

SMART FASHION RECOMMENDER

APPLICATION

SUBMITTED BY

GOLOTI PAVANI ACHARY	113319205013
DHARANI KUMAR V	113319205008
SUJITH RAJU S	113319205045
PUGAL OVIYA	113319205037

BACHELOR OF INFORMATION TECHNOLOGY

VELAMMAL INSTITUTE OF TECHNOLOGY

INDEX

1. INTRODUCTION

1. Project Overview
2. Purpose

2. LITERATURE SURVEY

1. Existing problem
2. References
3. Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

1. Empathy Map Canvas
2. Ideation & Brainstorming
3. Proposed Solution
4. Problem Solution fit

4. REQUIREMENT ANALYSIS

1. Functional requirement
2. Non-Functional requirements

5. PROJECT DESIGN

1. Data Flow Diagrams
2. Solution & Technical Architecture
3. User Stories

6. PROJECT PLANNING & SCHEDULING

1. Sprint Planning & Estimation
2. Sprint Delivery Schedule
3. Reports from JIRA

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

1. Feature 1
2. Feature 2
3. Database Schema (if Applicable)

8. TESTING

1. Test Cases
2. User Acceptance Testing

9. RESULTS

1. Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code

GitHub & Project Demo Link

1. INTRODUCTION

1.1 Project Overview

This smart fashion recommendation application aims to improvise user usage and produce suggestions for users to take up the clothes they require according to their needs and current trends.

Various attachments are made so that the user gets the best user experience. A recommendation system is a system that is programmed to predict future preferable items from a large set of collections. A recommendation system works either by using user preferences or by using the items most preferred by all users. The main challenge in building a fashion recommendation system is that it is a very dynamic industry. It changes very often when it comes to seasons, festivals, pandemic conditions like coronavirus, and many more.

Our application contains various software to come up with this most efficient application.

We have come up with a new innovative solution through which you can directly do your online shopping based on your choice without any search. It can be done by using the chatbot.

- Using a chatbot we can manage users' choices and orders.
- The chatbot can give recommendations to users based on their interests.
- It can promote the best deals and offers on that day.
- It will store the customer's details and orders in the database.
- The chatbot will send a notification to customers if the order is confirmed.
- Chatbots can also help in collecting customer feedback.

1.2 Purpose

smart fashion recommendation application aims to improvise user usage and

produce suggestions for users to take up the clothes they require according to their needs and current trends.

2. LITERATURE SURVEY

2.1 Existing Problem

Nowadays recommendation system plays a major role in sales and marketing by analyzing the customer's current site usage and previous browsing history, a recommendation engine can deliver relevant product recommendations as he or she shops based on said profile.

The data is collected in real time so the software can react as shopping habits change on the fly.

Shoppers become more engaged when personalized product recommendations are made to them across the customer journey.

Through individualized product recs, customers are able to dive more deeply into your product line without having to dive into (and very likely get lost in) an e-commerce rabbit hole.

Converting shoppers into customers takes a special touch.

Personalized interactions from a recommendation engine show your customer that he or she is valued as an individual, in turn, engendering long-term loyalty.

Comparing normal fashion application with application using recommendation system, In smart fashion recommendation system uses perfect chatbot to guide the user whereas normal application like Amazon,flipkart does not have a proper chatbot.

A recommendation engine can add your marketing and inventory control directives to a customer's profile to feature products that are

on clearance or overstocked so as to avoid unnecessary shopping friction and tone deafness.

2.2 References

https://www.researchgate.net/publication/334078160_An_Intelligent_Personalized_Fashion_Recommendation_System

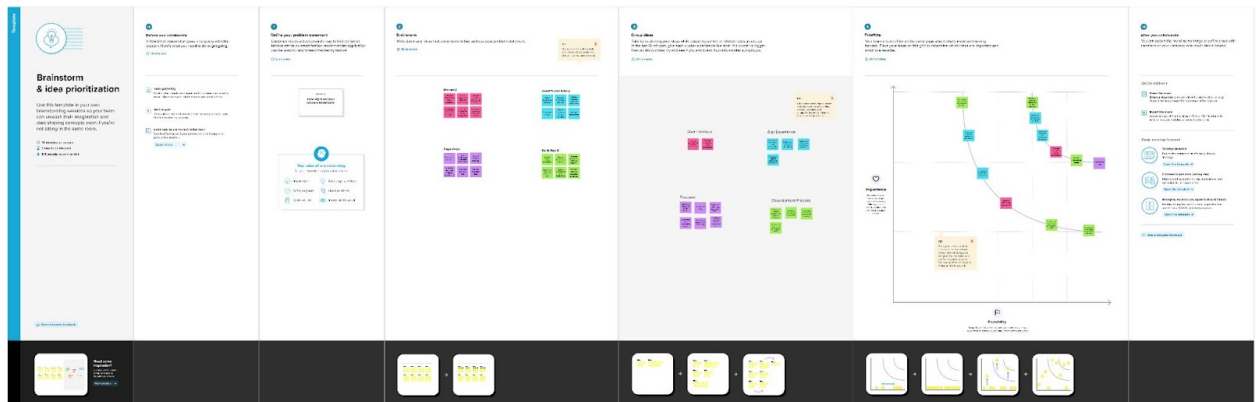
2.3 Problem Statement Definition

The customer needs the best possible way to find the latest fashion trends so smart fashion recommender application can be used to recommend trendy fashion.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Can

3.2 Ideation & Brainstorming



3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The customer needs the best possible way to find the latest fashion trends so a smart fashion recommender application can be used to recommend trendy fashion
3.	Novelty / Uniqueness	Using cloud computing technologies we have worked on giving the most interactive chatbots leading to more user-friendly and best UI/UX designing application.
4.	Social Impact / Customer Satisfaction	This chatbot helps to reduce the time consumption taken to search various products instead it will analyze the user's needs and find the best pick for the user
5.	Business Model (Revenue Model)	It takes minimum cost for the development and pay-as-you-go for the cloud services leading to much scalability and flexibility as more users start using the application the more revenue it will make booming the business and revenue model.

