PAAVAI ENGINEERING COLLEGE

(AUTONOMOUS)

SMART FASHION RECOMMENDER APPLICATION

Submitted by

VIJAYAKUMAR.R(TL)	**19104122**
ROHKITH ROSHAN.S	**19104082**
MUTHUPANDI.S	**19104066**
SUJITH.S	**19104105**

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

1.INTRODUCTION

Clothing is a kind of symbol that represents people's internal perceptions through their outer appearance. It conveys information about their choices, faith, personality, profession, social status, and attitude towards life. Therefore, clothing is believed to be a nonverbal way of communicating and a major part of people's outer appearance. Recent technological advancements have enabled consumers to track current fashion trends around the globe, which influence their choices. The fashion choices of consumers depend on many factors, such as demographics, geographic location, individual preferences, interpersonal influences, age, gender, season, and culture. Moreover, previous fashion recommendation research shows that fashion preferences vary not only from country to country but also from city to city. The combination of fashion preferences and the abovementioned factors associated with clothing choices could transmit the image features for a better understanding of consumers' preferences.

1.1 Project Overview:

The recommendation phase recommends the types of items that a user or consumer may prefer. Recommendations can be provided either directly based on the dataset collected during the information collection phase (which might be memory- or model-based) or through the browsing history of users observed by the system . Recommendations can also be provided by combining the learned information with the rating matrix to recommend learning resources

1.2 Purpose:

- Users find it difficult to choose their product, here the bot will assist user in receiving product recommendation.
- To reduce search time, from the user interaction with bot, the similar product will be displayed based on user's requirements.
- The implemented 3D model will help user to decide how the product will look on them.
- This would be a one stop solution for all kinds of users.

The purpose of a recommender system is to suggest relevant items to users. To achieve this task, there exist two major categories of methods:collaborative filtering methods and contentbased methods.

Collaborative filtering methods

Collaborative methods for recommender systems are methods that are based solely on the past interactions recorded between users and items to produce new recommendations. These interactions are stored in the so-called "user-item interactions matrix".

Content based methods

Unlike collaborative methods that only rely on the user-item interactions, content-based approaches use additional information
about users and/or items. If we consider the example of a movies recommender system, this additional information can be, for example, the age, the sex, the job or any other personal information for users as well as the category, the main actors, the duration or other characteristics for the movies (items).

2.LITERATURE SURVEY

This abstract proposed a personalized Fashion Recommender system that generates recommendations for the user based on an input given. Unlike the conventional systems that rely on the user's previous purchases and history, this project aims at using an image of a product given as input by the user to generate recommendations since many-a- time people see something that they are interested in and tend to look for products that are like that. We use neural networks to process the images from Deep Fashion dataset and a nearest neighbor backed recommender to generate the final recommendations

2.1Existing problem:

E-commerce retailers started implementing fashion
recommendation systems in the early 2000s. However, implementation was mostly in the development stage until 2007–2008. As with other products such as electronics and books, fashion products were also recommended based on the user's previous purchase history. With the continuous progress in computer vision algorithms, personalized recommendations utilizing personal factors and user reviews have become more popular today].

2.2References:

[n.d.]. Amazon "Made for You" Custom T-shirt. https://www.amazon.com/Made-for-You-Custom-T-shirt/dp/B08N6J8G5M. Accessed: 2021-04-17.

1.G. Mohammed Abdulla, Shreya Singh, and Sumit Borar. 2019. Shop your Right Size: A System for Recommending Sizes for Fashion products. In Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019. ACM, 327–334. https://doi.org/10.1145/3308560.3316599

2.Adewole Adewumi, Adebola Taiwo, Sanjay Misra, Rytis Maskeliunas, Robertas Damasevicius, Ravin Ahuja, and Foluso Ayeni. 2019. A Unified Framework for Outfit Design and Advice. 31–41. https://doi.org/10.1007/978-981-13-9364-8_3

3.Pankaj Agarwal, Sreekanth Vempati, and Sumit Borar. 2018. Personalizing Similar Product Recommendations in Fashion E- commerce. CoRR abs/1806.11371 (2018). arXiv:1806.11371 http://arxiv.org/abs/1806.11371

4.Charu C Aggarwal. 2016. Ensemble-based and hybrid recommender systems. In Recommender Systems. Springer, 199–224.

5.Divyansh Aggarwal, Elchin Valiyev, Fadime Sener, and Angela Yao. 2018. Learning Style Compatibility for Furniture. In Pattern Recognition - 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings (Lecture Notes in Computer Science), Thomas Brox, Andrés Bruhn, and Mario Fritz (Eds.), Vol. 11269. Springer, 552–566. https://doi.org/10.1007/978-3-030-12939-2_38

6.Kenan E Ak, Ashraf A Kassim, Joo Hwee Lim, and Jo Yew Tham. 2018. Learning attribute representations with localization for flexible fashion search. In Proceedings of the IEEE conference on computer vision and pattern recognition. 7708–7717.

7.Ziad Al-Halah and Kristen Grauman. 2020. From paris to berlin: Discovering fashion style influences around the world. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 10136–10145.

8.Ziad Al-Halah, Rainer Stiefelhagen, and Kristen Grauman. 2017. Fashion forward: Forecasting visual style in fashion. In Proceedings of the IEEE international conference on computer vision. 388–397.

9.Mohammed Al-Rawi and Joeran Beel. 2021. Probabilistic Color Modelling of Clothing Items. Recommender Systems in Fashion and Retail 734 (2021), 21.

10.Phoebe R Apeagyei et al. 2010. Application of 3D body scanning technology to human measurement for clothing fit. International Journal of Digital Content Technology and its Applications 4, 7 (2010), 58–68.

11.Susan P Ashdown, Suzanne Loker, Katherine Schoenfelder, and Lindsay Lyman-Clarke. 2004. Using 3D scans for fit analysis. Journal of Textile and Apparel, Technology and Management 4, 1 (2004), 1–12.

12.Krisztian Balog and Filip Radlinski. 2020. Measuring Recommendation Explanation Quality: The Conflicting Goals of Explanations. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 329–338.

13.Roja Bandari, Sitaram Asur, and Bernardo Huberman. 2021. The Pulse of News in Social Media: Forecasting Popularity. Proceedings of the International AAAI Conference on Web and Social Media 6, 1 (Aug. 2021), 26–33. https://ojs.aaai.org/index.php/ICWSM/article/view/14261

14.Debopriyo Banerjee, Krothapalli Sreenivasa Rao, Shamik Sural, and Niloy Ganguly. 2020. BOXREC: Recommending a Box of Preferred Outfits in Online Shopping. ACM Trans. Intell. Syst. Technol. 11, 6 (2020), 69:1–69:28. https: //doi.org/10.1145/3408890

15.Sean Bell and Kavita Bala. 2015. Learning visual similarity for product design with convolutional neural networks. ACM Trans. Graph. 34, 4 (2015), 98:1–98:10.

https://doi.org/10.1145/2766959

16.Elaine M Bettaney, Stephen R Hardwick, Odysseas Zisimopoulos, and Benjamin Paul Chamberlain. 2019. Fashion outfit generation for E- commerce. arXiv preprint arXiv:1904.00741 (2019).

2.4.Problem Statement Definition:

The main questions we aimed to answer through this literature review can be outlined as follows:

(a) What makes the fashion domain distinctive from other recommender systems domains?

(b) What are the main tasks which have been defined for fashion recommender systems?

(c) How image-based fashion recommender systems have been affected by computer vision advancements?

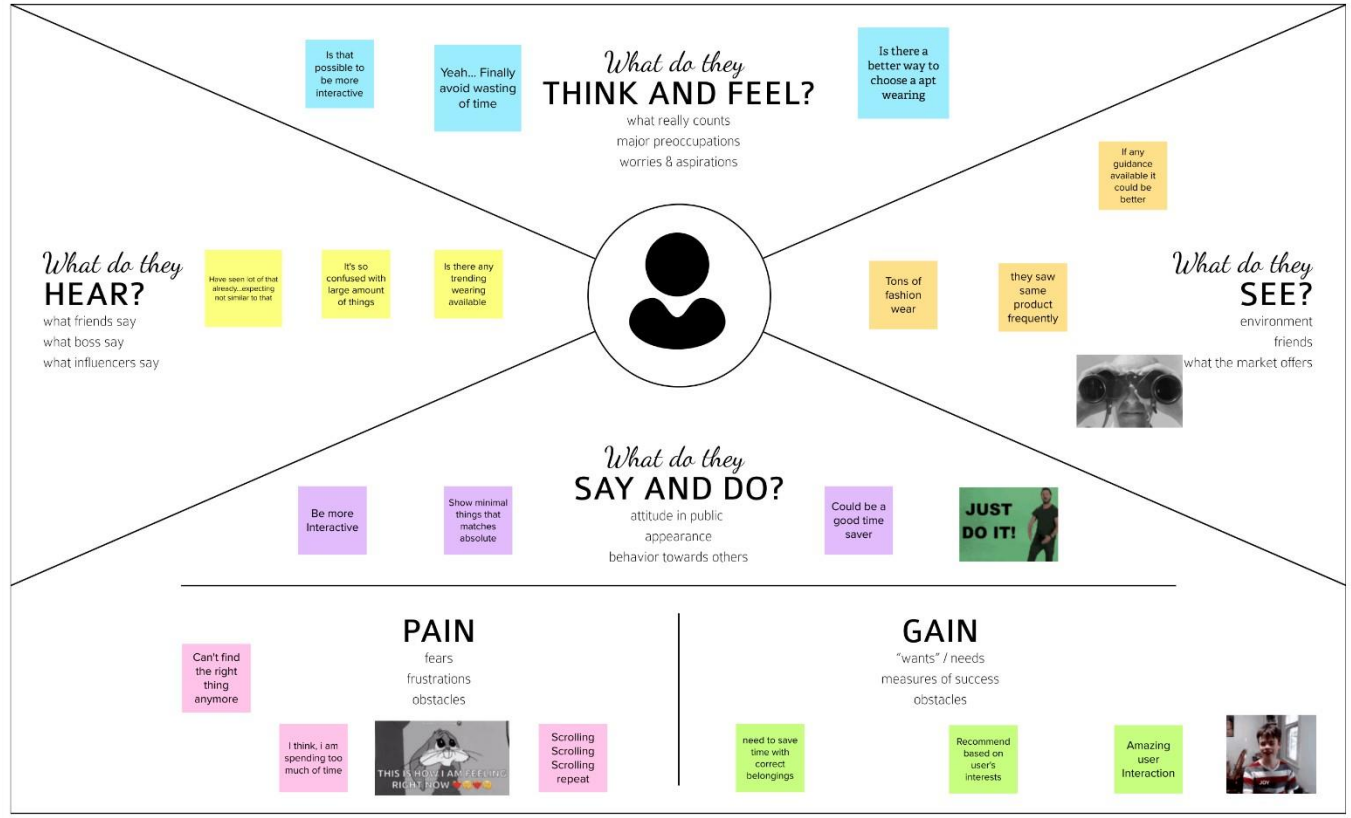
3.IDEATION & PROPOSED SOLUTION

3.1Empathy 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.



3.2.Ideation & Brainstorming:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the- box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Step 1 - Brainstorm & Idea Prioritization

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

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

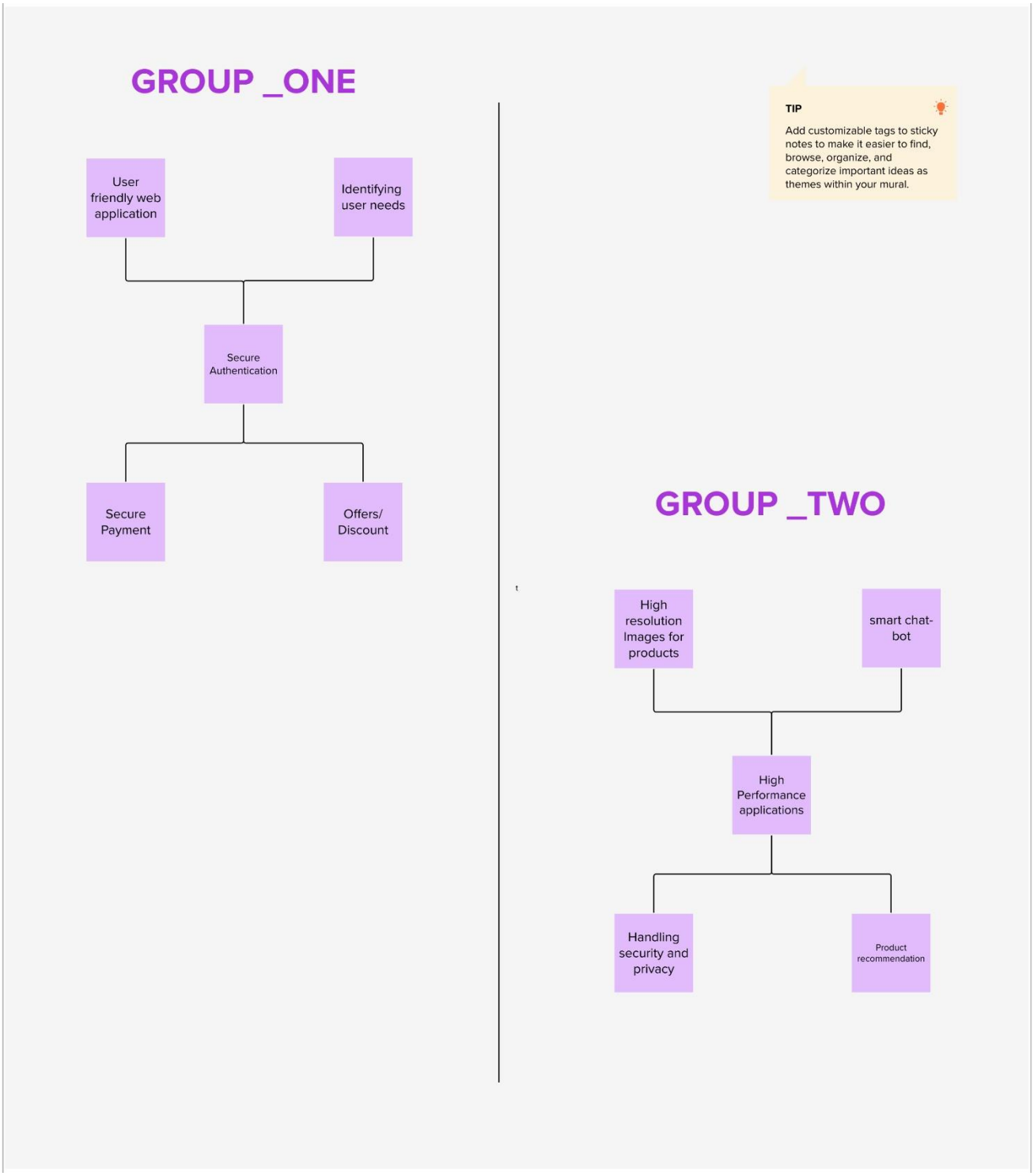
B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

