

Personal Expense Tracker Application

Project Report

Team ID	PNT2022TMID34415
Project Name	Project – Personal Expense Tracker Application
Team Members	Anne Merin Mathew 961819104015 Femina R. J 961819104033 Jerisha.J 961819104044 Jesu Anushaya Nila. B 961819104045

CONTENTS

1. INTRODUCTION

1. Project Overview
2. Purpose

2. LITERATURE SURVEY

1. Existing problem
2. References
3. Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

1. Empathy Map Canvas
2. Ideation & Brainstorming
3. Proposed Solution
4. Problem Solution fit

4. REQUIREMENT ANALYSIS

1. Functional requirement
2. Non-Functional requirements

5. PROJECT DESIGN

1. Data Flow Diagrams
2. Solution & Technical Architecture
3. User Stories

6. PROJECT PLANNING & SCHEDULING

1. Sprint Planning & Estimation
2. Sprint Delivery Schedule

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

1. Feature 1
2. Feature 2
3. Database Schema (if Applicable)

8. TESTING

1. Test Cases
2. User Acceptance Testing

9. RESULTS

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

Mobile applications are top in user convenience and have over passed the web applications in terms of popularity and usability. There are various mobile applications that provide solutions to manage personal and group expense but not many of them provide a comprehensive view of both cases. In this paper, we develop a mobile application developed for the android platform that keeps record of user personal expenses, his/her contribution in group expenditures, top investment options, view of the current stock market, read authenticated financial news and grab the best ongoing offers in the market in popular categories. With our application can manage their expenses and decide on their budget more effectively.

1.2 Purpose

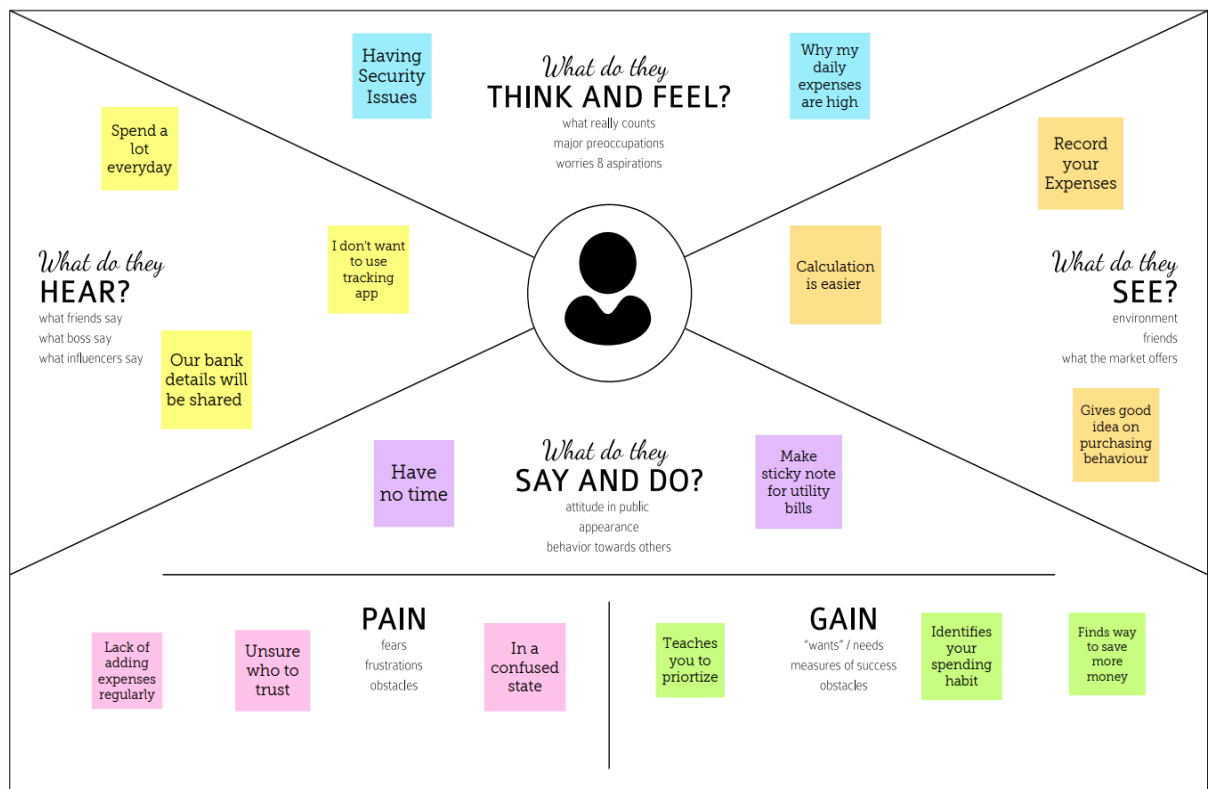
It also known as expense manager and money manager, an expense tracker is a software or application that helps to keep an accurate record of your money inflow and outflow. Many people in India live on a fixed income, and they find that towards the end of the month they don't have sufficient money to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

