Personal Expense Tracker

Team ID	PNT2022TMID15222
Project Name	Personal Expense Tracker

TABLE OF CONTENTS

1. **INTRODUCTION**
   1.1 Project Overview
   1.2 Purpose

2. **LITERATURE SURVEY**
   2.1 Existing problem
   2.2 References
   2.3 Problem Statement Definition

3. **IDEATION & PROPOSED SOLUTION**
   3.1 Empathy Map Canvas
   3.2 Ideation & Brainstorming
   3.3 Proposed Solution
   3.4 Problem Solution fit

4. **REQUIREMENT ANALYSIS**
   4.1 Functional requirement
   4.2 Non-Functional requirements

5. **PROJECT DESIGN**
   5.1 Data Flow Diagrams
   5.2 Solution & Technical Architecture
   5.3 User Stories

6. **PROJECT PLANNING & SCHEDULING**
   6.1 Sprint Planning & Estimation
   6.2 Sprint Delivery Schedule
   6.3 Reports from JIRA

7. **CODING & SOLUTIONING**
   7.1 Feature 1
   7.2 Feature 2

7.3 Database Schema

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code

GitHub & Project Demo Link

1. INTRODUCTION :

1.1 Project Overview :

In simple words, personal finance entails all the financial decisions and activities that a Finance app makes your life easier by helping you to manage your finances efficiently. A personal finance app will not only help you with budgeting and accounting but also give you helpful insights about money management.

Personal finance applications will ask users to add their expenses and based on their expenses wallet balance will be updated which will be visible to the user. Also, users can get an analysis of their expenditure in graphical forms. They have an option to set a limit for the amount to be used for that particular month if the limit is exceeded the user will be notified with an email alert.

1.2 Purpose :

An expense tracker is a software or application that helps to keep an accurate record of your money inflow and outflow. Many people in India live on a fixed income, and they find that towards the end of the month they don't have sufficient money to meet their needs.

2. LITERATURE SURVEY

2.1 Existing problem :

- No alter when the limit exceeded
- Poor Authentication
- No particular domain
- Poor visual representation

- No direct link with bank reports
- NO individual report on specific domain

2.2 References :

Hrithik Gupta, Anant Prakash Singh, Navneet Kumar and J. Angelin Blessy "Expense Tracker : A Smart Approach to Track Everyday Expense"

*S. Chandini, T. Poojitha, D. Ranjith, V.J. Mohammed Akram, M.S. Vani, V. Rajyalakshmi "*Online Income and Expense Tracker"

*Sali, Moussa; Abbo, Abdel Salam "*Personal Expense Assistant Management : An Android Based Application"

A Velmurugan1, J Albert Mayan1, P Niranjana2 and Richard Francis2 "Expense Manager Application"

*P. Thanapal , Mohammed Yaseen Patel, T.P. Lokesh Raj and J. Satheesh Kumar "*Income and Expense Tracker"

*Angad Manchanda "*EXPENSE TRACKER MOBILE APPLICATION"

*NUURA NAJATI BINTI MUSTAFA "*DAILY EXPENSE TRACKER MOBILE APPLICATION"

*Asthha Wahal,Muskan Agarwal "*Cloud based Expense Tracker"

2.3 Problem Statement Definition :

In simple words, personal finance entails all the financial decisions and activities that a Finance app makes your life easier by helping you to manage your finances efficiently. A personal finance app will not only help you with budgeting and accounting but also give you helpful insights about money management. Personal finance applications will ask users to add their expenses and based on their expenses wallet balance will be updated which will be visible to the user. Also, users can get an analysis of their expenditure in graphical forms. They have an option to set a limit for the amount to be used for that particular month if the limit is exceeded the user will be notified with an email alert.

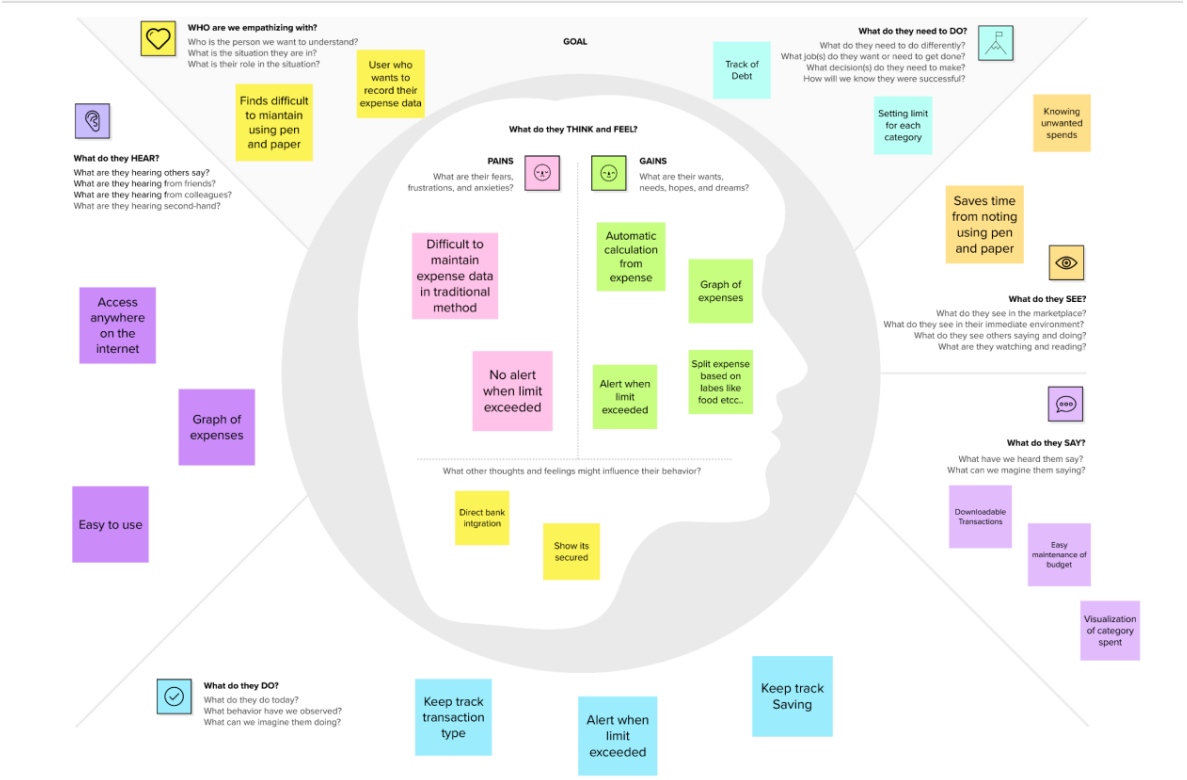
3. IDEATION & PROPOSED SOLUTION :