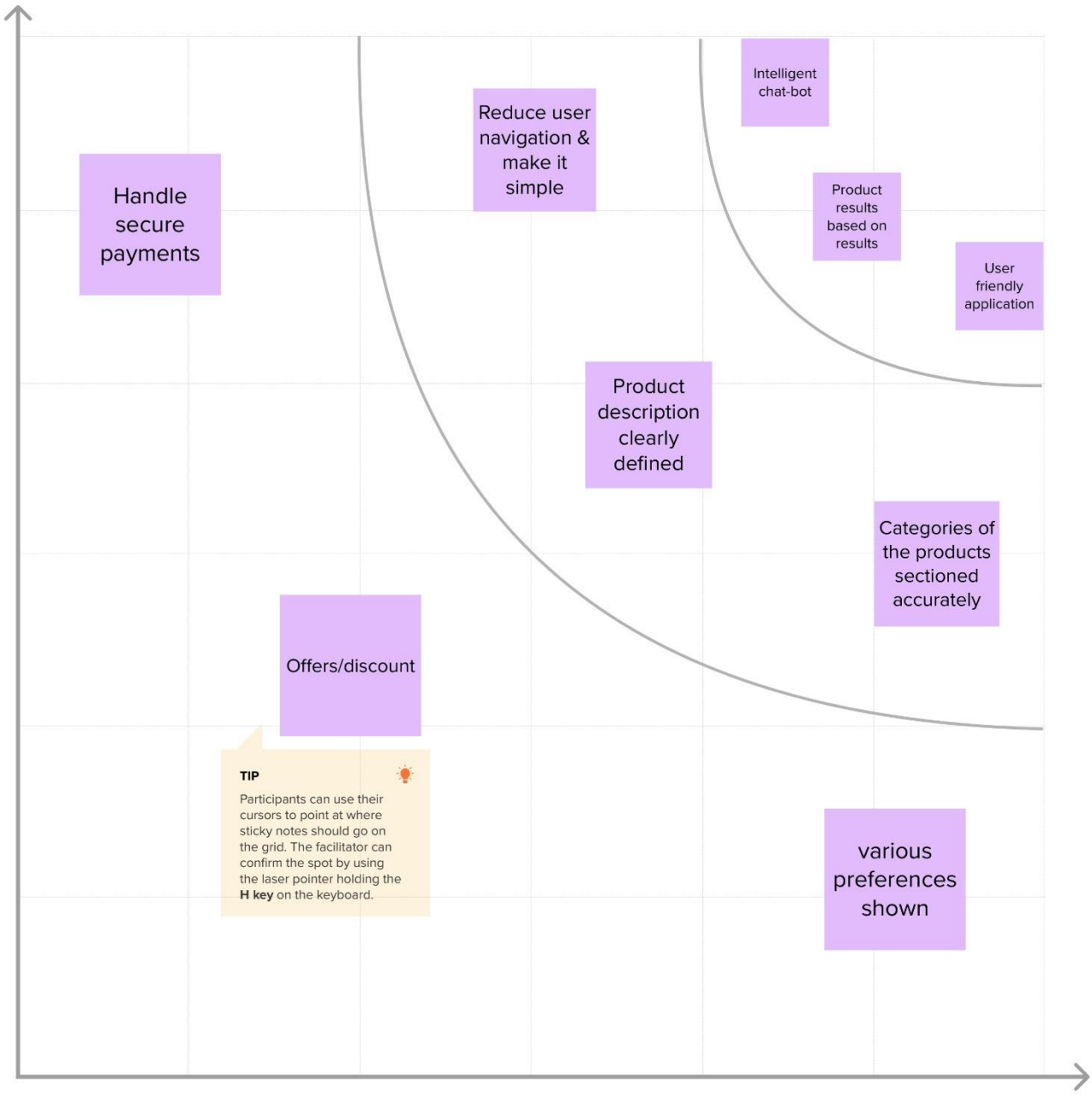
Open article →

Step 2 - Brainstorm

Step 3 - Group Ideas



Step 4 - Prioritize



3.3.Proposed Solution:

Proposed Solution Template:

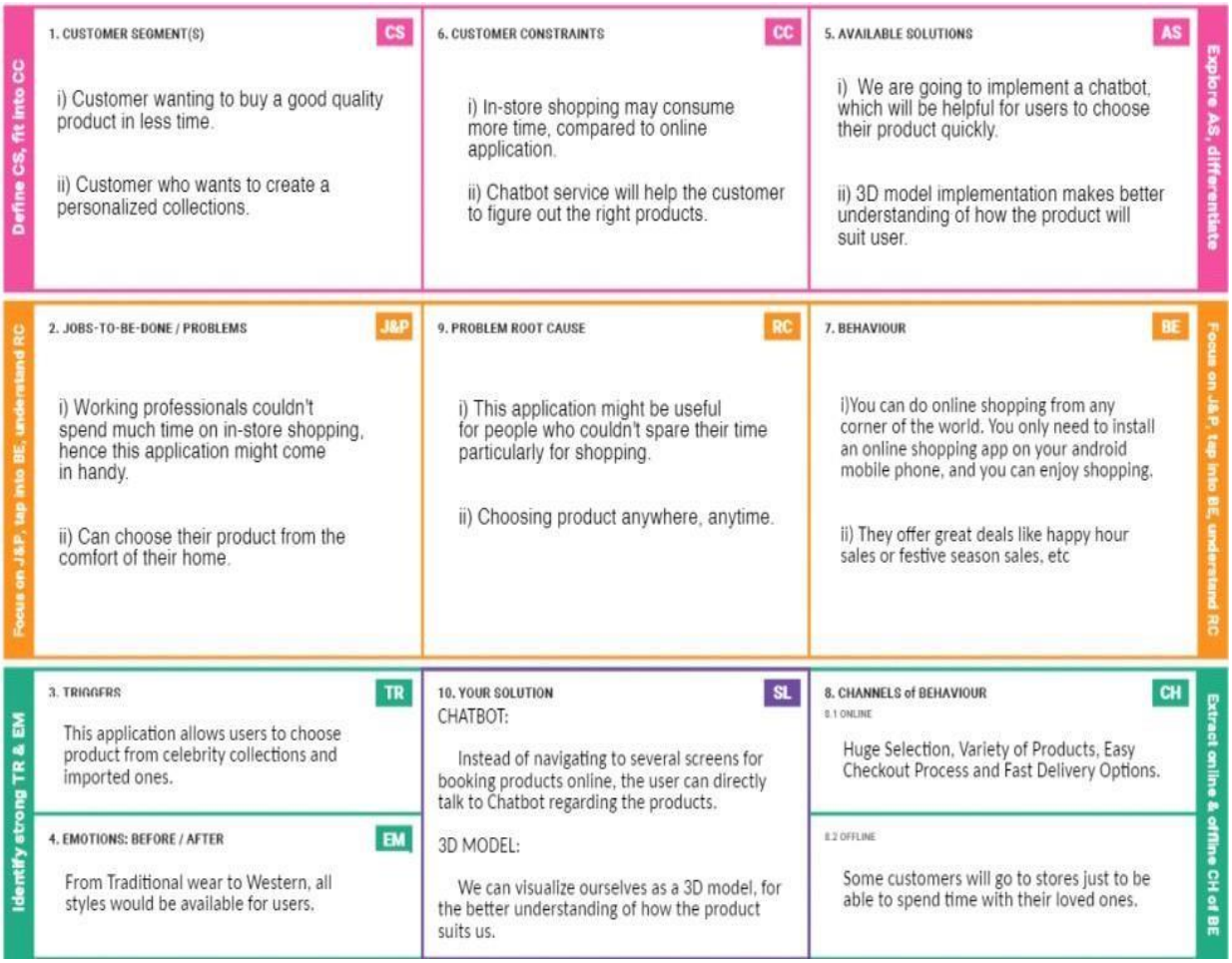
Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	• Complex user interface • Similar products appeared frequently • Proper guidance is not available • Customer less sticky to the application • User need to navigate across multiple pages to choose right product • Lack of interaction between application and user

2.	Idea / Solution description	By using Smart fashion recommender application: ● Collect feedback instantly ● Recommendation within a single page via chat-bot ● Effective recommendation of products ● Improve customer relationship, interactivity and services ● Reduce human error ● Proper guidance in accessing application.
3.	Novelty / Uniqueness	● Chat-bot asks and learns from user preference which recommends appropriate products to the user without making them to search through various filters. Reduces time in choosing right product thus increases sales.
4.	Social Impact / Customer Satisfaction	● Feedback from the user at the end of session or after placing order is one of the most important factor in deriving customer satisfaction and providing better services.
5.	Business Model (Revenue Model)	● The application can be developed at minimum cost with high performance and interactive user interface.
6.	Scalability of the Solution	● The solution can be made scalable by using micro service architecture provided that each server responsible for certain functionality of theapplication

3.4.Problem Solution fit:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why



4.REQUIREMENT ANALYSIS

4.1.Functional requirement:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through mobile number Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Advanced Search Capabilities	sorting and filtering options
FR-4	Checking item availability	item availability in specific locations
FR-5	Shopping cart	My cart button Add-to-cart button Remove-from-cart button
FR-6	Super-fast checkout	Online transfer, credit card payment, paying with mobile wallets
FR-7	Checking the shipping status	Option to easily check the shipping status of items ordered in the store

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 dataset is obtained from the external sources must be safe and recommended for analysis
NFR-2	**Security**	Organizations must protect their most critical business assets—your data—against unauthorized or unwanted use. They must combine people, processes, and technology to protect data throughout its lifecycle. Use a unified platform that integrates data security information across your entire enterprise and that ensures scalability on environments of any size across public cloud, on-premises, and hybrid cloud deployment
NFR-3	**Reliability**	The analysis gives suggestions and steps that can be carried to whole company's attrition problem, as a long-time solution
NFR-4	**Performance**	The performance of the analysis must be solving the problem fully, so that it gives a permanent solution to the problem faced
NFR-5	**Availability**	The dataset is analysed and solution is given to the problem faced and the solution must be available for the full process
NFR-6	**Scalability**	Data is growing at an exponential rate. Keeping up with new data sources across environments creates complexity at an unprecedented scale

5.PROJECT DESIGN

5.1Data Flow Diagrams:

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

SOLUTION ARCHITECTURE:

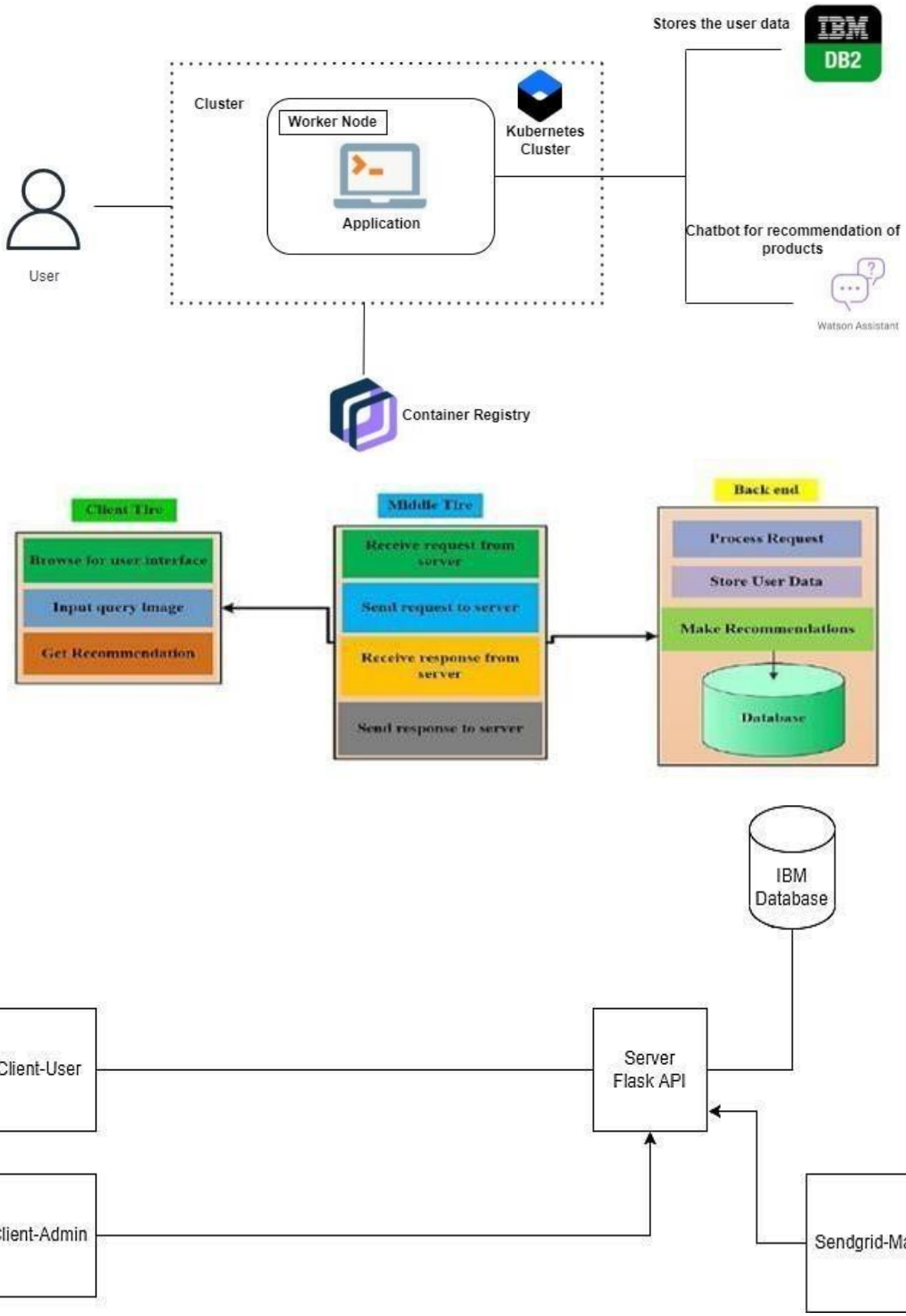
In recent years, the huge amount of information and users of the internet service, it is hard to know quickly and accurately what the user wants. This phenomenon leads to an extremely low utilization of information, also known as the information overload problem.

