## A PROJECT REPORT ON

Personal Expense Tracker Application

Domain : Cloud App Development

ID : PNT2022TMID20929

College Name : Sri Sai Ram Institute of Technology

S. Aiswarya R. Danujaa S. Sharmitha Sneha Ganesh

(412419106003) (412419106013) (412419106077) (412419106083)

Department of Department of Department of Department of

Electronics and Electronics and Electronics and Electronics and

Communication Communication Communication Communication

Engineering Engineering Engineering Engineering

**TABLE OF CONTENTS**

1. INTRODUCTION……………………………………………………………………………...... 3

1.1. Project Overview......................................................................................................... 3

1.2. Purpose....................................................................................................................... 3

2. LITERATURE SURVEY……………………………………………………………………........4

2.1. Existing problem.......................................................................................................... 4

2.2. References.................................................................................................................. 5

2.3. Problem Statement Definition...................................................................................... 5

3. IDEATION & PROPOSED SOLUTION…………………………………………………….......6

3.1. Empathy Map Canvas................................................................................................. 6

3.2. Ideation & Brainstorming........................................................................................ …..7

3.3. Proposed Solution........................................................................................................9

3.4. Problem Solution fit..................................................................................................... 9

4. REQUIREMENT ANALYSIS……………………………………………………………….......10

4.1. Functional requirement...............................................................................................10

4.2. Non-Functional requirements..................................................................................... 11

5. PROJECT DESIGN…………………………………………………………………….............12

5.1. Data Flow Diagrams....................................................................................................12

5.2. Solution & Technical Architecture................................................................................13

5.3. User Stories.................................................................................................................14

6. PROJECT PLANNING & SCHEDULING………………………………………………..........15

6.1. Sprint Planning & Estimation.......................................................................................15

6.2. Sprint Delivery Schedule.............................................................................................18

6.3. Reports from JIRA.......................................................................................................18

7. CODING & SOLUTIONING………………………………………………………………...... ..19

7.1. Feature 1..................................................................................................................... 19

8. TESTING……………………………………………………………………..............................20

8.1. Test Cases.................................................................................................................. 20

9. RESULTS……………………………………………………………………............................ 23

9.1. Performance Metrics.................................................................................................. 23

10. ADVANTAGES & DISADVANTAGES……………………………………………………..... 24

11. CONCLUSION…………………………………………………………………….................. 25

12. FUTURE SCOPE……………………………………………………………………............. 26

13. APPENDIX……………………………………………………………………....................... 27

13.1. Source Code............................................................................................................ 27

13.2. GitHub & Project Demo Link.................................................................................... 31

## **CHAPTER 1**

## **1.INTRODUCTION**

A Personal Expense Tracker Application is a particular form of digital diary that aids in keeping track of all of our cash transitions and moreover offers daily, weekly, monthly, and yearly reports on all financial activities.

User receives alerts to keep track of income and expenses that can

system for tracking the application. All data is kept in offline mode for easy access at any time and from any location. The Daily Expense Tracker's user interface is incredibly straightforward and appealing, making it simple to grasp and the finest approach to record our financial data.

### **1.1. PROJECT OVERVIEW**

Simply put, personal finance includes all of the financial decisions and actions that a finance software facilitates by assisting you in effectively managing your finances. A personal finance software will not only assist you with accounting and budgeting, but it will also provide you with valuable advice on money management.

Users of personal finance applications will be prompted to enter their costs, after which their wallet balance will be updated and displayed to them. Users can also receive a graphical analysis of their expenses. They can choose to establish a cap on how much can be used in that month, and if the cap is surpassed, the user will receive an email alert.

### **1.2. PURPOSE**

When you keep track of your spending, you can make sure your money is being utilised wisely and you will know where it goes. You can learn why you're in debt and how you got there by keeping track of your spending. You can then use this information to create a debt relief plan that works for you.

You may plan for both short-term and long-term expenses by using a budget to make sure you're not spending more than you're earning. It's a simple, practical solution for folks with all types of income and expenses to maintain order in their finances.

## **CHAPTER 2**

## **2. LITERATURE SURVEY**

A literature review is a piece of academic writing that places the academic literature on a particular topic in perspective to show knowledge of and understanding of it. This chapter shows the different techniques that have been implemented.

### **2.1. EXISTING PROBLEM**

The expense tracker existing system does not offer the user portable device management level, is only used on desktop software, and is therefore impossible to update anywhere expenses are done and is unable to update the location of the expense details disrupting that the proposed system provides. The user's daily, weekly, and monthly spending must be maintained in Excel sheets and CSV files at the moment. The ability to conveniently keep track of one's everyday costs does not now have a fully comprehensive answer.To do this, one must maintain a journal in a diary or computer system, and all calculations must be made by the user, which might occasionally result in errors that cause losses. Due to imperfect data maintenance, the current system is not user friendly. The sole negative where the rest are absent from this endeavor is that there will be no reminder to stay a human on a specified date. This project won't have any information because it doesn't remind people to do anything each month, which has some drawbacks. However, it can be used to calculate income and expenses, so we suggest a new project to solve this issue.

