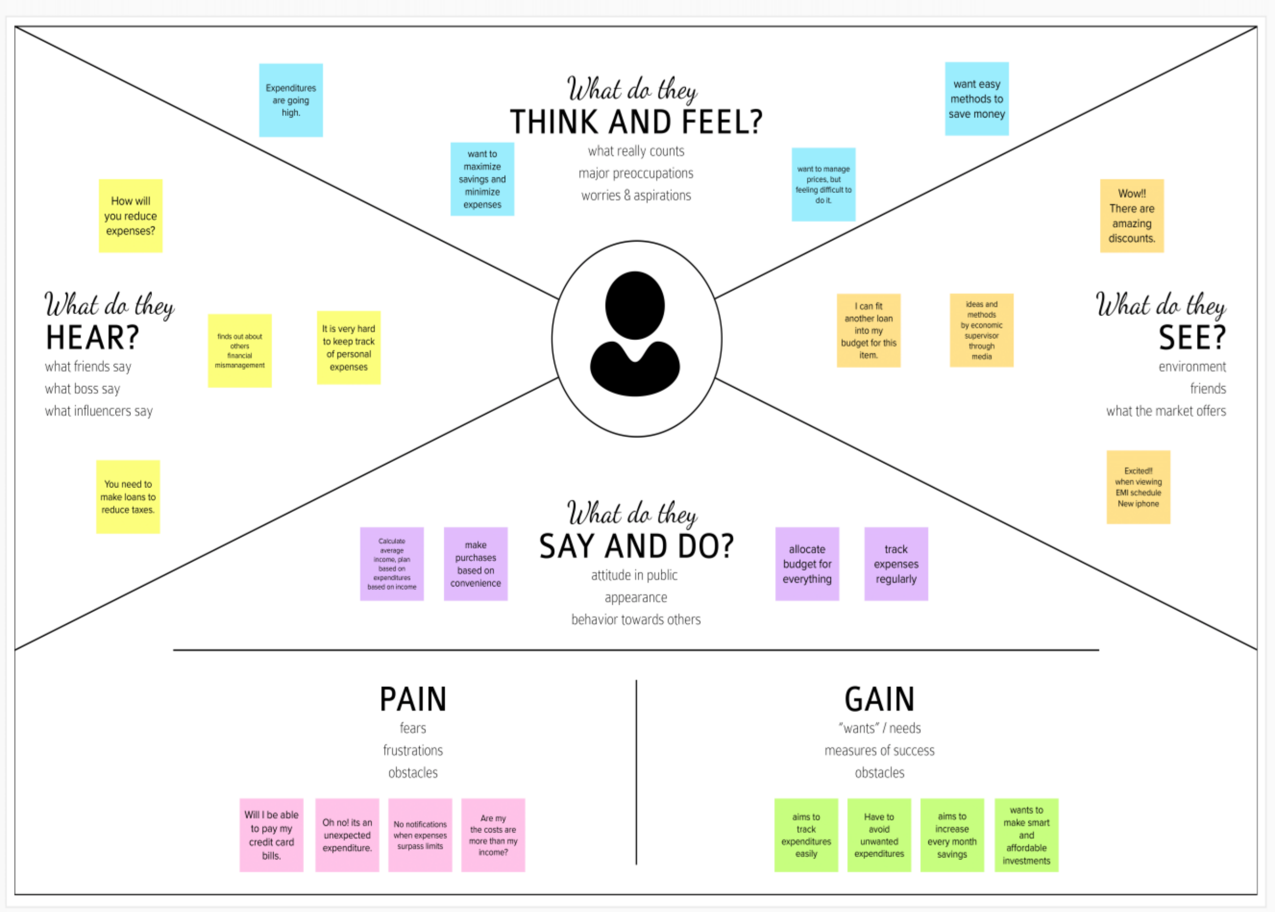
2.3 Problem Statement Definition

In order to guide the customers to manage all the expenses in their everyday life, a personal expense tracker has to be introduced to provide people with the best solution possible. The users arethe customers who need to manage expenses, and guide them through the various features of the application. So, an enhanced and smarter way of interaction with the customers has to be built to ensure efficient way of managing the expenses. In order to overcome the user satisfaction issues associated with tracking services, a Chabot will provide personal and efficient communication between the users. It is built to be the managing system that can facilitate customersto track expenses, add savings, loan management and getting reminders for the same.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviors and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



What do they THINK AND FEEL?

what really counts

major preoccupations

worries & aspirations

Expenditures are going high.

want to maximize savings and minimize expenses

want to manage prices, but feeling difficult to do it.

want easy methods to save money

What do they HEAR?

what friends say

what boss say

what influencers say

How will you reduce expenses?

finds out about others financial mismanagement

It is very hard to keep track of personal expenses

You need to make loans to reduce taxes.

What do they SEE?

environment

friends

what the market offers

Wow!! There are amazing discounts.

I can fit another loan into my budget for this item.

ideas and methods by economic supervisor through media

Excited!! when viewing EMI schedule New iphone

What do they SAY AND DO?

attitude in public

appearance

behavior towards others

Calculate average income, plan based on expenditures based on income

make purchases based on convenience

allocate budget for everything

track expenses regularly

PAIN

fears

frustrations

obstacles

Will I be able to pay my credit card bills.

Oh no! its an unexpected expenditure.

No notifications when expenses surpass limits

Are my the costs are more than my income?

GAIN

"wants" / needs

measures of success

obstacles

aims to track expenditures easily

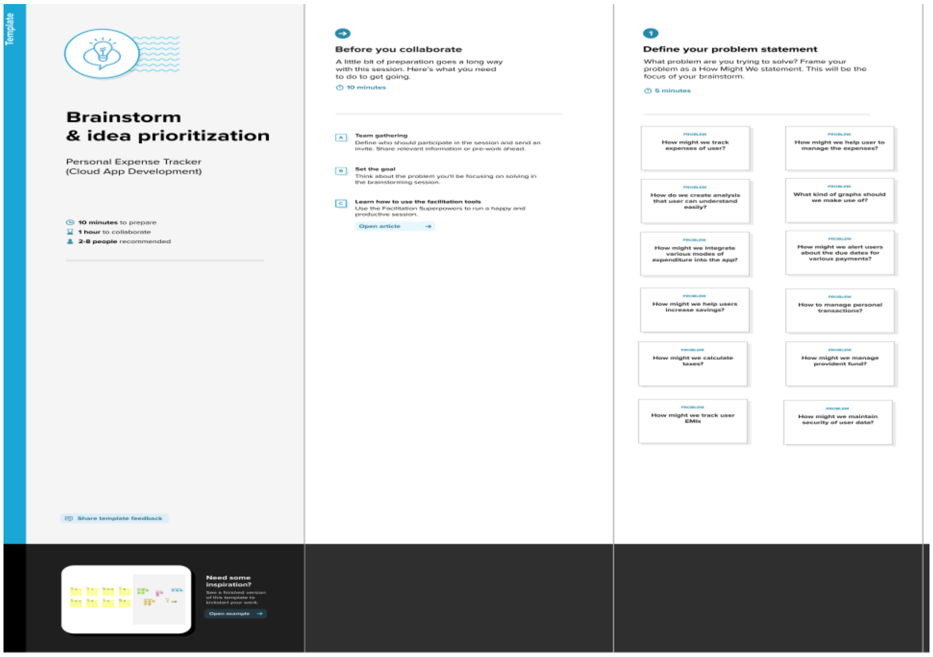
Have to avoid unwanted expenditures

aims to increase every month savings

wants to make smart and affordable investments

3.2 Ideation & BrainStroming

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.



Template

Brainstorm & idea prioritization

Personal Expense Tracker (Cloud App Development)

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

Share template feedback

Need some inspiration? See a finished version of this template to kickstart your work.

Open example

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

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

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

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

Open article

1 **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 expenses of user?

PROBLEM
How might we help user to manage the expenses?

PROBLEM
How do we create analysis that user can understand easily?

PROBLEM
What kind of graphs should we make use of?

PROBLEM
How might we integrate various modes of expenditure into the app?

PROBLEM
How might we alert users about the due dates for various payments?

PROBLEM
How might we help users increase savings?

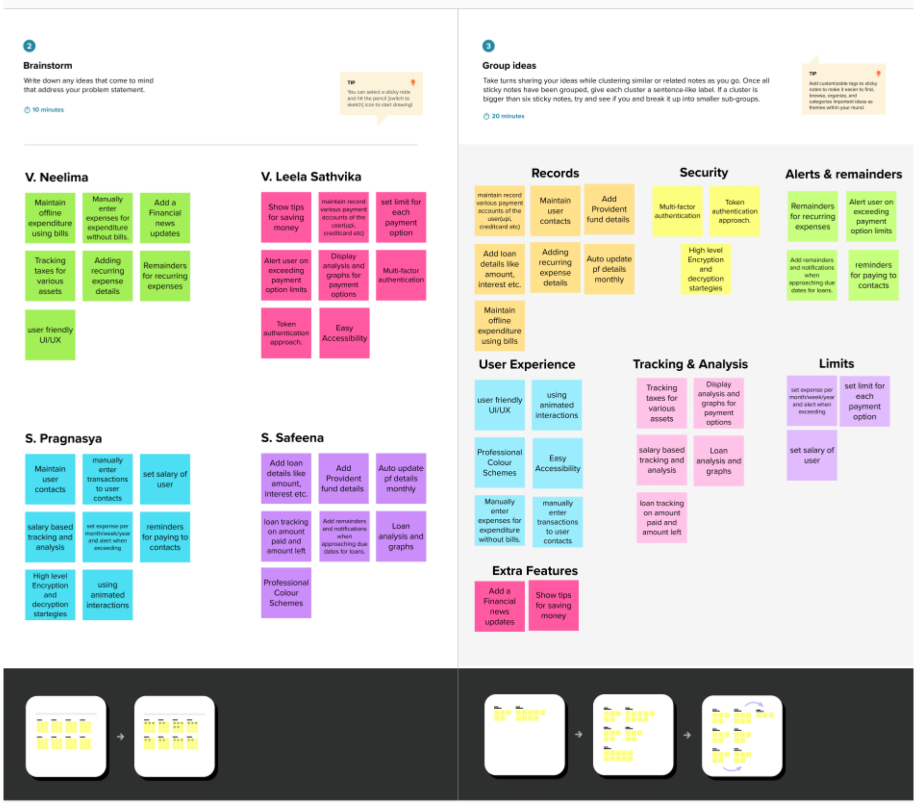
PROBLEM
How to manage personal transactions?

PROBLEM
How might we calculate taxes?

PROBLEM
How might we manage provident fund?

PROBLEM
How might we track user EMIs

PROBLEM
How might we maintain security of user data?



