SMART FASHION RECOMMENDER APPLICATION

1.1 PROJECT OVERVIEW

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 Chat bot.

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 Chat bot regarding the products. Get the recommendations based on information provided by the user.

Features of Chat bot :

- Using Chat bot we can manage user's choices and orders.
- The Chat bot 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 Chat bot will send a notification to customers if the order is confirmed.
- Chat bots can also help in collecting customer feedback.

1.2 PURPOSE

Fashion is such an industry where luxury goods can only be bought in a few physical boutiques and one to one customer service is crucial. The Internet changed this dramatically, by giving the customers a smooth but a very detached experience of shopping. This particular problem can be solved by Chat bots. Customers can be provided with personalized services through Fashion Chat bot, which can exchange messages, give required suggestions and information

2. LITERATURE SURVEY

2.1 EXISTING PROBLEMS

Unfortunately, with all the ease and convenience of online shopping, a shopping cart cannot say “this dress looks like it was made especially for you, ma’am” or “may I suggest a scarf which would look just right with this blouse?”

Lack of an efficient Chat bot connected with an agent seems makes it hard for the users to search the right product. Users also lack the experience of the review system, that might help them to check if the product is to purchased is good enough as mentioned in the application.

Fashion brands are turning to technology, artificial intelligence, in particular, to recreate that personalized atmosphere of a beauty boutique only right within their online applications.

2.2 REFERENCES

The references that follow are cited in the above text. Additional references for the papers filtered from results of the searches over the selected databases are listed below.

- Aarthi, N. G. (2020). Chatbot for retail shop evaluation. *International Journal of Computer Science and Mobile Computing*, *9*(3), 69–77. [Google Scholar]
- Ameen, N., Hosany, S., & Tarhini, A. (2021). Consumer interaction with cutting-edge technologies: Implications for future research. *Computers in Human Behavior*, *120*, 106761. [Crossref], [Web of Science ®], [Google Scholar]
- Chaves, A. P., & Gerosa, M. A. (2020). How should My Chatbot interact? A survey on social characteristics in human–chatbot interaction design. *International Journal of Human–Computer Interaction*, *0*(0), 1–30. [Taylor & Francis Online], [Google Scholar]
- Chen, S., Li, C., Ji, F., Zhou, W., & Chen, H. (2019). *Review-driven answer generation for product-related questions in E-commerce*. Proceedings of WSDM ‘19. ACM, New York, NY, USA, 411–419. [Crossref], [Google Scholar]
- Cheng, Y., & Jiang, H. (2020). How Do AI-driven chatbots impact user experience? Examining gratifications, perceived privacy risk, satisfaction, loyalty, and continued Use. *Journal of Business Economics and Management*, *64*(4), 592–614. [Google Scholar]
- Chung, M., Ko, E., Joung, H., & Kim, S. J. (2020). Chatbot e-service and customer satisfaction regarding luxury brands. *Journal of Business Research*, *117*, 587–595. [Crossref], [Web of Science ®], [Google Scholar]
- Colombi, C., Kim, P., & Wyatt, N. (2018). Fashion retailing “tech-gagement”: engagement fueled by new technology. *Research Journal of Textile and Apparel*, *22*(4), 390–406. [Crossref], [Web of Science ®], [Google Scholar]
- Copulsky, J. (2019). Do conversational platforms represent the next big digital marketing opportunity? *Applied Marketing Analytics*, *4*(4), 311–316. [Google Scholar]