### **2.2 REFERENCE**

[1] Expense Tracker ATIYA KAZI, PRAPHULLA S. KHERADE, RAJ S. VILANKAR, PARAG M. SAWANT May 2021

[2] Intelligent Online Budget Tracker Girish Bekaroo and Sameer Sunhaloo Proceedings of the 2007 Computer Science and 2007

[3] Online Income and Expense Tracker S. Chandini, T. Poojitha, D. Ranjith, V.J. Mohammed Akram, M.S. Vani, V. Rajyalakshmi Mar 2019

[4] Family Expense Manager Application Rajaprabha M N 2017

[5] A Novel Expense Tracker using Statistical Analysis Muskaan Sharma, Ayush Bansal, Dr. Raju Ranjan, Shivam Sethi June 2021

[6] Expense Tracker Hrithik Gupta, Anant Prakash Singh, Navneet Kumar and J. Angelin Blessy December 2020

[7] Expense Manager: An Expense Tracking Application using Image Processing Nupur Sawarkar, Pranay Yenagandula, Devang Shetye, Prof. Shruti Agrawal April 2022

[8] D2D Expense Tracker Application Anjali Kumar, Utkarsh Ra, Aman Kumar 2021

[9] Daily Expense Tracker Mobile Application Nuura Najati Binti Mustafa 2021

[10] Daily Expense Tracker Shivam Mehra, Prabhat Parashar 2021

### **2.3 PROBLEM STATEMENT DEFINITION**

In our daily life money is the most important portion and without it we cannot last one day on earth but if we keep on track all financial data then we can overcome this problem. Most of the people cannot track their expenses and income one way they face the money crisis and depression. This situation motivates us to make an android app to track all financial activities. Using the Personal Expense Tracker Application user can be tracking expenses day to day and making life tension free.

