

IBM – NALAIYA THIRAN PROJECT

PERSONAL EXPENSE TRACKER APPLICATION

INDUSTRY MENTOR : KUSBOO

FACULTY MENTOR : MANIMEKALAI V

TEAM ID : PNT2022TMID49185

TEAM LEAD : SABARISHKUMAR M

TEAM MEMBER : BARATHSUNDAR S

TEAM MEMBERS : KURINJIKANNAN V

TEAM MEMBER : SEENU G

TABLE OF CONTENT

CHAPTER	CONTENTS	PAGE NO
1	INTRODUCTION 1.1 PROJECT OVERVIEW 1.2 PURPOSE	4
2	LITERATURE SURVEY 2.1 EXISTING PROBLEM 2.2 REFERENCES 2.3 PROBLEM STATEMENT DEFINITION	6
3	IDEATION & PROPOSED SOLUTION 3.1 EMPATHY MAP CANVAS 3.2 IDEATION & BRAINSTROMING 3.3 PROPOSED SOLUTION 3.4 PROBLEM SOLUTION FIT	9
4	REQUIREMENT ANALYSIS 4.1 FUNCTIONAL REQUIREMENT 4.2 NON-FUNCTIONAL REQUIREMENTS	13
5	PROJECT DESIGN 5.1 DATA FLOW DIAGRAMS 5.2 SOLUTION & TECHNICAL ARCHITECTURE 5.3 USER STORIES	15
6	PROJECT PLANNING & SCHEDULING 6.1 SPRINT PLANNING & ESTIMATION 6.2 SPRINT DELIVERY SCHEDULE 6.3 REPORTS FROM JIRA	18

7	CODING & SOLUTIONING 7.1 FEATURE 1 7.2 FEATURE 2 7.3 DATABASE SCHEMA	22
8	TESTING 8.1 TEST CASES 8.2 USER ACCEPTANCE TESTING	39
9	RESULTS 9.1 PERFORMANCE METRICS	41
10	ADVANTAGES & DISADVANTAGES	43
11	CONCLUSION	44
12	FUTURE SCOPE	45
13	APPENDIX SOURCE CODE GITHUB & PROJECT DEMO LINK	46

1.INTRODUCTION

1.1PROJECT 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.

1.2PURPOSE

Expense tracker is software 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. LITERATURE SURVAY

2.1 EXISTING PROBLEM

In this Existing System, when an gets his/her salary he/she can add that into daily expense manager. Then after adding his/her salary details user can expense manager stores all the details. If the user wants all the detail of credit and debit he/she can get it through the Expense Tracker. Notification Manager also reminds about credit and debit details after the salary is updated

2.2 REFERENCES

1. Y. Anita, R. Ranjini, S. Gomathi, “Easy App for Expenses Manager Using Android”, International Journals of Computer Techniques, Volume: 3 Issue: 2, ISSN: 2394-2231 (March-April 2016).
2. N. ZahiraJahan MCA., M. Phil, K. I. Vinodhini, “Personalized Expense Managing Assistant Using Android”, International Journals of Computer Techniques (IJCT), Volume: 3 Issue: 2, ISSN: 2394-2231 (March-April 2016).
3. S. Chandini, T. Poojitha, D. Ranjith, V. J. Mohammed Akram, M. S. Vani, V.Rajyalakshmi, “Online Income and Expense Tracker”, International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 3, e-ISSN:2395-0056, p-ISSN: 2395- 0072 (March 2019).
4. P. Thanapal, Mohammed Yaseen Patel, T. P. Lokesh Raj and J. SatheeshKumar, “Income and Expense Tracker”, Indian Journal of Science andTechnology, Vol 8(S2), ISSN: 0974-5645 (January 2014).
5. Girish Bekaroo and Sameer Sunhaloo, “Intelligent Online BudgetTracker”, Computer Science and IT Education Conference (2014).

2.3 PROBLEM STATEMENT DEFINITION

Modern education does not focus on finance management. This is primarily due to lack of resources and the Indian value system on giving money to children. Failing to teach this valuable knowledge had left many Indians to recklessly spend their income and fall into vicious cycles of EMI and debt. Many of them are just a month's salary away from bankruptcy.

This issue is tackled by providing a web application for where people can plan their monthly expenses into categories, set alerts and get visual insights from their spending patterns.

1. Who does the problem affect ?

Young adults and earning middle class citizens.

2. What is the issue ?

Lack of financial literacy among people

3. When does the issue occur ?

Primarily when the person moves from college to job and starts earning their own money.

4. Where is the issue occurring ?

Especially among young engineers who are newly exposed to consumer centric market and services.

5. Why is it important that we fix the problem ?


The recent Buy Now Pay Later services and Credit apps have Made people spend more than what they earn and repay

3.IDAEATION&PROPOSED SOLUTION

3.1 EMPATHYMAP CANVAS



3.2 IDEATION&BRAINSTORMING



Brainstorm & idea prioritization

Executing a brainstorm isn't unique; holding a productive brainstorm is. Great brainstorms are ones that set the stage for fresh and generative thinking through simple guidelines and an open and collaborative environment. Use this when you're just kicking-off a new project and want to hit the ground running with big ideas that will move your team forward.

- 10 minutes to prepare
- 10-60 minutes to collaborate
- 10-20 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

Team gathering

Define who should participate in the session and send an invite. Share session information in your chat channel.

Set the goal

Think about the problem you're looking to solving in this brainstorming session.

Learn how to use the facilitation tools

Use the Facilitation Experiences to run a happy and productive session.

[Open the article](#)

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a *How Might We* statement. This will be the focus of your brainstorm.

10 minutes

Exercise

How might we... (present problem statement here?)

Rules of brainstorming

To run an efficient and productive session

1. Stop to begin.
2. Define judgments.
3. Go for volume.
4. Encourage wild ideas.
5. Listen to others.
6. If possible, be visual.

2 Brainstorm

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

10 minutes

Suburbs/Kurur	North/South	Karachi/Karachi	South
100% new green city	100% new green city	100% new green city	100% new green city
100% new green city	100% new green city	100% new green city	100% new green city
100% new green city	100% new green city	100% new green city	100% new green city

3 Brainstorm as a group

Have everyone share their ideas into the "group sharing space" within the template and have the team clearly read through them. As a team, sort and group them by the main topics or categories. Discuss and answer any questions. **10-20** min. Encourage "Yes, and..." and build on the ideas of other people during the day.

