PERSONAL EXPENSE TRACKER APPLICATION

IBM-Project-10247-1659151056

NALAIYA THIRAN PROJECT BASED LEARNING ON PROFESSIONAL READLINESS FOR INNOVATION, EMPLOYNMENT AND ENTERPRENEURSHIP

A PROJECT REPORT

BY

Dharun D(211519205042)

Deepak Kumar M (211519205034)

Harshavardhan P (211519205061)

Jeevan S (211519205063)

Karthik Girivas V R(211519205072)

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLGY

PANIMALAR INSTITUTE OF TECHNOLOGY,

Chennai - 600123

INDEX

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 Link

1.INTRODUCTION

1.1 Project Overview

Category: Cloud App Development

Team ID : PNT2022TMID26171

Skills Required:

IBM Cloud,HTML,Javascript,IBM Cloud Object Storage,Python-Flask,Kubernetes,Docker,IBM DB2,IBM Container Registry

Project Description:

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

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

1.2 Purpose

Personal finance management is an important part of people's lives. However, everyone does not have the knowledge or time to manage their finances in a proper manner. And, even if a person has time and knowledge, they do not bother with tracking their expenses as they find it tedious and time-consuming. Now, you don't have to worry about managing your expenses, as you can get access to an expense tracker that will help in the active management of your finances.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. While this problem can arise due to low salary, invariably it is due to poor money management skills.

People tend to overspend without realizing, and this can prove to be disastrous. Using a daily expense manager can help you keep track of how much you spend every day and on what. At the end of the month, you will have a clear picture where your money is going. This is one of the best ways to get your expenses under control and bring some semblance of order to your finances.Today, there are several expense manager applications in the market. Some are paid managers while others are free. Even banks like ICICI offer their customers expense tracker to help them out. Before you decide to go in for a money manager, it is important to decide the type you want.

2. LITERATURE SURVEY

2.1 Existing problem

In a study conducted by Forrester in 2016 surveying small and medium businesses (SMBs) across the world, 56% companies reported expense management as being the biggest challenge for their finance departments.

In another survey conducted by Levvel Research in 2018 in North America, respondents reported the following pain points in expense management before adopting automation:

- Manual entry and routing of expense reports (62%)
- Lack of visibility into spend data (42%)
- Inability to enforce travel policies (29%)
- Lost expense reports (24%)
- Lengthy expense approval system and reimbursement cycles (23%)

2.2 References

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
1.	**EXPENSE MANAGER APPLICATION. (2020)**	To Develop A Moblie Application That Keeps Record Of User Personal Expenses Contribution In Group Expenditure Top Investment Options View Of The Current Stock Market ,Read Authenticated Financial News	Android Studio	Cloud Application	Advantages: ➢ Keeps Track All Of Your Daily Transactions, Keeps Track Of Your Money Lent Or Borrowed. Disadvantages: ➢ Occupy Lot Of Space.
2.	**A NOVEL EXPENSE TRACKER USING STATISTICAL ANALYSIS. (2021)**	To Maintain And Manage Data Of Daily Expenditure In A More Precise Way.	SQL Lite	Cloud Application	Advantages: ➢ Its Suggest You With The Most Effective Investment Options. Disadvantages: ➢ The Work Done Being Is Not Accurate.

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
3.	**EXPENSE TRACKER. (2021)**	Facilitates The User To Keep Track And Manage Their Personal As Well As Business Expenses.	Android OS	Cloud Application	Advantages: ➢ Become Aware Of Poor Spending Habits And Take Care Of Your Finances Saving And Investment. Disadvantages: ➢ Searching And Referencing Is Difficult And Time-consuming.
4.	**EXPENSE TRACKER. (May 2021)**	The Application Keeps The Track Of The Income And Expenses Both Of User On A Day To Day Bases	Java	Cloud Application	Advantages: ➢ The Project Effectively Keeps Away From The Manual Figuring. Disadvantages: ➢ Report Generation Is A Tedious Process.

2.3 Problem Statement Definition

Customer Problem Statement :

A well-articulated customer problem statement allows us to find the ideal solution for the challenges our 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

Personal Expense Tracker Application:

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	an employee.	Make a monthly budget.	There are no facilities to set a budget.	I need to save money for my future plans.	Frustrated.
PS-2	A manager.	Keep track of my expenses.	Can't categorize the various types of expenses.	There is no option to organize the various expenses.	Uncomfortable.