3.1 Empathy Map Canvas :

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

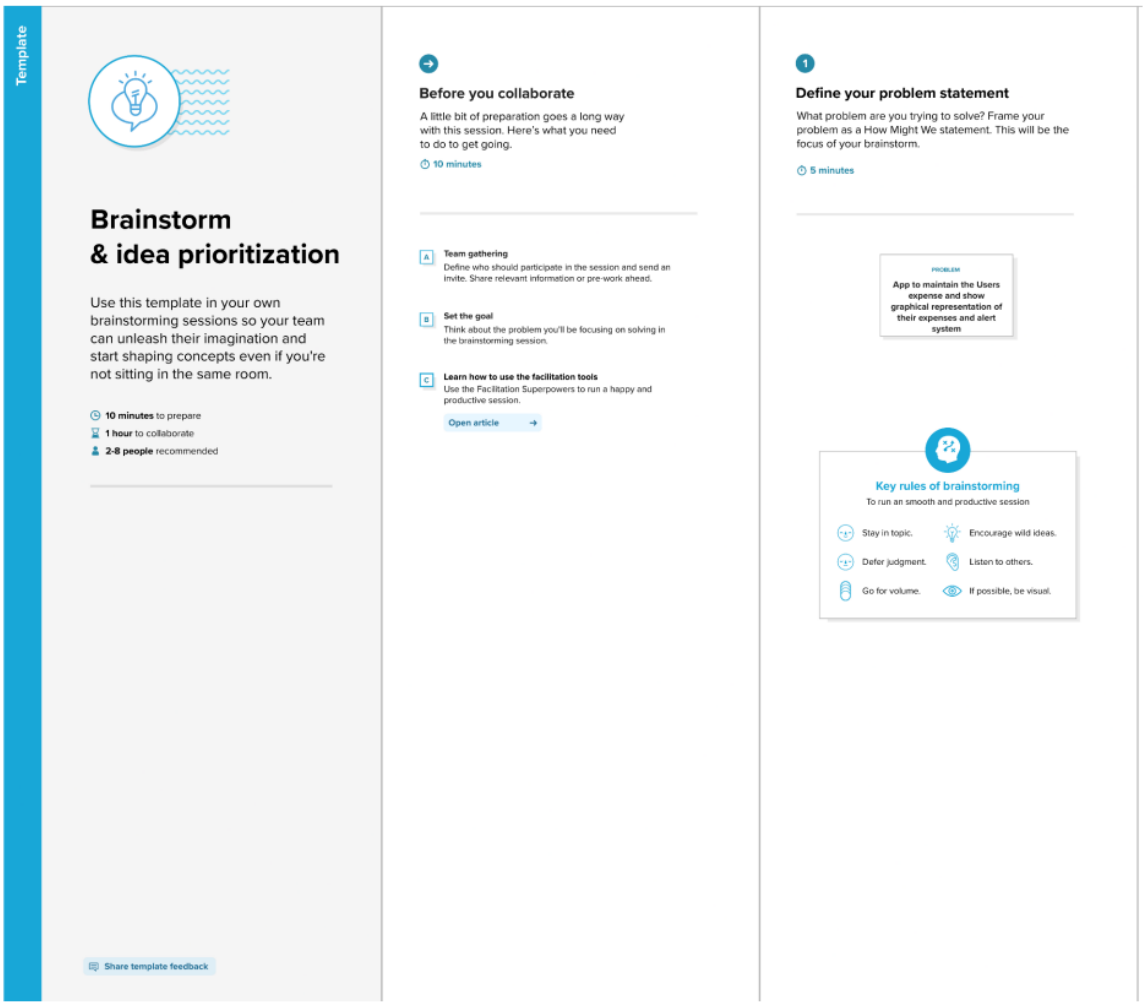
It is a useful tool to helps teams better understand their users.

