

Team ID	PNT2022TMID20376
Project Name	Personal Expense Tracker Application
Team members	Garishma k (Team Lead) Abirami S Amirta varshini N Bhoomika T Darshini R

CHAPTER 1

INTRODUCTION

The introduction briefs about evolution of fabric defect detection in industries and recent approaches.

1.1 Project Overview

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

When you track your spending, you know where your money goes and you can ensure that your money is used wisely. Tracking your expenditures also allows you to understand why you're in debt and how you got there. This will then help you design a befitting strategy of getting out of debt.

Budgeting ensures you're not spending more than you're making, allowing you to plan for short- and long-term expenses. It's an easy, helpful way for people with all types of income and expenses to keep their finances in order.

CHAPTER 2

LITERATURE SURVEY

Literature review was carried out to gain knowledge and improve the skills needed to complete this project. This chapter shows the different techniques that have been implemented.

2.1 Existing Problem

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.

2.2 References

- [1] <http://expense-manager.com/how-expense software/>
 - [2] <https://www.splitwise.com/terms>
 - [3] <http://code.google.com/p/socialauthandroid/wiki/Facebook>
 - [4] <http://code.google.com/p/socialauth-android>
 - [5] Developer.android.com
 - [6] <http://www.appbrain.com/app/expensemanager/ com.expensemanager>
 - [7] <https://www.xpenditure.com/en?>
 - [8] <http://expense-manager.com/how-expense software/>
 - [9] Donn Felker, "Android Application Development for Dummies", published by For Dummies, 2010.
 - [10] Ed Burnette, "Hello, Android: Introducing Google's Mobile Development Platform", published by Pragmatic Bookshelf, 2009.
- Lee, "Beginning Android Application Development", Published by WroxPress, 2011.

2.3 Problem Statement Definition

It is tough to keep track of all the financial decisions and activities that a person makes. Traditional expense tracking methods are inconvenient and unreliable. In order to get a quick overview about your total incomes and expenses and control spending , its convenient to digitize the process by having a personal expense tracker.

Who does the problem affect?	Working individuals, students and budget conscious consumers.
What are the boundaries of the problem?	Limited features to provide for expense tracking.
What is the issue?	To be vigilant about the expense spent increases financial stress. Being indecisive about the finances may result in less financial security and exceed the budget.
When does this issue occur?	When people are not able to track their expenses properly.
Where is the issue occurring?	In daily life of employees as well as students.
Why is it important that we fix the problem?	Fixing this issue will help users to better plan their budget and lead to financial well-being.

- Sophie, who is a homemaker, finds it hard to control her desire to shop. To stop herself from overindulging in impulsive purchases, she needs to track her expenses and hold herself accountable.
- Sam is a high school student who usually gets a limited allowance from his parents. Tracking his expenses and good budgeting technique allows him to spend on his regular expenses as well as on himself.
- Percy, who is a novice budgeter, finds it tedious to track and manage the expenses amongst his busy schedule. Prioritizing his expenses will help him to curtail his unnecessary expenditures.

