PERSONAL EXPENSE TRACKER APPLICATION

A PROJECT REPORT
Submitted by

R.SRIDEVI	(814619104018)
K.BHARATHI	(814619104003)
S.LAVANYA	(814619104012)
M.SUBASREE	(814619104302)

In partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

TRICHY ENGINEERING COLLEGE, TRICHY

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

TABLE OF CONTENTS

12. FUTURE SCOPE

13. APPENDIX

Source Code
GitHub & Project Demo Link

CHAPTER 1

1.Introduction

1.1 Project Overview

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

The personal expense tracker application is to read latest authenticated financial news.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.

- To view stock market.
- Offers in popular categories.
- Suggest you with the best investment options.
- Keep track of your money lent or borrowed.
- Keep track of all your daily transactions.

CHAPTER 2

Literature Survey

2.1 Existing System

Expense Manager is a multi-purpose finance related android application intended to run on android devices. The android application can be run on all android devices above android version 5.0. It is designed efficiently to give you the best suggestions for finance planning. The application size is less than 10 MB. It doesn’t need any high end hardware specification. .It can easily run on low end devices. The features of the app are designed in a way to help you for better finance management planning so that you can keep track of , analyse and optimize your budget or spending’s. In this application we are also going to collect user’s data with authenticated permissions and analyse and study their pattern expenses in certain category or by distinct kinds of spending that can be used for studying market trends. These analysis patterns can be derived using some data mining techniques such as clustering,classification and association.

In this world of growing technologies everything is digitalized. With large number of money transaction it is difficult for user to keep track of all your transaction. The main aim of this paper is to manage personal and group expenses. Not many applications provide a variety of features within a single application.

We develop an android application that keeps record of user personal his/her personal contribution towards group expenses, top investment options , view stock market , read authenticated news and grab the best ongoing offers in the market in popular categories.

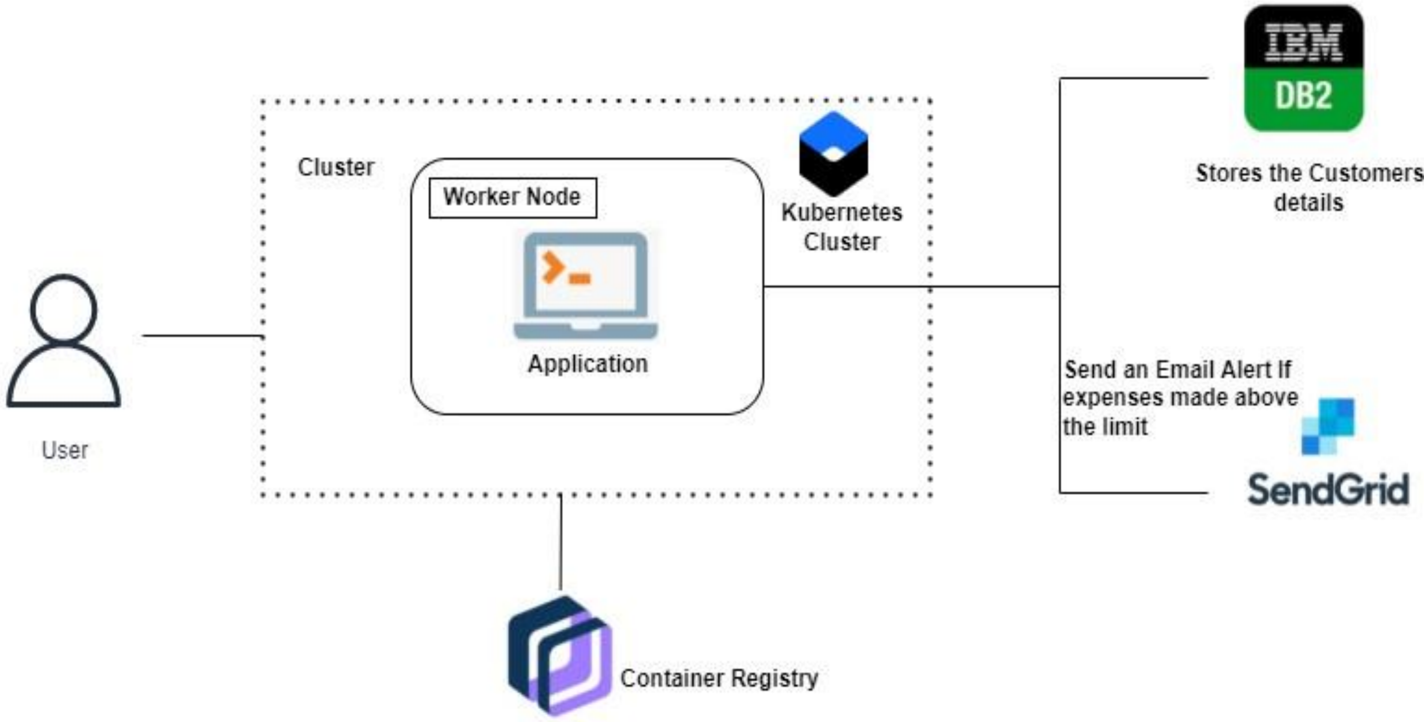
2.2 PROPOSED SYSTEM

In proposed system the personal expense tracker application is to read latest authenticated financial news.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.

- To view stock market.
- Offers in popular categories.
- Suggest you with the best investment options.
- Keep track of your money lent or borrowed.

Keep track of all your daily transactions.



REFERENCE

2.2 Reference

1. Andtek.T (2012).Just Expenses: Expense Tracker. Retrieved October 20, 2012, fromGooglePlay:https://play.google.com/store/apps/details?id=com.andtek.just.expense

2. Carvalho L. A and Basso C.(2014).Telecom Expense Management for Large Organizations; A Practical Guide. iUniverse LLC Bloomington, IN 47403.

3. Kim, A. (2013). Coin Keeper.International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 3, 032013.

4. Brownlow M. (2012). Smartphone statistics and market share. Email-Marketing-Reports, Retrieved from; http://www.email-marketing-reports.com/wireless-mobile/smartphonestatistics.htm,

5. Turpin M. (2017). Retrieved from; http://www.aberdeen.com/research/10255/10255-rr-travel-expense-procurement-po/content.aspx.

6. Peijiang, H.(2010). Expense Manager. Retrieved June 20, 2010, from http://tiny4.org/app/App_by_tinyfool/Tiny4Money.html: Vol.

7. Peijiang, H. (2012). Daily Expense Tracker (income and cash-flowLite). Retrieved August 20, 2012,from http://tiny4.org/app/App_by_tinyfool/Tiny4Money.

8. WildJ. (2010).Fundamental Accounting Principles (18th edition.). New York: McGraw-Hill Companies. pp. 630–633. ISBN 0-07-299653-6.Vol 18.

2.3Problem StatementDefinition

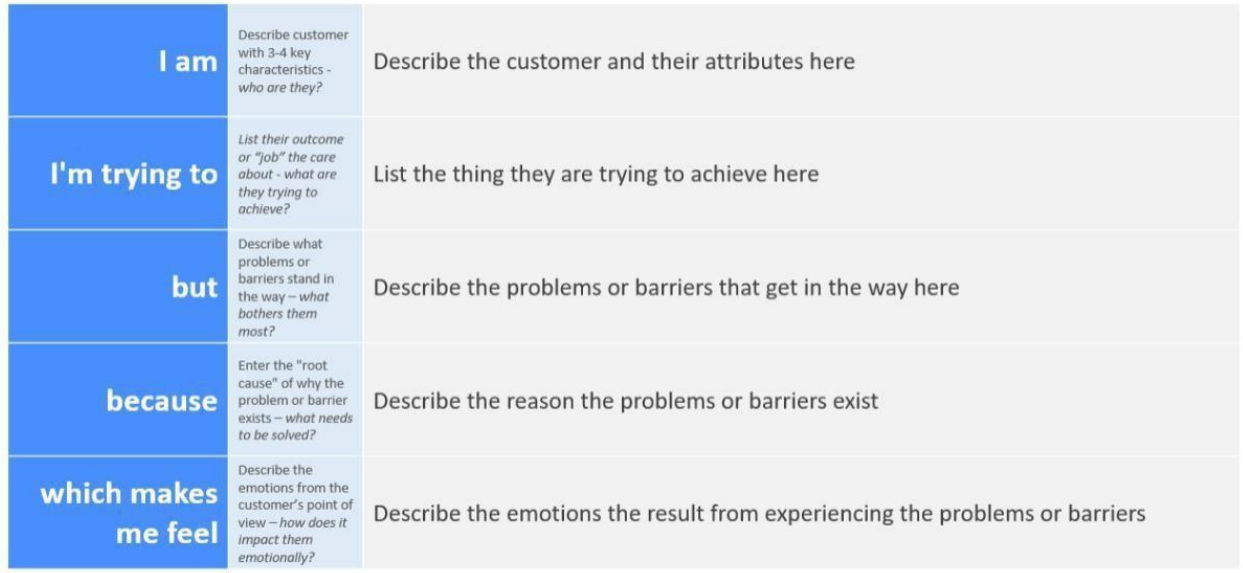
TITLE	AUTHOR/YEAR	MERIT	DEMERIT
Expense Manager	Peijiang/2010	This system provides budget alert in form of notifications to caution users whenever their spending limits are	This system is limited to having facility for keeping expense reports of the users that have not passed a year.

		exceeded .	
Expense Tracker	Andtek/2012	This system provides the users with the facility to save their daily expenses .	It does not provide the customization option for currencies.
Daily Expense Tracker	Peijiang (2012)	This application provides graphical layout wherever possible, for instance, using symbol for cash instead of written format.	It could not provide monthly, yearly and graphical reports which the users of the system had complained about.
Coin Keeper	Kim /2013	This system provides budget alert in form of notifications to caution users whenever their spending limits are exceeded.	It could not provide monthly, yearly and graphical reports which the users of the system had complained about.
Daily Expense Manager	Kim/2015	It has a features that help users to track their daily, weekly and monthly spending without having to spend much time on the application.	It did not provide synchronozation with web server

Problem Statement Definition

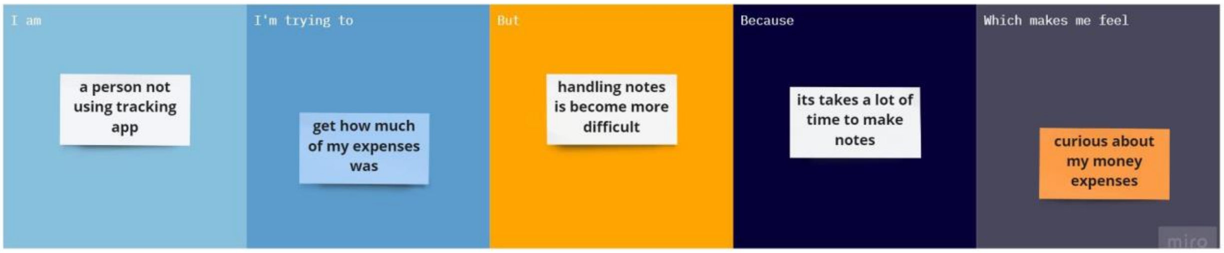
Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you’ll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.



