PROJECT REPORT
SMART FASHION RECOMMENDER APPLICTION
TEAM ID:PNT2022TMID37061

TABLE OF CONTENTS

1.INTRODUCTION

1.1 PROJECT OVERVIEW:

Nowadays, fashion applications and e-commerce are growing more and more.And it also has some problems when finding the customer's wanted product in the web applications. Having a chatbot that understands the algorithm of a specific application can be of great aid. We are implementing such a chat bot in a web application, which is fed with the knowledge of the application's algorithm and helps the user completely from finding their needs to processing the payment and initiating delivery. It works as an advanced filter search that can bring the user what they want with the help of pictorial and named representation by getting simple user information and activities.

The application also has two main UI interactions: one is the user panel and the other one is the admin panel. Users can interact with the chat bot to search for products, order them from the manufacturer or distributor through chatbot AI, and it can also make payment transactions, track the delivery, and so on. The admin interface enables the user to upload products' details ,user details, orders and find how many products have been bought; supervise the stock availability; and interact with the buyer regarding the product reviews.

1.2 PURPOSE:

An online shopping system is a process in which people are being provided with the option of purchasing goods and services directly from the seller, all in a real-time environment.People in large number are doing online shopping today, and it is not only because it is convenient as one can shop from home, but also because there is an ample number of varieties available, with a high competition of prices, and also it is easy to navigate for searching regarding any particular item.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM:

This system is not much user-friendly as one needs to go to the market physically and then select items only from the available list. So mostly it is difficult to get the product as per our desire.Description About the products is less available and are mostly verbal only. For this typeof shopping, one needs to have an ample amount of free time.Also, not really good markets exist everywhere, so many times good markets become out of reach for certain people.

2.2 REFERENCES:

1.Barnard, M. Fashion as Communication, 2nd ed.; Routledge: London, UK, 2008.

2.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, 13, 376–386, doi:10.1080/17543266.2020.1829096.

3.Karmaker Santu, S.K.; Sondhi, P.; Zhai, C. On application of learning to rank for ecommerce search. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku Tokyo Japan, 7– 11 August 2017; pp. 475–484, doi:10.1145/3077136.3080838.

4.Garude, D.; Khopkar, A.; Dhake, M.; Laghane, S.; Maktum, T. Skin-tone and occasion oriented outfit recommendation system. SSRN Electron. J. 2019, doi:10.2139/ssrn.3368058.

5.Kang, W.-C.; Fang, C.; Wang, Z.; McAuley, J. Visually-aware fashion recommendation and design with generative image models. In Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM), New Orleans, LA, USA, 18–21 November 2017; pp. 207–216, doi:10.1109/ICDM.2017.30.

6.Sachdeva, H.; Pandey, S. Interactive Systems for Fashion Clothing Recommendation. In Emerging Technology in Modelling and Graphics; Mandal, J.K., Bhattacharya, D., Eds.; Springer: Singapore, 2020; Volume 937, pp. 287–294, doi: 10.1007/978-981-13- 7403-6_27.

7.Sun, G.-L.; Wu, X.; Peng, Q. Part-based clothing image annotation by visual neighbour retrieval. Neurocomputing 2016, 213, 115–124, doi:10.1016/j.neucom.2015.12.141.

8.Zhang, Y.; Caverlee, J. Instagrammers, Fashionistas, and Me: Recurrent Fashion Recommendation with Implicit Visual Influence. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019 pp. 1583–1592, doi:10.1145/3357384.3358042.

9.Matzen, K.; Bala, K.; Snavely, N. StreetStyle: Exploring world-wide clothing styles from millions of photos. arXiv 2017, arXiv:1706.01869.

10. Guan, C.; Qin, S.; Ling, W.; Ding, G. Apparel recommendation system evolution: An empirical review. Int. J. Cloth. Sci. Technol. 2016, 28, 854–879, doi: 10.1108/ijcst-09-2015-0100.