## 

## 

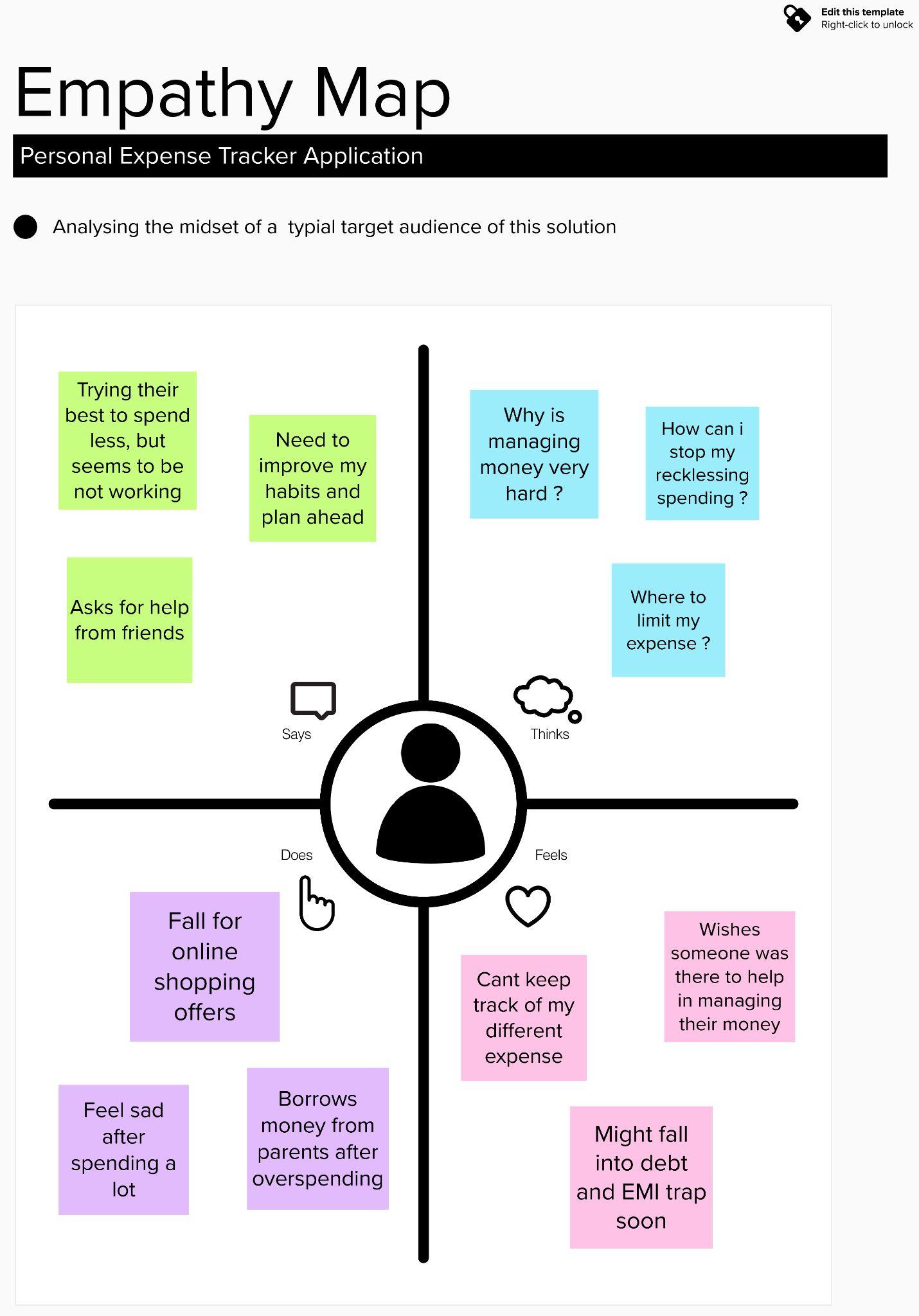
## 

## 

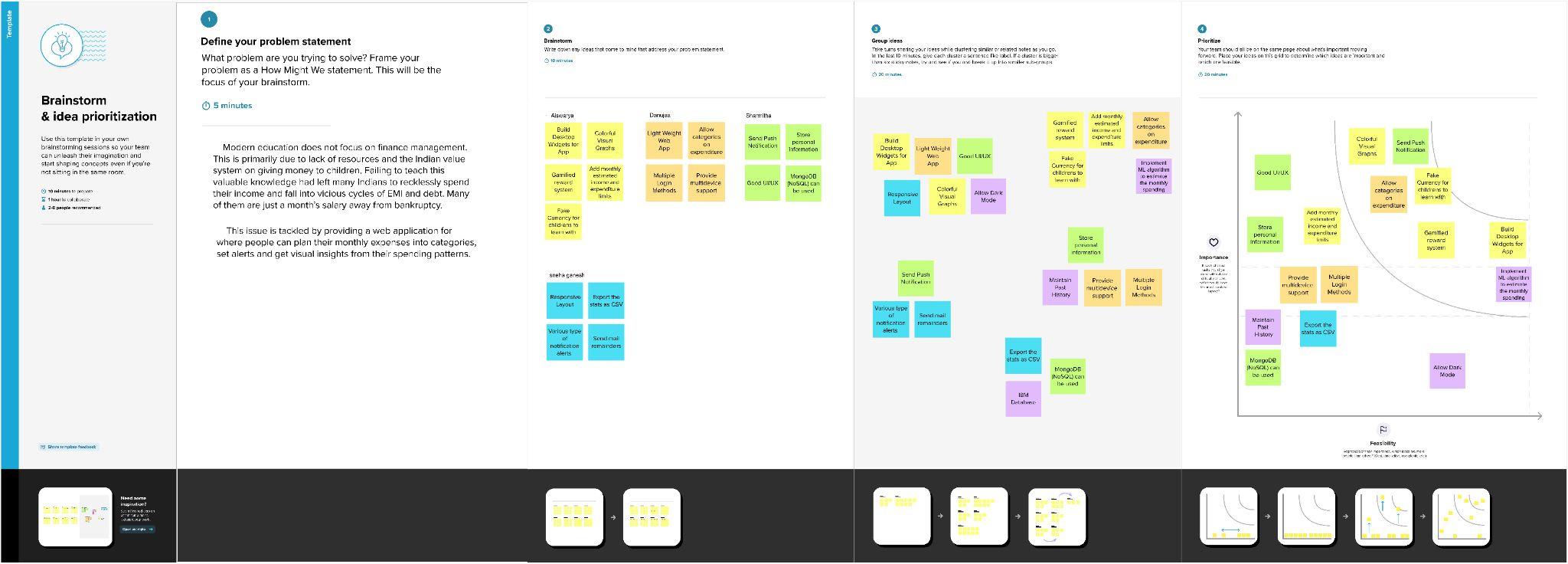
## **CHAPTER 3**

## **3. IDEATION & PROPOSED SOLUTION**

### **3.1 EMPATHY MAP CANVAS**



### **3.2. IDEATION & BRAINSTORMING**

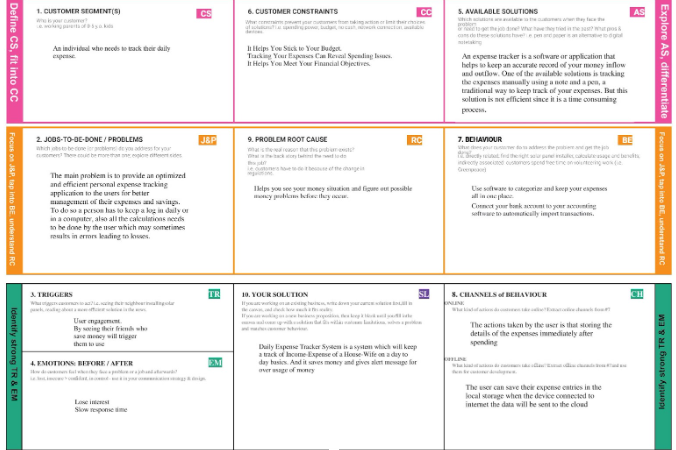


### **3.3. PROPOSED SOLUTION**

| **S.NO.** | **Parameter** | **Description** |
| --- | --- | --- |
| 1. | Problem Statement | In a paper-based expense tracker system it is difficult to track our monthly expenses manually. In a paper-based expense tracker system it is difficult to track our monthly expenses manually. The paper-based expense records may get lost in  case of fire accidents, floods etc. |
| 2. | Scalability of the Solution | This application can handle large numbers of users and data with high performance and security. This application can adapt for both large-scale and small-scale purposes. Easily available in all kinds  of devices. |
| 3. | Idea / Solution description | Daily expense management system which is specially designed for non-salaried and salaried personnel for keeping track of their daily expenditure in an easy and effective way through a computerized system which tends to eliminate manual paperwork. 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. 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. |
| 4. | Novelty / Uniqueness | The user gets notified when their expense exceeds the limit and also it reminds the user when they forgot to make an entry. Tracking expenses through SMS. Data analytics on expenses. Future expense prediction |