This Expense Tracker is a web application that facilitates the users to keep track and manage their personal as well as business expenses. This application helps the users

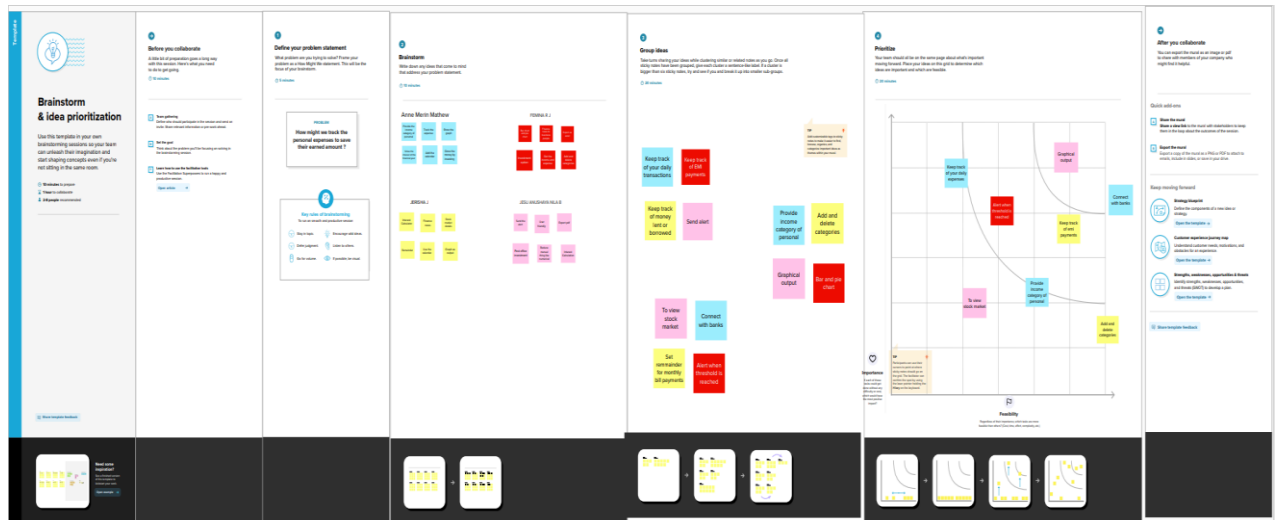
to keep a digital diary. It will keep track of a user's income and expenses on a daily basis. The user will be able to add his/her expenditures instantly and can review them anywhere and anytime with the help of the internet. He/she can easily import transactions from his/her mobile wallets without risking his/her information and efficiently protecting his/her privacy. This expense tracker provides a complete digital solution to this problem. Excel sheets do very little to help in tracking Furthermore, they don't have the advanced functionality of preparing graphical visuals automatically. Notonly it will save the time of the people but also it will assure error free calculations. The user just has to enter the income and expenditures and everything else will be performed by the system. Keywords: Expense Tracker, budget, planning, savings, graphical visualization of expenditure.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map canvas



3.2 Ideation & Brainstorming



3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> Manual calculation is difficult in the modern world so digitalized calculation is developing among the people. Manual calculation sometimes leads to wrong calculations for that the expense app is more useful for all the people. In the expense app we are adding more features such as automatic linking with the debit, credit, mobile banking etc....
2.	Idea / Solution description	<ul style="list-style-type: none"> User friendly Normal people can be able to aware about the finance using this app
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> Market trading and stock marketing and crypto recent updates can be added in this app.
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> It will help the people to track their expenses and alerts when you exceed the limit of your budget. For customer Satisfaction we are improving the user interface. Helps customer to reduce unwanted expenses.

5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> • Customer can use the app in open source. • Customer revenue can be increased using this by avoiding the unwanted Expenses.
6.	Scalability of the Solution	<ul style="list-style-type: none"> • To improve the financial management knowledge among all the people. • User friendly.

