

PERSONAL EXPENSE TRACKER APPLICATION

**CATEGORY: CLOUD APP DEVELOPMENT
TEAM ID: PNT2022TMID35580**

A PROJECT REPORT

BY

NITHIN RHAJ 2019115062

RAMYA N 2019115070

PAARUSH S 2019115048

SIDARTH S 2019115097

CONTENTS

1. INTRODUCTION

1.1 PROJECT OVERVIEW

1.2 PURPOSE

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

2.2 REFERENCES

2.3 PROBLEM STATEMENT DEFINITION

3. IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

3.2 IDEATION AND BRAINSTORMING

3.3 PROPOSED SOLUTION

3.4 PROBLEM SOLUTION FIT

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

4.2 NON-FUNCTIONAL REQUIREMENTS

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

6.2 SPRINT DELIVERY SCHEDULE

6.3 REPORTS FROM JIRA

7. CODING AND SOLUTIONING

7.1 FEATURE1

7.2 FEATURE 2

7.3 DATABASE SCHEMA

8. TESTING

8.1 TEST CASES

8.2 USER ACCEPTANCE TESTING

8.3 PERFORMANCE TESTING

9. RESULTS

9.1 PERFORMANCE METRICS

10. ADVANTAGES AND DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

INTRODUCTION

1.1 PROJECT OVERVIEW

In order to help users successfully manage their spending, this project entails developing a personal expense tracker application. The user-friendly programme is designed to allow users to plug in their expenses and visualise them so they can keep tabs on their spending. A user may also set a spending cap, and if this cap is exceeded, the user will receive an email notification.

1.2 PURPOSE

This project was created to assist people in managing their finances well because occasionally people struggle with saving and budgeting. This app provides the entire experience with regular email updates when the user crosses his budget set. A graphical representation of the expenditure done is also developed for users to track their spendings for the month.

LITERATURE SURVEY

	Title of the Paper	Methodology Used	Observations	Results + Conclusions	Limitations
1	Income and Expense Tracker	Android App	App has features like add income, add category, categories, export income, remove export files, and view expenses	The project successfully avoids the manual calculation for avoiding calculating the income and expense per month.	Only Supports Android Platform
2	Daily Expense Tracker	Android Application with Database	The project successfully avoids manual calculation which results in less human effort, saves time and number of calculation mistakes occurring is not less but zero.	The modules developed were efficient, attractive and easy to understand and easy to use.	Only Supports Android Platform
3	EXPENSE TRACKER	Mobile App using Database	a top-down approach, with the focus on what comes first, followed by how, and then on to subsequent levels of detail	application enables the user to better manage his expenses	Only Supports Android Platform
4	Expense Tracker : A Smart Approach to Track Everyday Expense	Java and MySQL Workbench based App	It will prove to be helpful for the people who are frustrated with their daily budget management, irritated because of amount of expenses and wishes to manage money and to preserve the record of their daily cost which may be useful to change their way of spending money.	Expense Tracker app is usable by anyone who are willing to manage their expenses and aiming to save for the future investments.	Difficult to export data
5	Expense Manager Application	Android Application backed by SQLite	This app has multiple features like Investment consultant, Stock Market viewer, Financial New viewer, Split Bill and Offer Zone	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	Only Supports Android Platform

Bekaroo, G., & Sunhaloo, S. Intelligent Online Budget Tracker. [1]

Features of the app include adding revenue, adding categories, exporting money, removing export files, and viewing. Although the product only supported the Android platform, it successfully avoided manual calculations for determining monthly income and expenses.

Chandini, S., Poojitha, T., Ranjith, D., Akram, V. M., Vani,

M. S., & Rajyalakshmi, V. (2019). Online Income and Expense Tracker. [3]

The project successfully eliminates manual calculation, resulting in less human work, time savings, and absolutely no calculation errors. The modules created were effective, appealing, simple to learn, and simple to utilise.

Underwood, D. (2011). A Case Study of Tracking Expenses by Commodity at Widget Farmers' Cooperative. [2]

The emphasis was on what comes first, then how, and then on to following levels of detail. This was a top-down strategy. The user of this programme can better control his costs.

Sabab, S. A., Islam, S. S., Rana, M. J., & Hossain, M. (2018, September). eExpense: A smart approach to track everyday expense. In 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT) (pp. 136-141). IEEE. [5]

It will be beneficial for those who are upset with managing their daily budgets, annoyed by the sheer volume of

expenses, and who wish to manage their finances and keep track of their daily costs in case they need to alter their spending habits. Anyone who wants to monitor their spending and wants to save for future investments can use this app.

Satpute, M. K., Kale, A., Mandal, A., & Krishnan, R.
SURVEY ON CLASSIFICATION ENGINE FOR MONETARY
TRANSACTIONS [4]

This programme contains a variety of features, including Offer Zone, Split Bill, Stock Market Viewer, and Financial News Viewer. This app was a smartphone application that kept track of all of your daily transactions, tracked any money you had lent or borrowed, offered you discounts on popular items, offered you the best investment possibilities, and allowed you to read the most recent, authenticated financial news.

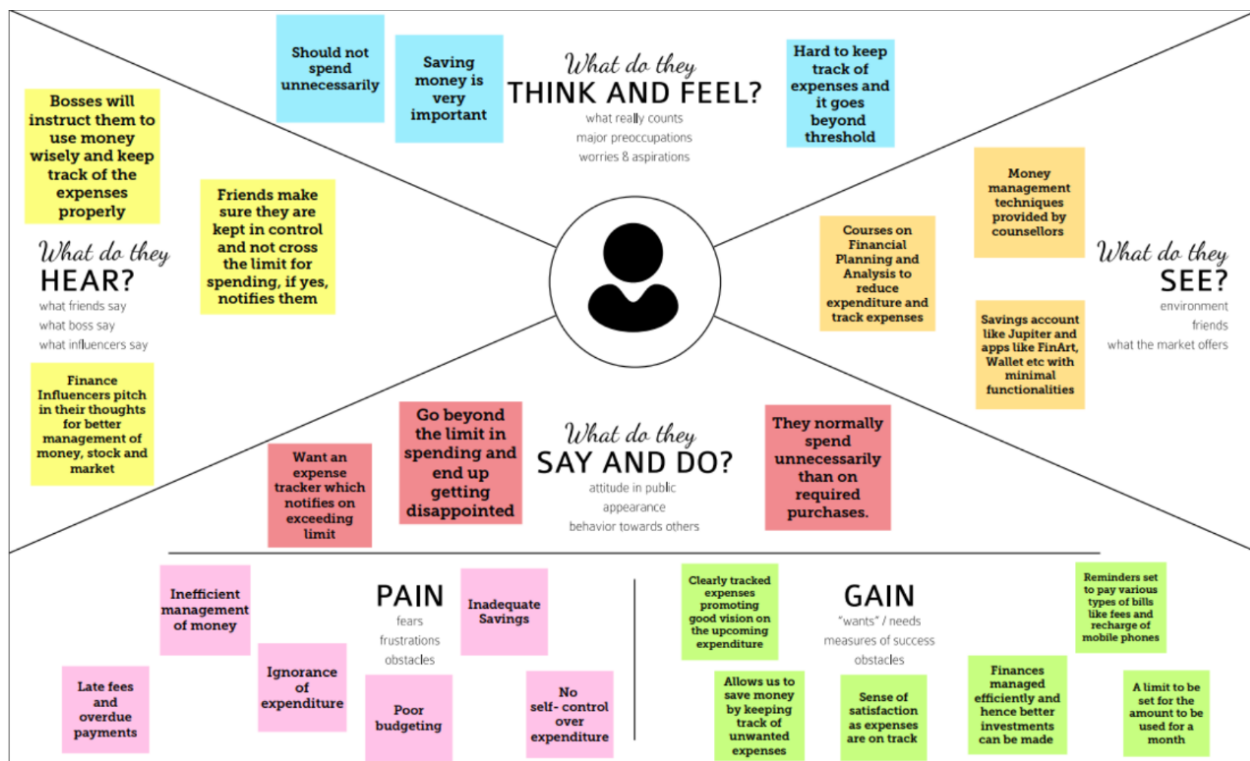
2.3 PROBLEM STATEMENT DEFINITION

Developing a Personal Expense Tracker Web Application and deploying It to Cloud, which 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

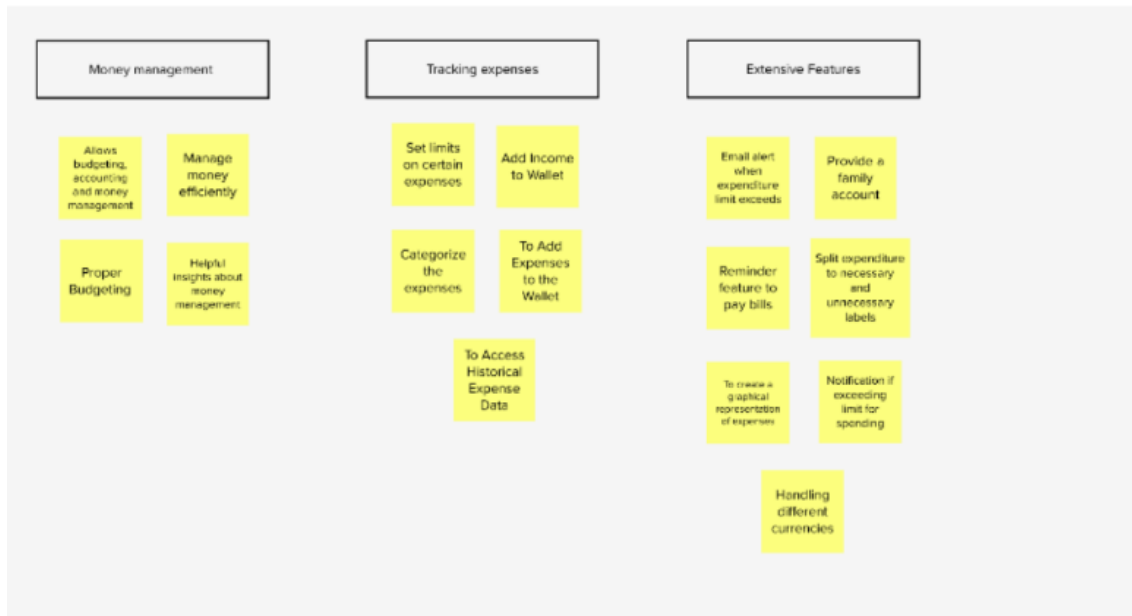
can get an analysis of their expenditure in graphical forms. They have an option to set a limit for the amount to be used for that particular month if the limit is exceeded the user will be notified with an email alert.

