IBM – NALAIYA THIRAN PROJECT

SMART FASHION RECOMMENDER APPLICATION

INDUSTRY MENTOR : KRISHNA CHAITANYA

FACULTYMENTOR : MADHAN SUBRAMANIYAM

TEAM ID : PNT2022TMID47029

TEAM LEADER : AAKASH.N

TEAM MEMBER : KAVYA.R

TEAM MEMBER : MEENACHI MADHUMITHA.V

TEAM MEMBER : SWATHIPRIYA.K

TEAM MEMBER : CHELLAM.P

ABSTRACT

Fashion is perceived as a meaningful way of self-expressing that people use for differentpurposes Fashionable products are highly demanded, and consequently, fashion is perceived as a desirable and profitable industry. Although this massive demand for fashion products provides an excellent

opportunity for companies to invest in fashion- related sectors, it also faces different challenges in answereing their customer needs. Fashion recommender systems have been introduced to address these needs.

In recent years, the huge amount of information and users of the internet service, it is hard to know quicklyand accurately what the user wants. This phenomenon leadsto anextremely low utilization of information, also known as the information overload problem. Traditionally, keywords are used to retrieveimages, but such methods requirea 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.

The fashion industry is rapidly expanding and playing a critical role in driving global economies. Due to this ever-growing industry,application of computerscience is risingrapidly to solve different problems in this industry. Many e-commerce sites around theworld allowtheir customers to purchase clothingitems over the internet predominantlyusing recommender systems for shoppers based on the customer's purchase history, similar buyingpatterns of other shoppers, items in the wish

The rapid progress of computer vision, cloud computing and artificial intelligence combinedwith the currentgrowing urge for online shoppingsystems opened anexcellent opportunity for the fashion industry. As a result, many studies worldwide arededicated to modern fashion related

TABLE OF CONTENTS

1.INTRODUCTION

1.1 PROJECT OVERVIEW

During the last few years,online shopping has been growing.In 2013, the total turnoverfor e-commerce in Europe expanded by 17% in contrast to the 12 months before and huge organizations can have hundreds and hundreds of products or even more from which we can select on websites. Both the customer and the business enterprise desire the client to easilydiscover applicable productsor items both throughout the search and when they are searching, and this is where recommender systems come into the picture.The greater part (62%) of US buyers with Web access presently shop online, to some degree, at least a month, and 1% say they do not buy from the internet, as indicated by a current report by Walker Sands. Of all the clients looking for items on the web, 63%of them buy garments (Burke, 2002), these being, quite possibly, the most purchased items.

The information uncover that women are more likely to buy online, with 71% of ladies doing this, contrasted with 52% of men (Reshma& Patil, 2012).Studies on clothingarein a growing development in general as a result of the tremendous market related to dress. In China, the serviceable market crushed 20 billion US dollars in 2016 (Jannah& Friedrich, 2013). Such huge market prospectsimpressively energize clothingapplicable exploration. Beingone of the new studiesin progress both at the national andinternational level, recommender systems have proved to be a large solution for e-commerce (Beetle al., 2013), but the internet options yet pose many strong and weakpoints.

Some of theseweaknesses consist of a lack of accuracyregarding information, whichis the more important weaknessamong others (Massa&

Bhattacharjee,2004). To decrease some of these weaknesses, collaborative filtering methods have been combined with content-based methodsto come up with hybrid recommender systems(Massa & Bhattacharjee, 2004). Moreover, explicit,and implicit remarkshave also been mixed toenhance the accuracy of recommenders (Massa & Bhattacharjee, 2004; Guo et al., 2014). The absence of precision is basically because of errors coming with the use of contradicting algorithms, incapable to realize contrasting issues between havingdistrustand faith, putting into consideration the web of having faith (Massa & Bhattacharjee, 2004; Abadi et al., 2016). Picture recovery can be depicted as the errand of looking outfor pics in a picturedata set. This is presentnot an astute thought, considering

everything.It has been explored on account of the way that the 1970s joined informational collection associations with PC vision, lookinginto the issue as indicatedthrough two uncommon perspectives, the first being text-based and the second one being visual-based.

In recent years, with the huge amount of information and users of the internetservice,it is hard to know quickly and accurately what the user wants. This phenomenon leads to extremely low utilization of information, also known as the information overload problem. Traditionally, keywordsare used to retrieve images,but such methodsrequirea 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, ContentBased Information Retrieval (CBIR) has gradually become a researchhotspot.

CBIR retrieves picture objects based entirely on the content. The content of an image needs to be represented by features that represent its uniqueness. Any picture objectcanbe represented by its specific shapes, colors, and textures. These visual characteristics of the image are used as input conditions for the query system, and as a result, the system will recommendthe nearest images and data set. This research designs and implementsa two-stage deep learning-based model that recommends a clothing fashion style. Thismodel can use a 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 retrievethe contiguous relatedimages for the recommendation. Based on data-driven, this thesis uses a convolutional neural networkas a visual extractor of image objects. This experimental model shows and achieves better resultsthan the ones of the previous schemes.

1.2 PURPOSE

The combination of fashion preferences and the above-mentioned factors associated with clothing choices could transmit the image features for a better understanding of consumers' preferences. Therefore, analyzingconsumers' choices and recommendations is valuable to fashion designers and retailers. A recommendation system is an artificialintelligence or AI algorithm, usually associated with machine learning, that uses Big Data to suggest or recommend additional products to consumers. These can be based on various criteria, including past purchases, search history, demographic information, and otherfactors. A recommender system aims **to** estimate the utility of a set of objectsbelonging to a given domain, starting from the information available about users and objects.

Product recommendation engines are an excellent way to deliver customers with an improved user experience. Leveraging advanced algorithms such as machine learning and AI, a recommendation system can help bring customers the relevant products they want or need. Product recommendations are part of an e-Commerce personalization strategy wherein products are dynamically populatedto a user on a webpage, app, or

email based on data such as customer attributes, browsing behavior, or situationalcontext—providing a personalized shoppingexperience.

2. LITERATURE SURVEY

Recommender Systems are typically characterized by their way to deal with the estimation of ratings. Here, we study different types of recommender systems. The definition was expressed in [27], [28] for the first time and has been considered widely.

Additionally,recommender systems are normally arrangedinto these categories,considering how suggestions are made:

1. **Content-based**: Recommendation of items happensbased on how the userfavored items previously.
2. **Collaborative-based**: In this case, we look for users with similar preferencesand tastes, and based on this, the user will be suggesteditems.

2.1 EXISTING PROBLEM

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. It is a problem-solving technique that improves the system and ensures that all the components of the systemwork efficientlyto accomplish their purpose.

2.2 REFERENCES:

i. Park, D. H., Choi, I. Y., Kim, H. K. & Kim, J. K. (2011). A review and classification of recommender systems 1research. International Proceedings of EconomicsDevelopment & Research.

ii. Schafer, J. B., Konstan,J. & Riedl, J. (1999).Recommender systems in ecommerce. Proceedings of the 1st ACM conference on electronic commerce, 1999.

iii. Beel, J., Langer, S., Genzmehr, M., Gipp, B., Breitinger, C. & Nürnberger(2013). A. Researchpaper recommender system evaluation: a quantitative literature survey , 2013.

iv. Chakraborty, S.; Hoque, S.M.A.; Kabir, S.M.F. Predicting fashion trend using runway images: Application of logistic regression in trend forecasting. *Int. J. Fash. Des. Technol.Educ.* **2020**,

v. Garude, D.; Khopkar, A.; Dhake, M.; Laghane, S.; Maktum, T. Skin-tone and occasionoriented outfit recommendation system. *SSRN Electron. J.* **2019**.

vi. Kang, W.-C.; Fang, C.; Wang, Z.; McAuley, J. Visually-aware fashionrecommendation and design with generative image models. In Proceedings of the 2017 IEEE

International Conference on Data Mining (ICDM),New Orleans,LA, USA, 18–21November 2017.

vii. Guan, C.; Qin, S.; Ling, W.; Ding, G. Apparel recommendation system evolution: An empirical review.*Int. J. Cloth. Sci. Technol.* **2016**, *28*, 854–879.

viii. Liu, Y.; Gao, Y.; Feng, S.; Li, Z. Weather-to-garment: Weather-oriented clothing recommendation. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo. (ICME), Hong Kong, China, 31 August 2017; pp. 181– 186.

ix. Yamaguchi, K.; Kiapour, M.H.; Ortiz, L.E.; Berg, T.L. Parsing clothing in fashionphotographs. In Proceedings of the 2012 IEEE Conference on ComputerVision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3570–3577.