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

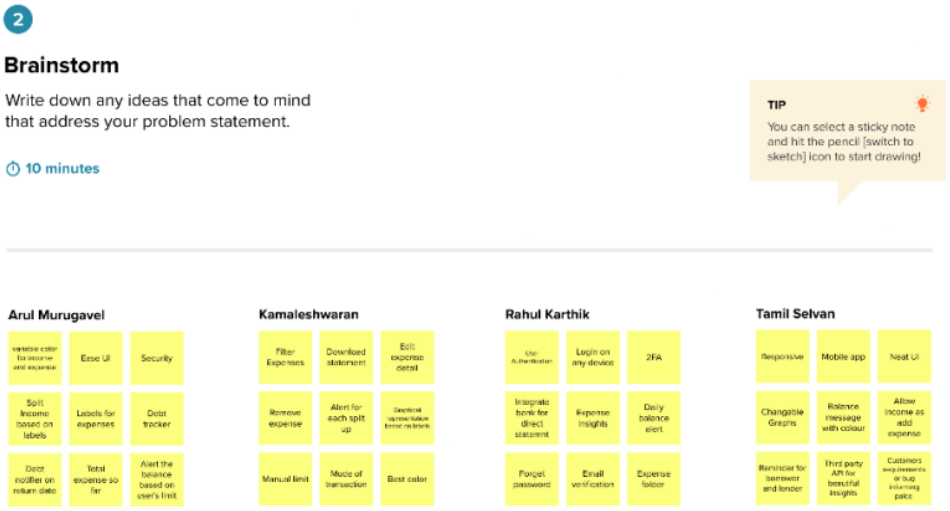


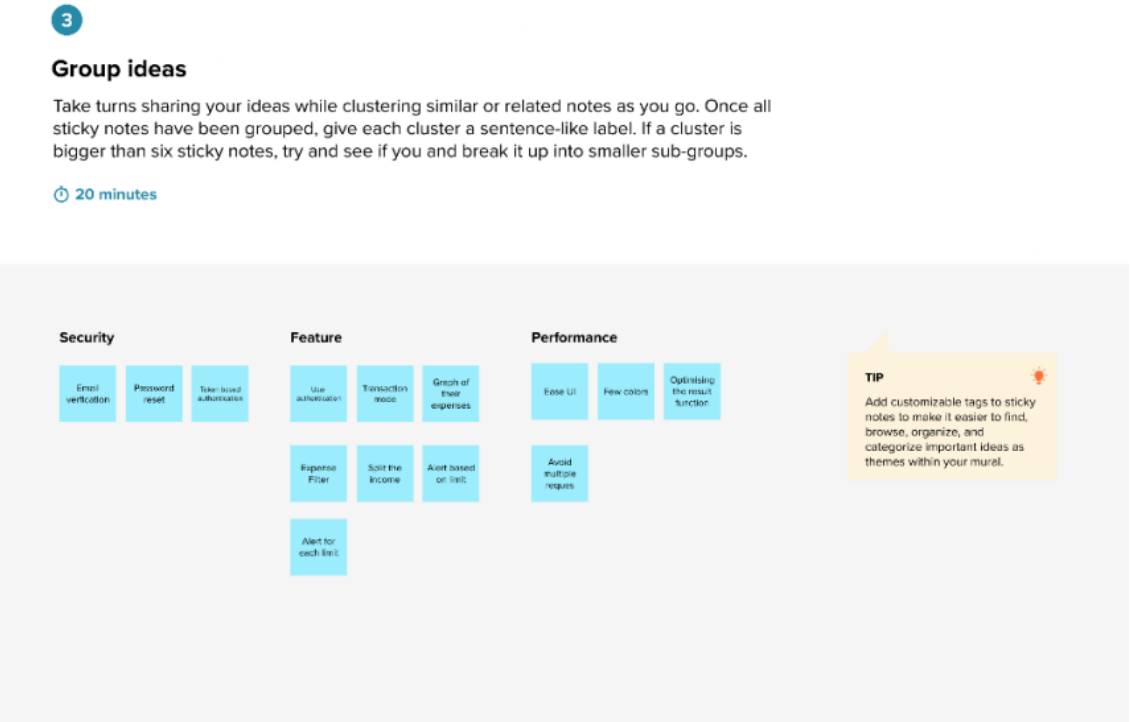
3.2 Ideation & Brainstorming

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



Step-2: Brainstorm, Idea Listing and Grouping :





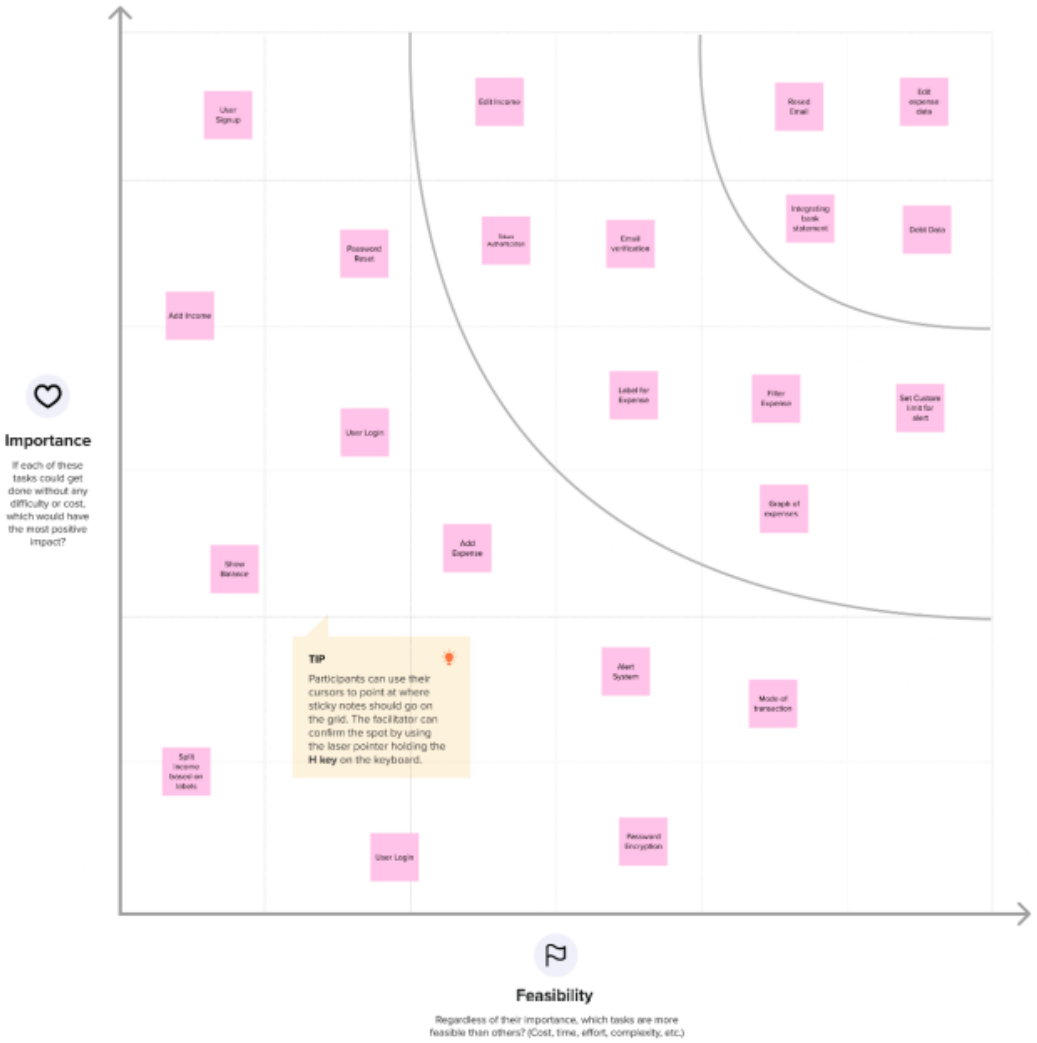
Step-3: Idea Prioritization :