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION AND BRAINSTORMING



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Person 1



Person 2

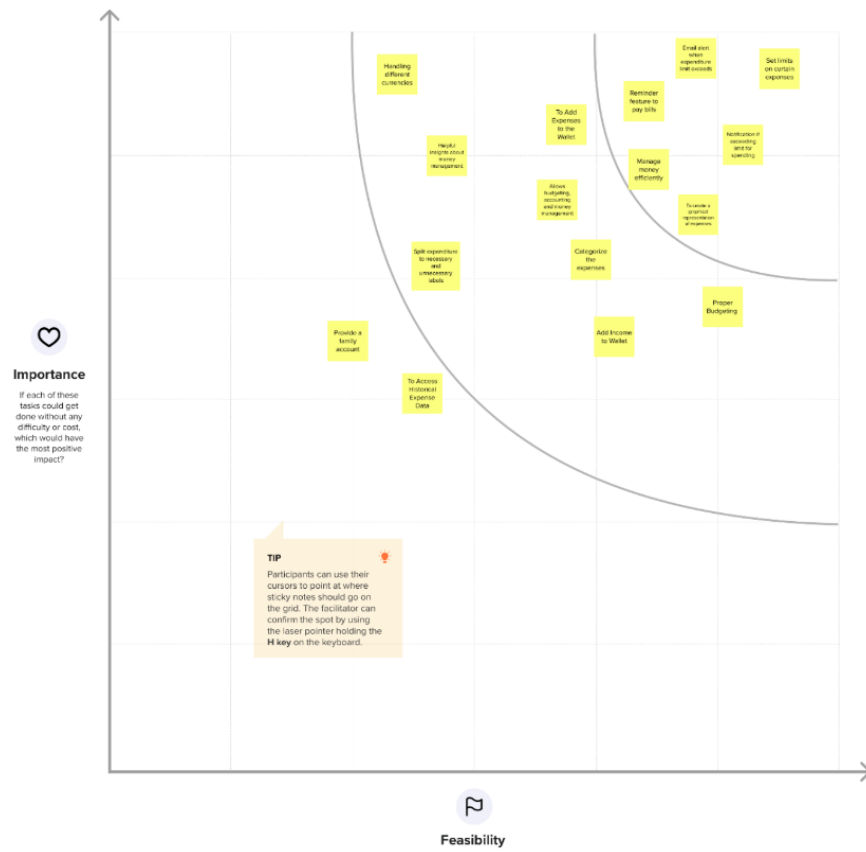


Person 3



Person 4





3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	It is hard for a spendthrift employee who is trying to track his expenses and manage his finances efficiently to save money, but there is no web application available with extensive features as most of the existing ones are mobile applications and are devoid of recommendation features and prediction of unnecessary expenditure.
2.	Idea / Solution description	A personal expense tracker web application which 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. A notification will be sent as a mail if the budget is crossed, paving the way for efficient money management.
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> - Reminder feature for paying bills - AI features using IBM Watson (Chatbot) for providing recommendations on efficient money management - Family wise account for tracking expenditure - Precise analytics for prediction
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> - Helps people to track their expenses and cut off unnecessary spending - Helps prevent impulse purchases saving more money - Helps manage the budget but allowing the user to allocate funds for recurring expenses - Remind users to pay the recurring bills reducing penalty fees - Efficient money management paving way for adequate savings

5.	Business Model (Revenue Model)	This application can support advertisements which can be a source of revenue for the issuers and in turn to remove the ads can make a small pricing for the customers.
6.	Scalability of the Solution	This application can further be extended in terms of adding features where billing and taxations could be handled, with smart sensing of expenditure through automatic extraction of information from photographs of printed bills.

3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) Students and adults with income who want to keep track of their expenses, save to buy something or to invest. CS	6. CUSTOMER CONSTRAINTS Spending power, budget, no cash, network connection, available devices, trust, confidentiality, sense of individuality. CC	5. AVAILABLE SOLUTIONS Customers have turned to people for help ie hiring someone to keep track of their personal expenses and give them suitable recommendations. Hired professionals could easily communicate what needs to be done. Hiring professionals cost a lot of money and some people tend to cheat their clients. AS	Explore AS, different
	2. JOBS-TO-BE-DONE / PROBLEMS Logging expenses, recommendations on expenditure, reminders and checks for needed transactions, spending limit, working in a different country or planning a trip requires currency change and limited budget. JP	9. PROBLEM ROOT CAUSE What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations. The root cause of this problem arises from money and how others are able to live in a better fashion with the same salary. Able to spend your money effectively and being able to buy what you want is the root causes of these problems. RC	7. BEHAVIOUR Communicate with their colleagues what kind of app they used and how useful it was and they will start using the app. They also use the Internet to get online reviews of the app and accordingly plan their. BE	
Focus on J&P, tap into BE, understand RC	3. TRIGGERS Seeing our colleague being able to manage his budget efficiently through the recommendations and expenditure tracking via this application triggers others to utilize the benefits and manage money effectively too. TR	10. YOUR SOLUTION The current solution involves having limits to expenses, adding expenses, adding income, grouping incomes, providing visualizations on how income has been spent. Our solution on top of the existing solution involves reminder feature for paying bills, building a checkbox for recommendations and precise analytics for prediction. SL	8. CHANNELS OF BEHAVIOUR 8.1 ONLINE Using the internet and many online forums like quora of how people are solving their issues relating to money expenditure. 8.2 OFFLINE Talking to your friends and family of how they solve these issues and applying them to your life. Hiring professionals who are able to give solid advice based on your problem. CH	Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

4. EMOTIONS: BEFORE / AFTER Customers feel lost and indecisive and end up becoming stingy and hence cannot spend their money effectively. After using the app customers get a sense of satisfaction and becoming confident and are able to spend their money and lead happy lives. EM		
--	--	--

Requirement Analysis

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form (UI) Store in Database
FR-2	User Login	Login portal Verification of username and password Error Message Display
FR-3	Adding Expenses	Expenses updated through form (in dashboard)
FR-4	Updating Expense Wallet	Incrementing/Decrementing balance in wallet
FR-5	Graphical representation of expenses	Generating charts based on historical expense data
FR-6	Email Notification	Checking of Budget exceeding limit Sending mail to user
FR-7	Chatbot Services	Training IBM Watson Query processing and accessing database Appropriate response generation

4.2 Non Functional Requirements

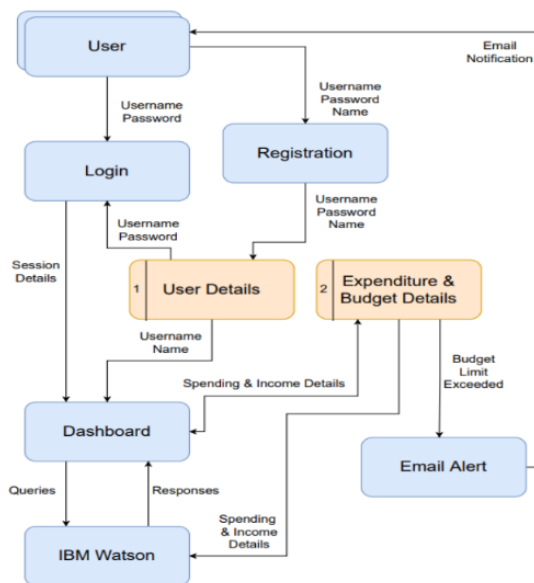
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	This product must be user-friendly by incorporating various features in the UI like navigation views, error messages being displayed if information entered is incorrect, tooltip texts provided for ease of use and accessibility features.
NFR-2	Security	This product ensures data is being secured, stored in IBM DB2 with user access control features, authentication involved during registration and login, users can only access their own expense data.
NFR-3	Reliability	Users can access their expense data and modify them 99 percent of the time without failure. This application is reliable as transactions and updating the expense wallet function supports atomicity and checks for irregularity or malfunction periodically.

NFR-4	Performance	This application consists of pages which must open within 2 seconds, the load time is reduced. This system responds back with accurate messages and error notifications under different load conditions. Rendering of data from the database must be faster including the concept of indexing in the database, thus keeping less response time.
NFR-5	Availability	This application is expected to have low down time as it is hosted on Kubernetes, being a distributed system. It should be made available 99.9 percent of time for users to access expense data and wallets
NFR-6	Scalability	This application can be scaled using horizontal scaling where more machines are added to the pool of servers to support more number of users at a particular instance of time.