- De Carolis, B., De Gemmis, M., & Lops, P. (2015). *A multimodal framework for recognizing emotional feedback in conversational recommender systems*. ACM International Conference Proceeding series, 2015-September, 11–18. [Crossref], [Google Scholar]
- De Carolis, B., de Gemmis, M., Lops, P., & Palestra, G. (2017). Recognizing users feedback from non-verbal communicative acts in conversational recommender systems. *Pattern Recognition Letters*, *99*, 87–95. [Crossref], [Web of Science ®], [Google Scholar]
- Diederich, S., Brendel, A. B., & Kolbe, L. (2019). On conversational agents in Information Systems research: Analyzing the past to guide future work. *Wirtschaftsinformatik*, *85*, 1550–1564. [Google Scholar]
- Eisman, E. M., Navarro, M., & Castro, J. L. (2016). A multi-agent conversational system with heterogeneous data sources access. *Expert Systems With Applications*, *53*, 172–191. [Crossref], [Web of Science ®], [Google Scholar]
- Gao, S., Zhao, D., Ren, Z., Yin, D., Zhao, Y., & Yan, R. (2019). *Product-aware answer generation in e-commerce question-answering*. WSDM, 2019, 429–437. [Crossref], [Google Scholar]
- Go, E., & Sundar, S. S. (2019). Humanizing chatbots: The effects of visual, identity and conversational cues on humanness perceptions. *Computers in Human Behavior*, *97*, 304–316. [Crossref], [Web of Science ®], [Google Scholar]
- Grewal, D., Roggeveen, A. L., & Nordfält, J. (2017). The future of retailing. *Journal of Retailing*, *93*(1), 1–6. [Crossref], [Web of Science ®], [Google Scholar]
- Jurafsky, D., & Martin, J. H. (2020). *Speech and language processing (3rd ed. Draft)*. Upper Saddle River, NJ: Prentice Hall. [Google Scholar]
- Jusoh, S. (2018). *Intelligent conversational agent for online sales*. Proceedings of ECAI 2018, 1–4. [Crossref], [Google Scholar]
- Kasilingam, D. L. (2020). Understanding the attitude and intention to use smartphone chatbots for shopping. *Technology in Society*, *62*, 101280. [Crossref], [Web of Science ®], [Google Scholar]
- Lai, P., & Westland, S. (2020). Machine learning for colour palette extraction from fashion runway images. *International Journal of Fashion design. Technology and Education*, *13*(3), 334–340. [Taylor & Francis Online], [Google Scholar]
- Li, J., Galley, M., Brockett, C., Gao, J., & Dolan, B. (2016). *A diversity-promoting objective function for neural conversation models*. Proceedings of NAACL-HLT2016, 110-119. [Crossref], [Google Scholar]
- Li, L., Li, C., & Ji, D. (2021). Deep context modeling for multi-turn response selection in dialogue systems. *Information Processing & Management*, *58*(1), 102415. [Crossref], [Web of Science ®], [Google Scholar]
- Liao, L., Zhou, Y., Ma, Y., Hong, R., & Chua, T. S. (2018). *Knowledge-aware multimodal fashion chatbot*. Proceedings of ACM MM 2018, 1265–1266. [Crossref], [Google Scholar]
- Liu, C., Jiang, J., Xiong, C., Yang, Y., & Ye, J. (2020). *Towards Building an Intelligent chatbot for customer service: Learning to respond at the appropriate time*. ACM SIGKDD KDD, 3377–3385. [Crossref], [Google Scholar]
- Marulli, F., Pota, M., & Esposito, M. (2018). *A comparison of character and word embeddings in bidirectional LSTMs for POS tagging in Italian*. KES-IIMSS-18, 14–23. [Google Scholar]
- Mauldin, M. L. (1994). Chatterbots, tinymuds, and the turing test: Entering the loebner prize competition. *AAAI*, *94*, 16–21. [Google Scholar]
- Merrilees, B., & Miller, D. (2019). Companion shopping: The influence on mall brand experiences. *Marketing Intelligence & Planning*, *37*(4), 465–478. [Crossref], [Web of Science ®], [Google Scholar]
- Moriuchi, E., Landers, V. M., Colton, D., & Hair, N. (2020). Engagement with chatbots versus augmented reality interactive technology in e-commerce. *Journal of Strategic Marketing*, *0*(0), 1–15. [Taylor & Francis Online], [Google Scholar]
- Morotti, E., Donatiello, L., & Marfia, G. (2020, March). *Fostering fashion retail experiences through virtual reality and voice assistants*. 2020 IEEE VRW (pp. 338-342). IEEE. [Crossref], [Google Scholar]
- Nazir, A., Khan, M. Y., Ahmed, T., Jami, S. I., & Wasi, S. (2019). A novel approach for ontology-driven information retrieving chatbot for fashion brands. *International Journal of Advanced Computer Science and Applications*, *10*(9), 546–552. [Crossref], [Web of Science ®], [Google Scholar]
- Nie, L., Wang, W., Hong, R., Wang, M., & Tian, Q. (2019). *Multimodal dialog system: Generating responses via adaptive decoders*. MM 2019, 1098–1106. [Crossref], [Google Scholar]

- Okoli, C. (2015). A guide to conducting a standalone systematic literature review. *Communications of the Association for Information Systems*, *37*(1), 43. [Google Scholar]
- Palma, M. D. C. O., Seeger, A. M., & Heinzl, A. (2020). Mitigating information overload in e-commerce interactions with conversational agents. In F. D. Davis, R. Riedl, J. vom Brocke, P.-M. Léger, A. Randolph, & T. Fischer (Eds.), *Information Systems and neuroscience* (pp. 221–228). Cham: Springer. [Crossref], [Google Scholar]
- Pricilla, C., Lestari, D. P., & Dharma, D. (2018). *Designing interaction for chatbot-based conversational commerce with user-centered design*. ICAICTA, 2018, 244–249. [Crossref], [Google Scholar]
- Qiu, M., Li, F. L., Wang, S., Gao, X., Chen, Y., Zhao, W., … Chu, W. (2017). *Alime chat: A sequence to sequence and rerank based chatbot engine*. ACL 2017 - (long papers), *2*, 498–503. [Crossref], [Google Scholar]
- Rese, A., Ganster, L., & Baier, D. (2020). Chatbots in retailers' customer communication: How to measure their acceptance? *Journal of Retailing and Consumer Services*, *56*, 102176. [Crossref], [Web of Science ®], [Google Scholar]
- Roh, J. S., Chi, Y. S., & Kang, T. J. (2010). Wearable textile antennas. International Journal of Fashion design. *Technology and Education*, *3*(3), 135–153. [Google Scholar]
- Sanny, L., Susastra, A., Roberts, C., & Yusramdaleni, R. (2020). The analysis of customer satisfaction factors which influence chatbot acceptance in Indonesia. *Management Science Letters*, *10*(6), 1225–1232. [Crossref], [Google Scholar]

2.3 PROBLEM STATEMENT DEFINTION

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. People feel difficult when search many websites to find fashion clothes and accessories.