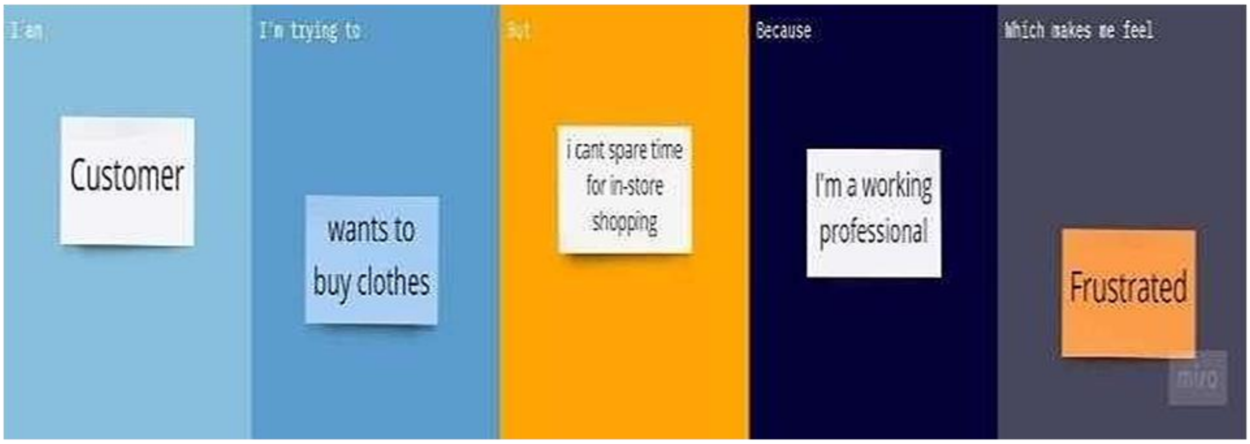
x. Polania, L.F.; Gupte, S. LearningFashion Compatibility Across Apparel Categories for Outfit Recommendation. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019;pp. 4489–4493.

2.3 PROBLEM STATEMENT DEFINITION

The personal information collected by recommenders raises the risk of unwanted exposure of that information. Also, malicious users can biasor sabotage the recommendations that areprovided to other users.In recent years, the textile and fashion industries have witnessed an enormous amount of growth in fast fashion. On e-commerce platforms, where numerous choices are available, an efficient recommendation system is required to sort, order, and efficiently convey relevant product content or information to users.

1. The problem of the work is to design static web applications deployments withcustomer deployment
2. Lack of interaction betweenapplication and user
3. User need to navigateacross multiple pages to choose right product
4. Confusion in choosing product

5. Lack of sales
6. Complex User Interface
7. Lack of proper guidance



3. IDEATION &PROPOSED SOLUTION

We have come up with a new innovative solution through which you can directly do youronline 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:

a. Admin and
b. 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 theusers are purchasing.

USER :
The user will login into the website and go through the products available on the website. Insteadof navigating to several screensfor booking products online, the usercan directly talk to Chatbot regarding the products. Get the recommendations based on information provided by the user.

FEATURES OF CHATBOT :

a.Using chatbot we can manage user's choices and orders.

b.The chatbot can give recommendations to the usersbased on their interests.

c.It can promotethe best deals and offerson that day.

d.It will store the customer's details and orders in the database.

e.The chatbot will send a notification to customers if the order is confirmed.

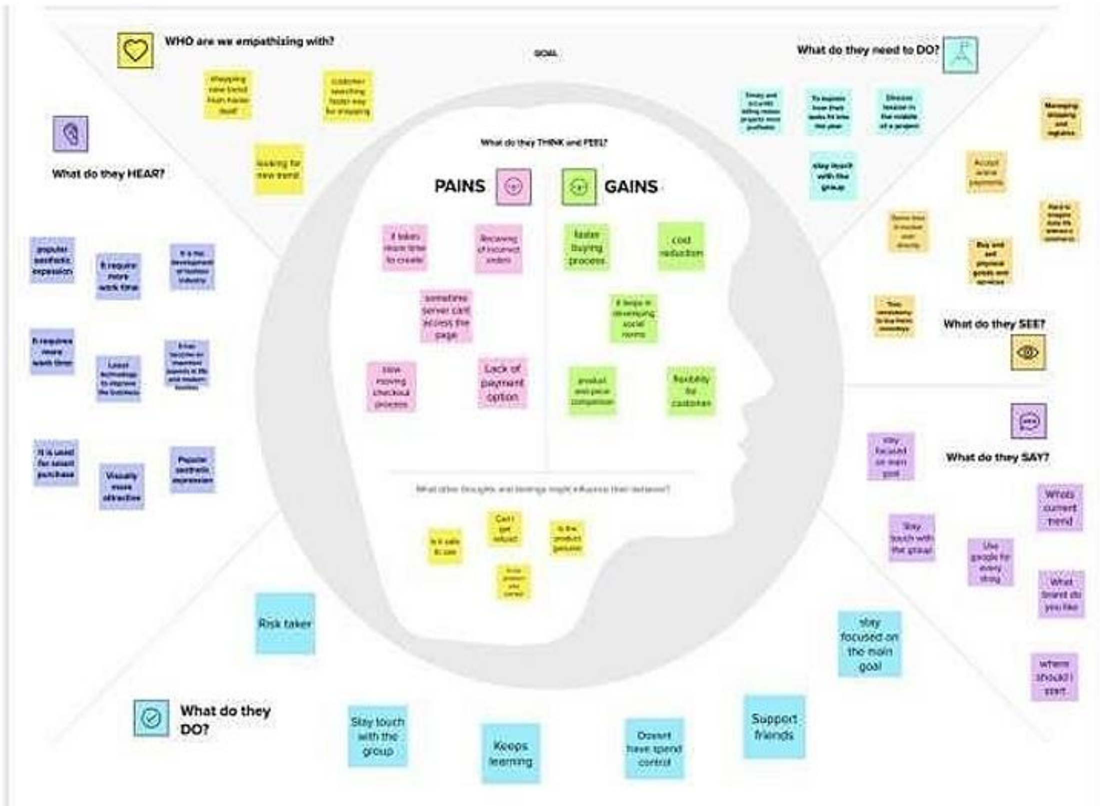
f.Chatbots can also help in collecting customerfeedback.

3.1 EMPATHY MAP CANVAS:

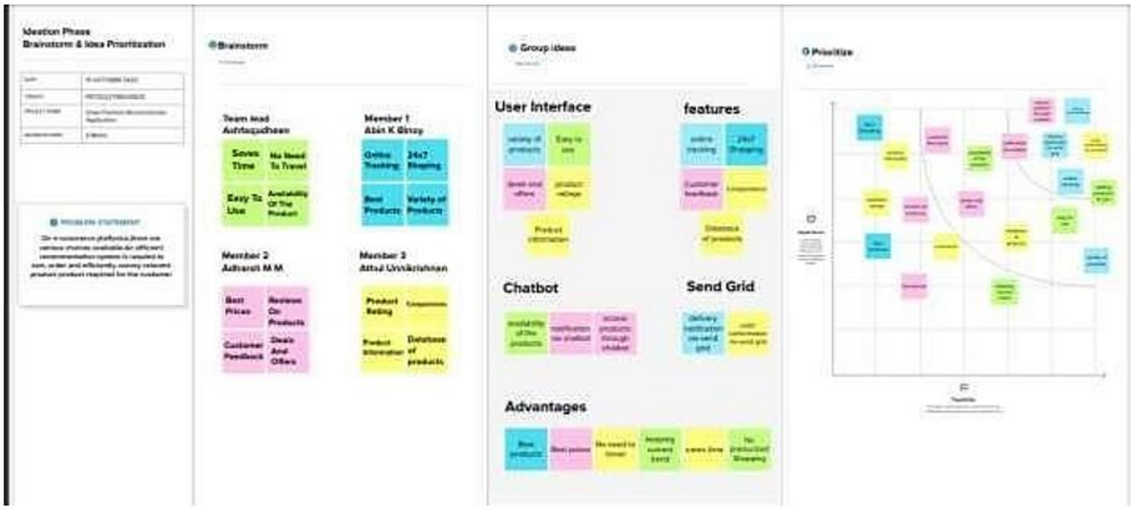
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exerciseof creating the map helps participants consider things fromthe user's perspective along with his or her goals and challenges. An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers.