3.4 Proposed Solution Fit

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) <ul style="list-style-type: none">• Working Individuals• Students• Budget conscious consumers	6. CUSTOMER CONSTRAINTS <ul style="list-style-type: none">• Internet Access• Device (Smartphone) to access the application• Data Privacy• Cost of existing applications• Trust	5. AVAILABLE SOLUTIONS <ul style="list-style-type: none">• Expense Diary or Excel sheet <p>PROS : Have to make a note daily which helps to be constantly aware</p> <p>CONS : Inconvenient, takes a lot of time</p>							
	2. JOBS-TO-BE-DONE / PROBLEMS <ul style="list-style-type: none">• To keep track of money lent or borrowed• To keep track of daily transactions• Alert when a threshold limit is reached	9. PROBLEM ROOT CAUSE <ul style="list-style-type: none">• Reckless spendings• Indecisive about the finances• Procrastination• Difficult to maintain a note of daily spendings (Traditional methods like diary)	7. BEHAVIOUR <ul style="list-style-type: none">• Make a note of the expenses on a regular basis.• Completely reduce spendings or spend all of the savings• Make use of online tools to interpret monthly expense patterns							
	3. TRIGGERS <ul style="list-style-type: none">• Excessive spending• No money in case of emergency 4. EMOTIONS <table><tr><td>BEFORE</td><td>AFTER</td></tr><tr><td>• Anxious</td><td>• Confident</td></tr><tr><td>• Confused</td><td>• Composed</td></tr><tr><td>• Fear</td><td>• Calm</td></tr></table>	BEFORE	AFTER	• Anxious	• Confident	• Confused	• Composed	• Fear	• Calm	10. YOUR SOLUTION <p>Creating an application to manage the expenses of an individual in an efficient and manageable manner, as compared to traditional methods</p>
BEFORE	AFTER									
• Anxious	• Confident									
• Confused	• Composed									
• Fear	• Calm									
Identify strong TR & EM										

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement	Sub Requirement
FR-1	User Registration	This is a form that collects information from you.
FR-2	Login	You will need to enter your username and password here.
FR-3	Calendar	The user must be able to add the information to their spending in a personal expense tracking application.
FR-4	Expense Tracker	The expense should be graphically represented in this application's report format.
FR-5	Report generation	The report must be represented graphically.
FR-6	Category	Users of this application will be able to add expense categories.

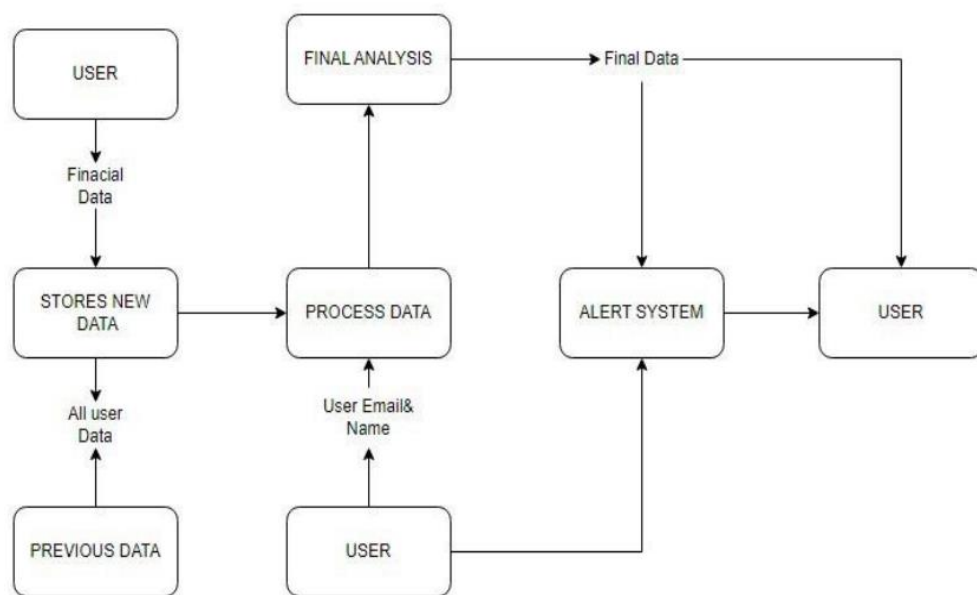
4.2 Non-Functional requirement

NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	Keeps an accurate record of your earnings and outgoings.
NFR-2	Security	A detailed accounting of your income and expenses.
NFR-3	Performance	There are categories of expenses as well as an option. Because of lightweight database support, the system's throughput is increased.
NFR-4	Availability	The application must be completely operational at all times.
NFR-5	Scalability	The application must always function in its entirety.

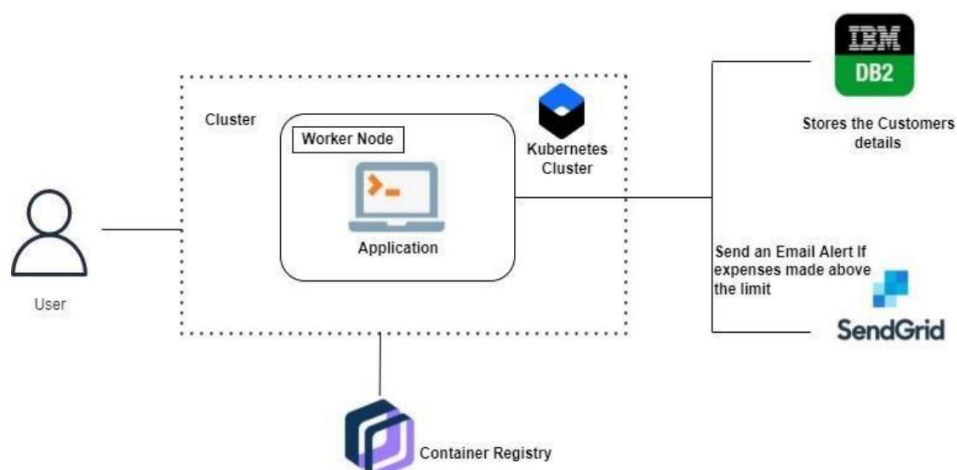
5. PROJECT DESIGN

5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is store.



