PERSONAL EXPENSE TRACKER APPLICATION

IBM-Project-39400-1660411348

NALAIYATHIRAN PROJECT BASE LEARNING ON PROFESSIONAL READINESS FOR INNOVATION, EMPLOYMENT AND ENTREPRENEURSHIP

A PROJECT REPORT

ANURAAM S - 962319104022

ANANTHAPADMANABHAN G - 962319104019

BALA CHANDRAN S - 962319104030

KANEESHMA LAJ K K- 962319104050

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA COLLEGE OF ENGINEERING AND TECHNOLOGY

ERACHAKULAM, NAGERCOIL

ANNA UNIVERSITY::CHENNAI 600 025

INDEX

PERSONAL EXPENSE TRACKER

1. INTRODUCTION

1.1 Project Overview

Objective:

    With the launch and increase in sales of smart phones over the last few years, people are using mobile applications to get their work done, which makes their lives easier. Mobile applications comprise various different categories such as Entertainment, Sports, Lifestyle, Education, Games, Food and Drink, Health and Fitness, Finance, etc. This Expense Tracker application falls in the Finance Category and serves the important purpose of managing finances which is a very important part of one's life.

Abstract:

    This Project is based on an expense and income tracking system. This project aims to create an easy, faster and smooth tracking system between the expense and the income. This project also offers some opportunities that will help the user to sustain all financial activities like digital automated diary. Most of the people cannot track this expense and income one way they face in many crisis in this daily expense tracker can help the people to tracking income expense day to day and making life tension free. This project will save time day and provide a responsible lifestyle.

1.2 Purpose

    The purpose of the project is to keep accurate record of our 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. They also helps you keep track of balance, outlook, cash-flow, spending, credit, budget, goals, planned payments, debts, shopping lists, warranties, loyalty cards, and currency rates.

2. LITERATURE SURVEY

2.1 Existing problem

The expense-tracker existing system does not provide the user portable device management level, existing systems only used on desktop software so unable to update anywhere expenses done and unable to update the location of the expense details disruptive that the proposed system provides. In existing, we need to maintain the excel sheets, CSV files for the user daily, weekly and monthly expenses. In existing, there is no as such complete solution to keep a track of its daily expenses easily. To do so a person as to keep a log in a diary or in a computer system, also all the calculations need to be done by the user which may sometimes results in mistakes leading to loses. It is not user friendly because data is not maintained perfectly.

2.2 References
- P. Thanapal, Mohammed Yaseen Patel, T.P.Lokesh Raj and J. Satheesh Kumar "Income and Expense Tracker", Indian Journal of Science andTechnology, Vol 8(S2), ISSN: 0974-5645(January 2014).
- Girish Bekaroo and Sameer Sunhaloo,"Intelligent Online Budget Tracker",ComputerScience and IT Education Conference (2014).