3.2 IDEATION & BRAINSTROMING:

A group problem-solving technique that involves the spontaneous contribution of ideasfrom all members of the group.The mulling over of ideas by one or more individuals in an attempt to devise or find asolution to a problem.



WHO are we empathizing with?
GOAL
What do they need to DO?
What do they HEAR?
looking for new trend
What do they THINK and FEEL?
PAINS
GAINS
popular aesthetic expression
It require more work time
sometime server cant access the page
Slow moving checkout process
Lack of payment option
faster buying process
cost reduction
product and price comparison
flexibility for customer
Popular aesthetic expression
Visually more attractive
What do they SEE?
What do they SAY?
stay focused on main goal
Whats current trend
Stay touch with the group
Use google for every thing
What brand do you like
where should I start
Can't get refund
Is the product genuine
Risk taker
What do they DO?
Stay touch with the group
Keeps learning
Doesnt have spend control
Support friends
stay focused on the main goal



3.3 PROPOSED SOLUTION :

SMART FASHION RECOMMENDER APPLICATION

Project team shall fill the following information in proposedsolution template.

S.No.	Parameter	Description
•	Problem Statement (Problem to besolved)	Customer can use the app to browse the products and add them to their shopping cart.The bot will assist users in receiving productrecommendation.
•	Idea / Solution description	We have come up with a new innovative solution through which you can directly do youronline shopping based on your choice without any search. It can be doneby using the chatbot.
•	Novelty / Uniqueness	Share design inspirations to chatbot. Utilize user's 3D model to find an outfit.

•	Social Impact / Customer Satisfaction	Instead of navigating to several screens for booking products online, the user can directly talk to Chatbot regarding the products. We can visualize ourselves as a 3D model, for thebetter understanding of how the product suitsus.
•	Business Model (Revenue Model)	While getting a big order from a major retailermight sound like a good thing for a fledgling brand, it means the brand has a short time to somehow produce that inventory and hire thenecessary employees without any money upfront.
•	Scalability of the Solution	Technological developments such as colorchanges and the integration of conductivesensors etc. Could revolutionize the way designers thinkabout fashion.

3.3 PROBLE

3. REQUIREMENT ANALYSIS

3.1 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 numberRegistration throughLinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Advanced SearchCapabilities	sorting and filtering options
FR-4	Checking itemavailability	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 cardpayment, paying withmobile wallets
FR-7	Checking the shipping status	Option to easily check the shipping status of itemsordered in the store

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**Sign up**	Register by using mobile number/ Register by using email id.
FR-2	**User Verification**	Verify via Email Verify via OTP
FR-3	**Login**	Login by using username / password
FR-4	**Profile Updation**	Update the profile details like Name,Gender, Age ,Address & mobile number ,etc,.
FR-5	**Chatbot**	Chatbot is useful to search products , view offers,discounts and stock availability. It is also used to solve queries and issues.
FR-6	**Ordering the product**	After confirming the product , buy the product via Cash on Delivery or online transactions.
FR-7	**Tracking the ordered Product**	After ordering the product , track the delivery via link received to your registered mobile number through SMS or registered email id.
FR-8	**Logout**	After receiving the product ,user can logout the account when he/she needs

4.2 NON-FUNCTIONAL REQUIREMENTS:

SMART FASHION RECOMMENDER APPLICATION

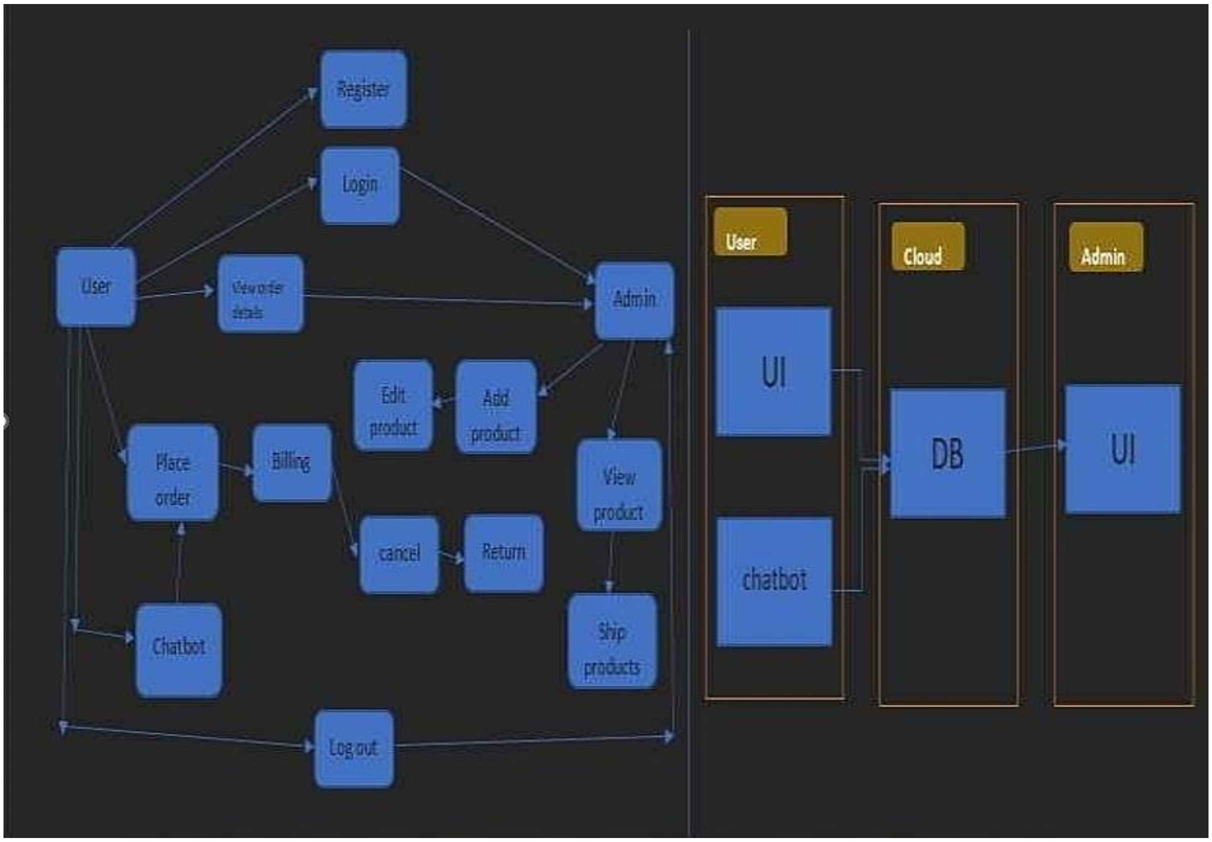
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application will be designed in such a way that any user can easily navigate through it and user can easily view , order and track the product until delivery.(Easy and Compact design.)
NFR-2	**Security**	Using of SSL (Secure Socket Layer) certificate (Python Flask to Cloud connect) will provide security to the project. The user details will be kept as more secure.
NFR-3	**Reliability**	To make sure the application doesn't go down due to network traffic and the details entered in this application is kept as highly confidential, so it is highly reliable.
NFR-4	**Performance**	It focus on loading the application as quickly as possible irrespective of the number of users/integrator traffic.
NFR-5	**Availability**	This application will be available to all users (network connectivity is necessary) at any given point of time. Users can access the chatbot for raising any queries/ questions.
NFR-6	**Scalability**	Chatbot can be very useful during festival season to know about offers and discounts. It will be helpful whenever we make online shopping.