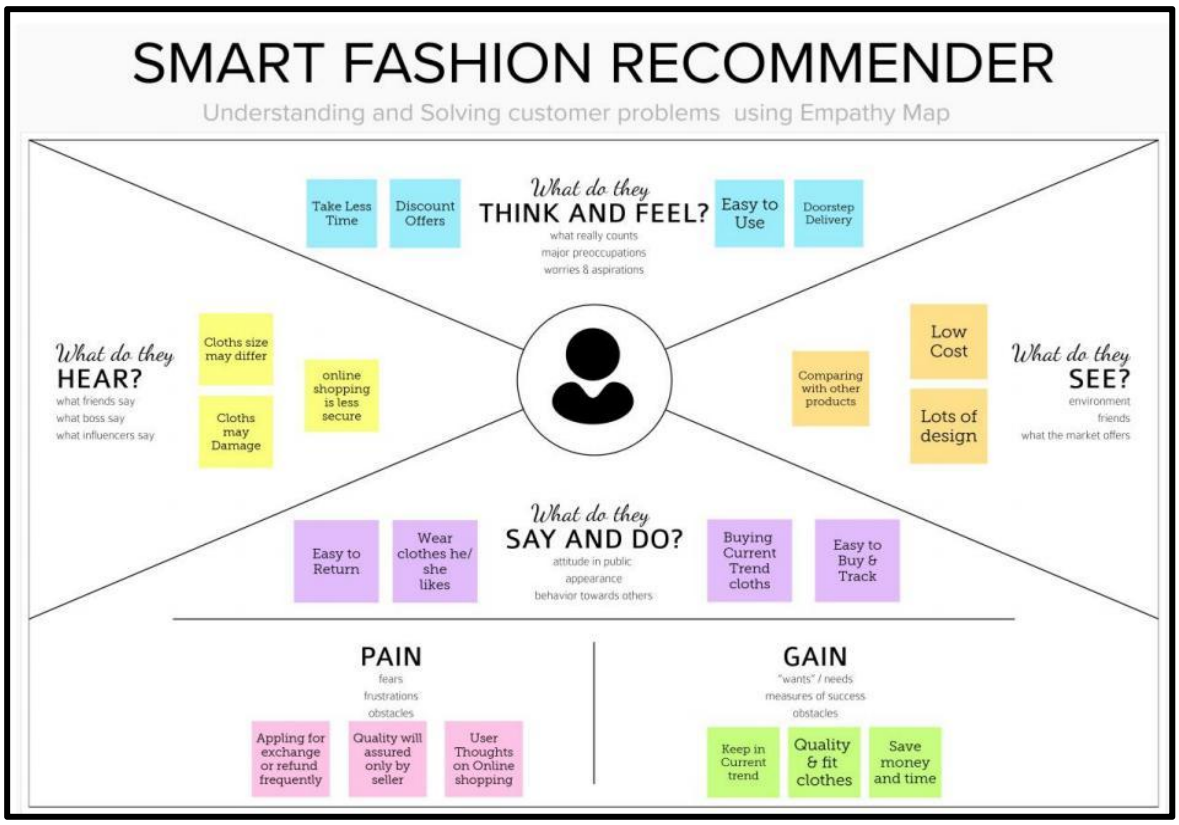
3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviour 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.

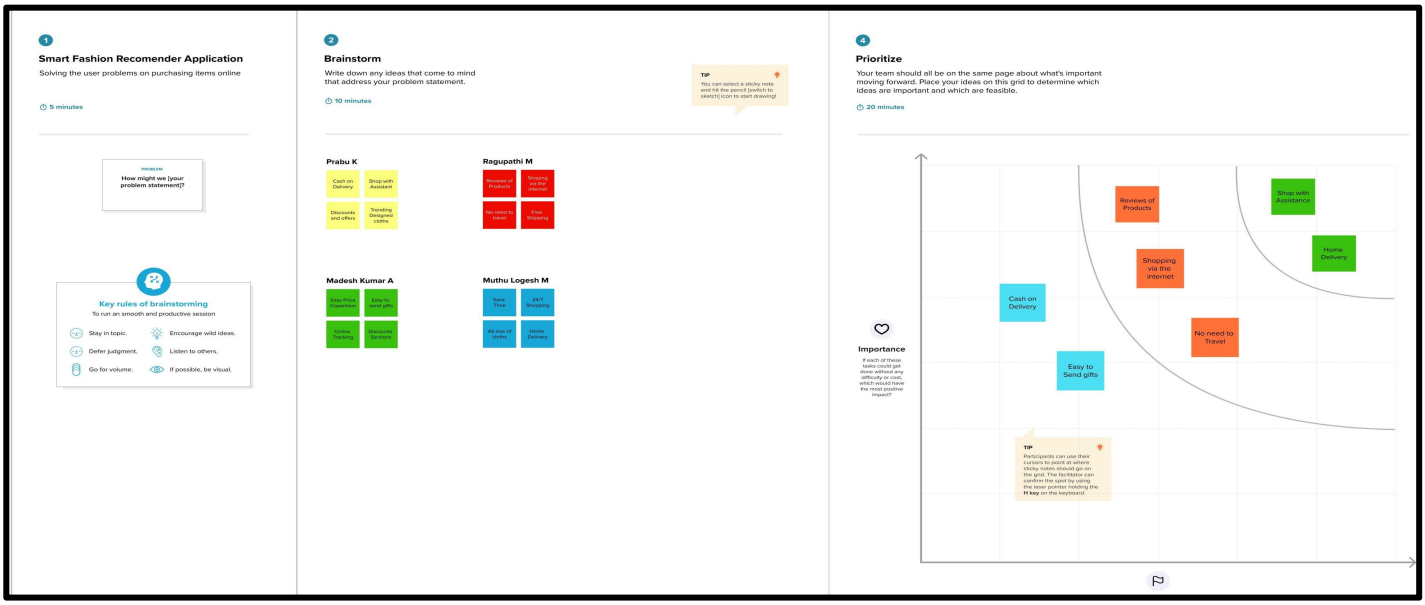


3.2 IDEATION AND BRAINSTORMING

3.2.1. Brainstorm & Idea Prioritization Template:

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.