5.2 Solution & Technical Architecture



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user& web user)	Registration	USN-1	As a user, I can register for the application by entering my email, and password, and confirming my password.	I can access my account/dashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application	I can receive a confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through a Google account.	I can register & access the dashboard with a Google Account login.	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering my email & password	I can access the application.	High	Sprint-1
	Dashboard	USN-6	As a user, I can see the expenditure details and the daily expense.	I can view the daily expenses and add the expense details.	High	Sprint-1
Customer Care Executive		USN-7	As a customer care executive, I can solve the problem that customers face.	I can provide support to customers at any time 24*7.	Medium	Sprint-1
Administrator	Application	USN-8	As an administrator, I can upgrade or update the application.	I can fix any bugs raised by customers and upgrade the application.	Medium	Sprint-1

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 <u>my</u> password.	2	High	Anne Merin Mathew
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application.	1	High	Femina R J
Sprint-1		USN-3	As a user, I can register for the application through Mobile.	2	Low	Jerisha J

Sprint-2		USN-4	As a user, I can register for the application through Gmail.	2	Medium	Jesu Anushaya Nila B
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password.	1	High	Anne Merin Mathew
Sprint-3	Dashboard	USN - 6	As a user I can see the expenditure details on the application.	3	High	Femina R J
Sprint-3	Limits	USN - 6	As a user I can set my monthly expense limit so that I receive a mail on exceeding that.	4	High	Anne Merin Mathew
Sprint-4	Reports	USN - 6	As a user I can view the graphical form of my expenses category wise.	5	Medium	Jerisha J

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	19 Nov 2022

7. Coding and Solutioning:

7.1 Features

Feature 1: Add Expense

Feature 2: Update Expense

Feature 3: Delete Expense

Feature 4: Set Limit

Feature 5: Send Alert Emails to users

7.2 Other Features

Track your expenses anywhere, anytime. Seamlessly manage your money and budget without any financial paperwork. Just click and submit your invoices and expenditures. Access, submit, and approve invoices irrespective of time and location. Avoid data loss by scanning your tickets and bills and saving in the app. Approval of bills and expenditures in real-time and get notified instantly. Quick settlement of claims and reduced human errors with an automated and streamlined billing process.

Code

App. Py:

```
from flask import Flask, render_template, request, redirect, session
import ibm_db

import re

app = Flask( __name__ )

app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-
6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT
=30699;SECURITY=SSL
;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mbs46040;PWD=
MIEpZ1DoqwMRpG vs", "", "")

#HOME--PAGE

@app.route("/home") def home():

    return render_template("homepage.html")

@app.route("/") def add():
```

```
return render_template("home.html")
```

```
#SIGN--UP--OR--REGISTER
```

```
@app.route("/signup") def signup():
```

```
    return render_template("signup.html")
```

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

```
def registet(): msg = "
```

```
    if request.method == 'POST' : username = request.form['username']
```

```
    email = request.form['email'] password = request.form['password']
```

```
    sql = "SELECT * FROM users 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) print(account)
```

```
    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 users VALUES (?, ?, ?)"
```

```
    prep_stmt = ibm_db.prepare(conn, insert_sql)
```

```
    ibm_db.bind_param(prepare_stmt, 1, username)
```

```
    ibm_db.bind_param(prepare_stmt, 2, email)
```

```
ibm_db.bind_param(prepare_stmt, 3, password)
ibm_db.execute(prepare_stmt)

msg = 'You have successfully registered !' elif request.method ==
'POST':

msg = 'Please fill out the form !'

return render_template('register.html', msg = msg)


@app.route('/dashboard') def dash():
return render_template('dashboard.html')


@app.route('/apply',methods =['GET', 'POST']) def apply():
msg = "

if request.method == 'POST' : username = request.form['username']
email = request.form['email']

qualification= request.form['qualification']

skills = request.form['skills']


jobs = request.form['s']

sql = "SELECT * FROM users 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)

print(account)

if account:
```

```

msg = 'there is only 1 job position! for you'
return render_template('apply.html', msg = msg)

insert_sql = "INSERT INTO job VALUES (?, ?, ?, ?, ?)"
prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prepare_stmt, 1, username)
ibm_db.bind_param(prepare_stmt, 2, email)
ibm_db.bind_param(prepare_stmt, 3, qualification)
ibm_db.bind_param(prepare_stmt, 4, skills)
ibm_db.bind_param(prepare_stmt, 5, jobs) ibm_db.execute(prepare_stmt)
msg = 'You have successfully applied for job !'
session['loggedin'] = True
TEXT = "Hello,a new application for job position" +jobs+"is requested"
elif request.method == 'POST':
    msg = 'Please fill out the form !'
    return render_template('apply.html', msg = msg)

@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return render_template('home.html')

if name  == ' main ':

