

Personal Expense Tracker Application **IBM-Project-34741-1660275861**

PERSONAL EXPENSE TRACKER APPLICATION

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

A PROJECT REPORT

AISHWARYA T (962819104007)

ARUNA K R (962819104018)

ASHIKA S J (962819104019)

JENISHA L (962819104047)

**BACHELOR OF ENGINEERING IN COMPUTER SCIENCE AND
ENGINEERING**

University College of Engineering

Nagercoil – 629 004

Personal Expense Tracker Application

IBM-Project-34741-1660275861

INDEX

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
3. Reports from JIRA

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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

1. INTRODUCTION

1.1 Project Overview

Category : Cloud App Development

Team ID : PNT2022TMID34843

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.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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%)

Personal Expense Tracker Application

IBM-Project-34741-1660275861

2.2 References

1. Expense Tracker Aman Garg , Mukul Goel , Sagar Mittal , Mr. Shekhar Singh
International Journal for Research in Applied Science & Engineering Technology
(IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429 Volume 9 Issue IV
Apr 2021

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. He can see the accurate duration for how long a particular product is being used by him. The monthly, and year-wise comparison of expenditures will be done by the app which will let the user know the area where he is spending the most. The user will be able to see the detailed analyses with the help of graphical visualizations. This project will provide a lot of benefits to the users with the help of which they will be surely able to keep track of each penny. It is time to stop using paper and excel sheets to keep track of your digital as well as cash payments. Using paper is not easy to manage. It is common to delete files accidentally or misplace files. This expense tracker provides a complete digital solution to this problem. Excel sheets do very little to help in tracking expenses. Furthermore, they don't have the advanced

functionality of preparing graphical visuals automatically. Not only it will save the time

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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. The modules are developed efficiently and also in an attractive manner. The application will eliminate sticky notes, spreadsheets, and ledgers that cause confusion, data inconsistency problems while recording and splitting expenses. With our application users can manage their expenses more effectively and they will be better at managing the expenses.

2. EXPENSE TRACKER APPLICATION R.Velmurugan ,Mrs.P.Usha An International Open Access , Peer-reviewed, Refereed Journal Unique Paper ID: 150860 Publication Volume & Issue: Volume 7, Issue 10 ISSN:2349-6002 ESTD Year:2014

Expense tracker is an android based application. This application allows the user to maintain a computerized diary. Expense tracker application which will keep a track of Expenses of a user on a day-to-day basis. This application keeps a record of your expenses and also will give you a category wise distribution of your expenses. With the help of this application user can track their daily/weekly/monthly expenses. This application will also has a feature which will help you stay on budget because you know your expenses. Expense tracker application will generate report at the end of month to show Expense via a graphical representation. We also have added a special feature which will distributes your expenses in different categories suitable for the user. An expense history will also be provided in application.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

3.Expense Manager Application A Velmurugan Albert Mayan J , Niranjana P and Richard Francis et al 2020 J. Phys.: Conf. Ser. 1712 012039 Associate Professor, School of Computing, Sathyabama Institute of Science and Technology, Chennai. U.G Student, Department of CSE, Sathyabama Institute of Science and Technology, Chennai

We have developed a mobile application that Keeps track of all of your daily transactions, keeps track of your money lent or borrowed ,suggests you with the most effective investment options, offers your discounts in popular categories , view exchange and to read latest authenticated financial news. This paper's main aim to eliminate the use of sticky notes, spreadsheets and handling of large chunks of data is successful, the new experience is hasslefree and very handy. Now, with our application user can manage his expenses more effectively. This application can also help digital marketing agencies in rolling out their advertising campaigns more effectively. As a part of further research, we considered adding certain features to create more enhanced experience to the user .We are also going to link this profile with their mobile number, email account, social networks so that the application offers portability, other features to be added are discussed above below within the future enhancement section. The application delivered efficiently in calculating split expenses and recording the expenses together accurately with date and time.

4.Expense Tracker Author(s): Praphulla S. Kherade ; Raj S. Vilankar ; Parag M. Sawant ; Atiya Kazi Paper ID: 1702687 Published Date: 04-05-2021 Published In: Iconic Research And Engineering Journals Publisher: IRE Journals e-ISSN: 2456-8880 Volume/Issue: Volume 4 Issue 11 May-2021.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

We are building an android application named as "Expense Tracker". As the name suggests, this project is an android app which is used to track the daily expenses of the user. It is like digital record keeping which keeps the records of expenses done by an user. The application keeps the track of the Income and Expenses both of user on a day-to-day basis. This application takes the income of an user and manage its daily expenses so that the user can save money. If you exceed daily expense allowed amount it will give you a warning, so that you don't spend much and that specific day. If you spend less money than the daily expense allowed amount, the money left after spending is added into user's savings. The application generates report of the expenses of each end of the month. The amount saved can be used for celebrating festivals, Birthdays or Anniversary.

2.3 Problem Statement Definition

Customer Problem Statement :

Personal expense tracker application helps the user to keep track of expenses in day- to- day basis and display a graphical representation at the end of week, month and year. So the user can set a financial goal for a future needs.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



Personal Expense Tracker Application

IBM-Project-34741-1660275861

3.2 Ideation & Brainstorming

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



Personal Expense Tracker Application

IBM-Project-34741-1660275861

Step-2 : Brainstorming, Idea Listening and Grouping

The image shows a screenshot of a digital brainstorming tool interface, divided into two main panels: "Brainstorming" and "Group Ideas".