3.2.2 Brainstorm, Idea Listing and Grouping:

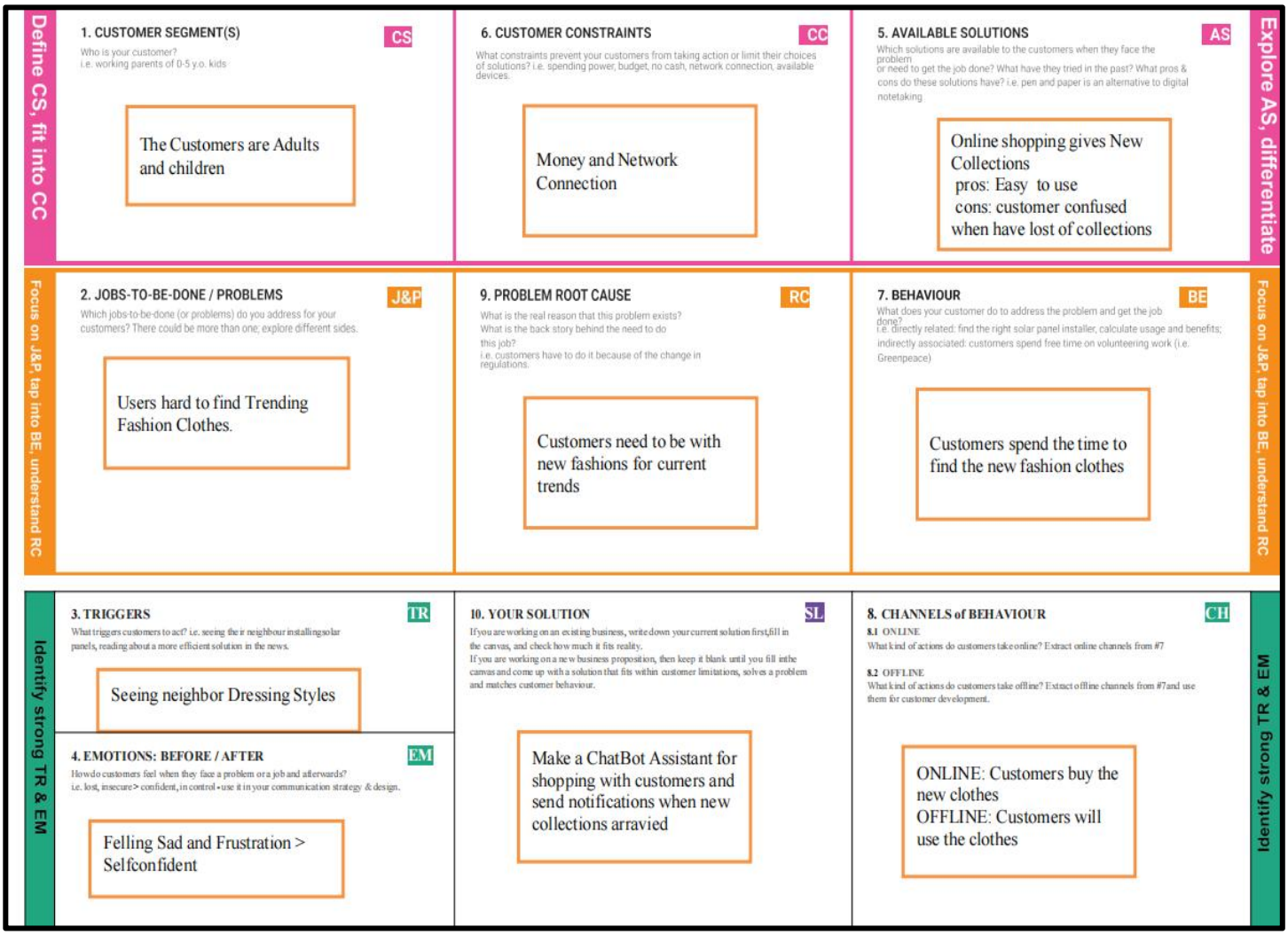


3.3 PROPOSED SOLUTION

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

S. No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	Customer feel difficult when search many websites to find fashion clothes and accessories.
2.	Idea / Solution description	Customer directly make online shoppingbased on customer choice without any search.
3.	Novelty / Uniqueness	The customer will talk to chat botregarding the products.
4.	Social Impact / Customer Satisfaction	The user friendly, interface form chat bot finding dress makes customer satisfied.
5.	Business Model (Revenue Model)	The chat bot sells our products tocustomer.
6.	Scalability of the Solution	We can easily scale our applicationby increases the items and products

3.4 PROBLEM SOLUTION FIT



4.REQUIREMENT ANALAYSIS

4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Interaction	Communication via Chat bot
FR-3	Buying Products	Through the chat Bot Recommendation
FR-4	Track Products	Ask the Chat Bot to Track my Orders
FR-5	Return Products	Through the chat Bot
FR-6	New Collections	Recommended from chat Bot

4.2 NON-FUNCTIONAL REQUIREMENTS

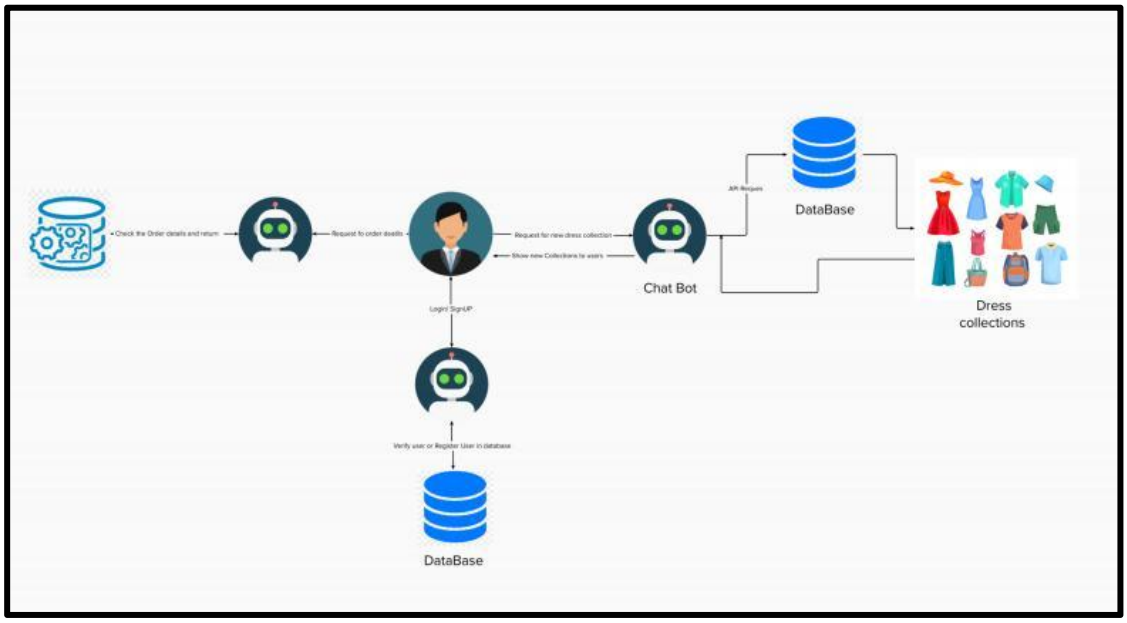
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Using Android or IOS or windows applications.
NFR-2	**Security**	The user data is stored securely in IBM cloud.
NFR-3	**Reliability**	The Quality of the services are trusted.
NFR-4	**Performance**	Its Provide smooth user experience.
NFR-5	**Availability**	The services are available for 24/7.
NFR-6	**Scalability**	Its easy to scale the size of users and products.