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

3.2.a Brainstorm

Garishma K



Abirami S



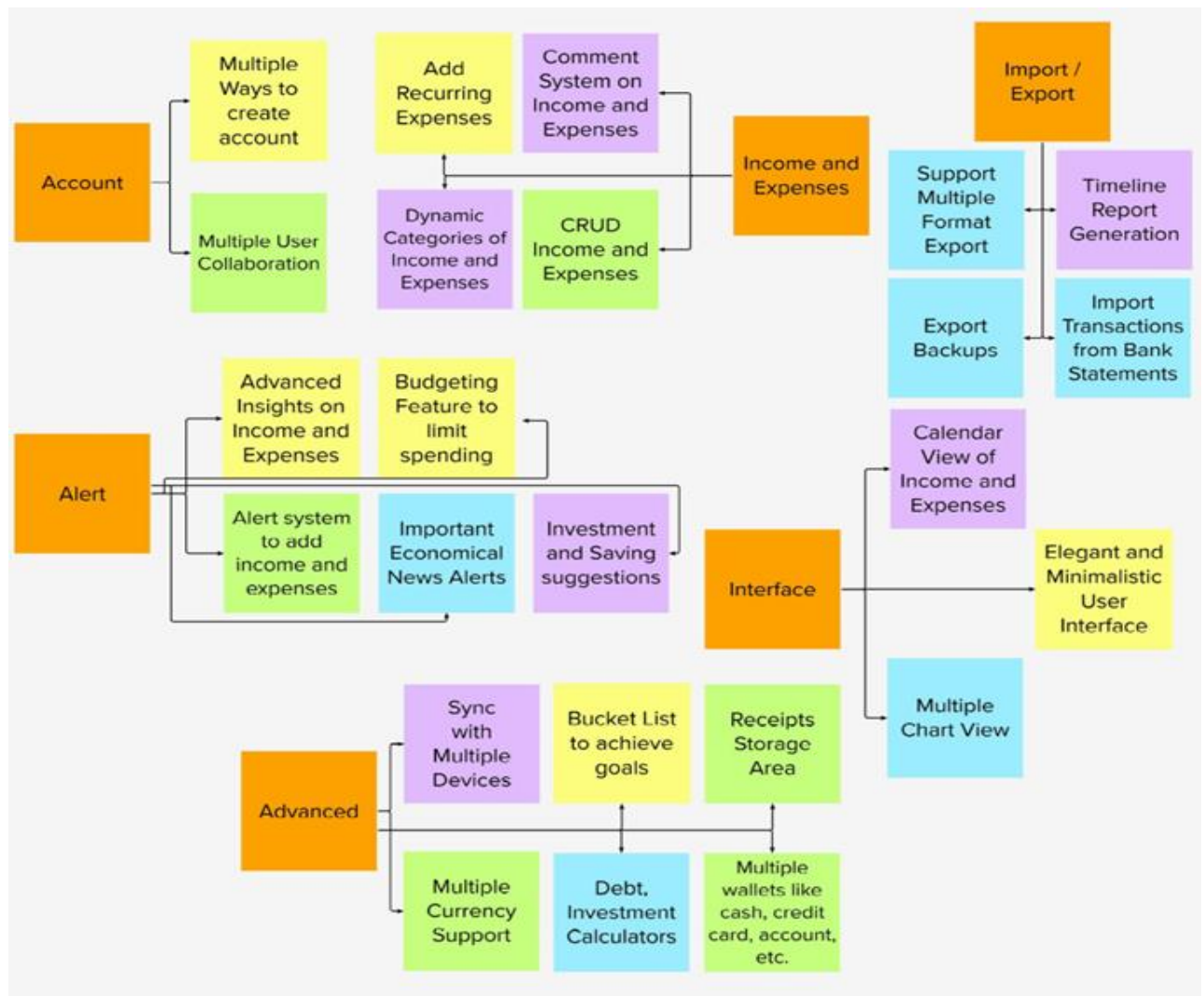
Amirta varshini N



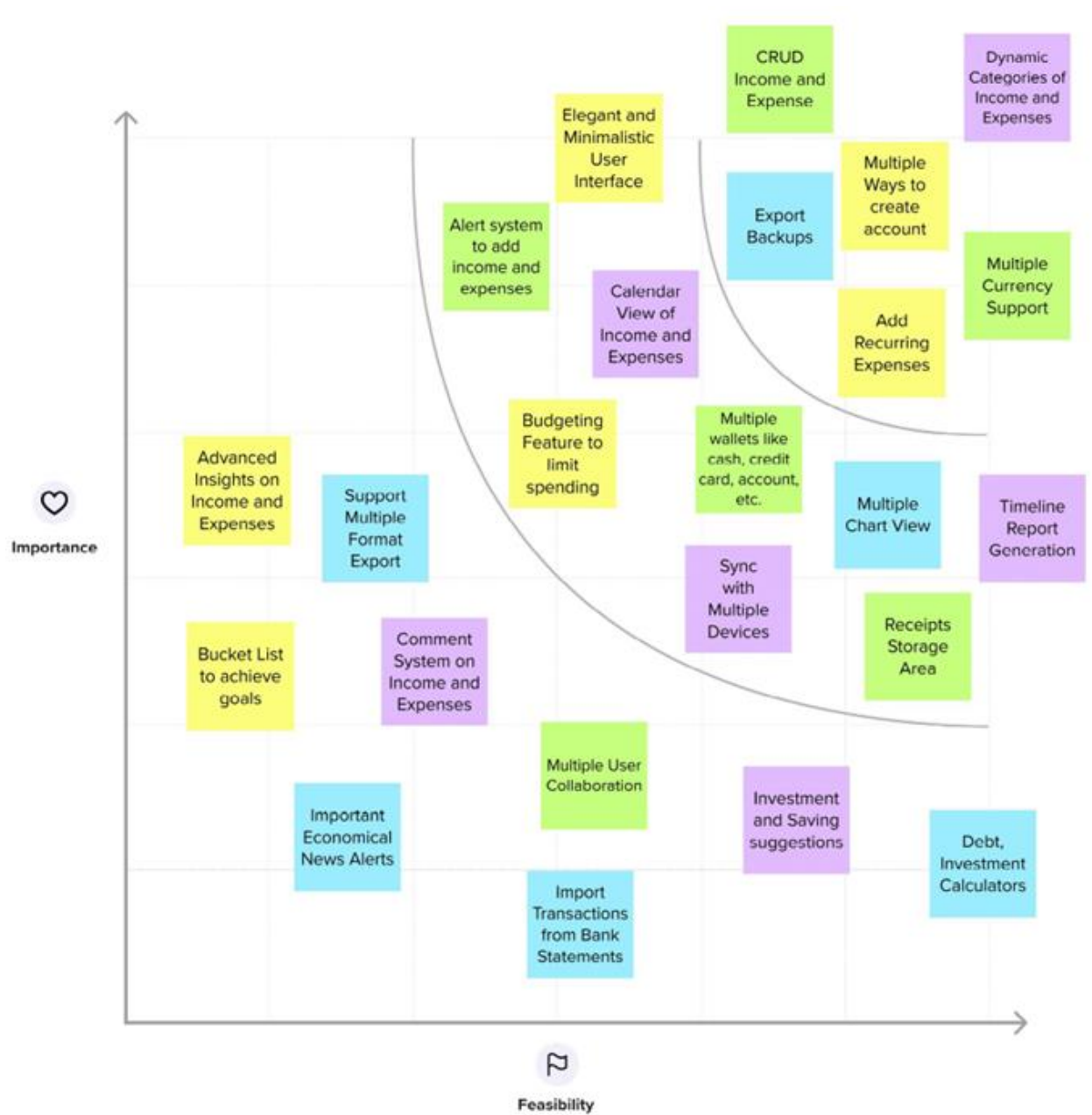
Darshini R and Bhoomika T



3.2.b Group Ideas



3.2.c Prioritize



3.3 Proposed Solution

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Attempting to manage the expenses of an individual in an efficient and manageable manner, as compared to the traditional way of expense tracking.
2.	Idea / Solution description	The application will be helpful for the individuals in not just managing their expenses, but also in enabling them to improve their investments.
3.	Novelty / Uniqueness	The application gives the user a chance to plan his/her monthly expenses at the start of the month. Besides this, the user gets a notification when he/she exceeds the limit that is set.
4.	Social Impact / Customer Satisfaction	With such applications, the public will start to plan their expenses better leading to their own financial stability. With more users, this application will ensure that financial state of our society improves.
5.	Business Model (Revenue Model)	Free trial for 1 month can be given to the users, so that a significant userbase is created. Following the free trial, the users can be given subscription for 3 months, 6 months or 1 year.
6.	Scalability of the Solution	Since the application takes the same set of input from all the users and does not perform many complex computations, it will be easy for us to scale the application to a larger set of users.

3.4 Problem Solution fit

PROBLEM-SOLUTION FIT

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) <ul style="list-style-type: none">• Working Individuals• Students• Budget conscious consumers	6. CUSTOMER CONSTRAINTS <ul style="list-style-type: none">• Internet Access• Device (Smartphone) to access the application• Data Privacy• Cost of existing applications• Trust	5. AVAILABLE SOLUTIONS <ul style="list-style-type: none">• Expense Diary or Excel sheet <p>PROS : Have to make a note daily which helps to be constantly aware</p> <p>CONS : Inconvenient, takes a lot of time</p>						