Brainstorming Panel:

- Header:** "Brainstorm" with a sub-instruction: "Write down any ideas that come to mind that address your problem statement." and a timer icon indicating "45 minutes".
- Content:** Four columns of sticky notes are visible, labeled "Ideas 1", "Ideas 2", "Ideas 3", and "Ideas 4". Each column contains several yellow sticky notes with text.
- Footer:** Two icons representing different brainstorming layouts (a grid and a free-form layout).

Group Ideas Panel:

- Header:** "Group Ideas" with a sub-instruction: "Take turns sharing your ideas while clustering similar or related notes as you go. In the last 15 minutes, give each cluster a name (or use labels). If a cluster is larger than six sticky notes, try and see if you can break it up into smaller sub-groups." and a timer icon indicating "20 minutes".
- Content:** A collection of sticky notes (yellow, orange, pink, and blue) are arranged in a hierarchical or clustered manner on a light gray background. Some notes are connected by lines, indicating relationships or groupings.
- Footer:** Three icons representing different grouping or clustering methods (a single note, a group of notes, and a group of notes with a circular arrow indicating a process).

Personal Expense Tracker Application

IBM-Project-34741-1660275861

Step-3 Ideation Prioritization



Personal Expense Tracker Application

IBM-Project-34741-1660275861

3.3 Proposed Solution

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Personal expense tracker application helps the user to keep track of expenses in day-to-day basis and display a graphical representation at the end of every week, month and year. So the user can set a financial goal for their future needs.
2.	Idea / Solution description	To manage Personal expenses in a more efficient and manageable way. We decided to find an easier way to get rid of this problem. So, our application attempts to free the user with as much as possible the burden of manual calculation and to keep the track of the expenditure. Instead of keeping a diary or a log of the expenses, this application enables the user to not just keep the control on the expenses but also to generate and save reports. With the help of this application, the user can manage their expenses on a daily, weekly and monthly basis as well as can generate and save their reports.
3.	Novelty / Uniqueness	A new feature of suggestions given based on the user's interest. Budget calculator helps to manage expense effectively and the status is displayed in percentage. User can track their expense by mentioning day, month and year.
4.	Social Impact / Customer Satisfaction	The main reason to track expenses is to identify and eliminate wasteful spending habits in your financial life. Moreover, consistently tracking your expenses will helps to maintain control of finances, and promote better financial habits like saving and investing. 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 on a daily, weekly, and monthly basis.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

5.	Business Model (Revenue Model)	Many expense tracker apps are free or low cost, so explore your options and decide if you need a paid app. These apps can help you gain control of your finances, pay down debt, and grow your net worth. Some even allow you to upload receipts, track mileage, and send invoices, which are all helpful if you run a business.
6.	Scalability of the Solution	Additional features can be added. Expenses can be customised in categories. Any country currency is supported.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

3.4 Problem Solution fit

Define CS, fit into CC RC	1. CUSTOMER SEGMENT(S) CS Man, who needs to manage finances effectively.	6. CUSTOMER CONSTRAINTS CC No control over spending, not reviewing financial plan regularly, lack of priorities, not saving enough.	5. AVAILABLE SOLUTIONS AS When Expense is out of control, The Personal Expense tracker is the solution to control the Expenses according to their needs as this application track the expense & generate the monthly Expense report.	Explore AS, differentiate RC
	2. JOBS-TO-BE-DONE / PROBLEMS J&P Expense Tracker system keep a track of Income Expenses on a day to day basis, this need to take income and divide in daily expense, If exceeds cut if from the income and if expense is less it need to add in the savings. This system needs to generate report at the end of every month to show income Expense curve.	9. PROBLEM ROOT CAUSE RC Not living on a budget, Not knowing the priorities in spending, Lacking a financial purpose.	7. BEHAVIOUR BE Customer should set their budget and spend accordingly. He can make use of management tips given by the system to save his money effectively.	
Identify strong TR & EM	3. TRIGGERS TR Customer finds that at the end of the month they don't have sufficient money to meet their needs which urges them to use Expense tracker.	10. YOUR SOLUTION SL A Personal Expense tracker application, with which will be able to allocate money to different priorities and this will also help to cutdown on unnecessary spending and helps to budget money so that can use it wisely.	8.CHANNELS of BEHAVIOUR CH Customer needs to input their monthly income and expense to manage their finance. Can view their generated report. Can view their Previous spending.	Identify strong TR & EM
	4. EMOTIONS: BEFORE / AFTER EM Before: Less financial security, a higher likelihood of going into dept, and more financial stress. After: More financial security, stress free financial control.			

Personal Expense Tracker Application

IBM-Project-34741-1660275861

4.REQUIREMENTS ANALYSIS

4.1 Functional Requirements

Functional Requirements :

Following are the functional requirements of the proposed solution.