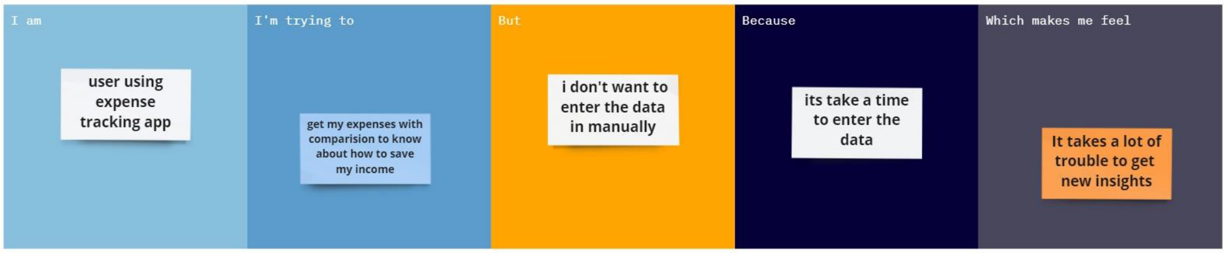
I am	Describe customer with 3-4 key characteristics - *who are they?*	Describe the customer and their attributes here
I'm trying to	*List their outcome or "job" the care about - what are they trying to achieve?*	List the thing they are trying to achieve here
but	Describe what problems or barriers stand in the way – *what bothers them most?*	Describe the problems or barriers that get in the way here
because	Enter the "root cause" of why the problem or barrier exists – *what needs to be solved?*	Describe the reason the problems or barriers exist
which makes me feel	Describe the emotions from the customer's point of view – *how does it impact them emotionally?*	Describe the emotions the result from experiencing the problems or barriers

Problem Statement 1:



I am	I'm trying to	But	Because	Which makes me feel
a person not using tracking app	get how much of my expenses was	handling notes is become more difficult	its takes a lot of time to make notes	curious about my money expenses

Problem Statement 2:



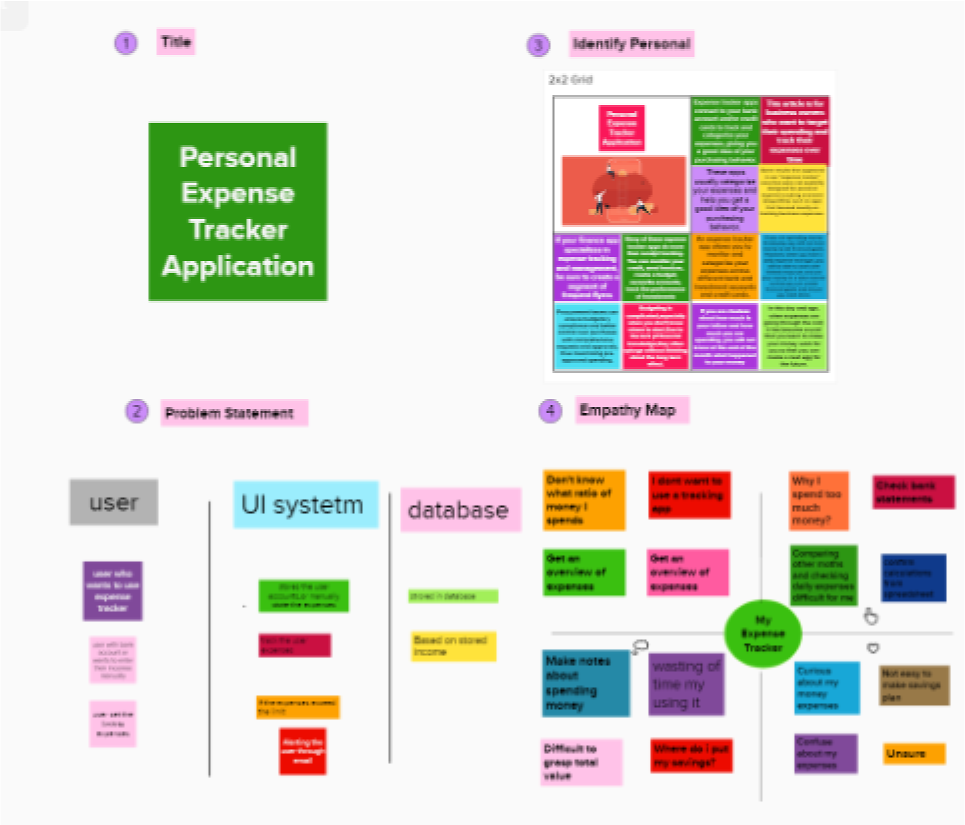
I am	I'm trying to	But	Because	Which makes me feel
user using expense tracking app	get my expenses with comparision to know about how to save my income	i don't want to enter the data in manually	its take a time to enter the data	It takes a lot of trouble to get new insights

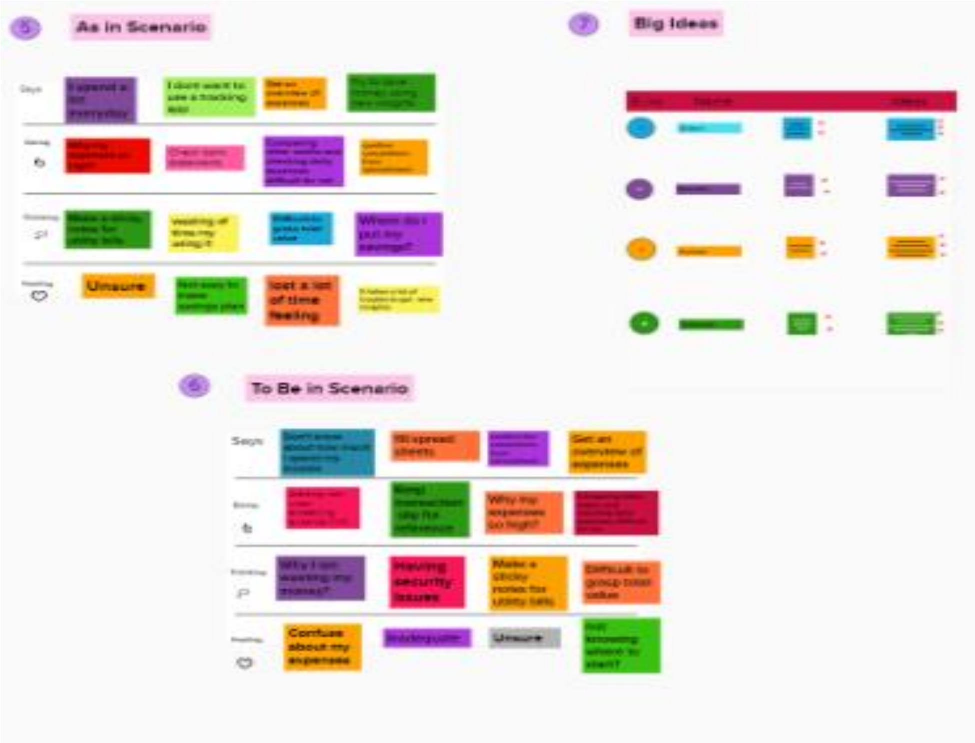
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	a person not using tracking app	get how much of my expenses was	handling notes is become more difficult	its takes a lot of time to make notes	curious about my money expenses
PS-2	user using expense tracking app	get my expenses with comparision to know about how to save my income.	i don't want to enter the data in manually	its take a time to enter the data	It takes a lot of trouble to get new insights

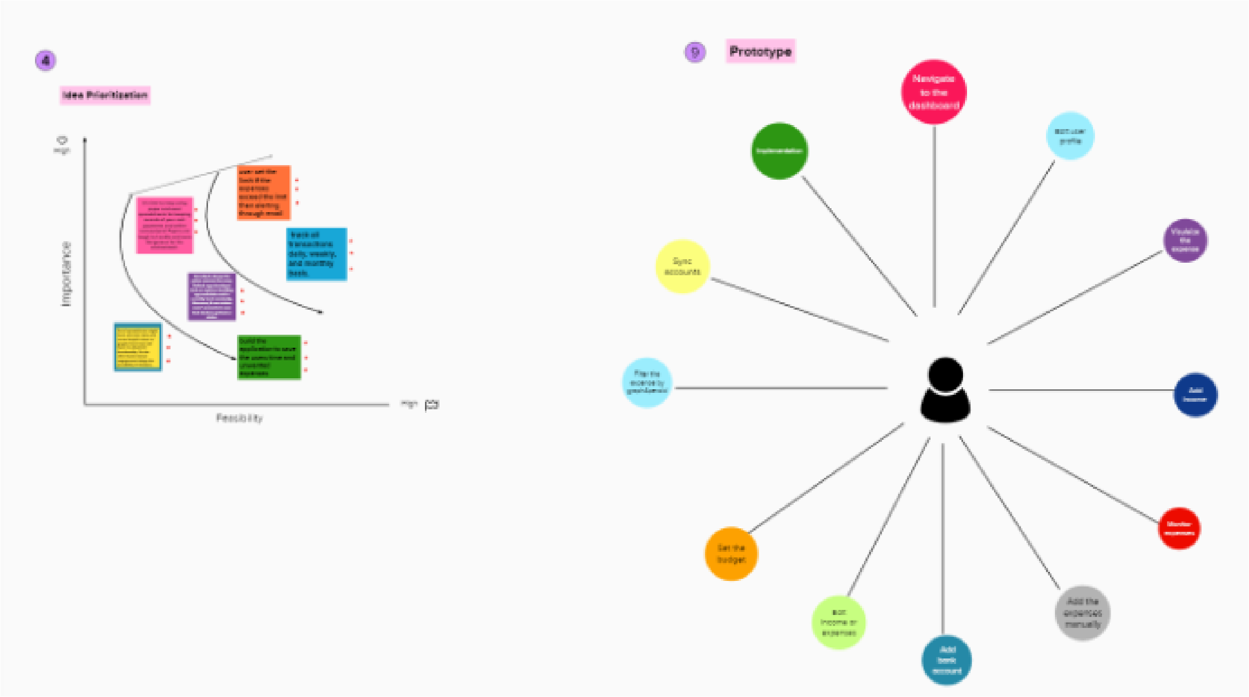
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 **Empathy Map Canvas:**

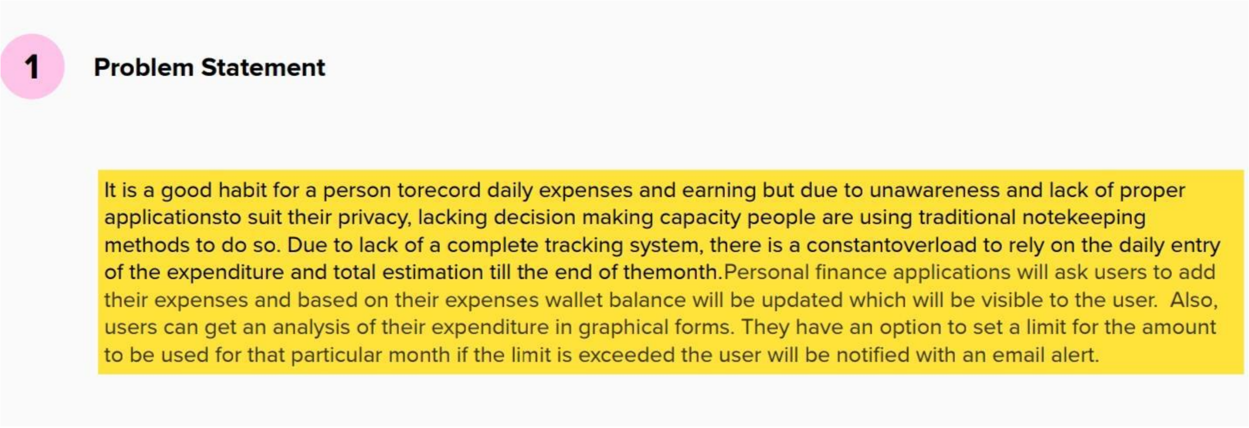

1
Title
Personal Expense Tracker Application
3
Identify Personal
2x2 Grid
2
Problem Statement
user
UI systetm
database
user who wants to use expense tracker
stored in database
Based on stored income
4
Empathy Map
Don't know what ratio of money I spends
I dont want to use a tracking app
Why I spend too much money?
Check bank statements
Get an overview of expenses
Get an overview of expenses
Comparing other moths and checking daily expenses difficult for me
My Expense Tracker
Make notes about spending money
wasting of time my using it
Difficult to grasp total value
Where do I put my savings?
Curious about my money expenses
Not easy to make savings plan
Confuse about my expenses
Unsure


5
As in Scenario
Says
I spend a lot everyday
I dont want to use a tracking app
Get an overview of expenses
Doing
Check bank statements
Thinking
Make a sticky notes for utility bills
wasting of time my using it
Where do I put my savings?
Feeling
Unsure
Not easy to make savings plan
lost a lot of time feeling
7
Big Ideas
6
To Be in Scenario
Says
fill spread sheets
Get an overview of expenses
Doing
Keep transaction slip for reference
Why my expenses so high?
Thinking
Why I am wasting my money?
Having security issues
Make a sticky notes for utility bills
Difficult to group total value
Feeling
Confuse about my expenses
Inadequate
Unsure
Not knowing where to start?