4

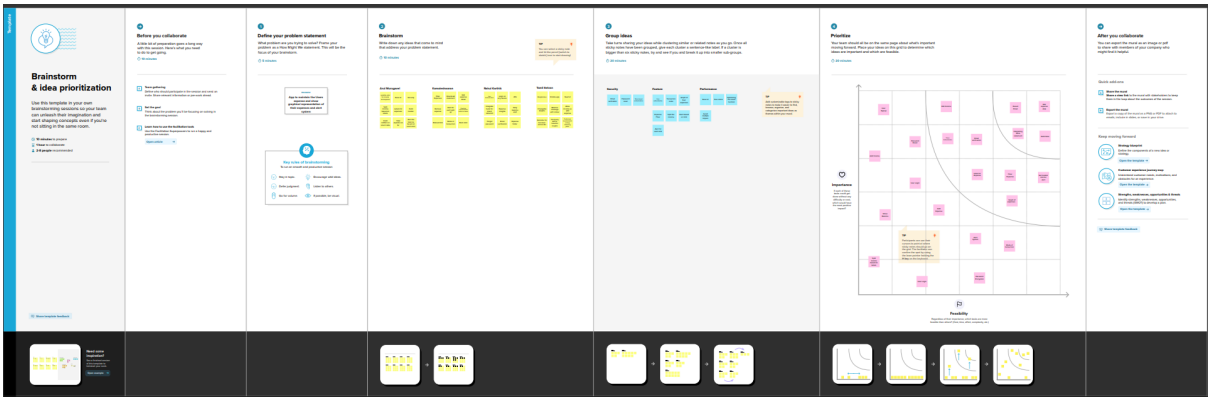
Prioritize

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



Entire pitch :