FR NO.	FUNCTIONAL REQUIREMENTS (EPIC)	SUB REQUIREMENT (STORY/SUB-TASK)
FR-1	User Registration	Registration through form. Registration through email.
FR-2	User Confirmation	Confirmation via email. Confirmation via OTP.
FR-3	Monthly budget	Tracking through graphical representation.
FR-4	Input data	Manual filling
FR-5	Notification	Send through email.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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	This application is really usable for the user for their budget tracking. It make the user much easier because the can track their expenditures in day, weekly, monthly wise. Expenditure and the savings are show in a graphical representation.
NFR-2	Security	The personal informations of the users are in a secure manner.
NFR-3	Reliability	The performance of the system would be really good. Probability of giving false information is very low. Because everything are set by a algorithm for the budget calculation.
NFR-4	Performance	The performance mostly depends on the user hand because if the user fill the data correctly the application shows the accurate information about the budget tracking.
NFR-5	Availability	The availability of the solution is effective and it should be helpful for the user.

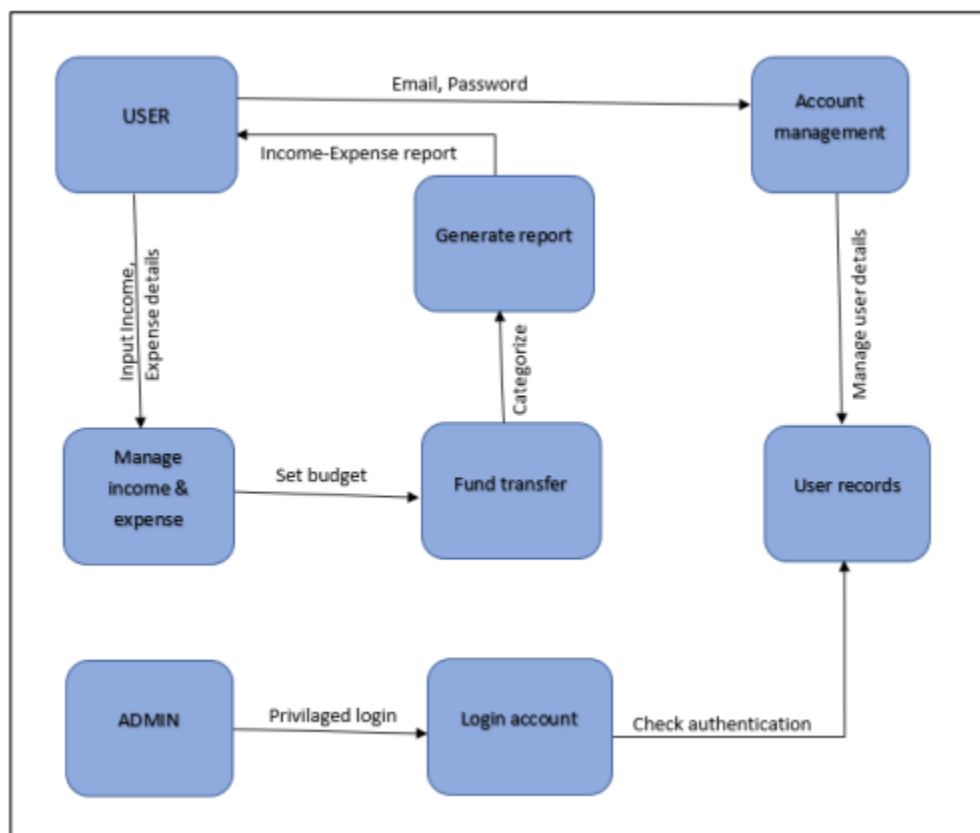
Personal Expense Tracker Application

IBM-Project-34741-1660275861

NFR-6	Scalability	It is scalability which decrease the budget planning of the user.
-------	-------------	---