|  |  |  |
| --- | --- | --- |
| 5. | Social Impact / Customer Satisfaction | The application should be able to generate reports of their spending and notify users if they have exceeded their budget. It is designed to be dynamic to produce the prediction. It also provides users’ personal information, their income as well as their expenses. This application can create awareness among common people about finance and stuff. This application also helps users to be financially responsible. It Reduces time rather than entering  details manually. |
| 6. | Business Model (Revenue Model) | This Application is provided for free of cost. But It will have some advertisements. In premium version there is no advertisement and contains  some additional features. |

### **3.4. PROBLEM SOLUTION FIT**



## **CHAPTER 4**

## **4. 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 email alert via SMS |
| FR-6 | User Budget Plan | Planning and Tracking of user expense and 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 |
| NFR-2 | Security | Authentication,authorisation and 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 |

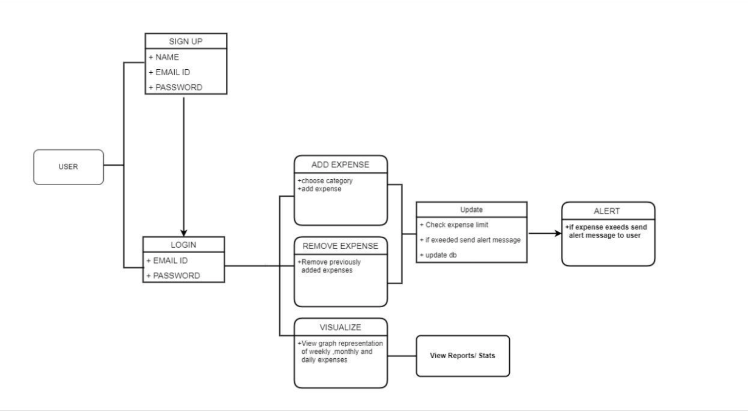
## **CHAPTER 5**

## **5. PRODUCT DESIGN**

### **5.1 Data Flow Diagrams**

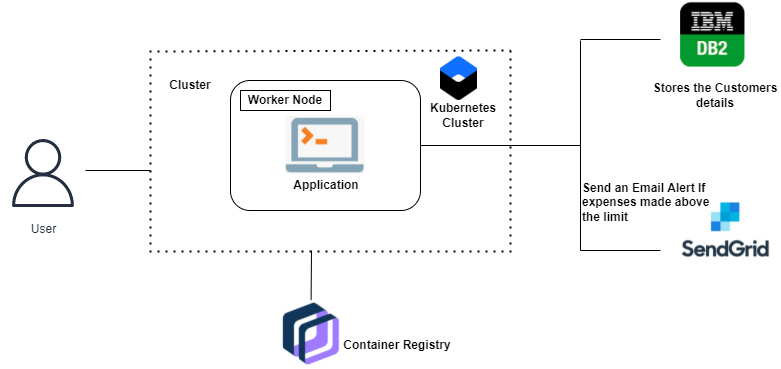
### 

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 Technical Architecture**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