5. PROJECT DESIGN

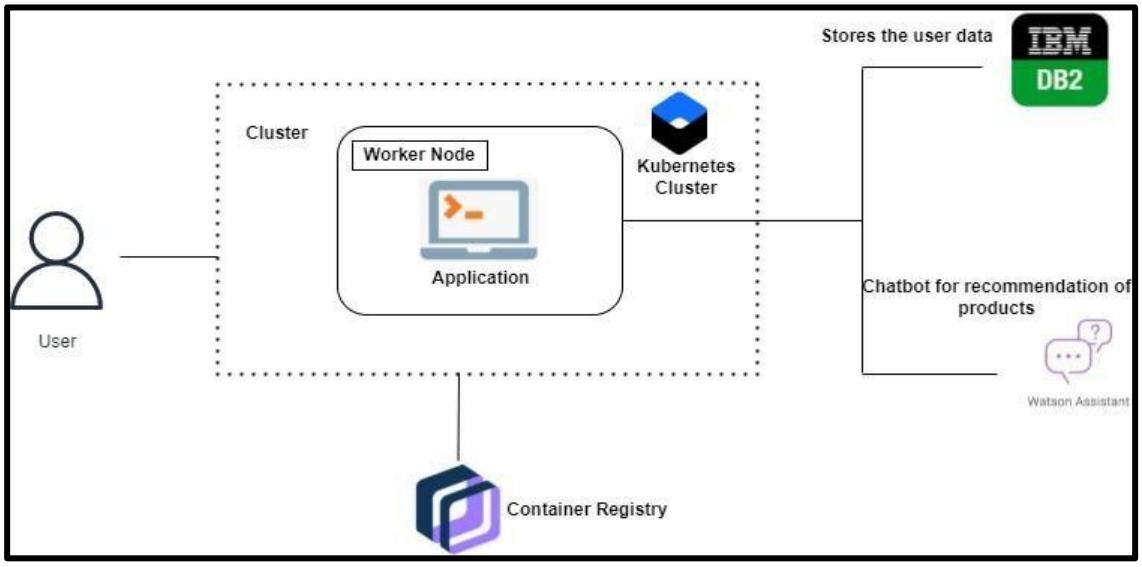
5.1 DATA FLOW DIAGRAMS

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.

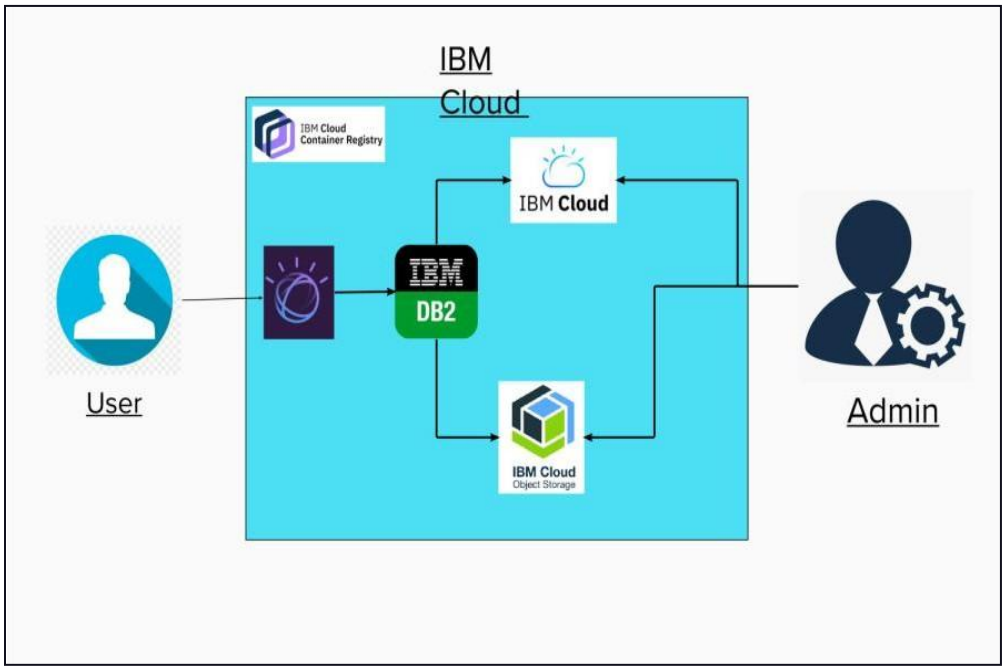


5.2 SOLUTION AND TECHNICAL ARCHITECTURE

5.2.1 SOLUTIONAL ARCHITECTURE



5.2.2 TECHNICAL ARCHITECTURE



5.3 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 confirmingmy password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation emailonce 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 applicationthrough Facebook	I can register & access the dashboard with Facebook login	Low	Sprint-2
		USN-4	As a user, I can register for the applicationthrough Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application byentering email & password		High	Sprint-1
	Dashboard	USN-6	As a user , I can view the dashboard and byproducts		High	Sprint-2
Customer (Webuser)	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	High	Sprint-1
Customer Care Executive	Contact of 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
Administrator	Check stock and Price , orders	USN-9	As a Administrator , I can Check the databaseAnd stock details and buying and selling prices	I am the administrator of company	High	Sprint-2