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flow While a system. A neat and clear DFD can depict the right among of the system requirement Graphically. It show how data enters and leaves the system , what changes the information,and where data is stored

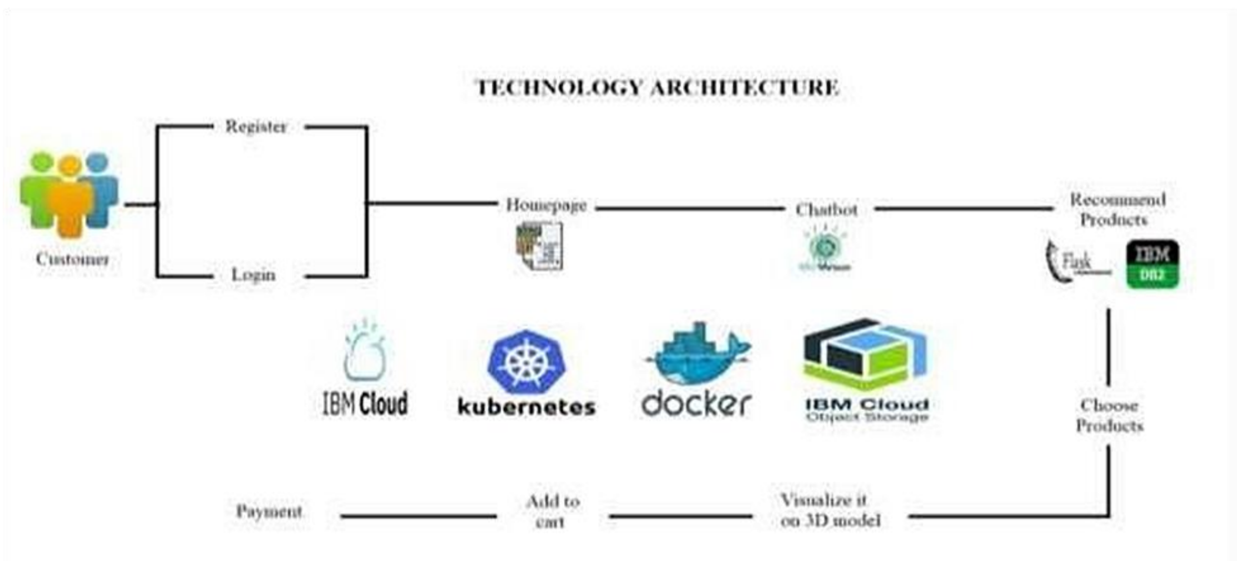


5.2 SOLUTION & TECHNICAL ARCHITECTURE:

We have developed 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
2. User

Instead of searching for products in the search bar and navigating to individual products tofind required preferences, this project leverages the use of chatbots to gather all required preferences and recommend products to the user. The solution is implemented in such a way as to improve the interactivity betweencustomers and applications. The chatbot sendsmessagesperiodically to notify offers and preferences. For security concerns,this application uses a token to authenticate and authorize users securely. The token has encodeduser id and role. Based on the encoded information, access to the resources is restricted to specific users.



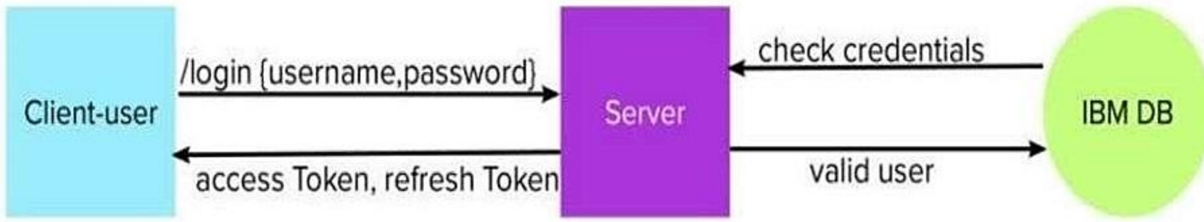


FIG 3: AUTHENTICATION PROCESS

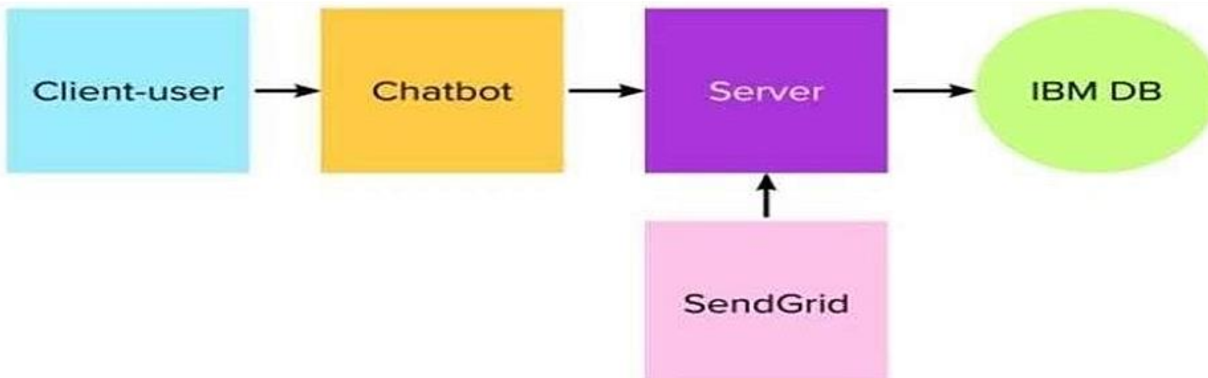


FIG 4: CLIENT-SERVER PROCESS

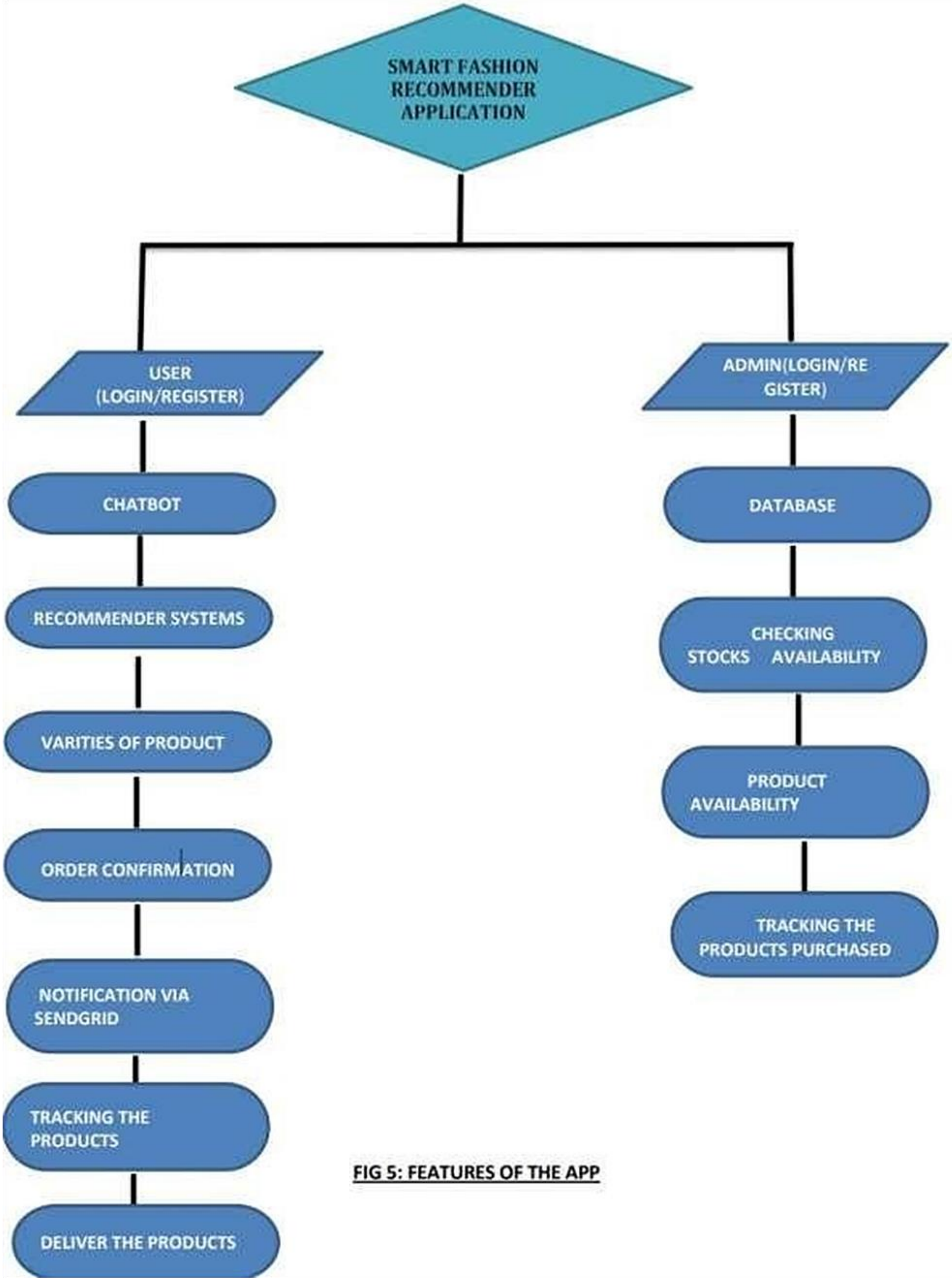


FIG 5: FEATURES OF THE APP

5.1 USER STORIES:

Use the below template to list all the user stories for the product.