PROJECT DESIGN

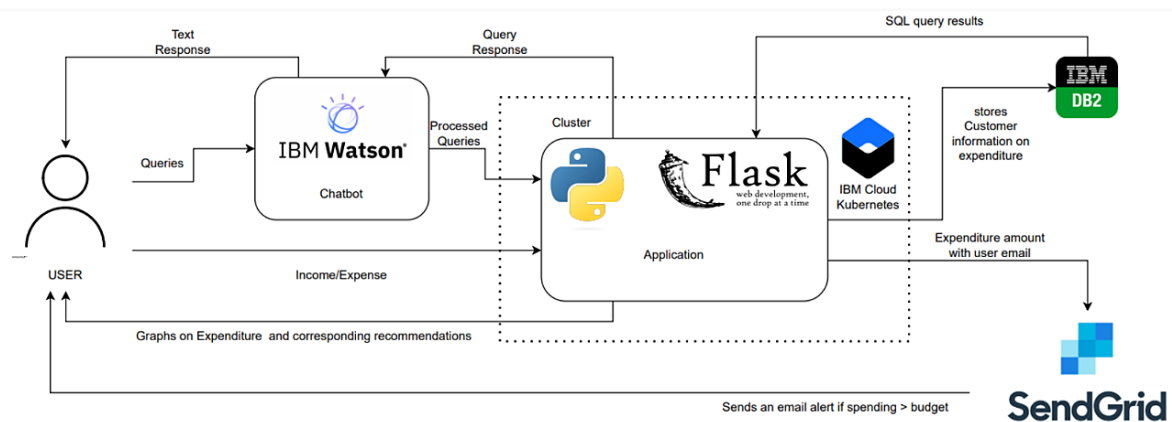
5.1 DATA FLOW DIAGRAM

Data Flow Diagram:

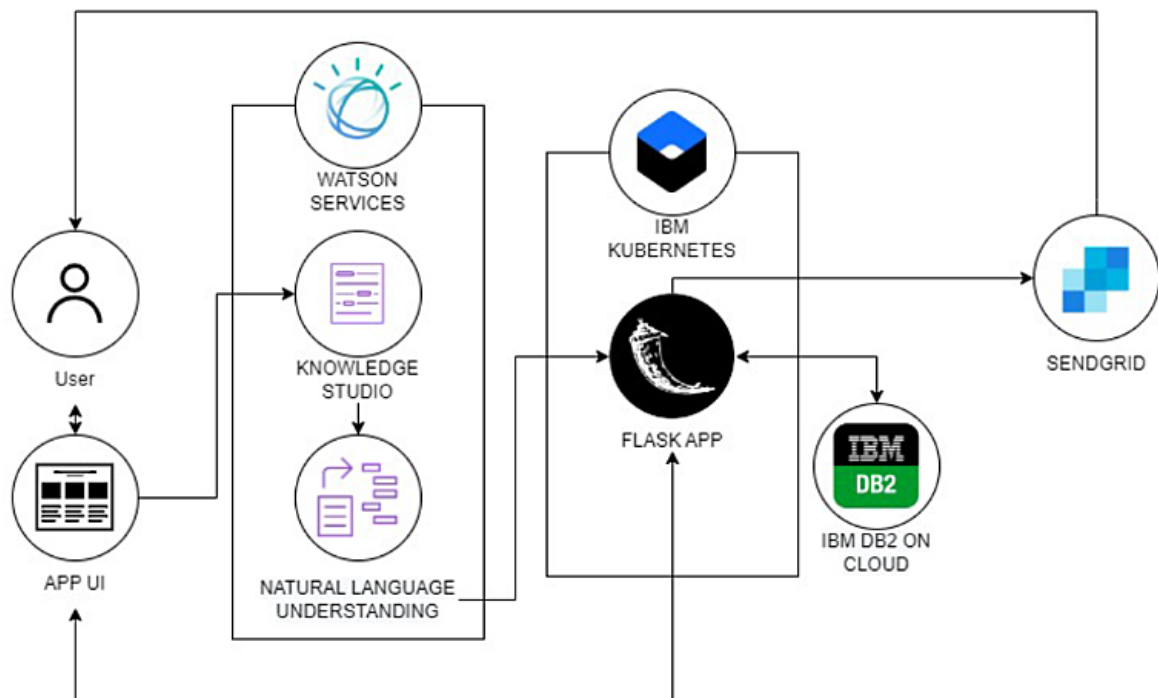


5.2 SOLUTION AND TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE - PERSONAL EXPENSE TRACKER APPLICATION



TECHNICAL ARCHITECTURE - PERSONAL EXPENSE TRACKER APPLICATION



PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	5	High	Nithinrhaj S Paarush S
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	6	High	Ramya N Sidarh S
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	4	High	Nithinrhaj S Paarush S Ramya N Sidarh S
Sprint-2	Dashboard	USN-4	As a user, I can access my dashboard after logging into the application	8	High	Nithinrhaj S Paarush S
Sprint-2		USN-5	As a user, I can view my past expenditure	8	High	Ramya N Sidarh S

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2		USN-6	As a user, I can add my expenses in the wallet	7	High	Nithinrhaj S Paarush S
Sprint-2		USN-7	As a user, I can add income	7	High	Ramya N Sidarh S
Sprint-3		USN-8	As a user, I can get an analysis of my expenditure in graphical forms	10	High	Nithinrhaj S Paarush S
Sprint-3		USN-9	As a user, I can add my recurring expenses	5	Low	Ramya N Sidarh S
Sprint-4	Chatbot	USN-10	As a user, I can ask the chatbot about any upcoming bills / expenses	10	Low	Nithinrhaj S Paarush S Ramya N Sidarh S
Sprint-4		USN-11	As a user, I can ask the chatbot about my current monthly expenditure	6	Low	Ramya N Sidarh S
Sprint-4		USN-12	As a user, I can ask the chatbot about my historical monthly expenditure	5	Low	Nithinrhaj S Ramya N

6.2 SPRINT DELIVERY SCHEDULE

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	15	6 Days	24 Oct 2022	29 Oct 2022	15	29 Oct 2022
Sprint-2	30	6 Days	31 Oct 2022	05 Nov 2022	30	05 Nov 2022
Sprint-3	15	6 Days	07 Nov 2022	12 Nov 2022	15	12 Nov 2022
Sprint-4	21	6 Days	14 Nov 2022	19 Nov 2022	11	19 Nov 2022

Velocity:

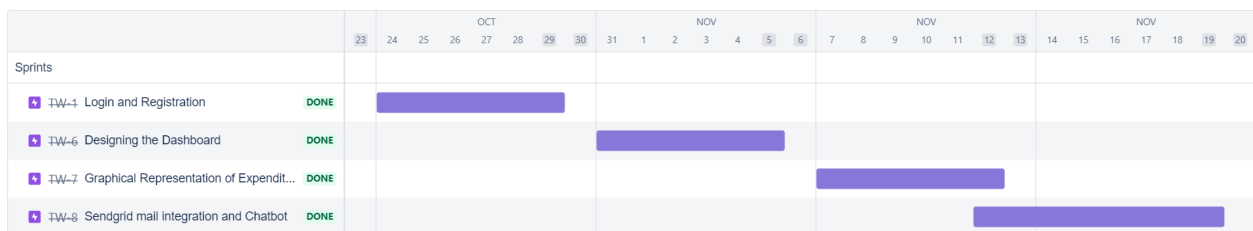
AV (Sprint 1) = 2.5

AV (Sprint 2) = 5

AV (Sprint 3) = 2.5

AV (Sprint 4) = 3.5

6.3 REPORTS FROM JIRA



CODING AND SOLUTIONING

7.1 LOGIN AND REGISTRATION

Source Code:

```
<!DOCTYPEhtml>
<html lang="en" dir="ltr">
  <head>
```

```
<metacharset="UTF-8">
<title>Personal Expense Tracker Login</title>
<linkrel="stylesheet"href="{{ url_for('static',
filename='css/style.login.css') }}" />

<linkrel="stylesheet"href="https://cdnjs.cloudflare.com/ajax/libs
/font-awesome/5.15.3/css/all.min.css"/>
    <metaname="viewport"content="width=device-width,
initial-scale=1.0" />
</head>
<body>
    <divclass="container">
        <inputtype="checkbox"id="flip">
        <divclass="cover">
            <divclass="front">
                <divclass="text">
                </div>
            </div>
            <divclass="back">
                <divclass="text">
                </div>
            </div>
        </div>
    </div>
```

```

<divclass="forms">
  <divclass="form-content">
    <divclass="login-form">
      <divclass="title">Login</div>
      <formonsubmit="submitLoginForm(event)">
        <divclass="input-boxes">
          <divclass="input-box">
            <i>class="fas fa-envelope"</i>

<inputtype="text" id="email" name="email" placeholder="Enter
your email" required>
          </div>
          <divclass="input-box">
            <i>class="fas fa-lock"</i>

<inputtype="password" id="pw" name="pw" placeholder="Enter
your password" required/>
            <i>onclick="toggleShowPass(event)" style
= "padding-left:330px;" class="fa fa-eye"</i>
          </div>

          <divclass="button input-box">
            <inputtype="submit" value="Submit">

```

```

        </div>
        <divclass="text sign-up-text">Don't have an
account? <labelfor="flip">Signup now</label></div>
    </div>
</form>
</div>
<divclass="signup-form">
    <divclass="title">Signup</div>
    <formonsubmit="submitRegisterForm(event)">
        <divclass="input-boxes">
            <divclass="input-boxes">
                <divclass="input-box">
                    <iclass="fas fa-user"></i>

<inputtype="text" id="name" name="name" placeholder="Enter
your name" required>
                </div>
                <divclass="input-box">
                    <iclass="fas fa-envelope"></i>

<inputtype="text" id="email" name="email" placeholder="Enter
your email" required>
            </div>