	2. JOBS-TO-BE-DONE / PROBLEMS <ul style="list-style-type: none">• To keep track of money lent or borrowed• To keep track of daily transactions• Alert when a threshold limit is reached	9. PROBLEM ROOT CAUSE <ul style="list-style-type: none">• Reckless spendings• Indecisive about the finances• Procrastination• Difficult to maintain a note of daily spendings (Traditional methods like diary)	7. BEHAVIOUR <ul style="list-style-type: none">• Make a note of the expenses on a regular basis.• Completely reduce spendings or spend all of the savings• Make use of online tools to interpret monthly expense patterns						
Identify strong TR & EM	3. TRIGGERS <ul style="list-style-type: none">• Excessive spending• No money in case of emergency	10. YOUR SOLUTION <p>Creating an application to manage the expenses of an individual in an efficient and manageable manner, as compared to traditional methods</p>	8. CHANNELS OF BEHAVIOUR <p>ONLINE</p> <p>Maintain excel sheets and use visualizing tools</p>						
	4. EMOTIONS <table><tr><td>BEFORE</td><td>AFTER</td></tr><tr><td>• Anxious</td><td>• Confident</td></tr><tr><td>• Confused</td><td>• Composed</td></tr><tr><td>• Fear</td><td>• Calm</td></tr></table>		BEFORE	AFTER	• Anxious	• Confident	• Confused	• Composed	• Fear
BEFORE	AFTER								
• Anxious	• Confident								
• Confused	• Composed								
• Fear	• Calm								

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Application Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User monthly expense tentative data	Data to be registered in the app
FR-4	User monthly income data	Data to be registered in the app
FR-5	Alert/ Notification	Alert through E-mail Alert through SMS
FR-6	User Budget Plan	Planning and Tracking of user expense vs budget limit

4.2 Non Functional Requirements

Following are the non-functional requirements of the proposed solution.

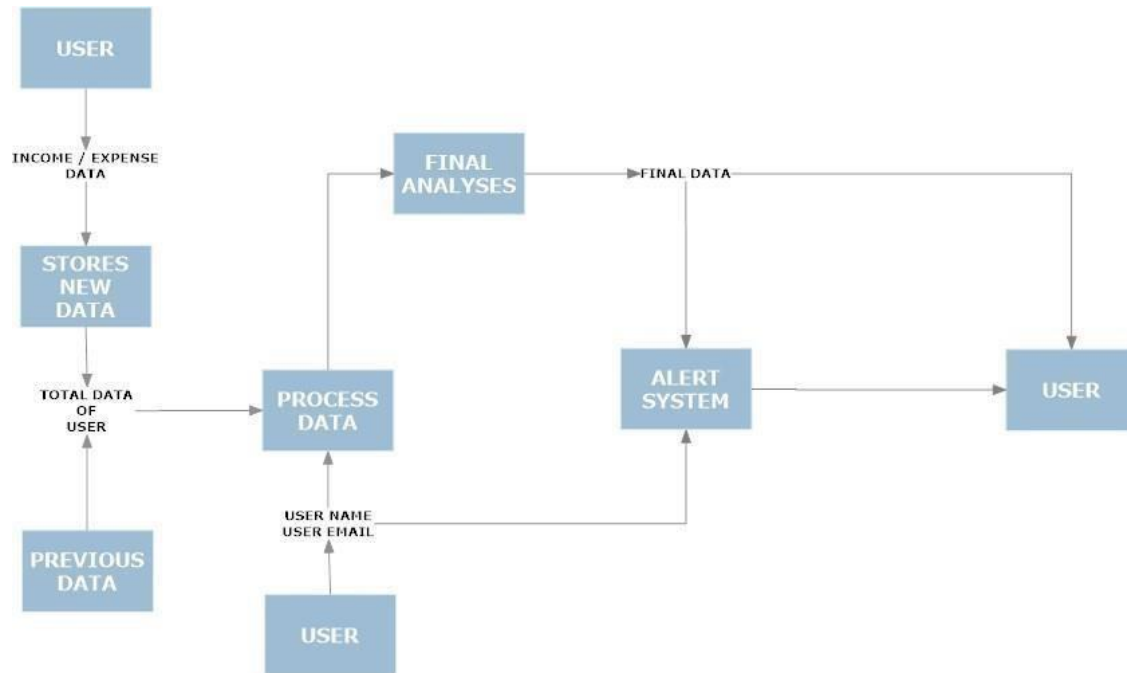
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Effectiveness, efficiency and overall satisfaction of the user while interacting with our application.
NFR-2	Security	Authentication, authorization, encryption of the application.
NFR-3	Reliability	Probability of failure-free operations in 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.