10 minutes

10

10 minutes

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)	People who are unaware of their daily expenses
2	Idea / Solution description	Proper savings can be done and we can reduce the daily expenses and the application gives proper analyses of our expenditure and income. It reduces man power for accounting and financing the money, human errors can be eradicated.
3	Novelty / Uniqueness	*Expense alerts *Graphical representation *Categories each of your transactions
4	Social Impact / Customer Satisfaction	*User Friendly Interface *Flexibility of adding day to day expenses
5	Business Model (Revenue Model)	Achieve your business goals with a tailored web app that perfectly fits your business
6	Scalability of the Solution	*Access Anywhere Anytime *Accurate Tracking *History of expenses

3.4 PROBLEM SOLUTION FIT

1. CUSTOMER SEGMENT(S) The person who wants to analyze their daily expenses. One who spends more money lavishly	6. CUSTOMER CONSTRAINTS Loss expense report Lack of available devices Manual adding of daily expenses	5. AVAILABLE SOLUTIONS Reduce the man power . Reduce the use of Pen and paper.
2. JOBS-TO-BE-DONE / PROBLEMS To notify them about their expenses. To keep the children expenses on track by the parents	9. PROBLEM ROOT CAUSE To track their daily expense Ability to provide valuable insights	7. BEHAVIOUR They will seek for the customer support. They will undergo mood swings due to the problem.
3. TRIGGERS Easy to handle Fast accessing of the reports To live a systematic lifestyle	10. YOUR SOLUTION Track our expenses Manage budgets Detailed insights of the savings and expenses	8. CHANNELS of BEHAVIOUR Quicker results Tension free Systematic savings and investments
4. EMOTIONS: BEFORE / AFTER Before Insecure Zero confidence Lavish spending of money After Secure Confident Time management and better savings		

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the nonfunctional 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 Form Registration through Gmail Registration through Email Account
FR-2	User Confirmation	Confirmation via Email
FR-3	Calendar	Personal expense tracker application must allow user to add the data to their expenses.
FR-4	Graphical Representation	This application should graphically represent the expense in the form of report.
FR-5	Report Generation	Graphical representation of report must be generated.
FR-6	Category	This application shall allow users to add categories of their expenses

4.2 NON – FUNCTIONAL REQUIREMENTS

These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

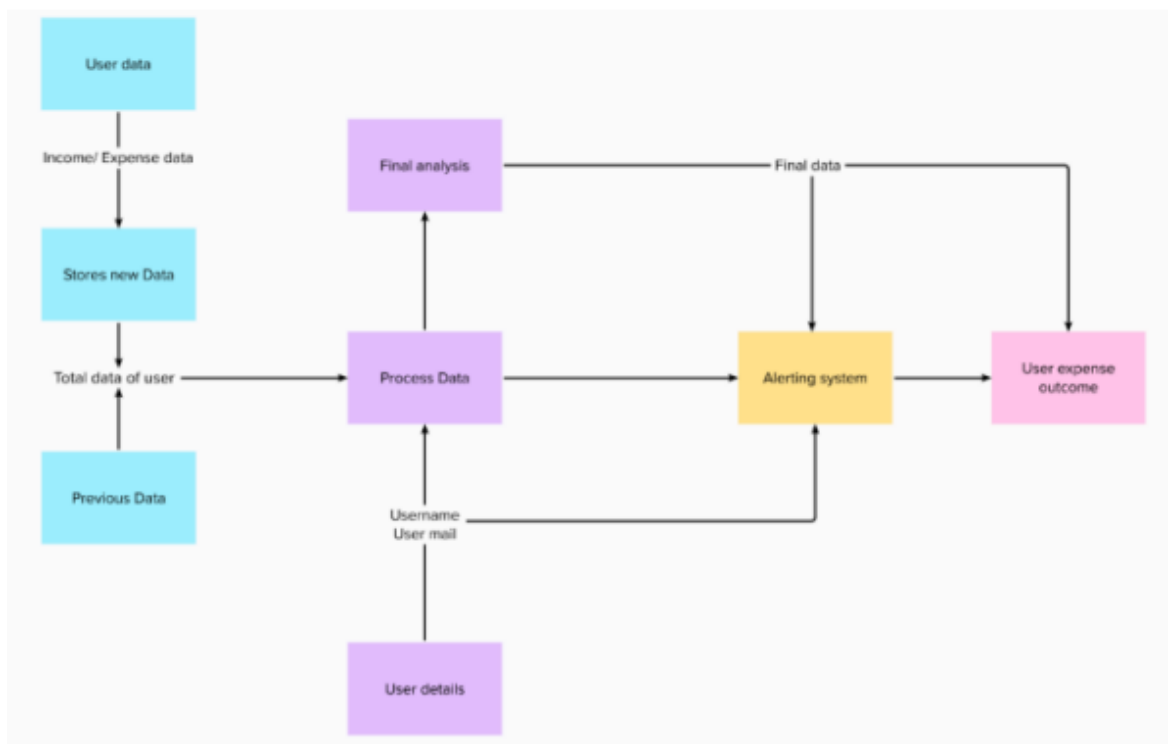
Following are the non-functional requirements of the proposed solution

FR No	Non-Functional Requirement	Description
NFR-1	Usability	Helps to keep an accurate record and track of their income and expenses easily.
NFR-2	Security	We save the password in the encrypted form so it will add more secure to the application user.
NFR-3	Reliability	Each data record is stored on a wellbuilt efficient database schema. There is no risk of data loss.
NFR-4	Performance	Expense kinds include categories and an option. The system's throughput is boosted because to the lightweight database support.
NFR-5	Availability	User can able to access the application with the help of the internet throw the web browser
NFR-6	Scalability	The ability to appropriately handle increasing demands

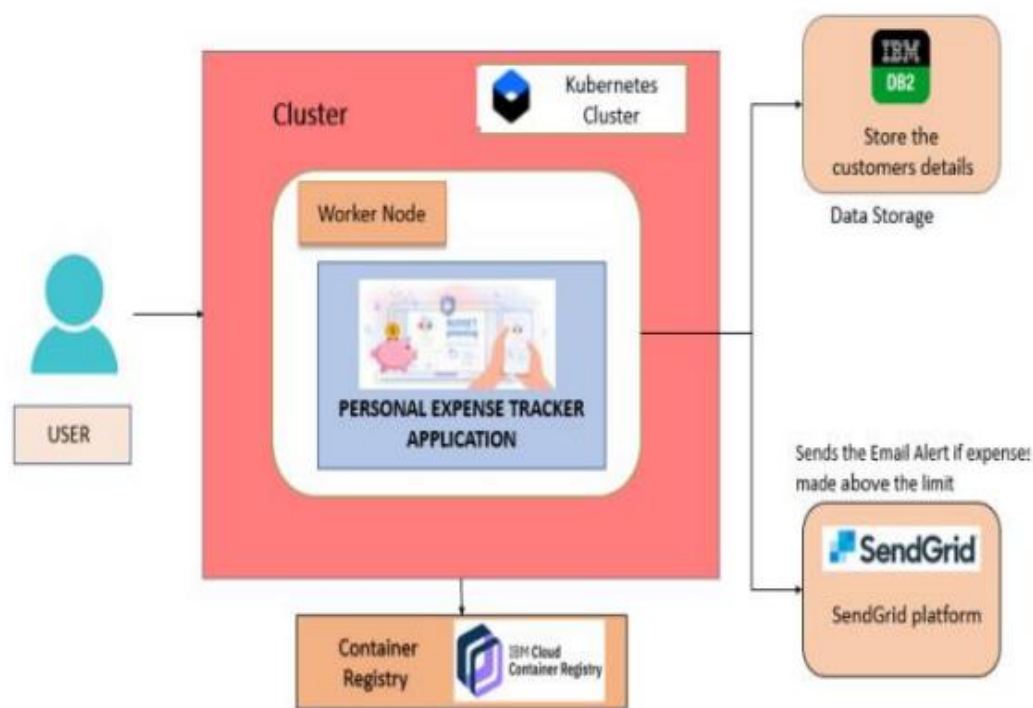
5.PROJECT DESIGN

5.1 DATAFLOW DIAGRAM

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 centers and leaves the system, what changes the information, and where data is stored.