5.PROJECT DESIGN

5.1 Data Flow Diagrams

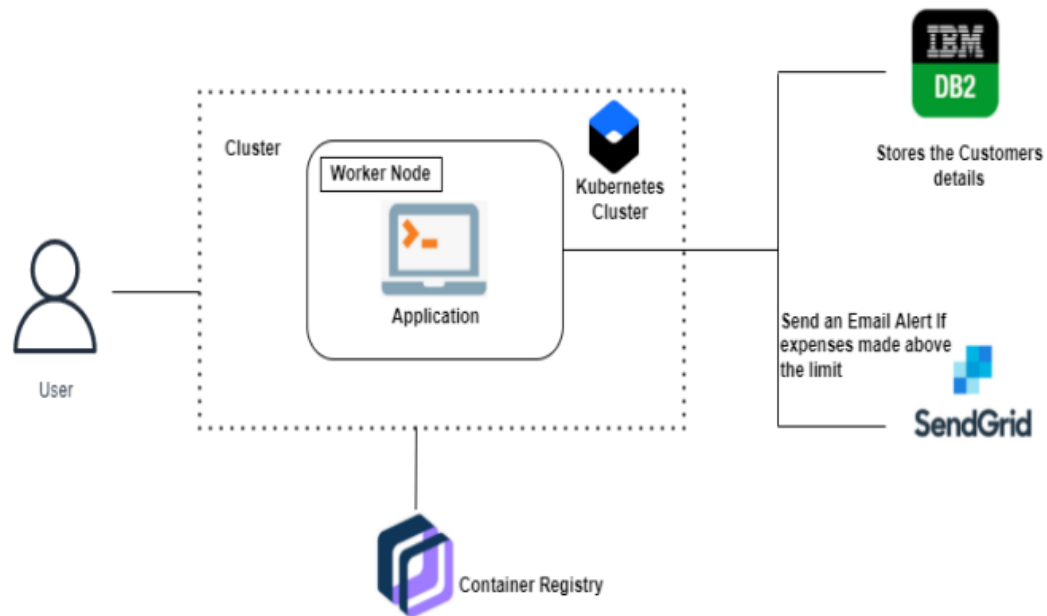


Personal Expense Tracker Application

IBM-Project-34741-1660275861

5.2 Solution & Technical Architecture

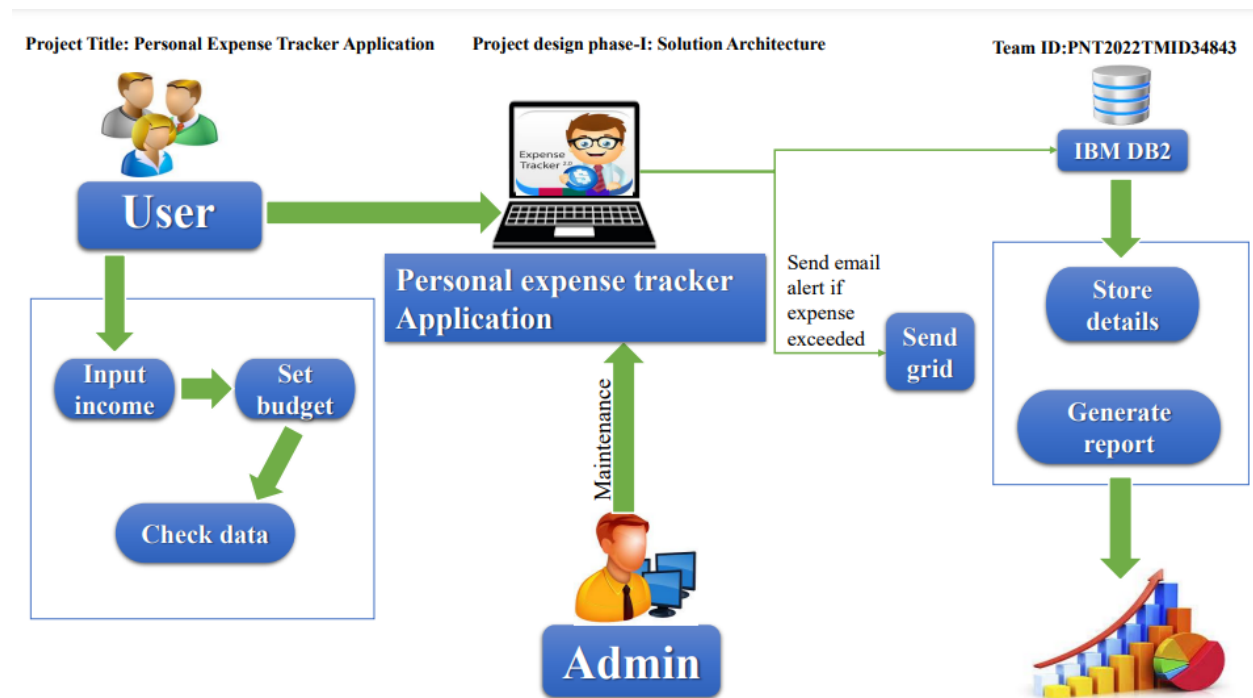
Technical Architecture:



Personal Expense Tracker Application

IBM-Project-34741-1660275861

Solution Architecture



Personal Expense Tracker Application

IBM-Project-34741-1660275861

5.3 Users Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	As a user, I can register for the application by entering my email, password.	I can access my account	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	I can receive confirmation email	High	Sprint-1
	Login	USN-3	As a user, I can log into the application by entering email & password.	I can access my account with Login	High	Sprint-2
		USN-4	As a user, I can give forgot password and verify through my email.	I can login even when password is unknown	Medium	Sprint-2
	Input	USN-5	As a user, I can give my Income and expense details regularly.	I can enter my income, expense.	High	Sprint-3
		USN-6	As a user, I can give my budget limit.	I can set budget	High	Sprint-3
		USN-7	As a user, I can enter my bills.	I can input bills.	Low	Sprint-5
	Dashboard	USN-8	As a user, once I logged in Graphical representation of my expenses is achieved.	I can view my updated dashboard	Medium	Sprint-4
		USN-9	As a user, I can track my expenses daily, monthly.	I can track my expenses.	High	Sprint-4
		USN-10	As a user, I can revisit my expenses on a specific day.	I can revisit my expenses	Low	Sprint-5
	Notifications	USN-11	As a user, I can receive alert messages on exceeding my budget.	I can be notified for exceeding budget.	High	Sprint-3
		USN-12	As a user, I can receive suggestion notifications on saving money.	I can be notified on saving tips.	Medium	Sprint-4
Customer Care Executive	Customer Care	USN-13	As an executive, I can solve the queries and issues regarding the application	I can give 24/7 support..	Low	Sprint-5
Administrator	Application	USN-14	As an administrator, I can upgrade or update the application.	I can fix the bugs and improve the application.	High	Sprint-5