6.PROJECT PLANING AND SCHEDULING

6.1 SPRINT PLANING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Team Members
Sprint-1	User Panel	USN-1	The user will login into the website and go through the products available on the website	20	Prabu K Madesh Kumar A Ragupathi M Muthu Logesh M
Sprint-2	Admin panel	USN-2	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.	20	Prabu K Madesh Kumar A Ragupathi M Muthu Loges M
Sprint-3	Chat Bot	USN-3	The user can directly talk to Chat bot regarding the products. Get the recommendations based on information provided by the user.	20	Prabu K Madesh Kumar A Ragupathi M Muthu Logesh M
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	Prabu K Madesh Kumar A Ragupathi M Muthu Logesh M

6.2 SPRINT DELIVERY SCHEDULING

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

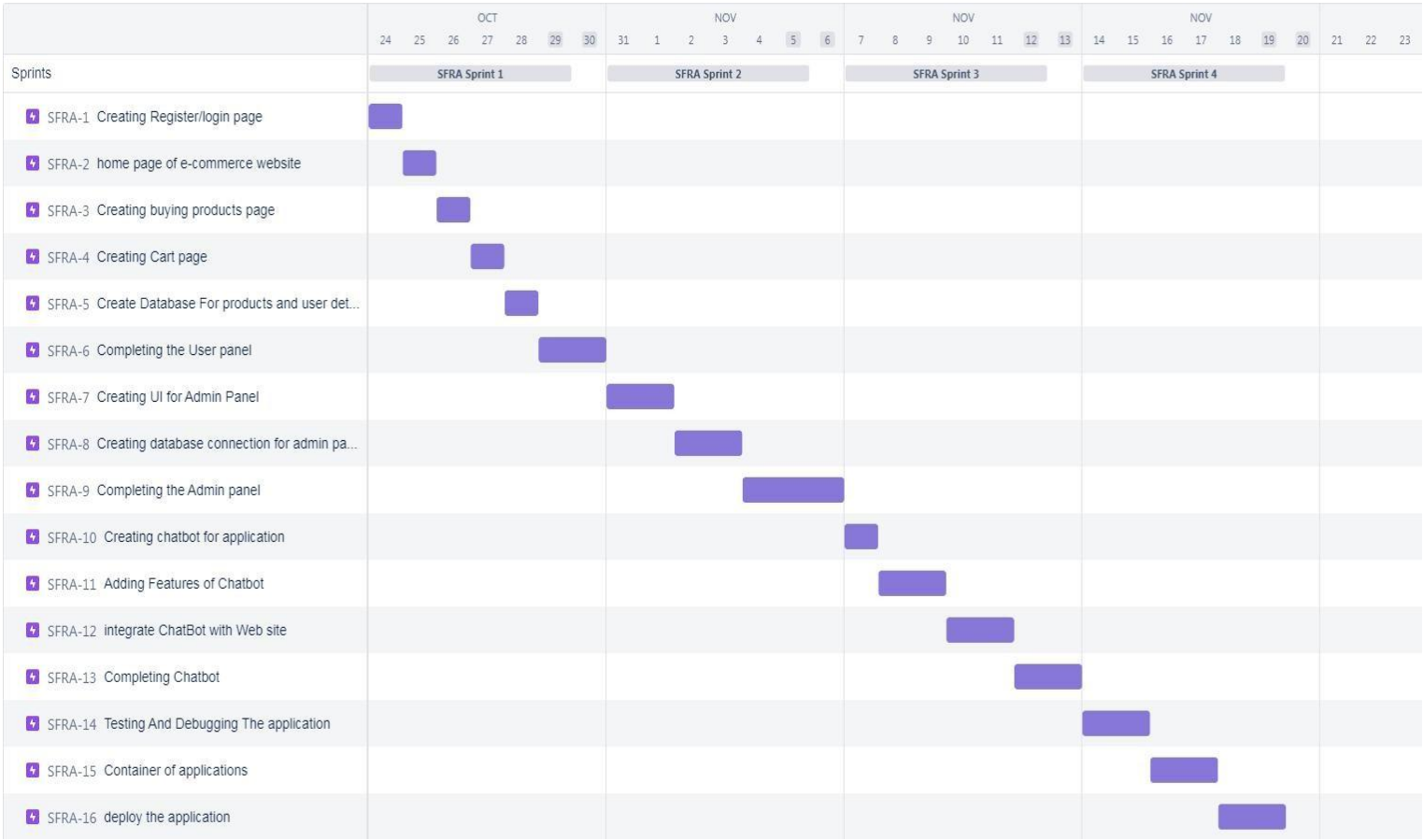
***Velocity*:**

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

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.3REPORTS FROM JIRA

The following image represents the reports from JIRA.