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

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

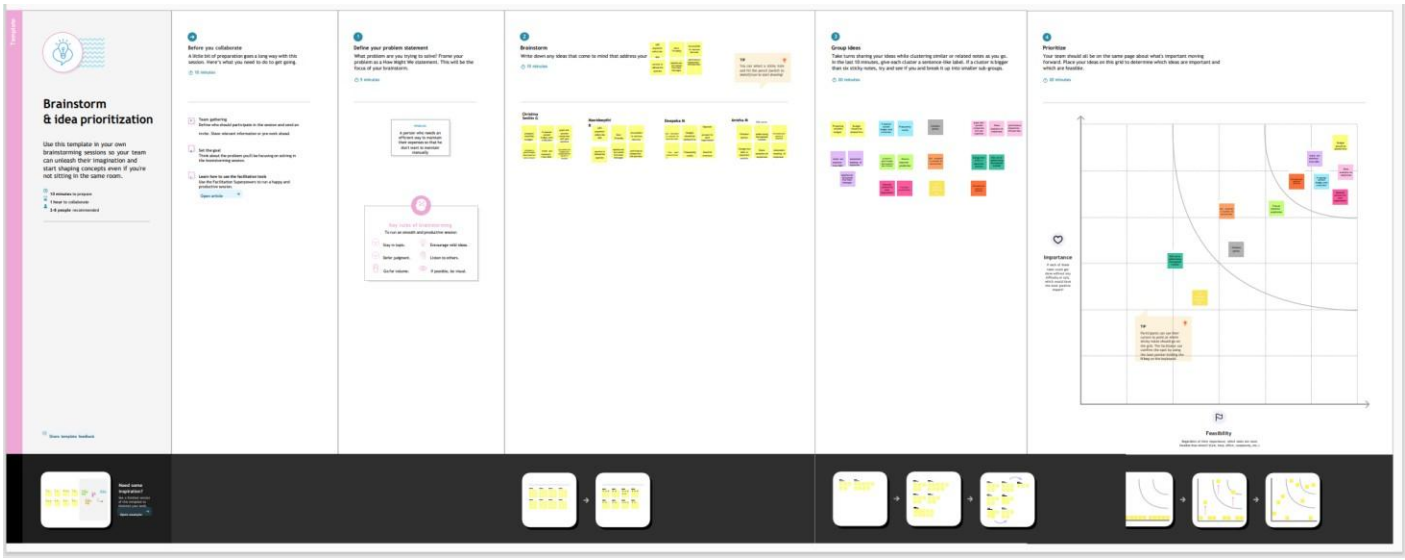
It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

3.2 Ideation & Brainstorming

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

Step-2 : Brainstorm,Idea Listing and Grouping

Step-3: Idea Prioritization



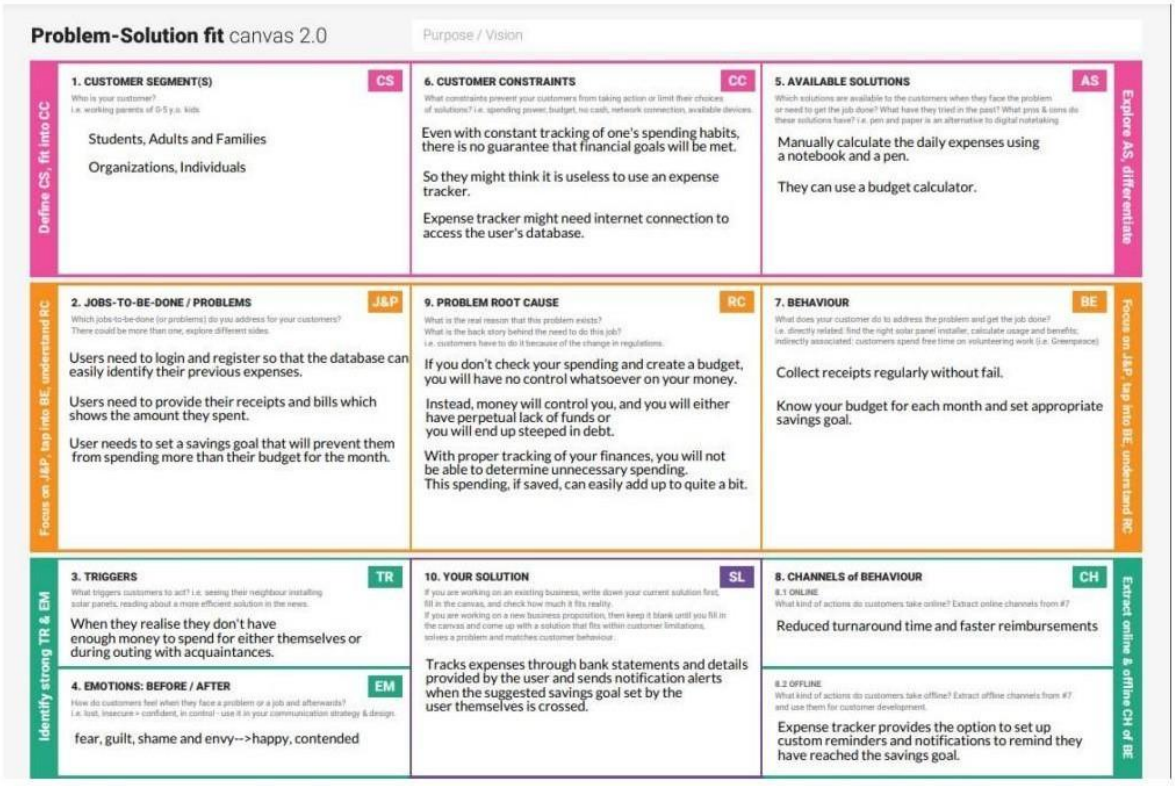
3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The problem of the current generation population is that they can't remember where all of the money they earned have gone and ultimately have to live while sustaining the little money they have left for their essential needs.
2.	Idea / Solution description	Tracks expenses through bank statements and details provided by the user and sends notification alerts when the suggested savings goal set by the user themselves is crossed.
3.	Novelty / Uniqueness	Basic finance tips can be given to users and also a plan to let users earn a small amount of money through ads etc.
4.	Social Impact / Customer Satisfaction	This helps the user to help them from issues like bankruptcy and save time from manual calculations.Also providing a proper balance

		between income and expense is a must for a comfortable livelihood.
5.	Business Model (Revenue Model)	Expense tracker follows service business model which uses Flask, Docker, Cloud and Kubernetes.
6.	Scalability of the Solution	Expense tracker is infinitely scalable since it is based on a per employee per month basis, this means that we'll never pay for capacity you don't need.