4
Idea Prioritization
High
Importance
Feasibility
High
user set the limit of the expenses exceed the limit then alerting through email
track all transactions daily, weekly, and monthly basis
build the application to save the users time and unwanted expenses
9
Prototype
Navigate to the dashboard
Implementation
Edit user profile
Visualize the expense
Sync accounts
Add Income
Monitor expenses
Set the budget
Edit income or expenses
Add bank account
Add the expenses manually

3.2 Ideation& Brainstorming

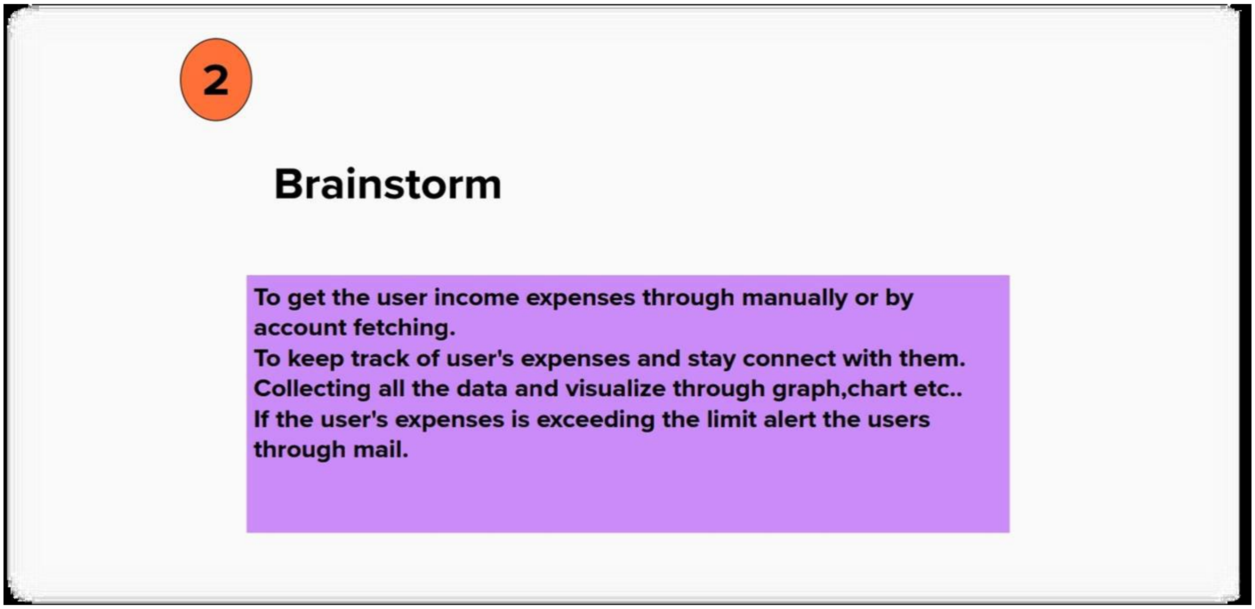
Step-1: Team Gathering, Collaboration and Select the Problem Statement



1 **Problem Statement**

It is a good habit for a person torecord daily expenses and earning but due to unawareness and lack of proper applicationsto suit their privacy, lacking decision making capacity people are using traditional notekeeping methods to do so. Due to lack of a complete tracking system, there is a constantoverload to rely on the daily entry of the expenditure and total estimation till the end of themonth.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.

Step-2: Brainstorm, Idea Listing and Grouping

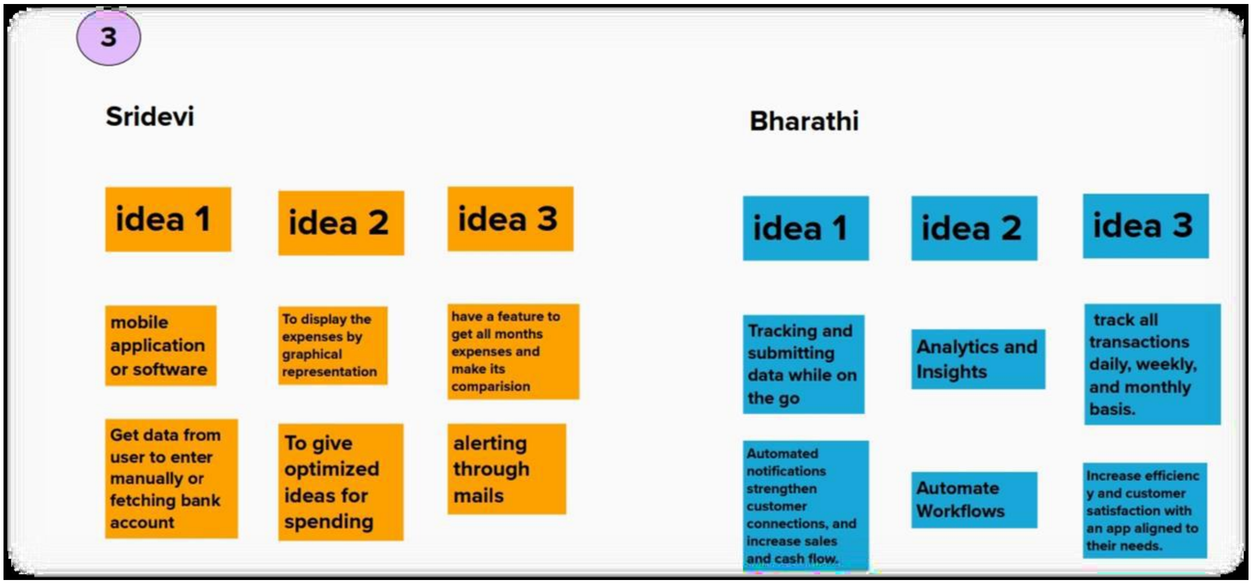


2

Brainstorm

To get the user income expenses through manually or by account fetching.
To keep track of user's expenses and stay connect with them.
Collecting all the data and visualize through graph,chart etc..
If the user's expenses is exceeding the limit alert the users through mail.

Steps 3:Idea Listing and Grouping



3

Sridevi

idea 1

mobile application or software

Get data from user to enter manually or fetching bank account

idea 2

To display the expenses by graphical representation

To give optimized ideas for spending

idea 3

have a feature to get all months expenses and make its comparision

alerting through mails

Bharathi

idea 1

Tracking and submitting data while on the go

Automated notifications strengthen customer connections, and increase sales and cash flow.

idea 2

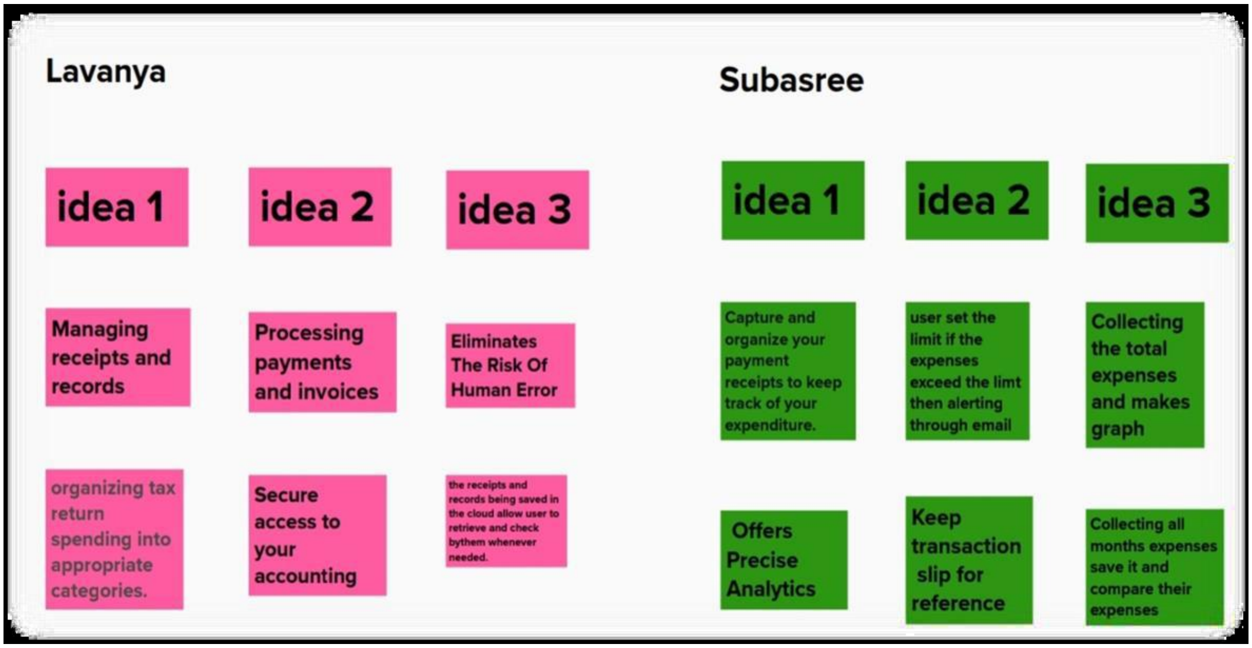
Analytics and Insights

Automate Workflows

idea 3

track all transactions daily, weekly, and monthly basis.

Increase efficienc y and customer satisfaction with an app aligned to their needs.



Lavanya

idea 1

Managing receipts and records

organizing tax return spending into appropriate categories.

idea 2

Processing payments and invoices

Secure access to your accounting

idea 3

Eliminates The Risk Of Human Error

the receipts and records being saved in the cloud allow user to retrieve and check bythem whenever needed.

Subasree

idea 1

Capture and organize your payment receipts to keep track of your expenditure.