2 **Brainstorm**

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

10 minutes

TIP
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

V. Neelima

Maintain offline expenditure using bills
Manually enter expenses for expenditure without bills.
Add a Financial news updates
Tracking taxes for various assets
Adding recurring expense details
Remainders for recurring expenses
user friendly UI/UX

V. Leela Sathvika

Show tips for saving money
maintain record various payment accounts of the user(upi, creditcard etc)
set limit for each payment option
Alert user on exceeding payment option limits
Display analysis and graphs for payment options
Multi-factor authentication
Token authentication approach.
Easy Accessibility

S. Pragnasya

Maintain user contacts
manually enter transactions to user contacts
set salary of user
salary based tracking and analysis
set expense per month/week/year and alert when exceeding
reminders for paying to contacts
High level Encryption and decryption startegies
using animated interactions

S. Safeena

Add loan details like amount, interest etc.
Add Provident fund details
Auto update pf details monthly
loan tracking on amount paid and amount left
Add remainders and notifications when approaching due dates for loans.
Loan analysis and graphs
Professional Colour Schemes

3 **Group ideas**

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, 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

TIP
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Records

maintain record various payment accounts of the user(upi, creditcard etc)
Maintain user contacts
Add Provident fund details
Add loan details like amount, interest etc.
Adding recurring expense details
Auto update pf details monthly
Maintain offline expenditure using bills

Security

Multi-factor authentication
Token authentication approach.
High level Encryption and decryption startegies

Alerts & remainders

Remainders for recurring expenses
Alert user on exceeding payment option limits
Add remainders and notifications when approaching due dates for loans.
reminders for paying to contacts

User Experience

user friendly UI/UX
using animated interactions
Professional Colour Schemes
Easy Accessibility
Manually enter expenses for expenditure without bills.
manually enter transactions to user contacts

Tracking & Analysis

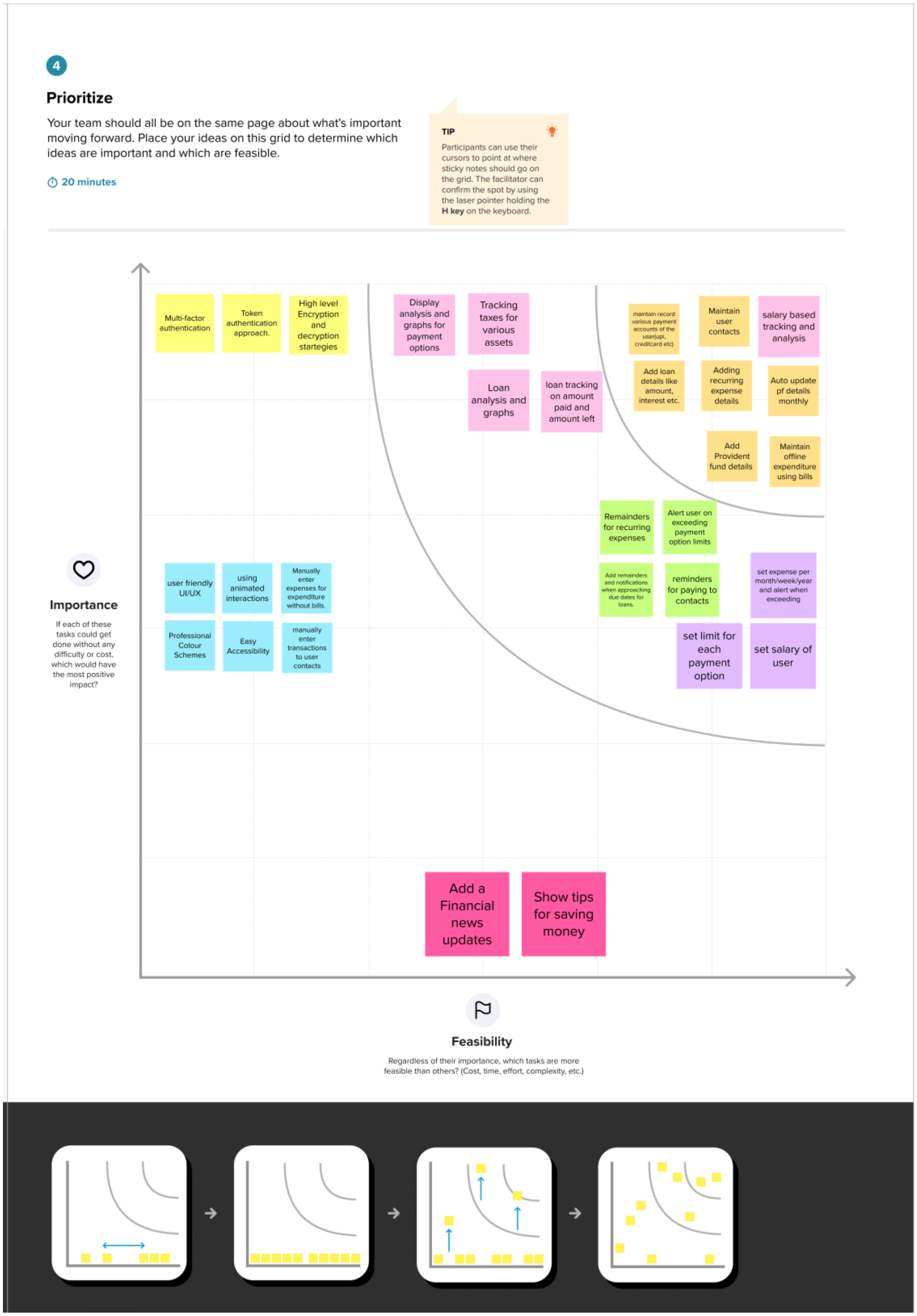
Tracking taxes for various assets
Display analysis and graphs for payment options
salary based tracking and analysis
Loan analysis and graphs
loan tracking on amount paid and amount left

Limits

set expense per month/week/year and alert when exceeding
set limit for each payment option
set salary of user

Extra Features

Add a Financial news updates
Show tips for saving money



4

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

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Importance

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

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

Multi-factor authentication
Token authentication approach.
High level Encryption and decryption startegies
Display analysis and graphs for payment options
Tracking taxes for various assets
Loan analysis and graphs
loan tracking on amount paid and amount left
maintain record various payment accounts of the user(upi, creditcard etc)
Maintain user contacts
salary based tracking and analysis
Add loan details like amount, interest etc.
Adding recurring expense details
Auto update pf details monthly
Add Provident fund details
Maintain offline expenditure using bills
Remainders for recurring expenses
Alert user on exceeding payment option limits
user friendly UI/UX
using animated interactions
Manually enter expenses for expenditure without bills.
Add remainders and notifications when approaching due dates for loans.
reminders for paying to contacts
set expense per month/week/year and alert when exceeding
Professional Colour Schemes
Easy Accessibility
manually enter transactions to user contacts
set limit for each payment option
set salary of user
Add a Financial news updates
Show tips for saving money

Shortlisted Ideas

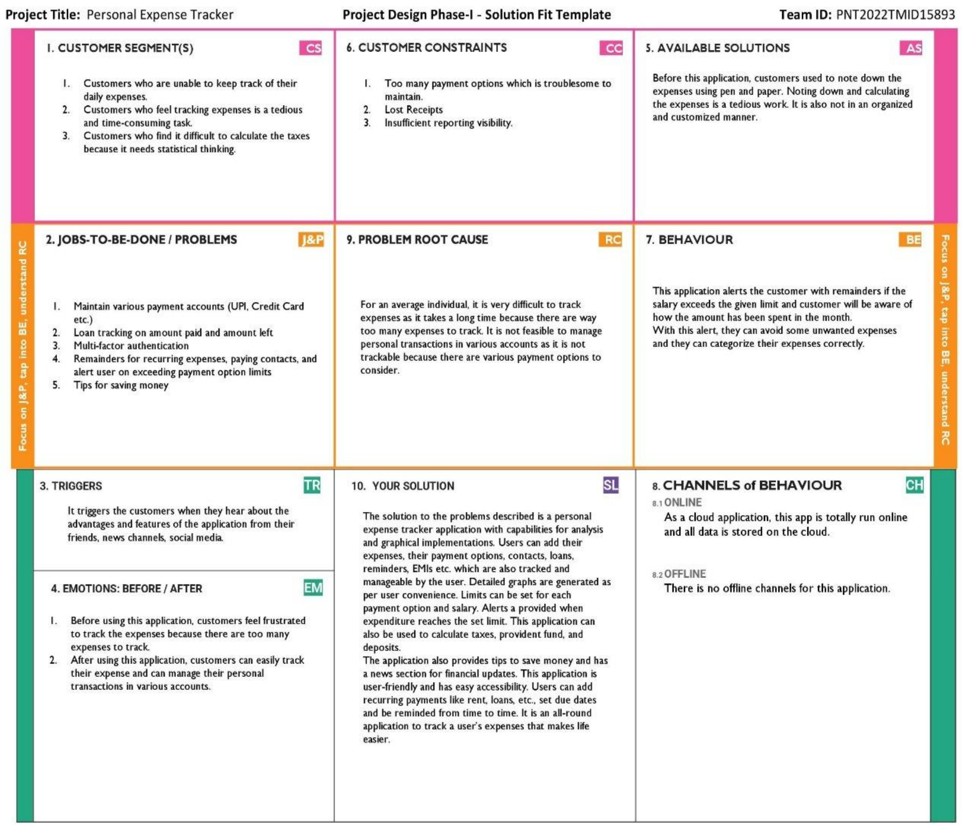
- Idea 1: Make it user friendly
- Idea 2: Collect only the required details from the customer.
- Idea 3: Build pre-requisite questions for queries regarding some type of loan.

3.3. Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	For an average individual, It is very difficult to track expenses as it takes a long time because there are way too many expenses to track. It is not feasible to manage personal transactions in various accounts as it is not trackable because there are various payment options to consider. This makes it hard to save money because of high unmanageable expenditures. It is also difficult to calculate the taxes because needs statistical thinking and management of various assets such as vehicles, properties, salary etc. Users try to analyze expenses but they can’t do it efficiently because it may involve a lot of data and graphical representations which makes it complicated. Users feel frustrated, confused, stressed etc. which makes them give up.
2.	Idea / Solution description	The solution to the problems described is a personal expense tracker application with capabilities for analysis and graphical implementations. Users can add their expenses, their payment options, contacts, loans, reminders, EMIs etc which are also tracked and manageable by the user. Detailed graphs are generated as per user convenience. Limits can be set for each payment option and salary. Alerts a provided when expenditure reaches the set limit. This application can also be used to calculate taxes, provident fund and deposits. The application also provides tips to save money and has a news section for financial updates. This application is user-friendly and has easy accessibility. Users can add recurring payments like rent, loans, etc, set due dates and be reminded from time to time. It is an all-round application to track a user’s expenses that makes life easier.
3.	Novelty / Uniqueness	It allows users to generate graphs for various expenses. It allows to set limits on finances and be alerted. It reminds us of recurring payments. Helps calculate taxes.