### 

### **5.3 User Stories**

Use the below template to list all the user stories of the product

| **User Type** | **Functional Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Acceptance criteria** | **Priority** | **Release** |
| --- | --- | --- | --- | --- | --- | --- |
| Customer (Mobileuser & 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 an administrator I can upgrade or update the application. | I can fix the bug which arises for the customers and users of the application | Medium |  |

## **CHAPTER 6**

## **6. PROJECT PLANNING & SCHEDULING**

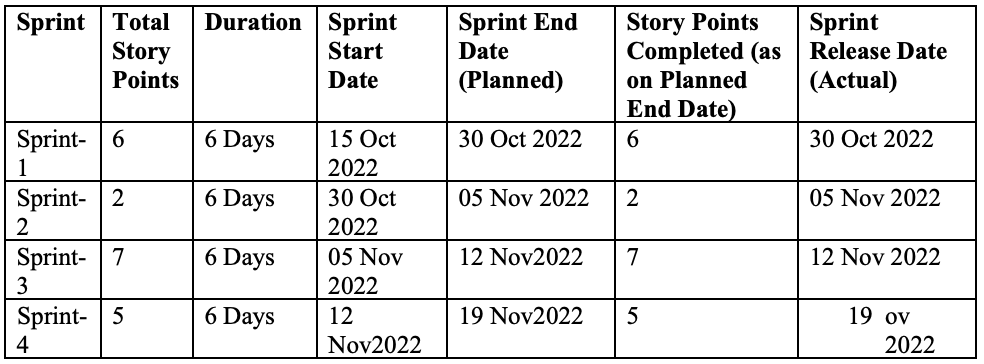
### **6.1 Sprint Planning and Estimation**

| **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 confirmingmy password. | 3 | High | Aiswarya. S  Danujaa. R |
| Sprint-1 | USN-2 | As a user, I will receive confirmation email once  I have registered for the application | 3 | High | Aiswarya. S  Danujaa. R |
| Sprint-1 | Login | USN-3 | As a user, I can log into the application by  entering email & password | 5 | High | Aiswarya. S  Danujaa. R |
| 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 | Aiswarya. S  Danujaa. R |
| Sprint-1 | USN-5 | Once logged In, Keep me logged for few hours  to avoid repeated login if the page is refreshed | 4 | Medium | Aiswarya. S  Danujaa. R |
| Sprint-2 | Expense | USN-6 | Add total income for the month and Allow for  edit option | 6 | High | Sharmitha. S  Sneha ganesh |
| Sprint-2 | USN-7 | Split the total income based on usage like  entertainment, food, shopping etc. | 2 | Low | Sharmitha. S  Sneha ganesh |
| Sprint-2 | USN-8 | Add the day to day expense. | 6 | High | Sharmitha. S  Sneha ganesh |
| Sprint-2 | USN-9 | Display the user added expense | 6 | High | Sharmitha. S  Sneha ganesh |
| Sprint-3 | USN-10 | Filter the expense data based on criteria | 6 | Medium | Aiswarya. S  Danujaa. R |

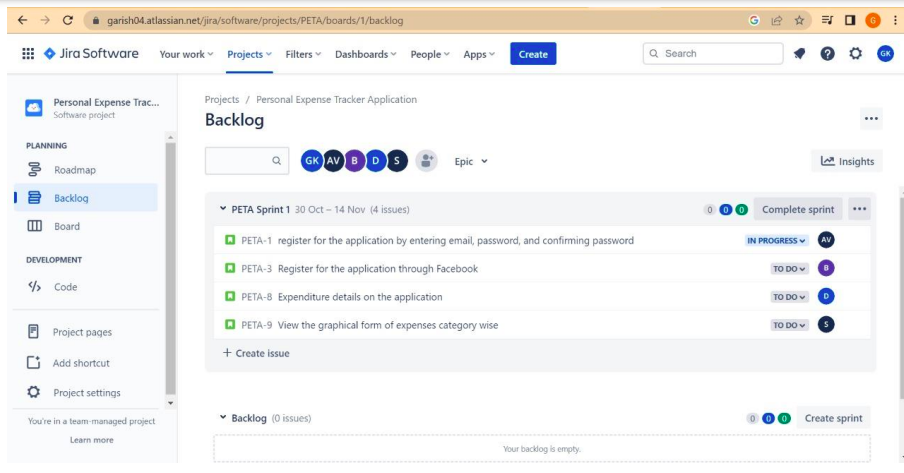
| Sprint-3 | Charts | USN-11 | As a user I can display it in graphs | 4 | Low | Aiswarya.S  Danujaa. R |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint-3 | Alerts | USN-12 | As a user I create custom alert for the balance | 10 | High | Aiswarya. S  Danujaa. R |
| Sprint-4 | Deployment | USN-13 | As a user I should able to access it anywhere in the net | 20 | High | Sharmitha. S  Sneha ganesh |