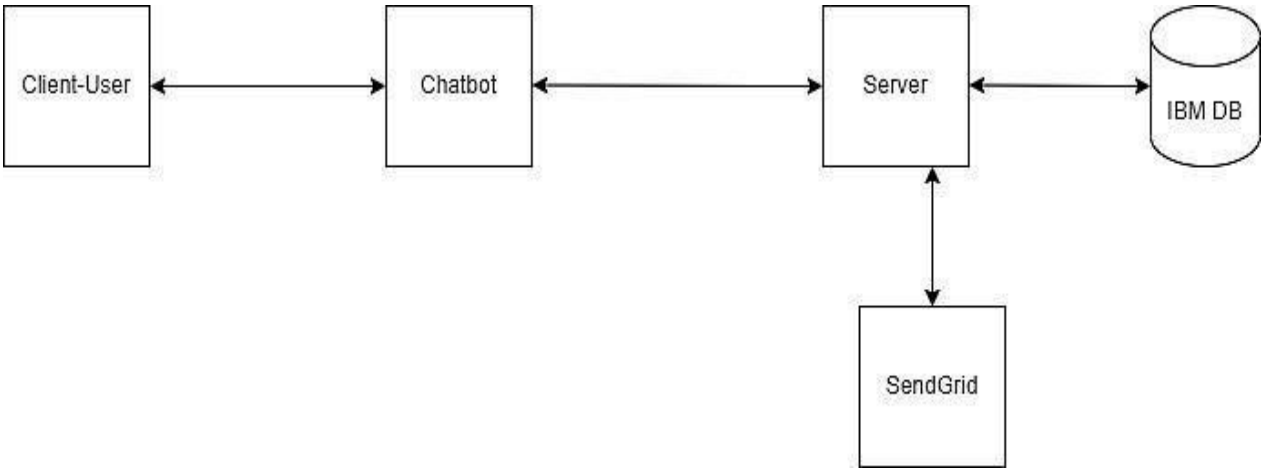
Traditionally, keywords are used to retrieve images, but such methods require a lot of annotations on the image data, which will lead to serious problems such as inconsistent, inaccurate, and incomplete descriptions, and a huge amount of work. To solve this problem, Content Based Information Retrieval (CBIR) has gradually become a research hotspot. CBIR retrieves picture objects based entirely on the content.

The content of an image needs to be represented by features that represent its uniqueness. Basically, any picture object can be represented by its specific shapes, colors, and textures. These visual characteristics of the image are used as input conditions for the query system, and a result the system will recommended nearest images and data set. This research designs and implements two-stage deep learning-based model that recommends a clothing fashion style. This model can use deep learning approach to extract various attributes from images with clothes to learn the user's clothing style and preferences.

These attributes are provided to the correspondence model to retrieve the contiguous related images for recommendation. Based on data-driven, this thesis uses convolutional neural network as a visual extractor of image objects. This experimental model shows and achieves better results than the ones of the previous schemes

EXAMPLE – SOLUTION ARCHITECTURE DIAGRAM


Stores the user data
IBM
DB2
Cluster
Worker Node
Application
Kubernetes
Cluster
User
Chatbot for recommendation of
products
Watson Assistant
Container Registry
Client Tire
Browse for user interface
Input query Image
Get Recommendation
Middle Tire
Receive request from
server
Send request to server
Receive response from
server
Send response to server
Back end
Process Request
Store User Data
Make Recommendations
Database
IBM
Database
Client-User
Server
Flask API
Client-Admin
Sendgrid-Mail



5.2Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirements	Sub Registration
FR-1	Registration	Registration can be done using mobile number or gmail and needed some user information
FR-2	Login	User only log in by user id and password,Which is given during registration
FR-3	Delivery confirmation	Confirmation via email and phone number
FR-4	Assistance	Bot is integrated with the application to make the usability simple

Proposed Solution Template:

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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	● Complex user interface ● Similar products appeared frequently ● Proper guidance is not available ● Customer less sticky to the application ● User need to navigate across multiple pages to choose right product ● Lack of interaction between application and user
2.	Idea / Solution description	By using Smart fashion recommender application: ● Collect feedback instantly ● Recommendation within a single page via chat-bot ● Effective recommendation of products ● Improve customer relationship, interactivity and services ● Reduce human error ● Proper guidance in accessing application.
3.	Novelty / Uniqueness	● Chat-bot asks and learns from user preference which recommends appropriate products to the user without making them to search through various filters. Reduces time in choosing right product thus increases sales.
4.	Social Impact / Customer Satisfaction	● Feedback from the user at the end of session or after placing order is one of the most important factor in deriving customer satisfaction and providing better services.
5.	Business Model (Revenue Model)	● The application can be developed at minimum cost with high performance and interactive user interface.
6.	Scalability of the Solution	● The solution can be made scalable by using micro service architecture provided that each server responsible for certain functionality of the application.

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

Sprint Schedule, and Estimation:

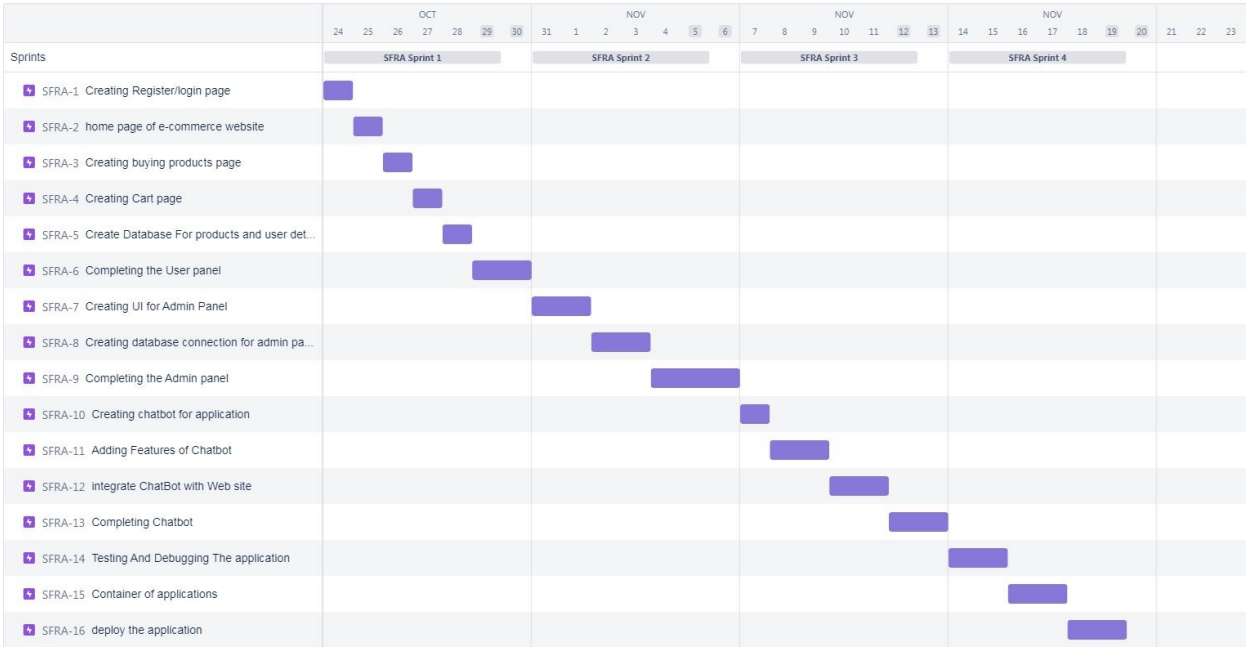
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Panel	USN-1	User login to the website with credentials and they will visit the products as they want to buy	20	High	SUJITH S MUTHU PANDI S VIJAYAK UMAR R
Sprint-2	Admin panel	USN-2	Here, Admin role is to keep track on the product database and stocks of the products that currently have. And also keep track on the products that user buys	20	High	ROHKITH ROSHAN S SUJITH S MUTHUPAN DI S
Sprint-3	Chat Bot	USN-3	Main feature of this application is chat bot. From this chat bots help user can navigate through the different screens. So, that purchasing of the user makes the simple	20	High	ROHKITH ROSHAN S SUJITH S VIJAYAKUM AR S
Sprint-4	final delivery	USN-4	Container of applications using docker kubernetes and deployment the application. Create the documentation and final submit the application.	20	High	ROHKITH ROSHAN S SUJITH S MUTHUPAN DI S VIJAYAKUM AR R

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

6.3. Reports from JIRA:

BURNDOWN CHART:



7.CODING & SOLUTIONING

7.1 Feature 1:

Deploying Chatbot In Website :

CONNECTING SCRIPT CODE:

```
<script>
  window.watsonAssistantChatOp
  tions = {
```

integrationID: "40072d56-fb6c-4240-9c90-f6e2bbcd9cd3", // The ID of this integration. region: "eu-gb", // The region your integration is hosted in.

serviceInstanceID: "365c5c99-5dd0-4a84-9ec2-24ba6029e35e", // The ID of your service instance. onLoad: function(instance) { instance.render(); }

};

setTimeout(function(){

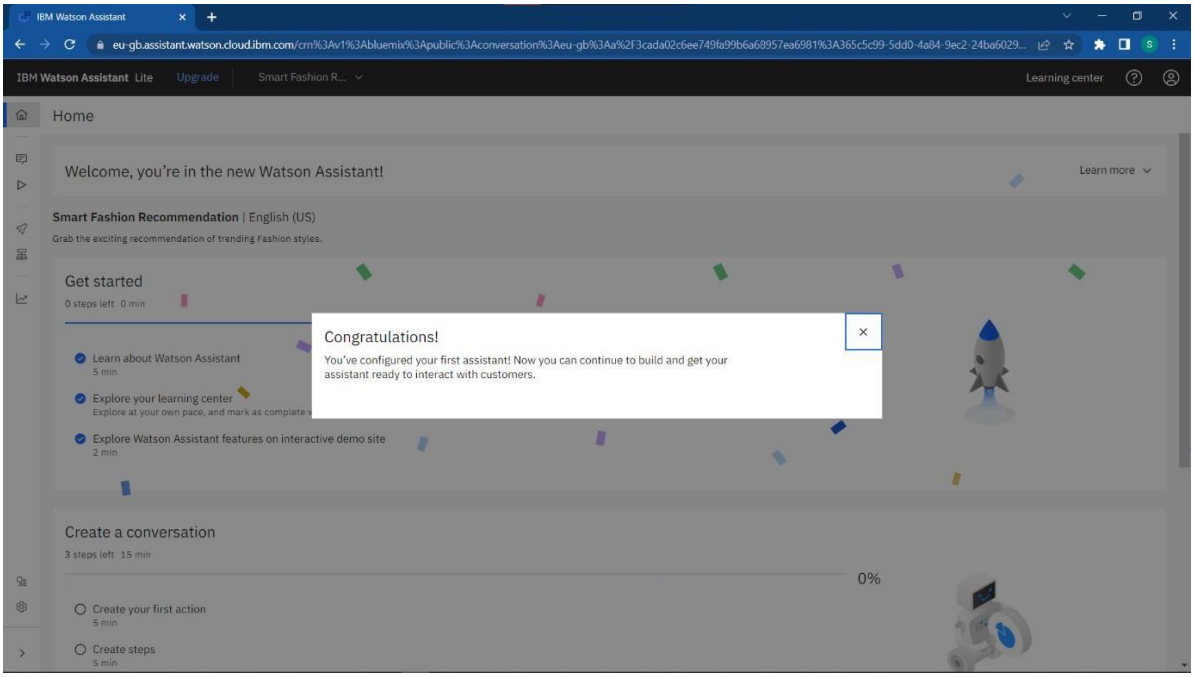
const t=document.createElement('script');

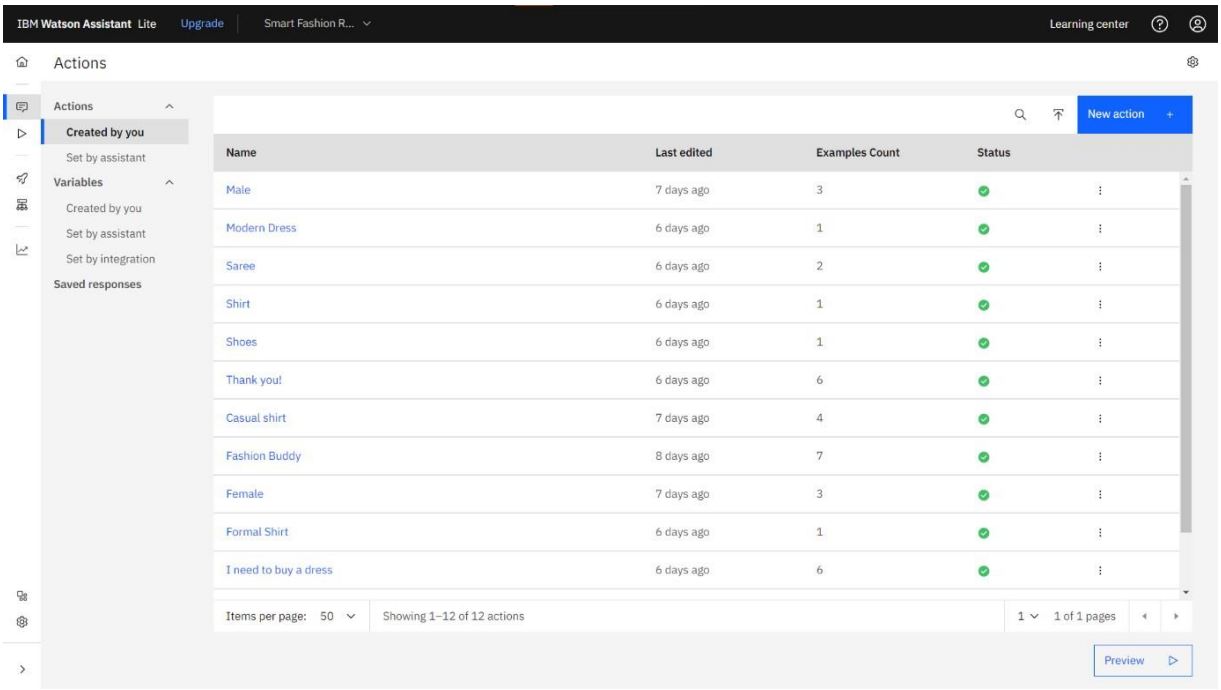
t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";