4.	Social Impact / Customer Satisfaction	It has a major impact on users as it helps them efficiently manage their finances. It reduces stress, anxiety and complication for users. It allows them to save considerable amount of money and time as time is money. It provides immense customer satisfaction as it is very user-friendly and is easily accessible. It is a place where user can find all expense and finance related stuff.
5	Business Model (Revenue Model)	The revenue is generated as users more users use the application.
6.	Scalability of the Solution	This application is very scalable as it can meet all the user's finance related demands

3.4. Problem Solution fit



Project Title: Personal Expense Tracker

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID15893

1. CUSTOMER SEGMENT(S) CS

1. Customers who are unable to keep track of their daily expenses.
2. Customers who feel tracking expenses is a tedious and time-consuming task.
3. Customers who find it difficult to calculate the taxes because it needs statistical thinking.

6. CUSTOMER CONSTRAINTS CC

1. Too many payment options which is troublesome to maintain.
2. Lost Receipts
3. Insufficient reporting visibility.

5. AVAILABLE SOLUTIONS AS

Before this application, customers used to note down the expenses using pen and paper. Noting down and calculating the expenses is a tedious work. It is also not in an organized and customized manner.

2. JOBS-TO-BE-DONE / PROBLEMS J&P

1. Maintain various payment accounts (UPI, Credit Card etc.)
2. Loan tracking on amount paid and amount left
3. Multi-factor authentication
4. Remainders for recurring expenses, paying contacts, and alert user on exceeding payment option limits
5. Tips for saving money

9. PROBLEM ROOT CAUSE RC

For an average individual, it is very difficult to track expenses as it takes a long time because there are way too many expenses to track. It is not feasible to manage personal transactions in various accounts as it is not trackable because there are various payment options to consider.

7. BEHAVIOUR BE

This application alerts the customer with remainders if the salary exceeds the given limit and customer will be aware of how the amount has been spent in the month.
With this alert, they can avoid some unwanted expenses and they can categorize their expenses correctly.

3. TRIGGERS TR

It triggers the customers when they hear about the advantages and features of the application from their friends, news channels, social media.

4. EMOTIONS: BEFORE / AFTER EM

1. Before using this application, customers feel frustrated to track the expenses because there are too many expenses to track.
2. After using this application, customers can easily track their expense and can manage their personal transactions in various accounts.

10. YOUR SOLUTION SL

The solution to the problems described is a personal expense tracker application with capabilities for analysis and graphical implementations. Users can add their expenses, their payment options, contacts, loans, reminders, EMIs etc. which are also tracked and manageable by the user. Detailed graphs are generated as per user convenience. Limits can be set for each payment option and salary. Alerts a provided when expenditure reaches the set limit. This application can also be used to calculate taxes, provident fund, and deposits.
The application also provides tips to save money and has a news section for financial updates. This application is user-friendly and has easy accessibility. Users can add recurring payments like rent, loans, etc., set due dates and be reminded from time to time. It is an all-round application to track a user's expenses that makes life easier.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
As a cloud application, this app is totally run online and all data is stored on the cloud.

8.2 OFFLINE
There is no offline channels for this application.

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

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

The 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 Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email
FR-3	User Login	Authentication
FR-4	Dashboard	Small graph previews, news updates, monthly savings and expense previews.
FR-5	Navigation Side Menu	Nav buttons to all modules, Sign out, Settings and profile module
FR-6	Expense Tracker Module	Add Expense, Delete Expense, Modify Expense, View Expenditure Graph,
FR-7	Remainders Module	Add Remainder, Add Recurring Expense Payment Remainder, Add Loan Remainders
FR-8	Savings and Finance Module	Add Monthly income, Modify monthly income, Add Other sources of income, View Savings graph
FR-9	User Profile Module	Edit User Profile
FR-10	Settings Module	Edit settings

4.2 Non-Functional requirements

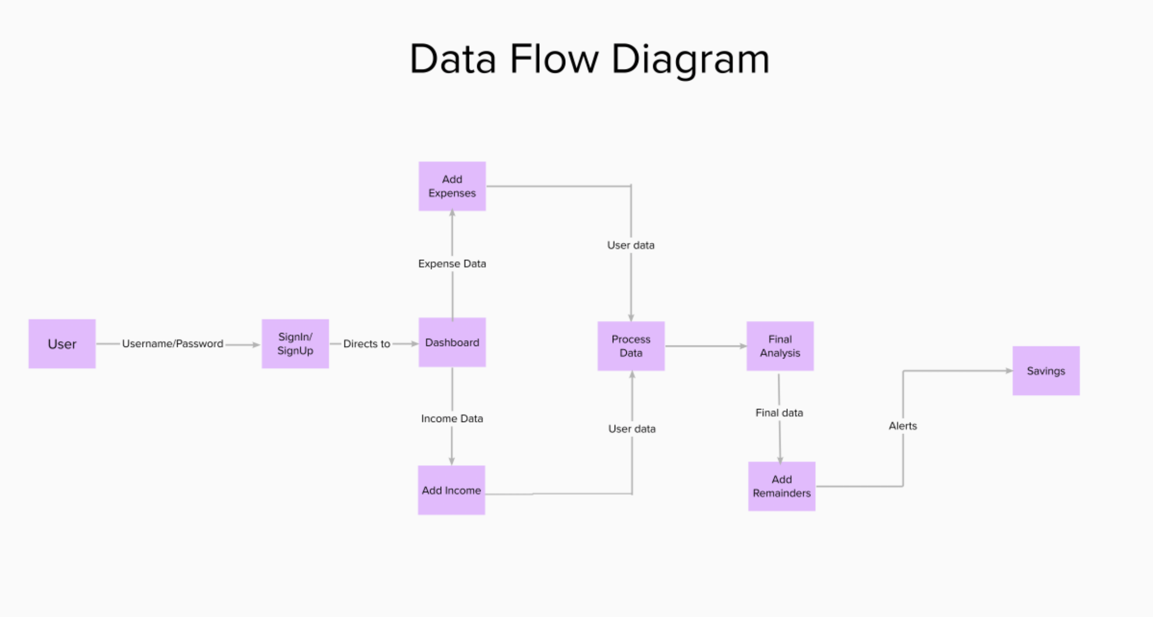
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The solution is easy to use as the UI is made to be accessible and user-friendly. Easy navigation is provided through an integrated sidemenu and chatbot assistance.
NFR-2	**Security**	Application is highly secure as all data is encrypted using a secure encryption algorithm and user data is accessible only on authentication.
NFR-3	**Reliability**	Application is highly reliable as it is deployed with IBM cloud assistance.
NFR-4	**Performance**	Performance is stable and smooth as it is very light weight application built with flask.
NFR-5	**Availability**	It is available easily as it is deployed on the internet.
NFR-6	**Scalability**	Application is scalable as it uses IBM cloud resources and microservices architecture. Extensions and modifications are done easily.

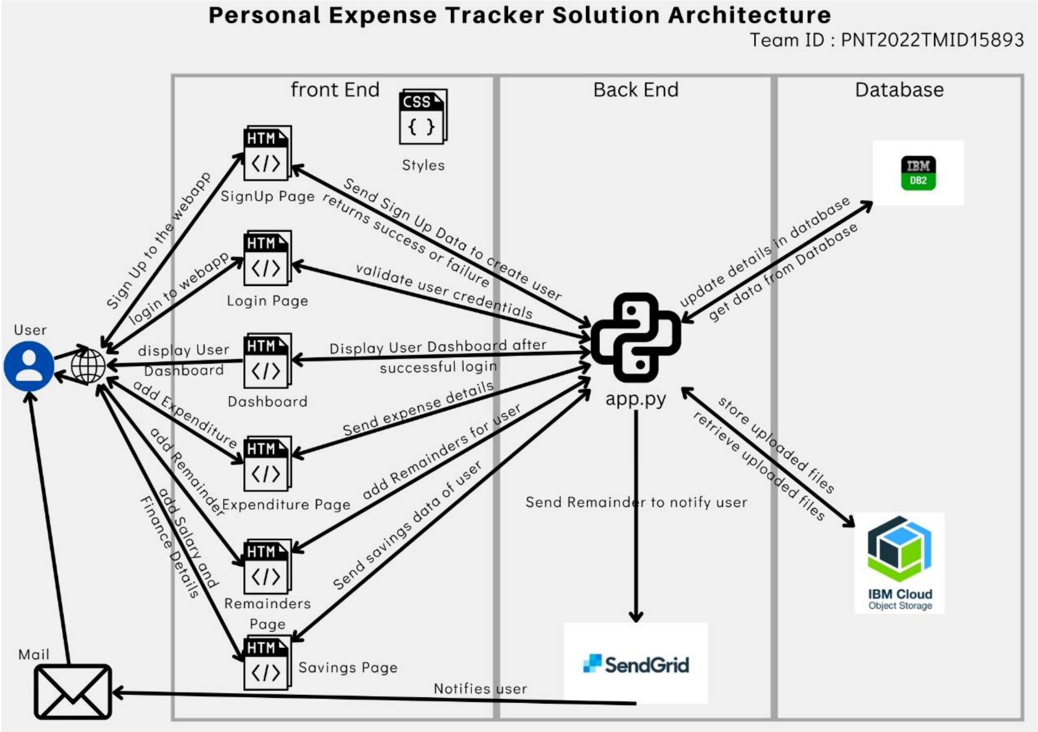
5. PROJECT DESIGN

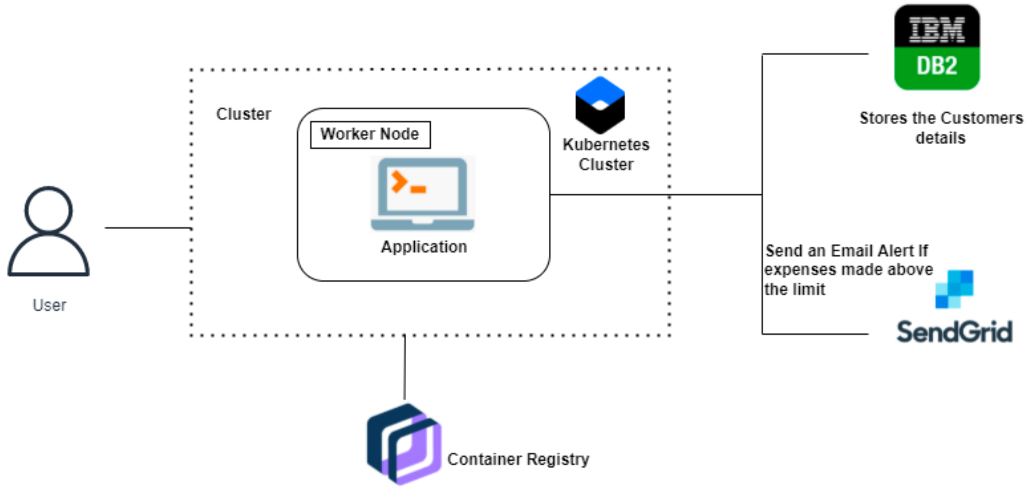
5.1 Data Flow Diagrams

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 (Webuser)	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	Sprint-1
		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	Sprint-1
		USN-3	As a user, I can register for the application through social media	I can register & access thedashboard with Social Media Login	Low	Sprint-1
		USN-4	As a user, I can register for the application through Gmail	I can register & access thedashboard with Gmail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can access the application	High	Sprint-1
	Dashboard	USN-6	As a user, I can navigate the different features of the application	I can do all CRUD operations	High	Sprint-1