5.2 SOLUTION&TECHNICAL ARCHITECTURE



5.3 USER STORIES

Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Release
Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Sprint-1
Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Sprint-2
Personalized	USN-3	As a user, I can register for the application through Facebook	2	Low	Sprint-3
Storage	USN-4	As a user, I can register for the application through Gmail	2	Medium	Sprint-1
Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Sprint-1
Dashboard	USN-6	As a user, I access my detail, manage my expense, add budget, expense report from the app etc..	2	High	Sprint-4

6. PROJECT PLANNING & SCHEDULE

6.1 SPRINT PLANNING&ESTIMATION

Milestones	Title	Description
Ideation Phase	Literature Survey and Information Gathering	Gathering Information by referring the technical papers, research publications etc
	Prepare Empathy Map	To capture user pain and gains Prepare List of Problem Statement
	Ideation	Prioritise a top 3 ideas based on feasibility and Importance
Project Design Phase-I	Proposed Solution	Solution include novelty,feasibility,business model,social impact and scalability of solution
	Problem Solution Fit	Solution fit document
	Solution Architecture	Solution Architecture
Project Design Phase-II	Customer Journey	To Understand User Interactions and experiences with application
	Functional Requirement	Prepare functional Requirement
	Data flow Diagrams	Data flow diagram
	Technology Architecture	Technology Architecture diagram
Project Planning Phase	Prepare Milestone&Activity list	Prepare the milestone&activity list of the project
Project Development Phase	Project Development Delivery of sprint 1,2,3 &4	Develop and submit the developed code by testing it

6.2 SPRINT DELIVERY SCHEDULE

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	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Sabarish Kumar.M
Sprint-2	Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	KurinjiKannan.V
Sprint-3	Personalized	USN-3	As a user, I can register for the application through Facebook	2		Barathsundar.S
Sprint-1	Storage	USN-4	As a user, I can register for the application through Gmail	2		Seenu G
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	KurinjiKannan.V
Sprint-4	Dashboard	USN-6	As a user, I access my detail, manage my expense, add budget, expense report from the app etc..	2	High	Sabarish Kumar.M

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	18	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	15	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	19	19 Nov 2022

Velocity

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

$$AV = \text{Sprint duration} / \text{Velocity} = 20 / 10 = 2$$

$$AV = 1.8$$

6.3 REPORTS FROM JIRA

	T	NOV	DEC	J
 PET-1 Creating the Login and Register Pages				
 PET-8 Home page of personal expense tracker appl...				
 PET-9 Creating database connectivity				
 PET-10 Building UI Personal Expense Tracker Appli...				
 PET-11 Sendgrid Integration with python Code				
 PET-12 Building Wattson assistant For Chatbot				
 PET-13 containerizing the App				
 PET-14 Upload Images through IBM Cloud				

7.CODING & SOLUTIONING

7.1 Feature1

Login.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

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

```
    <title>Login Page</title>
```

```
    <link rel='stylesheet'  
href='https://cdnjs.cloudflare.com/ajax/libs/twitter-  
bootstrap/4.1.3/css/bootstrap.min.css'>
```

```
    <link rel='stylesheet'  
href='https://fonts.googleapis.com/css?family=Muli'>
```

```
    <link rel="stylesheet" href="./style.css">
```

```
</head>
```

```
<body>
```

```
    <div class="pt-5"></div>
```

```
    <h1 class="text-center">Login</h1>
```

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

```
        <div class="row">
```

```
            <div class="col-md-5 mx-auto">
```

```

        <div id="box" class="card card-body">
            <div class="form-group required">
                <lSabel for="username">Username / Email</lSabel>
                <input type="text" class="form-control text-lowercase"
id="username" required="" name="username" value="">
            </div>
            <div class="form-group required">
                <label class="d-flex flex-row align-items-center"
for="password">Password
                    <a class="ml-auto border-link small-xl" href="/forget-
password">Forget?</a></label>
                    <input type="password" class="form-control" required=""
id="password" name="password" value="">
            </div>
            <div class="form-group mt-4 mb-4">
                <div class="custom-control custom-checkbox">
                    <input type="checkbox" class="custom-control-input"
id="remember-me" name="remember-me" data-parsley-multiple="remember-
me">
                    <label class="custom-control-label" for="remember-
me">Remember me?</label>
                </div>
                <div class="form-group pt-1">
                    <button onclick="main()" class="btn btn-primary btn-
block" type="submit">Log In</button>

```

```
</div>
```

```
<p class="small-xl pt-3 text-center">
```

```
<span class="text-muted">Not a member?</span>
```

```
<a href="register.html">Sign up</a>
```

```
</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script type="text/javascript">
```

```
function main(){
```

```
    window.open("main.html");
```

```
}
```

```
</script>
```

```
<style type="text/css">
```

```
    #box{
```

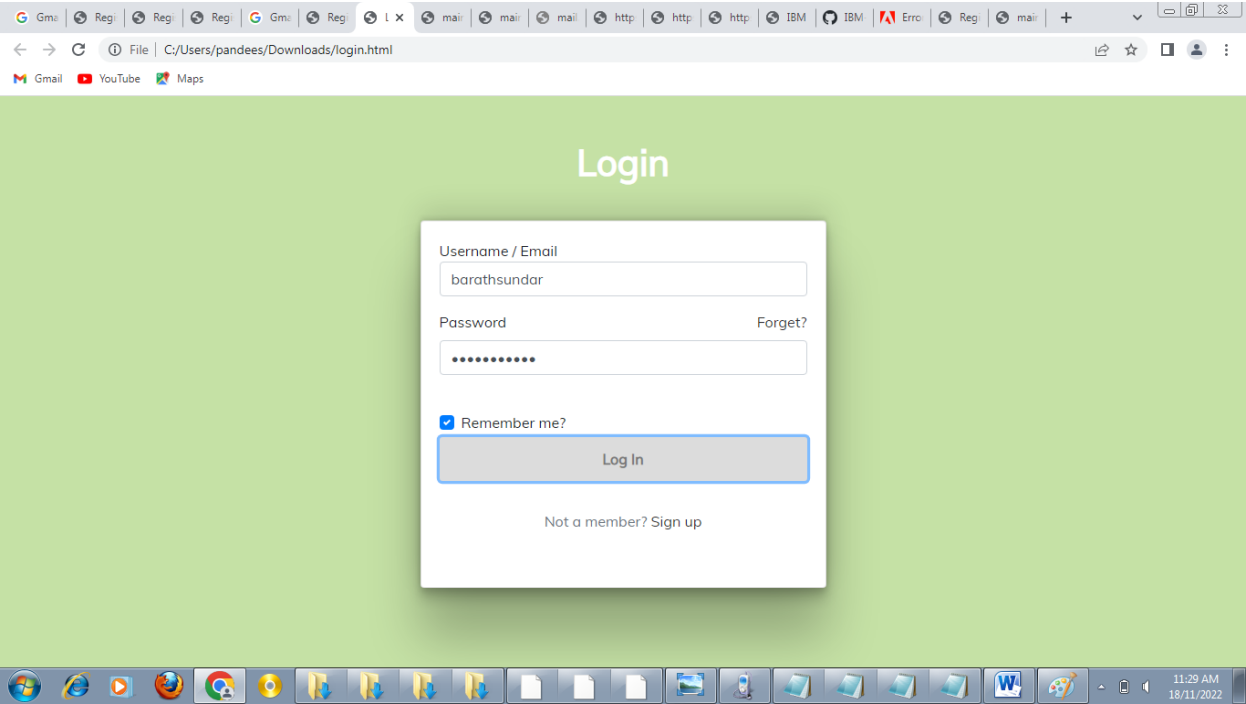
```
        box-shadow: rgba(0, 0, 0, 0.25) 0px 54px 55px, rgba(0, 0, 0, 0.12) 0px -12px 30px, rgba(0, 0, 0, 0.12) 0px 4px 6px, rgba(0, 0, 0, 0.17) 0px 12px 13px, rgba(0, 0, 0, 0.09) 0px -3px 5px;
```

```
    }
```

```
</style>
```

```
</body>
```