CHAPTER 5

PRODUCT DESIGN

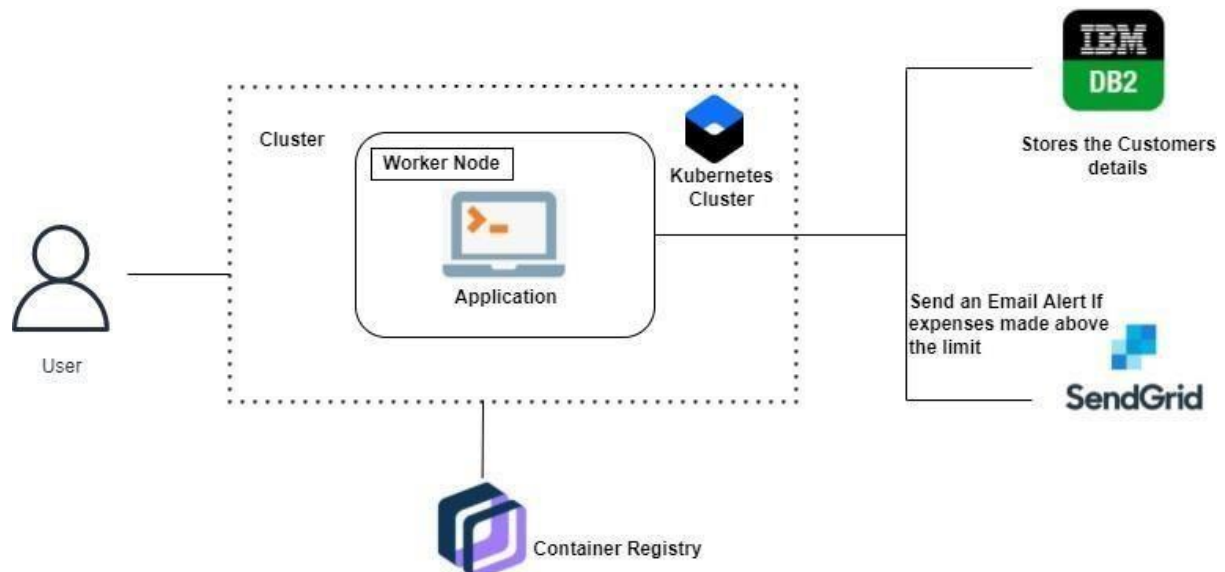
5.1 Data Flow Diagrams

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



5.2 Technical Architecture

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



5.3 User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobileuser & web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	
		USN- 3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	
	Login	USN -4	As a user, I can log into the application by entering email & password	I can access the application	High	
	Dashboard	USN -5	As a user I can enter my income and expenditure details.	I can view my daily expenses	High	
Customer Care Executive		USN 6	As a customer care executive I can solve the log in issues and other issues of the application.	I can provide support or solution at any time 24*7	Medium	
Administrator	Application	USN -7	As 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	

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

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

Sprint	Functional requirement (Epic)	User story number	User story/task	Story points	priority	Team members
Sprint-1	Registration	USN-1	Register for the application by entering email,password, and confirming password.	2	High	Amirta Varshini
Sprint-1		USN-2	As a user will receive confirmation email once have registered for the application	1	High	Amirta Varshini
Sprint-2		USN-3	Register for the application through Facebook	2	Low	Bhoomika
Sprint-1		USN-4	User register for the application through Gmail	2	Medium	Garishma
Sprint-1	Login	USN-5	User log into the application by entering email & password	1	High	Garishma
Sprint-3	Dashboard	USN-6	Expenditure details on the application	3	High	Darshini
Sprint-3	Limits	USN-6	User can set monthly expense limit so that receive a mail on exceeding	4	High	Darshini
Sprint-4	Reports	USN-6	View the graphical form of expenses category wise	5	Medium	Abirami

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	6	6 Days	15 Oct 2022	30 Oct 2022	6	30 Oct 2022
Sprint-2	2	6 Days	30 Oct 2022	05 Nov 2022	2	05 Nov 2022
Sprint-3	7	6 Days	05 Nov 2022	12 Nov 2022	7	12 Nov 2022
Sprint-4	5	6 Days	12 Nov 2022	19 Nov 2022	5	19 Nov 2022

6.3 Reports from JIRA

The screenshot shows the JIRA Backlog for the 'Personal Expense Tracker Application'. The left sidebar contains navigation links for 'PLANNING' (Roadmap, Backlog), 'DEVELOPMENT' (Board, Code), and 'Project pages' (Project pages, Add shortcut, Project settings). The main content area is titled 'Backlog' and shows a search bar, filters (GK, AV, B, D, S), and an 'Epic' dropdown. The 'PETA Sprint 1' (30 Oct - 14 Nov) is expanded, showing 4 issues: PETA-1 (register for the application by entering email, password, and confirming password) in 'IN PROGRESS' status, and PETA-3, PETA-8, and PETA-9 in 'TO DO' status. Below the sprint, there is an empty 'Backlog' section with the message 'Your backlog is empty.'