7. CODING & SOLUTIONING

Feature 1
Web application for fashion recommender

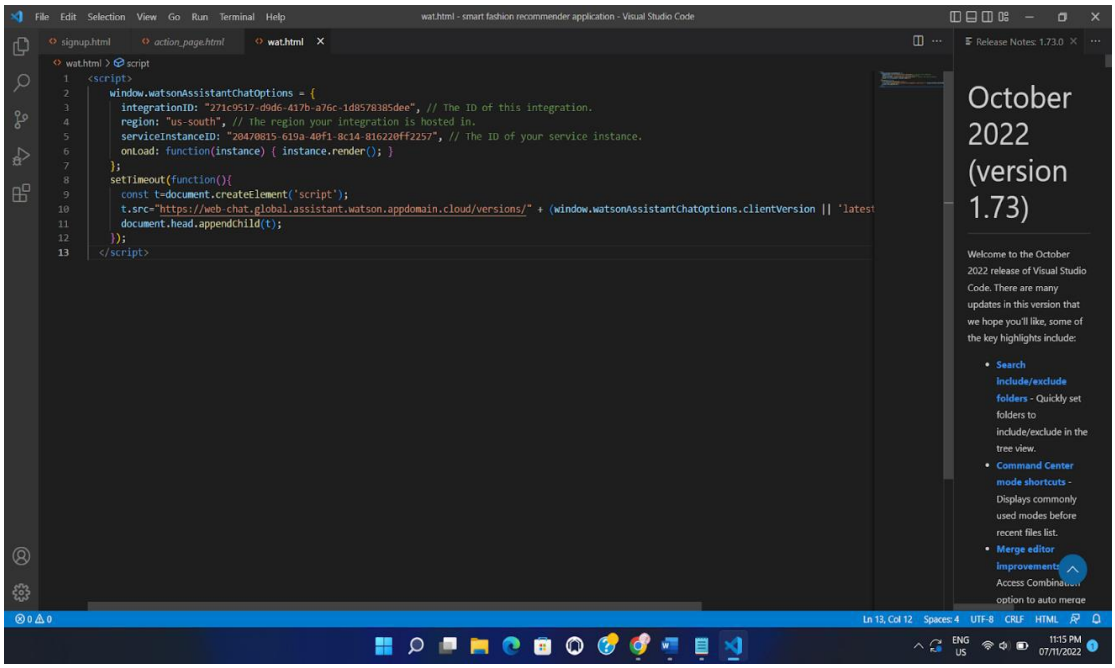
```
from flask import Flask, render_template, request, redirect, url_for, session
    import ibm_db
    import re

    app = Flask(__name__)

    app.secret_key = 'a'

    conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=9938aec0-8105-433e-
8bf90fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32459;Security
=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=dpk79343;PWD=29Jm7EbzOu
btoerk",",")

    @app.route('/')

    def homer():
      return render_template('index.html')


    @app.route('/Login',methods =['GET', 'POST'])
    def login():
      global userid
      msg = ''


      if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =? AND password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print (account)
        if account:
          session['loggedin'] = True
          session['id'] = account['USERNAME']
          userid=  account['USERNAME']
          session['username'] = account['USERNAME']
          msg = 'Logged in successfully !'

          msg = 'Logged in successfully !'
          return render_template('homepage.html', msg = msg)
        else:
          msg = 'Incorrect username / password !'
      return render_template('index.html', msg = msg)
```

```
@app.route('/Register', methods =['GET', 'POST'])
def registet():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO  users VALUES (?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('reg.html', msg = msg)

@app.route('/Homepage')
def dash():

    return render_template('homepage.html')

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

         qualification= request.form['qualification']
         skills = request.form['skills']
         jobs = request.form['s']
         sql = "SELECT * FROM users WHERE username =?"
         stmt = ibm_db.prepare(conn, sql)
         ibm_db.bind_param(stmt,1,username)
         ibm_db.execute(stmt)
         account = ibm_db.fetch_assoc(stmt)
         print(account)
         if account:
```

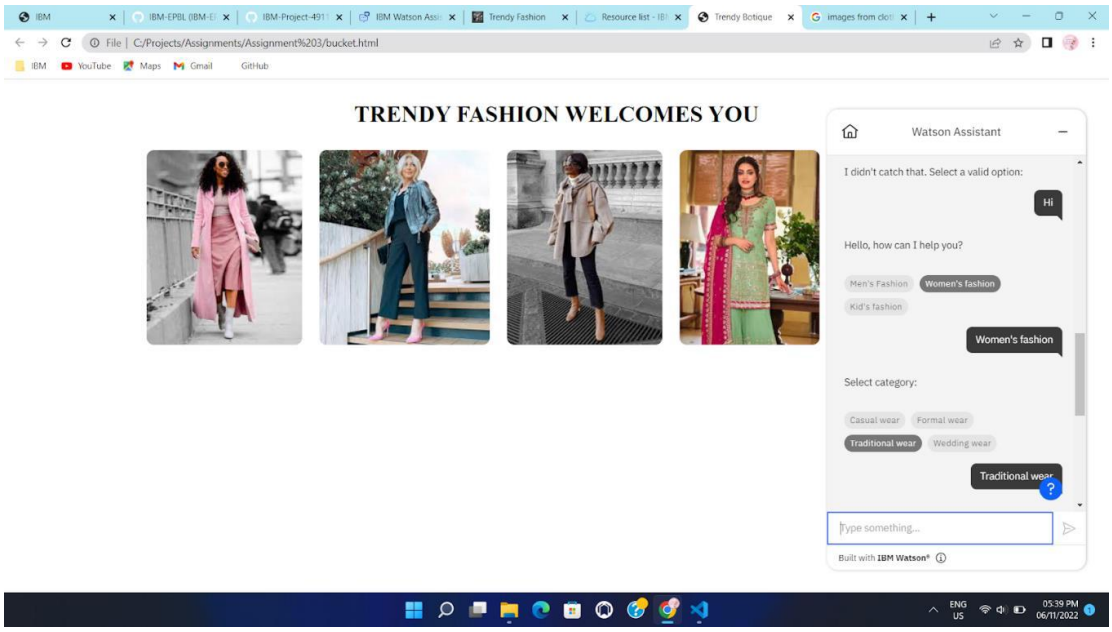
```
            msg = 'there is only 1 job position! for you'
            return render_template('apply.html', msg = msg)



        insert_sql = "INSERT INTO  job VALUES (?, ?, ?, ?, ?)"
        prep_stmt = ibm_db.prepare(conn, insert_sql)
        ibm_db.bind_param(prep_stmt, 1, username)
        ibm_db.bind_param(prep_stmt, 2, email)
        ibm_db.bind_param(prep_stmt, 3, qualification)
        ibm_db.bind_param(prep_stmt, 4, skills)
        ibm_db.bind_param(prep_stmt, 5, jobs)
        ibm_db.execute(prep_stmt)
        msg = 'You have successfully applied for job !'
        session['loggedin'] = True
        TEXT = "Hello,a new application for job position" +jobs+"is requested"




    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('apply.html', msg = msg)

@app.route('/display')
def display():
    print(session["username"],session['id'])

    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM job WHERE userid = % s', (session['id'],))
    account = cursor.fetchone()
    print("accountdislay",account)


    return render_template('display.html',account = account)

@app.route('/logout')

def logout():
   session.pop('loggedin', None)
   session.pop('id', None)
   session.pop('
if __name__ == '__main__':
   app.run(host='0.0.0.0')
username', None)
   return render_template('home.html')
```