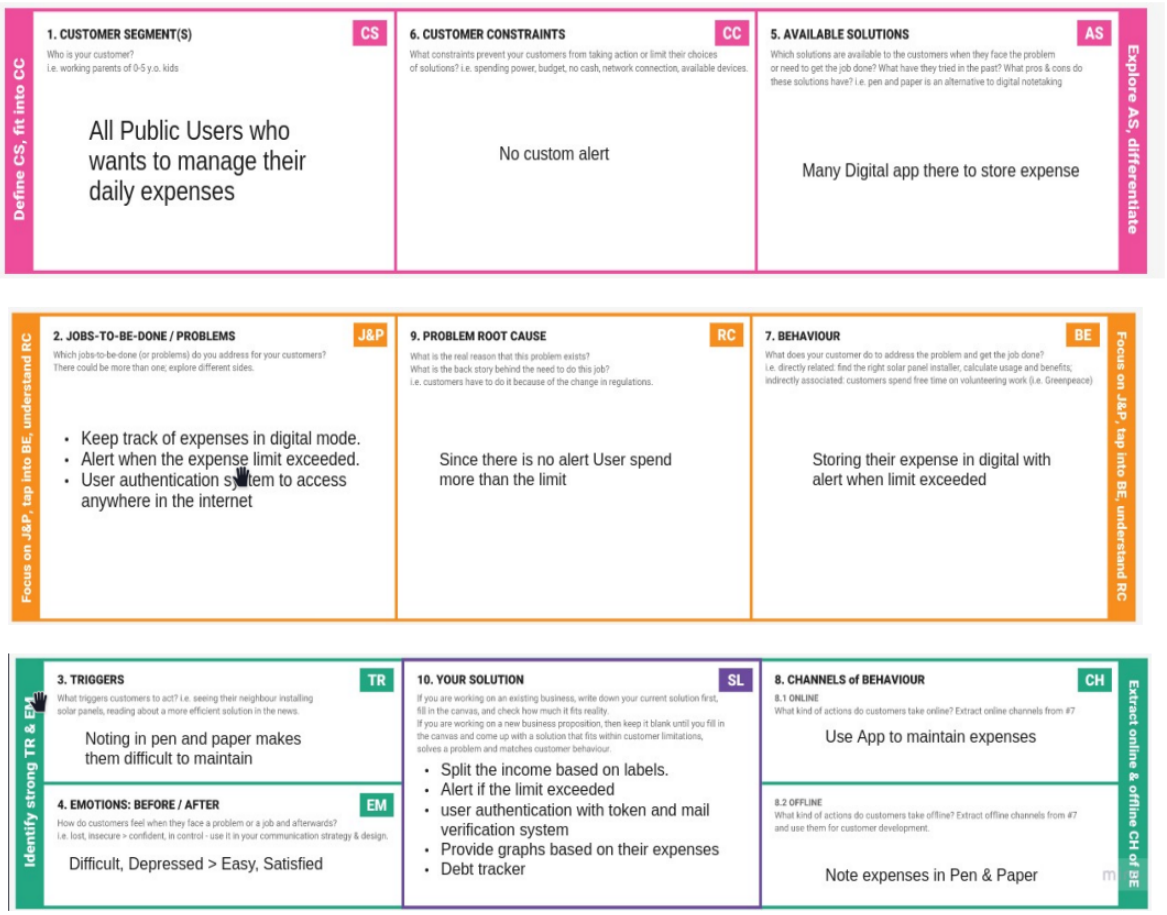


3.3 Proposed Solution :

S.No.	Parameter	Description
.	Problem Statement (Problem to be solved)	System to maintain day to day expenses. Keep track of daily expenses and income. Provide helpful insights based on expenses.
.	Idea / Solution description	• Split the income based on labels. • Alert if the limit exceeded • user authentication with token and mail verification system • Provide graphs based on their expenses • Debt tracker
.	Novelty / Uniqueness	• Keeping track of both borrower and lender • Set custom limit for alert • Split based on labels like for food, entertainment
.	Social Impact / Customer Satisfaction	Providing expense insights will help to reduce unnecessary expenses. Graph of their expenses helps to analyze their expenses. Alert system help to notify when the amount is exceeded
.	Business Model (Revenue Model)	Helps to maintain the record of transactions with secure and cloud storage so accessible anywhere in the internet.
.	Scalability of the Solution	• Provide optimized results. • Ease and Neat UI • Securing the user data

		• Authentication system

3.4 Problem Solution fit :



4. REQUIREMENT ANALYSIS :

4.1 Functional requirement :

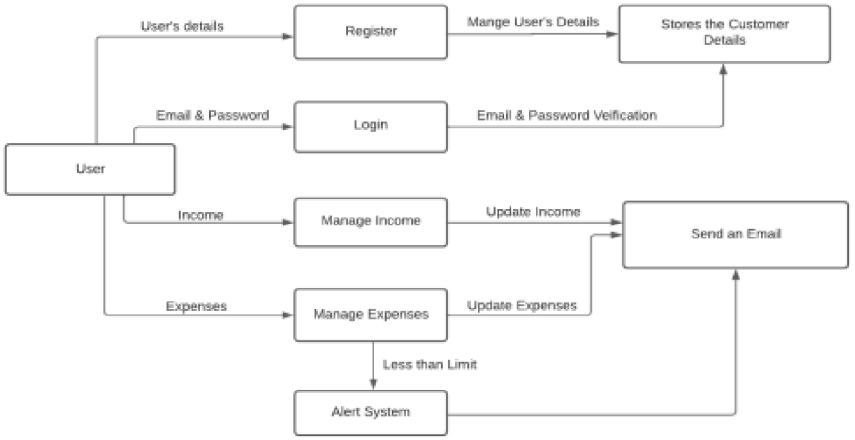
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Login	Enter the valid username and password
FR-4	Calendar	Personal expense tracker application shall allow user to add the data to their expenses.
FR-5	Expense Tracker	This application should graphically represent the expense in the form of report.
FR-6	Alert System	Send Alert email notifications to user if it reaches below the specific value
FR-7	Category	This application shall allow users to add categories of their expenses.

4.2 Non-Functional requirements :

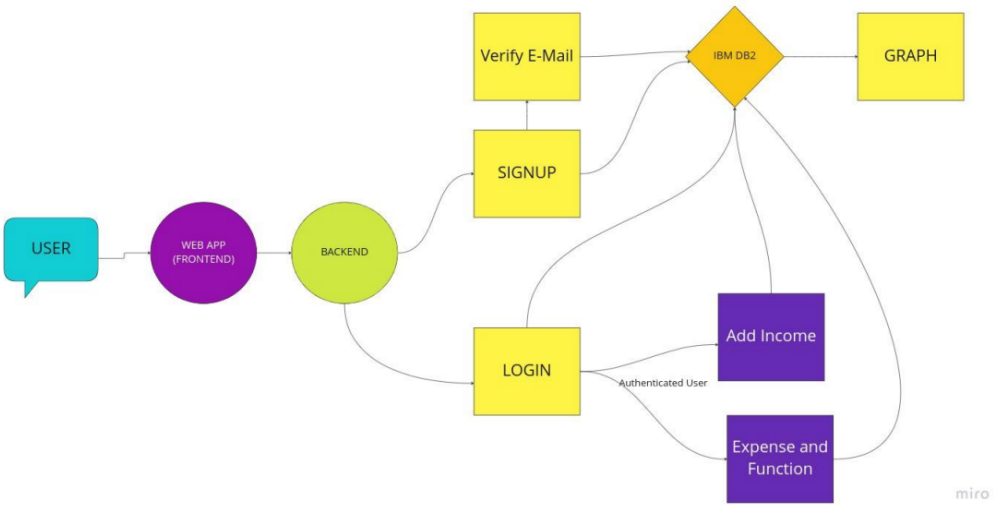
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Helps to keep an accurate record of your money inflow and outflow.
NFR-2	**Security**	Budget tracking apps are considered very safe from cybercriminals.
NFR-3	**Reliability**	Each data record is stored on a well-built efficient database schema. There is no risk of data loss
NFR-4	**Performance**	The types of expense are categories along with an option. Throughput of the system is increased due to light weight database support.
NFR-5	**Availability**	It is available all the time. No time constraint
NFR-6	**Scalability**	The ability to appropriately handle increasing demands.