	Home	USN-7	As a user, I can get the overview of the expenses.	I can view the expenses.	High	Sprint-1
	Track expenses- Add expenses	USN-8	As a user, I can able to add the new expenditure details.	I can add new expenses	High	Sprint-2
	Track expenses-graph	USN-9	As a user, I can able to get the details of the expenses in the form of a graph.	I can analyse the expenses	High	Sprint-2
	Payment Options- Add Card Details	USN-10	As a user, I can able to add a various card payment details.	I can add payment details from different cards.	Medium	Sprint-3
	Payment Options-Graph	USN-11	As a user, I can able to get the details of various card payment details in the form of a graph.	I can analyse the various payment options.	Medium	Sprint-3
	Loan Tracker-AddLoan	USN-12	As a user, I can be able to add new loan details in my application.	I can add new loan details	Low	Sprint-4
	Loan Tracker-Graph	USN-13	As a user, I can be able to get the details ofthe loan in the form a graph.	I can analyse the loandetails	Low	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

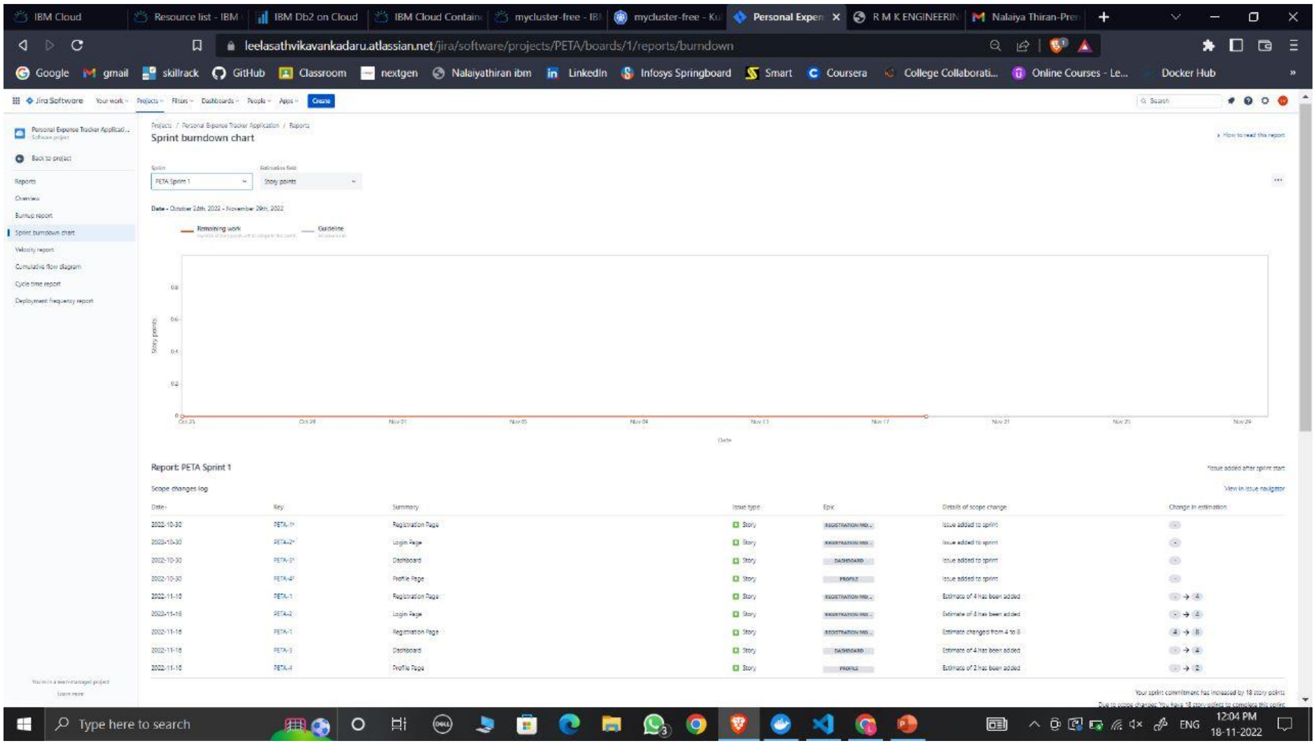
Team Members-

- VLS- Vankadaru Leela Sathvika
- PS- Pragnasya S
- SS- Shaik Safeena
- VN- Vayugandla Neelima

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.	4	High	PS, VLS
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	2	High	PS, VLS
Sprint-1		USN-3	As a user, I can register for the application through Gmail	2	Medium	PS, VLS
Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	4	High	VN, SS
Sprint-1	Dashboard	USN-5	As a user, I can navigate the different features of the application. I can get the overview of the expenses.	4	High	PS, VN, SS, VLS
Sprint-1	Profile	USN-6	As a User, I can view and change my profile settings.	2	Medium	VN, SS
Sprint-2	Track Expenses Page	USN-7	Consolidated view of all expense related operations and data	7	High	PS, VLS
Sprint-2	Track Expenses – Add Expenses	USN-8	As a user, I can able to add the new expenditure details	7	High	PS, SS
Sprint-2	Track Expenses - Graph	USN-9	As a user, I can able to get the details of the expenses in the form of a graph.	6	Low	SS, VN

Sprint-3	Track Savings Page	USN-10	Consolidated view of all savings related operations and data	7	High	PS, VLS
Sprint-3	Track Savings – Add Personal Income Details	USN-11	As a user, I can able to add a various income details.	7	High	SS, VN
Sprint-3	Track Savings – Graph	USN-12	As a user, I can able to get the details of various card payment details in the form of a graph	6	Low	SS, VN, VLS
Sprint-4	Track Reminders Page	USN-13	Consolidated view of all reminders related operations and data	6	High	PS, VLS
Sprint-4	Reminders – Add Remainder	USN-14	As a user, I can be able to add new reminders for payments in my application.	6	High	PS, VN
Sprint-4	Reminders - Add Recurring Reminder	USN-15	As a user, I can be able to add new reminders for recurring payments such as EMIS and loans in my application.	4	Medium	SS, VN
Sprint-4	Loan Tracker Graph	USN-16	As a user, I can be able to get the details of the loan in the form a graph	4	Low	SS, VLS

6.3 Reports from JIRA



Projects / Personal Expense Tracker Application / Reports

Sprint burndown chart

Sprint: PETA Sprint 1

Estimation field: Story points

Date - October 24th, 2022 - November 29th, 2022

Report: PETA Sprint 1

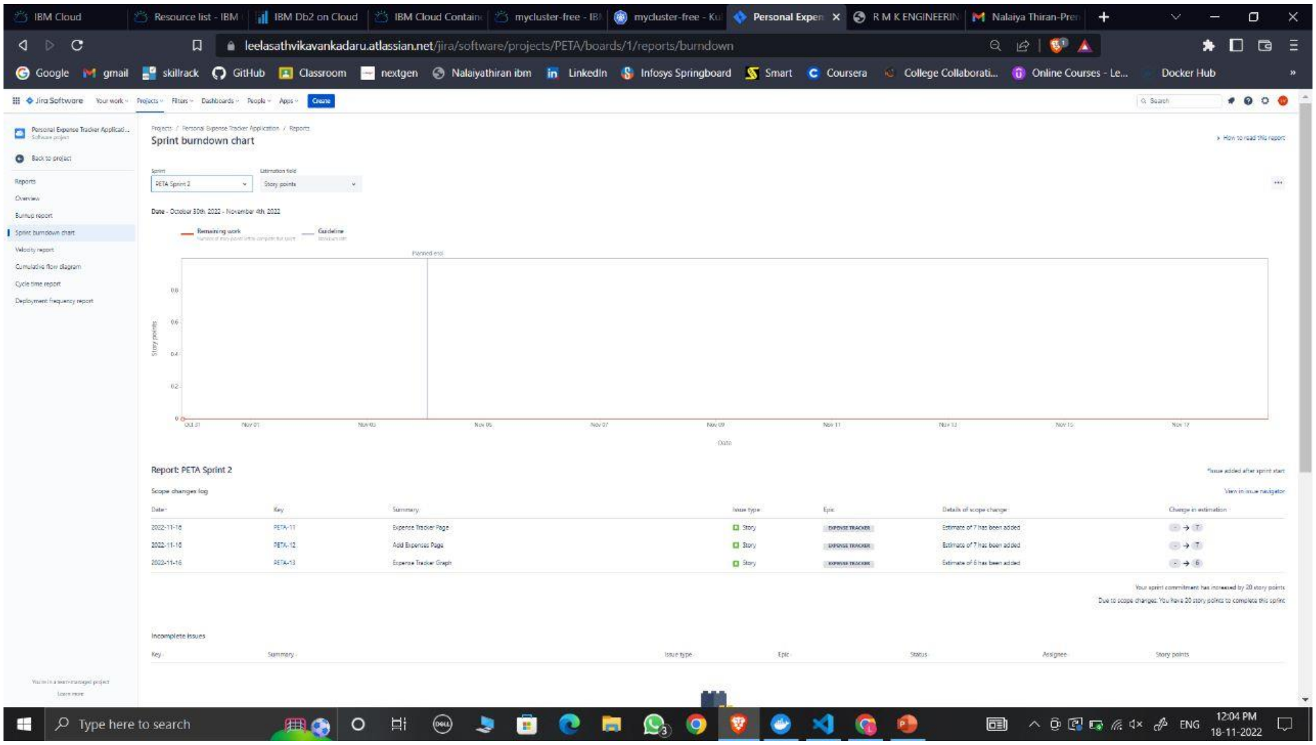
*Issue added after sprint start

Scope changes log

View in issue navigator

Date	Key	Summary	Issue type	Epic	Details of scope change	Change in estimation
2022-10-30	PETA-1*	Registration Page	Story	REGISTRATION MO...	Issue added to sprint	-
2022-10-30	PETA-2*	Login Page	Story	REGISTRATION MO...	Issue added to sprint	-
2022-10-30	PETA-3*	Dashboard	Story	DASHBOARD	Issue added to sprint	-
2022-10-30	PETA-4*	Profile Page	Story	PROFILE	Issue added to sprint	-
2022-11-16	PETA-1	Registration Page	Story	REGISTRATION MO...	Estimate of 4 has been added	- → 4
2022-11-16	PETA-2	Login Page	Story	REGISTRATION MO...	Estimate of 4 has been added	- → 4
2022-11-16	PETA-1	Registration Page	Story	REGISTRATION MO...	Estimate changed from 4 to 8	4 → 8
2022-11-16	PETA-3	Dashboard	Story	DASHBOARD	Estimate of 4 has been added	- → 4
2022-11-16	PETA-4	Profile Page	Story	PROFILE	Estimate of 2 has been added	- → 2