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	I can access my data by login	High	Sprint-1
	Dashboard	USN-6	As a user , I can view the dashboard and by products		High	Sprit -2
Customer (Web user)	Registration / Login	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard		Sprint -1
Customer Care Executive	Contact with Customers	USN-8	As a Customer customers care executive, I solve the customer Requirements and feedback	I can receive calls from customers	High	Sprint-1

6. PROJECT PLANNING AND SCHEDULE

6.1 SPRINT PLANNING & ESTIMATION:

Milestones	Activities	Description
Project Development Phase	Delivery of Sprint – 1,2,3,4	To develop the code and submit the developed code by testing it
Setting up App environment	Create IBM Cloud account	Signup for an IBM Cloud account
	Create flask project	Getting started with Flask to create project
	Install IBM Cloud CLI	Install IBM Command LineInterface
	Docker CLI Installation	Installing Docker CLI on laptop
	Create an account in send grid	Create an account in sendgrid. Use the service as email integration to our application for sending emails
Implementing web Application	Create UI to interact with Application	Create UI • Registration page • Login page • View products page • Add products page
	Create IBM DB2 & connect with python	Create IBM DB2 service in IBM Cloud and connect with python code with DB
Integrating sendgrid service	Sendgrid integration with python	To send emails form the application we need to integrate the Sendgrid service
Developing a chatbot	Building a chatbot and Integrate to application	Build the chatbot and Integrate it to the flask application
Deployment of App in IBMCloud	Containerize the App	Create a docker image of your application and push it to the IBM container registry
	Upload image to IBM container registry	Upload the image to IBM container registry
	Deploy in kubernetes cluster	Once the image is uploaded to IBM Container registry deploy the image to IBM Kubernetes cluster

Milestones	Activities	Description
Ideation Phase	Literature Survey	Literature survey on the selected project & information gathering
	Empathy Map	Prepare Empathy map to capture the user Panis & Gains, prepare list of problem statement
	Ideation	Organizing the brainstorming session and priorities the top 3 ideas based on feasibility & Importance
Project Design Phase I	Proposed Solution	Prepare proposed solution document which includes novelty, feasibility of ideas, business model, social impact, Scalability of solution
	Problem Solution Fit	Prepare problem solution fit document
	Solution Architecture	Prepare solution architecture document
Project Design Phase II	Customer Journey	Prepare customer journey map to understand the user interactions & experience with the application
	Functional requirement	Prepare functional & non functional requirement document
	Data Flow Diagram	Prepare Data Flow Diagramand user stories
	Technology architecture	Draw the technology architecture diagram
Project Planning Phase	Milestones & Activity list	Prepare milestones and activity list of the project
	Sprint Delivery Plan	Prepare sprint delivery plan

6.1 SPRINT DELIVERY SCHEDULE

Project Tracker, Velocity & Burndown Chart

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	18	6 Days	24 Oct 2022	29 Oct 2022	24	29 Oct 2022
Sprint-2	18	6 Days	31 Oct 2022	05 Nov 2022	24	05 Nov 2022
Sprint-3	18	6 Days	07 Nov 2022	12 Nov 2022	24	12 Nov 2022
Sprint-4	18	6 Days	14 Nov 2022	19 Nov 2022	24	19 Nov 2022

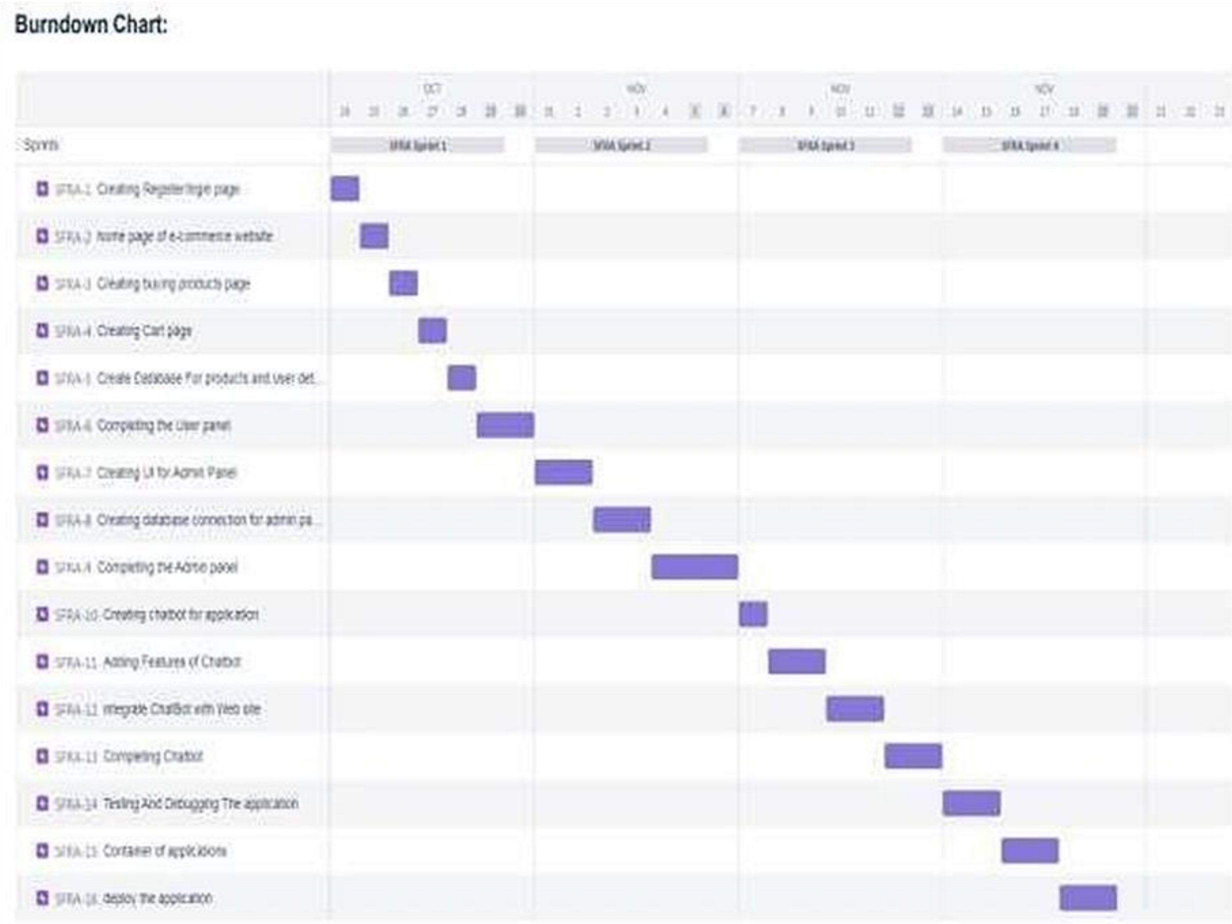
Velocity

Imagine we have a 6-day sprint duration, and the velocity of the team is 18(points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV = Sprint Duration / Velocity