```

```

        <divclass="input-box">
            <i>class="fas fa-lock"</i>

<inputtype="password" id="pass" name="pass" placeholder="Enter your password" required>
            <ionclick="toggleShowPass(event)" style
= "padding-left:330px;" class="fa fa-eye"></i>
        </div>
        <divclass="button input-box">
            <inputtype="submit" value="Submit">
        </div>
        <divclass="text sign-up-text">Already have
an account? <labelfor="flip">Login now</label></div>
    </div>
</form>
</div>
</div>
</div>
<scripttype="text/javascript" src="{ { url_for('static',
filename='js/script.login.js') } }"></script>
</body>
</html>

```

javascript file

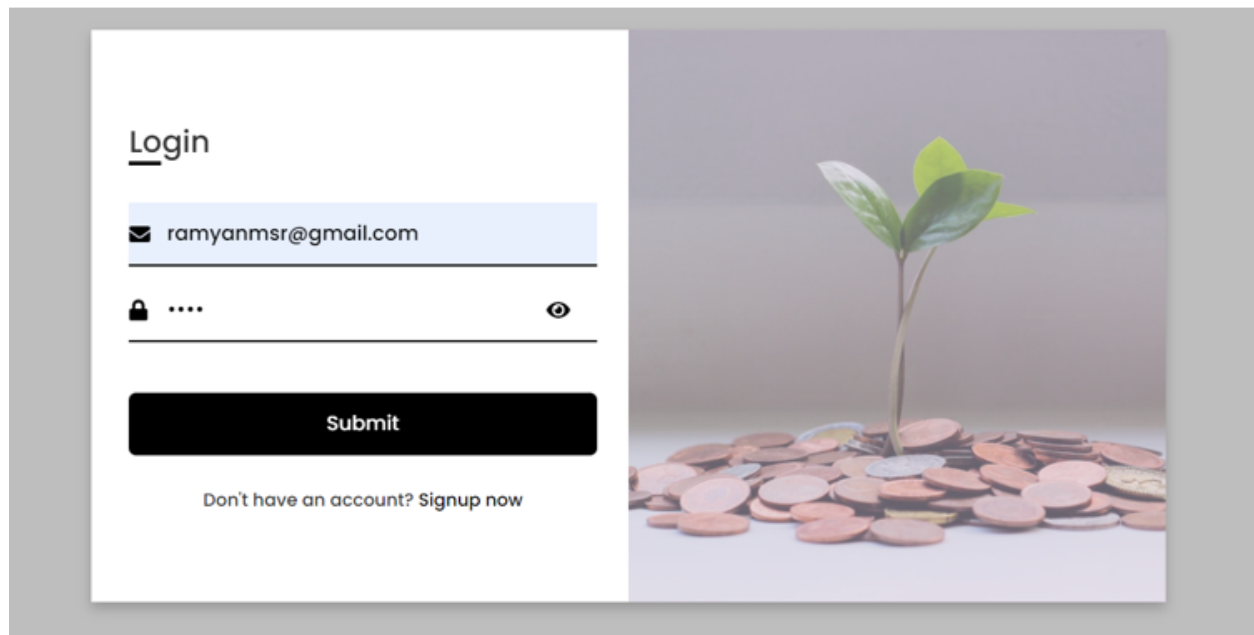
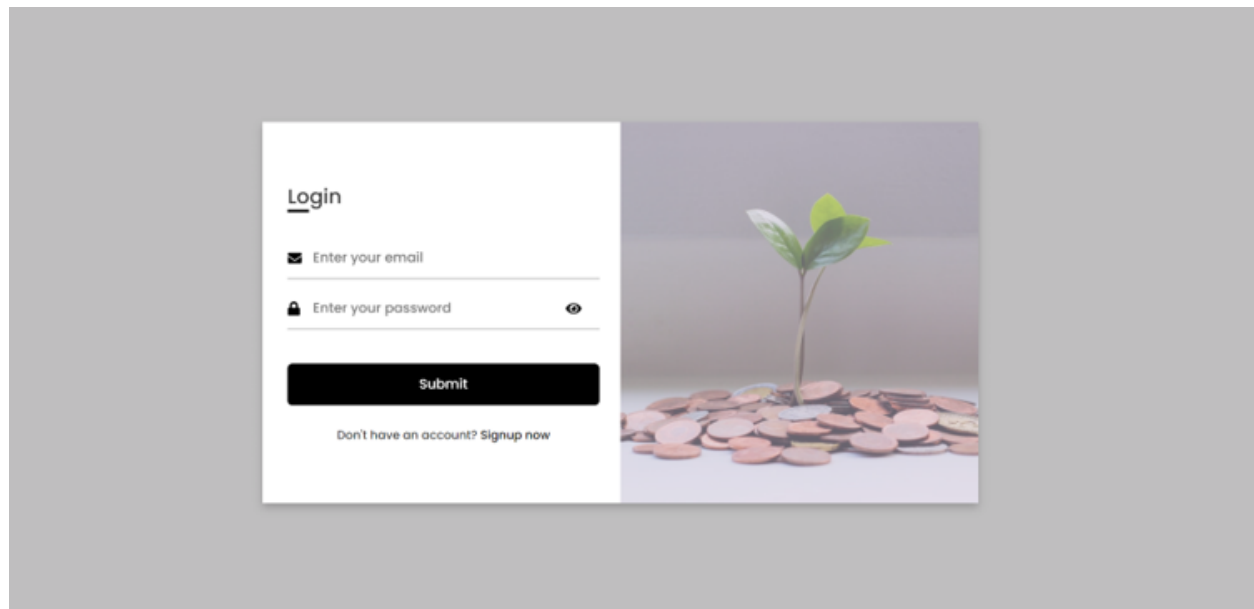
```
let toggleShowPass = (event) => {
  let x =
event.currentTarget.parentElement.querySelector("input");
  if (x.type === "password") {
    x.type = "text";
  } else {
    x.type = "password";
  }
}


const submitLoginForm = (e) => {
  e.preventDefault()
  let form = event.currentTarget;
  fetch("http://localhost:5000/api/login", {
    method: "POST",
    body: new FormData(form)
  })
  .then(body => body.json())
  .then((x) => {
    if (!x.Success) {
      alert(x.msg);
    } else {
```

```
        window.location = "http://localhost:5000/dashboard"
    }
})
}

const submitRegisterForm = (e) => {
    e.preventDefault()
    let form = event.currentTarget;
    fetch("http://localhost:5000/api/register", {
        method: "POST",
        body: new FormData(form)
    })
    .then(body => body.json())
    .then((x) => {
        if (!x.Success) {
            alert(x.msg);
        } else {
            window.location = "http://localhost:5000/"
        }
    })
}
```

OUTPUT





Signup


Enter your name

Enter your email

Enter your password ☐

Submit

[Already have an account? Login now](#)



Signup

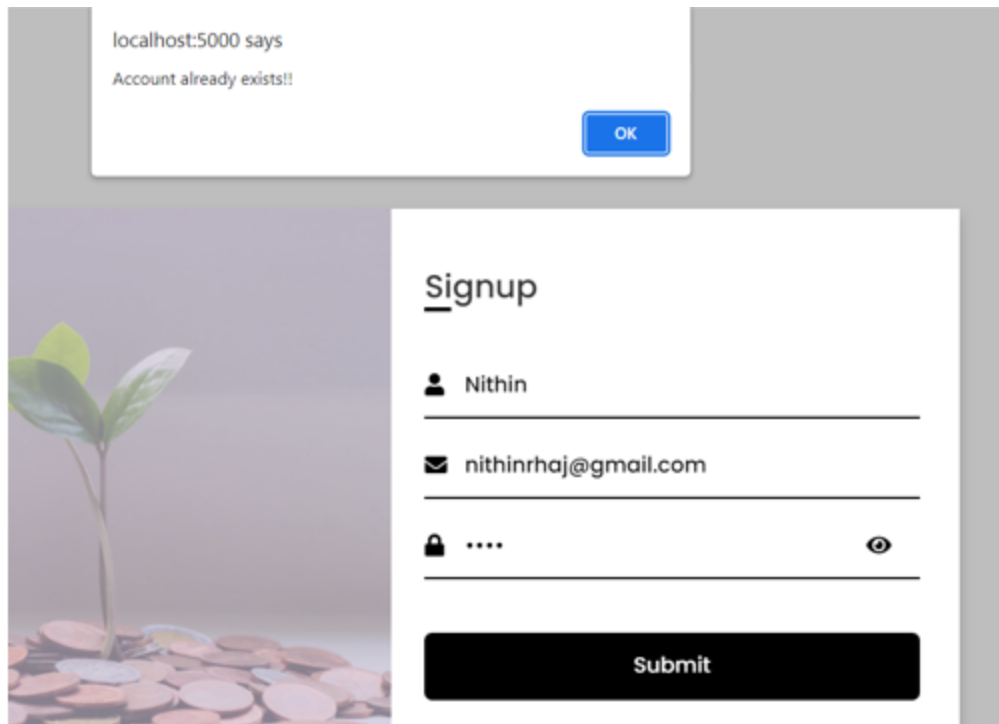
Nithin

nithinrhaj@gmail.com

.... ☐

Submit

[Already have an account? Login now](#)