2.3 Problem Statement Definition

At the instant, there is no as such complete solution present easily or we should say free of cost which enables a person to keep a track of its daily expenditure easily. To do so a person has to keep a log in a diary or in a computer, also all the calculations needs to be done by the user which may sometimes results in errors leading to losses.
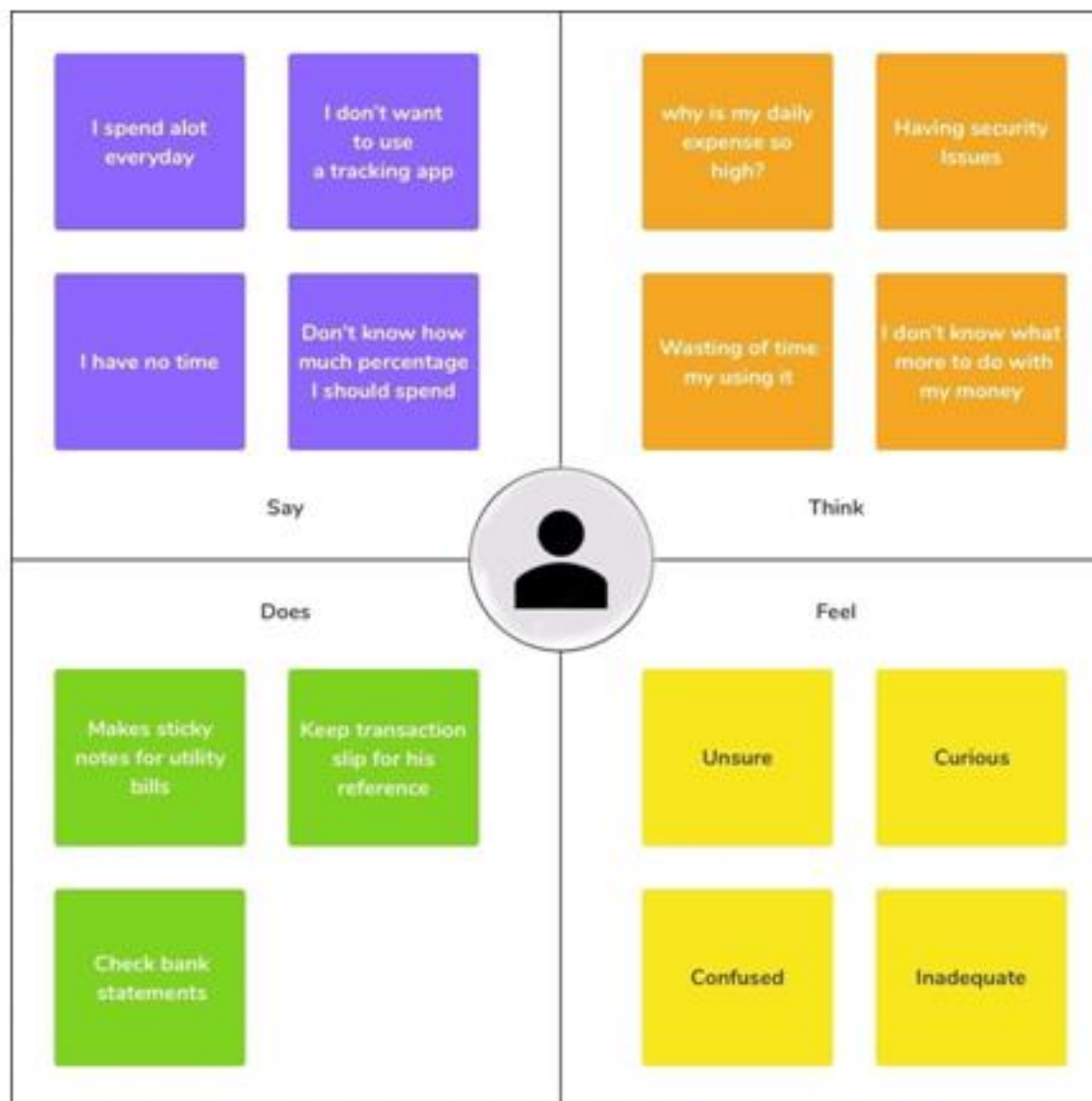
Due to lack of a complete tracking system, there is a constant overload to rely on the daily entry of the expenditure and total estimation till the end of the month. As the name itself suggests, this project is an attempt to manage our daily expenses in a more efficient and manageable way. The system attempts to free the user with as much as possible the burden of manual calculation and to keep the track of the expenditure.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

- An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.
- It is a useful tool to helps teams better understand their users.
- Creating an effective solution requires understanding the true problem and the person who is experiencing it.

  The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

3.1 Ideation & Brainstorming

Brainstorm,Idea, Listing and Grouping :

Idea Prioritization :

3.2   Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	I am Anuram and I am working in a IT company. I will get more salary in my company so that I spend more money. The problem I have is that in most times, I don't know how much money I have with me and I will spend more money in more places. mostly I will spend more amount. so I am facing more problems. So have any reliable sources to clear my problem in some of the my amount to reduce and I must know how much amount I have. There are many websites, but somehow, I don't feel they are authentic and real. It would make my world if those replies are from a real expert and I could clarify all my doubts in a single platform. Of course, I would need instant replies from a real expert who knows about the expense tracker I am asking for.
2.	Idea / Solution description	Creating a Personal Expense Tracker, where the Employees/Workerswill be usefulfor solving their expense problem. they also can raise their queries about the expenses. This problem will be mainly solved by AI. Whatever doubts they can ask can solve with AI itself. If their doubts are not solved by AI. We also assign an agent so they will bewill be assigned to them for replying/clarifying their issues.
3.	Novelty / Uniqueness	The AI plays the crucial part in expense tracker. AI itself solves all the problems for the Employees/Workers. Additionally we are providing some agents also. So they are experts in the solving all the problems and they also chat with Employees/Workers in polite way. The domain and they will communicate well with the users.
4.	Social Impact / User Satisfaction	Employees/workers will most probably satisfied with AI in case if they didn't satisfy with AI or if they didn't understand anything in AI. We have allotted well talented agents. So that they satisfied with the instant and valid replies. Also, it creates a doubtless society, that helps in spending money in limit for the users.
5.	Business Model (Revenue Model)	Employees/Workers can be charged a minimal amount based on the number of queries they ask to AI and Agents. They can raise in a said period of time if the time crosses then minimal amount will be charged.
6.	Scalability of the Solution	This idea is so much use to the Employees/Workers that the latter may refer this expense tracker to their friends and colleagues at work. Naturally, the user base grows so does the number of queries will be answered by AI and Agents. May be in the future, may be a cross-platform mobile application may be developed, making this Personal Expense Tracker much more accessible to the users.