</html>



Register.html

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
  <title> Register Page</title>
```

```
  <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.min.css'>
```

```
  <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Muli'>
```

```
  <link rel="stylesheet" href="./style.css">
```

```
</head>
```

```
<body>
```

```
<div class="pt-5">
```

```
  <h1 class="text-center">Register</h1>
```

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

```
    <div class="row">
```

```
      <div class="col-md-5 mx-auto">
```

```
        <div id="box" class="card card-body">
```

```
          <form id="submitForm" action="/login" method="post" data-  
parsley-validate="" data-parsley-errors-messages-disabled="true" novalidate=""  
_lpchecked="1"><input type="hidden" name="_csrf" value="7635eb83-1f95-  
4b32-8788-abec2724a9a4">
```

```

<div class="form-group required">
    <lSabel for="username">Username</lSabel>
    <input type="text" class="form-control text-lowercase"
id="username" required="" name="username" value="">
</div>

<div class="form-group required">
    <lSabel for="username">Email</lSabel>
    <input type="text" class="form-control text-lowercase"
id="username" required="" name="username" value="">
</div>

<div class="form-group required">
    <label class="d-flex flex-row align-items-center"
for="password">Password
    </label>
    <input type="password" class="form-control" required=""
id="password" name="password" value="">
</div>

<div class="form-group pt-1">
    <button onclick="reg()" class="btn btn-primary btn-block"
type="submit">Submit</button>
</div>

</form>

</div>

</div>

```

```
</form>
```

```
<script type="text/javascript">
```

```
function reg(){
```

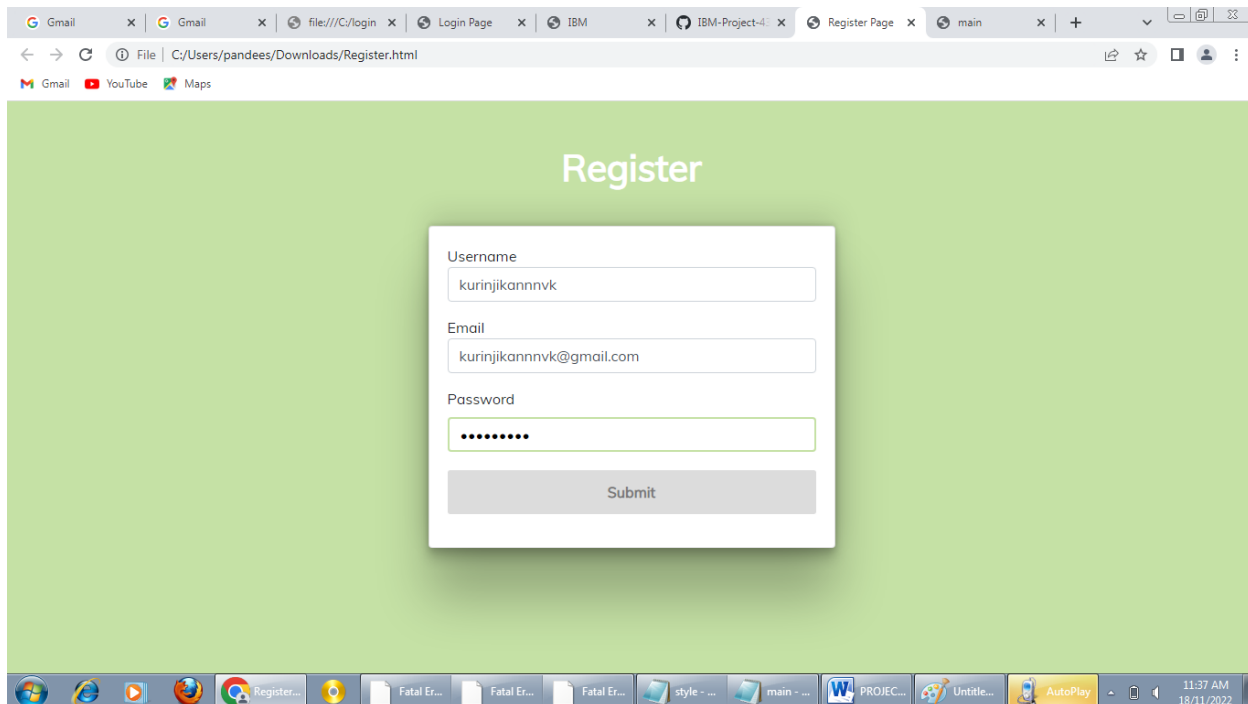
```
    window.open("login.html");
```

```
}
```

```
</script>
```

```
</body>
```

```
</html>
```



Style.html

```
body{  
  background: #C5E1A5 !important;  
  font-family: 'Muli', sans-serif;  
  
}  
  
h1{  
  color: #fff;  
  padding-bottom: 2rem;  
  font-weight: bold;  
}  
  
a{  
  color: #333;  
}  
  
a:hover{  
  color: #C5E1A5;  
  text-decoration: none;  
}  
  
.form-control:focus {
```

```
color: #000;
background-color: #fff;
border: 2px solid #C5E1A5;
outline: 0;
box-shadow: none;

}

.btn{
display: inline-block;
padding: 12px 24px;
background: rgb(220,220,220);
font-weight: bold;
```

7.2 Feature2

Main.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

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

```
  <title>main</title>
```

```
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
```

```
  <script src="https://unpkg.com/gridjs/dist/gridjs.umd.js"></script>
```

```
</head>
```

```
<body>
```

```
  <link rel='stylesheet' href='https://fonts.googleapis.com/css?family=Muli'>
```

```
<lable >MONTHLY SALARY</lable><br>
```

```
<input type="number" id="sal"><br>
```

```
<LABEL id="ren">HOUSE RENT</LABEL><br>
```

```
<INPUT type="number" id="rent"><br>
```

```

<label id="gross">GROSSERY</label><br>
<INPUT type="number" id="gros"><br>
<label id="fee">FEES</label><br>
<INPUT type="number" id="fees"><br>
<label id="ele">ELECTRIC BILL</label><br>
<INPUT type="number" id="eles"><br>
<button class="button-12" role="button" onclick="calc()">SUBMIT</button>
<div class="chartbox">
  <canvas id="myChart"></canvas>
</div>
<script type="text/javascript">

function calc() {
  var sal =document.getElementById('sal').value;
var ren =document.getElementById('rent').value;
var gross =document.getElementById('gros').value;
var fee =document.getElementById('fees').value;
var ele =document.getElementById('eles').value;

  var tot=parseInt(ren)+parseInt(gross)+parseInt(fee)+parseInt(ele);
  var bal=sal-tot;
var grp=[sal,ren,gross,fee,tot,bal,ele];

```