7.2 DASHBBOARD

```
<!DOCTYPEhtml>
<htmllang="en">
  <head>
    <metacharset="UTF-8">
    <title>The Wallet</title>

    <linkrel="stylesheet"href="https://cdnjs.cloudflare.com/ajax/libs
/meyer-reset/2.0/reset.min.css">
    <linkrel="stylesheet"href="{ { url_for('static',
filename='css/style.dashboard.css') } }">
```

```

</head>
<bodyonload="main()">
  <!-- partial:index.partial.html -->
  <divclass="app">
    <headerclass="app-header">
      <divclass="app-header-logo">
        <divclass="logo">
          <spanclass="logo-icon">
<imgsrc="https://assets.codepen.io/285131/almeria-logo.svg"/>
          </span>
          <h1class="logo-title">
            <span>The Wallet</span>
            <span>Expense Tracker</span>
          </h1>
        </div>
      </div>
      <divclass="app-header-navigation">
        <divclass="tabs">
          <marqueedirection="right"><h1>If you are
saving, you are succeeding...</h1></marquee>
        </div>
      </div>
    </div>
  </div>

```

```

<divclass="app-header-actions">
  <buttonclass="user-profile">
    <spanid="user_name">Loading...</span>
    <span>
      <imgsrc="{ { url_for('static',
filename='images/profile.webp') } }"/>
    </span>
  </button>
  <divclass="app-header-actions-buttons">
    <buttonclass="icon-button large"
onclick="logout()">
      <iclass="ph-sign-out"></i>
    </button>
  </div>
</div>
<divclass="app-header-mobile">
  <buttonclass="icon-button large">
    <iclass="ph-list"></i>
  </button>
</div>
</header>
<divclass="app-body">
  <divclass="app-body-navigation">

```

```

<navclass="navigation">
  <ahref="/graph">
    <iclass="ph-browsers"></i>
    <span>Dashboard</span>
  </a>
  <ahref="/dashboard">
    <iclass="ph-check-square"></i>
    <span>Expenses</span>
  </a>
</nav>
<footerclass="footer">
  <h1>The Wallet<small>©</small></h1>
</footer>
</div>
<divclass="app-body-main-content">
  <sectionclass="service-section">
    <h2>Expenses</h2>

    <divclass="tiles">
      <articleclass="tile">
        <divclass="tile-header">
          <span>
            <imgsrc="{ { url_for('static',

```

```

filename='images/cutlery.jpg') }}"style="height: 65px;"/>
    </span>
    <h3>
        <span><b> Food </b></span>
    </h3>
</div>
<a href="#divOne">
    <span>Add expenses</span>
    <spanclass="icon-button">
        <iclass="ph-caret-right-bold"></i>
    </span>
</a>
</article>
<articleclass="tile">
    <divclass="tile-header">
        <span>
            <imgsrc="{{ url_for('static',
filename='images/entertainment.jpg') }}"style="height: 65px;"/>
        </span>
        <h3>
            <span>Entertainment</span>
        </h3>
    </div>

```

```

        <a href="#divOne">
            <span>Add expenses</span>
            <span class="icon-button">
                <i class="ph-caret-right-bold"></i>
            </span>
        </a>
    </article>
    <article class="tile">
        <div class="tile-header">
            <span>
                
            </span>
            <h3>
                <span>Shopping</span>
            </h3>
        </div>
        <a href="#divOne">
            <span>Add expenses</span>
            <span class="icon-button">

```



```

        <iclass="ph-caret-right-
bold"></i>

        </span>
    </a>
</article>
<articleclass="tile">
    <divclass="tile-header">
        <span>
            <imgsrc="{ { url_for('static',
filename='images/EMI.jpg') }}"style="height: 65px;"/>
        </span>
        <h3>
            <span>EMI</span>
        </h3>
    </div>
    <ahref="#divOne">
        <span>Add expenses</span>
        <spanclass="icon-button">
            <iclass="ph-caret-right-
bold"></i>

```

```

        </span>
    </a>
</article>
<articleclass="tile">
    <divclass="tile-header">
        <span>
            <imgsrc="{ { url_for('static',
filename='images/house.jpeg') } }"style="height: 65px;"/>
        </span>
        <h3>
            <span>Rent</span>
        </h3>
    </div>
    <a href="#divOne">
        <span>Add expenses</span>
        <spanclass="icon-button">
            <iclass="ph-caret-right-
bold"></i>
        </span>
    </a>

```

```

</article>
<articleclass="tile">
  <divclass="tile-header">
    <span>
      <imgsrc="{ { url_for('static',
filename='images/misc.jpg') }}"style="height: 65px;"/>
    </span>
    <h3>
      <span>Others</span>
    </h3>
  </div>
  <a href="#divOne">
    <span>Add expenses</span>
    <spanclass="icon-button">
      <iclass="ph-caret-right-
bold"></i>
    </span>
  </a>
</article>
</div>

```

```

</section>
<sectionclass="transfer-section">
  <divclass="transfer-section-header">
    <h2>Recent Expenses</h2>
    <divclass="filter-options">
      <p>Filter </p>
      <buttonclass="icon-
button"onclick="reverseTransactions()">
        <iclass="ph-funnel"></i>
      </button>
    </div>
  </div>
  <divclass="transfers">
    </div>
  </section>
</div>
<divclass="app-body-sidebar">
  <sectionclass="payment-section">
    <h2>Digital Wallet</h2>
    <divclass="payment-section-header">

```

```

    <p>Saving, Investing and Tracking</p>
</div>

<divclass="payments">
    <divclass="payment">
        <divclass="card green">
            <span>&nbsp;</span>
            <span>
                <imgsrc="{{ url_for('static',
filename='images/wallet2.png') }}"style="height:
60px;"/>
            </span>
            <span>
                Balance
            </span>
        </div>
        <divclass="payment-details">
            <spanstyle="display: flex;justify-
content: space-between;">
                <h3>Wallet Balance</h3>

```

```
<buttononclick="openBalPopup()"id="myBtn">Add to
Wallet</button>
```

```
</span>
```

```
<div>
```

```
<spanid="balance">Loading...</span>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<divclass="faq">
```

```
<h1>Frequently Asked
```

```
Questions</h1>
```

```
<div>
```

```
<label>Do you lay a strong
foundation for long-term earnings and wealth?</label>
```

```
<spanstyle="color:#D1C1C1"align="justify">Take every
learning during college as your investment for the future.
Missing out on even one of those can seriously fritter
```

away that investment.

</div>

<div>

<label>What financial mistakes
should one refrain from?</label>

<spanstyle="color:#D1C1C1"align="justify">Ensure that
your under graduation loan amount should not be more
than your first year salary. Another mistake to avoid is
playing with credit card.

</div>

<div>

<label>How to manage debt and
boost your credit score?</label>

<spanstyle="color:#D1C1C1"align="justify">Do not be
influenced by your peers, but stick to maintaining your
own budget. Take up any campus employment to balance
your expenditures and make a saving, along with offering
and gaining valued experience.

```

        </div>
    </div>
</div>
</section>
</div>
</div>
<div id="myModal" class="modal">
    <!-- Modal content -->
    <div class="modal-content">

<span class="close" onclick="delParent(event)">&times;</span>

<form onsubmit="updateBalance(event)" style="color:black">

        <h1 style="font-weight: bolder; font-size:
x-large;"><center>Modify Wallet Balance</center></h1>
        <br/>
        <div style="margin-left:185px; margin-top:
2px;">

```



```

        <label for="amount" style="font-size:
large;"><b>Amount : </b></label>
        <input type="number" style="margin-
left:1px" placeholder="Enter Amount"
name="amount" required />

<button type="submit" class="btn">Add</button>
    </div>
</form>
</div>
</div>
<div class="overlay" id="divOne">
    <div class="wrapper">
        <h2>Please Fill up The Expenses
Form</h2><a class="close" href="#">&times;</
a>

        <div class="content">
            <div class="container">

```

```
<form onsubmit="addExpense(event)">
    <label>Expense
Category</label>
    <select name="category">

<option value="food">Food</option>

<option value="entertainment">Entertainment</
option>

<option value="shopping">Shopping</option>

<option value="emi">EMI</option>

<option value="rent">Rent</option>

<option value="others">Others</option>
```

```

        </select>
        <label>Date</label><br/>

<inputclass="popup"placeholder="Date.."type=
"date"required name="date">

        <label>Location</label>
        <inputplaceholder="Place
of Purchase..." type="text"class="popup">

<label>Amount</label><br/>

<inputclass="popup"type="number"placeholder
="Enter Amount..." name="amount"required />

<inputclass="popup"type="submit"value="Add
Expense">

        </form>

```

```

        </div>
    </div>
</div>
</div>
<!-- partial -->
<scriptsrc='https://unpkg.com/phosphor-
icons'></script>

<scriptsrc="https://unpkg.com/@babel/standalo
ne/babel.min.js"></script>

    <scripttype="text/babel"src="{ {
url_for('static',
filename='/js/script.dashboard.js') }}"></script>
<script>

    window.watsonAssistantChatOptions
= {

        integrationID:"8f7293cb-e55e-49e7-
```

```
a937-915057242eca", // The ID of this
integration.

    region:"au-syd", // The region your
integration is hosted in.

    serviceInstanceID:"68106b4c-302a-
41d4-bdf5-00458ba09a25", // The ID of your
service instance.

    onLoad:function(instance) {
instance.render(); }

};

setTimeout(function(){

constt=document.createElement('script');

    t.src="https://web-
chat.global.assistant.watson.appdomain.cloud/v
ersions/" +
(window.watsonAssistantChatOptions.clientVer
```

```

sion || 'latest') + "/WatsonAssistantChatEntry.js";
        document.head.appendChild(t);
    });
</script>

</body>
</html>

```

Java Script File to work as API:

```

/** @jsx createElement */
/** @jsxFragment createFragment */

const createElement = (tag, props, ...children)
=> {
    if (typeof tag === "function") return
tag(props, ...children);
    const element = document.createElement(tag);