6.	Scalability of the Solution	This solution uses cloud computing technology which is the most scalable and flexible pay-as-you-go solution that can be used whenever the scalability of users increases or decreases
----	-----------------------------	--

3.4 Problem Solution fit

Problem-Solution fit canvas 2.0

To create a Smart Fashion recommender application

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS 1.FASHION DESIGNER 2.TEENAGER 3.SENIOR-CITIZEN 4.FIRST-TIME MOTHER	6. CUSTOMER CONSTRAINTS CC 1.Spending time 2.Budget 3.No cash 4.climatic changes 5.product reviews by family and friends	5. AVAILABLE SOLUTIONS AS 1.Various cataloges are present for design suggestios 2.Experienced fashion reccomenders are present 3.Boutique suggestions are present but are costly	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P 1.Facing problems to get suggestions for the customers and understand their point of view 2.Unable to choose the best dresses for the occasion as people are not good at fashionable selection 3.Donot know how to choose the criteria to buy products 4.Doesnot have offers	9. PROBLEM ROOT CAUSE RC 1.Due to a raise in online shopping trends during the time of the pandemic people lost their communication skills and the catch for current fashion trends 2.Problem of gaining customers	7. BEHAVIOUR BE 1.Customer approaches and gives the required input details 2.Suggests various requirements according to the suggestion 3.Add user-friendly interface where the customer can interact and find the various fashion suggestions	
Identify strong TR & EM	3. TRIGGERS TR The customer is unable to reach the needs of his customer due to which his business is not getting improved and he wants a solution for this	10. YOUR SOLUTION SL To overcome the stated problems a smart fashion recommendation system can be proposed so that all the needs of the customer are satisfied along with providing them the platform to communicate and find the best product according to their needs.	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE 1.Checks for various updates online 2.Tries to search various fashion trends 3.Checks for various offers present online 4.Searches for various comments	Extract online & offline CH of BE
	4. EMOTIONS: BEFORE / AFTER EM The fashion designer feels less interactive with the customer as he is not updated upto the market he feels insecure and helpless		8.2 OFFLINE 1.Checks for discounts in various shops 2.Relative suggestions in various shops at cheap prices 3.Latest trends and various boutiques for dresses will be checked	

Problem-Solution fit canvas is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 license Created by Dania Neprakhina / Amaltama.com

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

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
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Data collection and storage	The data collected from the registration phase is stored and recommendation is provided according to their interest.
FR-4	Image recommendation	All the possible recommendations are suggested and are further used for making suggestions

4.2 Non-Functional requirements

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

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The dashboard of the application provides all the required suggestions and all the images of the clothing providing the user various options
NFR-2	Security	The security of the application is designed in such a way that the user can store their sensitive information and use that for further transactions

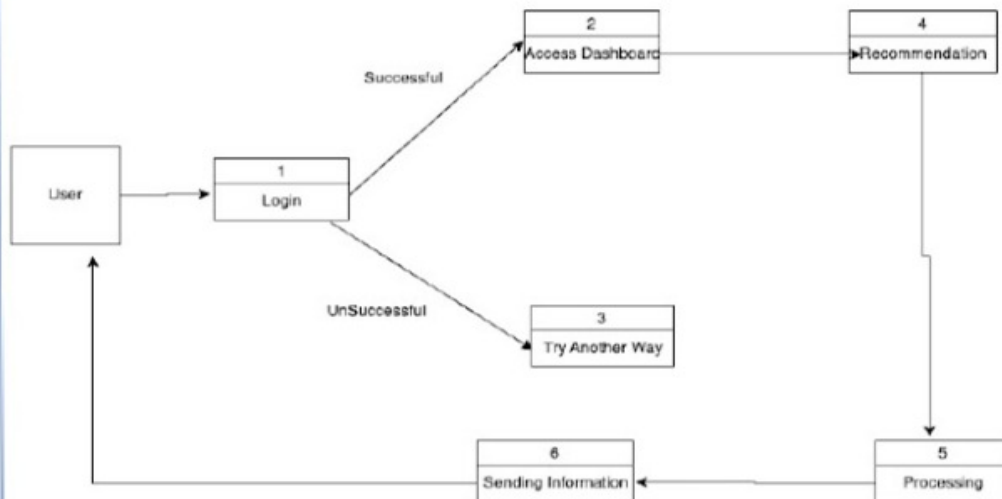
NFR-3	Reliability	This application is highly reliable as it provides various suggestions according to the needs of the user and catches up with the current trendy trends
NFR-4	Performance	The loading time of application is very reliable allowing it to be highly usable and user-friendly
NFR-5	Availability	The availability of products and the suggestion is clearly provided allowing it to be the best application for suggestions the chatbot also allows it

		to be the best recommender for the usage of the user
NFR-6	Scalability	The cloud storage is highly scalable allowing it to expand the application according to the needs of the users registering allowing it to be one of the flexible and scalable unit