3.4 Problem Solution fit

Define CS, fit into CC	**1. CUSTOMER SEGMENT(S)** **CS**	**6. CUSTOMER CONSTRAINTS** **CC**	**5. AVAILABLE SOLUTIONS** **AS**	Explore AS, differentiate

1. CUSTOMER SEGMENT(S) — CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Our customers are usually above 16 years old. Ranging from college students to working adults to retired professionals. Also, reputed organizations too.

6. CUSTOMER CONSTRAINTS — CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

1. Late replies for their queries
2. Complicated process to take over
3. High chance their queries may not be considered at all
4. Replies irrelevant to their queries
5. Advertisements shown

5. AVAILABLE SOLUTIONS — AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Customers most probably use **helpdesk**.
Pros:
1. Reasonably priced
2. Highly scalable for team of any size
Cons:
They do not understand the severity of all complaints and end up treating them all in the same way

---

2. JOBS-TO-BE-DONE / PROBLEMS — J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

✓ Tracking Expenses Will Build a Budget that works
✓ Monitor your spending to make sure all monthly expenses are covered
✓ Even with a successful budget check-in ensure your spending plan is up to update
✓ Quick Cash Storages
✓ Insufficient Cash Flow

9. PROBLEM ROOT CAUSE — RC

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.

1. Data Entry Errors
2. Unauthorized Bookings
3. Inaccurate Recients
4. No Visibility into spending trends and patterns
5. Lost Recients

7. BEHAVIOUR — BE

What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

1. Asking their friend's opinions
2. Checking solutions in the online forums
3. Using helpdesk
4. Solve the issues themselves based on their own knowledge
5. A monthly budget is a useful tool to track income and expenses.

---

3. TRIGGERS — TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news, etc.

Overtime, they get disappointed with late and irrelevant replies and triggered to act

4. EMOTIONS: BEFORE / AFTER — EM

How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure vs relieved, confident; use it in your communication strategy & design.

✗ Emotional factor–Spend more than they can reasonably afford.
✗ Sadness factor–Sadness trigger a chain of emotions that leads to extravagant tendencies.

10. YOUR SOLUTION — SL

If you are working on an existing business, write down your current solution first. If this is a new business, and also have a much difficulty. If you are working on a new business, you may copy-paste from the right side (SL) first. Once you have gone through the whole canvas and come up with a solution that fits within customer.

• Creating a Personal expense tracker
• Managing their expenses in a month
• Customers can keep track of expense in the month
• Automatic mail gets send to the user by the system if expense gets increased the limit.
• Users will be able to set limit on their expense.

8. CHANNELS of BEHAVIOUR — CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7.

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

ONLINE:
1. https://www.vox.com/
2. https://www.google.com/
3. https://www.quora.com/

OFFLINE:
1. Pencil and Paper
2. Envelope System

4. **REQUIREMENT ANALYSIS**

4.1 Functional requirement

- A functional requirement defines a function of a system or its component, where a function is described as a specification of behaviour between inputs and outputs.

- It specifies "what should the software system do?"

- It is mandatory

- Defined at a component level

- Usually easy to define

- Helps you verify the functionality of the software

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Signup form (customer)
FR-2	Forgot Password	Resetting the password by sending an OTP to user's mail (customer, agent, admin)
FR-3	User Login	Login through Login form (customer, agent, user)
FR-4	Agent creation (admin)	Create an agent profile with username, email and password
FR-5	Dashboard (User)	Show the User to enter the details of their expense details.
FR-6	Dashboard (User)	Show the User the expenditure analysis on the month.
FR-7	Dashboard (User)	Allows the user to edit their expenses.
FR-8	Budget creation (User)	User can able to set their budget limit on monthly basis.
FR-9	Expense Breaker	Shows the user for the expense breakdown based on their daily life activities and payments.