3.4 Problem Solution fit

Template:



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

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 E-mail Alert through SMS
FR-6	User Budget Plan	Planning and Tracking of user expense vs budget limit

4.2 Non-Functional requirements

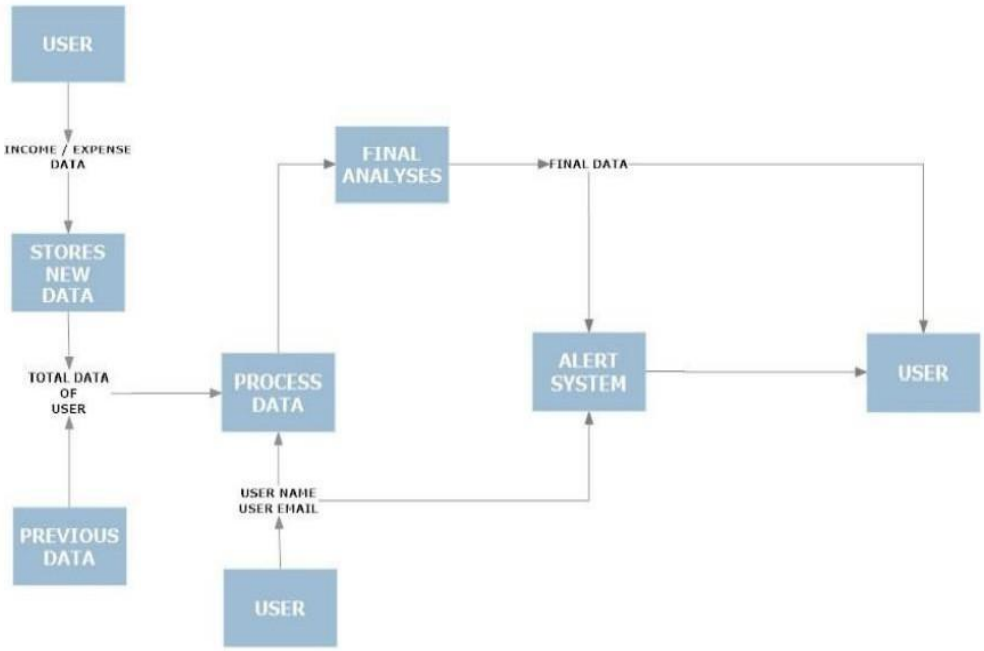
Non-functional Requirements:

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 while interacting with our application.
NFR-2	**Security**	Authentication, authorization, 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.