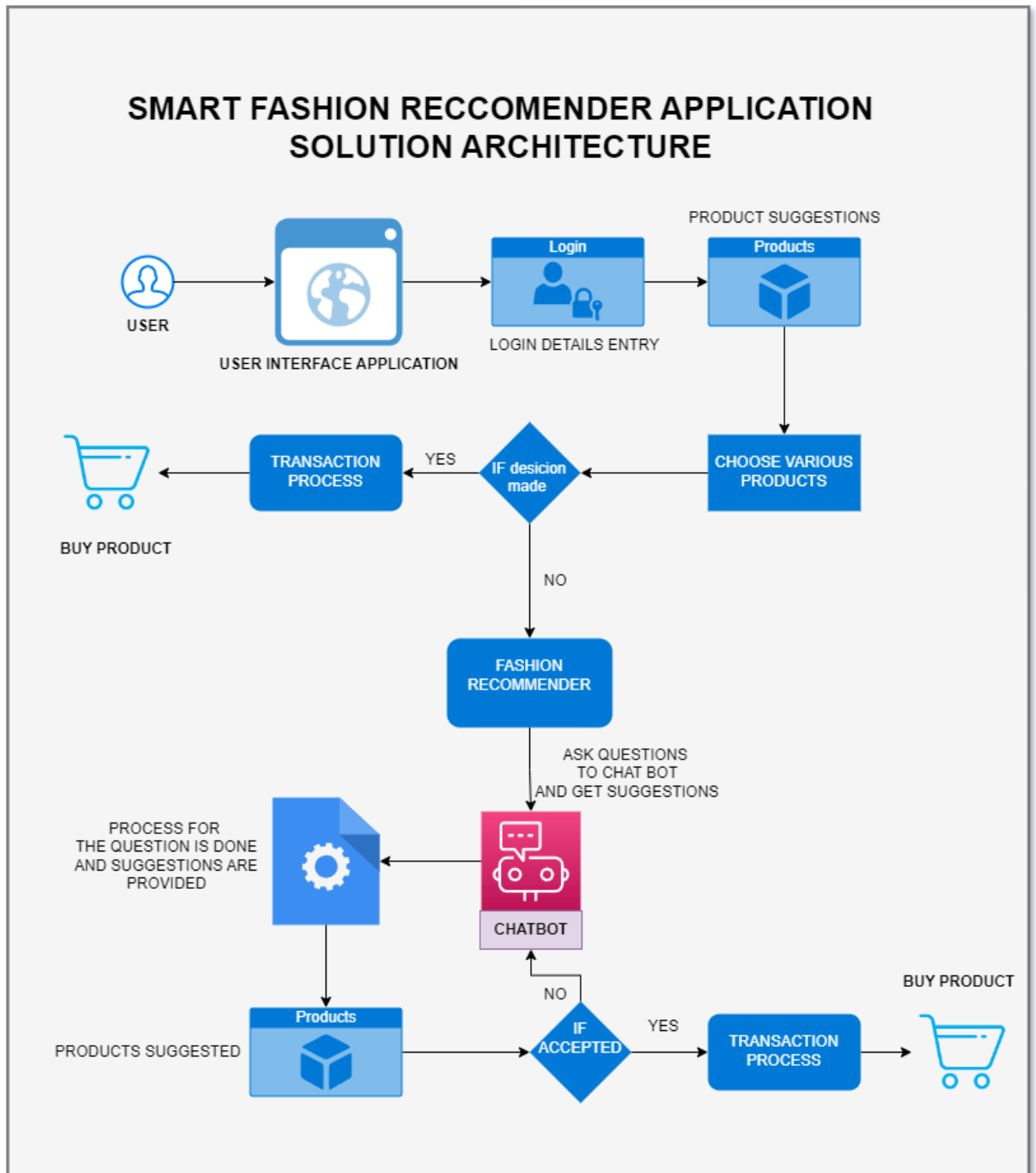
5.PROJECT DESIGN

5.1 Data Flow Diagrams

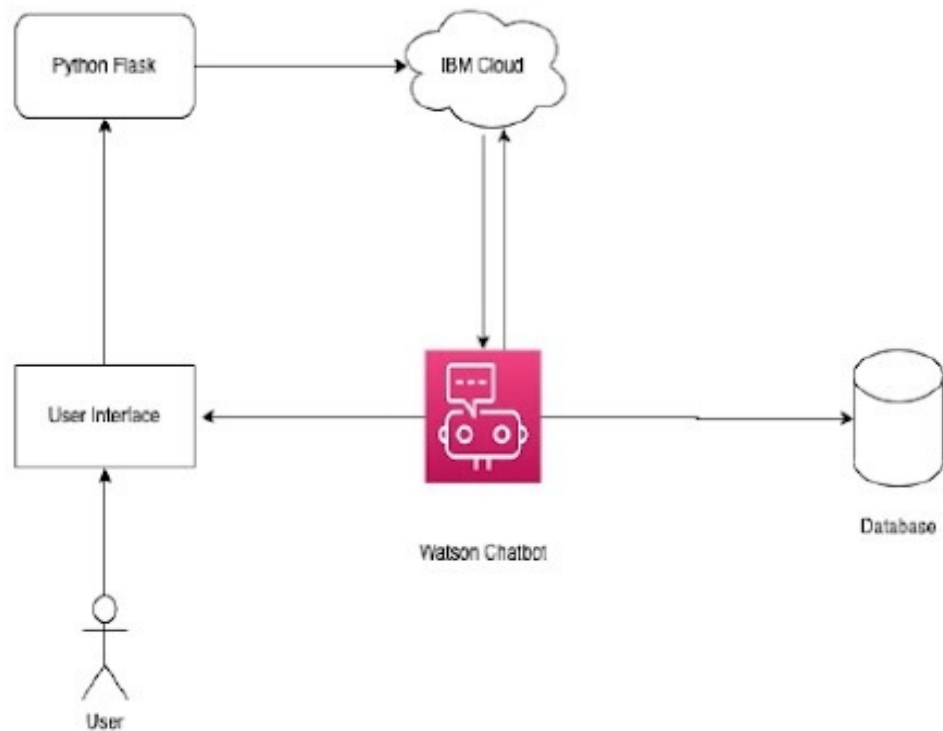
Example: DFD Level 0 (Industry Standard)



5.2 Solution & Technical Architecture



Example: (Simplified)



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile 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

		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user, I can check all the clothing suggestions available and choose my required dresses		high	Sprint-1
Customer (Webuser)	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
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1