FR-10	Expense by date	Shows the user how expense is added and calculated by the date entered.

4.2 Non-Functional requirements

- A non-functional requirement defines the quality attribute of a software system
- It places constraint on "How should the software system fulfil the functional requirements?"
- It is not mandatory
- Applied to system as a whole
- Usually more difficult to define
- Helps you verify the performance of the software

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Customers can use the application in almost all the webbrowsers. Application is with good looking and detailed UI, whichmakes it more friendly to use.
NFR-2	**Security**	Customers are asked to create an account for themselves using their email which is protected with an 8 character-long password, making it more secure.
NFR-3	**Reliability**	Customers can raise their queries and will be replied with a valid reply, as soon as possible, making the application even more reliable and trust-worthy.
NFR-4	**Performance**	Customers will have a smooth experience while using the application, as it is simple and is well optimised.
NFR-5	**Availability**	Application is available 24/7 as it is hosted on IBM Cloud
NFR-6	**Scalability**	In future, may be cross-platform mobile applications can be developed as the user base grows.

5. PROJECT DESIGN

5.1 Data Flow Diagrams
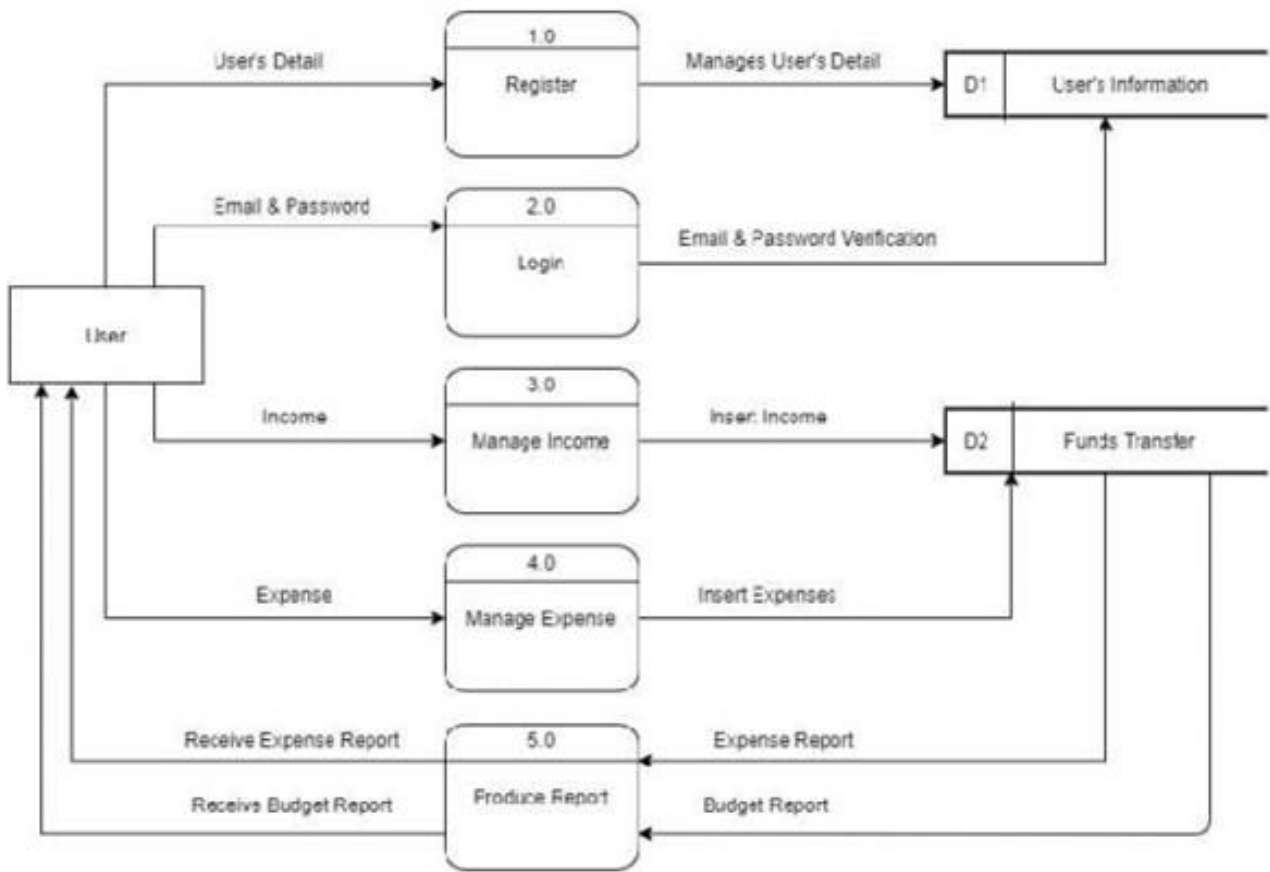
5.2 Solution & Technical Architecture

Technical Architecture:

5.3 User Stories

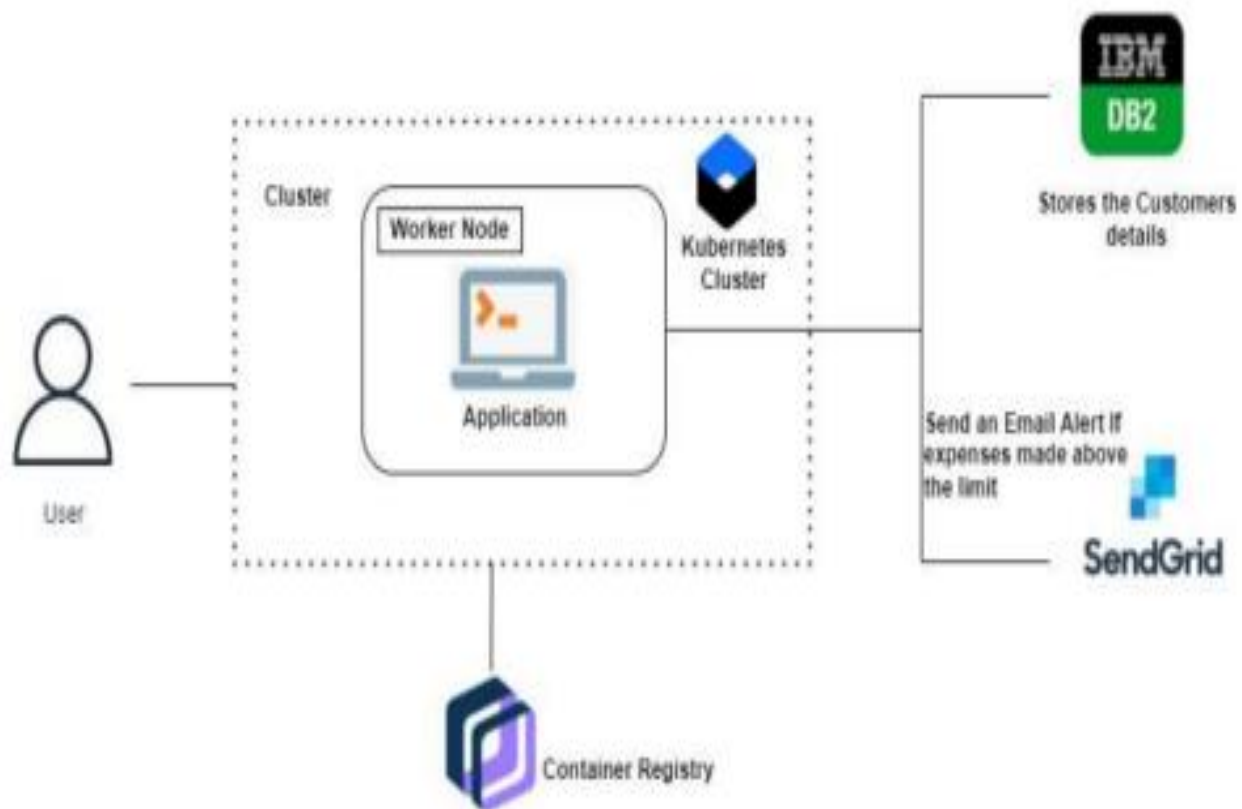| User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance criteria	Priority	Release