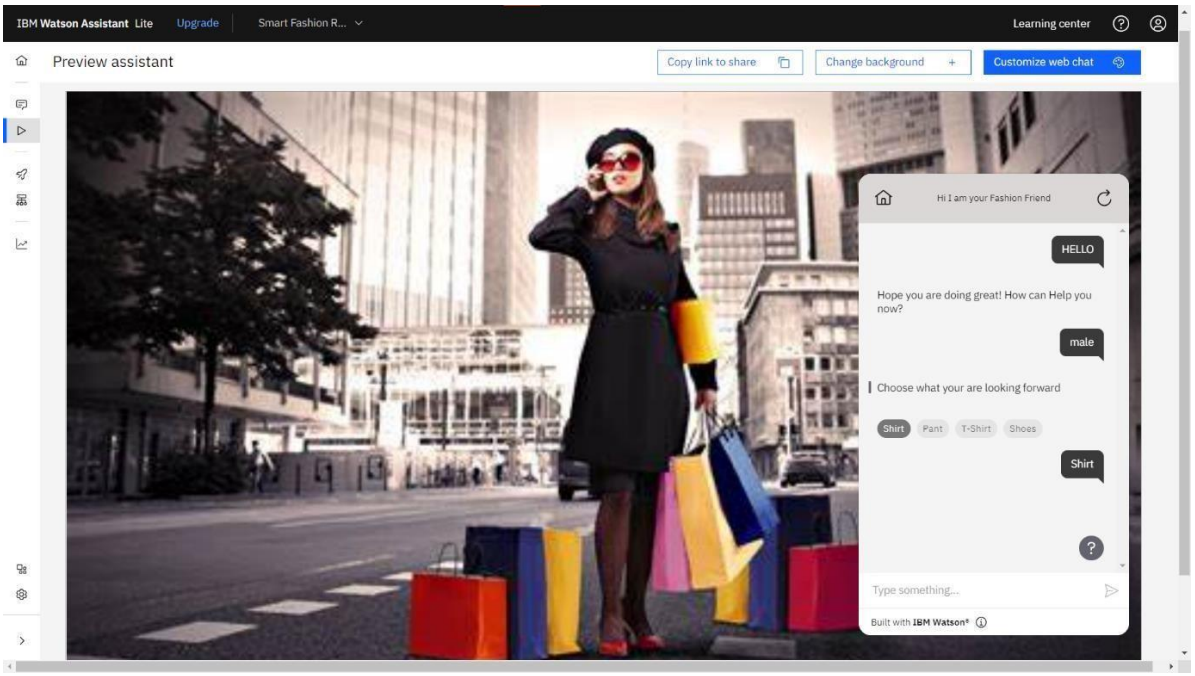
document.head.appendChild(t);

});

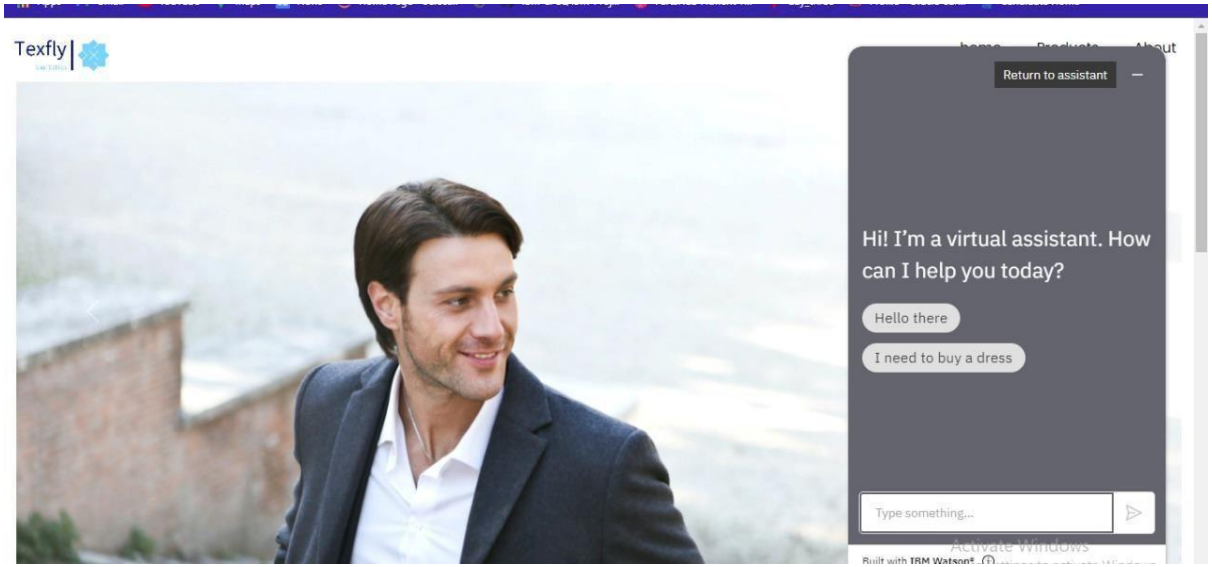
</script>//

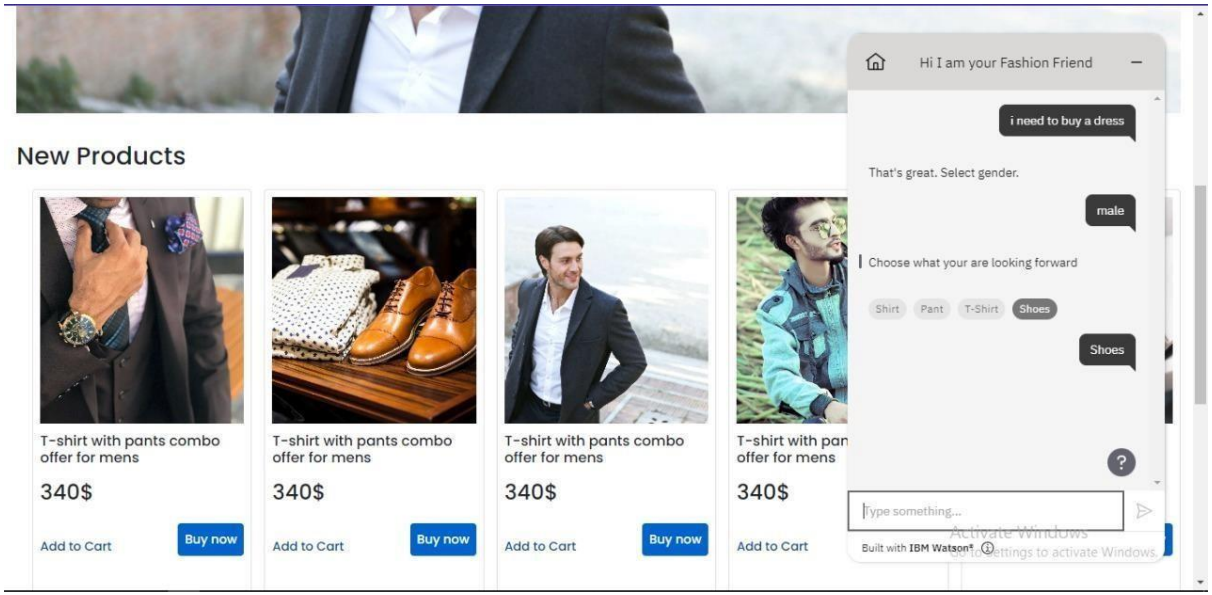



IBM Watson Assistant Lite
Upgrade
Smart Fashion R...
Learning center
Actions
Actions
Created by you
Set by assistant
Variables
Created by you
Set by assistant
Set by integration
Saved responses
New action
Name
Last edited
Examples Count
Status
Male 7 days ago 3
Modern Dress 6 days ago 1
Saree 6 days ago 2
Shirt 6 days ago 1
Shoes 6 days ago 1
Thank you! 6 days ago 6
Casual shirt 7 days ago 4
Fashion Buddy 8 days ago 7
Female 7 days ago 3
Formal Shirt 6 days ago 1
I need to buy a dress 6 days ago 6
Items per page: 50
Showing 1–12 of 12 actions
1 of 1 pages
Preview

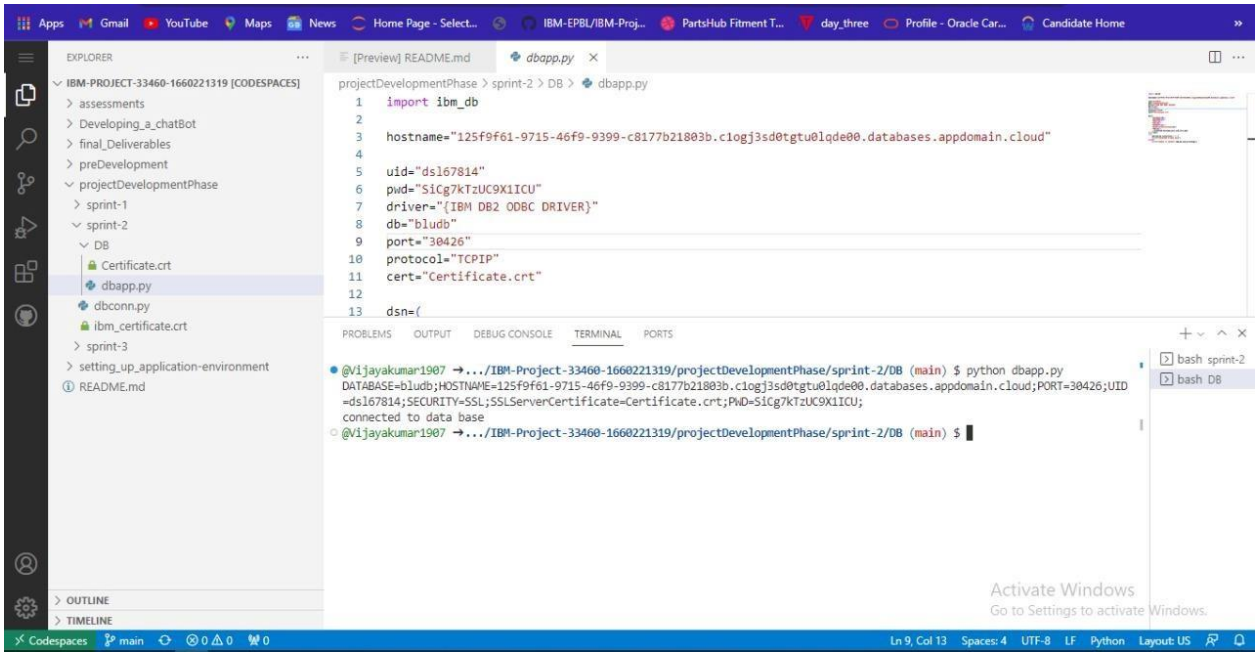

IBM Watson Assistant Lite
Upgrade
Smart Fashion R...
Learning center
Preview assistant
Copy link to share
Change background
Customize web chat
Hi I am your Fashion Friend
HELLO
Hope you are doing great! How can Help you now?
male
Choose what your are looking forward
Shirt
Pant
T-Shirt
Shoes
Shirt
Type something...
Built with IBM Watson®

INTEGRATE CHATBOT IN HTML:





7. 2.FEATURE DATABASE:



8.TESTING

8. 1.Test Cases

A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly. The purpose of a test case is to determine if different features within a system are performing as expected and to confirm that the system satisfies all

related standards, guidelines and customer requirements. The process of writing a test case can also help reveal errors or defects within the system.

Test cases are typically written by members of the quality assurance (QA) team or the testing team and can be used as step-by-step instructions for each system test. Testing begins once the development team has finished a system feature or set of features. A sequence or collection of test cases is called a test suite. A test case document includes test steps, test data, preconditions and the post conditions that verify requirements.

The benefits of an effective test case include:

- Guaranteed good test coverage.
- Reduced maintenance and software support costs.
- Reusable test cases.
- Confirmation that the software satisfies end-user requirements.

More satisfied customers will increase company profits. Overall, writing and using test cases will lead to business optimization. Clients are more satisfied, customer retention increases, the costs of customer service and fixing products decreases, and more reliable products are produced, which improves the company's reputation and brand image.

8.2.User Acceptance Testing

User acceptance testing (UAT), also called application testing or end-user testing, is a phase of software development in which the software is tested in the real world by its intended audience. User Acceptance Testing (UAT) is a type of testing

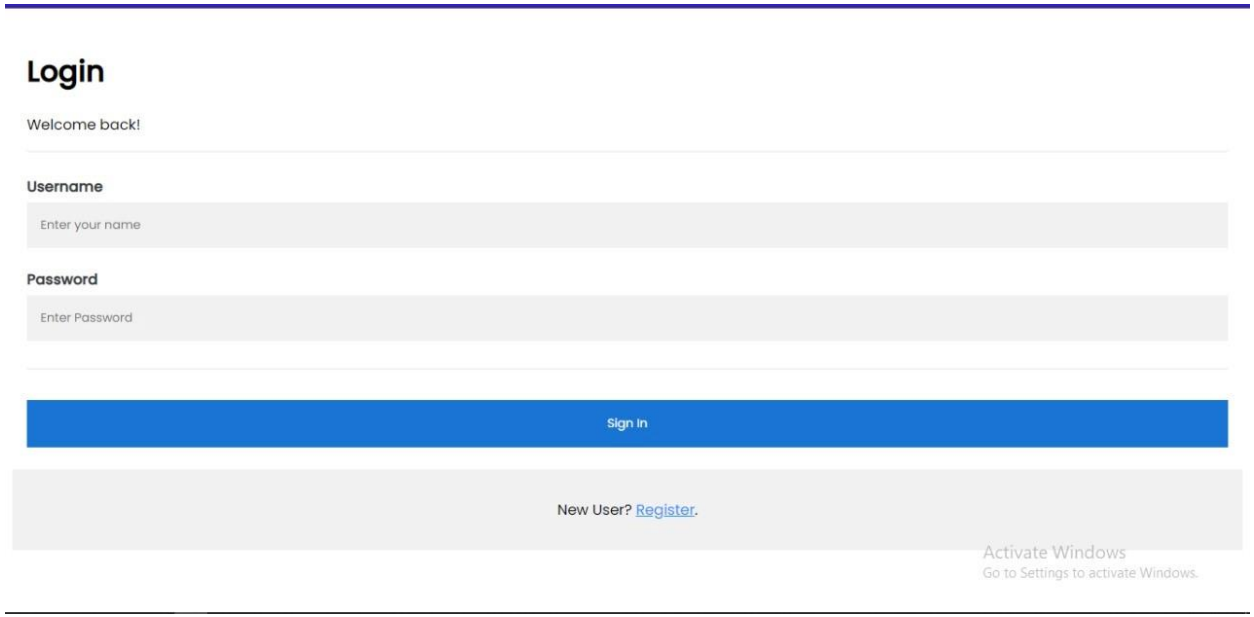
performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.