CHAPTER 7

7.1 Feature 1

1. Expense and revenue tracking.
2. Managing transaction receipts and records.
3. Paying taxes in time.
4. Processing payment and invoices.
5. Create in-depth reports.

CHAPTER 8

TESTING

8.1 Test Cases

Registration Page

Registration

File | C:/Users/Gowtham/Desktop/Praveen/Personal%20Expense%20Tracker/Project%20Development%20Phase/Sprint%201/Templates/registration.html

Personal Expense Tracker

Registration Form

Email:

Password:

Please make sure that the password meets the following requirements:

1. Minimum of 8 characters
2. Contains an upper case and a special character

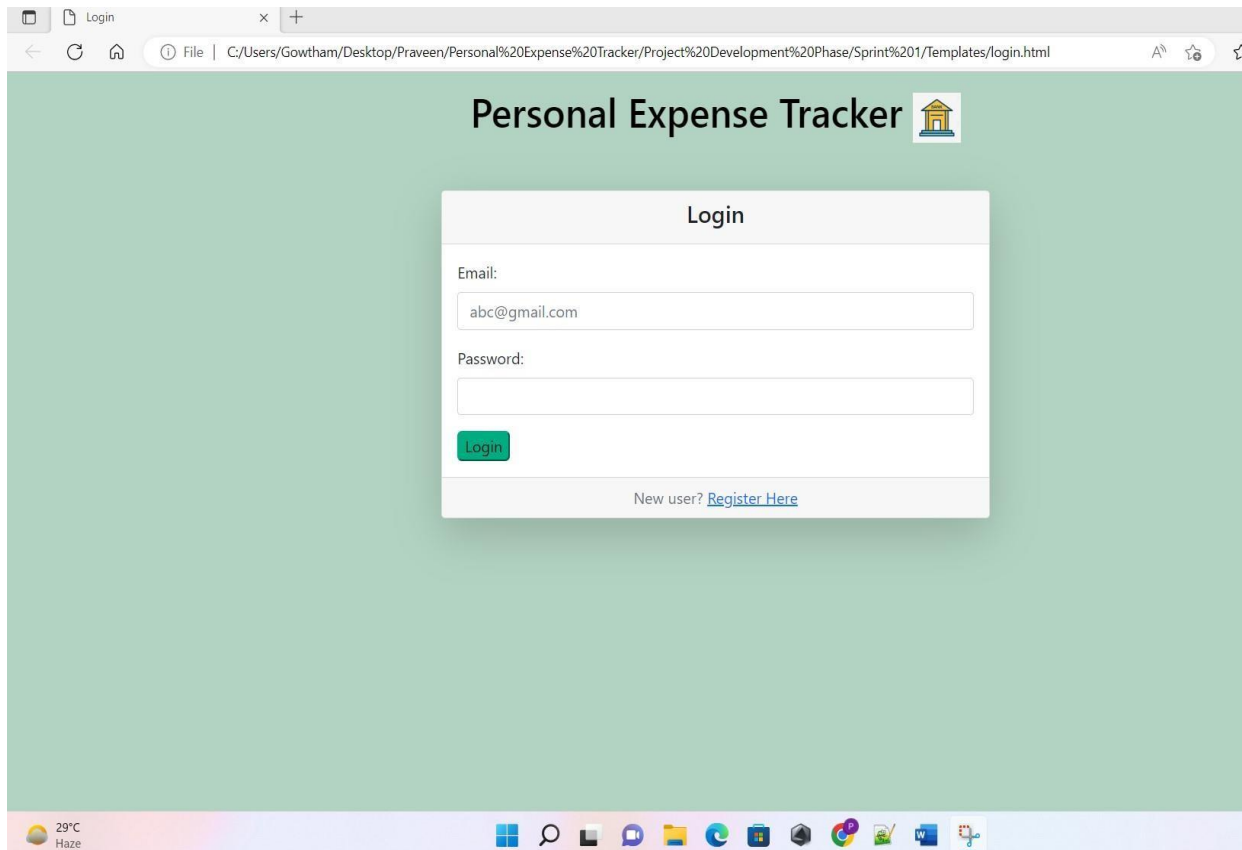
Confrim Password:

Initial Wallet Amount (Rs):

[Register](#)


Already an existing user? [Login Here](#)


LOGIN





ADD EXPENSE


Personal Expense Tracker

 Home


 Add Expense

 Modify Expense

 View Analysis

 Rewards & Goals


Add expense

Expense Made 

Amount Spent: (Rs)

Expense Category:

Date of Expense:



Description of Expense:

Submit Expense

CHAPTER 9

RESULTS

9.1 Performance Metrics

An application can be a very powerful tool for businesses if once the app becomes a success. However, the success of an app is measured through numbers, metrics, and analytics. Developing an app takes quite a lot, so once you've dedicated much time, money, and effort to the process, it's mandatory to measure mobile app performance.