```

const ctx = document.getElementById('myChart');
const myChart = new Chart(ctx, {
  type: 'bar',
  data: {
    labels: ['SALARY', 'HOUSE RENT', 'GROSSERIES', 'FESS', 'TOTAL',
'BALANCE', 'ELECTRIC BILL'],
    datasets: [{
      label: 'EXPENSES TRACKER',
      data: grp,
      backgroundColor: [
        'rgba(255, 99, 132, 0.2)',
        'rgba(54, 162, 235, 0.2)',
        'rgba(255, 206, 86, 0.2)',
        'rgba(75, 192, 192, 0.2)',
        'rgba(153, 102, 255, 0.2)',
        'rgba(255, 159, 64, 0.2)'
      ],
      borderColor: [
        'rgba(255, 99, 132, 1)',
        'rgba(54, 162, 235, 1)',
        'rgba(255, 206, 86, 1)',
        'rgba(75, 192, 192, 1)',
        'rgba(153, 102, 255, 1)',
        'rgba(255, 159, 64, 1)'
      ]
    }]
  }
});

```

```
        'rgba(255, 159, 64, 1)'
    ],
    borderWidth: 1
  }]
},
options: {
  scales: {
    y: {
      beginAtZero: true
    }
  }
}
});

}
```

</script>

<style type="text/css">

```
  INPUT{
    text-align:right;
  }
```

```
body{
```

```
background-color: #C5E1A5;

font-family: 'Muli', sans-serif;

padding: 40px;

}

.chartbox{

width: 1000px;

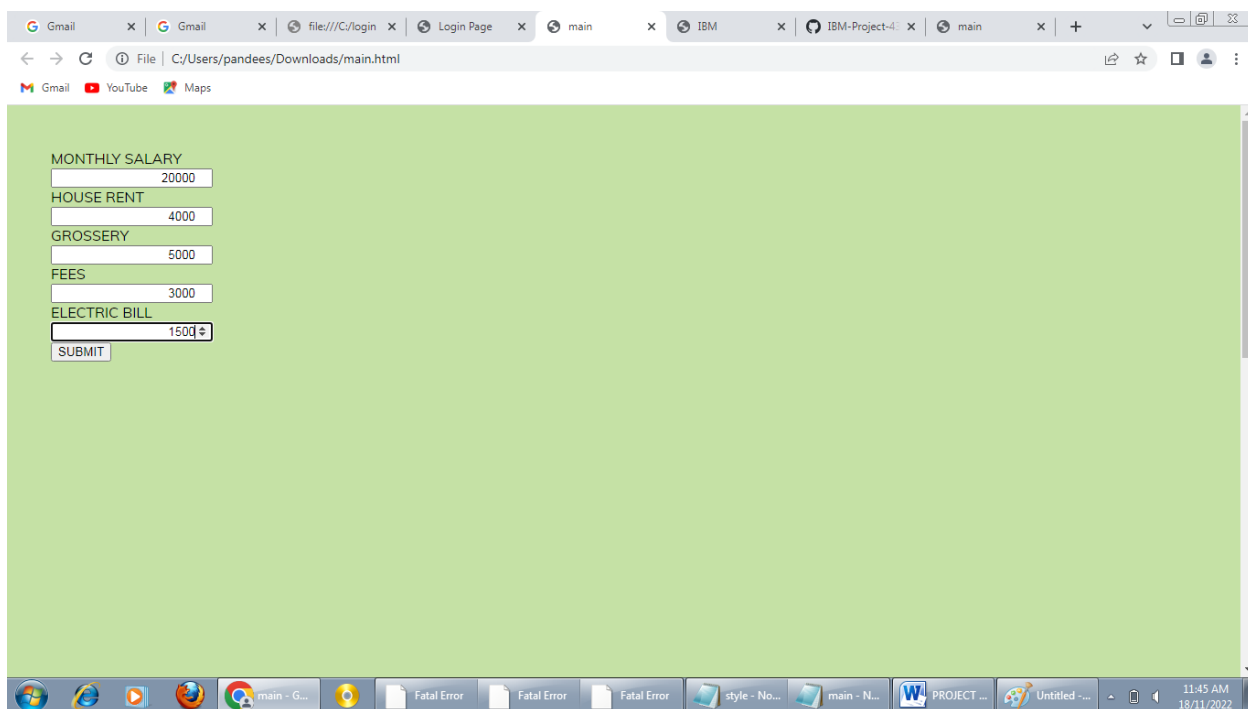
height: 1100px;

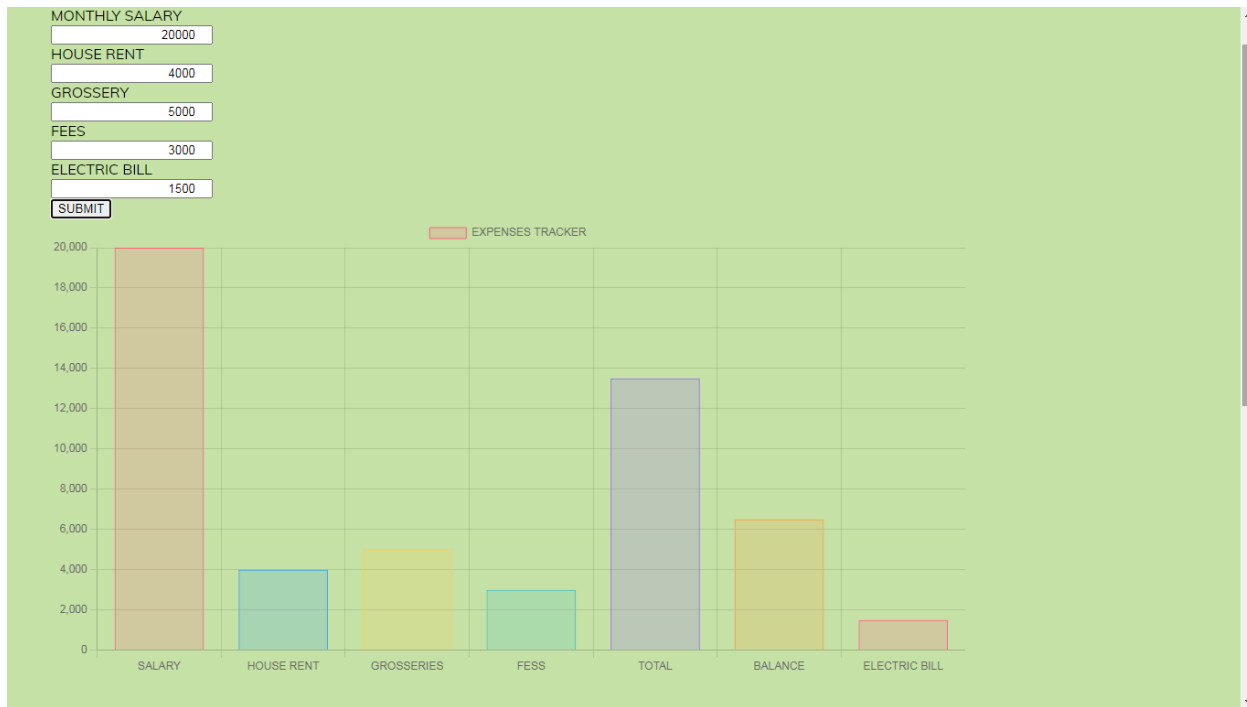
}

</style>
```

```
</body>
```

```
</html>
```





7.3 Database Schema

```
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 t['PASSWORD']})

document =

ibm_db.fetch_assoc(out)

def

insertTableData(conn,rollNo,username,email,pass word):