User	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account /dashboard	High	Sprint-1
	Login	USN-2	As a user, I can login to the application by entering correct email and password	I can access my account / dashboard	High	Sprint-1
	Dashboard	USN-3	As user, I can see all the expenses raised by me and lot more	I get all the info needed in my dashboard	High	Sprint-1
	Amount creation	USN-4	As a user, I can create a new expense with the detailed description of my query	I can ask my query	High	Sprint-2
	Address Column	USN-5	As a user ,I can have conversations with the assigned agent and get my queries clarified	My queries are clarified	High	Sprint-3
	Forgot password	USN-6	As a user, I can reset my password by this option in case I forgot my old password	I get access to my account again	Medium	Sprint-4
	Expenses details	USN-7	As a user, I can see the current status of my expenses	I get better understanding	Medium	Sprint-4
Agent (Web user)	Login	USN-1	As an agent, I can login to the application by entering correct email and password	I can access my account / dashboard	High	Sprint-3
	Dashboard	USN-2	As an agent, I can see all the expenses assigned to me by the admin	I can see the expenses to which I could answer	High	Sprint-3
	Address Column	USN-3	As an agent, I get to have conversations with the customer and clear his/her queries	I can clarify the issues	High	Sprint-3
	Forgot Password	USN-4	As a agent, I can reset my password by this option in case I forgot my old	I get access to my account again6	Medium	Sprint-4