Personal Expense Tracker Application

IBM-Project-34741-1660275861

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

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

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.	2	High	Aishwarya T
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	1	High	Ashika S J
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password.	1	High	Jenisha L
Sprint-1	Dashboard	USN-4	Login takes to the dashboard of the user.	2	High	Aruna K R

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Workspace	USN-1	As a user, I can track my personal expense in workspace.	2	High	Jenisha L
Sprint-2	Graphs	USN-2	As a user, I can view my monthly expenses in graphs.	1	Medium	Aishwarya T
Sprint-2	Connecting to IBM DB2	USN-3	As an admin, Linking database with dashboard.	2	High	Aruna K R
Sprint-2		USN-4	As an admin, Making dashboard interactive with JS.	2	High	Ashika S J
Sprint-3		USN-1	As an admin, wrapping up the server side works of frontend.	1	Medium	Aishwarya T
Sprint-3	Watson assistant	USN-2	Creating Chatbot for clarifying user's query.	1	Medium	Aruna K R
Sprint-3	SendGrid	USN-3	Sending alert mail to users about their expenses using SendGrid.	1	Low	Ashika S J
Sprint-3		USN-4	Integrating both frontend and backend.	2	High	Jenisha L
Sprint-4	Docker	USN-1	Creating image for website using docker.	2	High	Aruna K R
Sprint-4	Cloud Registry	USN-2	Uploading docker image to IBM Cloud registry.	2	High	Aishwarya T
Sprint-4	Kubernetes	USN-3	Creating container using the docker image and hosting the site.	2	High	Jenisha L
Sprint-4	Exposing	USN-4	Exposing IP/Ports for the site	2	High	Ashika S J

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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	20	6 Days	14 Oct 2022	20 Oct 2022	20	21 Oct 2022
Sprint-2	20	6 Days	22 Oct 2022	28 Oct 2022	20	29 Oct 2022
Sprint-3	20	6 Days	30 Oct 2022	05 Nov 2022	20	06 Nov 2022
Sprint-4	20	6 Days	7 Nov 2022	13 Nov 2022	20	14 Nov 2022

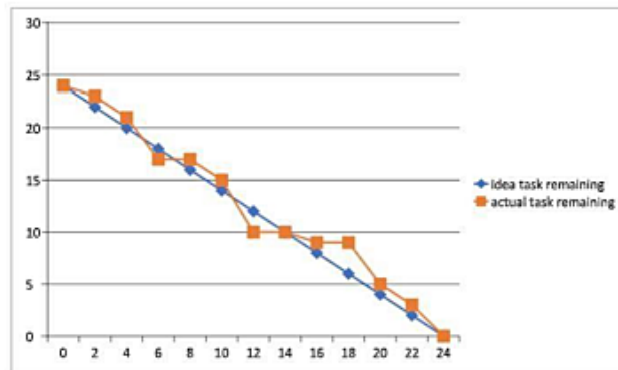
Velocity:

Imagine we have a 6-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)

$$AV = \text{sprint duration} / \text{velocity} = 20/6 = 3.3$$

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.



Personal Expense Tracker Application

IBM-Project-34741-1660275861

6.2 Sprint Delivery Schedule



Personal Expense Tracker Application

IBM-Project-34741-1660275861

6.3 Reports from JIRA

Personal Expense Tracker Application

IBM-Project-34741-1660275861

7.CODING & SOLUTIONING

7.1 Feature 1

We have added the data visualization methods for expenditure. The pie chart has 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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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>
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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

{% endfor %}

```
25</div>
```

```
26</div>
```

```
<section>
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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-
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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}}
```


Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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-
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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('tfoot
d').innerHTML
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```