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=0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=31198;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=bjn03696;PWD=e f96tLjX2VjzaCPX;", "", "")
```

```
print("Db connected")
```

except:

```
print("Error")
```

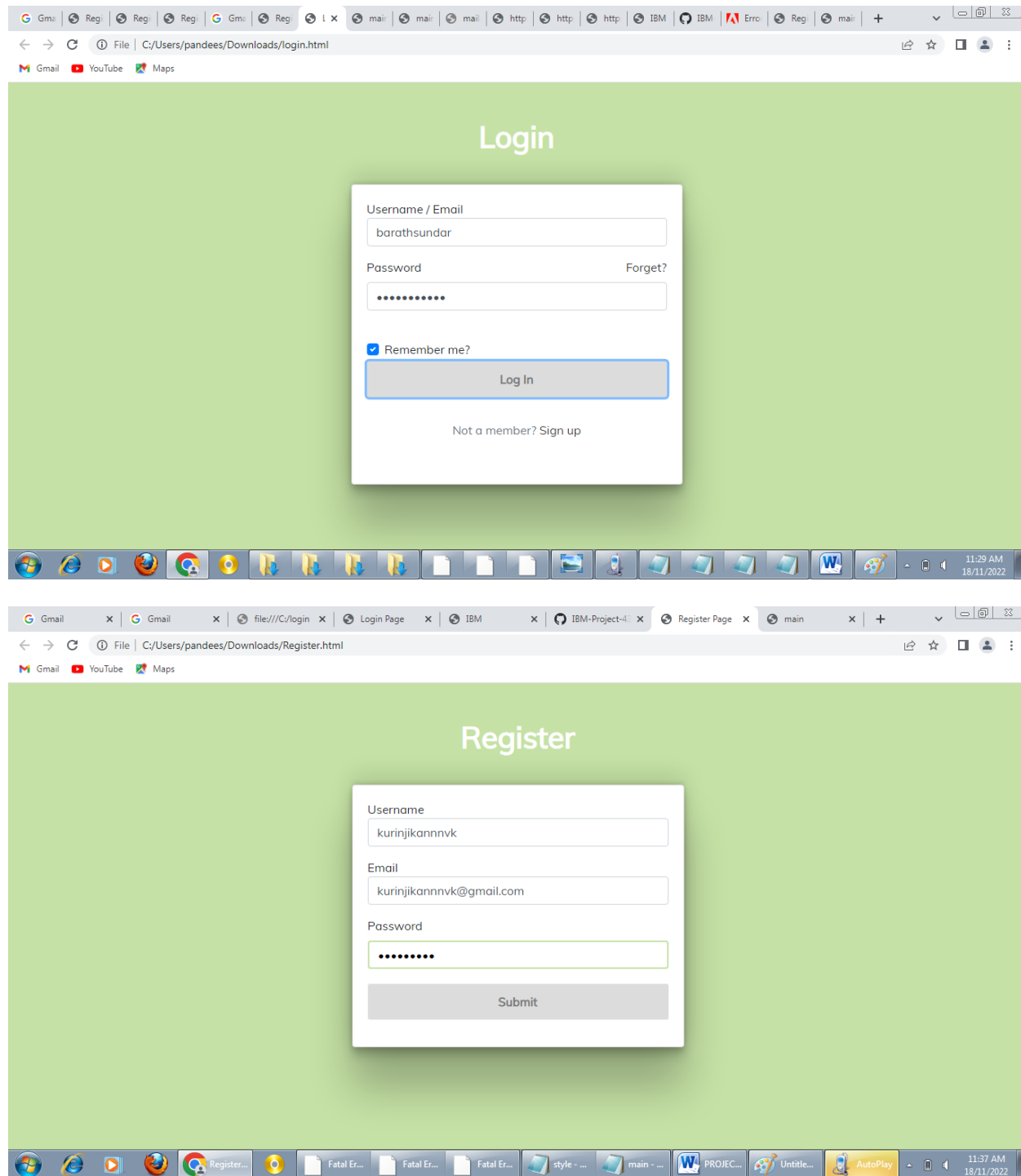
from flask import

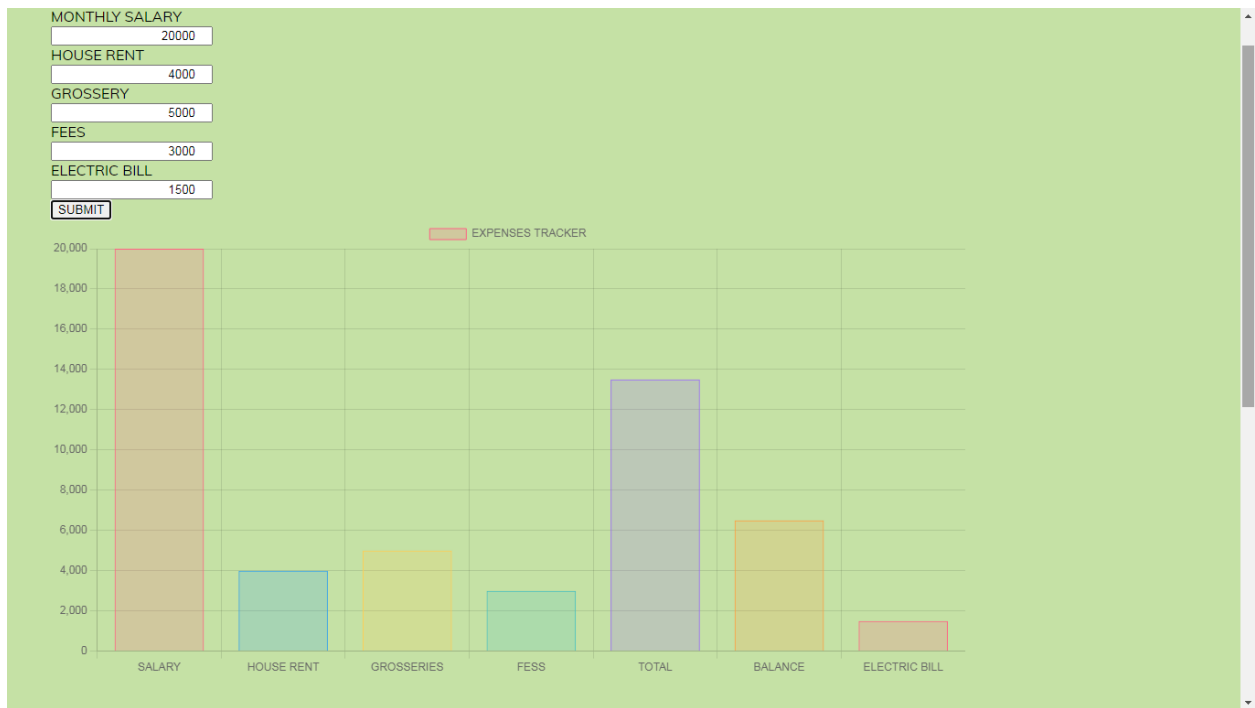
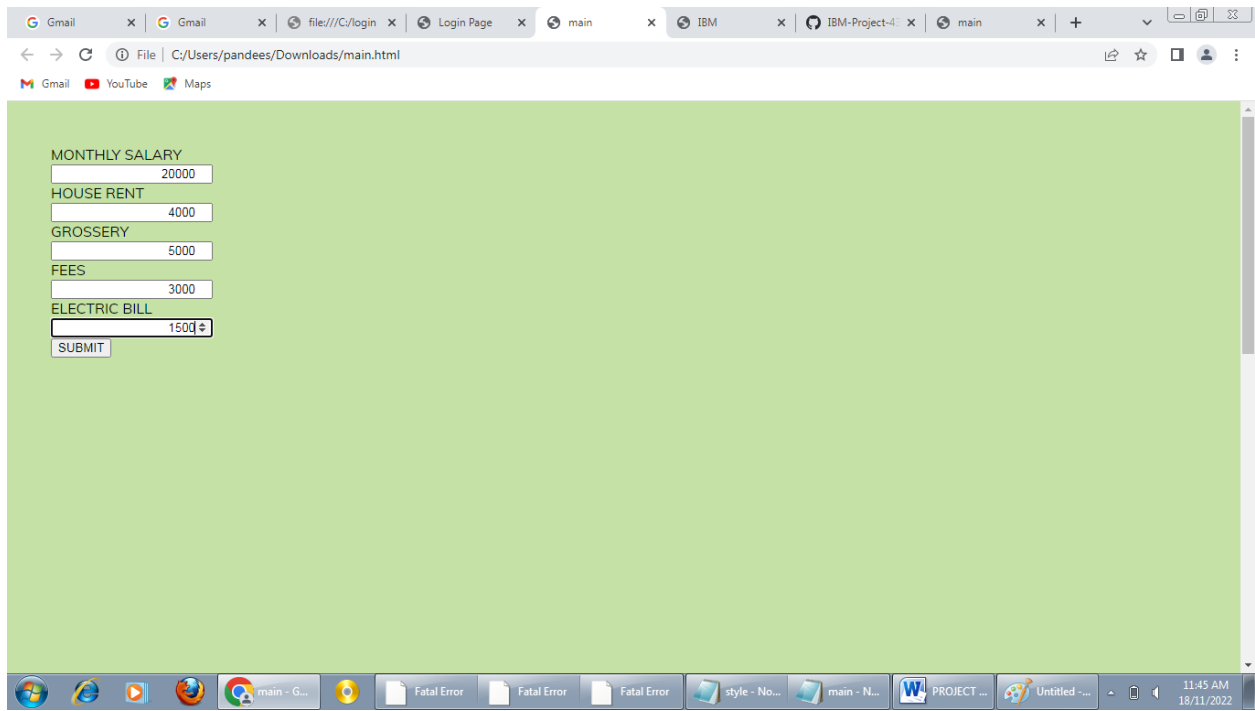
```
Flask,render_template,request,url_for,session
```