			password			
Admin (Web user)	Login	USN-1	As an admin, I can login to the application by entering correct email and password	I can access my account / dashboard	High	Sprint-1
	Dashboard	USN-2	As an admin, I can see all the expenses raised in the entire system and lot more	I can assign agents by seeing those tickets	High	Sprint-1
	Agent creation	USN-3	As an admin, I can create an agent for clarifying the user's queries	I can create agents	High	Sprint-2
	Assigning agent	USN-4	As an admin, I can assign an agent for each ticket created by the user	Enables agent to clarify the queries	High	Sprint-2
	Forgot password	USN-4	As an admin, I can reset my password by this option in case I forgot my old password	I get access to my account again	Medium	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story/Task	StoryPoints	Priority	Team members
Sprint1	Registration	USN-1	As a user ,I can register for the application by entering my email ,password, and confirming my password..	2	High	Anuraam

		USN-2	As a user ,I will receive confirmation email once I have registered for the application	1	High	Balachandran
	Login	USN-3	As a user ,I can login to the application by entering email &password	1	High	Kaneeshma
	Dashboard	USN-4	Logging in takes to the dashboard for the logged user.	`2	High	Ananthapadmanabhan

		USN-1	Workspace for personal expense tracking	2	High	Anuraam
	Workspace					
	Charts	USN-2	Creating Pie chart and statistics of customer's data of their expense	1	Medium	Kaneeshma
Sprint2	ConnectingtoIBMDB2	USN-3	Linking database with dashboard	2	High	.Ananthapadmanabhan
		USN-4	Making dashboard interactive with JS	2	High	Balachandran

|		USN-1	Wrapping up the server side works of frontend	1	Medium	Anuraam
	Watson Assistant	USN-2	Creating Chat bot for expense tracking and for clarifying user's query	1	Medium	BalaChandran
Sprint-3	Send Grid	USN-3	Using Send Grid to send mail to the user about their expenses	1	Low	Kaneeshma
		USN-4	Integrating both front end and backend	2	High	Ananthapadmanabhan

		USN-1	Creating image of website using docker	2	High	Kaneeshma
	Docker					
	Cloud Registry	USN-2	Uploading docker image to IBM Cloud registry	2	High	BalaChandran
Sprint-4	Kubernetes	USN-3	Create container using the docker image and	2	High	Anuraam

		hosting the site				
Exposing	USN-4	Exposing IP/Ports for the site	2		High	Ananthapadmanabhan

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date(planned)	Story points Completed(as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	6Days	28Oct2022	11Nov2022	20	29Oct2022
Sprint-2	20	6Days	04Nov2022	04Nov2022	20	05Nov2022
Sprint-3	20	6Days	06Nov2022	11Nov2022	20	12Nov2022
Sprint-4	20	6Days	13Nov2022	11Nov2022	20	19Nov2022