### 

### **6.2 Sprint Delivery Schedule**

****

### **6.3 Reports from JIRA**

****

## **CHAPTER 7**

### **7.1 Feature 1**

1. Track revenues & expenses.

2. Managing transaction receipts and records.

3. Record & arrange receipts

4. Paying taxes in time.

5. Processing payment and invoices.

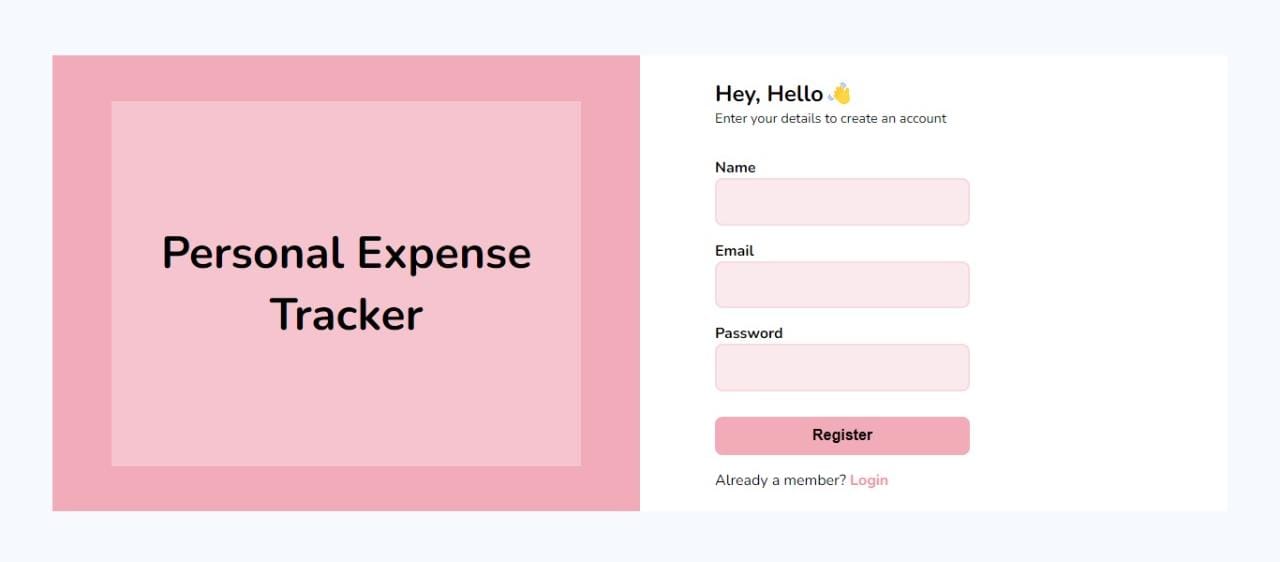
6. Create in-depth reports.