```
app=Flask(__name__)  
  
@app.route("/")  
  
@app.route("/login",methods=['POST','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:
```

8.TESTING

8.1 TEST CASES





9.RESULTS

9.1.PERFORMANCE METRICS

The performance of a recommendation algorithm is evaluated by using some specific metrics that indicate the accuracy of the system. The type of metric used depends on the type of filtering technique. Root Mean Square Error (RMSE), Receiver Operating Characteristics (ROC), Area Under Cover (AUC), Precision, Recall and F1 score is generally used to evaluate the performance or accuracy of the recommendation algorithms.

Root-mean square error (RMSE). RMSE is widely used in evaluating and comparing the performance of a recommendation system model compared to other models. A lower RMSE value indicates higher performance by the recommendation model. RMSE, as mentioned by ^[61], can be as represented as follows:

$$RMSE = \sqrt{\frac{1}{N_p} \sum_{u,i} (p_{ui} - r_{ui})^2} \quad (1)$$

where, N_p is the total number of predictions, p_{ui} is the predicted rating that a user u will select an item i and r_{ui} is the real rating.

Precision. Precision can be defined as the fraction of correct recommendations or predictions (known as True Positive) to the total number of recommendations provided, which can be as represented as follows:

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)} \quad (2)$$

It is also defined as the ratio of the number of relevant recommended items to the number of recommended items expressed as percentages.

Recall. Recall can be defined as the fraction of correct recommendations or predictions (known as True Positive) to the total number of correct relevant recommendations provided, which can be as represented as follows:

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \quad (3)$$

It is also defined as the ratio of the number of relevant recommended items to the total number of relevant items expressed as percentages.

F1 Score. F1 score is an indicator of the accuracy of the model and ranges from 0 to 1, where a value close to 1 represents higher recommendation or prediction

accuracy. It represents precision and recall as a single metric and can be as represented as follows:

$$F1\ score = 2 \times \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

Coverage. Coverage is used to measure the percentage of items which are recommended by the algorithm among all of the items.

Accuracy. Accuracy can be defined as the ratio of the number of total correct recommendations to the total recommendations provided

$$Accuracy = \frac{TP + FN}{TP + FN + TN + FP} \quad (5)$$

10. ADVANTAGES & DISADVANTAGES

Advantages

1. This system helps a person to reduce their expenses.
2. Easy to use.
3. You'll have better insight into your spending habits.
4. It provides a better overview and comprehensive analysis.
5. Saving and Investment.

Disadvantages

1. May get inaccurate results if data is not inserted in correct manner.
2. User have to entry every record manually.
3. Person who is handling system must have some technical knowledge

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 userfriendly, 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. 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 your employees' morale

- Quickly Determine How Profitable Business Is
- Effective expense tracking and reporting to avoid conflict

12.FUTURE SCOPE

Expense management software solutions have come a long way from being mere reporting tools. They have evolved because of the advances in the spheres of automation and artificial intelligence. Expense management software that employ the latest technology are an absolute must-have for every organization, irrespective of the size or industry.

Companies have been gradually adopting expense management software in a bid to mitigate the losses associated with expense frauds and wasted time. The speed of adoption and change, however, needs to keep pace with emerging trends in the industry.

13.APPENDIX

SOURCE CODE

Login.html

```
<!DOCTYPE html>

<html>

<head>

    <meta charset="utf-8">

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

    <title>Login Page</title>

    <link rel='stylesheet'
href='https://cdnjs.cloudflare.com/ajax/libs/twitter-
bootstrap/4.1.3/css/bootstrap.min.css'>

    <link rel='stylesheet'
href='https://fonts.googleapis.com/css?family=Muli'>

    <link rel="stylesheet" href="./style.css">

</head>

<body>

    <div class="pt-5"></div>

    <h1 class="text-center">Login</h1>
```

```

<div class="container">

  <div class="row">

    <div class="col-md-5 mx-auto">

      <div id="box" class="card card-body">

        <div class="form-group required">

          <lSabel for="username">Username /
Email</lSabel>

          <input type="text" class="form-control text-
lowercase" id="username" required="" name="username" value="">