101             labels: ['Food',
102                     'Entertainment', 'Business', 'Rent',
103                     'EMI', 'Other'],
104
105             datasets: [{
106                 label:
107                     'Expenses Chart',
108
109                 data: [food,
110                       entertainment,
111                       business, rent, emi,
112                       other],
113
114                 backgroundColor: [
115                     'rgb(255, 99, 132)',
116                     'rgb(0, 0, 0)',
117                     'rgb(255, 205, 86)',
118                     'rgb(201, 203,
119                     207)',
120                 ],
121             }],
122         },
123     ],
124     legend: {
125         position: 'bottom',
126     },
127     title: {
128         display: true,
129         text: 'Expenses Chart',
130     },
131     xaxis: {
132         type: 'category',
133         labels: ['Food', 'Entertainment', 'Business', 'Rent', 'EMI', 'Other'],
134     },
135     yaxis: {
136         type: 'number',
137         labels: 'Amount',
138     },
139     plugins: [
140         {
141             id: 'expenses-chart',
142             afterDatasetUpdate: (chart, dataset) => {
143                 // Add labels to the segments
144                 dataset.data.forEach((value, index) => {
145                     chart.getSegment(index).label = dataset.labels[index];
146                 });
147             },
148         },
149     ],
150     options: {
151         // Chart options
152     },
153 });
154
155 // Create the chart
156 const chart = new Chart(ctx, config);
157
158 // Add labels to the segments
159 chart.data.datasets[0].data.forEach((value, index) => {
160     chart.getSegment(index).label = chart.data.datasets[0].labels[index];
161 });
162
163 // Display the chart
164 document.getElementById('expenses-chart').appendChild(chart.canvas);
165
166 // Add a title to the chart
167 chart.title.text = 'Expenses Chart';
168
169 // Add a legend to the chart
170 chart.legend.position = 'bottom';
171
172 // Add a title to the chart
173 chart.title.text = 'Expenses Chart';
174
175 // Add a legend to the chart
176 chart.legend.position = 'bottom';
177
178 // Add a title to the chart
179 chart.title.text = 'Expenses Chart';
180
181 // Add a legend to the chart
182 chart.legend.position = 'bottom';
183
184 // Add a title to the chart
185 chart.title.text = 'Expenses Chart';
186
187 // Add a legend to the chart
188 chart.legend.position = 'bottom';
189
190 // Add a title to the chart
191 chart.title.text = 'Expenses Chart';
192
193 // Add a legend to the chart
194 chart.legend.position = 'bottom';
195
196 // Add a title to the chart
197 chart.title.text = 'Expenses Chart';
198
199 // Add a legend to the chart
200 chart.legend.position = 'bottom';
201
202 // Add a title to the chart
203 chart.title.text = 'Expenses Chart';
204
205 // Add a legend to the chart
206 chart.legend.position = 'bottom';
207
208 // Add a title to the chart
209 chart.title.text = 'Expenses Chart';
210
211 // Add a legend to the chart
212 chart.legend.position = 'bottom';
213
214 // Add a title to the chart
215 chart.title.text = 'Expenses Chart';
216
217 // Add a legend to the chart
218 chart.legend.position = 'bottom';
219
220 // Add a title to the chart
221 chart.title.text = 'Expenses Chart';
222
223 // Add a legend to the chart
224 chart.legend.position = 'bottom';
225
226 // Add a title to the chart
227 chart.title.text = 'Expenses Chart';
228
229 // Add a legend to the chart
230 chart.legend.position = 'bottom';
231
232 // Add a title to the chart
233 chart.title.text = 'Expenses Chart';
234
235 // Add a legend to the chart
236 chart.legend.position = 'bottom';
237
238 // Add a title to the chart
239 chart.title.text = 'Expenses Chart';
240
241 // Add a legend to the chart
242 chart.legend.position = 'bottom';
243
244 // Add a title to the chart
245 chart.title.text = 'Expenses Chart';
246
247 // Add a legend to the chart
248 chart.legend.position = 'bottom';
249
250 // Add a title to the chart
251 chart.title.text = 'Expenses Chart';
252
253 // Add a legend to the chart
254 chart.legend.position = 'bottom';
255
256 // Add a title to the chart
257 chart.title.text = 'Expenses Chart';
258
259 // Add a legend to the chart
260 chart.legend.position = 'bottom';
261
262 // Add a title to the chart
263 chart.title.text = 'Expenses Chart';
264
265 // Add a legend to the chart
266 chart.legend.position = 'bottom';
267
268 // Add a title to the chart
269 chart.title.text = 'Expenses Chart';
270
271 // Add a legend to the chart
272 chart.legend.position = 'bottom';
273
274 // Add a title to the chart
275 chart.title.text = 'Expenses Chart';
276
277 // Add a legend to the chart
278 chart.legend.position = 'bottom';
279
280 // Add a title to the chart
281 chart.title.text = 'Expenses Chart';
282
283 // Add a legend to the chart
284 chart.legend.position = 'bottom';
285
286 // Add a title to the chart
287 chart.title.text = 'Expenses Chart';
288
289 // Add a legend to the chart
290 chart.legend.position = 'bottom';
291
292 // Add a title to the chart
293 chart.title.text = 'Expenses Chart';
294
295 // Add a legend to the chart
296 chart.legend.position = 'bottom';
297
298 // Add a title to the chart
299 chart.title.text = 'Expenses Chart';
300
301 // Add a legend to the chart
302 chart.legend.position = 'bottom';
303
304 // Add a title to the chart
305 chart.title.text = 'Expenses Chart';
306
307 // Add a legend to the chart
308 chart.legend.position = 'bottom';
309
310 // Add a title to the chart
311 chart.title.text = 'Expenses Chart';
312
313 // Add a legend to the chart
314 chart.legend.position = 'bottom';
315
316 // Add a title to the chart
317 chart.title.text = 'Expenses Chart';
318
319 // Add a legend to the chart
320 chart.legend.position = 'bottom';
321
322 // Add a title to the chart
323 chart.title.text = 'Expenses Chart';
324
325 // Add a legend to the chart
326 chart.legend.position = 'bottom';
327
328 // Add a title to the chart
329 chart.title.text = 'Expenses Chart';
330
331 // Add a legend to the chart
332 chart.legend.position = 'bottom';
333