		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
Customer Care Executive	Customer support	USN-1	As a user i want to contact the customer care for finding the queries with the products and the suggestions	I was able to attend the customer care very easily and it was really easy	high	sprint-1
Administrator	chatbot	USN-1	As a user i want to check with my easy suggestion process where i can interact and get the things which i want and get the correct suggestions	I got the results accurately and it was good to check the trendy things	high	sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & 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.	2	high	3
Sprint - 1		USN-2	As a user, I will receive a confirmation email once I have registered for the application	1	high	1
Sprint - 2		USN-3	As a user, I can register for the application through Facebook	2	low	1
Sprint - 1		USN-4	As a user, I can register for the application through google account	2	high	1

Sprint - 1	LOGIN	USN-5	As a user, I can log into the	1	high	1
------------	-------	-------	-------------------------------	---	------	---

			application by entering email & password			
Sprint-1	DASHBOARD	USN-6	As a customer I can check with all the clothes available on the website and choose the ones which I want	3	high	3
Sprint-4	Customer support	USN-7	As a user I want to contact with the customer support when there is any query with the product or the application	2	low	2
Sprint-1	User details display	USN-8	As a customer I should be able to see all my details filled with the registration process	2	high	2
Sprint-2	Recommendation system	USN-9	As a customer, I should be able to get a recommendation based on the latest products in market trends	5	high	4
Sprint-2		USN-10	As a customer I should be updated with various offers available	2	medium	2
Sprint-3		USN-11	As a customer I should be able to chat and get the reviews of other customers	3	high	2

Sprint-4	Cart management	USN-12	As a customer I can add and manage all the chosen products and place my order	5	high	4
----------	-----------------	--------	---	---	------	---

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	6 Days	24 Oct 2022	29 Oct 2022	20	29 oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 nov 2022

6.3 Reports from JIRA

The screenshot shows the Jira Roadmap view for the 'smart fashion' project. The left sidebar contains navigation options: Roadmap (selected), Board, Code, Project pages, Add shortcut, and Project settings. The main area displays a roadmap grid with columns for August and September. The grid lists four tasks, all marked as 'DONE':

Task	Status
SF-2 create user design using fla...	DONE
SF-7 create interactive design for...	DONE
SF-8 attach a chat bot	DONE
SF-9 deploy in docker	DONE

Below the grid is a '+ Create Epic' button. At the bottom right, there are tabs for 'Today', 'Weeks', 'Months' (selected), and 'Quarters'. A 'Quickstart' panel on the right offers a 'Create a project' guide, including steps like 'Map out your project goals', 'Identify small chunks of work', and 'Monitor and manage risk'.

The screenshot shows the Jira SF board view for the 'smart fashion' project. The left sidebar is the same as the previous view. The main area displays a Kanban board with three columns: 'TO DO', 'IN PROGRESS', and 'DONE 5 ISSUES'. The 'TO DO' column has a '+ Create issue' button. The 'IN PROGRESS' column is empty. The 'DONE 5 ISSUES' column contains five tasks, all marked as 'DONE':

Task	Status
create the user interface design	DONE
create the interactive design for the mobile application	DONE
upload all the images using the cIBM cloud	DONE
attach a chat bot	DONE
deploy using docker	DONE

A banner at the top asks 'Does your team need more from Jira? Get a free trial of our Standard plan.' The 'Quickstart' panel on the right offers a 'Create a project' guide, including steps like 'Let work flow with kanban' and 'Identify small chunks of work'.

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

7.1 Feature 1

```
from flask import Flask;
from flask import render_template;

app = Flask(__name__)

@app.route("/")
def welcome():
    return render_template('home.html')

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

.dklogo{
    margin-right: 50px;
    margin-left: 20px;
}

.searchbar{
    padding: 0.375rem 0.75rem;
    margin-right: 10px;
    border: 1px solid gray;
    border-radius: 10px;
}

.sreachbarbutton{
    margin-right: 10px;
    padding: 0.375rem 0.75rem;
    border: 1px solid #42ba96;
```

```
}
```

```
.sreachbarbutton:hover {  
  background-color: #42ba96;  
  color: white;  
}
```

```
.itemsDisInfo{  
  display: grid;  
  grid-template-columns: auto auto auto;  
  
}
```

```
.smartfashionheading{  
  position: absolute;  
  top: 20%;  
  left: 32%;  
  color: aliceblue;  
}
```

```
.drouptext{  
  margin-left: 4%;  
}
```

```
.droupbuttonstyle{  
  margin-left: 30%;  
  width: fit-content;  
}
```

```
.drouptextm{  
  margin-right: 60%;  
}
```

```
.drouptextw{  
  margin-right: 52%;  
}
```

```
.drouptextk{  
  margin-right: 62%;  
}
```

```
.drouptexte{  
  margin-right: 49%;  
}
```

```
.login{  
  border: 1px solid black;  
  width: fit-content;  
  margin: auto;  
  margin-top: 6rem;  
  padding: 20px;  
}
```

```
.loginHeader{  
  text-align: center;  
  color: blue;  
}
```

```
.loginforminput{  
  display: block;  
  margin: 20px;  
}
```

```
.loginbutton{  
  text-align: center;  
  margin: 1rem;  
  margin-left: 80px;  
}
```

```
.forgetPasswordlink{  
  text-align: right;  
}
```

```
.forgetPasswordlinktext{  
  margin-right: 10px;  
}
```

```
.loginregisterbutton{  
  margin-left: 30%;  
}
```

```
.registation{  
  border: 1px solid black;  
  width: fit-content;  
  padding: 30px;  
  margin: auto;
```

```
    margin-bottom: 20px;
}
```

```
.registrationheader{
    text-align: center;
    color: blue;
}
```

```
.registrationlable{
    display: block;
}
```

```
.registrationNameinput{
    margin-right: 10px;
    margin: 10px;
}
```

```
.registerPhonenumberinput{
    margin: 10px;
    width: 100%;
    display: block;
}
```

```
.registerEmailinput{
    margin: 10px;
    width: 100%;
    display: block;
}
```

```
.registergender{
    display: block;
}
```

```
.registeraddressinput{
    margin: 10px;
    width: 100%;
    display: block;
}
```

```
.registercityinput{
  margin-right: 10px;
  margin: 10px;
}
```

```
.registerdisticinput{
  margin-right: 10px;
  margin: 10px;
}
```

```
.registerpasswordinput{
  margin: 10px;
  width: 100%;
  display: block;
}
```

```
.registerconfirmpasswordinput{
  margin: 10px;
  width: 100%;
  display: block;
}
```

```
.registrationButton{
  text-align: center;
  margin: 1rem;
  margin-left: 80px;
}
```