Offers Precise Analytics

idea 2

user set the limit if the expenses exceed the limt then alerting through email

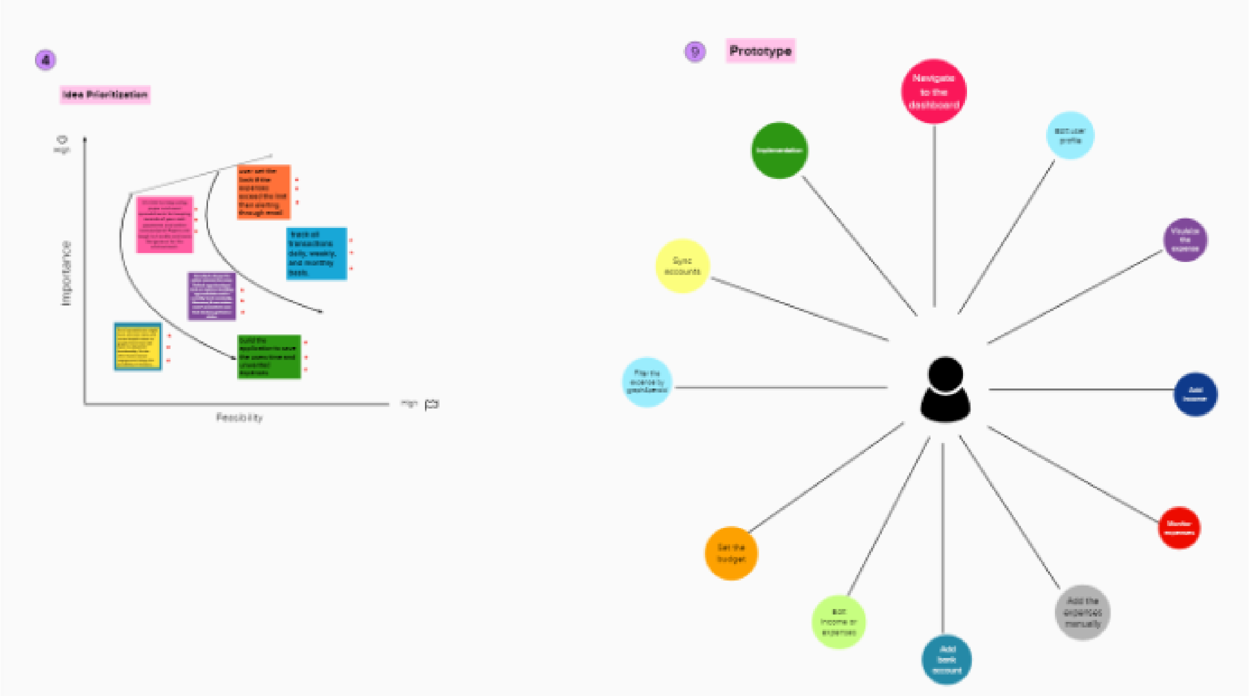
Keep transaction slip for reference

idea 3

Collecting the total expenses and makes graph

Collecting all months expenses save it and compare their expenses

Step-4: Idea Prioritization

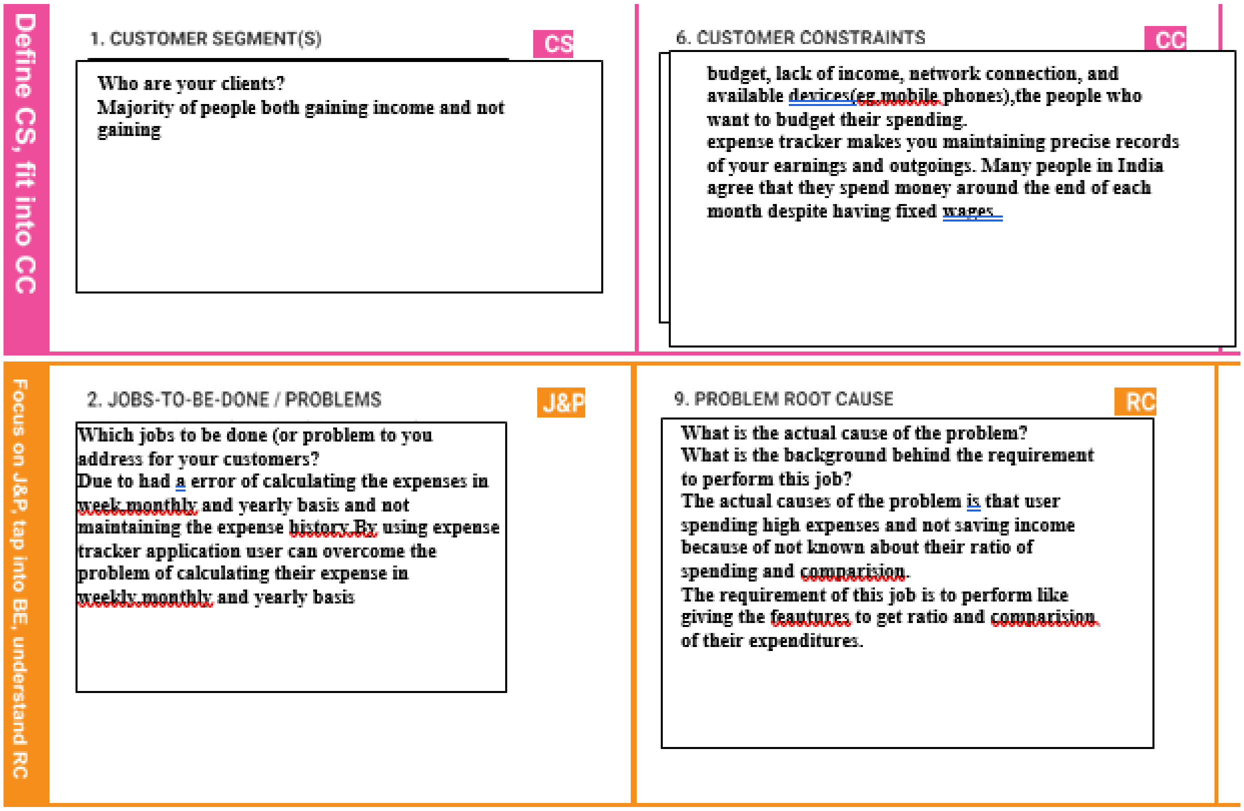

4
Idea Prioritization
Importance
High
Feasibility
High
user set the limit if the expenses exceed the limit then alerting through email
track all transactions daily, weekly, and monthly basis
build the applications to save the users time and unwanted expenses
9
Prototype
Navigate to the dashboard
Implementation
Edit user profile
Visualize the expense
Sync accounts
Filter the expense by
Add income
Monitor expenses
Set the budget
Edit income or expenses
Add bank account
Add the expenses manually

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	People who are not having time to overview their expenses and not aware of their daily expenses.
2.	Idea / Solution description	Unwanted expenses can be reduced and saving your income by having a correct analysis of your expenditure and income. Taking notes to

		calculate expenses lead to error that error can be reduced .
3.	Novelty / Uniqueness	Ratio of graphical representation of your expenses,categorize your each spendings,alert through mail.
4.	Social Impact / Customer Satisfaction	User friendly interface,better handling of day to day expenses
5.	Business Model (Revenue Model)	Achieve your business goals by using web application to get a better financial progress that perfectly suits into your business.
6.	Scalability of the Solution	Tracking your expenses accurately,access this resourses at anywhere and anytime,providing your history of expenses.

3.4Problem Solution fit



Define CS, fit into CC
CS
CC
J&P
RC
Focus on J&P, tap into BE, understand RC

1. CUSTOMER SEGMENT(S)

Who are your clients?
Majority of people both gaining income and not gaining

6. CUSTOMER CONSTRAINTS

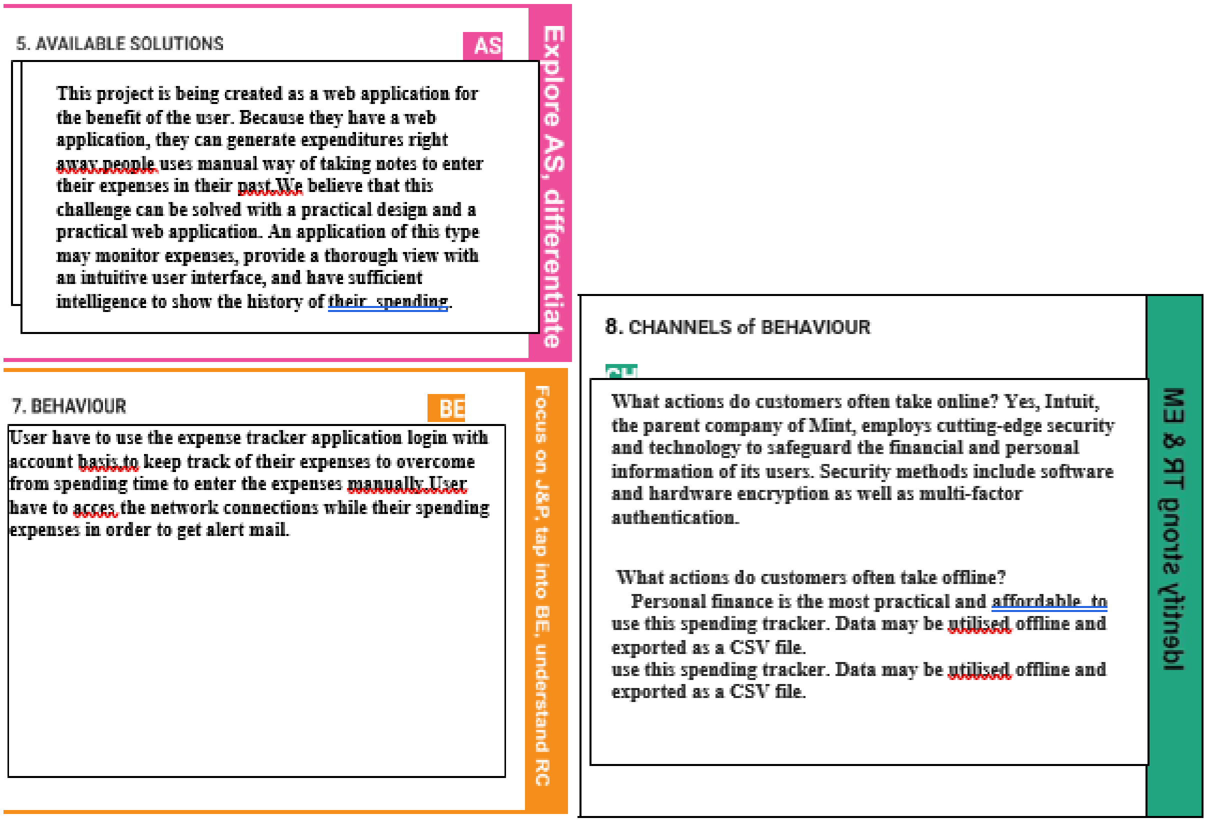
budget, lack of income, network connection, and available devices(eg.mobile phones),the people who want to budget their spending.
expense tracker makes you maintaining precise records of your earnings and outgoings. Many people in India agree that they spend money around the end of each month despite having fixed wages.

2. JOBS-TO-BE-DONE / PROBLEMS

Which jobs to be done (or problem to you address for your customers?
Due to had a error of calculating the expenses in week,monthly and yearly basis and not maintaining the expense history.By using expense tracker application user can overcome the problem of calculating their expense in weekly,monthly and yearly basis

9. PROBLEM ROOT CAUSE

What is the actual cause of the problem?
What is the background behind the requirement to perform this job?
The actual causes of the problem is that user spending high expenses and not saving income because of not known about their ratio of spending and comparision.
The requirement of this job is to perform like giving the feautures to get ratio and comparision of their expenditures.


AS
Explore AS, differentiate
BE
Focus on J&P, tap into BE, understand RC
CH
Identify strong TR & EM

5. AVAILABLE SOLUTIONS

This project is being created as a web application for the benefit of the user. Because they have a web application, they can generate expenditures right away.people uses manual way of taking notes to enter their expenses in their past.We believe that this challenge can be solved with a practical design and a practical web application. An application of this type may monitor expenses, provide a thorough view with an intuitive user interface, and have sufficient intelligence to show the history of their spending.