6.3 Reports from JIRA
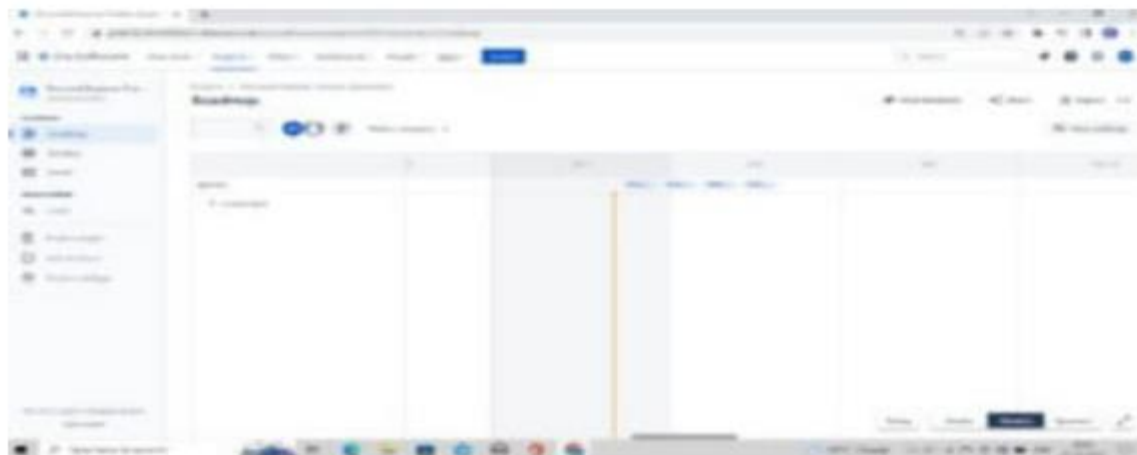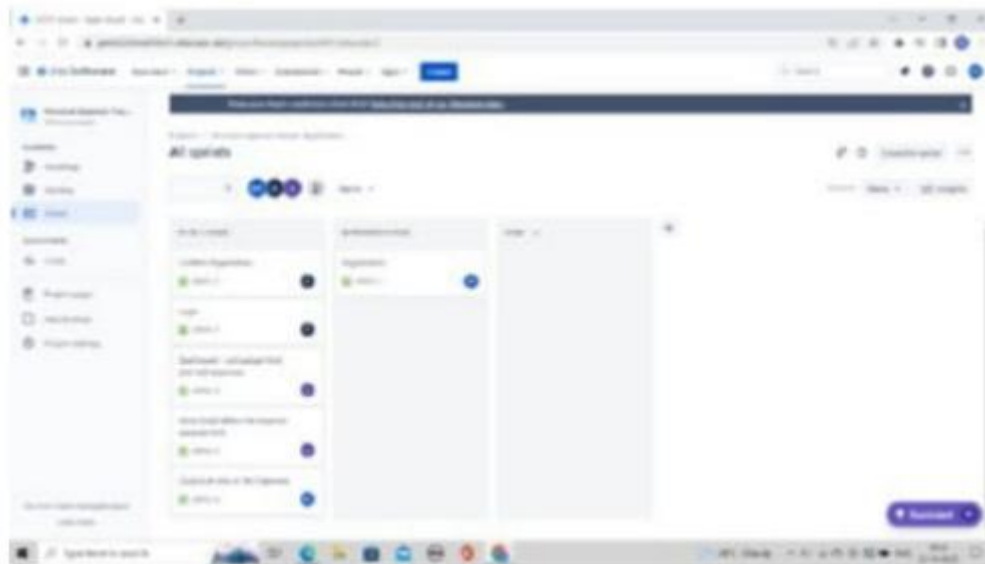
7. CODING & SOLUTIONING

7.1   Feature 1

1. We have added the data visualization methods for expenditure.
2. The pie chart have been used to represent the monthly expenses.
3. 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.
4. For example, it is possible to understand the industry count or percentage of a variable level from the division by areas or sectors.
5. The recommended use for pie charts is two-dimensional, as three-dimensional use can be confusing.

6. The dimensions form sectors of the measurement values.
7. They can have one or two sizes and up to two measures.
8. The first dimension is used to define the angle of each sector that makes up the chart.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.
9. 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:

```
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>
```

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

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

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

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

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

```
80                      <div class="card-body">
81                      <div class="row">
82                          <div class="col-md-
   6">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('tfood').innerHT
   ML
92              let entertainment =
   document.getElementById('tentertainment'
```