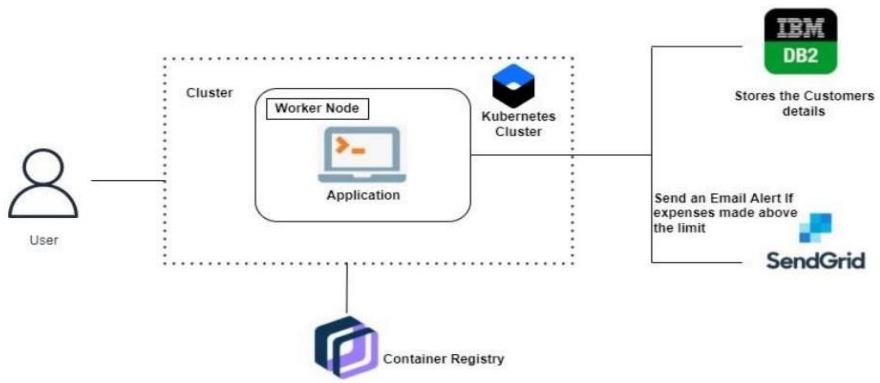
5. PROJECT DESIGN

5.1 Data Flow Diagrams

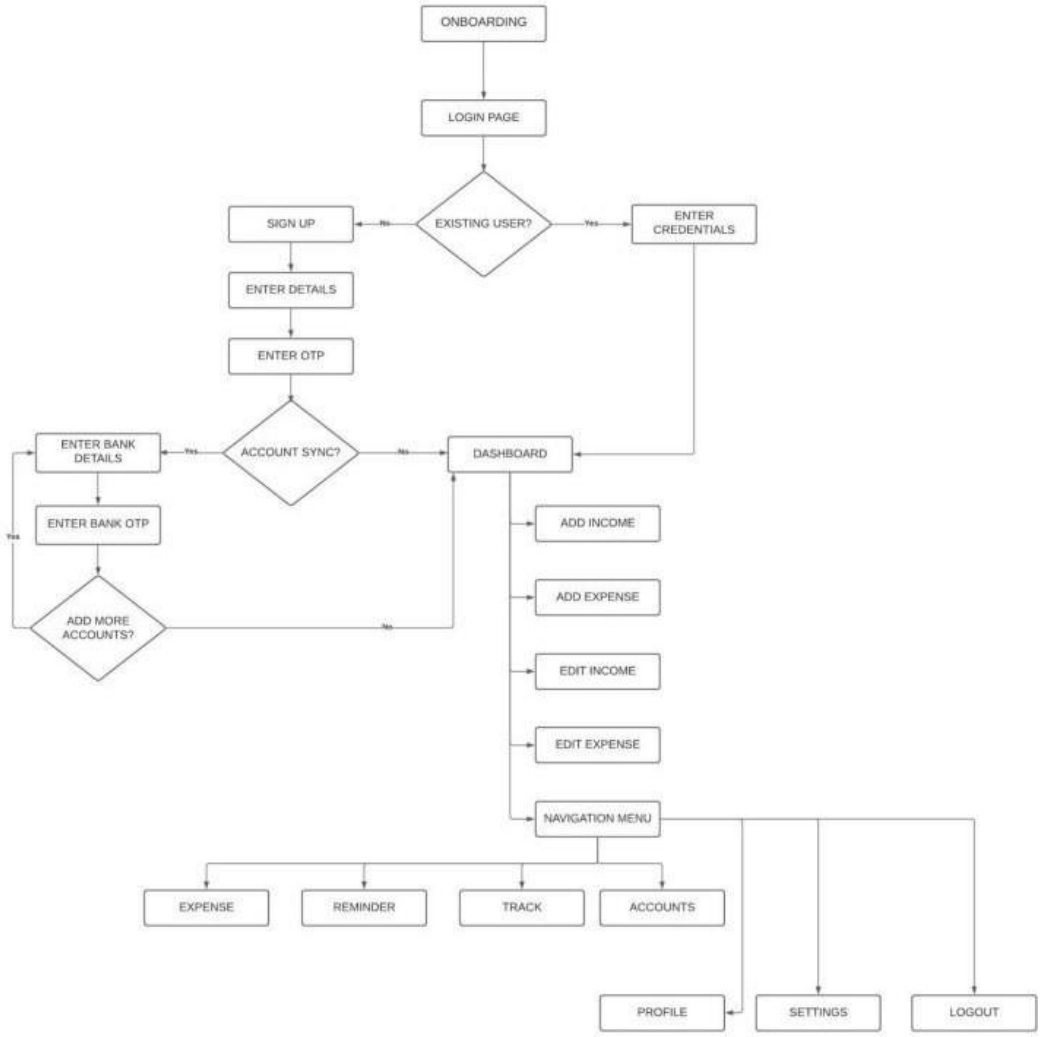


5.2 Solution & Technical Architecture

Technical Architecture:



Solution architecture:



5.3 User Stories

User Stories

Use the below template to list all the user stories for the product.

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, 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 a administrator I can upgrade or update the application.	I can fix the bug which arises for the customers and users of the application	Medium	

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule.

Sprint	Functional Requirements (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
S-1	User Panel	USN-1	The user will access the website and view the productsit provides after registering in.	20	High	DHARUN D DEEPAK KUMAR M HARSVARDHANH P JEEVAN S KARTHIK GIRIVAS V R
S-2	Admin panel	USN-2	The administrator's task is to look over the stock database and monitor on everything that people are buying.	20	High	DHARUN D DEEPAK KUMAR M HARSVARDHANH P JEEVAN S KARTHIK GIRIVAS V R
S-3	Chat Bot	USN-3	The user can directly talk to Chatbot regarding the products. Get the recommendations based on information provided by the user.	20	High	DHARUN D DEEPAK KUMAR M HARSVARDHANH P JEEVAN S KARTHIK GIRIVAS V R
S-4	final delivery	USN-4	Container of applications using docker kubernetes and deployment the application. Create the documentation and final submit the application	20	High	DHARUN D DEEPAK KUMAR M HARSVARDHANH P JEEVAN S KARTHIK GIRIVAS V R

Project Tracker, Velocity & Burndown Chart: (4 Marks)