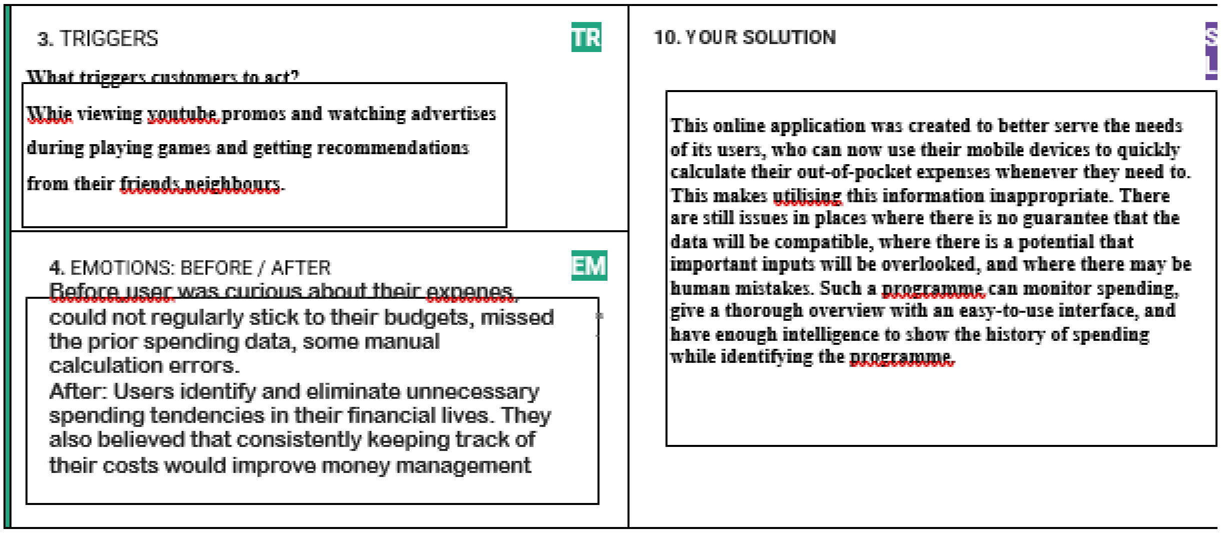
7. BEHAVIOUR

User have to use the expense tracker application login with account basis.to keep track of their expenses to overcome from spending time to enter the expenses manually.User have to acces the network connections while their spending expenses in order to get alert mail.

8. CHANNELS of BEHAVIOUR

What actions do customers often take online? Yes, Intuit, the parent company of Mint, employs cutting-edge security and technology to safeguard the financial and personal information of its users. Security methods include software and hardware encryption as well as multi-factor authentication.

What actions do customers often take offline?
Personal finance is the most practical and affordable to use this spending tracker. Data may be utilised offline and exported as a CSV file.
use this spending tracker. Data may be utilised offline and exported as a CSV file.



3. TRIGGERS

What triggers customers to act?

Whie viewing youtube promos and watching advertises during playing games and getting recommendations from their friends,neighbours.

4. EMOTIONS: BEFORE / AFTER

Before user was curious about their expenes, could not regularly stick to their budgets, missed the prior spending data, some manual calculation errors.
After: Users identify and eliminate unnecessary spending tendencies in their financial lives. They also believed that consistently keeping track of their costs would improve money management

10. YOUR SOLUTION

This online application was created to better serve the needs of its users, who can now use their mobile devices to quickly calculate their out-of-pocket expenses whenever they need to. This makes utilising this information inappropriate. There are still issues in places where there is no guarantee that the data will be compatible, where there is a potential that important inputs will be overlooked, and where there may be human mistakes. Such a programme can monitor spending, give a thorough overview with an easy-to-use interface, and have enough intelligence to show the history of spending while identifying the programme.

CHAPTER 4

4.REQUIREMENT ANALYSIS

4.1Functional Requirements:

FR NO	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	User Registration	Form for collecting details
FR-2	Login	Enter username and password
FR-3	Account Credentials	If the user choose the automate method,then user have to enter the account credentials accurately.
FR-4	Calendar	Personal expense tracker application must 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	Report generation	Graphical representation of report must be generated.
FR-7	Category	This application allow users to add categories of their expense

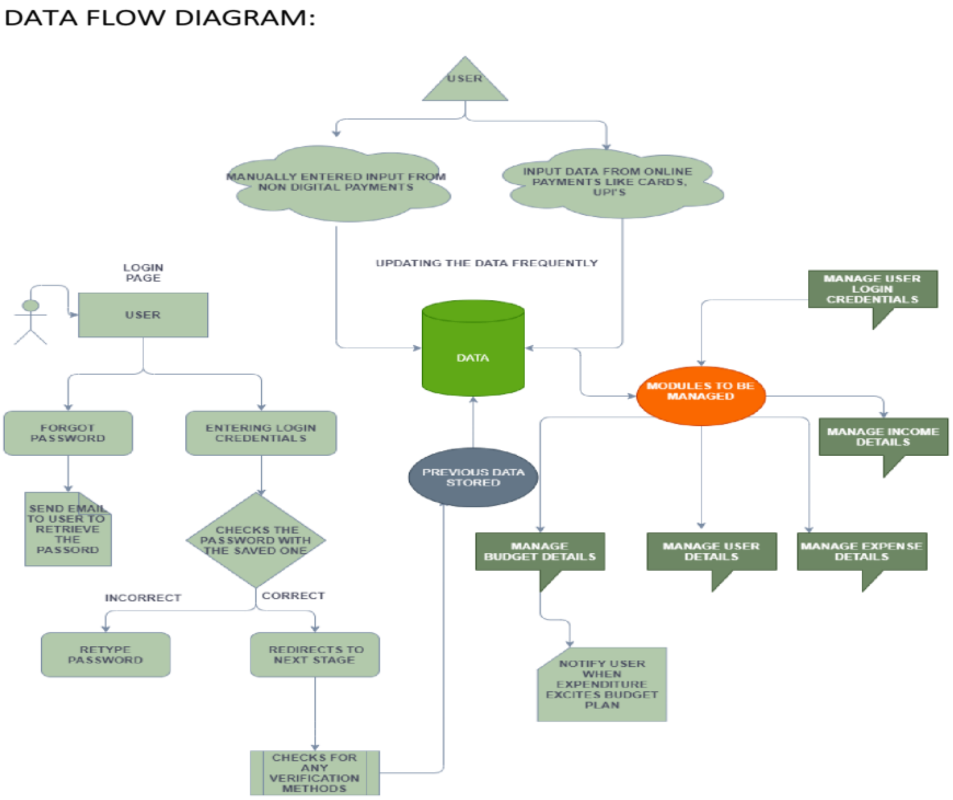
4.2Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Helps to keep an accurate record of your income and expenses.
NFR-2	**Security**	Budget tracking apps are considered very safe from those who commit cyber crimes
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**	the application must have a 100% up-time.
NFR-6	**Scalability**	The ability to appropriately handle increasing demands.

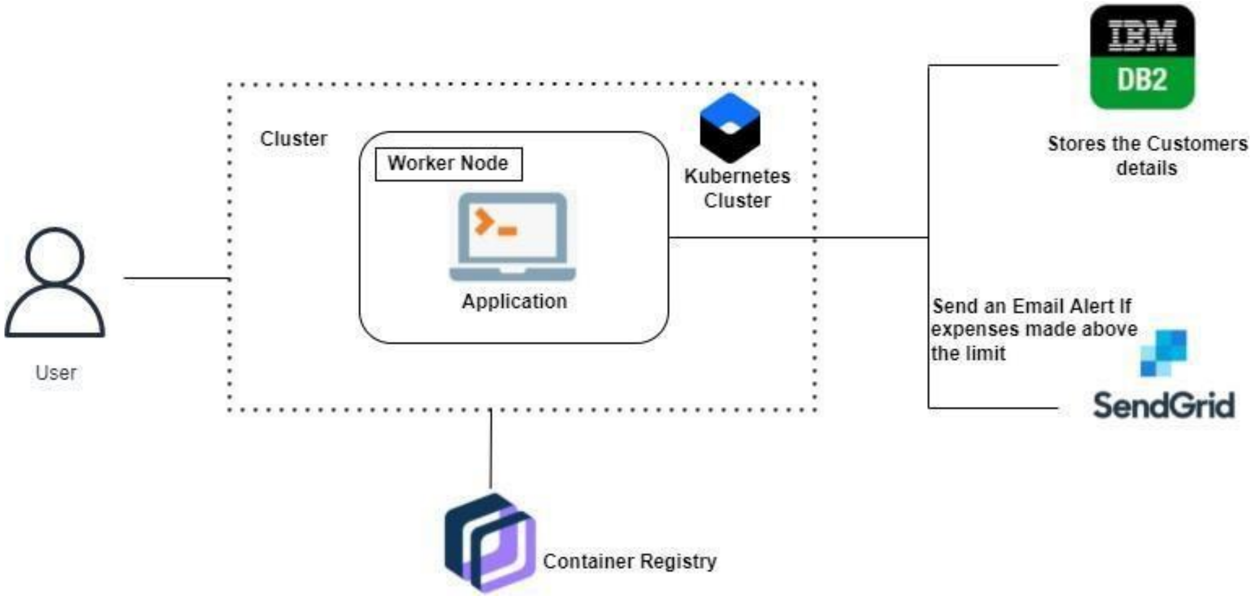
CHAPTER 5

5. PROJECT DESIGN

5.1 Data Flow Diagrams

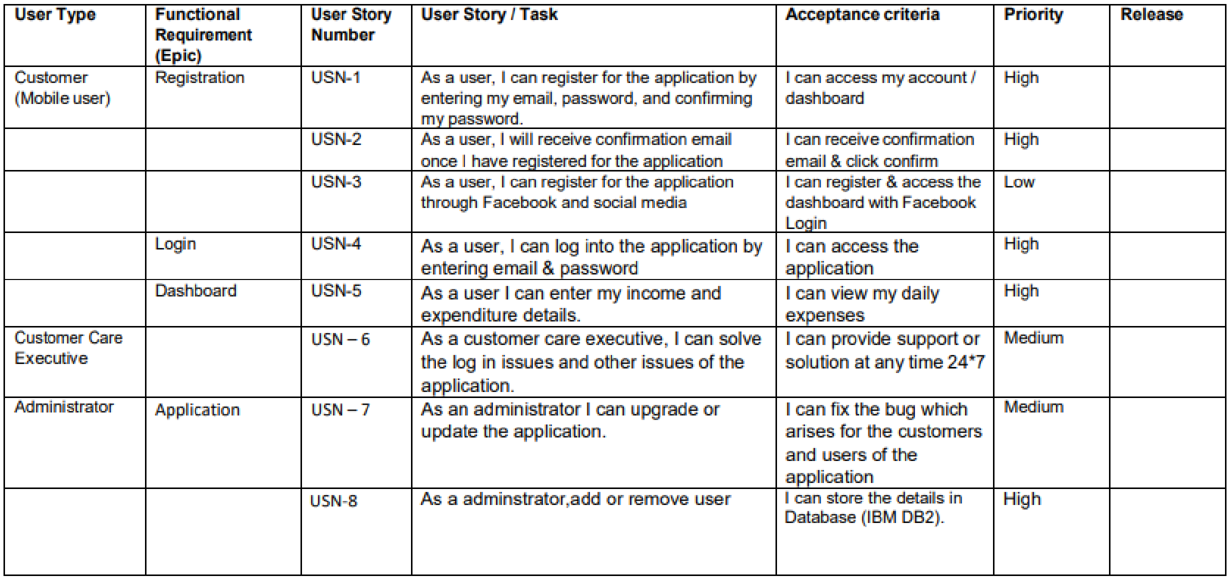

DATA FLOW DIAGRAM:
USER
MANUALLY ENTERED INPUT FROM NON DIGITAL PAYMENTS
INPUT DATA FROM ONLINE PAYMENTS LIKE CARDS, UPI'S
UPDATING THE DATA FREQUENTLY
LOGIN PAGE
USER
DATA
MANAGE USER LOGIN CREDENTIALS
MODULES TO BE MANAGED
FORGOT PASSWORD
ENTERING LOGIN CREDENTIALS
MANAGE INCOME DETAILS
PREVIOUS DATA STORED
SEND EMAIL TO USER TO RETRIEVE THE PASSORD
CHECKS THE PASSWORD WITH THE SAVED ONE
MANAGE BUDGET DETAILS
MANAGE USER DETAILS
MANAGE EXPENSE DETAILS
INCORRECT
CORRECT
RETYPE PASSWORD
REDIRECTS TO NEXT STAGE
NOTIFY USER WHEN EXPENDITURE EXCITES BUDGET PLAN
CHECKS FOR ANY VERIFICATION METHODS