AV = 24/6 = 4

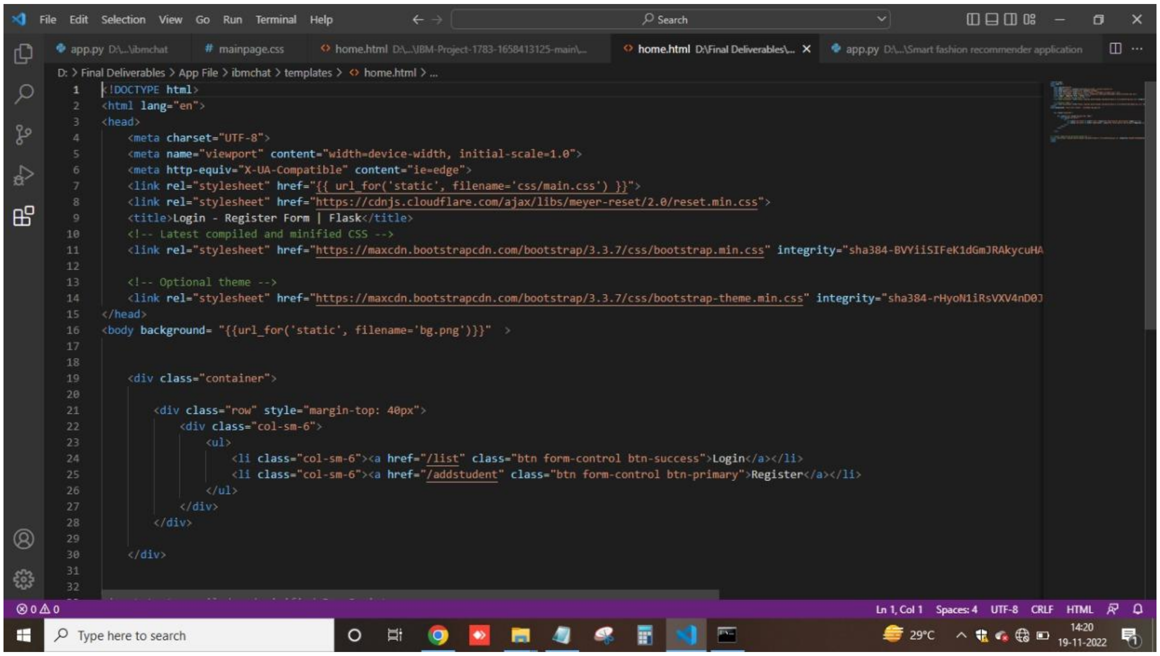
6.1 EPORTS FROM JIRA



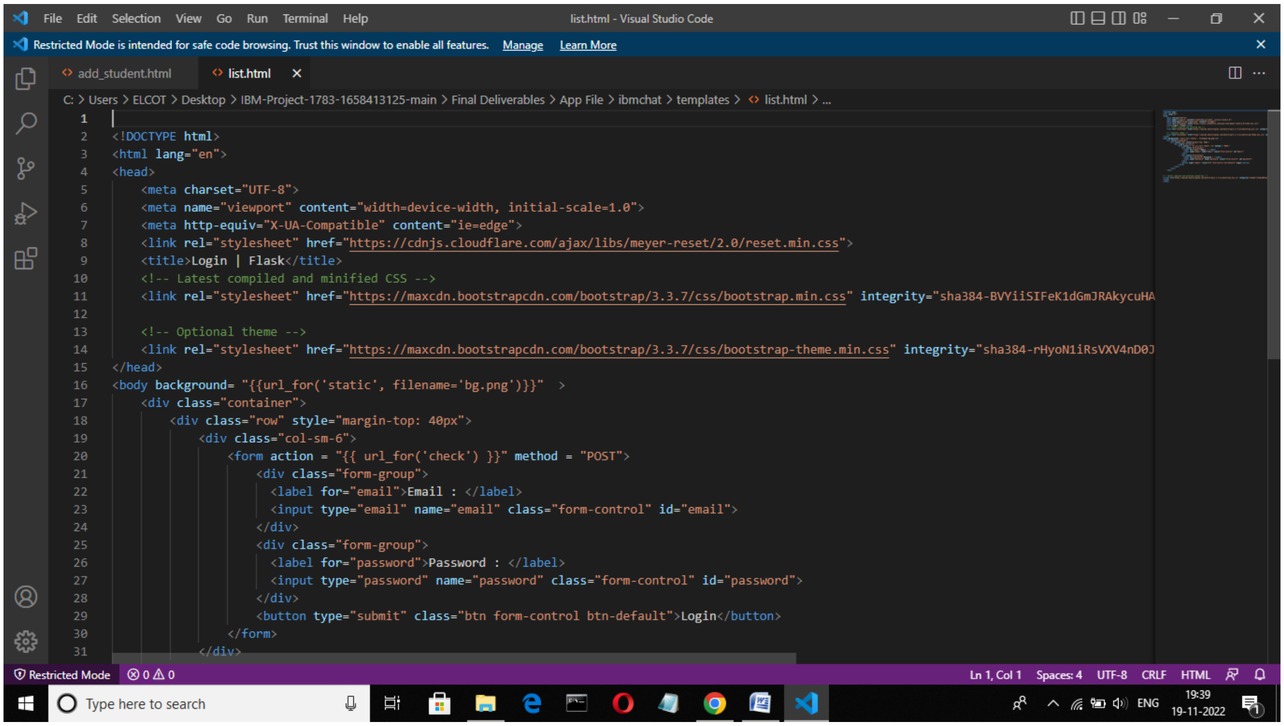
CODING AND SOLUTIONING

7.1 FEATURE-1:

HOMEPAGE.HTML:

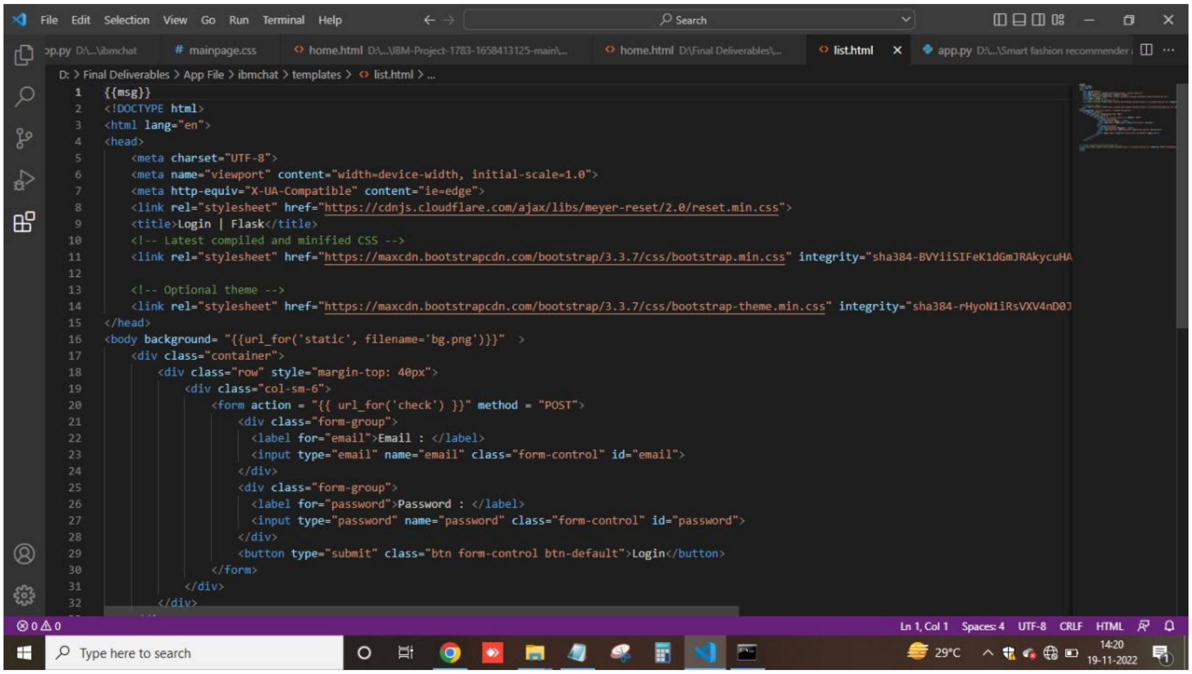


REGISTER HTML:



7.2 FEATURE-2:

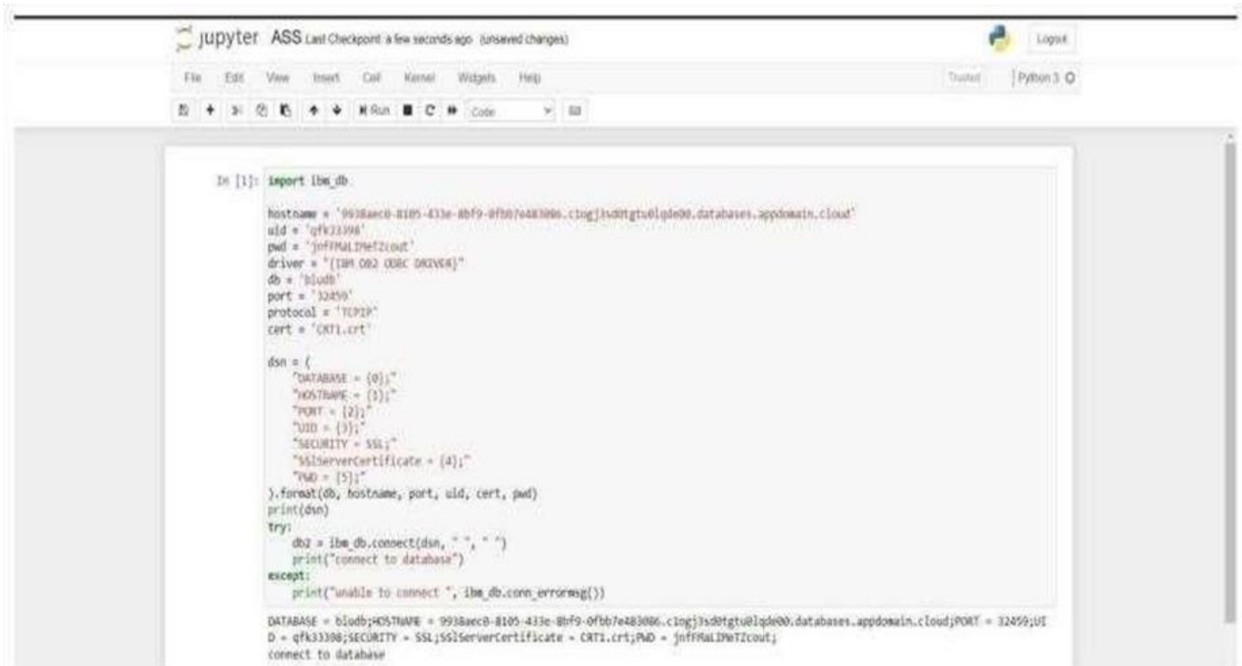
LOGIN.HTML:



CHATBOT (SOURCE CODE):

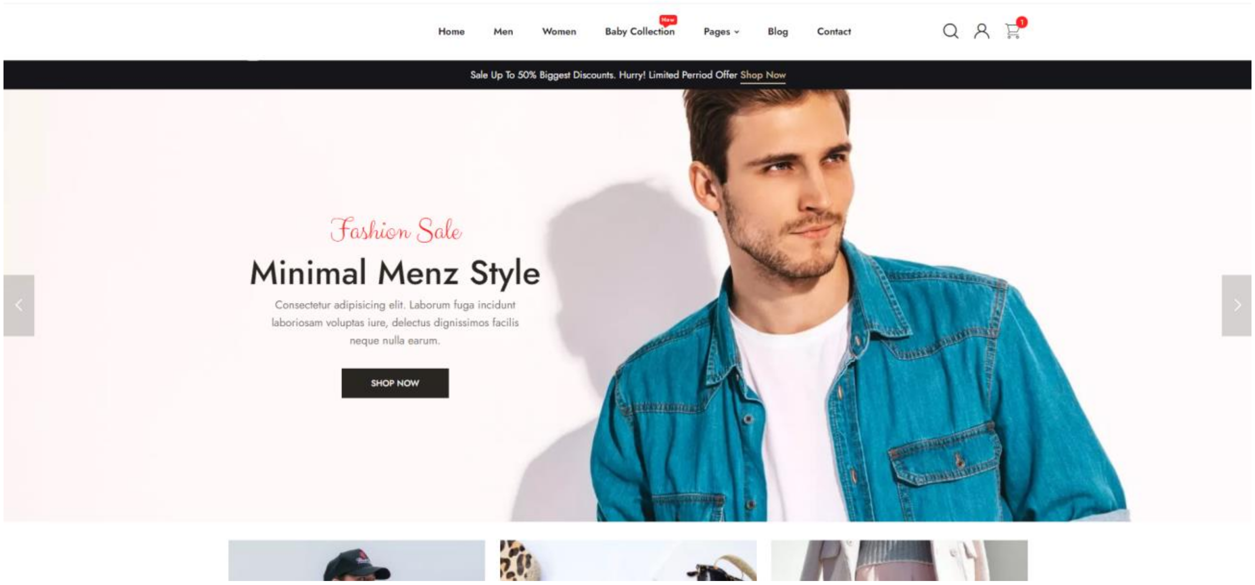
```
function  talk(){
    var know = {
    "Hi" : "Hello ",
    "What say about your smart fashion?" : "This shopping hall is 2011 is
start in smart fashion shopping and all vertical dress available.",
    "What about your dress collection?" : "Court and shirts, Sarees, Jean
Pants, Kid's Dress, ect...",
    "How to Purchasing?" : "Your see the top navbar, click on purchase button
then you have purchasing dresses.",
    "Your followers" : "I have my family of 5000 members, i don't have
follower ,have supportive Famiy ",
    "ok" : "Thank You So Much ",
    "Bye" : "Okay! Will meet soon.."
    };
    var user = document.getElementById('userBox').value;
    document.getElementById('chatLog').innerHTML = user + "<br>";
    if (user in know) {
    document.getElementById('chatLog').innerHTML = know[user] + "<br>";
    }else{
    document.getElementById('chatLog').innerHTML = "Sorry,I didn't understand
<br>";
    }
    }
```

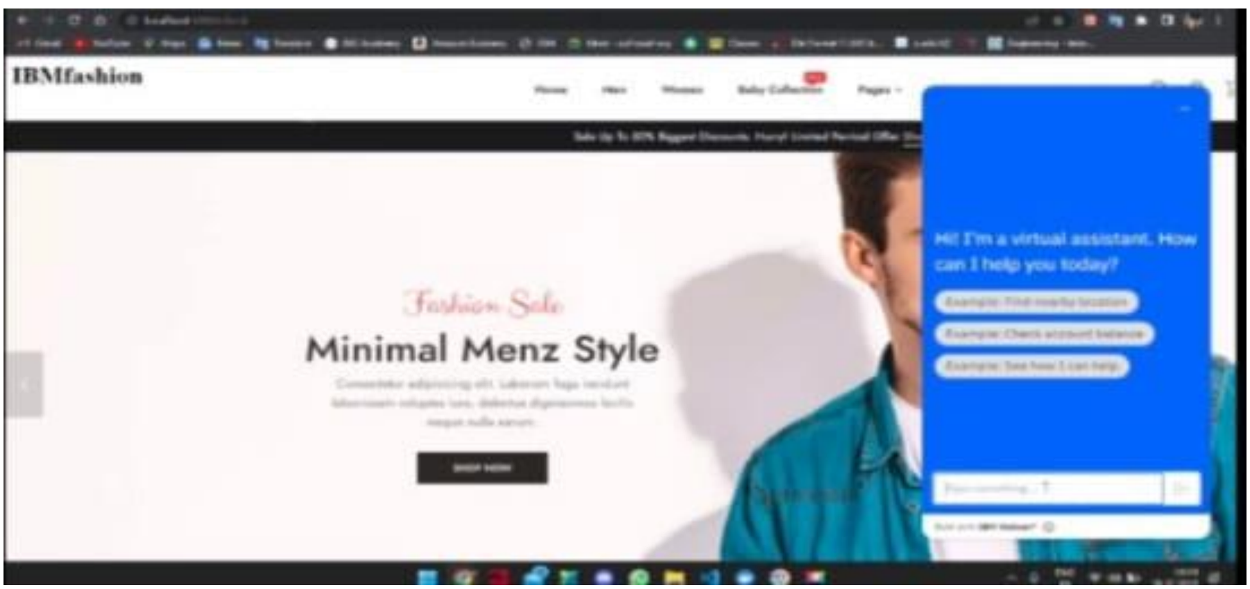
7.3 DATABASE SCHEMA:



TESTING

8.1 TEST CASES:





9.RESULTS

9.1 PERFORMANCE METRICS:

The performance of a recommendation algorithm is evaluated by using some specific metrics that indicate the accuracy of the system.The type of metric used depends on the type of filteringtechnique. Root Mean Square Error (RMSE), Receiver Operating Characteristics (ROC), Area Under Cover

(AUC), Precision, Recall and F1 score is generally used to evaluatethe performance or accuracy of the recommendation algorithms.