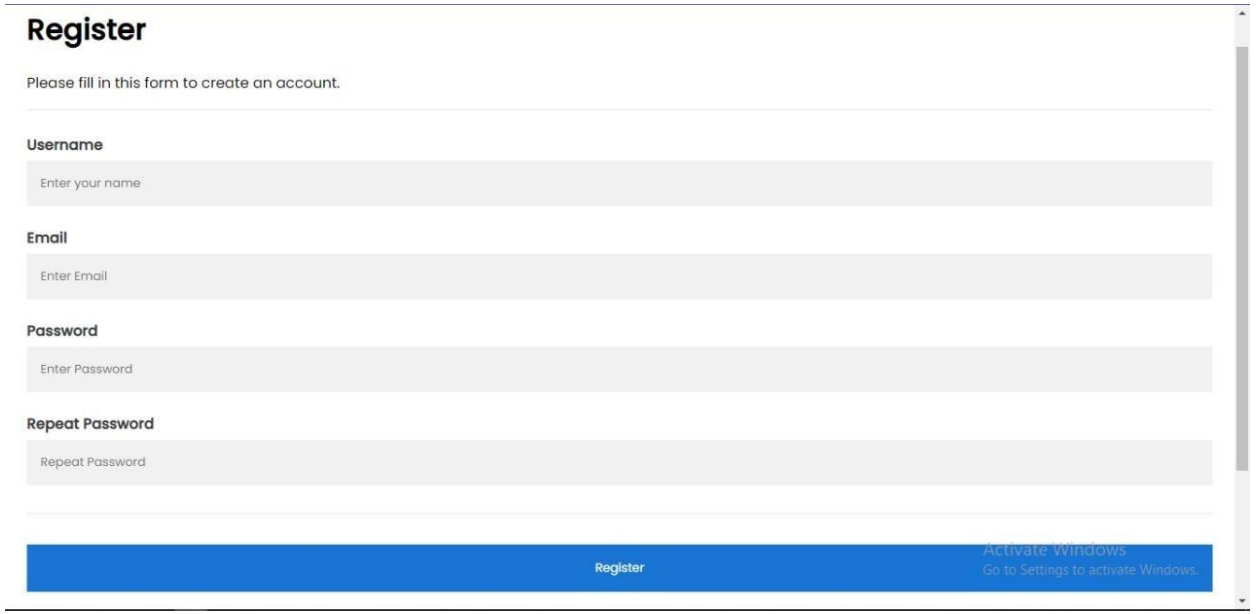
Need of User Acceptance Testing:

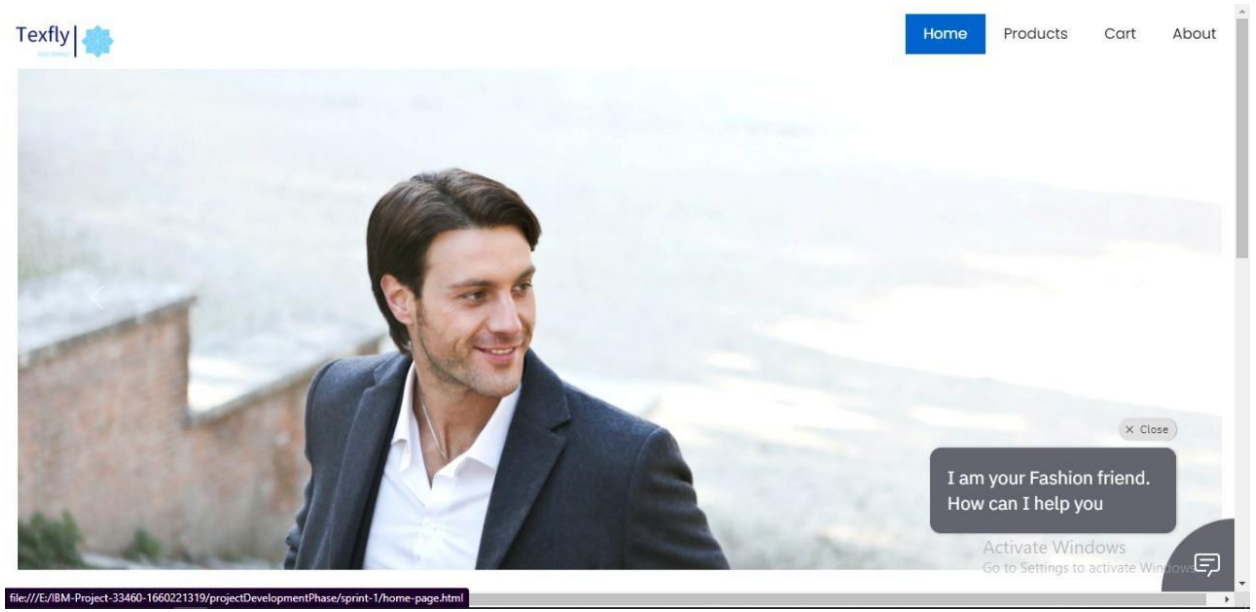
Need of User Acceptance Testing arises once software has undergone Unit, Integration and System testing because developers might have built software based on requirements document by their own understanding and further required changes during development may not be effectively communicated to them, so for testing whether the final product is accepted by client/end-user, user acceptance testing is needed.

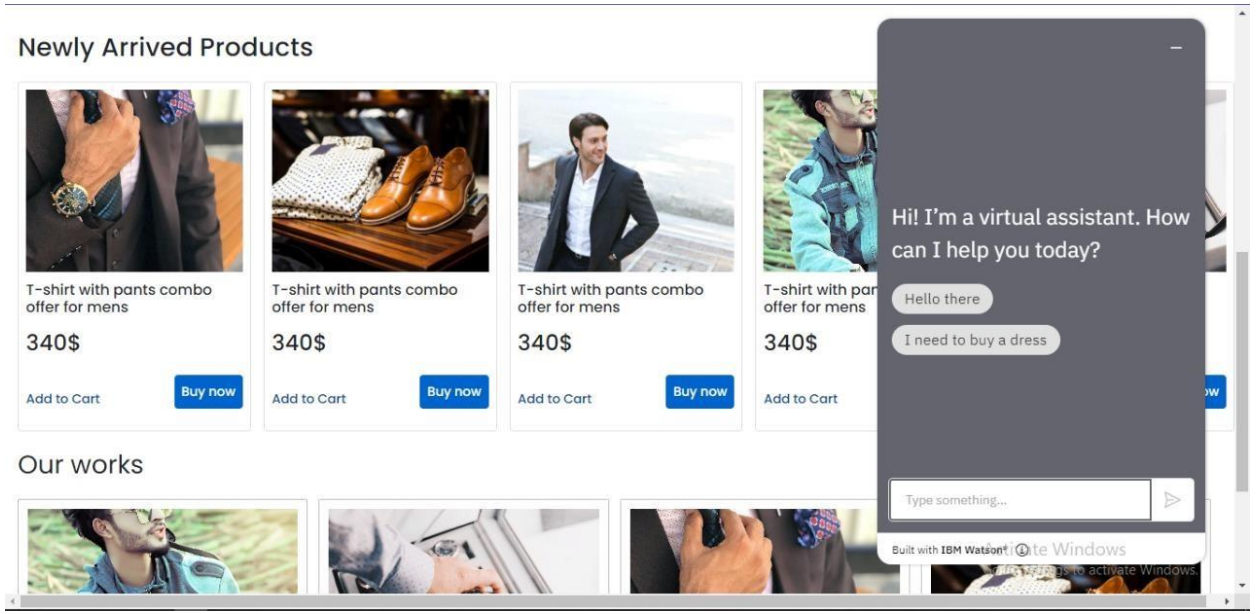
Developers code software based on requirements document which is their “own” understanding of the requirements and may not actually be what the client needs from the software. Requirements changes during the project may not be communicated effectively to the developers.

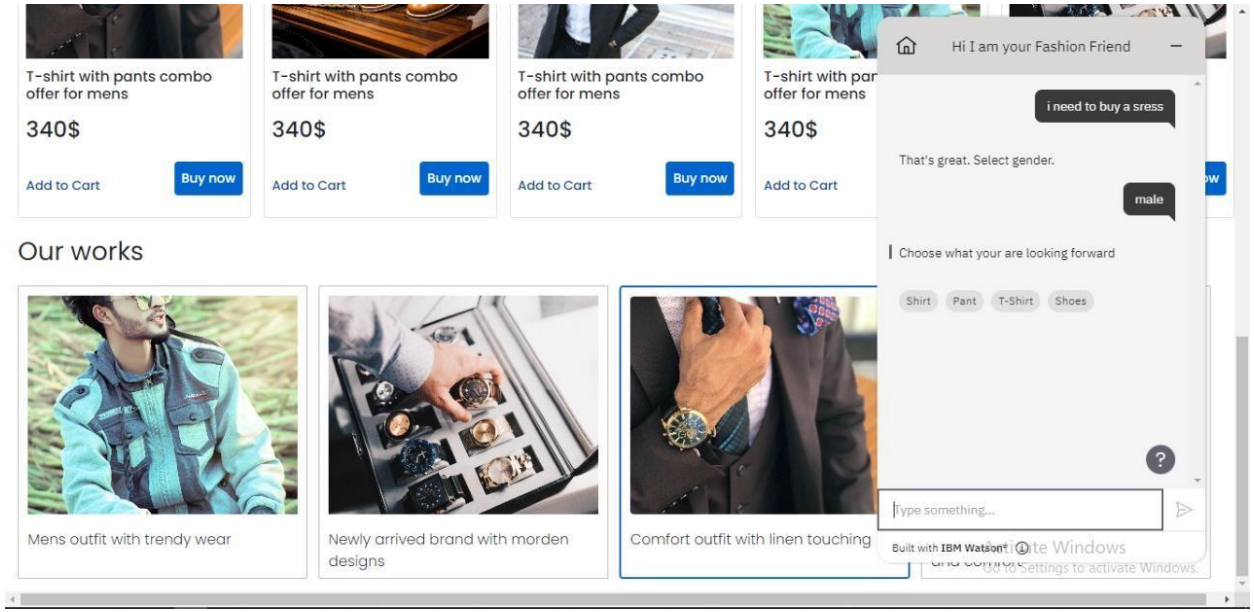
9.RESULT

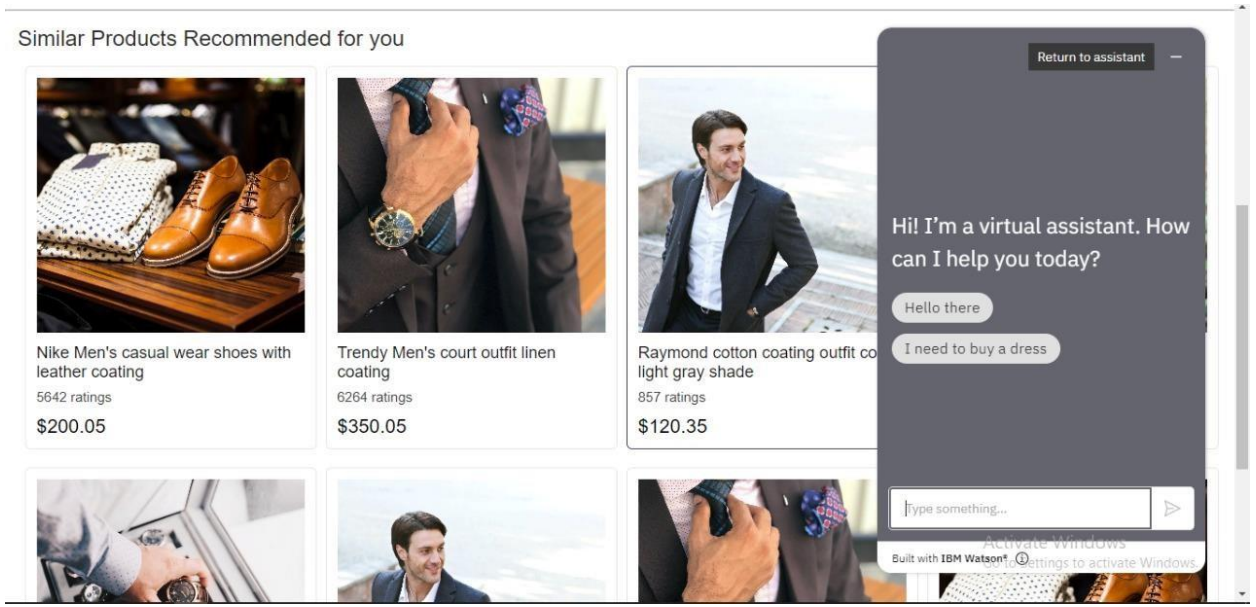

Login
Welcome back!
Username
Enter your name
Password
Enter Password
Sign In
New User? Register.
Activate Windows
Go to Settings to activate Windows.

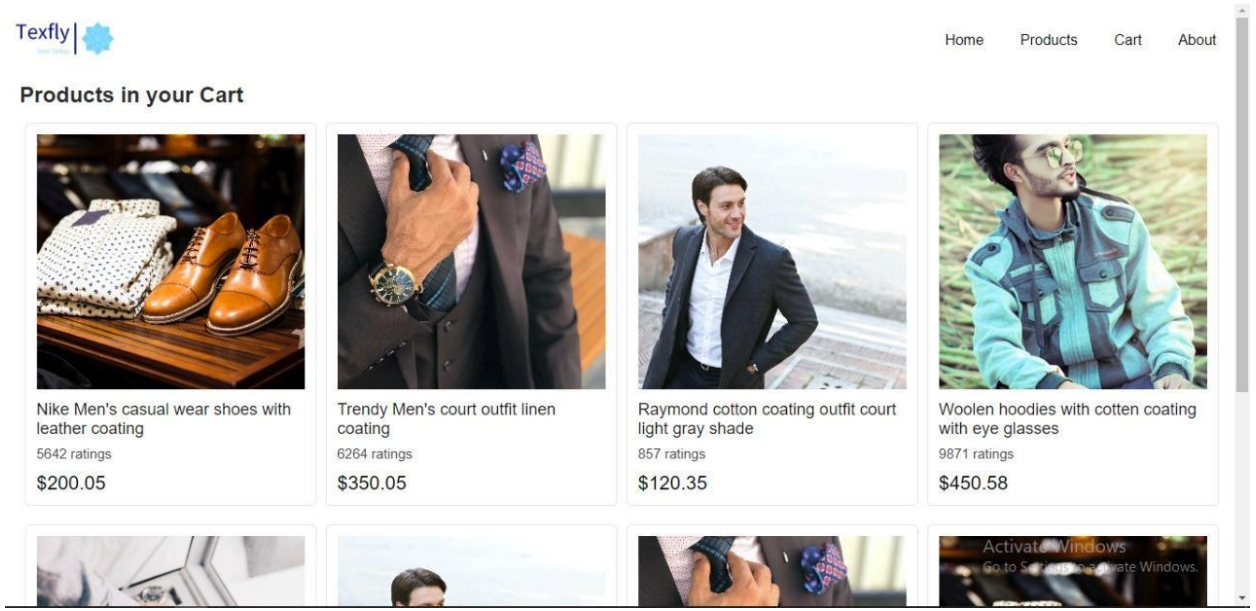

Register
Please fill in this form to create an account.
Username
Enter your name
Email
Enter Email
Password
Enter Password
Repeat Password
Repeat Password
Register
Activate Windows
Go to Settings to activate Windows.