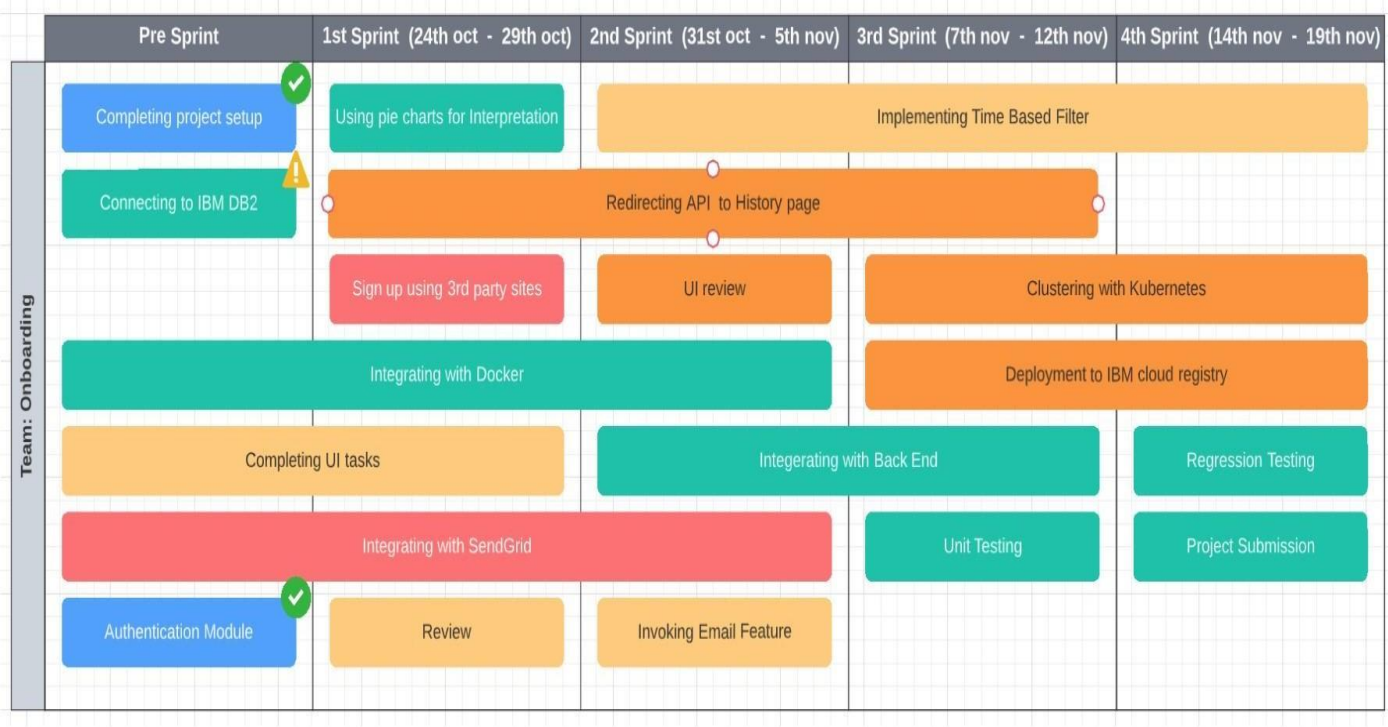
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	16	6 Days	24 Oct 2022	29 Oct 2022	16	29 Oct 2022
Sprint-2	12	6 Days	31 Oct 2022	05 Nov 2022	10	05 Nov 2022
Sprint-3	14	6 Days	07 Nov 2022	12 Nov 2022	13	12 Nov 2022
Sprint-4	14	6 Days	14 Nov 2022	19 Nov 2022	13	19 Nov 2022

Velocity:

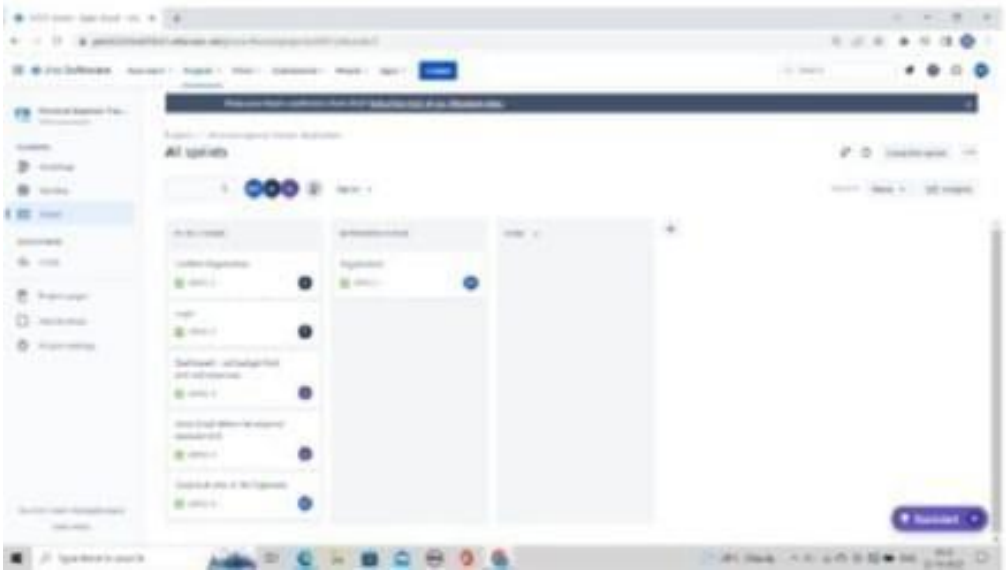
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

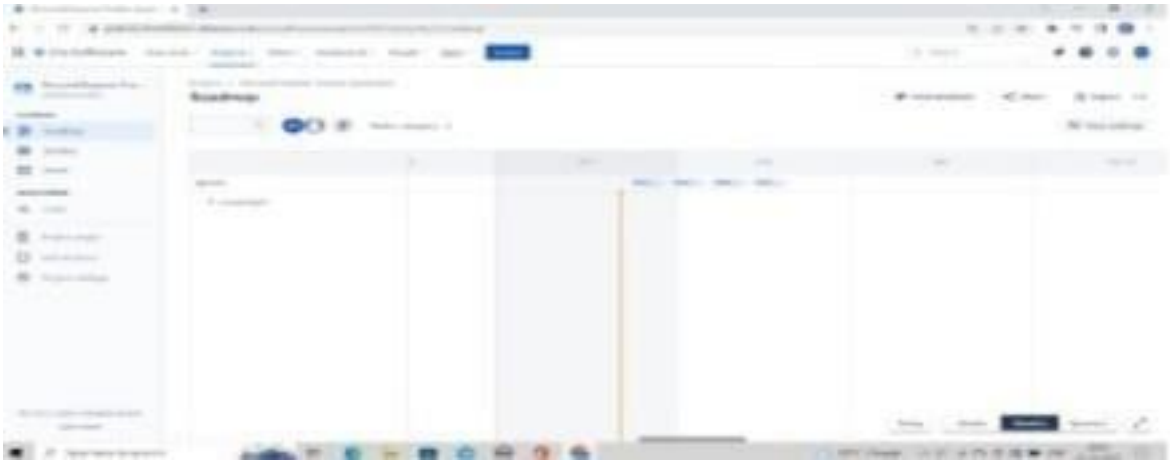
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.2 Sprint Delivery Schedule



6.3 Reports from JIRA





7.CODING & SOLUTIONING

7.1 Feature 1

We have added the data visualization on methods for expenditure. The pie chart have been used to represent the monthly expenses. The pie chart is a pictorial representation of data that makes it possible to visualize the relationships between the parts and the whole of a variable. For example, it is possible to understand the industry count or percentage of a variable level from the
division by areas or sectors. The recommended use for pie charts is two-dimensional, as three-dimensional use can be confusing.