```

```
    Object.entries(props || {}).forEach(([name,
value]) => {
        if (name.startsWith("on") &&
name.toLowerCase() in window)

element.addEventListener(name.toLowerCase().
substr(2), value);
        else element.setAttribute(name,
value.toString());
    });

    children.forEach(child=> {
        appendChild(element, child);
    });

    return element;
};
```

```
const appendChild = (parent, child) => {  
  if (Array.isArray(child))  
    child.forEach(nestedChild=>  
appendChild(parent, nestedChild));  
  else  
    parent.appendChild(child.nodeType ? child  
: document.createTextNode(child));  
};  
  
const createFragment = (props, ...children) => {  
  return children;  
};  
  
const displayBalance = () => {  
  let val = document.querySelector("#balance")  
  fetch("http://localhost:5000/api/balance")
```



```

.then(body=>body.json())
.then((x)=>{
    val.innerHTML=x.balance;
})
}

const displayName = () => {
    let val =
document.querySelector("#user_name")
    fetch("http://localhost:5000/api/session")
    .then(body=>body.json())
    .then((x)=>{
        if ("name" in x){
            val.innerHTML=x.name;
        } else {
            window.location =
"http://localhost:5000"

```

```

    }

  })
}

const Transaction = (props) => (
  <div class="transfer">
    <div class="transfer-logo">

    </div>
    <dl class="transfer-details">
      <div>
        <dt>{props? props.category :
"other"}</dt>
        <dd>{props? props.category :
"other"}</dd>

```

```

    </div>

    <div>
      <dt>{props? props.date : ""}</dt>
      <dd>Date payment</dd>
    </div>
  </dl>

  <divclass="transfer-number">
    {props? props.amount : 0}
  </div>
</div>
)

const reverseTransactions = () => {

displayTransactions(displayTransactions.reverse
);
}

```

```
const displayTransactions = (bool) => {  
  if (bool===true) {  
    displayTransactions.reverse = false;  
  }  
  if (bool===false){  
    displayTransactions.reverse = true;  
  }  
  letval = document.querySelector("#balance")  
  fetch("http://localhost:5000/api/expenses")  
  .then(body=>body.json())  
  .then((x)=>{  
    letnode =  
document.querySelector(".transfer-section")  
    if (displayTransactions.reverse){  
      x = x.reverse();  
    }  
  })  
}
```

```

node
  .replaceChild(
    <divclass="transfers">

{x.map(row=>(<Transactioncategory={row.cate
gory}date={row.date}amount={row.amount}/>
))}

    </div>
    ,node.querySelector(".transfers"))
  })
}

const delParent = (e) => {
  e.currentTarget.parentElement.remove();
}

const updateBalance = (e) => {

```

```
e.preventDefault()
let form = event.currentTarget;
let form_data = new FormData();
form_data.append("category", "others")
let yourDate = new Date()
form_data.append("date",
yourDate.toISOString().split('T')[0])
fetch("http://localhost:5000/api/balance")
.then(body => body.json())
.then((x) => {
    form_data.append("amount", +(x.balance) -
form.querySelector("input[name='amount']").va
lue)
    fetch("http://localhost:5000/api/expenses",
{
        method: "POST",
        body: form_data
```

```
    })  
    .then(body=>body.json())  
    .then((x)=>{  
        if (!x.Success){  
            //alert(x.msg);  
            main();  
        } else {  
            window.location =  
"http://localhost:5000/dashboard"  
        }  
    })  
})  
}  
  
const addExpense = (e) => {  
    e.preventDefault()  
    letform = event.currentTarget;
```

```

fetch("http://localhost:5000/api/expenses", {
  method:"POST",
  body:new FormData(form)
})
.then(body=>body.json())
.then((x)=>{
  if (!x.Success){
    //alert(x.msg);
    main();
  } else {
    window.location =
"http://localhost:5000/dashboard"
  }
})
}

const openBalPopup = (e) => {

```



```
var modal =  
document.getElementById("myModal");  
var btn =  
document.getElementById("myBtn");  
var span =  
document.getElementsByClassName("close")[0]  
];  
btn.onclick = function() {  
    modal.style.display = "block";  
  
document.querySelector("body").style.overflow  
="hidden";  
}  
span.onclick = function() {  
    modal.style.display = "none";  
  
document.querySelector("body").style.overflow
```

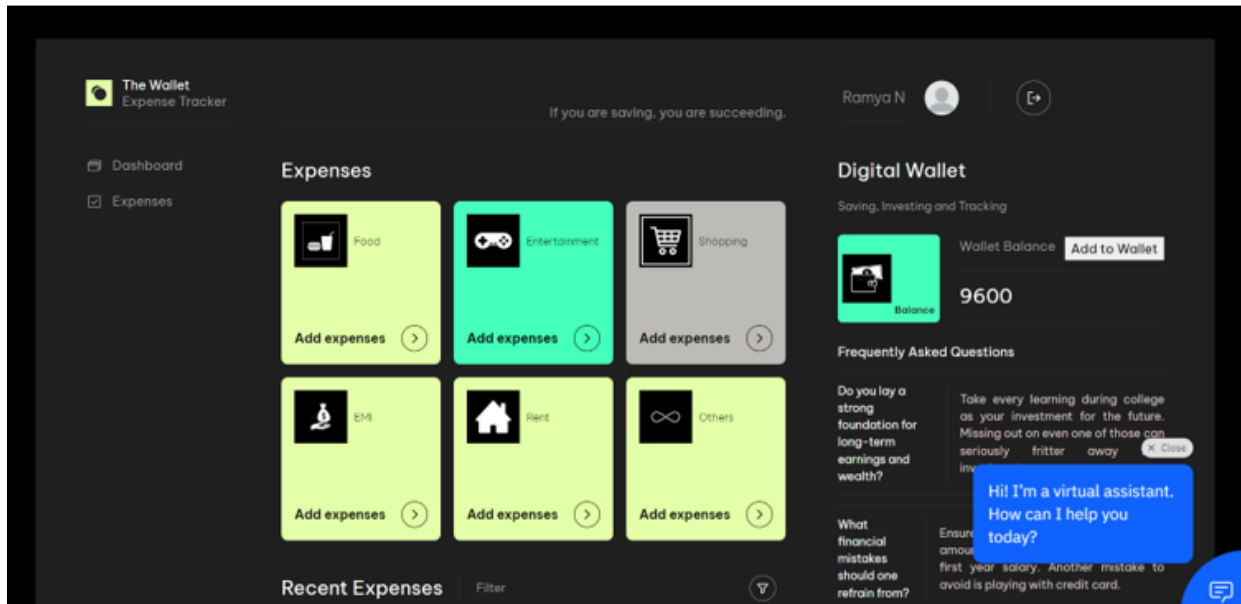
```
= "unset";  
  }  
  
}  
  
const logout = () => {  
  fetch("http://localhost:5000/api/session",{  
    method: "DELETE"  
  })  
  .then(() => {  
    window.location = "http://localhost:5000/"  
  })  
}  
  
const main = () => {  
  displayBalance()  
  displayTransactions.reverse = true;
```

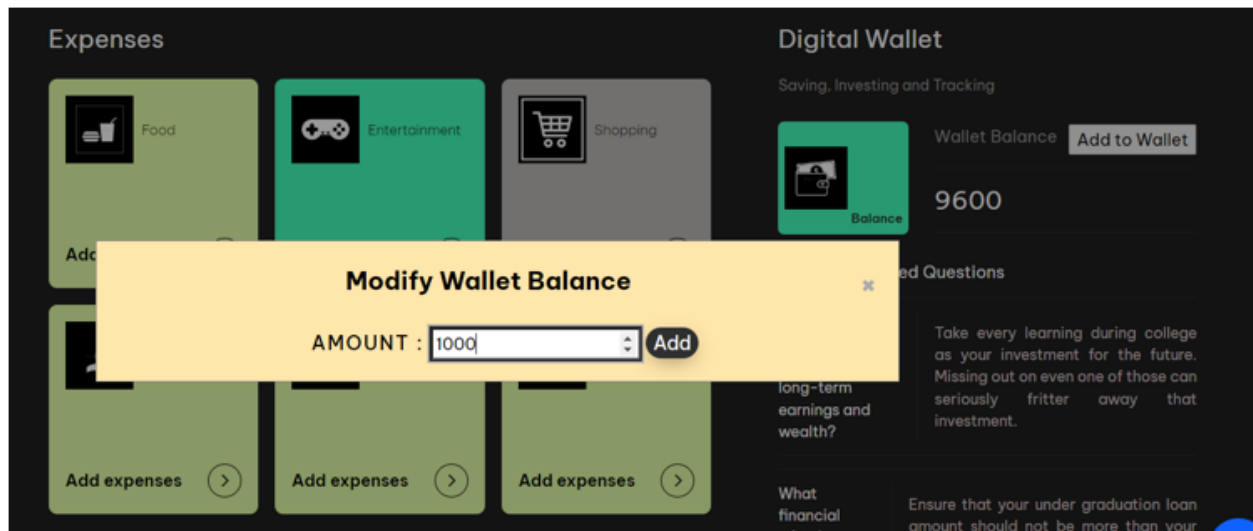
```
displayTransactions()
displayName()
}

window.onload = main

main()
```

OUTPUT





Please Fill up The Expenses Form

EXPENSE CATEGORY

Food

DATE

dd-mm-yyyy

LOCATION

Place of Purchase...

AMOUNT

Enter Amount...