Texfly
Home
Products
Cart
About
× Close
I am your Fashion friend.
How can I help you
Activate Windows
Go to Settings to activate Windows.
file:///E:/IBM-Project-33460-1660221319/projectDevelopmentPhase/sprint-1/home-page.html


Newly Arrived Products
T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with par
offer for mens
340$
Add to Cart
Our works
Hi! I'm a virtual assistant. How can I help you today?
Hello there
I need to buy a dress
Type something...
Built with IBM Watson


T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with pants combo offer for mens
340$
Add to Cart
Buy now
T-shirt with par
offer for mens
340$
Add to Cart
Hi I am your Fashion Friend
i need to buy a sress
That's great. Select gender.
male
Choose what your are looking forward
Shirt
Pant
T-Shirt
Shoes
Type something...
Built with IBM Watson
Our works
Mens outfit with trendy wear
Newly arrived brand with morden designs
Comfort outfit with linen touching


Similar Products Recommended for you
Nike Men's casual wear shoes with leather coating
5642 ratings
$200.05
Trendy Men's court outfit linen coating
6264 ratings
$350.05
Raymond cotton coating outfit co
light gray shade
857 ratings
$120.35
Return to assistant
Hi! I'm a virtual assistant. How can I help you today?
Hello there
I need to buy a dress
Type something...
Built with IBM Watson®
Activate Windows
Go to Settings to activate Windows.



10.ADVANTAGES AND DISADVANTAGES

10.1. ADVANTAGES:

- Speed up the process of decision and purchase based on the previous statistics.
- A recommendation engine can bring traffic to were sites. It accomplishes this with customized email messages and target blasts.
- Easy recommendations make fewer searches and sometimes end up in good deals

- User reviews will give accurate information, this is also an advantage if we purchase online as we can see other reviews too, most of the time honest.

10.2. DISADVANTAGES:

- As it is CBF domain-dependent, rigorous domain knowledge is required to make precise recommendations.
- The model only recommends products based on an existing database of previous users' interest, which restricts its expansion.
- This method suffers limited content analysis issues, meaning users are restricted to the items already recommended.

11.CONCLUSION

- This article sheds some light on different applications related to these systems, tracked the research progress through the years, and illustrated the field's rapid growth. Although scientists have achieved significant milestones, many unsolved matters remain.
- Recent advancements in cloud computing have helped ease the fashion industry' s transition from customer stores into modern online shops equipped with high-tech features such as virtual try-on and fashion synthesis systems.
- Another issue is the systems' performance compared to human abilities; another important factor is the applicability of methods regarding computational effort and energy efficiency.

12.FUTURE SCOPE:

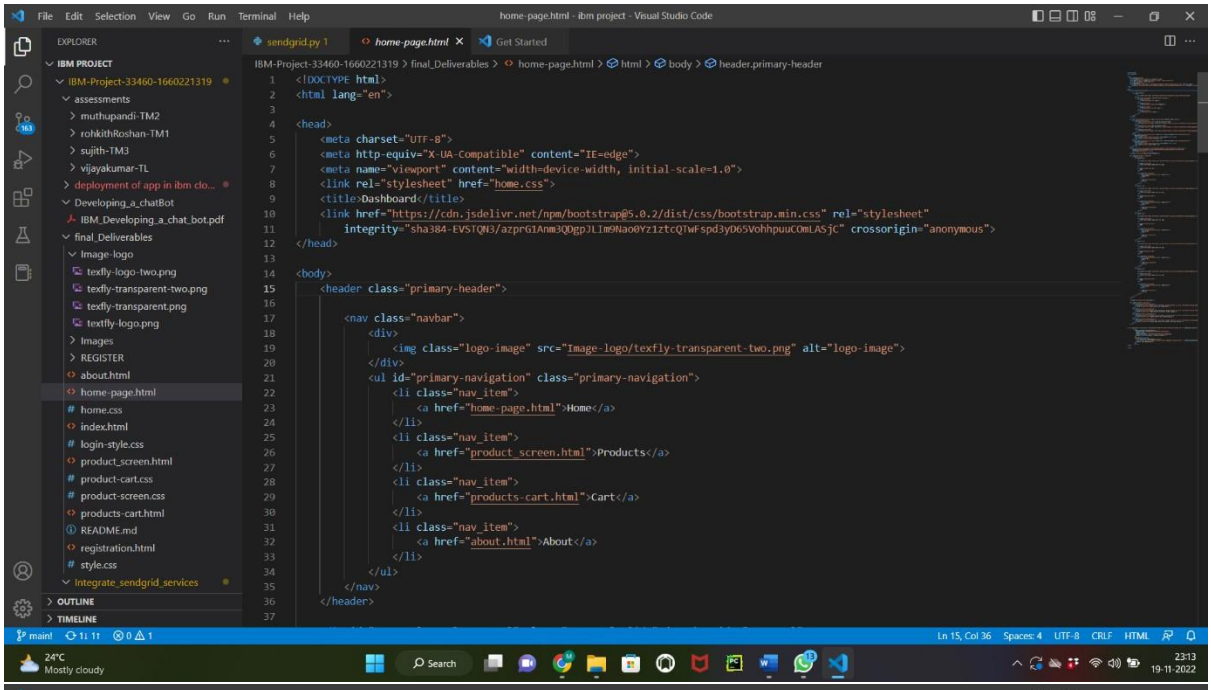
- Besides, while the majority of research in fashion recommender systems is mainly based on similarity based retrieval techniques, there is a need for more studies in the development of new functions such as designing clothes, which are highly demanded in future fashion recommender systems.

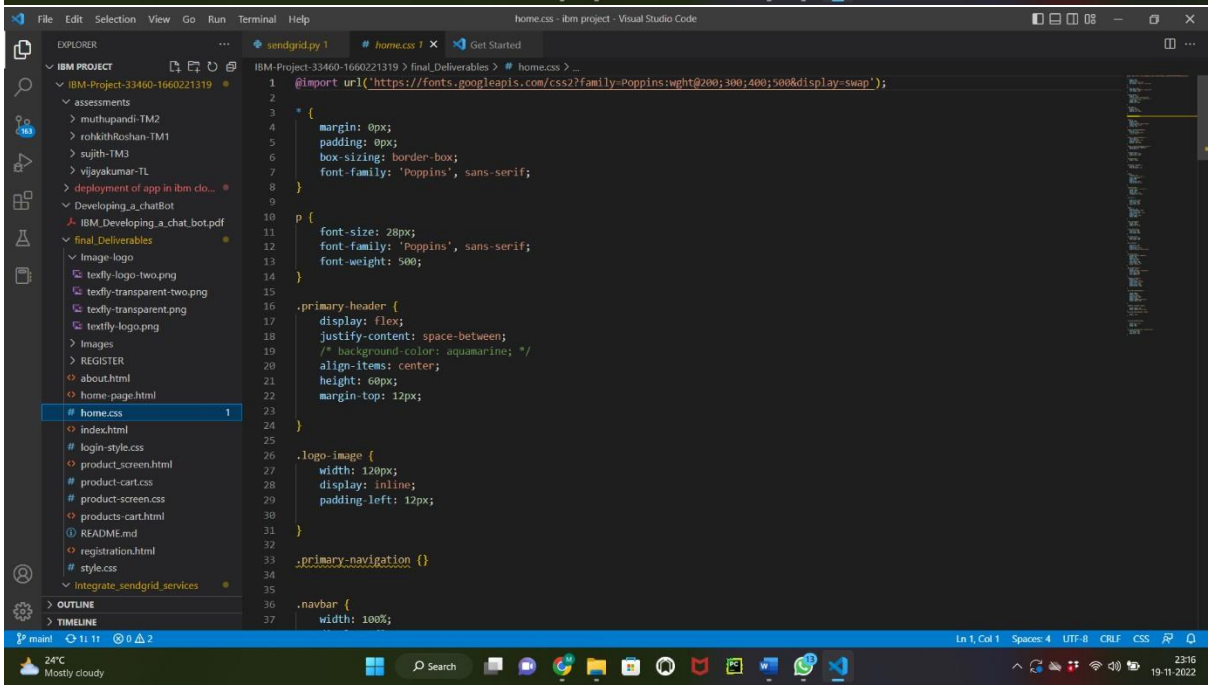
- Furthermore, most of the current fashion datasets do not contain outfit compatibility annotations, or they are limited in terms of size and the type of annotations they provide. Consequently, most researchers built their dataset, which is a labor- costing process, and most of them are not accessible publicly for further research.

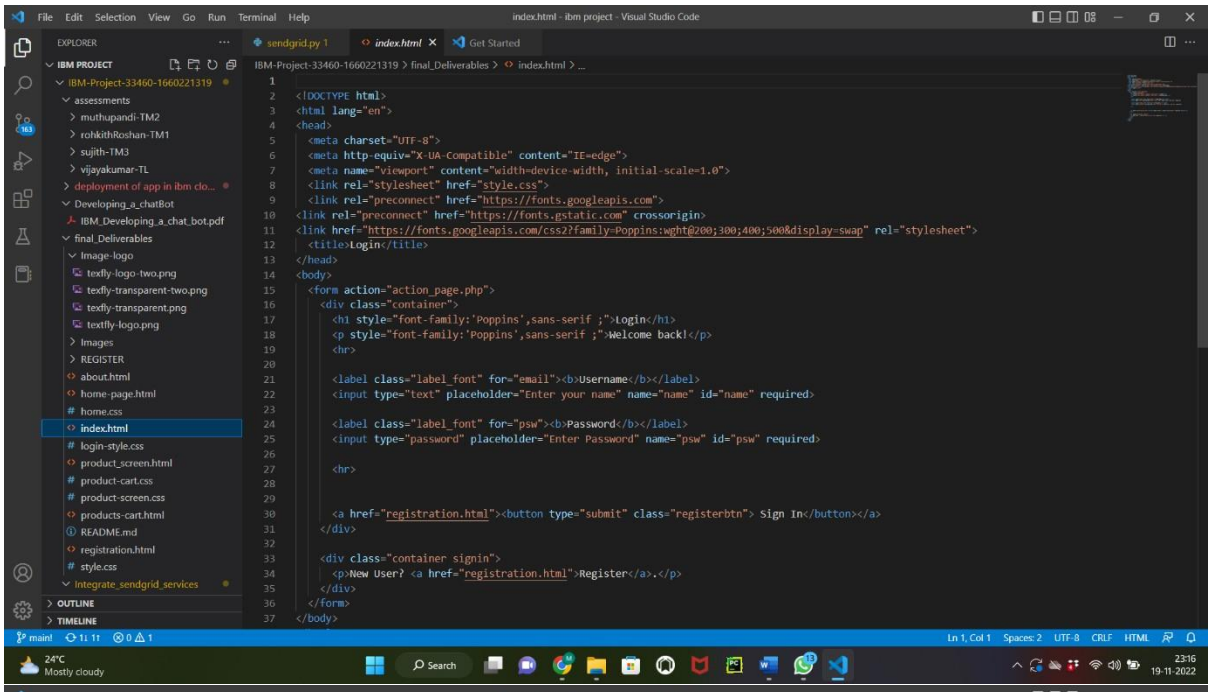
- So, the other future direction for subsequent studies may be focusing on developing automatic annotation methods, constructing large-scale rich annotated data sets for particular task definitions in fashion recommender systems.

13.APPENDIX:

SOURCE CODE:



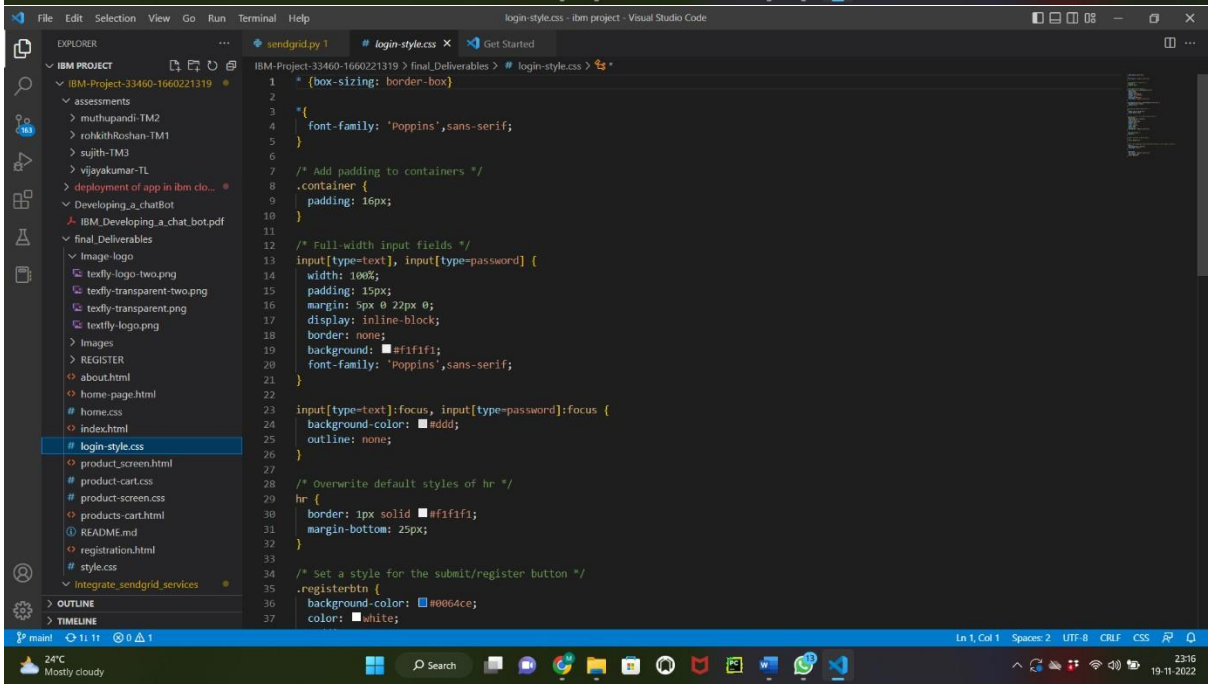


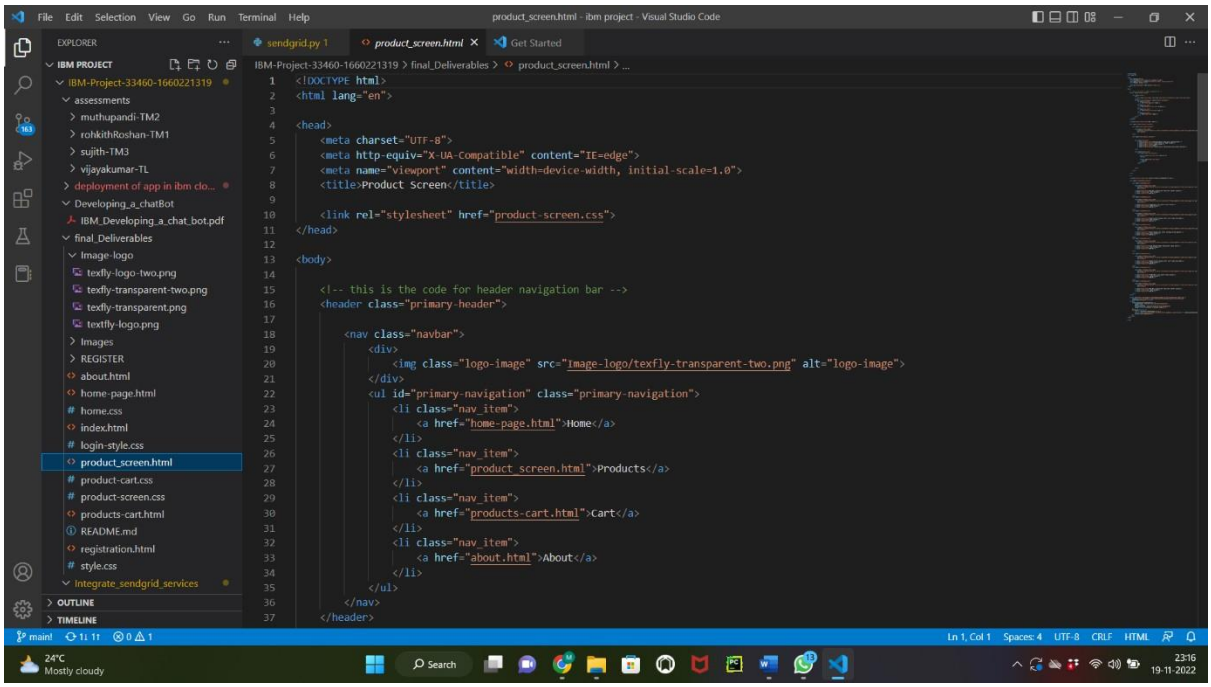

File Edit Selection View Go Run Terminal Help
index.html - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
index.html
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > index.html > ...
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="style.css">
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500&display=swap" rel="stylesheet">
<title>Login</title>
</head>
<body>
<form action="action_page.php">

<h1 style="font-family:'Poppins',sans-serif ;">Login</h1>
Welcome back!
<hr>
<label class="label_font" for="email">Username</label>
<input type="text" placeholder="Enter your name" name="name" id="name" required>
<label class="label_font" for="psw">Password</label>
<input type="password" placeholder="Enter Password" name="psw" id="psw" required>
<hr>
<button type="submit" class="registerbtn"> Sign In</button>

New User? Register.

</form>
</body>
main!
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
HTML
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
login-style.css - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
login-style.css
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > login-style.css > *
* {box-sizing: border-box}
*{
font-family: 'Poppins',sans-serif;
}
/* Add padding to containers */
.container {
padding: 16px;
}
/* Full-width input fields */
input[type=text], input[type=password] {
width: 100%;
padding: 15px;
margin: 5px 0 22px 0;
display: inline-block;
border: none;
background: #f1f1f1;
font-family: 'Poppins',sans-serif;
}
input[type=text]:focus, input[type=password]:focus {
background-color: #ddd;
outline: none;
}
/* Overwrite default styles of hr */
hr {
border: 1px solid #f1f1f1;
margin-bottom: 25px;
}
/* Set a style for the submit/register button */
.registerbtn {
background-color: #0064ce;
color: white;
main!
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
CSS
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
product_screen.html - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
product_screen.html
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > product_screen.html > ...
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Product Screen</title>
<link rel="stylesheet" href="product-screen.css">
</head>
<body>
<!-- this is the code for header navigation bar -->
<header class="primary-header">
<nav class="navbar">

<img class="logo-image" src="image-logo/texfly-transparent-two.png" alt="logo-image">

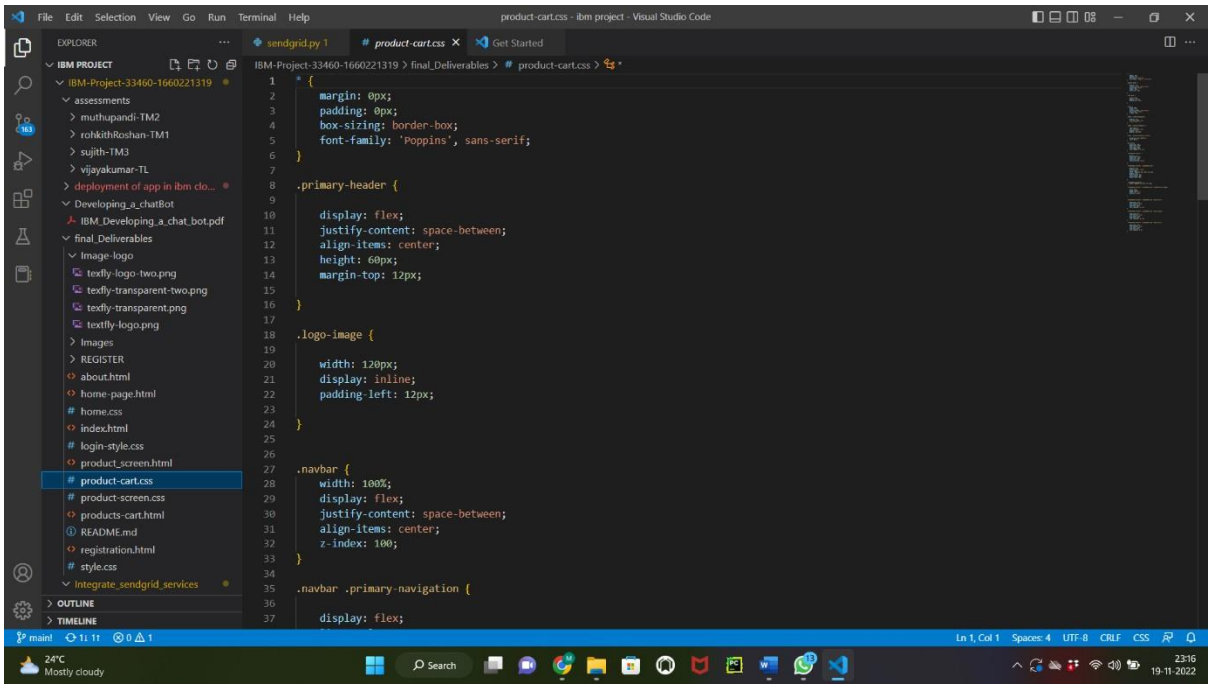
<ul id="primary-navigation" class="primary-navigation">
<li class="nav_item">
Home

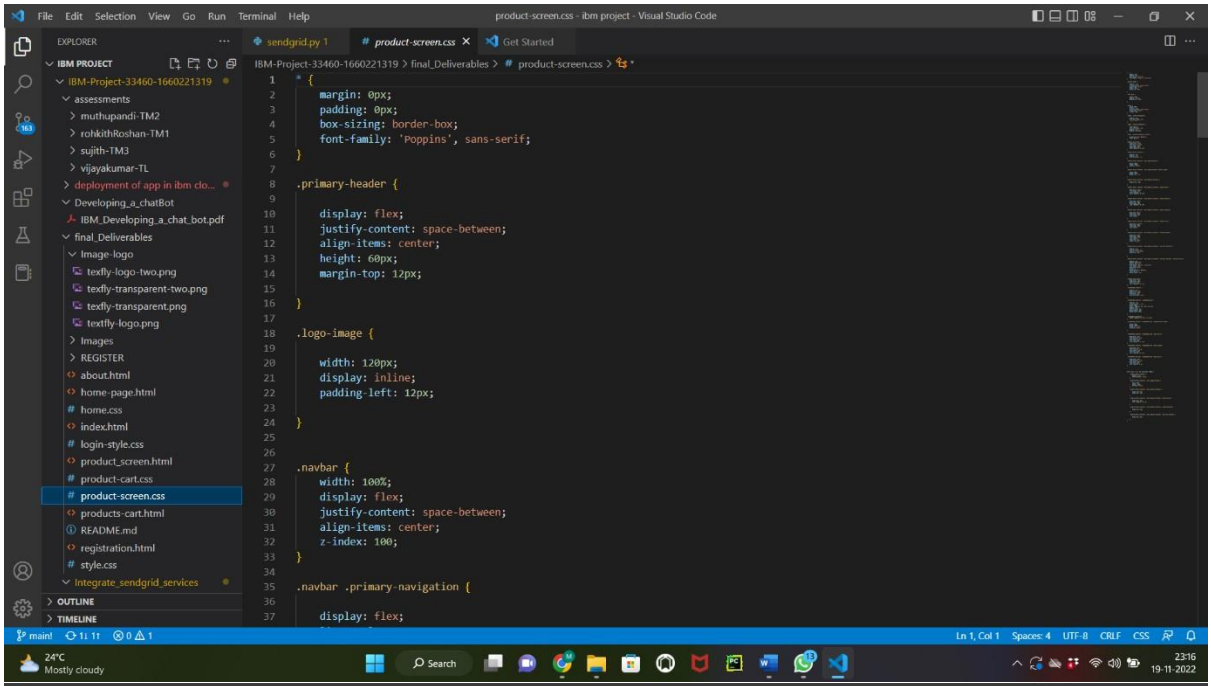
<li class="nav_item">
Products

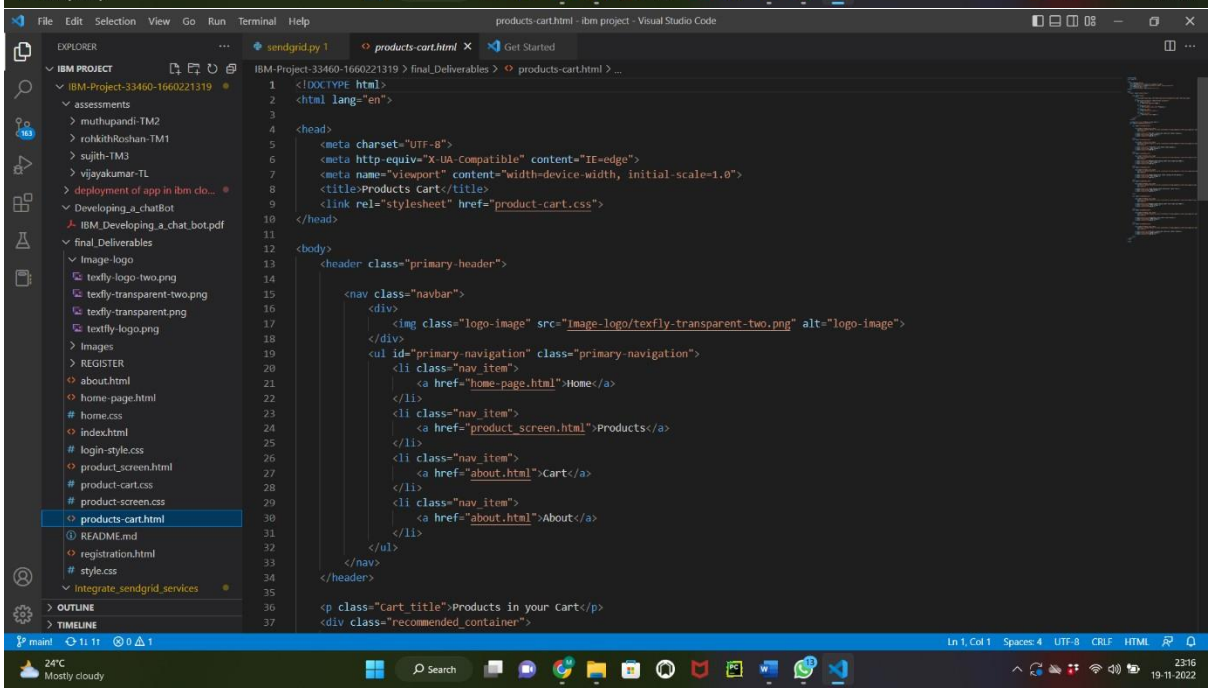
<li class="nav_item">
Cart

<li class="nav_item">
About

</nav>
</header>
main!
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
HTML
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
product-cart.css - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
product-cart.css
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > product-cart.css > *
* {
margin: 0px;
padding: 0px;
box-sizing: border-box;
font-family: 'Poppins', sans-serif;
}
.primary-header {
display: flex;
justify-content: space-between;
align-items: center;
height: 60px;
margin-top: 12px;
}
.logo-image {
width: 120px;
display: inline;
padding-left: 12px;
}
.navbar {
width: 100%;
display: flex;
justify-content: space-between;
align-items: center;
z-index: 100;
}
.navbar .primary-navigation {
display: flex;
main!
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
CSS
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
product-screen.css - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
product-screen.css
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > product-screen.css
* {
margin: 0px;
padding: 0px;
box-sizing: border-box;
font-family: 'Poppins', sans-serif;
}
.primary-header {
display: flex;
justify-content: space-between;
align-items: center;
height: 60px;
margin-top: 12px;
}
.logo-image {
width: 120px;
display: inline;
padding-left: 12px;
}
.navbar {
width: 100%;
display: flex;
justify-content: space-between;
align-items: center;
z-index: 100;
}
.navbar .primary-navigation {
display: flex;
main!
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
CSS
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
products-cart.html - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
products-cart.html
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > products-cart.html > ...
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Products Cart</title>
<link rel="stylesheet" href="product-cart.css">
</head>
<body>
<header class="primary-header">
<nav class="navbar">