Your sprint commitment has increased by 18 story points



Sprint burndown chart

Date - October 30th, 2022 - November 4th, 2022

Report: PETA Sprint 2

*Issue added after sprint start

Scope changes log

View in issue navigator

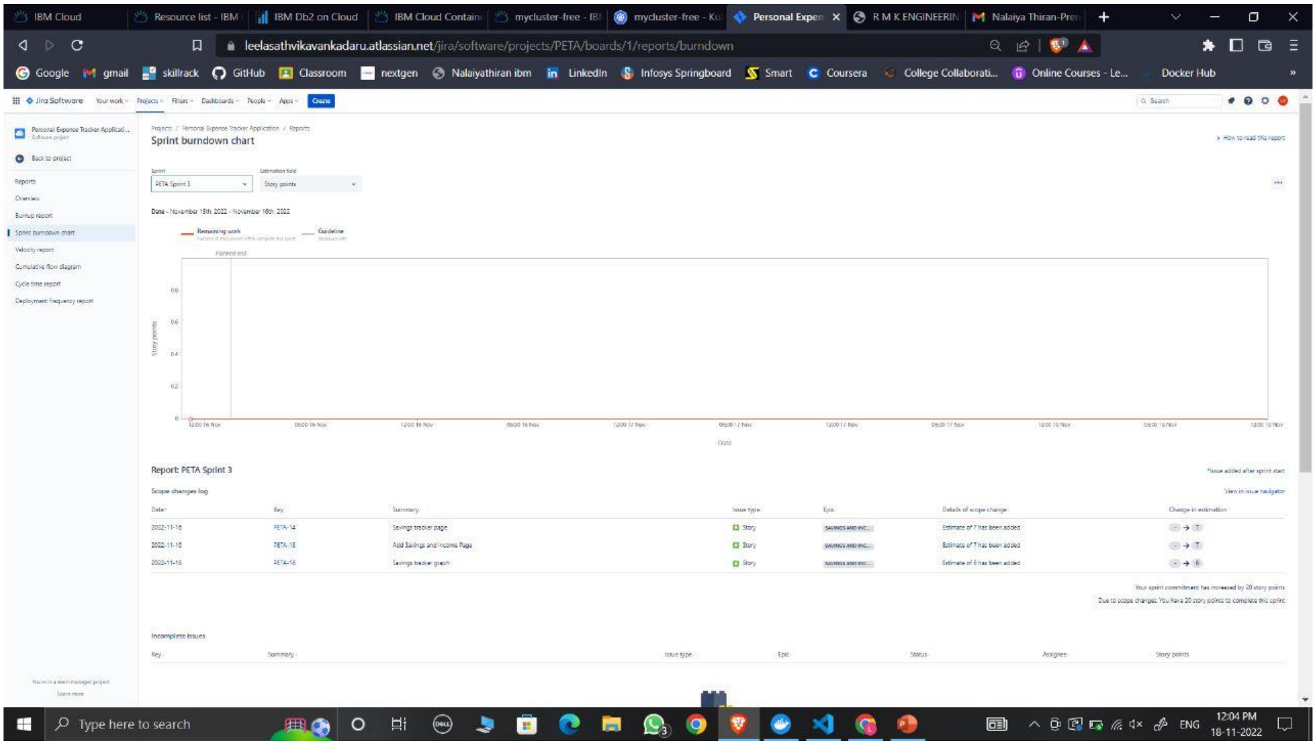
Date	Key	Summary	Issue type	Epic	Details of scope change	Change in estimation
2022-11-16	PETA-11	Expense Tracker Page	Story	EXPENSE TRACKER	Estimate of 7 has been added	- → 7
2022-11-16	PETA-12	Add Expenses Page	Story	EXPENSE TRACKER	Estimate of 7 has been added	- → 7
2022-11-16	PETA-13	Expense Tracker Graph	Story	EXPENSE TRACKER	Estimate of 6 has been added	- → 6

Your sprint commitment has increased by 20 story points

Due to scope changes: You have 20 story points to complete this sprint

Incomplete issues

Key	Summary	Issue type	Epic	Status	Assignee	Story points



Sprint burndown chart

Date - November 15th, 2022 - November 16th, 2022

Report: PETA Sprint 3

*Issue added after sprint start

Scope changes log

View in issue navigator

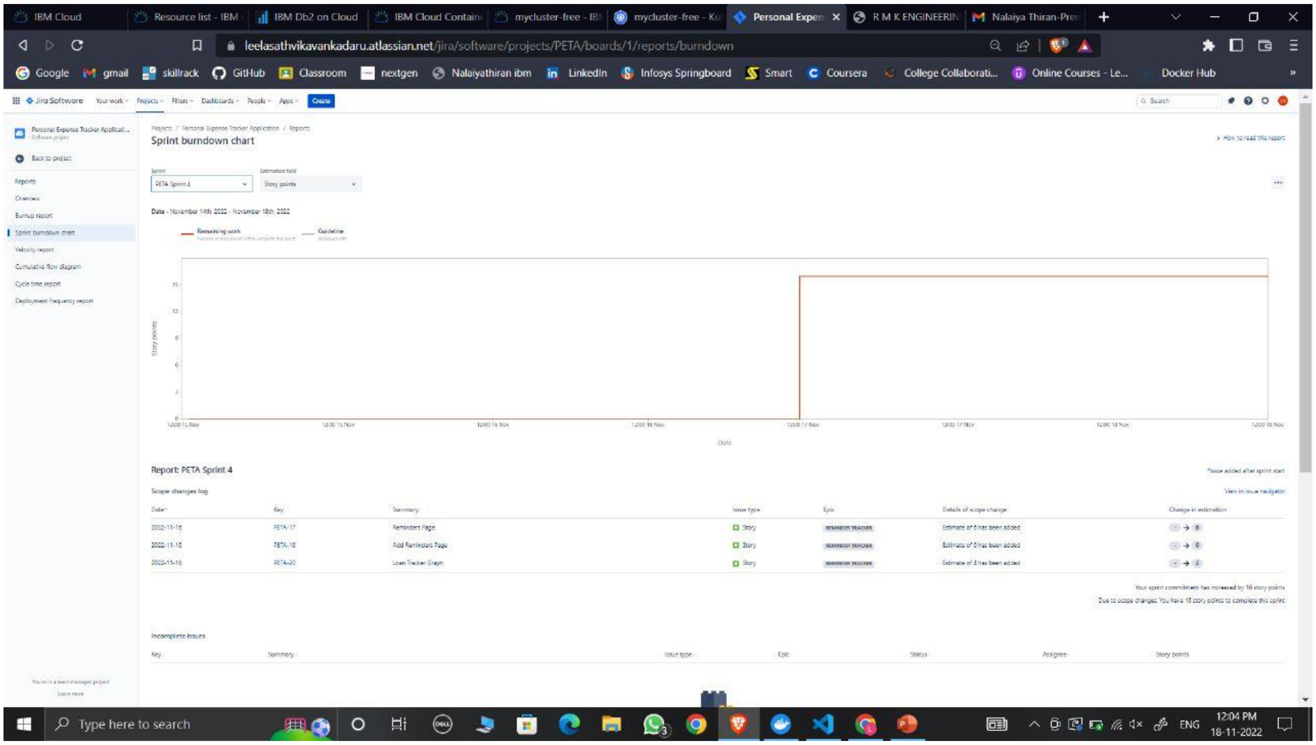
Date	Key	Summary	Issue type	Epic	Details of scope change	Change in estimation
2022-11-16	PETA-14	Savings tracker page	Story	SAVINGS AND INC...	Estimate of 7 has been added	- → 7
2022-11-16	PETA-15	Add Savings and Income Page	Story	SAVINGS AND INC...	Estimate of 7 has been added	- → 7
2022-11-16	PETA-16	Savings tracker graph	Story	SAVINGS AND INC...	Estimate of 6 has been added	- → 6

Your sprint commitment has increased by 20 story points

Due to scope changes: You have 20 story points to complete this sprint

Incomplete issues

Key	Summary	Issue type	Epic	Status	Assignee	Story points



Projects / Personal Expense Tracker Application / Reports

Sprint burndown chart

Sprint: PETA Sprint 4

Estimation field: Story points

Date : November 14th 2022 - November 18th 2022

Report: PETA Sprint 4

*Issue added after sprint start

Scope changes log

View in issue navigator

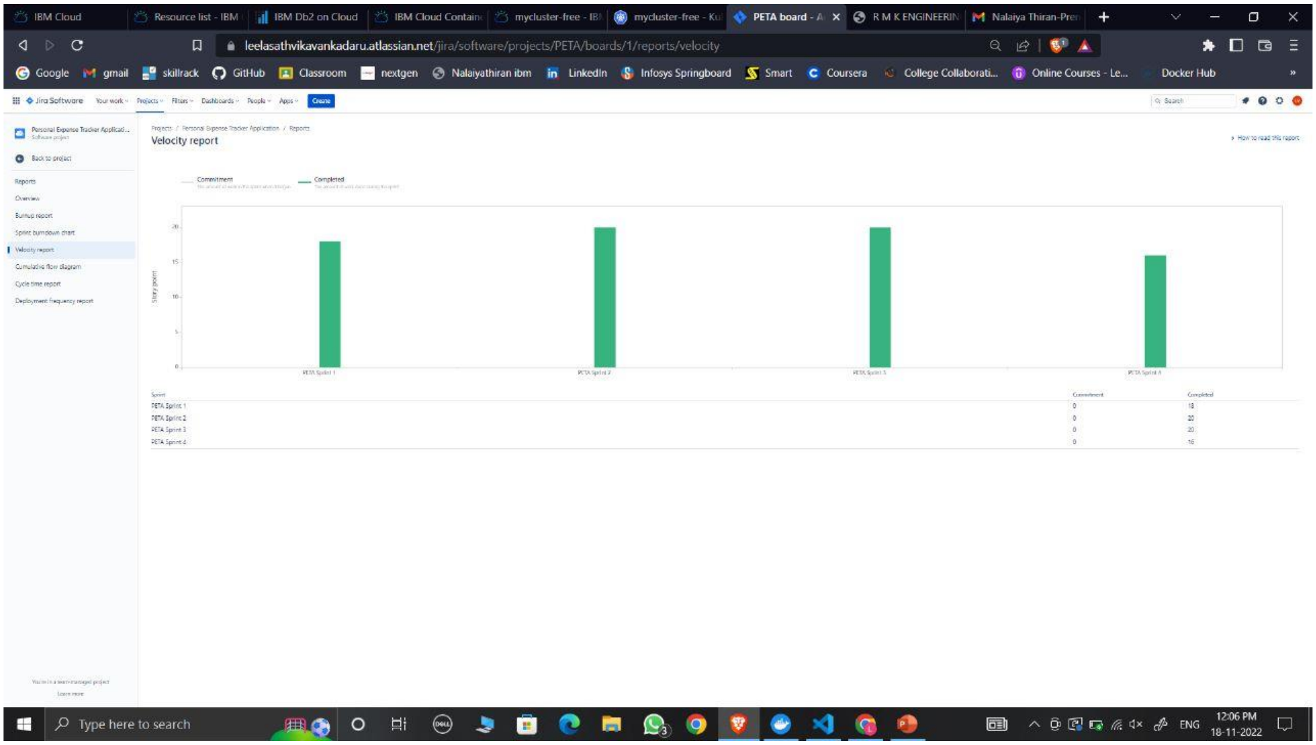
Date	Key	Summary	Issue type	Epic	Details of scope change	Change in estimation
2022-11-16	PETA-17	Reminders Page	Story	REMINDER TRACKER	Estimate of 6 has been added	- → 6
2022-11-16	PETA-18	Add Reminders Page	Story	REMINDER TRACKER	Estimate of 6 has been added	- → 6
2022-11-16	PETA-20	Loan Tracker Graph	Story	REMINDER TRACKER	Estimate of 4 has been added	- → 4