```



```
app.run(host='0.0.0.0')
```

Sendemail.py:

```
import smtplib
```

```
import sendgrid
```

```
import os
```

```
from sendgrid.helpers.mail import Mail, Email, To, Content
```

```
SUBJECT = "expense tracker"
```

```
s = smtplib.SMTP('smtp.gmail.com', 587)
```

```
def sendmail(TEXT,email):
```

```
    print("sorry we cant process your candidature")
```

```
    s = smtplib.SMTP('smtp.gmail.com', 587)
```

```
    s.starttls()
```

```
    s.login("anne@gmail.com", "ANNE@IBM")
```

```
    message = 'Subject: {}\n\n{}'.format(SUBJECT, TEXT)
```

```
    s.sendmail("nandhini@gmail.com", email, message)
```

```
    s.quit()
```

```
def sendgridmail(user,TEXT):
```

```
    from_email = Email("anne@gmail.com")
```

```
    to_email = To(user)
```

```
    subject = "Alert email"
```

```
    content = Content("Your monthly limit has been exceeded!!!",TEXT)
```

```
    mail = Mail(from_email, to_email, subject, content)
```

```
# Get a JSON-ready representation of the Mail object

mail_json = mail.get()

# Send an HTTP POST request to /mail/send

response = sg.client.mail.send.post(request_body=mail_json)
print(response.status_code)

print(response.headers)
```

The other code features are submitted in GitHub: refer the link
<https://github.com/IBM-EPBL/IBM-Project-42556-1660668110>

8.TESTING:

8.1 TESTING:

- Login Page (Functional)
- Login Page (UI)
- Add Expense Page (Functional)

8.2User Acceptance Testing:

1.Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of [product name] project time of the release to user acceptance testing (UAT)

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they are resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	8	15
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	9	2	4	11	20
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	0	1	8
Totals	22	14	11	22	51