<img class="logo-image" src="Image-logo/texfly-transparent-two.png" alt="logo-image">

<ul id="primary-navigation" class="primary-navigation">
<li class="nav_item">
Home

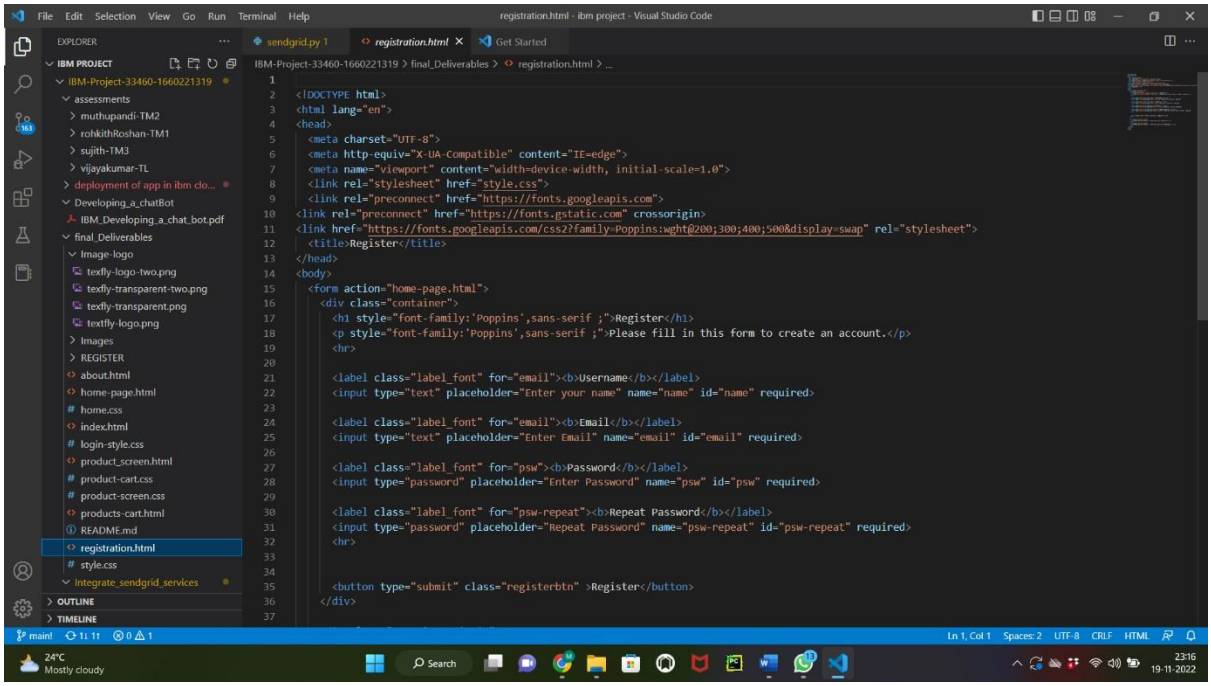
<li class="nav_item">
Products

<li class="nav_item">
Cart

<li class="nav_item">
About

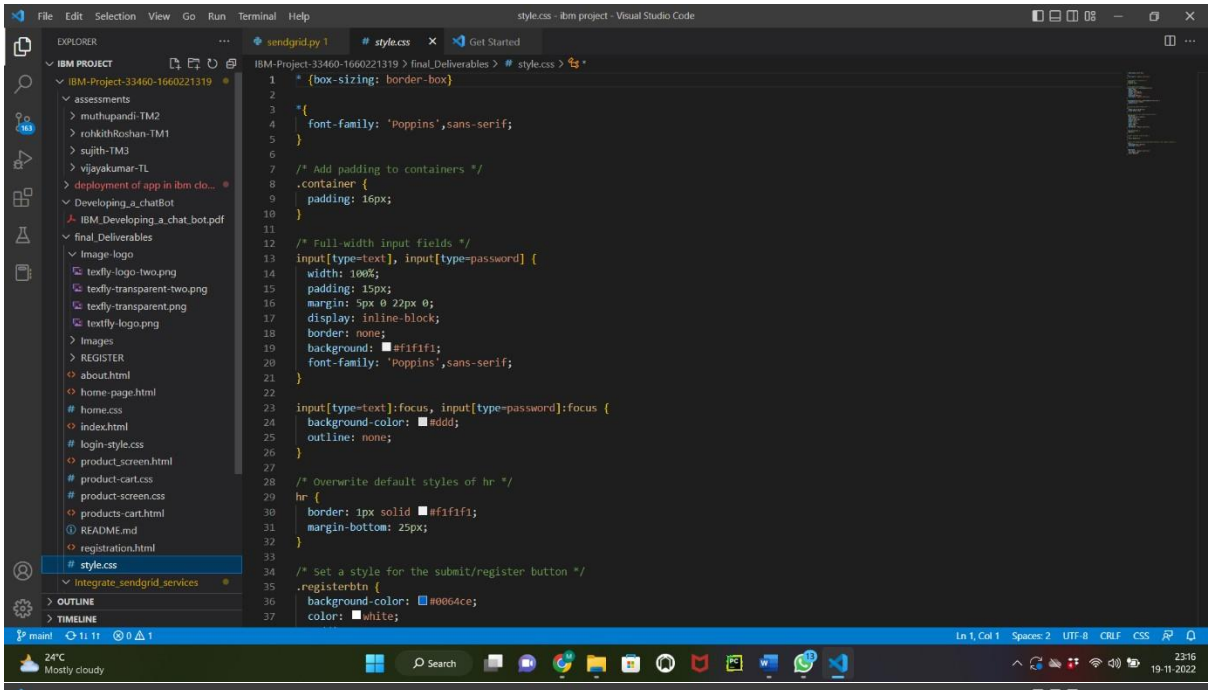
</nav>
</header>
Products in your Cart

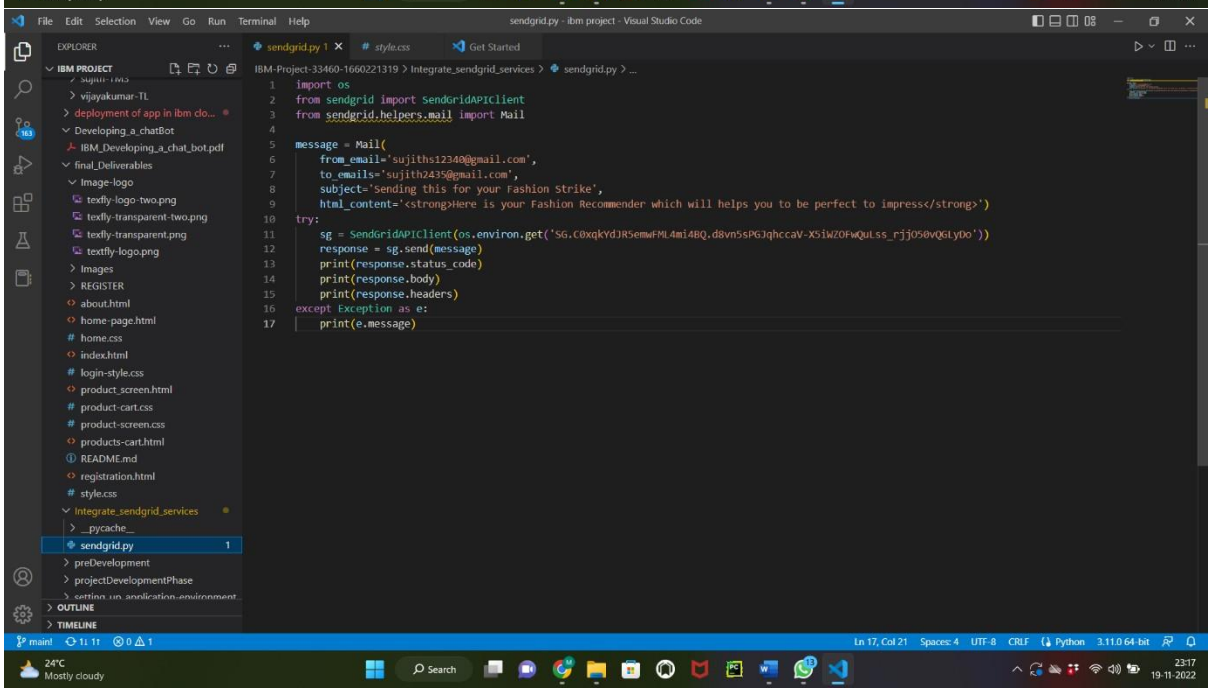
main!
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
HTML
24°C
Mostly cloudy
Search
23:16
19-11-2022

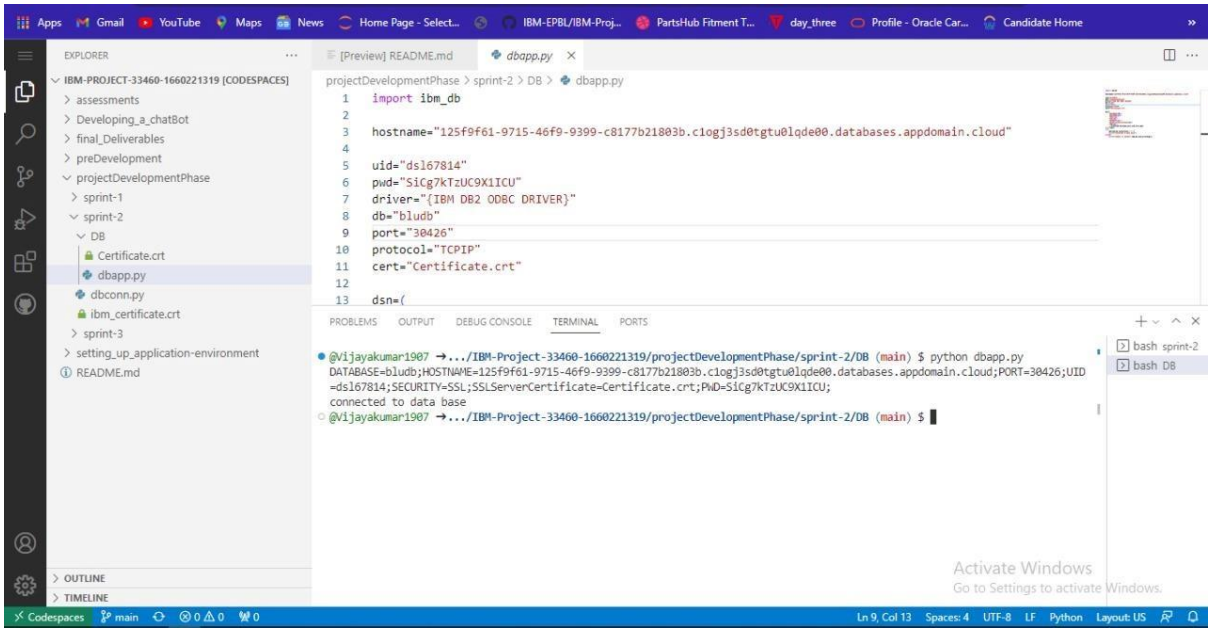

File Edit Selection View Go Run Terminal Help
registration.html - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
registration.html
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > registration.html > ...
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="style.css">
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500&display=swap" rel="stylesheet">
<title>Register</title>
</head>
<body>
<form action="home-page.html">

<h1 style="font-family:'Poppins',sans-serif ;">Register</h1>
Please fill in this form to create an account.
<hr>
<label class="label_font" for="email">Username</label>
<input type="text" placeholder="Enter your name" name="name" id="name" required>
<label class="label_font" for="email">Email</label>
<input type="text" placeholder="Enter Email" name="email" id="email" required>
<label class="label_font" for="psw">Password</label>
<input type="password" placeholder="Enter Password" name="psw" id="psw" required>
<label class="label_font" for="psw-repeat">Repeat Password</label>
<input type="password" placeholder="Repeat Password" name="psw-repeat" id="psw-repeat" required>
<hr>
<button type="submit" class="registerbtn" >Register</button>

main!
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
HTML
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
style.css - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
IBM-Project-33460-1660221319
assessments
muthupandi-TM2
rohkithRoshan-TM1
sujith-TM3
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
OUTLINE
TIMELINE
sendgrid.py 1
style.css
Get Started
IBM-Project-33460-1660221319 > final_Deliverables > style.css > *
* {box-sizing: border-box}
*{
font-family: 'Poppins',sans-serif;
}
/* Add padding to containers */
.container {
padding: 16px;
}
/* Full-width input fields */
input[type=text], input[type=password] {
width: 100%;
padding: 15px;
margin: 5px 0 22px 0;
display: inline-block;
border: none;
background: #f1f1f1;
font-family: 'Poppins',sans-serif;
}
input[type=text]:focus, input[type=password]:focus {
background-color: #ddd;
outline: none;
}
/* Overwrite default styles of hr */
hr {
border: 1px solid #f1f1f1;
margin-bottom: 25px;
}
/* Set a style for the submit/register button */
.registerbtn {
background-color: #0064ce;
color: white;
main!
Ln 1, Col 1
Spaces: 2
UTF-8
CRLF
CSS
24°C
Mostly cloudy
Search
23:16
19-11-2022


File Edit Selection View Go Run Terminal Help
sendgrid.py - ibm project - Visual Studio Code
EXPLORER
IBM PROJECT
vijayakumar-TL
deployment of app in ibm clo...
Developing_a_chatBot
IBM_Developing_a_chat_bot.pdf
final_Deliverables
Image-logo
texfly-logo-two.png
texfly-transparent-two.png
texfly-transparent.png
textfly-logo.png
Images
REGISTER
about.html
home-page.html
home.css
index.html
login-style.css
product_screen.html
product-cart.css
product-screen.css
products-cart.html
README.md
registration.html
style.css
Integrate_sendgrid_services
__pycache__
sendgrid.py 1
preDevelopment
projectDevelopmentPhase
OUTLINE
TIMELINE
sendgrid.py 1
style.css
Get Started
IBM-Project-33460-1660221319 > Integrate_sendgrid_services > sendgrid.py > ...
import os
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail
message = Mail(
from_email='sujiths12340@gmail.com',
to_emails='sujith2435@gmail.com',
subject='Sending this for your Fashion Strike',
html_content='Here is your Fashion Recommender which will helps you to be perfect to impress')
try:
sg = SendGridAPIClient(os.environ.get('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'))
response = sg.send(message)
print(response.status_code)
print(response.body)
print(response.headers)
except Exception as e:
print(e.message)
main!
Ln 17, Col 21
Spaces: 4
UTF-8
CRLF
Python
3.11.0 64-bit
24°C
Mostly cloudy
Search
23:17
19-11-2022



GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-33460-1660221319.git