5. PROJECT DESIGN :

5.1 Data Flow Diagrams :



5.2 Solution & Technical Architecture :



Proposed Solutions :

S.No.	Parameter	Description
1	Problem Statement (Problem to be solved)	System to maintain day to day expenses. Keep track of daily expenses and income. Provide helpful insights based on expenses.
2	Idea / Solution description	● Split the income based on labels. ● Alert if the limit exceeded ● user authentication with token and mail verification system ● Provide graphs based on their expenses ● Debt tracker
3	Novelty / Uniqueness	● Keeping track of both borrower and lender ● Set custom limit for alert ● Split based on labels like for food, entertainment
4	Social Impact / Customer Satisfaction	Providing expense insights will help to reduce unnecessary expenses. Graph of their expenses helps to analyze their expenses. Alert system help to notify when the amount is exceeded
5	Business Model (Revenue Model)	Helps to maintain the record of transactions with secure and cloud storage so accessible anywhere in the internet.
6	Scalability of the Solution	● Provide optimized results. ● Ease and Neat UI ● Securing the user data ● Authentication system

5.3 User Stories :

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (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	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
	Login	USN-3	As a user, I can log into the application by entering email & password	I can access all the feature	High	Sprint-1
	Dashboard & Logout	USN-4	As a user, once I logged in I can access all the features of the web app and Logout once I completed all the work.	I can access all the feature and shut it down once logged out	High	Sprint-1
		USN-5	Once logged In, Keep me logged for few hours to avoid repeated login if the page is refreshed	I can remain logged in if the page refreshes	Medium	Sprint-1
	Expense	USN-6	Add total income for the month and Allow for edit option	I can add Total income and edit if it changes a lot	High	Sprint-2
		USN-7	Split the total income based on usage like entertainment, food, shopping etc.	I can split it as per the usage	Low	Sprint-2
		USN-8	Add the day to day expense.	I can add the income and expense data along with label	High	Sprint-2
		USN-9	Display the user added expense	I can see the past expense along with pending balance	High	Sprint-2
		USN-10	Filter the expense data based on criteria	I can filter the past data with custom data and type	Medium	Sprint-3

	Show charts	USN-11	As a user I can display it in graphs	I can see the bar, pie chart etc	Low	Sprint-3
	Show alerts	USN-12	As a user I create custom alert for the balance	I should receive mail once the level reached below	High	Sprint-3
	Deployment	USN-13	As a user I should able to access it anywhere in the net	I should access the site from any device	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

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.	3	High	Rahul Karthik Kamaleshwaran
Sprint-1		USN-2	As a user, I will receive confirmation	3	High	Arul Murugavel

			email once I have registered for the application			
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	5	High	Tamil Selvan
Sprint-1	Dashboard & Logout	USN-4	As a user, once I logged in I can access all the features of the web app and Logout once I completed all the work.	5	High	Kamaleshwaran
Sprint-1		USN-5	Once logged In, Keep me logged for few hours to avoid repeated login if the page is refreshed	4	Mediu m	Arul Murugavel
Sprint-2	Expense	USN-6	Add total income for the month and Allow for edit option	6	High	Kamaleshwaran
Sprint-2		USN-7	Split the total income based on usage like entertainment , food, shopping etc.	2	Low	Rahul Karthik
Sprint-2		USN-8	Add the day to day expense.	6	High	Arul Murugavel
Sprint-2		USN-9	Display the user added expense	6	High	Tamil Selvan
Sprint-3		USN-10	Filter the expense data	6	Mediu m	Split Between 4

			based on criteria			
Sprint-3	Charts	USN-11	As a user I can display it in graphs	4	Low	Arul Murugavel Kamalesh
Sprint-3	Alerts	USN-12	As a user I create custom alert for the balance	10	High	Tamil Selvan Rahul Karthik
Sprint-4	Deployment	USN-13	As a user I should able to access it anywhere in the net	20	High	Split between 4

6.2 Sprint Delivery Schedule :

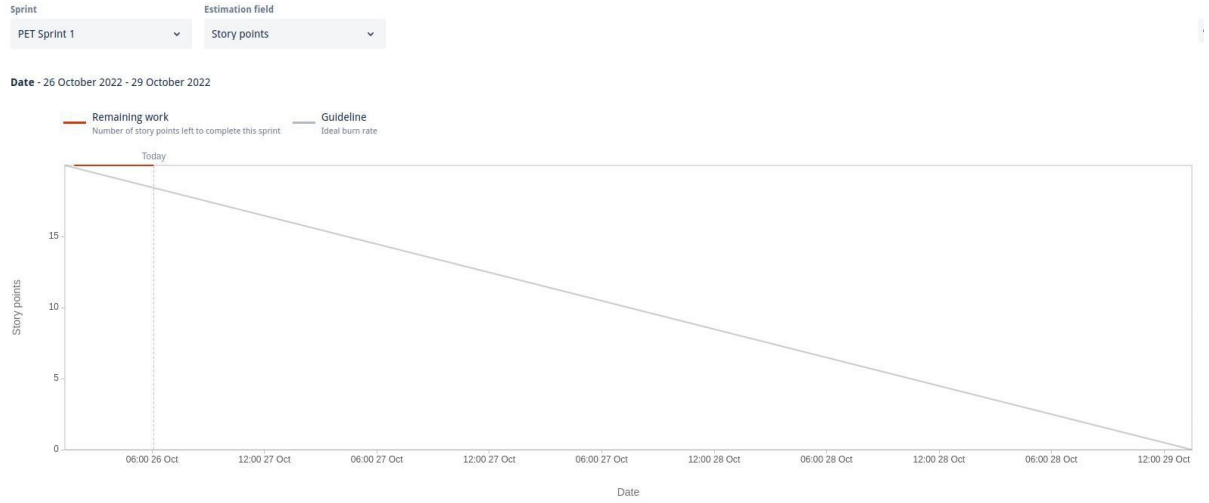
Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3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