CHAPTER 10

ADVANTAGES & DISADVANTAGES

10.1 Advantages

- Improved customer service
- Cloud-based solution
- Order Fulfillment
- Harness Customer Loyalty and Retention

10.2 Disadvantages

- System Clash
- Reduced Physical Audits
- No solution to improve or eliminate bottlenecks in the service cycle

CHAPTER 11

CONCLUSION

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

CHAPTER 12

FUTURE SCOPE

- 1) It will have various options to keep record (for example Food, Travelling Fuel, Salary etc.).
- 2) Automatically it will keep on sending notifications for our daily expenditure.
- 3) In today's busy and expensive life, we are in a great rush to make moneys, but at the end of the month we broke off. As we are unknowingly spending money on title and unwanted things. So, we have come over with the plan to follow our profit.
- 4) Here user can define their own categories for expense type like food, clothing, rent and bills where they have to enter the money that has been spend and likewise can add some data in extra data to indicate the expense.

CHAPTER 13

APPENDIX

SOURCE CODE:

Registration Page

```
<!doctype html>

<html lang="en">

<head>
  <!-- Required meta tags -->

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASj
C" crossorigin="anonymous">

  <title>Registration</title>
</head>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIax
VXM"crossorigin="anonymous"></script>

<body style="background-color:#B2D3C2">
  <div class="container mt-3">
    <h1 style="color: black; text-align: center;">

      Personal Expense Tracker 
    </h1>

    <div class="container mt-2" style="width: 600px;">
      <div class="card shadow-lg bg-white rounded">
        <div class="card-header" style="text-align: center;">

          <h4>Registration Form</h4>

        </div>
        <div class="card-body">
```

```

<form action="/" method="POST">
  <div class="mb-3">
    <label for="email" class="form-label">Email: </label>
    <input type="email" class="form-control" name="email" id="email"
placeholder="abc@gmail.com">
  </div>

  <div class="mb-3">
    <label for="password" class="form-label">Password: </label>

    <input type="password" class="form-control"
name="password" id="password"></input>

    <p style="color: gray;" class="mt-3">Please make sure that the
password meets the following requirements:</p>

    <ol style="color: gray;"><li>Minimum of 8
characters</li><li>Contains an upper case and a special character</li></ol>

  </div>
  <div class="mb-3">
    <label for="confirmpassword" class="form-label">Confirm
Password:
</label>

    <input type="password" class="form-control"
name="confirmpassword"
id="confirmpassword" placeholder="*****">

  </div>
  <div class="mb-3">
    <label for="wallet" class="form-label">Initial Wallet Amount (Rs):
</label>

    <input type="number" class="form-control" name="wallet"
id="wallet"
placeholder="10000.0
0">
  </div>

  <button type="submit" style="background-color:#00AD83;
border-color:#00AD83; border-radius:5px;">Register</button>

</form>
</div>

<div class="card-footer text-muted" style="text-align:center">
  Already an existing user? <span><a href="login">Login
Here</a></span>
</div>

```

```

        </div>
    </div>
</div>
</body>
</html>

```

OUTPUT:

The screenshot shows a web browser window with a single tab titled 'Registration'. The address bar shows the file path: 'C:/Users/Gowtham/Desktop/Praveen/Personal%20Expense%20Tracker/Project%20Development%20Phase/Sprint%201/Templates/registration.html'. The main content area has a light green background. At the top center, the text 'Personal Expense Tracker' is displayed next to a small icon of a building. Below this, a white box with a grey border contains the 'Registration Form'. The form has the following fields and elements:

- Email:** A text input field containing 'abc@gmail.com'.
- Password:** A text input field.
- Requirements:** Below the password field, it says 'Please make sure that the password meets the following requirements:' followed by a list:
 1. Minimum of 8 characters
 2. Contains an upper case and a special character
- Confirm Password:** A text input field showing eight asterisks '*****'.
- Initial Wallet Amount (Rs):** A text input field containing '10000.00'.
- Register:** A green button with white text.
- Login Link:** Below the button, it says 'Already an existing user? [Login Here](#)'.

LOGIN

```

<!doctype html>
<html lang="en">

<head>

    <!-- Required meta tags -->
    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->