The dimensions form sectors of the measurement values; they can have one or two sizes and up to two measures. The first dimension is used to define the angle of each sector that makes up the chart and the second dimension optionally determines the radius of each sector. Additionally, these plots are useful for comparing data over a fixed period since they do not show changes over time. Therefore, their use should be considered if:

- You are looking to categorize and compare a set of data.
- You only have positive values.
- You have less than seven categories since a larger number can make it difficult to perceive each segment.

CODE :

todayExpenses.html & app.py

```
1{% extends 'base.html' %}
2{% block body %}
3<div class="container ">
4<div class="row">
5                                   <div class="col-md-5">
6                                   <h3 class="mt-5">Today Expense
   Breakdown</h3>
7                                   <div class="card shadow mb-2 bg-
                                    white rounded-pill">
8                                   <div class="card-body ">
9                                   <div class="row">
10                                  <div class="col-md-
   6">TIME</div>
11                                  <div class="col-md-6">
   AMOUNT   </div>
12                                  </div>
```

```
13                  </div>
14                  </div>
15                  {% for row in texpense %}
16                  <div class="card shadow mb-2
                    bg-white rounded-bottom">
17                  <div class="card-body ">
18                  <div class="row">
19                  <div id ="ttime" class="col-md-
                    6">{{row [0]}}</div>
20                  <div id="tamount" class="col-
                    md-6">   {{row[1] }}
</div>
21                  </div>
22                  </div>
23                  </div>
24                  {% endfor %}
25</div>
26</div>
```