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

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

where, Np is the total number of predictions, pui is the predicted rating that a user u will select an item i and rui is the real rating.

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

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

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

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

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

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

F1 Score. F1 score is an indicator of the accuracyof the model and rangesfrom 0 to 1, whereavalue close to 1 represents higher recommendation or prediction accuracy.It represents precisionand recall as a singlemetric and can be as represented as follows:

$$F1\ score = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

Coverage. Coverage is used to measure the percentage of items whichare recommended by thealgorithm among all of the items.

Accuracy. Accuracy can be definedas the ratio of the number of total correctrecommendationsto the total recommendations provided, which can be as represented as follows:

$$Accuracy = \frac{TP + FN}{TP + FN + TN + FP} \tag{5}$$

Intersection over union (IoU). It represents the accuracy of an objectdetector used on a specificdataset.

$$IoU = \frac{TP}{TP + FN + FP} \tag{6}$$

ROC. ROC curve is used to conducta comprehensive assessment of the algorithm's performance [57].

AUC. AUC measures the performance of recommendation and its baselines as well as the quality of the ranking based on pairwise comparisons [5].

Rank aware top-N metrics. The rank aware top-N recommendation metric finds some of the interesting and unknown items that are presumed to be most attractive to a user [63]. Mean reciprocal rank (MRR), mean average precision (MAP) and normalized discounted cumulative gain(NDCG) are three most popular rank aware metrics.

MRR. MRR is calculated as a mean of the reciprocal of the position or rank of first relevant recommendation [64][65]. MRR as mentioned by [64][65] can be expressed as follows:

$$MRR = \frac{1}{N_u} \sum_{u \in N_u} \frac{1}{L_u^n [k] \in R_u} \tag{7}$$

where u, Nu and Ru indicate specificuser, total numberof users and the set of itemsrated by theuser, respectively. L indicates list of rankinglength (n) for user (u) and k represents the positionof the item found in the he lists L.

MAP: MAP is calculated by determining the mean of average precision at the points where relevantproducts or items are found. MAP as mentioned by [65] can be expressed as follows.

$$MAP = \frac{1}{N_u |R_u|} \sum_{k=1}^{n} \mathbb{1} (L_u^n [k] \in R_u) P_u@k \tag{8}$$

where Pu represents precision in selecting relevant item for the user. NDCG: NDCG is calculated by determining the graded relevanceand positional information ofthe recommended items, which can be expressed as follows [65].

$$NDCG_u = \frac{\sum_{k=1}^{n} G(u,n,k) D(k)}{\sum_{k=1}^{n} G^*(u,n,k) D(k)} \tag{9}$$

where D (k) is a discounting function, G (u, n, k) is the gain obtained recommending an item found at k-thposition from the list L and G^* (u, n,k) is the gain related to k-thitem in the idealranking of n size for u user.

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES:

1. Smart fashion recommender application is the user friendly.
2. With the help of chatbot user can find the products very easily.
3. This application used to discover the product based on the users choice,very easily and quickly.

DISADVANTAGES:

1. It need active internet connection.
2. Privacy concerns.
3. Too many choices.
4. Cold-start problem.

11.CONCLUSION

The Fashion Recommendation System is mainly used to recommend the best possible outfit combinations to a user who has no fashion sense based on their wardrobe . It may not always provide the best possible outfit to wear for an occasion as the system is dependent completelyon the clothes present in the user's wardrobe. Also another reason is that fashion is highly dependent on the time period. However the system does a great job in inculcating a fashion sense among the users and can providethe best recommendations based on the user's wardrobe.Since the systemis implemented as a website,it is very easy for the end users to access as wellas use. The scope of this system can be expanded by including the ability to detect the variousdesign and patterns on clothing, and to increase the number of occasions.

:

12.FUTURE SCOPE

In the future, to implement this recommendation system to be extended to include male and non-binary fashion items including apparel, footwear, accessories etc. This work can further be enhanced to predict fashion items based on the skin colour and weather conditions.

Future research should concentrate on including time series analysisand accurate categorization of product images based on the variation in colour, trend and clothing style in order to developan effective recommendation system. The proposed modelwill follow brand-specific personalization campaigns and hence it will ensure highly curated and tailored. offerings for users. Hence, this research will be highly beneficial for researchers interested in using augmented and virtual reality featuresto develop recommendation systems.

13.APPENDIX

SOURCE CODE:

```
from flask import Flask, render_template, request, redirect, url_for, session
from markupsafe import escape
import os
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30119;SECURITY=S
SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=xhx40038;PWD=BDz5ow7439yj5P
Ed",",")
print ("Database connection established", conn)

app = Flask(__name__)


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

@app.route('/addstudent')
def new_student():
    message =
Mail(from_email="nithishjaganathanpersonal@gmail.com",to_emails="nithishjaganathan@gmai
l.com",subject="Account Registered Successfully",html_content="<p>Your account has been
created using you provided email address.</p>")

    try:
     sg = SendGridAPIClient("SG.Xng1uu2bQKSzCgu8j_Hj8Q.UFutNdzc2iwdrMfcbbdP4nmBa-
r3NEex-KWLdtMUbTo")
     response = sg.send(message)

    except Exception as e:
     print(e)
    return render_template('add_student.html')
@app.route('/list')
def list():
  return render_template('list.html')
```

```
@app.route('/addrec',methods = ['POST', 'GET'])
def addrec():
  if request.method == 'POST':

    name = request.form['name']
    email = request.form['email']
    password = request.form['password']

    sql = "SELECT * FROM userdata WHERE name=? "
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,name)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)

    if account:
      return render_template('list.html', msg="You are already a user, please login using your
details")
    else:
      insert_sql = "INSERT INTO userdata VALUES (?,?,?)"
      prep_stmt = ibm_db.prepare(conn, insert_sql)
      ibm_db.bind_param(prep_stmt, 1, name)
      ibm_db.bind_param(prep_stmt, 2, email)
      ibm_db.bind_param(prep_stmt, 3, password)
      ibm_db.execute(prep_stmt)
    return render_template('home.html', msg="Registered successfully")



@app.route('/check',methods = ['POST', 'GET'])
def check():
   if request.method == 'POST':
    email = request.form['email']
    password = request.form['password']

    sql = "SELECT * FROM userdata WHERE email=? and password= ?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.bind_param(stmt,2,password)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    if account:
      return render_template('result.html', msg="")
    else:
      return render_template('list.html', msg="Please check your credentials!")
```

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)





  # # while student != False:
  # #   print ("The Name is : ",  student)

  # print(student)


# @app.route('/posts/edit/<int:id>', methods=['GET', 'POST'])
# def edit(id):
#     post = BlogPost.query.get_or_404(id)

#     if request.method == 'POST':
#         post.title = request.form['title']
#         post.author = request.form['author']
#         post.content = request.form['content']
#         db.session.commit()
#         return redirect('/posts')
#     else:
#         return render_template('edit.html', post=post)
```

GITHUB & PROJECT DEMO LINK:

GITHUB LINK:
https://github.com/IBM-EPBL/IBM-Project-52502-1661007451

PROJECT DEMO LINK:

Project Tracker, Velocity & Burndown Chart

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	18	6 Days	24 Oct 2022	29 Oct 2022	24	29 Oct 2022
Sprint-2	18	6 Days	31 Oct 2022	05 Nov 2022	24	05 Nov 2022
Sprint-3	18	6 Days	07 Nov 2022	12 Nov 2022	24	12 Nov 2022
Sprint-4	18	6 Days	14 Nov 2022	19 Nov 2022	24	19 Nov 2022

Velocity

Imagine we have a 6-day sprint duration, and the velocity of the team is 18(points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV = Sprint Duration / Velocity

AV = 24/6 = 4