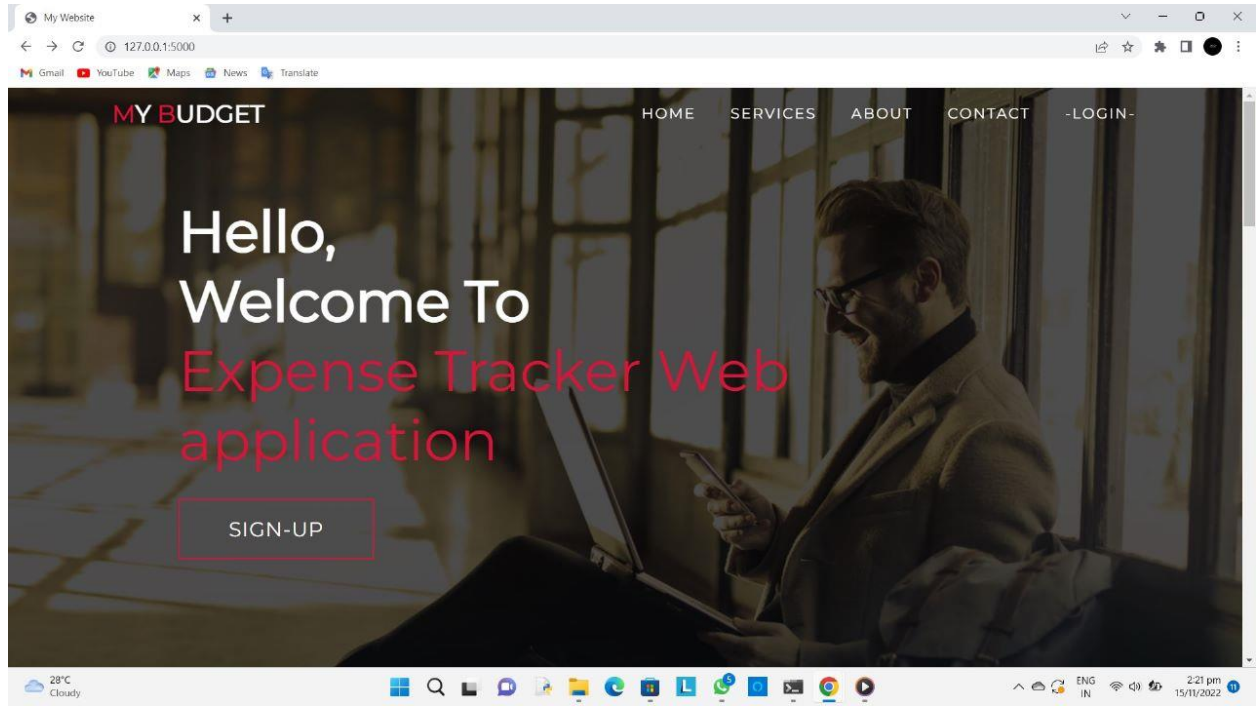
3. Test Case Analysis

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

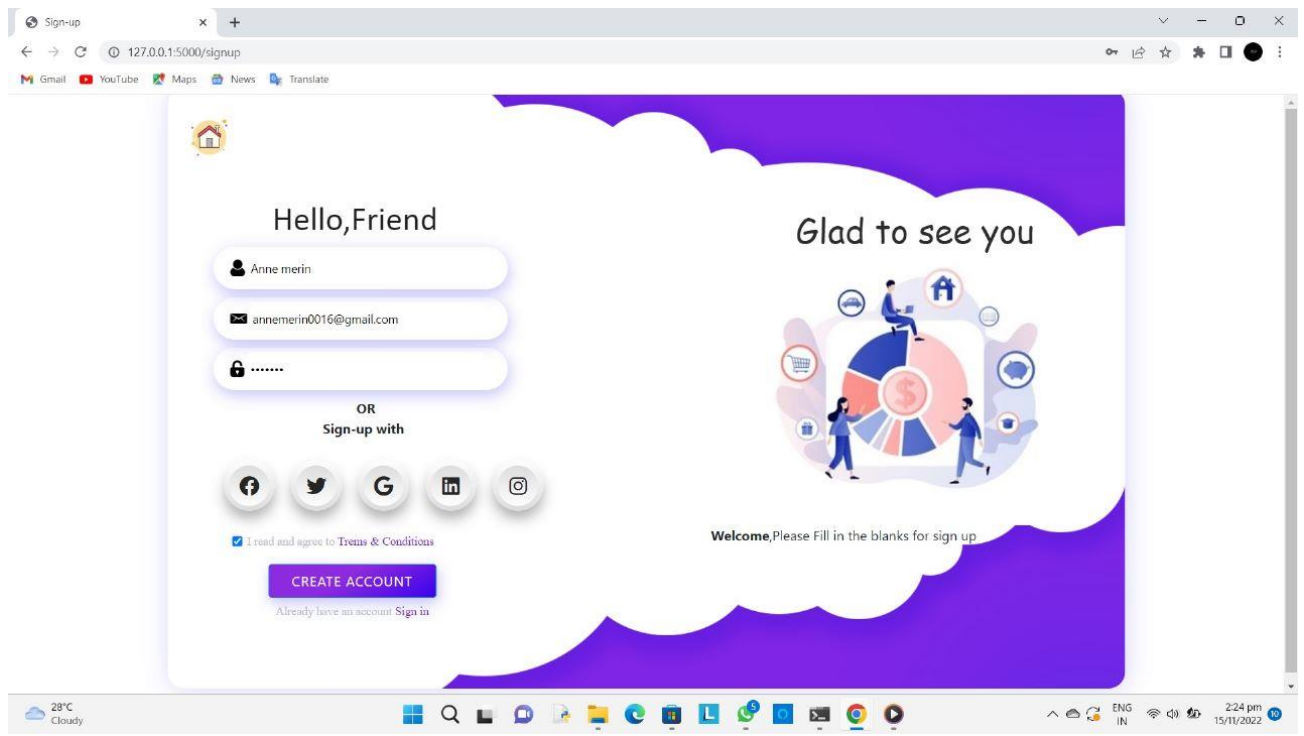
Section	Total Cases	Not Tested	Fail	Pass
Interface	7	0	0	7
Login	43	0	0	43
Logout	2	0	0	2
Limit	3	0	0	3