Your sprint commitment has increased by 16 story points

Due to scope changes. You have 16 story points to complete this sprint

Incomplete issues

Key	Summary	Issue type	Epic	Status	Assignee	Story points



Projects / Personal Expense Tracker Application / Reports

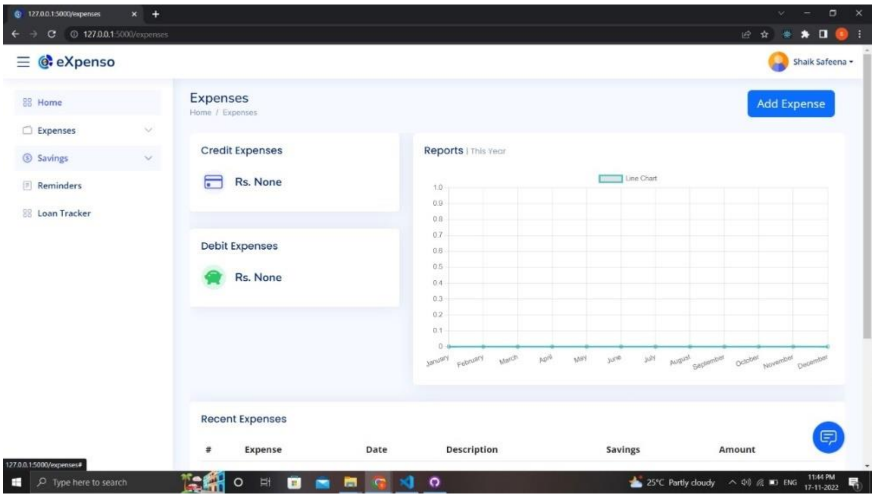
Velocity report

Sprint	Commitment	Completed
PETA Sprint 1	0	18
PETA Sprint 2	0	20
PETA Sprint 3	0	20
PETA Sprint 4	0	16

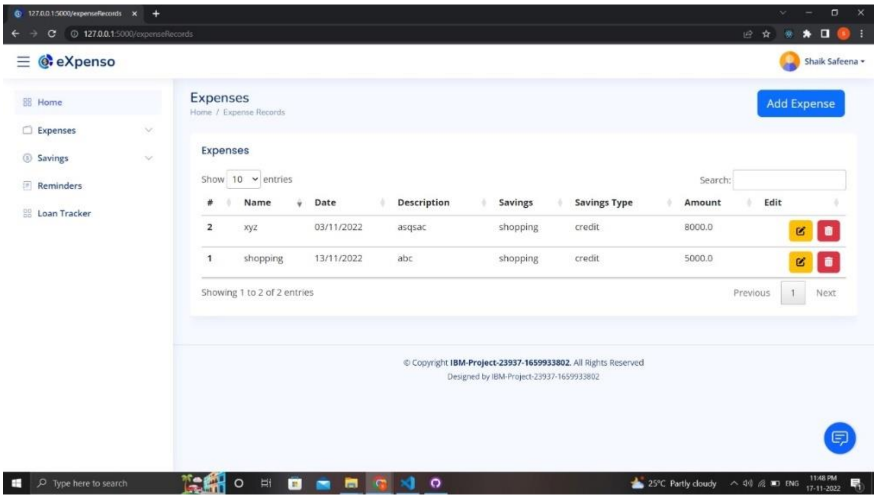
7. CODING & SOLUTIONING

7.1 Feature 1

7.1.1 Expenses



7.1.2 Expenses -Record



Expenses

Show 10 entries Search:

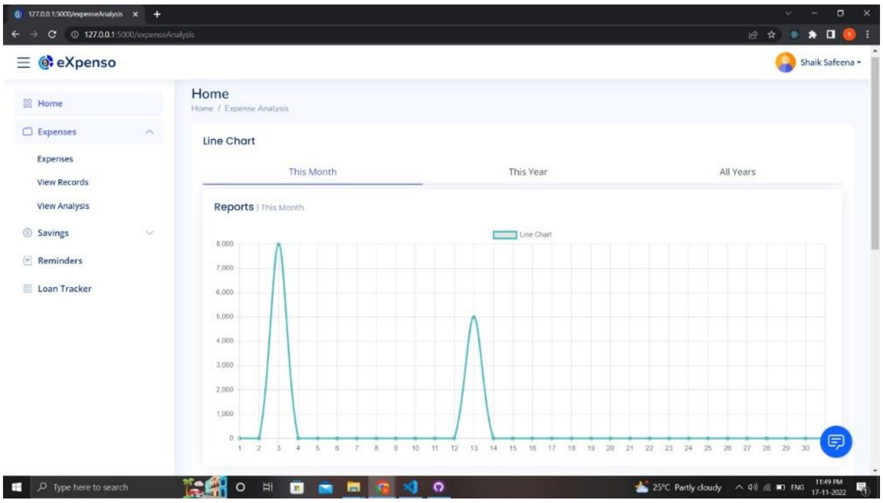
#	Name	Date	Description	Savings	Savings Type	Amount	Edit
2	xyz	03/11/2022	asqsac	shopping	credit	8000.0	
1	shopping	13/11/2022	abc	shopping	credit	5000.0	

Showing 1 to 2 of 2 entries Previous 1 Next

© Copyright **IBM-Project-23937-1659933802**. All Rights Reserved

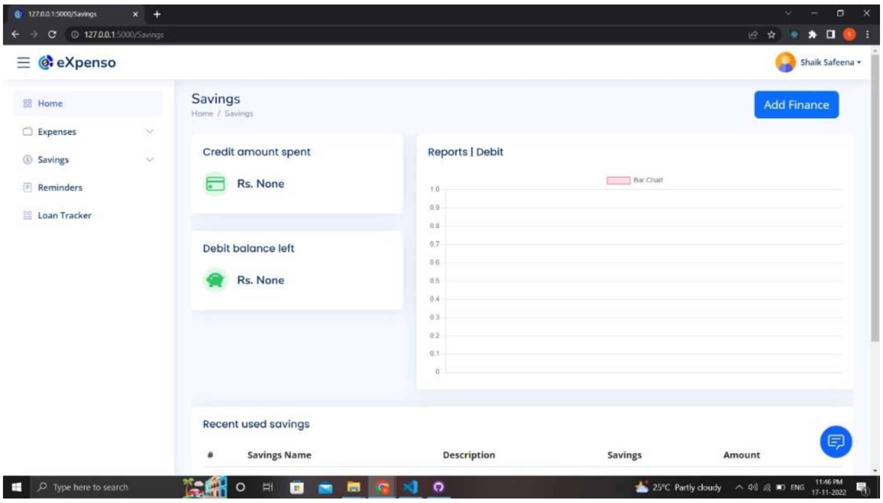
Designed by IBM-Project-23937-1659933802

7.1.3 Expenses - Analysis



7.2 Feature 2

7.2.1 Savings



Credit amount spent

Rs. None

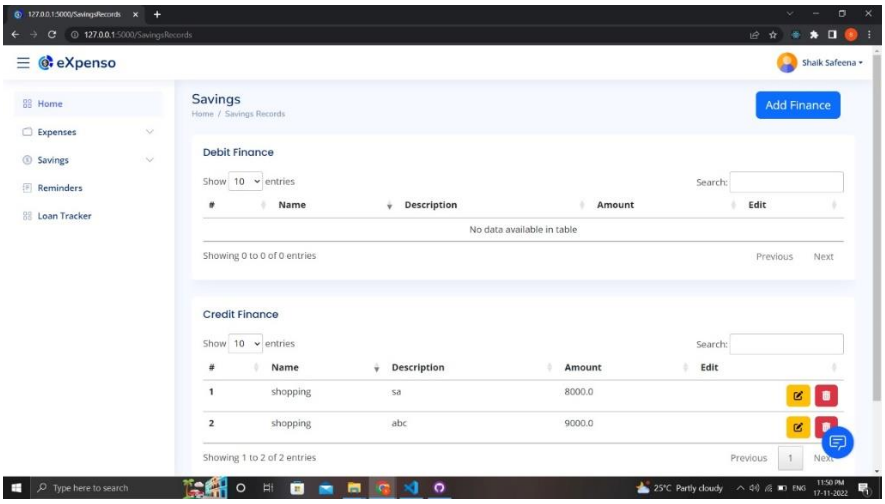
Debit balance left

Rs. None

Recent used savings

#	Savings Name	Description	Savings	Amount

7.2.2 Savings- Records



Debit Finance

Show 10 entries

Search:

#	Name	Description	Amount	Edit
No data available in table				

Showing 0 to 0 of 0 entries

Previous Next

Credit Finance

Show 10 entries

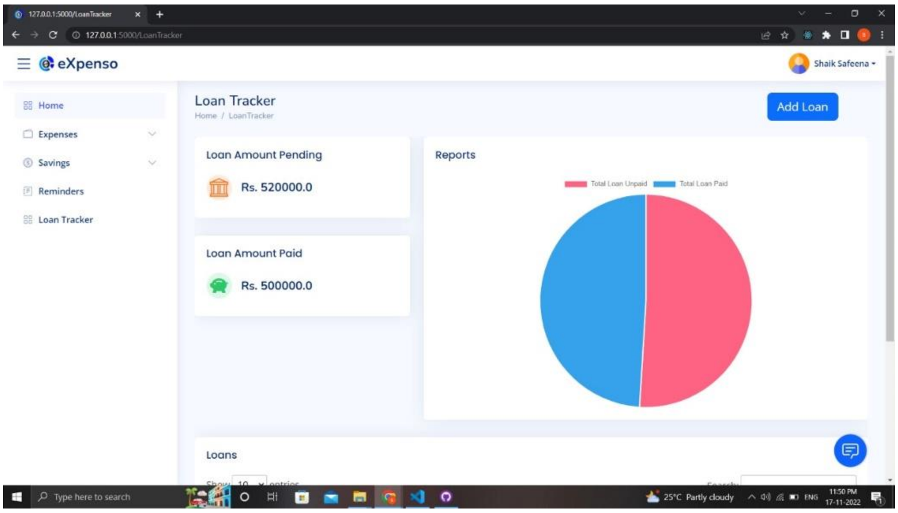
Search:

#	Name	Description	Amount	Edit
1	shopping	sa	8000.0	
2	shopping	abc	9000.0	

Showing 1 to 2 of 2 entries

Previous 1 Next

7.2.2 Savings- Analysis



Loan Amount Pending

Rs. 520000.0

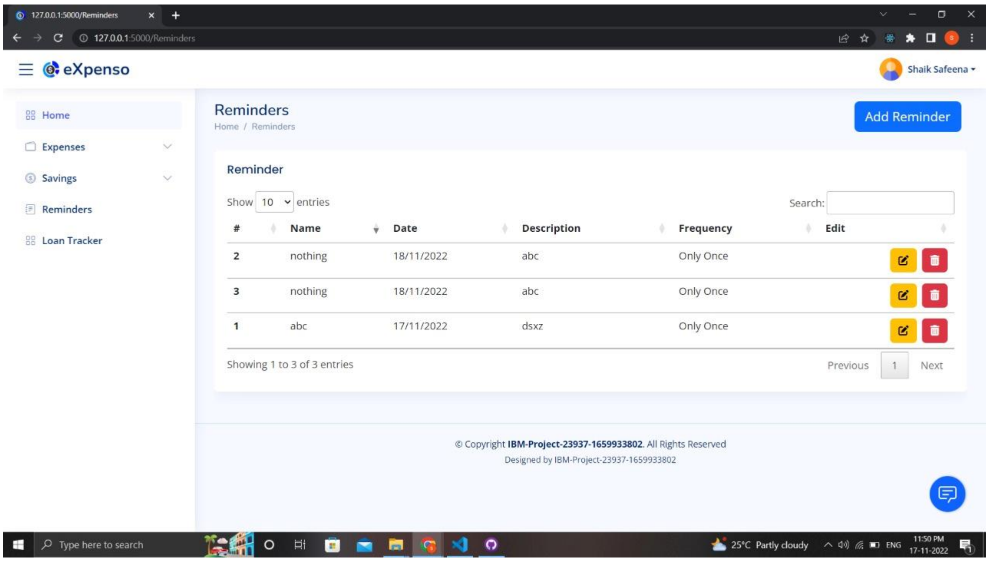
Loan Amount Paid

Rs. 500000.0

Loans

7.3 Feature 3

7.3.1 Reminders



Reminders

Home / Reminders

Reminder

Show 10 entries Search:

#	Name	Date	Description	Frequency	Edit
2	nothing	18/11/2022	abc	Only Once	
3	nothing	18/11/2022	abc	Only Once	
1	abc	17/11/2022	dsxz	Only Once	

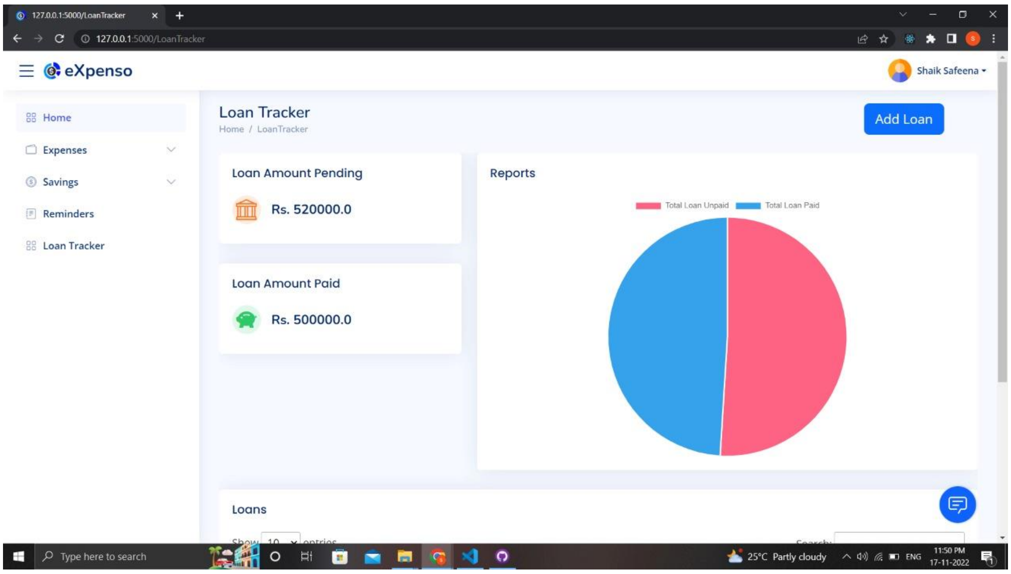
Showing 1 to 3 of 3 entries Previous 1 Next

© Copyright **IBM-Project-23937-1659933802**. All Rights Reserved

Designed by IBM-Project-23937-1659933802

7.4 Feature 4

7.4.1 Loan Tracker



Loan Tracker

Home / LoanTracker

Loan Amount Pending

Rs. 520000.0

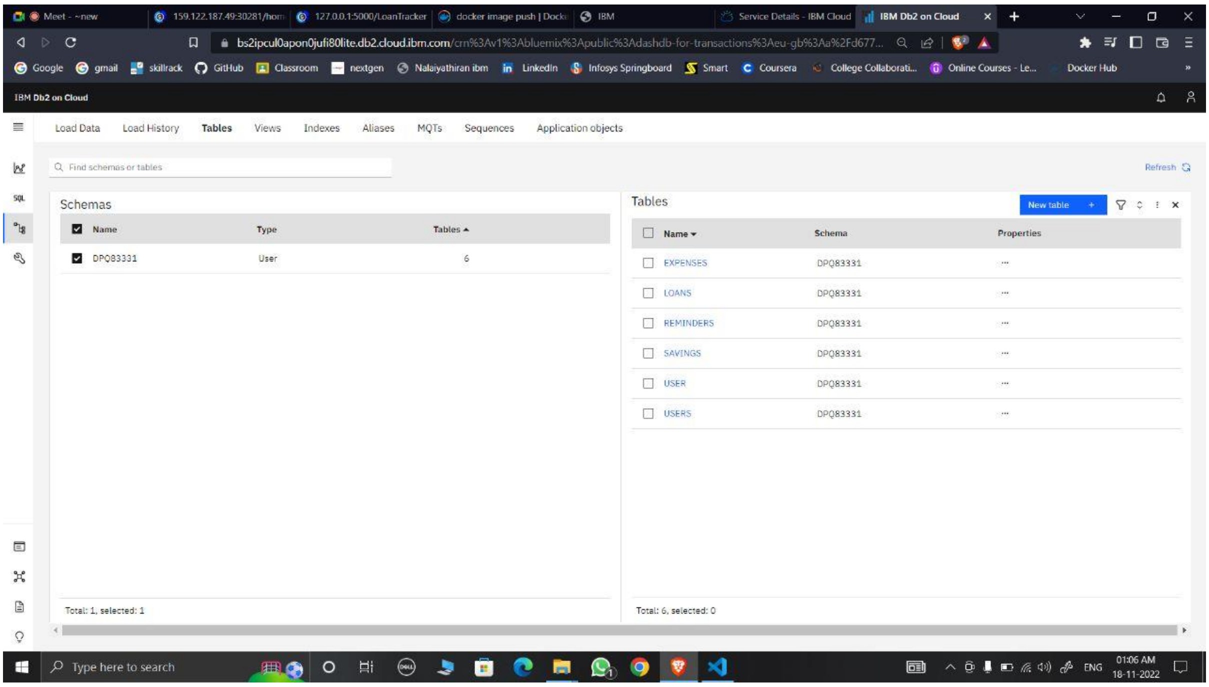
Loan Amount Paid

Rs. 500000.0

Loans

Show 10 entries Search:

7.5 Database Schema



Schemas

☑	Name	Type	Tables ▲
☑	DPQ83331	User	6

Total: 1, selected: 1

Tables

□	Name ▾	Schema	Properties
□	EXPENSES	DPQ83331	···
□	LOANS	DPQ83331	···
□	REMINDERS	DPQ83331	···
□	SAVINGS	DPQ83331	···
□	USER	DPQ83331	···
□	USERS	DPQ83331	···

Total: 6, selected: 0

8 TESTING

8.1 Test Cases

S.No	Test Cases	Passed/ Failed
1.	Login and Logout	Passed
2.	Profile Creation and Editing	Passed
3.	Expenses – Add, Edit and Delete	Passed
4.	Expense Analysis Module	Passed
5.	Savings – Add, Edit and Delete	Passed
6.	Savings Analysis Module	Passed
7.	Reminders – Add, Edit and Delete	Passed
8.	Loan Tracker – Add, Edit and Delete	Failed to Deploy

8.2 User Acceptance Testing

S.No	Test Cases	Yes/ No
1.	Keyword driven	Yes
2.	Responds in manually drafted rules	Yes
3.	Manages multiple users	Yes
4.	Conversational Paradigm	Yes
3.	Learns from real interactions	Yes
4.	Training via historical data	No
5.	Has decision-making skills	No

9. RESULTS

9.1 Performance Metrics

Cloud performance monitoring and testing tools help organizations gain visibility into their cloud environments, using specific metrics and techniques to assess performance. Efficient cloud performance is critical for maintaining business continuity and ensuring all relevant parties gain access to cloud services. This is true for basic cloud usage of public clouds and complex hybrid clouds and multi-cloud architectures. Cloud performance metrics enable you to effectively monitor your cloud resources, to ensure all components communicate seamlessly. Typically, cloud performance metrics measure input/output operations per second (IOPS), filesystem performance, caching, and autoscaling.

S. No	Test Cases	Time
1.	Error rates	1 second
2.	Response times	3 seconds
3.	Request rates	1.14 seconds
4.	Customer experience	Good

10. ADVANTAGES & DISADVANTAGES

10.1 Advantages

- Available 24/7 across the globe
- Easy tracking of savings and expenses
- User friendly dashboards
- Understandable charts and graphs
- Updated to the latest details
- Easy to setup and communicate

10.2 Disadvantages

- Direct connection with bank account could not be achieved
- Loan Tracker page could not be deployed
- Multiple profiles not available

11. CONCLUSION

This project focuses on tracking the personal finances of a user. It is implemented in a user-friendly and accessible manner to achieve all the financial asset maintenance goals of an individual. The new system has overcome most of the limitations of the existing system and works according to the design specification given. The project what we have developed is work more efficient than the other income and expense tracker. The project avoids the manual calculation for avoiding calculating the income and expense per month. The modules are developed with efficient and also in an attractive manner. The developed systems dispense the problem and meet the needs of by providing reliable and comprehensive information. All the requirements projected by the user have been met by the system.