6.3 Reports from JIRA :

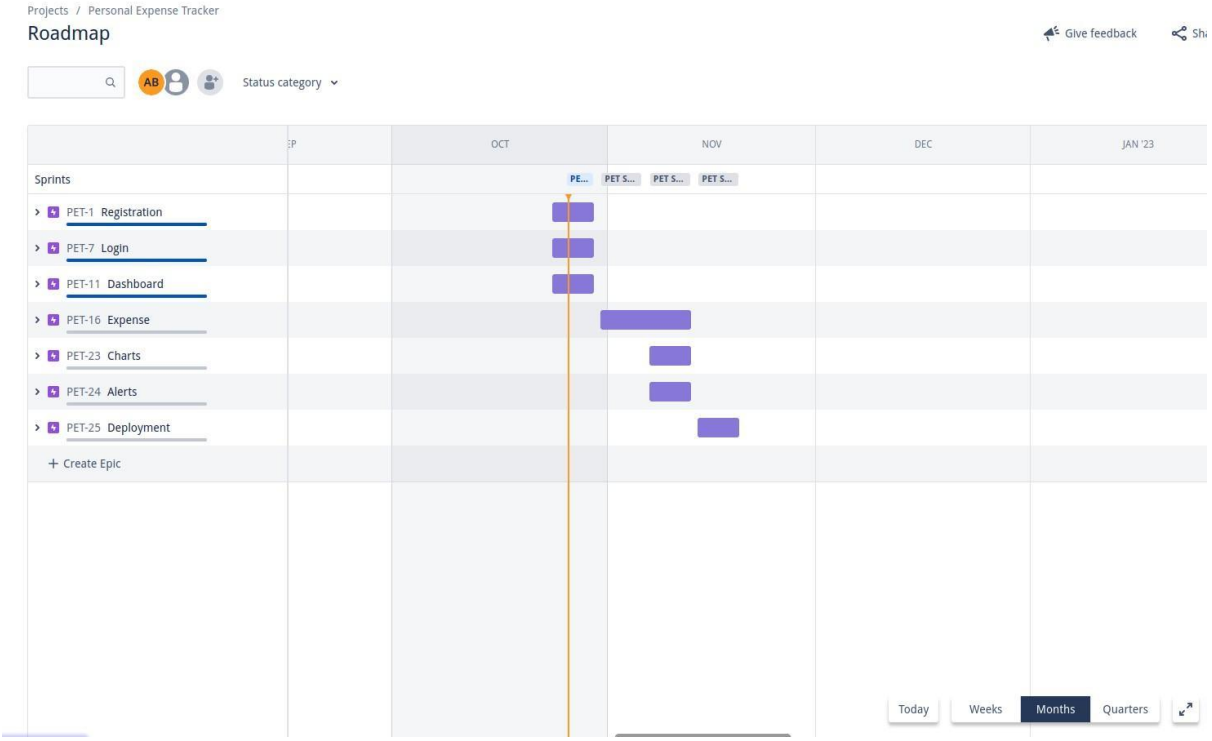
Project Planning :

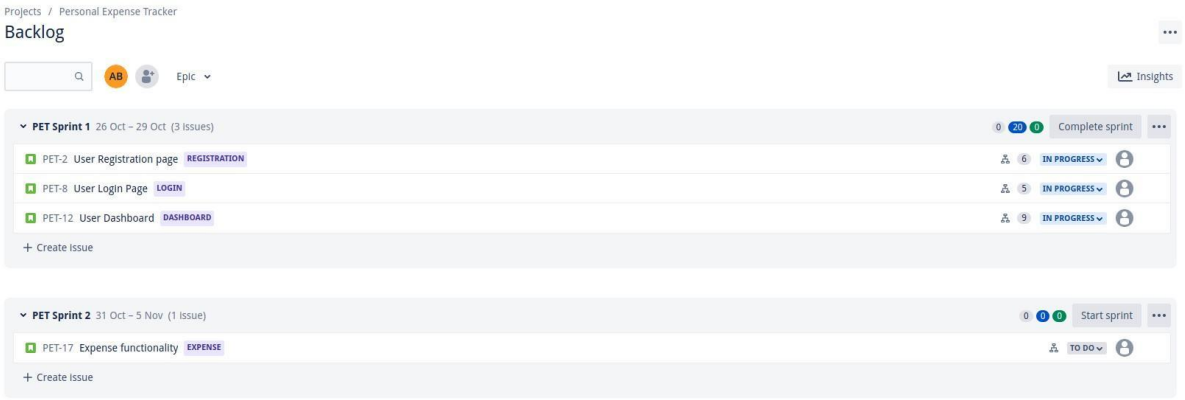
Burndown Chart:

A burndown chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



Project Planning:





7. CODING & SOLUTIONING

7.1 Feature 1

Checking user has valid jwt token to provided authenticated usage

```python
def create_jwt_token(data):
    data["exp"] = generate_timestamp(1, True)
    token = jwt.encode(data, getenv('JWT_SECRET_KEY'), algorithm="HS256")
    return token


def validate_jwt_token(token):
    try:
        decoded_content = jwt.decode(token, getenv('JWT_SECRET_KEY'), algorithms=["HS256"])
    except jwt.ExpiredSignatureError:
        return {"is_valid": False, "message": "Token expired"}
    except jwt.InvalidSignatureError:
        return {"is_valid": False, "message": "Invalid Token"}

    return {"is_valid": True, "payload": decoded_content}
```

```python
from functools import wraps
from flask import request, after_this_request

def token_required(f):
    @wraps(f)
    def decorated(*args, **kwargs):
        auth_token = request.cookies.get("auth_token")
        if(not auth_token):
            return ({"message": "No Token"}, 400)
        res = validate_jwt_token(auth_token)
        if(not res["is_valid"]):
            @after_this_request
            def set_cookie(response):
                response.set_cookie('auth_token', value="", path="/", secure="None", samesite="None", httponly=True)
                return response

            return ({"message": res["message"]}, 404)

        return f(res["payload"], *args, **kwargs)
    return decorated
```