334 // Add a title to the chart
335 chart.title.text = 'Expenses Chart';
336
337 // Add a legend to the chart
338 chart.legend.position = 'bottom';
339
340 // Add a title to the chart
341 chart.title.text = 'Expenses Chart';
342
343 // Add a legend to the chart
344 chart.legend.position = 'bottom';
345
346 // Add a title to the chart
347 chart.title.text = 'Expenses Chart';
348
349 // Add a legend to the chart
350 chart.legend.position = 'bottom';
351
352 // Add a title to the chart
353 chart.title.text = 'Expenses Chart';
354
355 // Add a legend to the chart
356 chart.legend.position = 'bottom';
357
358 // Add a title to the chart
359 chart.title.text = 'Expenses Chart';
360
361 // Add a legend to the chart
362 chart.legend.position = 'bottom';
363
364 // Add a title to the chart
365 chart.title.text = 'Expenses Chart';
366
367 // Add a legend to the chart
368 chart.legend.position = 'bottom';
369
370 // Add a title to the chart
371 chart.title.text = 'Expenses Chart';
372
373 // Add a legend to the chart
374 chart.legend.position = 'bottom';
375
376 // Add a title to the chart
377 chart.title.text = 'Expenses Chart';
378
379 // Add a legend to the chart
380 chart.legend.position = 'bottom';
381
382 // Add a title to the chart
383 chart.title.text = 'Expenses Chart';
384
385 // Add a legend to the chart
386 chart.legend.position = 'bottom';
387
388 // Add a title to the chart
389 chart.title.text = 'Expenses Chart';
390
391 // Add a legend to the chart
392 chart.legend.position = 'bottom';
393
394 // Add a title to the chart
395 chart.title.text = 'Expenses Chart';
396
397 // Add a legend to the chart
398 chart.legend.position = 'bottom';
399
400 // Add a title to the chart
401 chart.title.text = 'Expenses Chart';
402
403 // Add a legend to the chart
404 chart.legend.position = 'bottom';
405
406 // Add a title to the chart
407 chart.title.text = 'Expenses Chart';
408
409 // Add a legend to the chart
410 chart.legend.position = 'bottom';
411
412 // Add a title to the chart
413 chart.title.text = 'Expenses Chart';
414
415 // Add a legend to the chart
416 chart.legend.position = 'bottom';
417
418 // Add a title to the chart
419 chart.title.text = 'Expenses Chart';
420
421 // Add a legend to the chart
422 chart.legend.position = 'bottom';
423
424 // Add a title to the chart
425 chart.title.text = 'Expenses Chart';
426
427 // Add a legend to the chart
428 chart.legend.position = 'bottom';
429
430 // Add a title to the chart
431 chart.title.text = 'Expenses Chart';
432
433 // Add a legend to the chart
434 chart.legend.position = 'bottom';
435
436 // Add a title to the chart
437 chart.title.text = 'Expenses Chart';
438
439 // Add a legend to the chart
440 chart.legend.position = 'bottom';
441
442 // Add a title to the chart
443 chart.title.text = 'Expenses Chart';
444
445 // Add a legend to the chart
446 chart.legend.position = 'bottom';
447
448 // Add a title to the chart
449 chart.title.text = 'Expenses Chart';
450
451 // Add a legend to the chart
452 chart.legend.position = 'bottom';
453
454 // Add a title to the chart
455 chart.title.text = 'Expenses Chart';
456
457 // Add a legend to the chart
458 chart.legend.position = 'bottom';
459
460 // Add a title to the chart
461 chart.title.text = 'Expenses Chart';
462
463 // Add a legend to the chart
464 chart.legend.position = 'bottom';
465
466 // Add a title to the chart
467 chart.title.text = 'Expenses Chart';
468
469 // Add a legend to the chart
470 chart.legend.position = 'bottom';
471
472 // Add a title to the chart
473 chart.title.text = 'Expenses Chart';
474
475 // Add a legend to the chart
476 chart.legend.position = 'bottom';
477
478 // Add a title to the chart
479 chart.title.text = 'Expenses Chart';
480
481 // Add a legend to the chart
482 chart.legend.position = 'bottom';
483
484 // Add a title to the chart
485 chart.title.text = 'Expenses Chart';
486
487 // Add a legend to the chart
488 chart.legend.position = 'bottom';
489
490 // Add a title to the chart
491 chart.title.text = 'Expenses Chart';
492
493 // Add a legend to the chart
494 chart.legend.position = 'bottom';
495
496 // Add a title to the chart
497 chart.title.text = 'Expenses Chart';
498
499 // Add a legend to the chart
500 chart.legend.position = 'bottom';
501
502 // Add a title to the chart
503 chart.title.text = 'Expenses Chart';
504
505 // Add a legend to the chart
506 chart.legend.position = 'bottom';
507
508 // Add a title to the chart
509 chart.title.text = 'Expenses Chart';
510
511 // Add a legend to the chart
512 chart.legend.position = 'bottom';
513
514 // Add a title to the chart
515 chart.title.text = 'Expenses Chart';
516
517 // Add a legend to the chart
518 chart.legend.position = 'bottom';
519
520 // Add a title to the chart
521 chart.title.text = 'Expenses Chart';
522
523 // Add a legend to the chart
524 chart.legend.position = 'bottom';
525
526 // Add a title to the chart
527 chart.title.text = 'Expenses Chart';
528
529 // Add a legend to the chart
530 chart.legend.position = 'bottom';
531
532 // Add a title to the chart
533 chart.title.text = 'Expenses Chart';
534
535 // Add a legend to the chart
536 chart.legend.position = 'bottom';
537
538 // Add a title to the chart
539 chart.title.text = 'Expenses Chart';
540
541 // Add a legend to the chart
542 chart.legend.position = 'bottom';
543
544 // Add a title to the chart
545 chart.title.text = 'Expenses Chart';
546
547 // Add a legend to the chart
548 chart.legend.position = 'bottom';
549
550 // Add a title to the chart
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
110      'rgb(54,  
111      162, 235)',  
112      'rgb(215, 159,  
113      64)'  
114    ],  
115    },  
116    options: {  
117      responsive: true,  
118      plugins: {  
119        legend: {  
120          position:  
121            'bottom',  
122          },  
123          title: {  
124            display: true,  
125            text: 'EXPENSE  
BREAKDOWN'  
126          }  
127        }  
128      }  
129    }  
130  }  
131}
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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    }
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