The newly developed system consumes less processing time, and all the details are updated and processed immediately. Since the screen provides online help messages and is very user-friendly, any user will get familiarized with its usage. Module s are designed to be highly flexible so that any failure requirements can be easily added to the modules without facing many problems. The best organizations have a way of tracking and handling these reimbursements. This project also focuses on giving set reminders for users who wish to be up to date in their payments. This project gives aesthetic charts and analysis for the user's financial records.

12. FUTURE SCOPE

The future of project lies entirely on how the customers get benefitted from the interaction and the interface. We would have to make improvements in the application to make it more user-friendly. The following areas could have a serious impact on our scope:

i. Support for multiple accounts

ii. Direct Connection with bank

iii. Tips and Recommendations on where to save money.

I. APPENDIX

1. SOURCE CODE

a) app.py

```
import os
from flask import *
from database import *
from models import *
from random import *
from flask_mail import Mail, Message
from datetime import *
from time import *
database = Database()

app = Flask(_name_)
app.secret_key = "FlaskNotFoundError"
app.config['MAIL_SERVER'] = 'smtp.sendgrid.net'
app.config['MAIL_PORT'] = 587
app.config['MAIL_USE_TLS'] = True
app.config['MAIL_USERNAME'] = 'apikey'
app.config['MAIL_PASSWORD'] = credentials.SENDGRID_API_KEY
app.config['MAIL_DEFAULT_SENDER'] = credentials.MAIL_DEFAULT_SENDER
mail = Mail(app)

#Index
@app.route('/')
def index():
    return render_template('index.html')

#Signup
@app.route('/signup', methods = ['GET','POST'])
def signup():
    msg=None
    if request.method == 'POST':
        name = request.form['name']
        email = request.form['email']
        password = request.form['password']
        account = database.fetchUser(email)
        if account:
            flash("You are already a member, please login using your details")
            return redirect('signin')
        else:
            if database.insertSignUpUserData(email,password,name):
                msg = Message('eXpenso Registration', recipients=[email])
                msg.body = 'Thank you for registering with eXpenso! Happy Managing!!'
                msg.html = """<h1>Sucessfully Registered with eXpenso</h1>
                        <h3>Thank you for registering with eXpenso! Happy Managing!!</h3>
                    """
                mail.send(msg)
                flash("Registration successfull...")
                return redirect('signin')
            else:
                msg="Unable to Register!! Try again"
    return render_template('signup.html',msg=msg)
#Signin
@app.route('/signin',methods = ["GET","POST"])
def signin():
    invalidLogin = None
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']
        if database.fetchUser(email):
            fetchedPassword = database.fetchPassword(email)
            if fetchedPassword==password:
                session['email']=email
            return redirect('home')
          else:
            invalidLogin="Your Password is wrong!!"
        else:
          invalidLogin="You have not Registered yet!!"
    return render_template('signin.html',invalidLogin=invalidLogin)


@app.route('/signout')
def signOut():
    if 'email' in session:
        session.pop('email',None)
        return redirect('/')
    return redirect('/')


#Home
@app.route('/home')
def presentHome():
    if 'email' not in session:
        return redirect('/')
    email=session['email']
    user = database.fetchUser(email)
    totalExpenses = database.getTotalExpenseAmount(email)
    totalSavings = database.getDebitSavingsAmount(email)
    expenseFilter = "year"
    expenses = database.fetchExpensesPreview(email,5)
    monthExpenses=database.getExpensesThisYear(email)
    monthLabels=['January', 'February', 'March', 'April', 'May', 'June',
'July','August','September','October','November','December']
    monthExpenseList = [0]*12
    for expense in monthExpenses:
        monthExpenseList[int(expense["MONTH"])-1]=expense["AMOUNT"]
    totalLoanPaid = database.getTotalLoanPaid(email)
    totalLoanLeft = database.getTotalLoanLeft(email)
    reminders = database.readRemindersWithLimit(email)
    reminderList = []
    for reminder in reminders:
        days = (date(int(reminder["YEAR"]),int(reminder["MONTH"]),int(reminder["DATE"])) - date.today()).days
        label = "left"
        if days<0:
            continue
        elif days == 0:
            label = "today"
        name = reminder["REMINDERNAME"]
        reminderdate = reminder["DATE"]+"/"+reminder["MONTH"]
        description = reminder["DESCRIPTION"]
        reminderList.append([days,name,description,label,reminderdate])
    return render_template('home.html',user = user,expenseFilter = expenseFilter,totalExpenses = totalExpenses, totalSavings = totalSavings, expenses = expenses, monthLabels = monthLabels, monthExpenseList = monthExpenseList, totalLoanPaid = totalLoanPaid , totalLoanLeft = totalLoanLeft,reminders = reminderList)
```

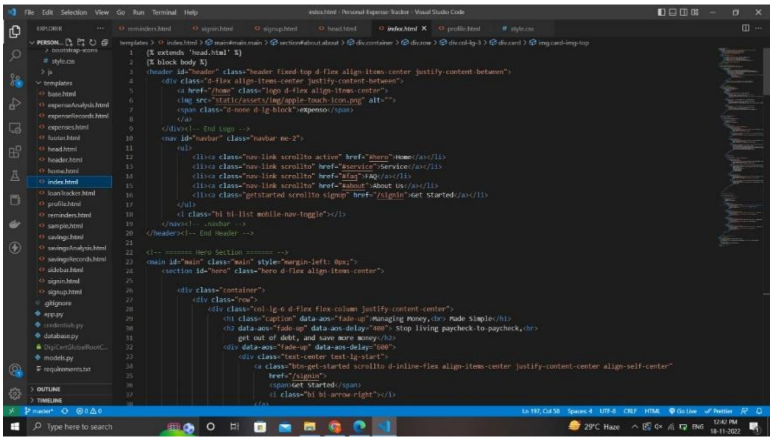
b) database.py

```
import credentials
import ibm_db
import ibm_boto3
from ibm_botocore.client import Config, ClientError
from datetime import *
from models import *
conn = ibm_db.connect("DATABASE="+credentials.DB2_DATABASE_NAME+";HOSTNAME="+credentials.DB2_HOST_NAME+";PORT="+credentials.DB2_PORT+";SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID="+credentials.DB2_UID+";PWD="+credentials.DB2_PWD+"",",")
cos = ibm_boto3.resource("s3",
    ibm_api_key_id=credentials.COS_API_KEY_ID,
    ibm_service_instance_id=credentials.COS_INSTANCE_CRN,
    config=Config(signature_version="oauth"),
    endpoint_url=credentials.COS_ENDPOINT
)
class Database:
    def _init_(self) -> None:
        pass
    def fetchUser(self,email):
        sql = "SELECT email,name,phone,country FROM user WHERE email=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            user =
User(account["EMAIL"],account["NAME"],account["PHONE"],account["COUNTRY"])
            return user
        return None
    def fetchPassword(self,email):
        try:
            sql = "SELECT password FROM user WHERE email = ?"
            stmt = ibm_db.prepare(conn, sql)
            ibm_db.bind_param(stmt,1,email)
            ibm_db.execute(stmt)
            account = ibm_db.fetch_assoc(stmt)
            if account:
                return account["PASSWORD"]
            else:
                return False
        except:
            return False
    def insertSignUpUserData(self,email,password,name):
        try:
            insert_sql = "INSERT INTO user(email,password,name) VALUES (?,?,?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, email)
            ibm_db.bind_param(prep_stmt, 2, password)
            ibm_db.bind_param(prep_stmt, 3, name)
            ibm_db.execute(prep_stmt)
        except:
            print("error")
            return False
        return True
    def updateUserData(self,email,name,country,phone):
        try:
            sql = "update user set name = ? , country = ?, phone = ? where email = ?;"
            stmt = ibm_db.prepare(conn, sql)
            ibm_db.bind_param(stmt,1,name)
            ibm_db.bind_param(stmt,2,country)
            ibm_db.bind_param(stmt,3,phone)
            ibm_db.bind_param(stmt,4,email)
            ibm_db.execute(stmt)
        except:
            return False
        return True
    def updatePassword(self,email,password):
        try:
            sql = "update user set password = ? where email = ?;"
            stmt = ibm_db.prepare(conn, sql)
            ibm_db.bind_param(stmt,1,password)
            ibm_db.bind_param(stmt,2,email)
            ibm_db.execute(stmt)
        except:
            return False
        return True
    def fetchExpensesPreview(self,email,limit=10):
        sql ="SELECT
expensename,date,month,year,expenses.description,savingsname,savingstype,expenses.amount
```

FROM expenses join savings on expenses.savingsid=savings.savingsid WHERE expenses.email=? order by expenseid desc limit ?;"

```
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.bind_param(stmt,2,limit)
    ibm_db.execute(stmt)
    expense = ibm_db.fetch_both(stmt)
    expenseList = []
    while expense != False:
        expenseList.append(expense)
        expense = ibm_db.fetch_both(stmt)
    return expenseList
```

c) FRONTEND PART:



```
{% extends 'head.html' %}
{% block body %}
<header id="header" class="header fixed-top d-flex align-items-center justify-content-between">
    <div class="d-flex align-items-center justify-content-between">
        <a href="/home" class="logo d-flex align-items-center">
        <img src="static/assets/img/apple-touch-icon.png" alt="">
        <span class="d-none d-lg-block">eXpenso</span>
        </a>
    </div><!-- End Logo -->
    <nav id="navbar" class="navbar me-2">
        <ul>
            <li><a class="nav-link scrollto active" href="#hero">Home</a></li>
            <li><a class="nav-link scrollto" href="#service">Service</a></li>
            <li><a class="nav-link scrollto" href="#faq">FAQ</a></li>
            <li><a class="nav-link scrollto" href="#about">About Us</a></li>
            <li><a class="getstarted scrollto signup" href="/signin">Get Started</a></li>
        </ul>
        <i class="bi bi-list mobile-nav-toggle"></i>
    </nav><!-- .navbar -->
</header><!-- End Header -->

<!-- ======= Hero Section ======= -->
<main id="main" class="main" style="margin-left: 0px;">
    <section id="hero" class="hero d-flex align-items-center">

        <div class="container">
            <div class="row">
                <div class="col-lg-6 d-flex flex-column justify-content-center">
                    <h1 class="caption" data-aos="fade-up">Managing Money,<br> Made Simple</h1>
                    <h2 data-aos="fade-up" data-aos-delay="400"> Stop living paycheck-to-paycheck,<br>
                        get out of debt, and save more money</h2>
                    <div data-aos="fade-up" data-aos-delay="600">
                        <div class="text-center text-lg-start">
                            <a class="btn-get-started scrollto d-inline-flex align-items-center justify-content-center align-self-center"
                                href="/signin">
                                <span>Get Started</span>
                                <i class="bi bi-arrow-right"></i>
                            </a>
```

2. **GitHub:** https://github.com/IBM-EPBL/IBM-Project-23937-1659933802

3. **Project Demo Link:**

   https://drive.google.com/drive/folders/1Zoef61TFQbDTdd6bNZ5_SCXbtcLTDzQo?usp=sharing

   Deployed Link: http://159.122.187.49:32671/