9.RESULTS

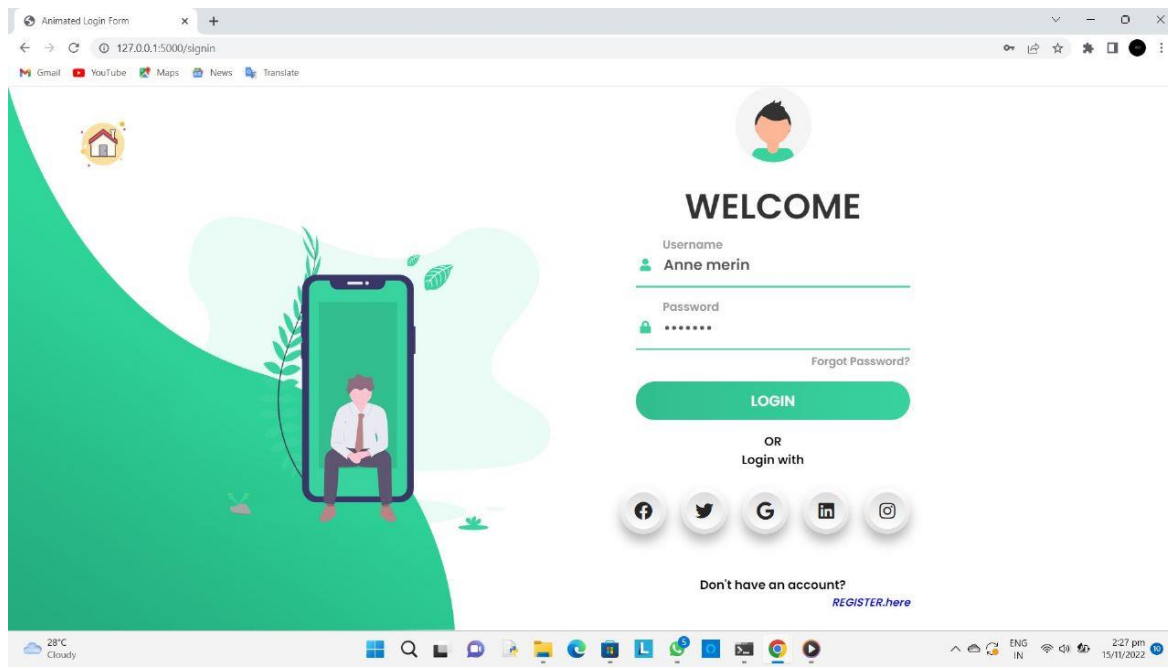
9.1 Home Page:



9.2 Sign Up Page:

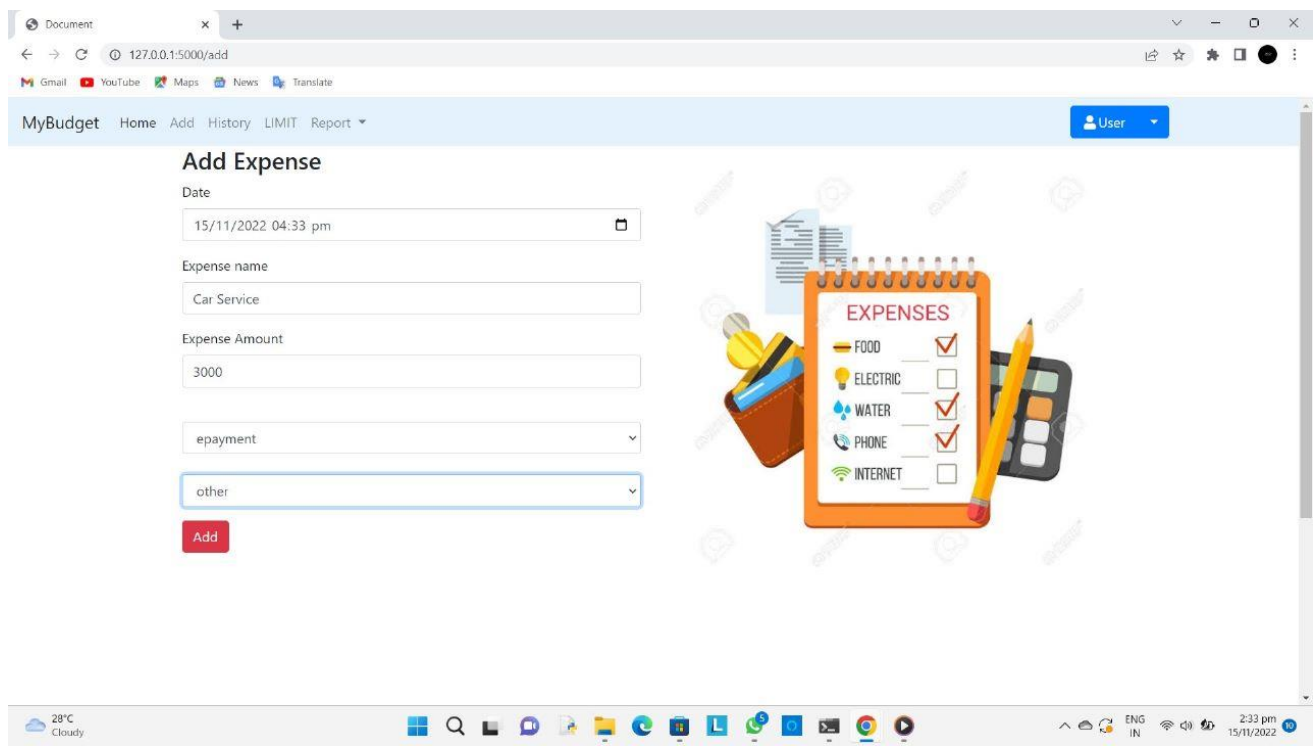


9.3 Login Page:



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/signin". The page features a green background with a large smartphone illustration on the left, showing a person sitting inside. On the right, there is a "WELCOME" section with a user profile icon and the name "Anne merin". Below this, there are input fields for "Username" (containing "Anne merin") and "Password" (masked with dots). A "Forgot Password?" link is present next to the password field. A green "LOGIN" button is below the password field. Underneath the login button, it says "OR Login with" followed by social media icons for Facebook, Twitter, Google, LinkedIn, and Instagram. At the bottom, there is a link "Don't have an account? REGISTER.here". The browser's taskbar at the bottom shows the system clock as 2:27 pm on 15/11/2022.