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

```
<head>
```

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

```
  <link rel="stylesheet"
```

```
    href="https://image-storage-assignment-3.s3.jp-tok.cloud-object-
storage.appdomain.cloud/index.css">
```

```
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet">
```

```
  <script
```



```
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>
```

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
  integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
  crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
  integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
  crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"
  integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYI"
  crossorigin="anonymous"></script>
<title>home</title>
```

```
</head>
```

```
<body>
```

```
<!-- <div>
  
</div> -->
<div class="headerstyle">
  <div>
    <div>
      <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <a class="navbar-brand dklogo" href="#">
          <img src="" width="90" height="30" class="d-inline-block align-top"
            alt="">
        </a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarText"
          aria-controls="navbarText" aria-expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarText">
          <ul class="navbar-nav mr-auto">
            <li class="nav-item active">
              <a class="nav-link" href="#">Home</a>
```

```

        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Mens</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Womans</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">kids</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Electronics</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Fashion</a>
        </li>
    </ul>
</div>
<div>
    <form class="form-inline ">
        <input class="searchbar" type="search" placeholder="Search">
        <button class="searchbar sreachbarbutton" type="submit">Search</button>
        <a href="/login"><button class="searchbar sreachbarbutton"
type="button">login</button></a>
        <a href="/signup"><button class="searchbar sreachbarbutton"
type="button">Signup</button></a>
    </form>
</div>
</nav>
</div>
</div>
</div>

<div>
    <h3 class="smartfashionheading">Smart Fashion Recommender Application</h3>
    
</div>

```

```

<div class="row">
  <div class="col-3">

    <div class="d-flex flex-column flex-shrink-0 p-3 text-bg-light" style="height: 100%;">
      <a href="/" class="d-flex align-items-center mb-3 mb-md-0 me-md-auto text-white
text-decoration-none">
        <svg class="bi pe-none me-2" width="40" height="32">
          <use xlink:href="#bootstrap" />
        </svg>
        <span class="fs-4 text-black">MENU</span>
      </a>
      <hr>
      <ul class="nav nav-pills flex-column mb-auto">
        <li class="nav-item">
          <a href="#" class="nav-link text-black" aria-current="page">
            <svg class="bi pe-none me-2" width="16" height="16">
              <use xlink:href="#home" />
            </svg>
            <button class="btn btn-light w-100 text-start drouptext">Home</button>
          </a>
        </li>
        <li>
          <div class="btn-group w-100 mt-3">
            <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-
split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
              <span class="sr-only drouptextm">Mens</span>
            </button>
            <div class="dropdown-menu w-100">
              <a class="dropdown-item" href="#">Action</a>
              <a class="dropdown-item" href="#">Another action</a>
              <a class="dropdown-item" href="#">Something else here</a>
            </div>
          </div>
        </li>
        <li>
          <div class="btn-group w-100 mt-3">
            <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-
split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
              <span class="sr-only drouptextw">Womans</span>
            </button>
            <div class="dropdown-menu w-100">

```

```

        <a class="dropdown-item" href="#">Action</a>
        <a class="dropdown-item" href="#">Another action</a>
        <a class="dropdown-item" href="#">Something else here</a>
    </div>
</div>
</li>
<li>
    <div class="btn-group w-100 mt-3">
        <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
            <span class="sr-only drouptextk">Kids</span>
        </button>
        <div class="dropdown-menu w-100">
            <a class="dropdown-item" href="#">Action</a>
            <a class="dropdown-item" href="#">Another action</a>
            <a class="dropdown-item" href="#">Something else here</a>
        </div>
    </div>
</li>
<li>
    <div class="btn-group w-100 mt-3">
        <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
            <span class="sr-only drouptexte">Electronic</span>
        </button>
        <div class="dropdown-menu w-100">
            <a class="dropdown-item" href="#">Action</a>
            <a class="dropdown-item" href="#">Another action</a>
            <a class="dropdown-item" href="#">Something else here</a>
        </div>
    </div>
</li>
</ul>
</div>

```

```
</div>
```

```
<div class="col-9 itemsDisInfo">
```

```
<div>
```

```
<!-- <a style="float:right;"
```

```

        href="https://api.au-syd.assistant.watson.cloud.ibm.com/instances/3b89fcd1-0be9-
4167-9840-a77d7f0a9f7f">URL
        FOR
        HOSPITAL CHAT BOT</a>
        <h1 class="ibmheading">Bucked created in IBM Object Storage</h1>
        <h3 class="mb-5 ibmheading">Accessing the 5 public images & a css page uploaded
in IBM Object Storage
        </h3> -->

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

</html>

```

7.2 Feature 2

```

from flask import Flask;
from flask import render_template;

app = Flask(__name__)

@app.route("/")
def welcome():
    return render_template('home.html')

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

import io

```

```

from flask import Flask,redirect,url_for,render_template,request
import ibm_boto3
from ibm_botocore.client import Config, ClientError

COS_ENDPOINT="https://s3.tok.ap.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID=""
COS_INSTANCE_CRN=""

cos = ibm_boto3.resource("s3",
    ibm_api_key_id=COS_API_KEY_ID,
    ibm_service_instance_id=COS_INSTANCE_CRN,
    config=Config(signature_version="oauth"),
    endpoint_url=COS_ENDPOINT
)

app=Flask(__name__)