```

```
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLA
SjC" crossorigin="anonymous">
```

```
<title>Login</title>
</head>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min
.js" integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/t
WtIaxVXM" crossorigin="anonymous"></script>
```

```
<body style="background-color:#B2D3C2">
<div class="container mt-3">
  <h1 style="color: black; text-align: center;">
    Personal Expense Tracker 
  </h1>
  <div class="container mt-5" style="width: 600px;">
    <div class="card shadow-lg bg-white rounded">
      <div class="card-header" style="text-align: center;">
        <h4>Login</h4>
      </div>
      <div class="card-body">
```

```
<form action="/login" method="POST">
  <div class="mb-3">

    <label for="email" class="form-label">Email: </label>

    <input type="email" class="form-control" name="email"
id="email" placeholder="abc@gmail.com">

  </div>
  <div class="mb-3">

    <label for="passowrd" class="form-label">Password: </label>

    <input type="password" class="form-control"
name="password" id="password"></input>

  </div>
  <button type="submit" style="background-
color:#00AD83; border-color:#00AD83; border-
radius:5px;">Login</button>

</form>
</div>

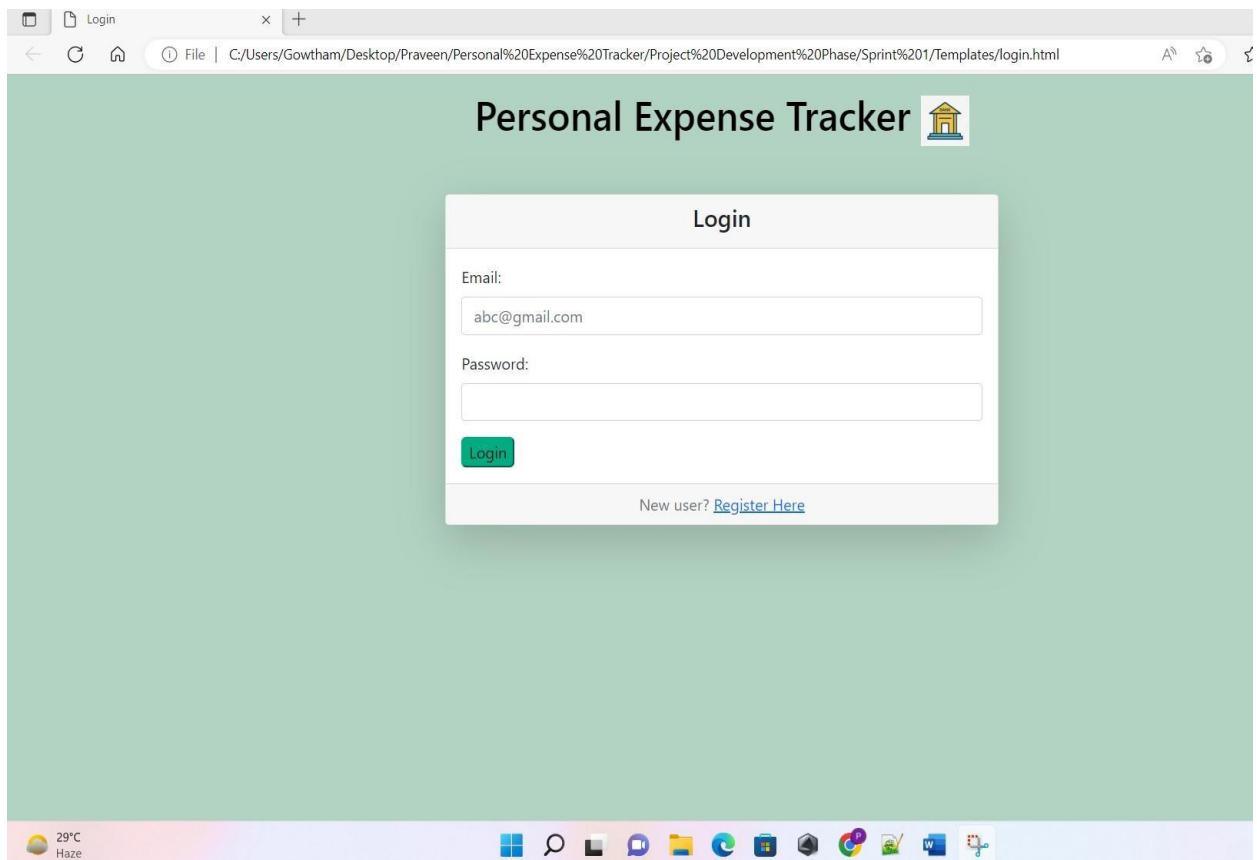
<div class="card-footer text-muted" style="text-align:center">

  New user? <span><a href="registration.html">Register
  Here</a></span>

</div>
</div>

</div>
</body>
</html>
```

OUTPUT:



Add Expense

```
<!doctype html>
```

```
<html lang="en">
```

```
<head>
```

```
<!-- Required meta tags -->
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<!-- Bootstrap CSS -->
```

```
<link
```

```
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
```

```
rel="stylesheet" integrity="sha384-
```

```
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMLA
```

```
SjC" crossorigin="anonymous">
```

```
<title>AddExpense</title>
```



```
</head>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min
.js" integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/
tWtIaxVXM" crossorigin="anonymous"></script>
```

```
<div class="container-fluid" >

  <div class="row flex-nowrap">
    <div class="col-auto col-md-3 col-xl-2 px-sm-2 px-0"
style="background-color:#B2D3C2">

      <div class="d-flex flex-column align-items-center align-items-sm-
start px-3 pt-2 min-vh-100" style="color:black">

        <p class="d-flex align-items-center pb-3 mb-md-0 me-md-auto
text-white text-decoration-none">

          <span class="fs-5 d-none d-sm-inline" style="color:black;
font-weight:bold;">Personal Expense Tracker</span>

          
        </p>

        <ul class="nav nav-pills flex-column mb-sm-auto mb-0 align-items-
center align-items-sm-start" id="menu">
```

```
<li class="nav-item mt-2">
  <a href="dashboard.html" class="nav-link align-middle px-0"
style="color:black;">

    <span class="ms-1 d-none d-sm-inline">Home</span>

  </a>
</li>
<li class="nav-item mt-2" style="background-color:#00AD83;
height: 50px;width: 150px; border-radius: 5px;" >

  <a href="addexpense.html" class="nav-link px-0
align-middle"style="color:black;">

    <span class="ms-1 d-none d-sm-inline">Add Expense</span>

  </a>
</li>
<li class="nav-item mt-2">
  <a href="modifyexpense.html" class="nav-link px-0 align-middle"
style="color:black;">

    <span class="ms-1 d-none d-sm-inline">Modify
Expense</span><imgsrc="edit_icon.svg" style="width:20px;height:20px;margin-
left: 5px;">

  </a>
</li>
<li class="nav-item mt-2">

  <a href="analysis.html" class="nav-link px-0
align-middle"style="color:black;">

    <span class="ms-1 d-none d-sm-inline">View
Analysis</span><imgsrc="graph.png" style="width:20px;height:20px;margin-left:
5px;">

  </a>
</li>
<li class="nav-item mt-2">

  <a href="rewards.html" class="nav-link px-0 align-middle"
style="color:black;">

    <span class="ms-1 d-none d-sm-inline">Rewards &
Goals</span><imgsrc="reward.png" style="width:20px;height:20px;margin-left:
5px;">
```

```

        </a>
    </li>
</ul>
</div>

</div>
<div class="col py-3" style="background-color:black">
    <h3 style="color:white; text-align: center;">Add expense</h3>

    <div class="container mt-3" style="width: 600px;">
        <div class="card shadow-lg bg-white rounded">
            <div class="card-header" style="text-align: center;">

                <span style="display:inline-flex"><h4>Expense
Made</h4><imgsrc="pay.png" style=" margin-left:10px; width:30px;
height:30px"></span>

            </div>

            <div class="card-body">
                <form>
                    <div class="mb-3">

                        <label for="amountspent" class="form-label">Amount Spent: (Rs)
</label>

                        <input type="number" class="form-control" name="amountspent"
id="amountspent" placeholder="100.00">

                    </div>
                    <div class="mb-3">
                        <label for="expensecategory" class="form-label">Expense
Category:
</label>

                        <input type="text" class="form-control" name="expensecategory"
id="expensecategory"></input>

                    </div>

                    <div class="mb-3">
                        <label for="date" class="form-label">Date of Expense: </label>
                        <input type="date" class="form-control" name="date"
id="date"></input>

                    </div>
                </form>
            </div>
        </div>
    </div>

```

```

<div class="mb-3">
  <label for="description" class="form-label">Description of
  Expense:
</label>
  <input type="text" class="form-control" name="description"
  id="description"></input>
</div>

</form>

</div>
<div class="card-footer text-muted" style="text-align:center">

  <button type="submit" style="background-
  color:#00AD83; border-color:#00AD83; border-radius:5px;">Submit
  Expense</button>

</div>

</div>

</div>

</div>

</div>

</div>

</html>

```

Output:

The screenshot displays a web browser window with a single tab titled 'AddExpense'. The address bar shows the file path: `C:/Users/Gowtham/Desktop/Praveen/Personal%20Expense%20Tracker/Project%20Development%20Phase/Sprint%201/Templates/addexpense.html`. The application has a dark green sidebar on the left with the title 'Personal Expense Tracker' and a house icon. Below the title, there are four menu items: 'Home' with a house icon, 'Add Expense' with a green button and a plus icon, 'Modify Expense' with a pencil icon, 'View Analysis' with a bar chart icon, and 'Rewards & Goals' with a crown icon. The main content area has a black background. At the top right, it says 'Add expense' in white. Below this, there is a white box titled 'Expense Made' with a plus icon. Inside this box, there are four input fields: 'Amount Spent: (Rs)' with the value '100.00', 'Expense Category:' (empty), 'Date of Expense:' with the placeholder 'dd-mm-yyyy' and a calendar icon, and 'Description of Expense:' (empty). At the bottom of the white box is a green button labeled 'Submit Expense'.

The python code to Connect with DB

```
import
ibm_db
dictionary=
{}
def printTableData(conn):
sql = "SELECT * FROM
userdetails" out =
ibm_db.exec_immediate(conn,
```

```

sql)document =
ibm_db.fetch_assoc(out) while
document != False:
dictionary.update({ document['USERNAME']:
document['PASSWORD']})
document =
ibm_db.fetch_assoc(out)def
insertTableData(conn,rollno,username,email,password):
sql="INSERT INTO
userdetails(rollno,username,email,password) VALUES
({},{},{},{})".format(rollno,username,email,password)
out = ibm_db.exec_immediate(conn,sql)
print('Number of affected rows :
',ibm_db.num_rows(out),"\\n")def
updateTableData(conn,rollno,username,email,password):
sql = "UPDATE userdetails SET (username,email,password)=('',{},{})
WHERE rollno={ }".format(username,email,password,rollno)
out = ibm_db.exec_immediate(conn, sql)
print('Number of affected rows : ',
ibm_db.num_rows(out), "\\n")def
deleteTableData(conn,rollno):
sql = "DELETE FROM userdetails WHERE rollno={ }".format(rollno)

```

```

out = ibm_db.exec_immediate(conn, sql) print('Number of affected rows : ',
ibm_db.num_rows(out), "\n")

try:

conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-
4859-88f0-
a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30
756;Security=
SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=ctb99199;PWD=
GybYxLw1rHz86oSh;"", "", "")

print("Db
connected")

except:

print("Error")

from flask import
Flask,render_template,request,url_for,session

app=Flask(__name__)

@app.route("/")

@app.route("/login",methods=['POST
T','GET'])def login():

if request.method=="POST":

printTableData(conn)

username=request.form['username']

password=request.form['password']

try: if dictionary[username] == password and username in dictionary:

```