126

}

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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     texpanse = []  
8  
9     while dictionary1 != False:  
10         temp = []  
11         temp.append(dictionary1["TN"])
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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
```

Personal Expense Tracker Application

IBM-Project-34741-1660275861

```
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", texpanse = texpanse,
66         expense = expense, total = total ,
67         t_food = t_food,t_entertainment =
68         t_entertainment,
69         t_business = t_business, t_rent =
70         t_rent,
71         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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

sendemail.py

```
1  def sendgridmail(user,TEXT):
2      from_email = Email("tproduct8080@gmail.com")
3      to_email = To(user)
4      subject = "Crossed the Expenditure Limit"
5      content = Content("text/plain",TEXT)
6      mail = Mail(from_email, to_email, subject, content)
7
8      # Get a JSON-ready representation of the
      Mail object
9      mail_json = mail.get()
10     # Send an HTTP POST request to /mail/send
11     response = sg.client.mail.send.post(request_body=mail_json)
12     print(response.status_code)
13     print(response.headers)
```

7.3 Data Schema

Table : 1)

REGISTER

id INT NOT NULL GENERATEWD ALWAYS AS IDENTITY, username VARCHAR(255)

Personal Expense Tracker Application

IBM-Project-34741-1660275861

NOT NULL, password VARCHAR(255) NOT NULL

EXPENSES

id INT NOT NULL GENERATEWD ALWAYS AS IDENTITY, userid INT NOT NULL, date
TIMESTAMP NOT NULL, expensename VARCHAR(255) NOT NULL, amount INT NOT
NULL, paymode VARCHAR(255) NOT NULL < category VARCHAR(255) NOT NULL

LIMITS

id INT NOT NULL GENERATED ALWAYS AS IDENTITY, userid VARCHAR(255) NOT
NULL, limits VARCHAR(255) NOT NULL

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requsite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(V/N)	BUG ID	Executed By
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button.	None	1. Go to website 2. Home page appears	Username: JAJ3 Password: 123456	Login/Signup popup should display	Working as expected	Pass	-			Ashika SJ
LoginPage_TC_002	UI	Home Page	Verify the UI elements in Login/Signup popup	Home	1. Go to website 2. Enter details and click login	Username: JAJ3 Password: 123456	Application should show below UI elements: a. small text box b. Password text box c. Login button with blue colour d. New customer? Create account link e. Lost password? Recovery password link	Working as expected	Pass	-			Aishwarya T
LoginPage_TC_003	Functional	Home page	Verify user is able to log into application with Valid credentials	Username & password	1. Go to website 2. Enter details and click login	Username: JAJ3 Password: 123456	User should navigate to user account homepage	Working as expected	Pass	-			Aruna KR
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with Invalid credentials	Username & password	1. Go to website 2. Enter details and click login	Username: JAJ3 Password: 123456	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass	-			Jeevitha L
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with Invalid credentials	Login first	1. Go to website 2. Enter details and click login	Username: JAJ3 Password: 123456	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass	-			Ashika SJ
LoginPage_TC_005	Functional	Login page	Verify user is able to log into application with Invalid credentials	Login first	1. Go to website 2. Enter details and click login	Username: JAJ3 Password: 123456	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass	-			Aruna KR
AddExpensePage_TC_005	Functional	Add Expense page	Verify whether user is able to add expense or not	Have some expense to add	1. Add date, expense name and other details. 2. Check if the expense gets added	add rent - 5000	Application adds expense	Working as expected	Pass	-			Aishwarya T

Personal Expense Tracker Application

IBM-Project-34741-1660275861

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Personal Expense Tracker project at the 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 were 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
Signup	8	0	0	8
Final Report Output	4	0	0	4

9.RESULT

9.1 Performance Metrics

Personal Expense Tracker Application

IBM-Project-34741-1660275861

- 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.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

- 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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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

Personal Expense Tracker Application

IBM-Project-34741-1660275861

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 unpredictable 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.

Personal Expense Tracker Application

IBM-Project-34741-1660275861

13.APPENDIX

Source Code Github Link :

<https://github.com/IBM-EPBL/IBM-Project-34741-1660275861>

Project Demo Link :

https://drive.google.com/file/d/1OhD3pEgiqUsX_lvX8B0NbYbvafWgecm/view?usp=drivesdk