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

```
101                        labels: ['Food',
   'Entertainment', 'Business', 'Rent',
   'EMI', 'Other'],
102                        datasets: [{
103                            label:
   'Expenses Chart',
104                            data: [food,
   entertainment, business, rent, emi,
   other],
105
   backgroundColor: [
106                            'rgb(255, 99,
   132)',
107                            'rgb(0, 0,
   0)',
108                            'rgb(255, 205,
   86)',
109                            'rgb(201, 203,
   207)',
110                            'rgb(54,
   162, 235)',
111                            'rgb(215,
   159, 64)'
112                        ],
113                    }]
114                },
115                options: {
116                    responsive: true,
117                    plugins: {
118            legend: {
119              position: 'bottom',
120            },
121            title: {
122              display: true,
123              text: 'EXPENSE BREAKDOWN'
124            }
125         }
126            }
```

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

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

```python
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
```

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

7.Feature 2

1. Email notifications will be sent to the users once they cross the expenditure limit through send grid mail system.
2. Most notifications are transactional,meaning a recipient's action
   or account activity triggers them.
3. Butsome notifications are marketing related, encouraging the recipientto take a
4. specification..
5. Ecommerce product notifications informrecipients about new products or discounts. Plusunlike generalmarketing emails, these are highly personalized and focus on singleproduct.
6. For example, if a customer views an item on your websiteand that item goes on sal, you can send the customer a notificationto let them know this is the best time to by. Users also intoreceiving notifications when an out-of-stock item is back in stock.
7. 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.
8. 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.

```python
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)
```

7.3Database Schema :

Tables :
 1) REGISTER

            id INT NOT NULL GENERATED ALWAYS AS IDENTITY, username VARCHAR(255) NOT NULL, email VARCHAR(255) NOT NULL, password VARCHAR(255) NOT NULL

2) EXPENSES

            id INT NOT NULL GENERATED ALWAYS AS IDENTITY, userid INT NOT NULL, date TIMESTAMP NOT NULL, expense name VARCHAR(255) NOT NULL, amount INT NOT NULL, paymode VARCHAR(255) NOT NULL, category VARCHAR(255) NOT NULL

 3) LIMITS

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

8. TESTING

8.1   Test Cases

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

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

8.2   User Acceptance Testing

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

9. RESULTS

9.1   Performance Metrics

Tracking income and expenses:

Monitoring the income and tracking all expenditures (through bank accounts, mobile wallets, and credit & debit cards).

Transaction Receipts:

Capture and organize your payment receipts to keep track of your expenditure.

Organizing Taxes:

Import your documents to the expense tracking app, and it will streamline your income and expenses under the appropriate tax categories.

Payments & Invoices:

Accept and pay from credit cards, debit cards, net banking, mobile wallets, and bank transfers, and track the status of your invoices and bills in the mobile app itself. Also, the tracking app sends reminders for payments and automatically matches the payments with invoices.

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:-

- The best organizations have a way of tracking and handling these reimbursements. This ideal practice guarantees that the expenses tracked are accurately and in a timely manner. From a company perspective, timely settlements of these expenses when tracked well will certainly boost employees morale

- Financially Aware and Improve Money Management tracking your expenditures ensures you achieve your project financial targets. How is that? By clearly understanding your project spending using project budget limits, you can aptly make the necessary changes to complete your project within time and budget.

- Effective expense tracking and reporting to avoid conflict. As a project manager or business owner, you can set clear policies for the expense types and reimbursement limits to avoid misunderstandings are about costs. Tracking the project expenses by asking team members to provide receipts is helpful to avoid conflict and maintain compliance also. An excellent reporting mechanism is extremely helpful to support the amount to be reimbursed to your team and also invoicing to your customer.

DISADVANTAGES

- Does not provide the user portable device management level.

- Unable to update the location of the expense details as existing system is only used on desktop software.

11. CONCLUSION

The new system has overcome most of the limitations of the existing system and works according to the design specification given. The project what we have developed is work more efficient than the other income and expense tracker. The project successfully avoids the manual calculation for avoiding calculating the income and expense per month. The modules are developed with efficient and also in an attractive manner. The developed systems dispense the problem and meet the needs of by providing reliable and comprehensive information. All the requirements projected by the user have been met by the system. The newly developed system consumes less processing time and all the details are updated and processed immediately. Since the screen provides online help messages and is very user friendly, any user will get familiarized with its usage. Module s are designed to be highly flexible so that any failure requirements can be easily added to the modules without facing many problems. The best organizations have a way of tracking and handling these reimbursements

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.

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

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

13. APPENDIX

Source Code

GitHub & Project Demo Link