## **CHAPTER 8**

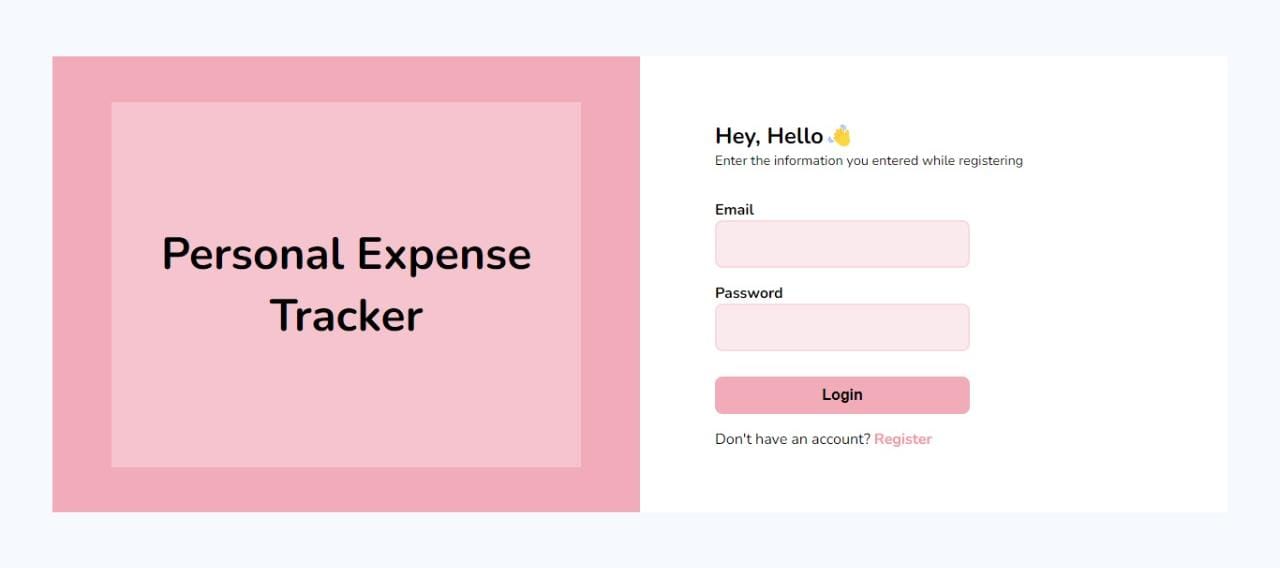
## **TESTING**

### **8.1 Test Cases**

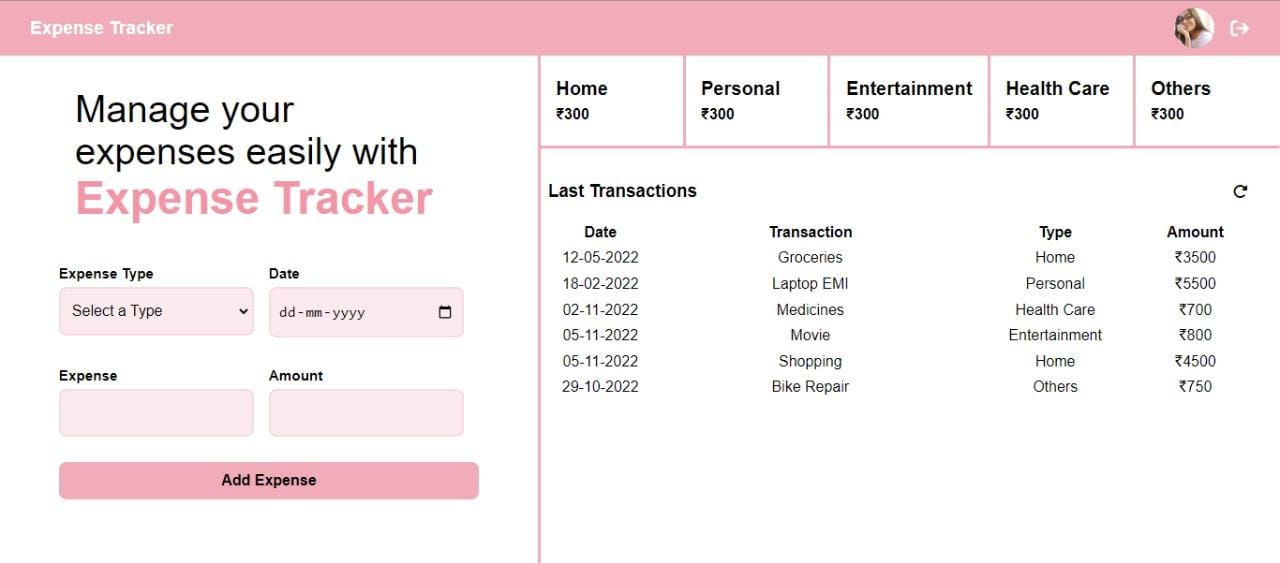
**Registration Page**



**LOGIN**



**ADD EXPENSE**



## **CHAPTER 9**

## **RESULTS**

### **9.1Performance Metrics**

An application can be a very powerful tool for businesses if once the app becomes a success.

However, the success of an app is measured through numbers, metrics, and analytics.

Developing an app takes quite a lot, so once you’ve dedicated much time, money, and effort to the process, it’s mandatory to measure mobile app performance.

## **CHAPTER 10**

## **ADVANTAGES & DISADVANTAGES**

### **10.1 Advantages**

• Improved customer service

• Cloud-based solution

• Order Fulfillment

• Harness Customer Loyalty and Retention

### **10.2 Disadvantages**

• System Clash

• Reduced Physical Audits

• No solution to improve or eliminate bottlenecks in the service cycle

## **CHAPTER 11**

## **11. CONCLUSION**

Taking proper care of our record is crucial in every business, no matter how big or little, we must understand. We must educate ourselves about the idea of effective inventory management and its applications because we can see that managers do not fully grasp it. A company's inventory management system is one of the reasons for its failure. Many customs to combat failure are present, and we can start from this point. Modern technologies can support us in managing and keeping an eye on our inventory. We may learn, put new ideas into practice, and assess our company.

## 

## 

## **CHAPTER 12**

## **12. FUTURE SCOPE**

1) It will have a variety of record-keeping choices (such as food, travel expenses, salary, etc.).

2) It will continue to give updates about our daily spending automatically.

3) Despite being in a haste to make money in today's hectic and expensive world, we eventually gave up. As we naively waste money on unnecessary items and titles. We came over with the intention of following our profit.

4) The user can specify their own expense categories here, such as those for food, clothing, rent, and bills, where they must input the money that has been spent.