7.2 Feature 2

Sending Mail for Confirmation and Usage alert

```python
from sendgrid.helpers.mail import Mail
from ..config.mail_config import get_mail_config
from os import getenv

def send_mail(email, data, templateID):
    try:
        sg = get_mail_config()

        FROM_EMAIL = getenv("FROM_MAIL")
        TO_EMAIL = [(email, 'User')]
        message = Mail(
            from_email=FROM_EMAIL,
            to_emails=TO_EMAIL)
        message.dynamic_template_data = data
        message.template_id = templateID

        sg.send(message)
        return True

    except:
        return False
```

```python
from .mail import send_mail
def send_confirmation_token(email):
    token = generate_confirmation_token(email)

    confirm_url =f"{getenv('BASE_URL')}/confirm.html?token={token}"
    data = {
        "confirm_url": confirm_url
    }

    to_email = email
    templateID =
    res = send_mail(to_email, data, templateID)

    return res
```

7.3 Database Schema

User Schema:

```sql
CREATE TABLE user (
   id INT NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY (START WITH
1 INCREMENT BY 1),
   email VARCHAR(50) NOT NULL UNIQUE,
   password_hash VARCHAR(64)  NOT NULL,
   verified BOOLEAN NOT NULL DEFAULT FALSE,
   next_resend bigint,
   total_amount bigint  DEFAULT 0,
   timestamp bigint DEFAULT 0,
   alert bigint default 0,
   is_send boolean default false
);
```

Expense Table:

```sql
CREATE TABLE expense (
   id INT NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY (START WITH
1 INCREMENT BY 1),
   user_id INT NOT NULL,
   amount DECIMAL NOT NULL,
   is_income BOOLEAN NOT NULL,
   label VARCHAR(50) NOT NULL DEFAULT 'OTHER',
   timestamp BIGINT NOT NULL ,
   constraint fk_user_id
           foreign key (user_id)
           references user(id)
```

```
);
```

Split Table:

```
CREATE TABLE split_income (
    id INT NOT NULL PRIMARY KEY GENERATED ALWAYS AS IDENTITY (START WITH
1 INCREMENT BY 1),
   user_id INT NOT NULL,
   amount DECIMAL NOT NULL,
   label VARCHAR(50) NOT NULL DEFAULT 'OTHER',
   constraint fk_user_id
           foreign key (user_id)
           references user(id)
);
```

8. Testing

8.1 Test Case

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)
HomePage_TC_001	Functional	Home page - Split Income	Verify user is able to Split income		1. Go to the site 2. Click on split income button 3. Provide necessary data 4. Click update btn		Application should update split income data	Working as e	Pass		N
HomePage_TC_001	Functional	Home page - Add Expense	Verify user is able to add expense		1. Go to the site 2. Click on add expense button 3. Provide necessary data 4. Click update btn		Application should update a expense data	Working as e	Pass		N
HomePage_TC_001	Functional	Home page - Display Expense	Verify user is able to see the past expense		1. Go to the site 2. Past expense is shown under past expense tab		Application shows the past expense data	Working as e	Pass		
HomePage_TC_001	Functional	Home page - Filter income	Verify user is able to filter income		1. Go to the site 2. Under past expense tab 3. Provide necessary filter details 4. Click update btn		Application should update past expense with filter data	Working as e	Pass		N
HomePage_TC_001	Functional	Home page - Usage alert	Verify user is able to add custom usage alert		1. Go to the site 2. Click on set alert button 3. Provide necessary data 4. Click update btn		Application should update alert	Working as e	Pass		N
HomePage_TC_001	Functional	Home page - Usage alert	Verify user is receiving usage alert mail		1. Go to the site 2. Update balance that is below the alert level 3. Go to your mail check the mail is received		Application should send usage alert	Working as e	Pass		N
HomePage_TC_001	Functional	Home page - Chart	Verify user is able to see chart from their past expense data		1. Go to the site 2. Scroll to bottom to Graphical tab		Application should create chart from the past expense data	Working as e	Pass		N

8.2 User Acceptance Testing

1. 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	Subtotal
By Design	0	0	0	0	0
Duplicate	0	0	2	0	2
External	0	0	0	0	0

Fixed	0	0	0	3	0
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	0	0
Won't Fix	0	0	3	0	3
Totals	0	0	5	4	6

2. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Login	4	0	0	4
Income	1	0	0	1
Split Income	1	0	0	1
Add expense	1	0	0	1
Display expense	1	0	0	1
Filter Income	1	0	0	1
usage alert	2	0	0	2
chart	1	0	0	1

9. Results

9.1 Performance Metrics

S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score
1	Personal Expense Tracke	New	Low	No Changes	Moderate		>5 to 10%	ORANGE

NFT - Detailed Test Plan				
S.No	Project Overview	NFT Test approach	ssumptions/Dependencies/Risk	Approvals/SignOff
1	Personal Expense Tracker	Manual	Device with internet	

End Of Test Report								
S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff
1	Personal Expense Tracker	Manual		As expected, working fine		Laptop or Desktop gives better use	None	

Team ID	PNT2022TMID15222
Project Name	Personal Expense Tracker

10. ADVANTAGES & DISADVANTAGES

- It Helps You Stick to Your Budget
- Tracking Your Expenses Can Reveal Spending Issues
- It Helps You Meet Your Financial Objectives
- Effective expense tracking and reporting to avoid conflict

11. CONCLUSION

Thus, we have developed such an application which helps the users to reduce their effort of handling daily expenses. That the application will have various components of updating and viewing users' expenditures. As part of research, we considered adding certain components to the application to make it more useful to the user. Some of the extra Components are like enabling users to register to the application using an existing email or social network account, it will synchronize the users profile data to the application.

12. FUTURE SCOPE

The Future Enhancements of the application can be allowed to support in all the upcoming android versions. History can be set to view all the details in the app even if the particular data is deleted from the database. Statistics could be prepared based on the Income, Expense details of the user. Sharing files via Bluetooth, WhatsApp can be allowed. Printing the details of the particular income or expense details can be made. Some of the extra components are like enabling users to register to the application using an existing email or social network account, it will synchronize the users profile data to the application**.**

13. Appendix

- **Source Code - GitHub**
- **App Link**
- **Demo Video Link**