9.4 Add Expense Page:



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/add". The page has a light blue header with "MyBudget" and navigation links: "Home", "Add", "History", "LIMIT", and "Report". A user profile icon labeled "User" is in the top right. The main content area is titled "Add Expense" and contains several input fields: "Date" (15/11/2022 04:33 pm), "Expense name" (Car Service), "Expense Amount" (3000), and a dropdown menu for "epayment" (other). A red "Add" button is at the bottom left. On the right side, there is a large illustration of a notepad titled "EXPENSES" with a checklist: FOOD (checked), ELECTRIC (unchecked), WATER (checked), PHONE (checked), and INTERNET (unchecked). The browser's taskbar at the bottom shows the system clock as 2:33 pm on 15/11/2022.

9.5 Break down of Expense Page:

EXPENSES					
TV	2000	₹ creditcard	EMI	Edit	Delete
home rent	5000	₹ cash	rent	Edit	Delete
Rent	1000	₹ cash	rent	Edit	Delete
chocolate	400	₹ epayment	food	Edit	Delete
Loan	3000	₹ onlinebanking	business	Edit	Delete

EXPENSE BREAKDOWN

9.6 Limit Page:

Personal Expense Tracker Home Add History LIMIT Report ▾

Currently your MONTHLY limit is ₹ (50000.)

ENTER the MONTHLY LIMIT to avoid over EXPENSES

ENTER

10.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

One of the major pros of tracking spending is always being aware of the state of one's personal finances. Tracking what you spend can help you stick to your budget, not just in a general way, but in each category such as housing, food, transportation and gifts. While a con is that manually tracking all cash that is spent can be irritating as well as time consuming, a pro is that doing this automatically can be quick and simple. Another pro is that many automatic spending tracking software programs are available for free. Having the program on a hand-held device can be a main pro since it can be checked before spending occurs in order to be sure of the available budget.

DISADVANTAGES:

A con with any system used to track spending is that one may start doing it then taper off until it's forgotten about all together. Yet, this is a risk for any new goal such as trying to lose weight or quit smoking. If a person first makes a budget plan, then places money in savings before spending any each new pay period or month, the tracking goal can help. In this way, tracking spending and making sure all receipts are accounted for only needs to be done once or twice a month. Even with constant tracking of one's spending habits, there is no guarantee that financial goals will be met. Although this can be considered to be a con of tracking spending, it could be changed into a pro if one makes up his or her mind to keep trying to properly manage all finances.

11.CONCLUSION

A comprehensive money management strategy requires clarity and conviction for decision- making. You will need a defined goal and a clear vision for grasping the business and personal finances. That's when an expense tracking app comes into the picture. An expense tracking app is an exclusive suite of services for people who seek to handle their earnings and plan their expenses and savings efficiently. It helps you track all transactions like bills, refunds, payrolls, receipts, taxes, etc., on a daily, weekly, and monthly basis.

12. FUTURE SCOPE

- Achieve your business goals with a tailored mobile app that perfectly fits your business.
- Scale-up at the pace your business is growing.
- Deliver an outstanding customer experience through additional control over the app.
- Control the security of your business and customer data.
- Open direct marketing channels with no extra costs with methods such as pushnotifications.
- Boost the productivity of all the processes within the organization.
- Increase efficiency and customer satisfaction with an app aligned to their needs.
- Seamlessly integrate with existing infrastructure.
- Ability to provide valuable insights.
- Optimize sales processes to generate more revenue through enhanced data collection.
- Chats: Equip your expense tracking app with a bot that can understand and answer all user queries and address their needs such as account balance, credit score, etc.
- Prediction: With the help of AI, your mobile app can predict your next purchase, according to your spending behavior. Moreover, it can recommend products and provide unique insights on saving money. It brings out the factors causing fluctuations in your expenses.

13.APPENDIX:

SOURCE CODE

The source code has been uploaded in GitHub. To refer the final source code click this link “ <https://github.com/IBM-EPBL/IBM-Project-42556-1660668110> “

GITHUB AND PROJECT DEMO LINK

The GitHub link: “ <https://github.com/IBM-EPBL/IBM-Project-42556-1660668110> “

The Demo link: “ https://drive.google.com/file/d/1O_BXrWguIErIGWSBqRE0_bLv8xwC-Smz/view?usp=share_link “