5.2 Solution& Technical Architecture



Application Characteristics

5.1 User Stories



User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile 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 and social media	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	
		USN-8	As a adminstrator,add or remove user	I can store the details in Database (IBM DB2).	High	

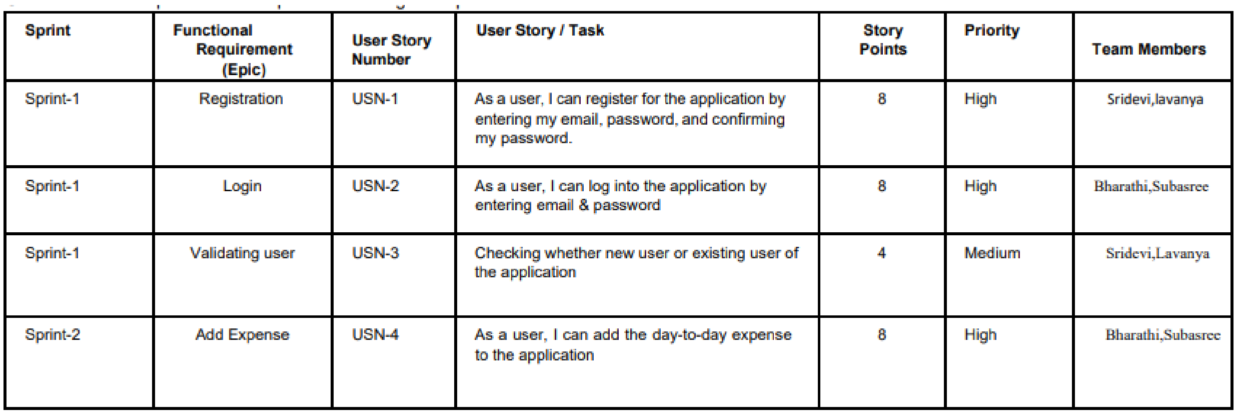
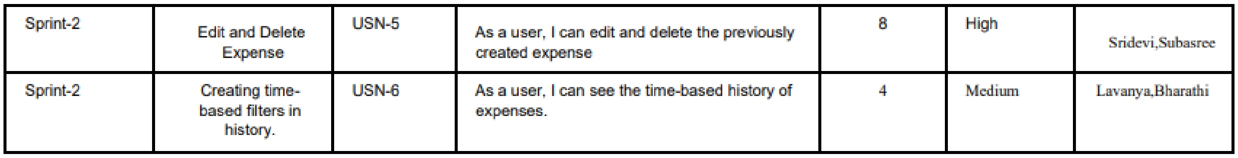
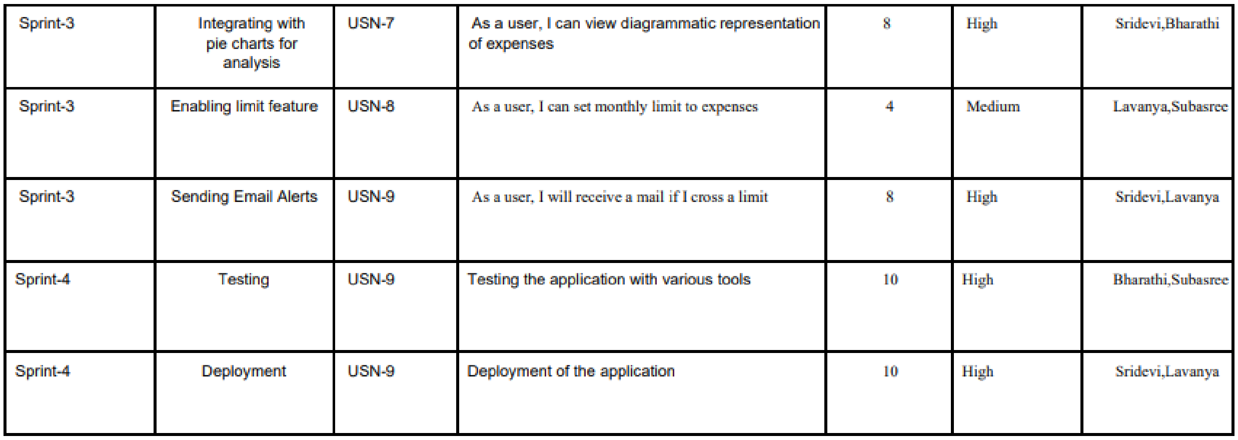
CHAPTER 6
PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

TITLE	DESCRIPTION	DATE
Ideation	List the by organizing the brainstorm and prioritize the top 3 ideas based on the feasibility & importance.	19 SEPTEMBER 2022
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	16 OCTOBER 2022
Problem Solution Fit	Prepare problem - solution fit document	19 OCTOBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	19 OCTOBER 2022
Solution Architecture	Prepare solution architecture document	17 OCTOBER 2022
Customer Journey	Prepare the customer journey maps to	10 OCTOBER 2022

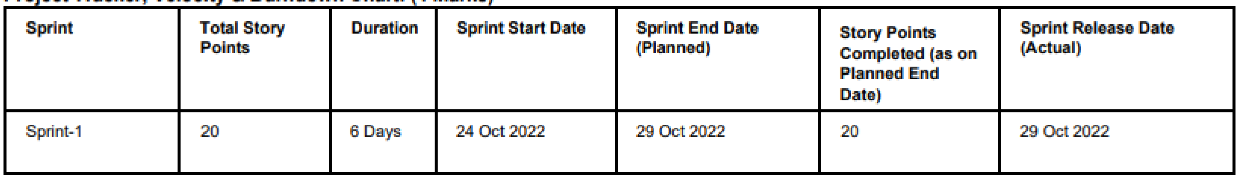
	understand the user interactions & experiences with the application (entry to exit).	
Functional Requirement	Prepare the functional requirement document.	9 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	10 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram	29 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project	29 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS

6.2 Sprint Delivery Schedule

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.	8	High	Sridevi,lavanya
Sprint-1	Login	USN-2	As a user, I can log into the application by entering email & password	8	High	Bharathi,Subasree
Sprint-1	Validating user	USN-3	Checking whether new user or existing user of the application	4	Medium	Sridevi,Lavanya
Sprint-2	Add Expense	USN-4	As a user, I can add the day-to-day expense to the application	8	High	Bharathi,Subasree
Sprint-2	Edit and Delete Expense	USN-5	As a user, I can edit and delete the previously created expense	8	High	Sridevi,Subasree
Sprint-2	Creating time-based filters in history.	USN-6	As a user, I can see the time-based history of expenses.	4	Medium	Lavanya,Bharathi
Sprint-3	Integrating with pie charts for analysis	USN-7	As a user, I can view diagrammatic representation of expenses	8	High	Sridevi,Bharathi
Sprint-3	Enabling limit feature	USN-8	As a user, I can set monthly limit to expenses	4	Medium	Lavanya,Subasree
Sprint-3	Sending Email Alerts	USN-9	As a user, I will receive a mail if I cross a limit	8	High	Sridevi,Lavanya
Sprint-4	Testing	USN-9	Testing the application with various tools	10	High	Bharathi,Subasree
Sprint-4	Deployment	USN-9	Deployment of the application	10	High	Sridevi,Lavanya

Project Tracker, Velocity & Burndown Chart:



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

Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

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

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

AV = 20/6 = 3.33

Burndown Chart:

A burn down 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.

https://www.visual-paradigm.com/scrum/scrum-burndown-chart/
https://www.atlassian.com/agile/tutorials/burndown-charts

Reference:
https://www.atlassian.com/agile/project-management

https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software
https://www.atlassian.com/agile/tutorials/epics
https://www.atlassian.com/agile/tutorials/sprints

https://www.atlassian.com/agile/project-management/estimation

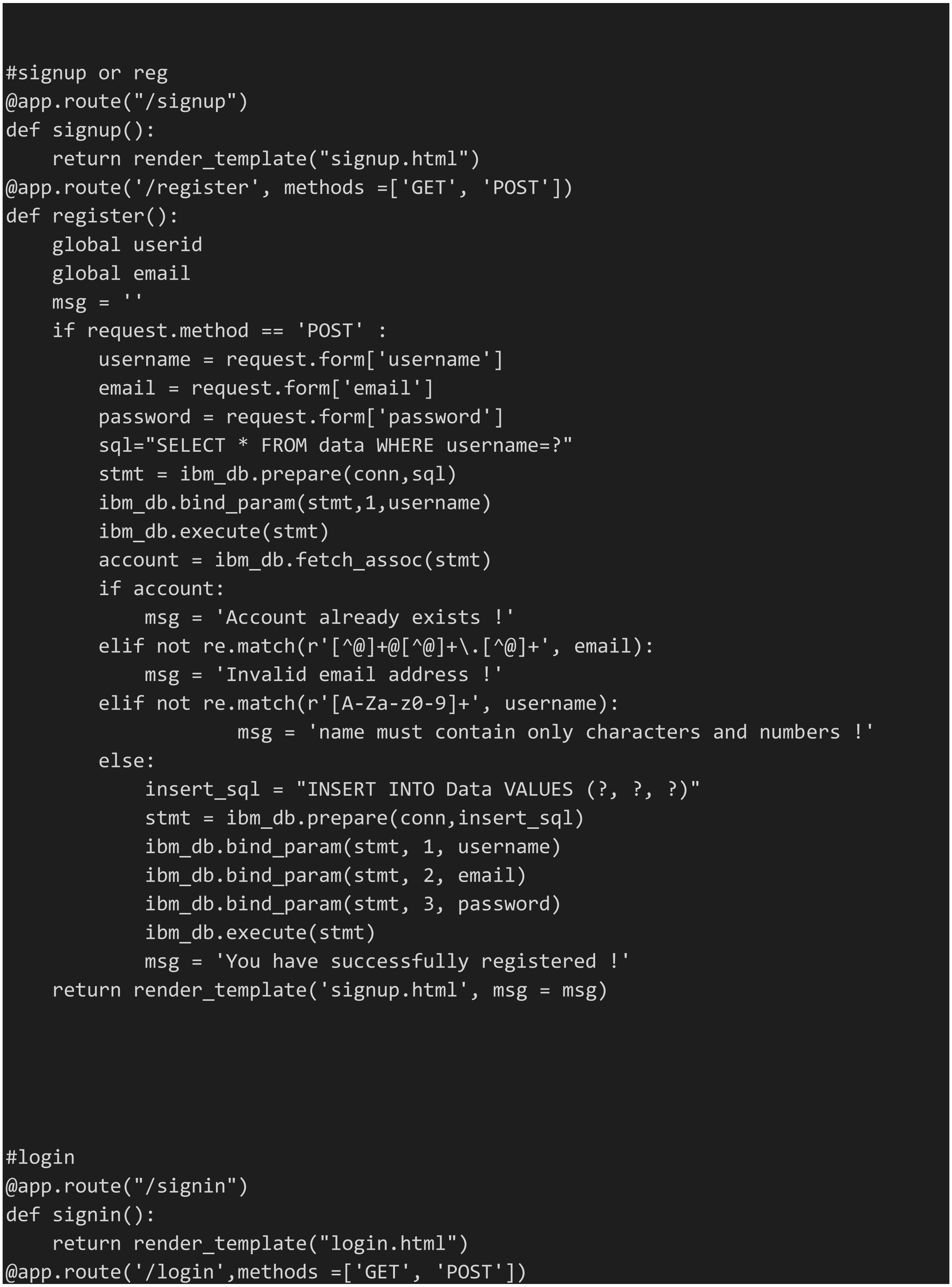
https://www.atlassian.com/agile/tutorials/burndown-charts