@app.route('/')
def index():
    try:
        files = cos.Bucket('hospital-flask').objects.all()
        files_names = []
        for file in files:
            files_names.append(file.key)
            print(file)
            print("Item: {0} ({1} bytes)".format(file.key, file.size))
        return render_template('index.html',files=files_names)

    except ClientError as be:
        print("CLIENT ERROR: {0}\n".format(be))
        return render_template('index.html')
    except Exception as e:
        print("Unable to retrieve bucket contents: {0}".format(e))
        return render_template('index.html')

@app.route('/uploader',methods=['POST'])
def upload():
    name_file=request.form['filename']
    f = request.files['file']

```

```

try:
    part_size = 1024 * 1024 * 5

    file_threshold = 1024 * 1024 * 15

    transfer_config = ibm_boto3.s3.transfer.TransferConfig(
        multipart_threshold=file_threshold,
        multipart_chunksize=part_size
    )

    content = f.read()
    cos.Object('hospital-flask', name_file).upload_fileobj(
        Fileobj=io.BytesIO(content),
        Config=transfer_config
    )
    return redirect(url_for('index'))

except ClientError as be:
    print("CLIENT ERROR: {0}\n".format(be))
    return redirect(url_for('index'))

except Exception as e:
    print("Unable to complete multi-part upload: {0}".format(e))
    return redirect(url_for('index'))

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

```

8. TESTING

8.1 Test Cases

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2 User Acceptance Testing

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	8	6	1	4	20
Duplicate	2	0	2	0	4
External	2	2	0	2	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

9. RESULTS

9.1 Performance Metrics

2			NFT - Risk Assessment							
3	S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Volume Changes	Risk Score	Justification
4	1	Smart Fashion Reccor	UI/UX	Low	No Changes	Moderate		>5 to 10%	ORANGE	this has the front page
5	2	Smart Fashion Recommendation syst	chat bot	low	no changes	moderate		<3 to 5%	orange	this acts as a interactive system for recommendation
6			NFT - Detailed Test Plan							
7										
8			S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff			
9			1	smart fashion recommen	recommendation system	improper algorithm and malfunction	APPROVED			
10			End Of Test Report							
11										
12	S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff	
13	1	recommendation system	performance	yes	passed in the test	GO	nil	detected and closed	approved	
14										
15										

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES :

- Using chatbot we can manage user's choices and orders.
- The chatbot can give recommendations to the users based on their interests.
- It can promote the best deals and offers on that day.
- It will store the customer's details and orders in the database.
- The chatbot will send a notification to customers if the order is confirmed.
- Chatbots can also help in collecting customer feedback.

DISADVANTAGES :

yet to become much efficient and takes more time to be defined

11. CONCLUSION

Software Required:

Python, Flask , Docker

System Required:

8GB RAM, Intel Core i3, OS-Windows/Linux/MAC , Laptop or Desktop

We have come up with a new innovative solution through which you can directly do your online shopping based on your choice without any search. It can be done by using the chatbot.

In this project you will be working on two modules :

1. Admin and
2. User

Admin:

The role of the admin is to check out the database about the stock and have a track of all the things that the users are purchasing.

User :

The user will login into the website and go through the products available on the website. Instead of navigating to several screens for booking products online, the user can directly talk to Chatbot regarding the products. Get the recommendations based on information provided by the user.

Features of Chatbot :

- Using chatbot we can manage user's choices and orders.
- The chatbot can give recommendations to the users based on their interests.
- It can promote the best deals and offers on that day.
- It will store the customer's details and orders in the database.
- The chatbot will send a notification to customers if the order is confirmed.
- Chatbots can also help in collecting customer feedback.

12. FUTURE SCOPE

This application acts as one of the different application and is much useful. various algorithms and APIs can be added to make it much useful in the future.

13. APPENDIX

13.1 Source Code

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

<head>
  <link rel="stylesheet" href="/static/homepage.css">
  <link rel="stylesheet"
    href="https://image-storage-assignment-3.s3.jp-tok.cloud-object-
storage.appdomain.cloud/index.css">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet">
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>

  <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
    integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
    crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
    integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
    crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"
    integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYI"
    crossorigin="anonymous"></script>
  <title>home</title>

</head>
```

```

<body>
  <!-- <div>
    
  </div> -->
  <div class="headerstyle">
    <div>
      <div>
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <a class="navbar-brand dklogo" href="#">
            <img src="" width="90" height="30" class="d-inline-block align-top"
              alt="">
          </a>
          <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarText"
            aria-controls="navbarText" aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse" id="navbarText">
            <ul class="navbar-nav mr-auto">
              <li class="nav-item active">
                <a class="nav-link" href="#">Home</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="#">Mens</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="#">Womans</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="#">kids</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="#">Electronics</a>
              </li>
              <li class="nav-item">
                <a class="nav-link" href="#">Fashion</a>
              </li>
            </ul>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

        <form class="form-inline ">
            <input class="searchbar" type="search" placeholder="Search">
            <button class="searchbar sreachbarbutton" type="submit">Search</button>
            <a href="/login"><button class="searchbar sreachbarbutton"
type="button">login</button></a>
            <a href="/signup"><button class="searchbar sreachbarbutton"
            type="button">Signup</button></a>
        </form>
    </div>
</nav>
</div>
</div>
</div>

```

```

</div>

```

```

<div>
    <h3 class="smartfashionheading">Smart Fashion Recommender Application</h3>
    
</div>
<div class="row">
    <div class="col-3">