Feature 2

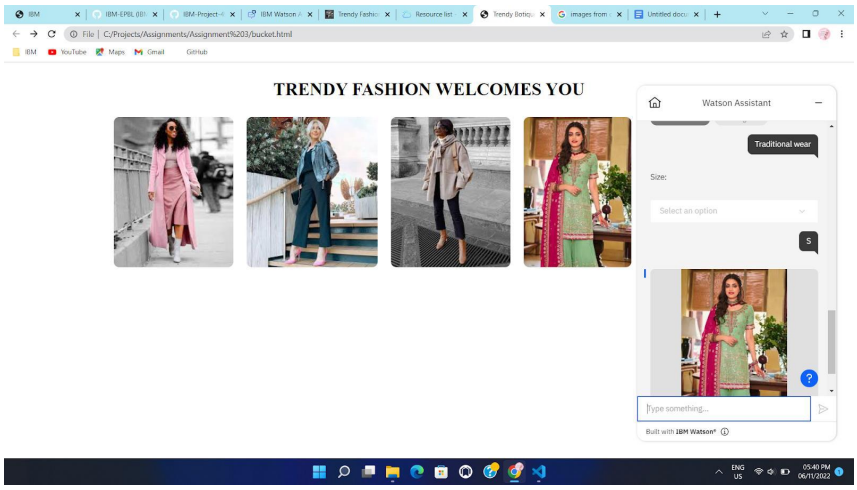
Chatbot

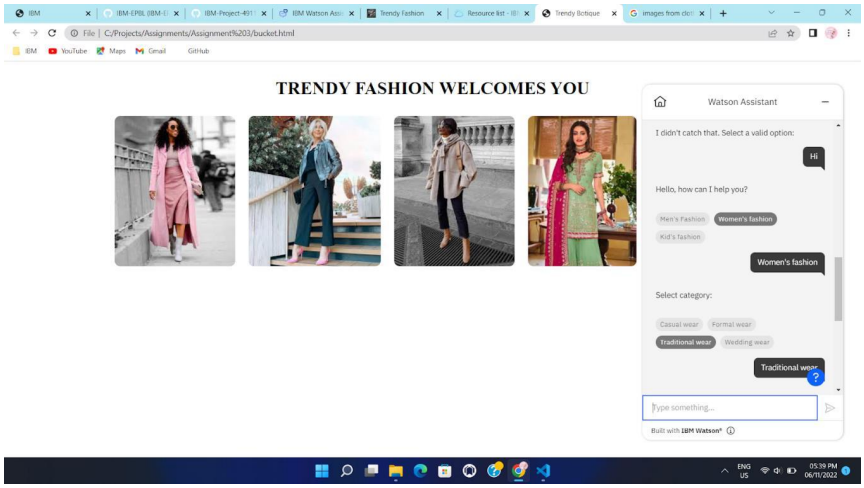


8. TESTING


TRENDY FASHION WELCOMES YOU
Watson Assistant
I didn't catch that. Select a valid option:
Hi
Hello, how can I help you?
Men's Fashion
Women's fashion
Kid's fashion
Women's fashion
Select category:
Casual wear
Formal wear
Traditional wear
Wedding wear
Traditional wear
Type something...
Built with IBM Watson®

9. RESULTS





Thus, a web application for smart fashion recommender is created and a chatbot (Watson Asssistant) is deployed on it and it helps customer feel easier when search many websites to find fashion clothes and accessories.

10. ADVANTAGES & DISADVANTAGES

Advantages of recommender system:

With continuing advances in artificial intelligence (AI), recommendations are no longer aimed at general audiences, or even those of a certain segment. Using deep learning-based recommendation engines, marketers today can target consumers with hyper-personalized recommendations at the individual level, based on metrics like persona, location, interests, real-time online behaviour and so on. This will not only allow marketers to drive online traffic through retargeting ads or email marketing, but also reduce customer irritation and churn rates.

Personalized product recommendations also help engage customers by serving them with products or services that are highly relevant to them. This will encourage higher average order values and increase conversions. In the long term, using personalized recommendations will show your customers that you understand and value them, increasing customer satisfaction and loyalty.

Disadvantages of recommender system:

1. Significant investments required

Recommendation engines are a big investment, not only financially, but in terms of time, too.

2. The complex onboarding process

Bringing a recommendation engine into your business can be a complex affair.

3. Lack of data analytics capability

Like all AI-based technologies, recommendation engines rely on data – if you do not have high-quality data, or cannot crunch and analyze it properly, you will not be able to make the most of the recommendation engine.

4. Inability to capture changes in user behavior

Consumers do not stand still – they are constantly behaving and evolving both as people and customers.

5. Privacy concerns

11.CONCLUSIONS

This paper presented an interdisciplinary, comprehensive review of what has been done in the field of conversational agents for fashion and retail e-commerce. Thus, this study adds to specialized literature in the field of chat bot design for fashion e-commerce by providing a comprehensive map of chat bot approaches that can be deployed by retailers. This study also revealed that despite the increasing investment in chat bots for e-commerce in general, research and use in fashion e-commerce are still relatively modest. Thus, further research and development on this topic are still needed.

12.FUTURE SCOPES

How might such recommender systems be utilized to make predictions regarding the fashion trends of the future? This aspect can be linked to popularity forecasting in the fashion domain since trendy items will likely be popular.

13.APPENDIX

Unlike other areas, fashion recommendations shouldn’t be based solely on personal taste and past activity of the customer. There are many external factors (many of which are emotional) that make creating a fashion recommendation system all the more complex. Public perceptions must be taken into account, as well as fashion rules, dress rules and current trends.

13.1 SOURCE CODE

Link - https://github.com/IBM-EPBL/IBM-Project-33396-1660219605/tree/main/Final%20Deliverables/Source%20Code

13.2 GITHUB AND PROJECT DEMO LINK

Link - https://github.com/IBM-EPBL/IBM-Project-33396-1660219605