CHAPTER 7

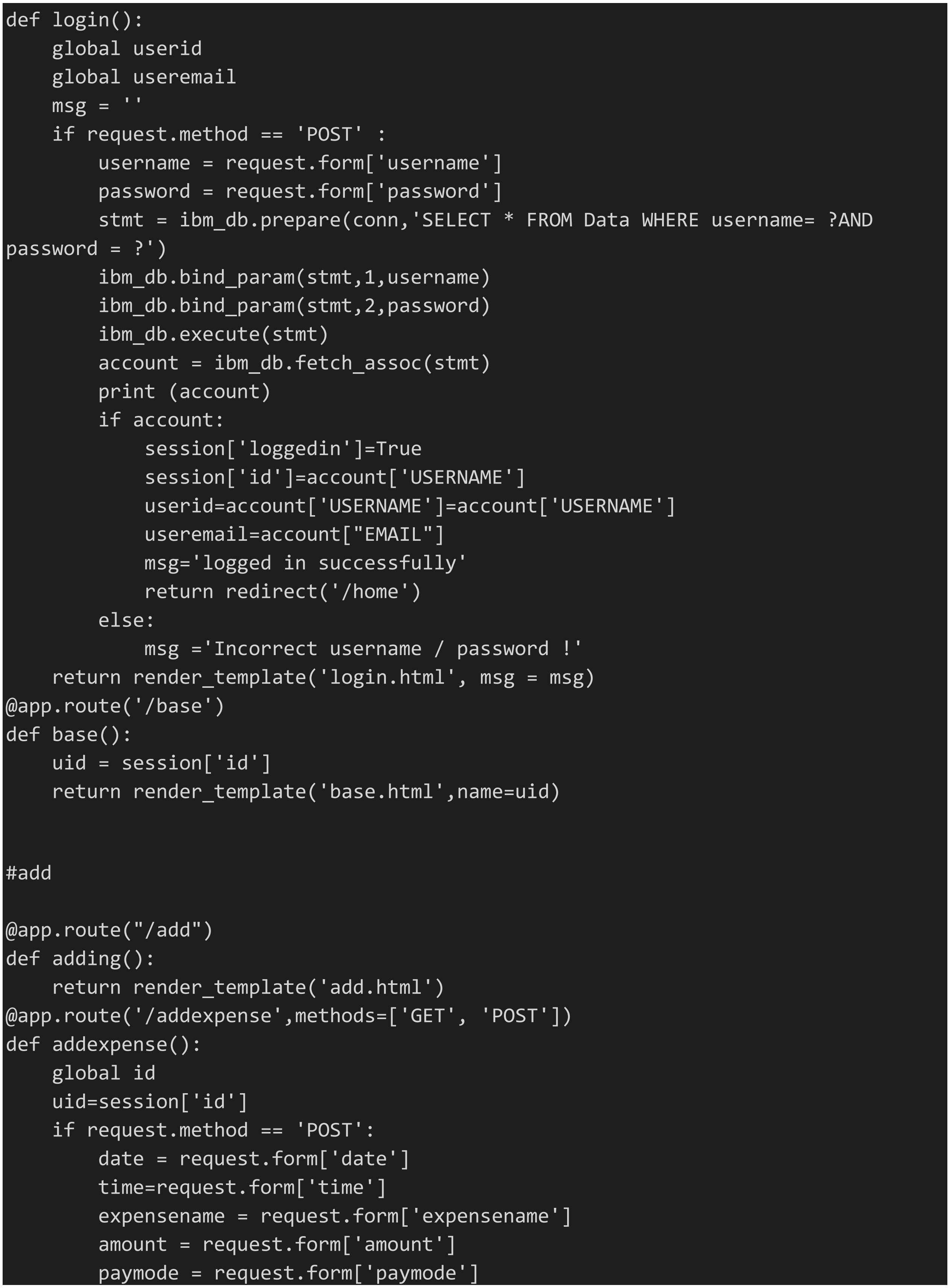
CODING & SOLUTIONING

7.1 Feature1

```
import re
import ibm_db
import ibm_db_dbi

from flask_mail import Mail, Message

from flask import (Flask, flash, redirect, render_template, request, session,
                   url_for)
import pandas as pd
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.base import MIMEBase


app = Flask(__name__)
mail = Mail(app)
app.secret_key = 'a'

app.config['MAIL_SERVER']='smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = 'sridevi032002@gmail.com'
app.config['MAIL_PASSWORD'] = 'srichlmdevi9236'
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True


conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc0-
d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=31321;SECURITY=S
SL;SSLServerCertificate=Certificate.crt;UID=fqd68289;PWD=Pt7Lr9FUjtzoM73p",'','')
connection=ibm_db_dbi.Connection(conn)
#homepage
@app.route("/home")
def home():
    return render_template("homepage.html")
@app.route("/")
def add():
    return render_template("home.html")
```



```
#signup or reg
@app.route("/signup")
def signup():
    return render_template("signup.html")
@app.route('/register', methods =['GET', 'POST'])
def register():
    global userid
    global email
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        sql="SELECT * FROM data WHERE username=?"
        stmt = ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
                    msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO Data VALUES (?, ?, ?)"
            stmt = ibm_db.prepare(conn,insert_sql)
            ibm_db.bind_param(stmt, 1, username)
            ibm_db.bind_param(stmt, 2, email)
            ibm_db.bind_param(stmt, 3, password)
            ibm_db.execute(stmt)
            msg = 'You have successfully registered !'
    return render_template('signup.html', msg = msg)




#login
@app.route("/signin")
def signin():
    return render_template("login.html")
@app.route('/login',methods =['GET', 'POST'])
```

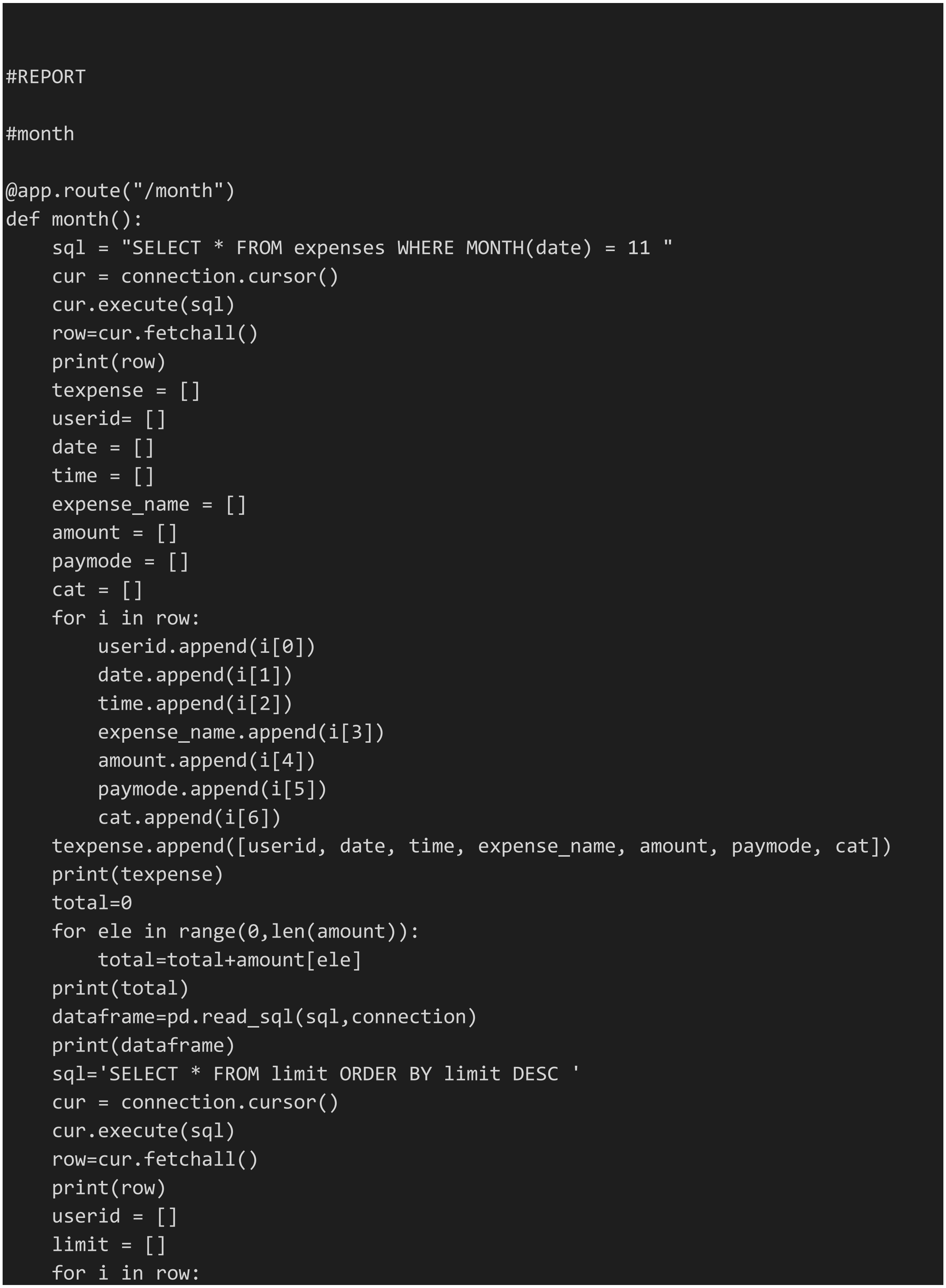


```
def login():
    global userid
    global useremail
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        stmt = ibm_db.prepare(conn,'SELECT * FROM Data WHERE username= ?AND 
password = ?')
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print (account)
        if account:
            session['loggedin']=True
            session['id']=account['USERNAME']
            userid=account['USERNAME']=account['USERNAME']
            useremail=account["EMAIL"]
            msg='logged in successfully'
            return redirect('/home')
        else:
            msg ='Incorrect username / password !'
    return render_template('login.html', msg = msg)
@app.route('/base')
def base():
    uid = session['id']
    return render_template('base.html',name=uid)


#add

@app.route("/add")
def adding():
    return render_template('add.html')
@app.route('/addexpense',methods=['GET', 'POST'])
def addexpense():
    global id
    uid=session['id']
    if request.method == 'POST':
        date = request.form['date']
        time=request.form['time']
        expensename = request.form['expensename']
        amount = request.form['amount']
        paymode = request.form['paymode']
        category = request.form['category']
        uid=session['id']
        print(id)
        sql = "INSERT INTO
expenses(userid,date,time,expensename,amount,paymode,category) VALUES(? ,?, ?, ?,
?, ?, ?)"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,session['id'])
        ibm_db.bind_param(stmt,2,date)
        ibm_db.bind_param(stmt,3,time)
        ibm_db.bind_param(stmt,4,expensename)
        ibm_db.bind_param(stmt,5,amount)
        ibm_db.bind_param(stmt,6,paymode)
        ibm_db.bind_param(stmt,7,category)
        ibm_db.execute(stmt)
        print(date+""+ time + " " + expensename + " " + amount + " " + paymode +
" " + category)
    return redirect("/display")



#display
@app.route('/display')
def graph():
    param = "SELECT * FROM expenses ORDER BY date DESC "
    dataframe=pd.read_sql(param,connection)
    print(dataframe)
    dic=dataframe.to_dict('dict')
    print(dic)
    select="SELECT * FROM expenses "
    cur = connection.cursor()
    cur.execute(select)
    row=cur.fetchall()
    print(row)
    texpense = []
    userid= []
    date = []
    time = []
    expense_name = []
    amount = []
    paymode = []
    cat = []
    for i in row:
        userid.append(i[0])
        date.append(i[1])
        time.append(i[2])
        expense_name.append(i[3])
        amount.append(i[4])
        paymode.append(i[5])
        cat.append(i[6])
    texpense.append([userid, date, time, expense_name, amount, paymode, cat])
    print(texpense)
    dataframe=pd.read_sql(select,connection)
    print(dataframe)
    dic=dataframe.to_dict('dict')
    total=0
    for ele in range(0,len(amount)):
        total=total+amount[ele]
    print(total)
    return render_template('display.html',  total = total
,tables=[dataframe.to_html(classes='data')], titles=dataframe.columns.values)
```