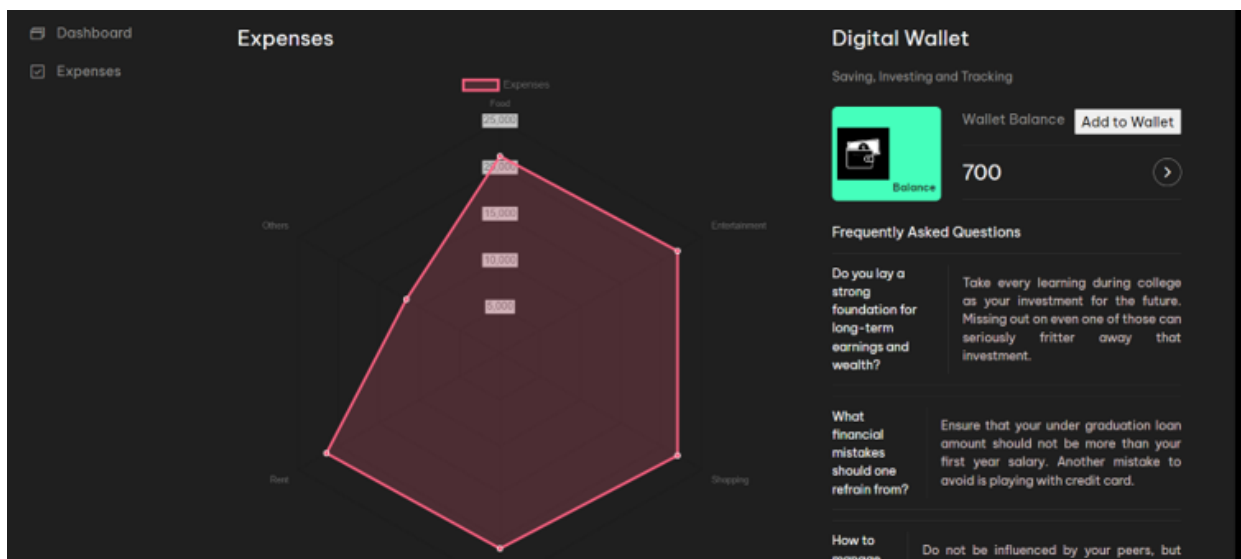
ADD EXPENSE

Add expenses

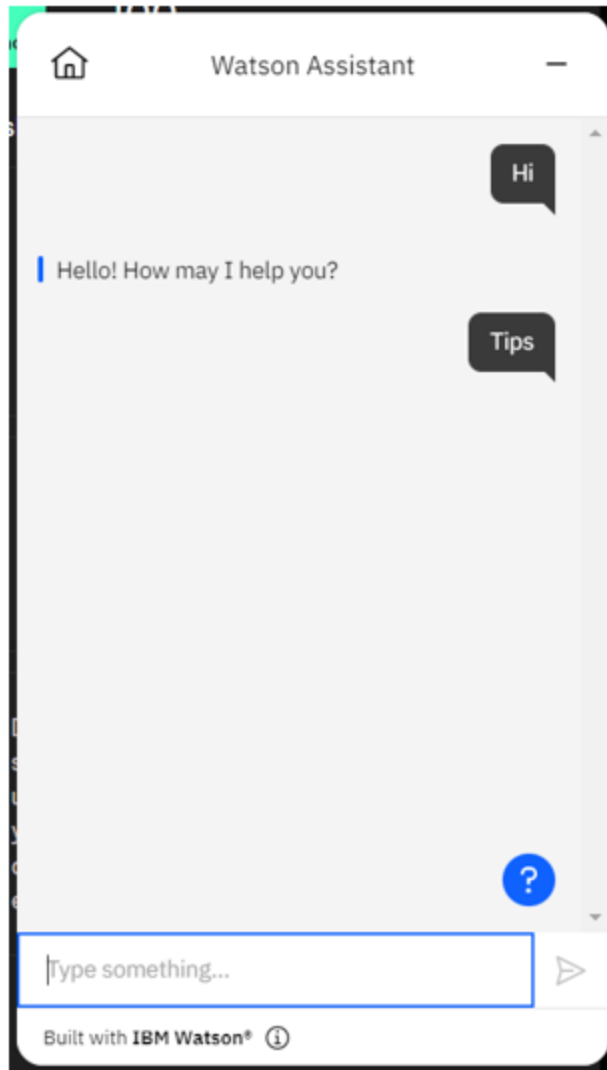
Add expenses

Recent Expenses				<div> <div>should one refrain from?</div> <div>first year salary. Another mistake to avoid is playing with credit card.</div> </div>	
Filter					
\$	food	2022-11-10	-300	<div>How to manage debt and boost your credit score?</div> <div>Do not be influenced by your peers, but stick to maintaining your own budget. Take up any campus employment to balance your expenditures and make a saving, along with offering and gaining valued experience.</div>	
\$	others	2022-11-19	0		
\$	others	2022-11-19	-8600		
\$	food	2022-11-10	-400		
\$	others	2022-11-18	10000		
\$	others	2022-11-18	10000		
\$	rent	2022-11-17	-20000		
\$	emi	2022-11-17	-20000		
\$	shopping	2022-11-17	-20000		

7.3. VISUALIZATION CHARTS



7.4 WATSON CHAT BOT



7.5 EMAIL SERVICE

Source Code

```

166 def send(user, email):
167     print(email)
168     sql_stmt="SELECT SUM(amount) as balance from expenses where expenses.user=?"
169     stmt=ibm_db.prepare(conn,sql_stmt)
170     ibm_db.bind_param(stmt,1,user)
171     ibm_db.execute(stmt)
172     row = ibm_db.fetch_assoc(stmt)
173     ret = {}
174     if row != False and row["BALANCE"]!=None and row["BALANCE"]>0:
175         message=Mail(from_email="curusamsr@gmail.com",
176                     to_email=email,
177                     subject='ALERT!!! Budget Exceeded!',
178                     plain_text_content='Budget has been crossed for this month',
179                     html_content='<strong>Hi user,<br>You have crossed your budget for this
180
181         try:
182             sg = SendGridAPIClient(os.environ.get("SENDGRID_API_KEY"))
183             print(os.environ.get("SENDGRID_API_KEY"))
184             response=sg.send(message)
185             print(response.status_code)
186             print(response.body)
187             print(response.headers)
188         except Exception as e:
189             print(e)
190
191 if __name__ == "__main__":
192     app.run(host="0.0.0.0", debug=True)

```

OUTPUT

If expenditure exceeds the budget, a notification will be sent as an email:

```

ramyanmsr@gmail.com
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
127.0.0.1 - - [19/Nov/2022 14:22:33] "GET /api/balance HTTP/1.1" 200 -
202
b''
Server: nginx
Date: Sat, 19 Nov 2022 08:52:43 GMT
Content-Length: 0
Connection: close
X-Message-Id: udinYwVjRz2ztHETaAMCda
Access-Control-Allow-Origin: https://sendgrid.api-docs.io
Access-Control-Allow-Methods: POST
Access-Control-Allow-Headers: Authorization, Content-Type, On-behalf-of, x-sg-elias-acl
Access-Control-Max-Age: 600
X-No-CORS-Reason: https://sendgrid.com/docs/Classroom/Basics/API/cors.html
Strict-Transport-Security: max-age=600; includeSubDomains

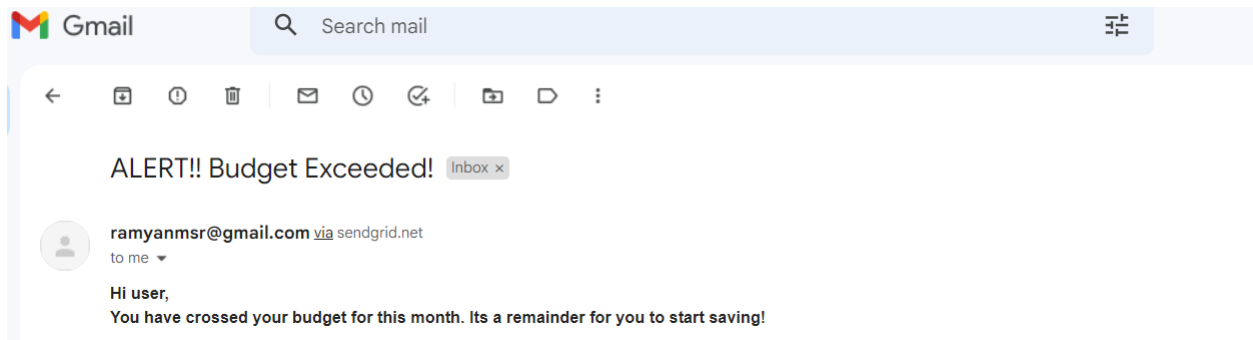
```

ramyanmsr@gmail.com via sendgrid.net Thu, Nov 17, 11:50 PM (2 days ago) ☆ ↶ ⓘ

to me

Hi user,
You have crossed your budget for this month. ---you have made max expenditure in so and so category! Its a reminder for you to start saving!

Reply Forward



7.5 DATABASE SCHEMA

Tables were created in IBM Db2 and accessed in the flask files for processing the information.

Tables			New table +	Filter	Sort	More	Close
<input type="checkbox"/> Name ▾	Schema	Properties					
<input type="checkbox"/> ASSIGN	DGP98488	...					
<input type="checkbox"/> EXPENSES	DGP98488	...					
<input type="checkbox"/> TEST	DGP98488	...					
<input type="checkbox"/> USER	DGP98488	...					