```

```

        <div class="d-flex flex-column flex-shrink-0 p-3 text-bg-light" style="height: 100%;">
            <a href="/" class="d-flex align-items-center mb-3 mb-md-0 me-md-auto text-white
text-decoration-none">
                <svg class="bi pe-none me-2" width="40" height="32">
                    <use xlink:href="#bootstrap" />
                </svg>
                <span class="fs-4 text-black">MENU</span>
            </a>
            <hr>
            <ul class="nav nav-pills flex-column mb-auto">
                <li class="nav-item">
                    <a href="#" class="nav-link text-black" aria-current="page">
                        <svg class="bi pe-none me-2" width="16" height="16">
                            <use xlink:href="#home" />
                        </svg>
                        <button class="btn btn-light w-100 text-start drouptext">Home</button>

```

```

        </a>
    </li>
    <li>
        <div class="btn-group w-100 mt-3">
            <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                <span class="sr-only drouptextm">Mens</span>
            </button>
            <div class="dropdown-menu w-100">
                <a class="dropdown-item" href="#">Action</a>
                <a class="dropdown-item" href="#">Another action</a>
                <a class="dropdown-item" href="#">Something else here</a>
            </div>
        </div>
    </li>
    <li>
        <div class="btn-group w-100 mt-3">
            <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                <span class="sr-only drouptextw">Womans</span>
            </button>
            <div class="dropdown-menu w-100">
                <a class="dropdown-item" href="#">Action</a>
                <a class="dropdown-item" href="#">Another action</a>
                <a class="dropdown-item" href="#">Something else here</a>
            </div>
        </div>
    </li>
    <li>
        <div class="btn-group w-100 mt-3">
            <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                <span class="sr-only drouptextk">Kids</span>
            </button>
            <div class="dropdown-menu w-100">
                <a class="dropdown-item" href="#">Action</a>
                <a class="dropdown-item" href="#">Another action</a>
                <a class="dropdown-item" href="#">Something else here</a>
            </div>
        </div>
    </li>

```

```
</li>
  <div class="btn-group w-100 mt-3">
    <button type="button" class="btn btn-light dropdown-toggle dropdown-toggle-split" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
      <span class="sr-only drouptexte">Electronic</span>
    </button>
    <div class="dropdown-menu w-100">
      <a class="dropdown-item" href="#">Action</a>
      <a class="dropdown-item" href="#">Another action</a>
      <a class="dropdown-item" href="#">Something else here</a>
    </div>
  </div>
</li>
</ul>
</div>
```

```
</div>
<div class="col-9 itemsDisInfo">
  <div>
```

```
    <!-- <a style="float:right;"
      href="https://api.au-syd.assistant.watson.cloud.ibm.com/instances/3b89fcd1-0be9-
4167-9840-a77d7f0a9f7f">URL
      FOR
      HOSPITAL CHAT BOT</a>
    <h1 class="ibmheading">Bucked created in IBM Object Storage</h1>
    <h3 class="mb-5 ibmheading">Accessing the 5 public images & a css page uploaded
in IBM Object Storage
    </h3> -->
```

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

```
</html>
```

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

```

<head>
  <link rel="stylesheet" href="/static/loginpage.css">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet">
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>
  <title>login</title>
</head>
<body style="background-image: url('/static/backimage.jpg');background-repeat: no-
repeat;background-size: cover;">
  <div class="login" style="background-color: white;">
    <h2 class="loginHeader">Login</h2>
    <form>
      <input type="text" placeholder="User name or Email" class="loginforminput"
id="loginformuseremail">
      <input type="password" placeholder="password" class="loginforminput"
id="loginformuserpassword">
      <div>
        <a href="#" class="forgetPasswordlink">
          <p class="forgetPasswordlinktext">Forget Password?</p>
        </a>
      </div>

      <div>
        <input type="submit" value="Login" class="btn btn-primary loginbutton">
      </div>
      <hr>
      <div>
        <p>If you are new user then Register</p>
        <a href="/signup"><input type="button" class="btn btn-primary loginregisterbutton"
value="SIGN UP"></a>
      </div>
    </form>
  </div>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" href="/static/signuppage.css">

```



```

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet">
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></script>
    <title>WELCOME</title>
</head>
<body style="background-image: url('/static/backimage.jpg');background-repeat: no-
repeat;background-size: cover;">
    <div class="registration" style="background-color: white;">
        <h3 class="registrationheader">SIGN UP</h3>
        <form>
            <div formGroupName="name">
                <label for="Name" class="registrationlable">Name:</label>
                <input type="text" placeholder="First Name" name="Firstname"
class="registrationNameinput">
                <input type="text" placeholder="Last Name" name="Lastname"
class="registrationNameinput">
            </div>

            <label for="phonenumber" class="registrationlable">Phone Number:</label>
            <input type="number" placeholder="Phone Number" class="registerPhonenumberinput"
name="phonenumber">

            <label for="Email" class="registrationlable">Email</label>
            <input type="email" placeholder="Email" class="registerEmailinput">

            <div>
                <label for="Address" class="registrationlable">Address:</label>
                <input type="text" placeholder="Street Address Line1" class="registeraddressinput">
                <input type="text" placeholder="Street Address Line2" class="registeraddressinput">
                <input type="text" placeholder="City" class="registercityinput" >
                <input type="text" placeholder="Region" class="registerregioninput" >
                <div>
                    <input type="text" placeholder="Distic" class="registerdisticinput" >
                    <input type="number" placeholder="Pin-Code" class="registerpininput">
                </div>
            </div>

            <div>

```

```

        <label for="gender" class="registrationlable">Gender:</label>
        <div>
            <input type="radio" name="gender" value="male">Male
        </div>
        <div>
            <input type="radio" name="gender" value="female">Female
        </div>
        <div>
            <input type="radio" name="gender" value="others">others
        </div>

    </div>

    <div>
        <label for="password" class="registrationlable">Password:</label>
        <input type="password" placeholder="Password" class="registerpasswordinput">
        <input type="password" placeholder="Confirm Password"
class="registerconfirmpasswordinput">
    </div>

    <div>
        <input type="submit" class="btn btn-primary registrationButton" value="Register">
    </div>
</form>
</div>
</body>
</html>