        </div>

        <div class="form-group required">

          <label class="d-flex flex-row align-items-center"
for="password">Password

          <a class="ml-auto border-link small-xl"
href="/forget-password">Forget?</a></label>

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

        </div>

        <div class="form-group mt-4 mb-4">

          <div class="custom-control custom-checkbox">

            <input type="checkbox" class="custom-
control-input" id="remember-me" name="remember-me" data-parsley-
multiple="remember-me">

```

```

        <label class="custom-control-label"
for="remember-me">Remember me?</label>

    </div>

    <div class="form-group pt-1">

        <button onclick="main()" class="btn btn-primary
btn-block" type="submit">Log In</button>

    </div>

    <p class="small-xl pt-3 text-center">

        <span class="text-muted">Not a member?</span>

        <a href="register.html">Sign up</a>

    </p>

    </div>

</div>

<script type="text/javascript">

    function main(){

        window.open("main.html");

    }

</script>

<style type="text/css">

```



```
#box{  
    box-shadow: rgba(0, 0, 0, 0.25) 0px 54px 55px, rgba(0,  
0, 0, 0.12) 0px -12px 30px, rgba(0, 0, 0, 0.12) 0px 4px 6px, rgba(0, 0, 0,  
0.17) 0px 12px 13px, rgba(0, 0, 0, 0.09) 0px -3px 5px;  
}
```

```
</style>
```

```
</body>
```

```
</html>
```

Register.html

```
<!DOCTYPE html>
```

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

```
<head>
```

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

```
<title> Register Page</title>
```

```
<link rel='stylesheet'  
href='https://cdnjs.cloudflare.com/ajax/libs/twitter-  
bootstrap/4.1.3/css/bootstrap.min.css'>
```

```
<link rel='stylesheet'  
href='https://fonts.googleapis.com/css?family=Muli'>
```

```
<link rel="stylesheet" href="./style.css">
```

```

</head>

<body>

<div class="pt-5">

  <h1 class="text-center">Register</h1>

  <div class="container">

    <div class="row">

      <div class="col-md-5 mx-auto">

        <div id="box" class="card card-body">

          <form id="submitForm" action="/login" method="post"
data-parsley-validate="" data-parsley-errors-messages-disabled="true"
novalidate="" _lpchecked="1"><input type="hidden" name="_csrf"
value="7635eb83-1f95-4b32-8788-abec2724a9a4">

            <div class="form-group required">

              <lSabel for="username">Username</lSabel>

              <input type="text" class="form-control text-
lowercase" id="username" required="" name="username" value="">

            </div>

            <div class="form-group required">

              <lSabel for="username">Email</lSabel>

              <input type="text" class="form-control text-
lowercase" id="username" required="" name="username" value="">

            </div>

```

```

        <div class="form-group required">
            <label class="d-flex flex-row align-items-center"
for="password">Password
                </label>
                <input type="password" class="form-control"
required="" id="password" name="password" value="">
            </div>
        <div class="form-group pt-1">
            <button onclick="reg()" class="btn btn-primary
btn-block" type="submit">Submit</button>
        </div>
    </form>
</div>
</div>
</form>
<script type="text/javascript">
    function reg(){
        window.open("login.html");
    }
</script>
</body>
</html>

```

Style.html

```
body{  
    background: #C5E1A5 !important;  
    font-family: 'Muli', sans-serif;  
  
}  
  
h1{  
    color: #fff;  
    padding-bottom: 2rem;  
    font-weight: bold;  
}  
  
a{  
    color: #333;  
}  
  
a:hover{  
    color: #C5E1A5;  
    text-decoration: none;  
}  
  
.form-control:focus {  
  
    color: #000;
```

```
background-color: #fff;
border:2px solid #C5E1A5;
outline: 0;
box-shadow: none;

}

.btn{
display: inline-block;
padding: 12px 24px;
background: rgb(220,220,220);
font-weight: bold;
```

Main.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <title>main</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
```

```

<script src="https://unpkg.com/gridjs/dist/gridjs.umd.js"></script>

</head>

<body>

  <link rel='stylesheet'
href='https://fonts.googleapis.com/css?family=Muli'>

  <label >MONTHLY SALARY</lable><br>

  <input type="number" id="sal"><br>

  <LABEL id="ren">HOUSE RENT</LABEL><br>

  <INPUT type="number" id="rent"><br>

  <lable id="gross">GROSSERY</lable><br>

  <INPUT type="number" id="gros"><br>

  <lable id="fee">FEES</lable><br>

  <INPUT type="number" id="fees"><br>

  <label id="ele">ELECTRIC BILL</label><br>

  <INPUT type="number" id="eles"><br>

  <button class="button-12" role="button"
onclick="calc()">SUBMIT</button>

  <div class="chartbox">

    <canvas id="myChart"></canvas>

  </div>

```

```
<script type="text/javascript">
```

```
function calc() {
```

```
    var sal=document.getElementById('sal').value;
```

```
    var ren=document.getElementById('rent').value;
```

```
    var gross=document.getElementById('gros').value;
```

```
    var fee=document.getElementById('fees').value;
```

```
    var ele=document.getElementById('eles').value;
```

```
    var tot=parseInt(ren)+parseInt(gross)+parseInt(fee)+parseInt(ele);
```

```
    var bal=sal-tot;
```

```
    var grp=[sal,ren,gross,fee,tot,bal,ele];
```

```
    const ctx = document.getElementById('myChart');
```

```
    const myChart = new Chart(ctx, {
```

```
        type: 'bar',
```

```
        data: {
```

```
            labels: ['SALARY', 'HOUSE RENT', 'GROSSERIES', 'FESS',  
'TOTAL', 'BALANCE', 'ELECTRIC BILL'],
```

```
            datasets: [{
```

```
                label: 'EXPENSES TRACKER',
```

```
                data: grp,
```

```
    backgroundColor: [  
        'rgba(255, 99, 132, 0.2)',  
        'rgba(54, 162, 235, 0.2)',  
        'rgba(255, 206, 86, 0.2)',  
        'rgba(75, 192, 192, 0.2)',  
        'rgba(153, 102, 255, 0.2)',  
        'rgba(255, 159, 64, 0.2)'  
    ],  
    borderColor: [  
        'rgba(255, 99, 132, 1)',  
        'rgba(54, 162, 235, 1)',  
        'rgba(255, 206, 86, 1)',  
        'rgba(75, 192, 192, 1)',  
        'rgba(153, 102, 255, 1)',  
        'rgba(255, 159, 64, 1)'  
    ],  
    borderWidth: 1  
  }  
},  
options: {  
  scales: {
```



```
        y: {
            beginAtZero: true
        }
    }
});
```

```
}
```

```
</script>
```

```
<style type="text/css">
```

```
    INPUT{
        text-align:right;
    }
```

```
body{
    background-color: #C5E1A5;
    font-family: 'Muli', sans-serif;
    padding: 40px;
}
```

```
.chartbox{
    width: 1000px;
```

```
height: 1100px;  
}  
</style>  
  
</body>  
</html>
```

GITHUB & PROJECT DEMO LINK

GITHUB LINK: <https://github.com/IBM-EPBL/IBM-Project-43002-1660711876>

DEMO LINK: <https://youtu.be/yQDAeeyeJSo>