<section>

```
28                                  <div class="row">
29                                  <div class="col-md-6">
30                                  <h3 class="mt-5">Expense
   Breakdown BY Category</h3>
31                                  <div class="card shadow
                                    mb-
   2 bg-white rounded-bottom">
32                                  <div class="card-body ">
33                                  <div class="row">
34                                  <div class="col-md-
   6">Food</div>
35                                  <div id="tfood" class="col-
                                    md-6">   {{ t_food}}
                                    </div>
36                                  </div>
37                                  </div>
38                                  </div>
39 <div class="card shadow mb-2
bg-white rounded">
```

```
40                          <div class="card-body">
41                          <div class="row">
42                          <div class="col-md-
  6">Entertainment</div>
43                          <div id="tentertainment"
                            class="col-md-6">
  {{ t_entertainment}}  </div>
44                          </div>
45                          </div>
46                          </div>
47                          <div class="card shadow mb-2
                            bg-white rounded">
48                          <div class="card-body">
49                          <div class="row">
50                          <div class="col-md-
  6">Business</div>
51                          <div id="tbusiness"
                            class="col-md-6">
                            {{t_business}}
```

```
52                          </div>
53                          </div>
54                          </div>
55                          <div class="card shadow mb-2
                            bg-white rounded">
56                          <div class="card-body">
57                          <div class="row">
58                          <div  class="col-md-
   6">Rent</div>
59                          <div id="trent" class="col-md-
                            6"> {{  t_rent }}  </div>
60                          </div>
61                          </div>
62                          </div>
63                          <div class="card shadow mb-2
                            bg-white rounded">
64                          <div class="card-body">
65                          <div class="row">
66                          <div class="col-md-
```

```
   6">EMI</div>
67                          <div id="temi" class="col-md-
                            6">{{ t_EMI }}  </div>
68                          </div>
69                          </div>
70                          </div>
71                          <div class="card shadow mb-2
                            bg-white rounded">
72                          <div class="card-body">
73                          <div class="row">
74                          <div class="col-md-
   6">Other</div>
75                          <div id="tother" class="col-md-
                            6"> {{ t_other}}</div>
76                          </div>
77                          </div>
78                          </div>
79                          <div class="card shadow mb-2
                            btn-outline-danger rounded-
                            pill">
```

```
80                          <div class="card-body">
81                          <div class="row">
82                          <div class="col-
                            md6">Total</div>
83                          <div class="col-md-
   6">₹ {{total}}  </div>
84                          </div>
85                          </div>
86                          </div>
87                          </div>
88                          <div class="col-md-6">
89                          <canvas id="myChart"
                            width="400"
                            height="400"></canvas>
90                          <script>
91                          let food =
                            document.getElementById('tfoo
                            d').innerHT ML
92                          let entertainment =
                            document.getElementById('tent
                            ertainment'
```

```
    ).innerHTML
93                  let business =
                    document.getElementById('tbusi
                    ness').inn erHTML
94                  let rent =
                    document.getElementById('trent'
                    ).innerHT ML
95                  let emi =
                    document.getElementById('temi')
                    .innerHT ML
96                  let other =
                    document.getElementById('tothe
                    r').innerH TML
97                  var ctx =
                    document.getElementById('myCh
                    art').getCo ntext('2d');
98                  var myChart = new Chart(ctx, {
99                  type: 'doughnut',
100                 data: {
```

```
101                                     labels: ['Food',
                'Entertainment', 'Business', 'Rent',
    'EMI', 'Other'],
102                                     datasets: [{
103                                     label:
    'Expenses Chart',
104                                     data: [food,
                                        entertainment,
                                        business, rent, emi,
                                        other],
105                                     backgroundColor: [
106                                     'rgb(255, 99, 132)',
107                                     'rgb(0, 0, 0)',
108                                     'rgb(255, 205, 86)',
109                                     'rgb(201, 203,
    207)',
```

```
110                                 'rgb(54,
   162, 235)',
111                                 'rgb(215, 159,
                                    64)'
112                                 ],
113                                 }]
114                                 },
115                                 options: {
116                                 responsive: true,
117                                 plugins: {
118                                 legend: {
119                                 position:
                                    'bottom',
120                                 },
121                                 title: {
122                                 display: true,
123                                 text: 'EXPENSE
                                    BREAKDOWN'
124                                 }
125                                 }
126                                 }
```

```
127                 });
128                 </script>
129             </div>
130         </div>
131     </div>
132 </section>
133 </div>
134 {% endblock %}
```

```
1  @app.route("/today")
2  def today():
3
4          param1 = "SELECT TIME(date) as tn, amount FROM expenses
   WHERE userid = " + str(session['id']) + " AND DATE(date) =
   DATE(current timestamp) ORDER BY date DESC"
5          res1 = ibm_db.exec_immediate(ibm_db_conn, param1)
6          dictionary1 = ibm_db.fetch_assoc(res1)
7          texpense = []
8
9              while dictionary1 != False:
10             temp = []
11             temp.append(dictionary1["TN"])
```

```
12      temp.append(dictionary1["AMOUNT"])
13      texpense.append(temp)
14      print(temp)
15      dictionary1 = ibm_db.fetch_assoc(res1) 16
17              param = "SELECT * FROM expenses WHERE userid = " + str(session['id']) + " AND
                DATE(date) = DATE(current timestamp) ORDER BY date DESC"
18              res = ibm_db.exec_immediate(ibm_db_conn, param)
19              dictionary = ibm_db.fetch_assoc(res)
20              expense = []
21              while dictionary != False:
22              temp = []
23              temp.append(dictionary["ID"])
24              temp.append(dictionary["USERID"])
25              temp.append(dictionary["DATE"])
26              temp.append(dictionary["EXPENSENAME"])
27              temp.append(dictionary["AMOUNT"])
28              temp.append(dictionary["PAYMODE"])
29              temp.append(dictionary["CATEGORY"])
30              expense.append(temp)
31              print(temp)
32              dictionary = ibm_db.fetch_assoc(res) 33
34          total=0
35          t_food=0
36          t_entertainment=0
37          t_business=0
38          t_rent=0
39          t_EMI=0
40          t_other=0
41
```

```
42        for x in expense:
43            total += x[4]
44            if x[6] == "food":
45                t_food += x[4]
46            elif x[6] == "entertainment":
47                t_entertainment  += x[4]
48            elif x[6] == "business":
49                t_business  += x[4]
50            elif x[6] == "rent":
51                t_rent  += x[4]
52            elif x[6] == "EMI":
53                t_EMI  += x[4]
54            elif x[6] == "other":
55                t_other  += x[4]
56        print(total)
57        print(t_food)
58        print(t_entertainment)
59        print(t_business)
60        print(t_rent)
61        print(t_EMI)
62        print(t_other)
63
64
65        return render_template("today.html", texpense = texpense,
  expense = expense,  total = total ,
66                               t_food = t_food,t_entertainment =
  t_entertainment,
67                               t_business = t_business,  t_rent =
  t_rent,
68                               t_EMI =  t_EMI,  t_other =  t_other )
```

7.2 Feature 2

Email notifications will be sent to the users once they cross the expenditure limit through send grid mail system. Most notifications are transactional, meaning a recipient's action or account activity triggers them. But some notifications are marketing related, encouraging the recipient to take a specific action. Ecommerce product notifications inform recipients about new products or discounts. Plus, unlike general marketing emails, these are highly personalized and focus on a single product. For example, if a customer views an item on your website and that item goes on sale, you can send the customer a notification to let them know this is the best time to buy. Users can also opt into receiving notifications when an out-of-stock item is back in stock.

Notification emails tend to perform well because the content is highly relevant to the recipient. But the only way for the recipient to know this is if you state the content clearly in the subject line.