```

```

.dklogo{
    margin-right: 50px;
    margin-left: 20px;
}

```

```

.searchbar{
    padding: 0.375rem 0.75rem;
    margin-right: 10px;
    border: 1px solid gray;
    border-radius: 10px;
}

```

```
}
```

```
.sreachbarbutton{  
  margin-right: 10px;  
  padding: 0.375rem 0.75rem;  
  border: 1px solid #42ba96;  
}
```

```
.sreachbarbutton:hover {  
  background-color: #42ba96;  
  color: white;  
}
```

```
.itemsDisInfo{  
  display: grid;  
  grid-template-columns: auto auto auto;  
  
}
```

```
.smartfashionheading{  
  position: absolute;  
  top: 20%;  
  left: 32%;  
  color: aliceblue;  
}
```

```
.drouptext{  
  margin-left: 4%;  
}
```

```
.droupbuttonstyle{  
  margin-left: 30%;  
  width: fit-content;  
}
```

```
.drouptextm{  
  margin-right: 60%;  
}
```

```
.drouptextw{  
  margin-right: 52%;  
}
```

```
.drouptextk{  
    margin-right: 62%;  
}
```

```
.drouptexte{  
    margin-right: 49%;  
}
```

```
.login{  
    border: 1px solid black;  
    width: fit-content;  
    margin: auto;  
    margin-top: 6rem;  
    padding: 20px;  
}
```

```
.loginHeader{  
    text-align: center;  
    color: blue;  
}
```

```
.loginforminput{  
    display: block;  
    margin: 20px;  
}
```

```
.loginbutton{  
    text-align: center;  
    margin: 1rem;  
    margin-left: 80px;  
}
```

```
.forgetPasswordlink{  
    text-align: right;  
}
```

```
.forgetPasswordlinktext{  
    margin-right: 10px;  
}
```

```
.loginregisterbutton{  
    margin-left: 30%;  
}
```

```
}
```

```
.registration{  
  border: 1px solid black;  
  width: fit-content;  
  padding: 30px;  
  margin: auto;  
  margin-bottom: 20px;  
}
```

```
.registrationheader{  
  text-align: center;  
  color: blue;  
}
```

```
.registrationlable{  
  display: block;  
}
```

```
.registrationNameinput{  
  margin-right: 10px;  
  margin: 10px;  
}
```

```
.registerPhonenumberinput{  
  margin: 10px;  
  width: 100%;  
  display: block;  
  
}
```

```
.registerEmailinput{  
  margin: 10px;  
  width: 100%;  
  display: block;  
  
}
```

```
.registergender{  
  display: block;  
}
```

```
.registeraddressinput{  
  margin: 10px;  
  width: 100%;  
  display: block;  
  
}
```

```
.registercityinput{  
  margin-right: 10px;  
  margin: 10px;  
}
```

```
.registerdisticinput{  
  margin-right: 10px;  
  margin: 10px;  
}
```

```
.registerpasswordinput{  
  margin: 10px;  
  width: 100%;  
  display: block;  
}
```

```
.registerconfirmpasswordinput{  
  margin: 10px;  
  width: 100%;  
  display: block;  
}
```

```
.registrationButton{  
  text-align: center;  
  margin: 1rem;  
  margin-left: 80px;  
}
```

```

import io
from flask import Flask,redirect,url_for,render_template,request
import ibm_boto3
from ibm_botocore.client import Config, ClientError

COS_ENDPOINT="https://s3.tok.ap.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID=""
COS_INSTANCE_CRN=""

cos = ibm_boto3.resource("s3",
    ibm_api_key_id=COS_API_KEY_ID,
    ibm_service_instance_id=COS_INSTANCE_CRN,
    config=Config(signature_version="oauth"),
    endpoint_url=COS_ENDPOINT
)

app=Flask(__name__)

@app.route('/')
def index():
    try:
        files = cos.Bucket('hospital-flask').objects.all()
        files_names = []
        for file in files:
            files_names.append(file.key)
            print(file)
            print("Item: {0} ({1} bytes)".format(file.key, file.size))
        return render_template('index.html',files=files_names)

    except ClientError as be:
        print("CLIENT ERROR: {0}\n".format(be))
        return render_template('index.html')
    except Exception as e:
        print("Unable to retrieve bucket contents: {0}".format(e))
        return render_template('index.html')

@app.route('/uploader',methods=['POST'])
def upload():
    name_file=request.form['filename']

```

```

f = request.files['file']
try:
    part_size = 1024 * 1024 * 5

    file_threshold = 1024 * 1024 * 15

    transfer_config = ibm_boto3.s3.transfer.TransferConfig(
        multipart_threshold=file_threshold,
        multipart_chunksize=part_size
    )

    content = f.read()
    cos.Object('hospital-flask', name_file).upload_fileobj(
        Fileobj=io.BytesIO(content),
        Config=transfer_config
    )
    return redirect(url_for('index'))

except ClientError as be:
    print("CLIENT ERROR: {0}\n".format(be))
    return redirect(url_for('index'))

except Exception as e:
    print("Unable to complete multi-part upload: {0}".format(e))
    return redirect(url_for('index'))

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

from flask import Flask;

```



```

from flask import render_template;

app = Flask(__name__)

@app.route("/")
def welcome():
    return render_template('home.html')

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

```

```

from flask import Flask;
from flask import render_template;

app = Flask(__name__)

@app.route("/")
def welcome():
    return render_template('home.html')

@app.route('/signup')
def singup():
    return render_template('signup.html')

@app.route('/login')
def login():
    return render_template('loginpage.html')

```

13.2 GitHub & Project Demo Link

<https://github.com/IBM-EPBL/IBM-Project-53401-1661400149>

https://drive.google.com/file/d/1icuvu1PpX50w1NFaqBhLRZPKD1ynwgIX/view?usp=share_link

https://drive.google.com/file/d/1rtGs5eiputDTohBNBvBD5xbT_1MxEIKB/view?usp=sharing