## **CHAPTER 13**

## **13. APPENDIX**

### **13.1. SOURCE CODE**

from flask import Flask, session, redirect, render\_template, request, url\_for

import ibm\_db, re

app = Flask(\_name\_)

app.secret\_key = "sss"

connection = ibm\_db.connect("DATABASE=bludb;HOSTNAME=6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SSL;SSLServerCertificate=DigiCert.crt;UID=zbx23177;PWD=0609PXFDuWyfoBlJ",'','')

info = ""

@app.route('/')

def main():

if session and session['active']:

return redirect(url\_for("expenses"))

else:

page = "login"

if request.values.get('query') == "register":

page = "register"

return render\_template('index.html', info=info, page=page)

@app.route('/create-account', methods=['POST'])

def create():

global info

if request.method == 'POST':

name = request.form["name"]

email = request.form["email"]

pwd = request.form["password"]

stmt = ibm\_db.prepare(connection, "SELECT \* FROM account WHERE email=?")

ibm\_db.bind\_param(stmt, 1, email)

ibm\_db.execute(stmt)

acc = ibm\_db.fetch\_assoc(stmt)

if acc:

info = "Email Already exists"

return redirect(url\_for('main', query="register"))

elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):

info = "Enter a valid Email"

return redirect(url\_for('main', query="register"))

else:

info = ""

stmt1 = ibm\_db.prepare(connection, "INSERT INTO account VALUES (?,?,?)")

ibm\_db.bind\_param(stmt1, 1, name)

ibm\_db.bind\_param(stmt1, 2, pwd)

ibm\_db.bind\_param(stmt1, 3, email)

ibm\_db.execute(stmt1)

session['active'] = True

session['email'] = email

return redirect(url\_for("expenses"))

@app.route('/login', methods=['POST'])

def login():

global info

if request.method == "POST":

email = request.form["email"]

pwd = request.form["password"]

stmt = ibm\_db.prepare(connection, "SELECT \* FROM account WHERE email=? AND password=?")

ibm\_db.bind\_param(stmt, 1, email)

ibm\_db.bind\_param(stmt, 2, pwd)

ibm\_db.execute(stmt)

acc = ibm\_db.fetch\_assoc(stmt)

print(acc)

if acc:

info = ""

session['active'] = True

session['email'] = acc['EMAIL']

return redirect(url\_for('expenses'))

else:

info = "Email and Password doesn't match"

return redirect(url\_for('main'))

@app.route('/logout')

def logout():

session.clear()

return redirect(url\_for('main'))

@app.route('/expenses', methods=['GET', 'POST'])

def expenses():

global info

expenses\_amount = [0, 0, 0, 0, 0]

expenses\_list = []

if session and session['active']:

email = session['email']

stmt = ibm\_db.prepare(connection, "SELECT expense, date, expense\_type, amount FROM expenses WHERE email=?")

ibm\_db.bind\_param(stmt, 1, email)

ibm\_db.execute(stmt)

val = ibm\_db.fetch\_assoc(stmt)

while val:

expenses\_list.append(val)

if val['EXPENSE\_TYPE'] == "Home":

amount\_index = 0

elif val['EXPENSE\_TYPE'] == "Personal":

amount\_index = 1

elif val['EXPENSE\_TYPE'] == "Entertainment":

amount\_index = 2

elif val['EXPENSE\_TYPE'] == "Health Care":

amount\_index = 3

else:

amount\_index = 4

expenses\_amount[amount\_index] += val['AMOUNT']

val = ibm\_db.fetch\_assoc(stmt)

print(expenses\_amount)

print(expenses\_list)

info = ""

expenses\_list = expenses\_list[::-1]

return render\_template("expenses.html", expenses\_list=expenses\_list, expenses\_amount=expenses\_amount)

else:

info = "Your session is expired"

return redirect(url\_for('main'))

@app.route('/add', methods=['POST'])

def add():

if request.method == 'POST':

expense\_type = request.form["type"]

date = format\_date(request.form["date"])

expense = request.form["expense"]

amount = request.form["amount"]

email = session['email']

stmt = ibm\_db.prepare(connection, "INSERT INTO expenses VALUES (?,?,?,?,?)")

ibm\_db.bind\_param(stmt, 1, email)

ibm\_db.bind\_param(stmt, 2, expense)

ibm\_db.bind\_param(stmt, 3, date)

ibm\_db.bind\_param(stmt, 4, expense\_type)

ibm\_db.bind\_param(stmt, 5, amount)

ibm\_db.execute(stmt)

return redirect(url\_for("expenses"))

def format\_date(date):

return date[-2:] + "-" + date[5:7] + "-" + date[:4]

if \_name\_ == '\_main\_':

app.run(debug=True)

### **13.2. GITHUB AND PROJECT DEMO LINK**

GitHub Link : <https://github.com/IBM-EPBL/IBM-Project-13179-1659513044>

Project Demo Link :