7.2 FEATURE 2

```
 #limit
@app.route("/limit" )
def limit():
       return render_template("limit.html")

@app.route("/limitnum" , methods = ['GET','POST' ])
def limitnum():
    global userid
    if request.method == "POST":
        number= request.form['number']
        sql = "INSERT INTO limit(userid,limit) VALUES(?,?)"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,session['id'])
        ibm_db.bind_param(stmt,2,number)
        ibm_db.execute(stmt)
    return redirect('/base')
#today
```

```
@app.route("/today")
def today():
    select="SELECT * FROM expenses WHERE  DATE(date) = DATE(current date) ORDER
BY date DESC"
    cur = connection.cursor()
    cur.execute(select)
    row=cur.fetchall()
    print(row)
    texpense = []
    userid= []
    date = []
    time = []
    expense_name = []
    amount = []
    paymode = []
    cat = []
    for i in row:
        userid.append(i[0])
        date.append(i[1])
        time.append(i[2])
        expense_name.append(i[3])
        amount.append(i[4])
        paymode.append(i[5])
        cat.append(i[6])
    texpense.append([userid, date, time, expense_name, amount, paymode, cat])
    print(texpense)
    dataframe=pd.read_sql(select,connection)
    print(dataframe)
    dic=dataframe.to_dict('dict')
    total=0
    for ele in range(0,len(amount)):
        total=total+amount[ele]
    print(total)

    #   cursor = mysql.connection.cursor()
    #   cursor.execute('SELECT * FROM expenses WHERE userid = % s AND DATE(date)
= DATE(NOW()) AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))
    #   expense = cursor.fetchall()


    return render_template("today.html", texpense = texpense,   total = total ,
                           expense_name=expense_name,date=date, cat=cat,
paymode=paymode,
                           time=time,tables=[dataframe.to_html(classes='data')],
titles=dataframe.columns.values)
```



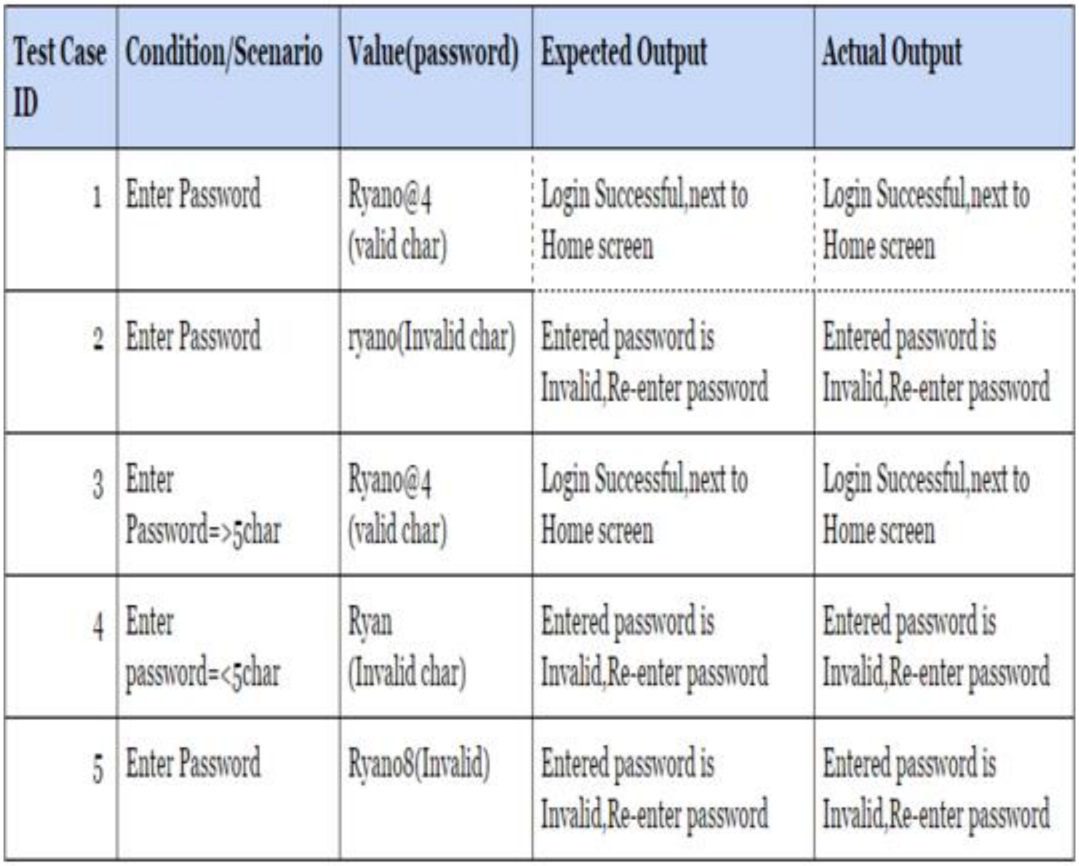
```
#REPORT

#month

@app.route("/month")
def month():
    sql = "SELECT * FROM expenses WHERE MONTH(date) = 11 "
    cur = connection.cursor()
    cur.execute(sql)
    row=cur.fetchall()
    print(row)
    texpense = []
    userid= []
    date = []
    time = []
    expense_name = []
    amount = []
    paymode = []
    cat = []
    for i in row:
        userid.append(i[0])
        date.append(i[1])
        time.append(i[2])
        expense_name.append(i[3])
        amount.append(i[4])
        paymode.append(i[5])
        cat.append(i[6])
    texpense.append([userid, date, time, expense_name, amount, paymode, cat])
    print(texpense)
    total=0
    for ele in range(0,len(amount)):
        total=total+amount[ele]
    print(total)
    dataframe=pd.read_sql(sql,connection)
    print(dataframe)
    sql='SELECT * FROM limit ORDER BY limit DESC '
    cur = connection.cursor()
    cur.execute(sql)
    row=cur.fetchall()
    print(row)
    userid = []
    limit = []
    for i in row:
        userid.append(i[0])
        limit.append(i[1])
    ttotal=0
    for ele in range(0,len(limit)):
        ttotal=total+amount[ele]
    print(ttotal)
    if ttotal<=total:
        msg = Message(
                'Hello',
                sender ='sridevi032002@gmail.com',
                recipients = 'useremail'
               )
        msg.body = 'You have exceeded your expense limit'
        mail.send(msg)
        return 'Sent'
    else:
        print(ttotal)

    return render_template("month.html", texpense = texpense,  total = total ,
                           expense_name=expense_name,date=date, cat=cat,
paymode=paymode,
                           time=time,tables=[dataframe.to_html(classes='data')],
titles=dataframe.columns.values)


#year
@app.route("/year")
def year():
    select="SELECT * FROM expenses WHERE  YEAR(date)=2022"
    cur = connection.cursor()
    cur.execute(select)
    row=cur.fetchall()
    print(row)
    texpense = []
    userid= []
    date = []
    time = []
    expense_name = []
    amount = []
    paymode = []
    cat = []
    for i in row:
        userid.append(i[0])
        date.append(i[1])
        time.append(i[2])
        expense_name.append(i[3])
        amount.append(i[4])
        paymode.append(i[5])
        cat.append(i[6])
    texpense.append([userid, date, time, expense_name, amount, paymode, cat])
    print(texpense)
    total=0
    for ele in range(0,len(amount)):
        total=total+amount[ele]
    print(total)
    dataframe=pd.read_sql(select,connection)
    print(dataframe)

    #   cursor = mysql.connection.cursor()
    #   cursor.execute('SELECT * FROM expenses WHERE userid = % s AND DATE(date)
= DATE(NOW()) AND date ORDER BY `expenses`.`date` DESC',(str(session['id'])))
    #   expense = cursor.fetchall()


    return render_template("year.html", texpense = texpense,   total = total ,
                           expense_name=expense_name,date=date, cat=cat,
paymode=paymode,
                           time=time,tables=[dataframe.to_html(classes='data')],
titles=dataframe.columns.values)
#log-out

@app.route('/logout')

def logout():
   session.pop('loggedin', None)
   session.pop('id', None)
   session.pop('username', None)
   session.pop('email',None)
   return render_template('home.html')



if __name__ == "__main__":
    app.run(debug=True)
```

Database ibm_db is used

CHAPTER 8

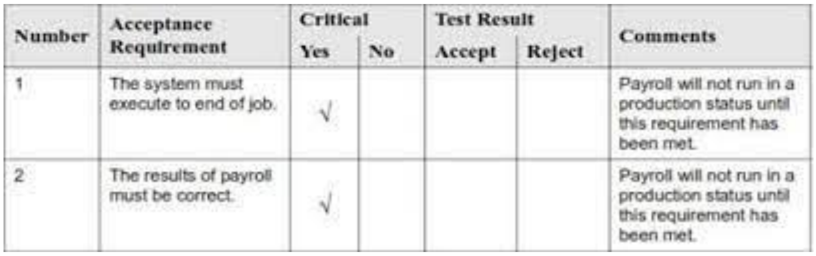
TESTING

8.1 Test Case



Test Case ID	Condition/Scenario	Value(password)	Expected Output	Actual Output
1	Enter Password	Ryano@4 (valid char)	Login Successful,next to Home screen	Login Successful,next to Home screen
2	Enter Password	ryano(Invalid char)	Entered password is Invalid,Re-enter password	Entered password is Invalid,Re-enter password
3	Enter Password=>5char	Ryano@4 (valid char)	Login Successful,next to Home screen	Login Successful,next to Home screen
4	Enter password=<5char	Ryan (Invalid char)	Entered password is Invalid,Re-enter password	Entered password is Invalid,Re-enter password
5	Enter Password	Ryano8(Invalid)	Entered password is Invalid,Re-enter password	Entered password is Invalid,Re-enter password

8.2 User Acceptance Testing



Number	Acceptance Requirement	Critical		Test Result		Comments
		Yes	No	Accept	Reject	
1	The system must execute to end of job.	√				Payroll will not run in a production status until this requirement has been met.
2	The results of payroll must be correct.	√				Payroll will not run in a production status until this requirement has been met.

CHAPTER 9

Results

9.1 Performance Metrics

Performance of expense tracker is to get the total expense amount of current date,month,year and if expense is exceeding the limt alert email will be sent.

CHAPTER 10

Advantages

t’s simple to set up and use. When you’re creating your own method of tracking your finances, you first have to figure out how you’re going to do that. Are you going to use pen and paper, or software, or an excel spreadsheet? What are you going to track? How are you going to input that data, and how often are you going to do it? With an automated app, it tracks everything for you in real time. It has a wealth of information, so no matter what data you feel is important to track, it is all there and available for you – you just need to take a look to see it. There’s an easy user interface for everything as well. Whether you want to **set up a budget**, track a type of expense, or look over your financial history, there’s a tab or an option ready and waiting for you.

Disadvantages

- Negligence during approval
- Negligence during auditing

CHAPTER 11

Conclusion

This 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.

CHAPTER 12

Future Scope

Expense management software solutions have come a long way from being mere reporting tools. They have evolved because of the advances in the spheres of automation and artificial intelligence. Expense management software that employ the latest technology are an absolute must-have for every organization, irrespective of the size or industry.

Companies have been gradually adopting expense management software in a bid to mitigate the losses associated with expense frauds and wasted time. The speed of adoption and change, however, needs to keep pace with emerging trends in the industry.

CHAPTER 13

Appendix

Source code :

Team Id: PNT2022TMID45939

Link=https://github.com/IBM-EPBL/IBM-Project-40920-1660637445/tree/main/Final%20deliverables/sourcecode

Demo link:

https://drive.google.com/file/d/1DpwyDpJhF7oiQYa6Iauqv3uib1_AzEl7/view?usp=sharing_eil_se_dm&ts=637912e2