For example, the subject line "New Sign-in to Your Account" gets straight to the point, letting the user know why you sent this notification.

sendemail.py

```
import  smtplib
import sendgrid as sg
import os
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail, Email, To, Content
SUBJECT = "personal expense tracker"
s = smtplib.SMTP('smtp.gmail.com', 587)

def sendmail(TEXT,email):
    from_email = Email("tour7107@gmail.com")
    to_email = To(email)
    subject = "Sending with SendGrid is Fun"
    content = Content("text/plain",TEXT)
```

```
    mail = Mail(from_email, to_email, subject, content)
    try:
        sg=SendGridAPIClient('SG.PJq2l00SRKyfdCTfm5b4XQ.I5p737Yx2imRbcRxykMCtRlq1
iZ7s0_cTEDwQbm8nI4')
        response = sg.send(mail)
        print(response.status_code)
        print(response.body)
        print(response.headers)
    except Exception as e:
        print(e)
```

7.3 Database Schema

Tables :

1) Admin

id INT NOT NULL GENERATED ALWAYS AS IDENTITY,username VARCHAR(32) NOT NULL, email VARCHAR(32) NOT NULL,

password VARCHAR(32) NOT NULL

2) EXPENSE

id INT NOT NULL GENERATED ALWAYS AS IDENTITY, userid INT NOT NULL, date TIMESTAMP(12) NOT NULL,expensename VARCHAR(32) NOT NULL,

amount VARCHAR(32) NOT NULL,

paymode VARCHAR(32) NOT NULL,

category VARCHAR(32) NOT NULL

3) LIMIT

id INT NOT NULL GENERATED ALWAYS AS IDENTITY,userid VARCHAR(32) NOT NULL,

limit VARCHAR(32) NOT NULL

8.TESTING

8.1 Test Cases

Test Case ID	Purpose	TestCases	Result
TC1	Authentication	Password with length less than 4 characters	Password cannot be less than 4 characters
TC2	Authentication	User name with length less than 2 characters	User name cannot be less than 2 characters
TC3	Authentication	Valid user name with minimum 2 characters	User name accepted

TC4	Authentication	User name le blank	User name cannot be less than 2 characters
TC5	Authentication	Password field le blank	Password cannot be empty
TC6	Authentication	Minimum 4 characters valid password	Password accepted
TC7	Authentication	Password and Confirm Password did not match	Please enter same password
TC8	Authentication	Confirm Password field le blank	Please enter same password

8.2 User Acceptance Testing

Technical Requirment Document (TSD)	
Test Case ID	**Test Case Description**
TC_001	Verify if user is able to order single product.
TC_002	Verify if user is able to order multiple products.
TC_003	Verify if user can apply single or multiple filters
TC_004	Verify if user can apply different sort by
TC_005	Verify if user is able to pay by Master Card
TC_006	Verify if user is able to pay by Debit Card
TC_007	Verify if user is able to pay fully by reward points
TC_008	Verify if user is able to pay partially by reward points

9.RESULTS

9.1 Performance Metrics

- Tracking income and expenses: Monitoring the income and tracking all expenditures (through bank accounts, mobile wallets, and credit & debit cards).
- Transaction Receipts: Capture and organize your payment receipts to keep track of your expenditure.
- Organizing Taxes: Import your documents to the expense tracking app, and it will streamline your income and expenses under the appropriate tax categories.
- Payments & Invoices: Accept and pay from credit cards, debit cards, net banking, mobile wallets, and bank transfers, and track the status of your invoices and bills in the mobile app itself. Also, the tracking app sends reminders for payments and automatically matches the payments with invoices.
- Reports: The expense tracking app generates and sends reports to give a detailed insight about profits, losses, budgets, income, balance sheets, etc.,
- E-commerce integration: Integrate your expense tracking app with your eCommerce store and track your sales through payments received via multiple payment methods.
- Vendors and Contractors: Manage and track all the payments to the vendors and contractors added to the mobile app.
- Access control: Increase your team productivity by providing access control to particular users through custom permissions.
- Track Projects: Determine project profitability by tracking labor costs, payroll, expenses, etc., of your ongoing project.

- Inventory tracking: An expense tracking app can do it all. Right from tracking products or the cost of goods, sending alert notifications when the product is running out of stock or the product is not selling, to purchase orders.
- In-depth insights and analytics: Provides in-built tools to generate reports with easy-to-understand visuals and graphics to gain insights about the performance of your business.
- Recurrent Expenses: Rely on your budgeting app to track, streamline, and automate all the recurrent expenses and remind you on a timely basis.

10. ADVANTAGES & DISADVANTAGES

1. **Achieve your business goals** with a tailored mobile app that perfectly fits your business.
2. **Scale-up** at the pace your business is growing.
3. Deliver an **outstanding** customer experience through additional control over the app.
4. Control the **security** of your business and customer data
5. Open **direct marketing channels** with no extra costs with methods such as push notifications.
6. **Boost the productivity** of all the processes within the organization.
7. Increase **efficiency** and **customer satisfaction** with an app aligned to their needs.
8. **Seamlessly integrate** with existing infrastructure.
9. Ability to provide **valuable insights**.
10. Optimize sales processes to generate **more revenue** through enhanced data collection.

11. CONCLUSION

From this project, we are able to manage and keep tracking the daily expenses as well as income. While making this project, we gained a lot of experience of working as a team. We discovered various predicted and unpredicted problems and we enjoyed a lot solving them as a team. We adopted things like video tutorials, text tutorials, internet and learning materials to make our project complete.

12. FUTURE

The project assists well to record the income and expenses in general.
However, this project has some limitations:

- The application is unable to maintain the backup of data once it is uninstalled.
- This application does not provide higher decision capability.

To further enhance the capability of this application, we recommend the following features to be incorporated into the system:

- Multiple language interface.
- Provide backup and recovery of data.
- Provide better user interface for user.
- Mobile apps advantage.

13. APPENDIX

Source Code Github Link :

https://github.com/IBM-EPBL/IBM-Project-10247-1659151056