Table definition

TEST

Approximate 14 rows (32.0 KB)
Updated on 2022-11-18 17:44:41

Name	Data type	Nullable	Length	Scale	
EMAIL	VARCHAR	Y	320	0	👁
USERNAME	VARCHAR	N	320	0	👁
PASSWORD	CHAR	Y	60	0	👁
USERID	INTEGER	N		0	👁

Table definition

⋮ ×

EXPENSES

Approximate 65 rows (32.0 KB)
Updated on 2022-11-19 16:39:43

Name	Data type	Nullable	Length	Scale	
USER	INTEGER	N		0	👁
DATE	DATE	N	4	0	👁
AMOUNT	BIGINT	N		0	👁
CATEGORY	VARCHAR	N	32	0	👁
EXPENSEID	INTEGER	N		0	👁

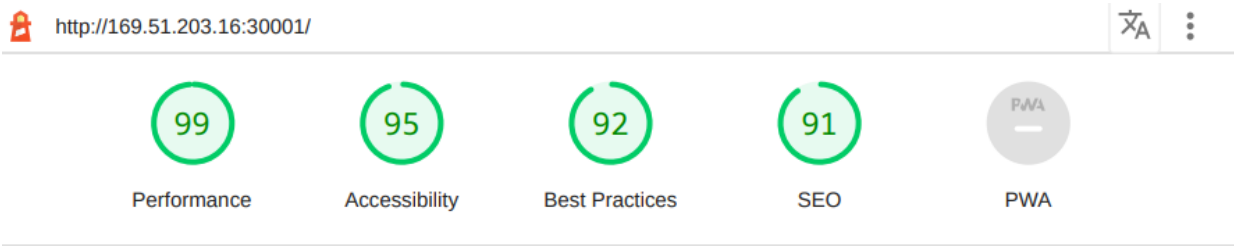
TESTING

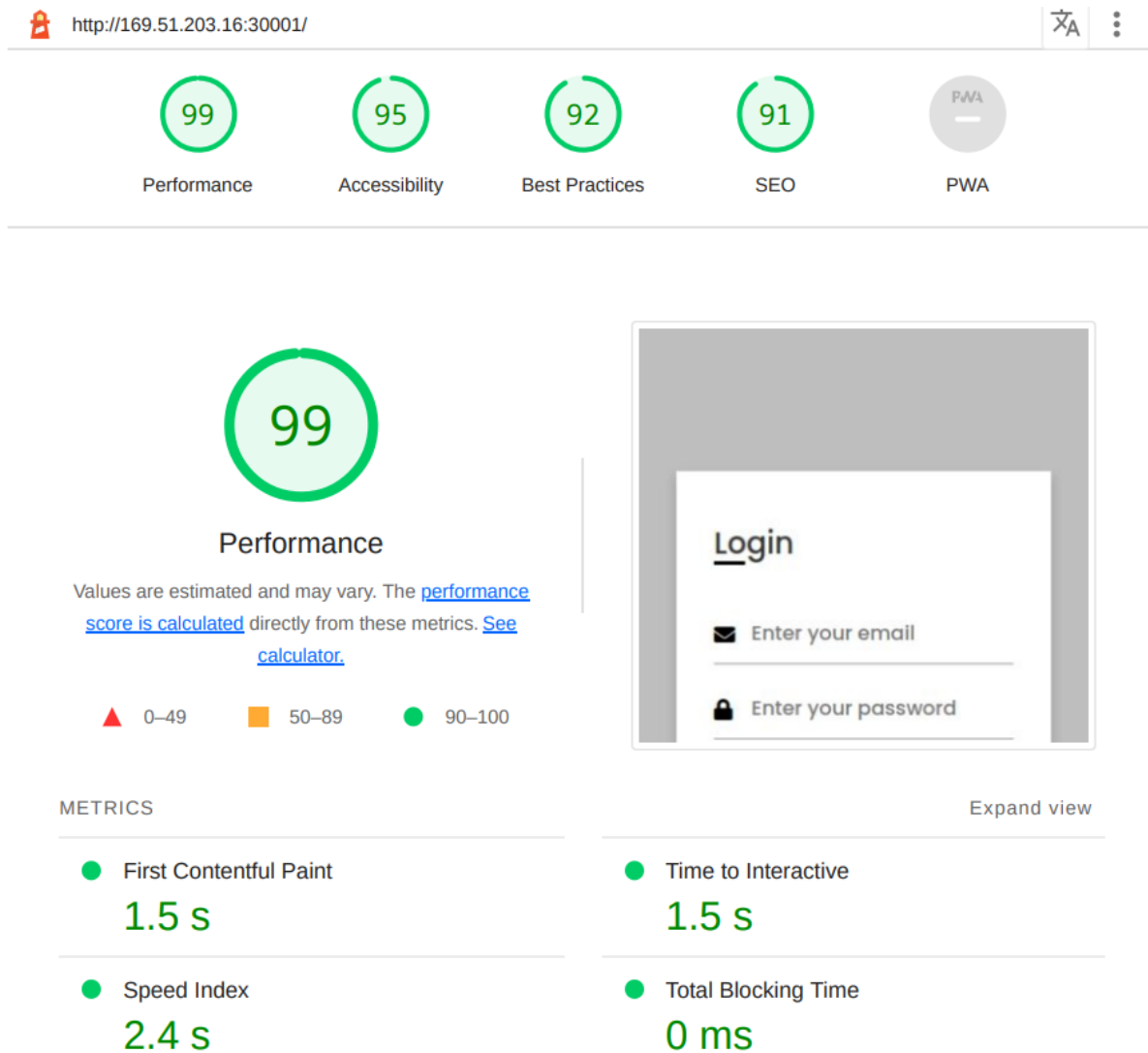
8.1 TEST CASES

Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
Login Page	Submit Signup Form	Enter the URL	Fill up Form and Click Submit	http://169.51.203.16:30001/	Creates user	Working as expected	Pass
Login Page	Submit Login Form	Sign Up Completed	Fill up Form and Click Submit	http://169.51.203.16:30001/	Login to dashboard	Working as expected	Pass
Graph	Display Graph	Add Expenses	Login and click the dashboard link	http://169.51.203.16:30001/	Valid spending graph	Working as expected	Pass
Dashboard	Add Expenses	Login	Fill up expense form and submit	http://169.51.203.16:30001/	Expenses is added	Working as expected	Pass

RESULTS

9.1 PERFORMANCE METRICS







PWA OPTIMIZED

-
- ▲ Does not register a service worker that controls page and `start_url` ✓
 - ▲ Is not configured for a custom splash screen Failures: No manifest was fetched. ✓
 - ▲ Does not set a theme color for the address bar.
Failures: No manifest was fetched, No ``<meta name="theme-color">`` tag found. ✓
 - ▲ Content is not sized correctly for the viewport
The viewport size of 409px does not match the window size of 360px. ✓
 - Has a `<meta name="viewport">` tag with `width` or `initial-scale` ✓
 - ▲ Does not provide a valid `apple-touch-icon` ✓
 - ▲ Manifest doesn't have a maskable icon No manifest was fetched ✓
-

	img	...images/cutlery.jpg (169.51.203.16)	15.7 KiB	14.8 KiB
	img	...images/EMI.jpg (169.51.203.16)	15.0 KiB	14.1 KiB
	img	...images/misc.jpg (169.51.203.16)	12.9 KiB	12.1 KiB
	img	...images/entertainment.jpg (169.51.203.16)	12.3 KiB	11.6 KiB

Use HTTP/2

0.3 s ^

HTTP/2 offers many benefits over HTTP/1.1, including binary headers and multiplexing. [Learn more](#).

URL	Protocol
/dashboard (169.51.203.16)	http/1.1
...css/style.dashboard.css (169.51.203.16)	http/1.1
...images/profile.webp (169.51.203.16)	http/1.1
...images/cutlery.jpg (169.51.203.16)	http/1.1
...images/entertainment.jpg (169.51.203.16)	http/1.1

ADVANTAGES AND DISADVANTAGES

The benefits of this project include the ability for users to take advantage of the application and successfully manage their finances through reminders for various bill payments and the ability to receive alerts each time they go over their budget. Through numerous visualisation charts, the user may also observe how they are using their money. Additionally, the Watson chatbot greatly facilitates user navigation throughout the entire programme.

The absence of recommendation tools that tell users what to do with their money and how to save money to attain specific goals is one of this application's shortcomings. Additionally, neither currency nor language changing features are included

CONCLUSION

Through this project we have learnt how to plan and build a project from scratch and coordinate with our team members. Additionally, we have learnt how a project is structured in a professional manner. Our app is able to help people manage and effectively spend their money.

FUTURE SCOPE

In order for users to benefit and see what our app has to offer that will help them in attaining their goals, we plan on adding recommendation systems on effective management of money to our future scope. Additionally, we hope to provide joint family accounts and introduce additional languages to the app.

APPENDIX

- [1] Bekaroo, G., & Sunhaloo, S. Intelligent Online Budget Tracker.
- [2] Underwood, D. (2011). A Case Study of Tracking Expenses by Commodity at Widget Farmers' Cooperative.
- [3] Chandini, S., Poojitha, T., Ranjith, D., Akram, V. M., Vani, M. S., & Rajyalakshmi, V. (2019). Online Income and Expense Tracker.
- [4] Satpute, M. K., Kale, A., Mandal, A., & Krishnan, R. SURVEY ON CLASSIFICATION ENGINE FOR MONETARY TRANSACTIONS
- [5] Sabab, S. A., Islam, S. S., Rana, M. J., & Hossain, M. (2018, September). eExpense: A smart approach to track everyday expense. In 2018 4th International Conference on Electrical Engineering and Information & Communication

Technology (iCEEiCT) (pp. 136-141). IEEE.