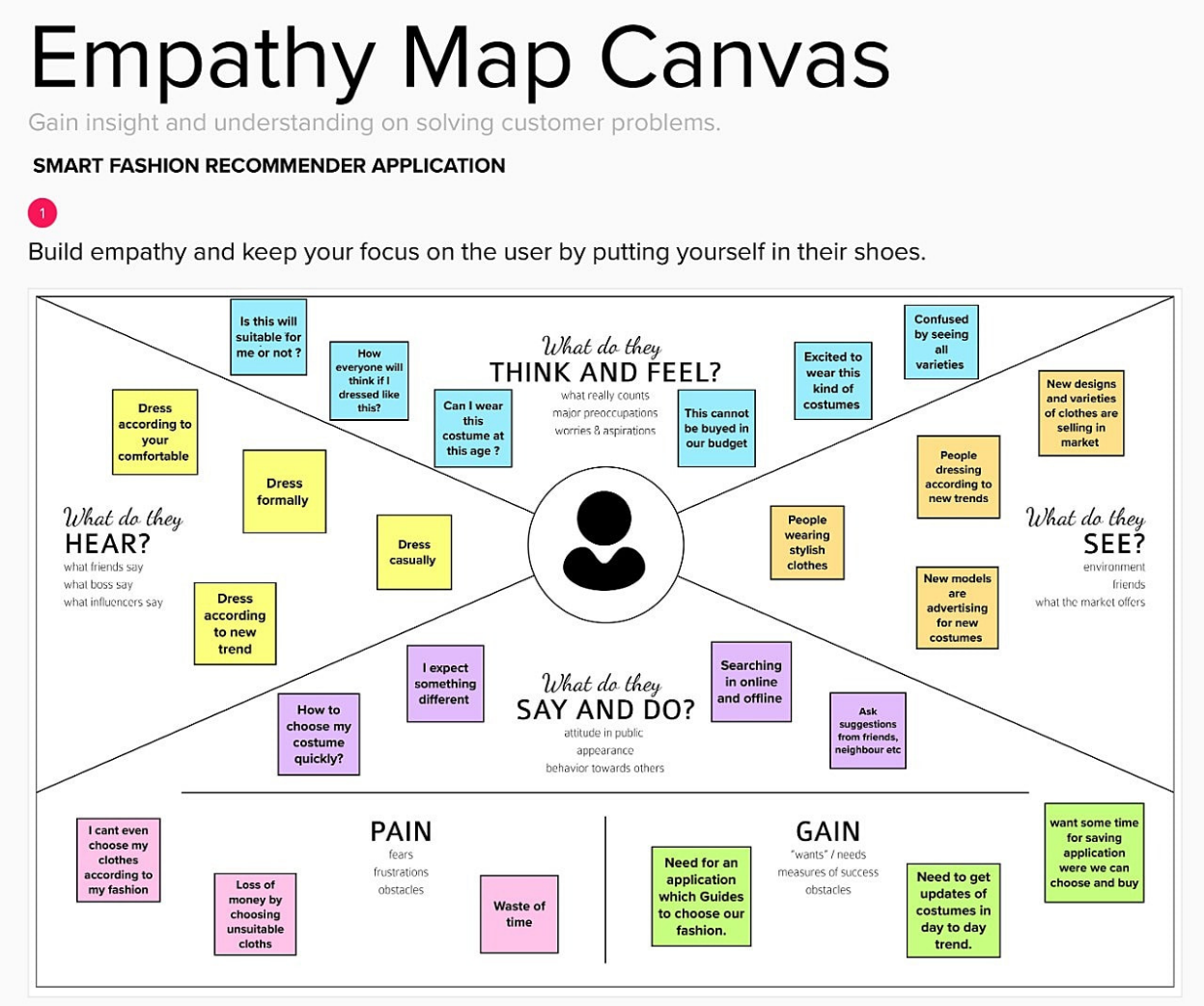
2.3 PROBLEM STATEMENT DEFINITION:

I am customer **I'm trying to** Get 24/7 service and Want Quality products with Time management **But** we cant get proper service, it tooks excess of time **Because** lack of resolving issues and Often changing of trends **Which makes me feel** irritating so we cant able to purchase online products with proper guidance and also having inconvenient service while ordering the products.

3.IDEATION &PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS:

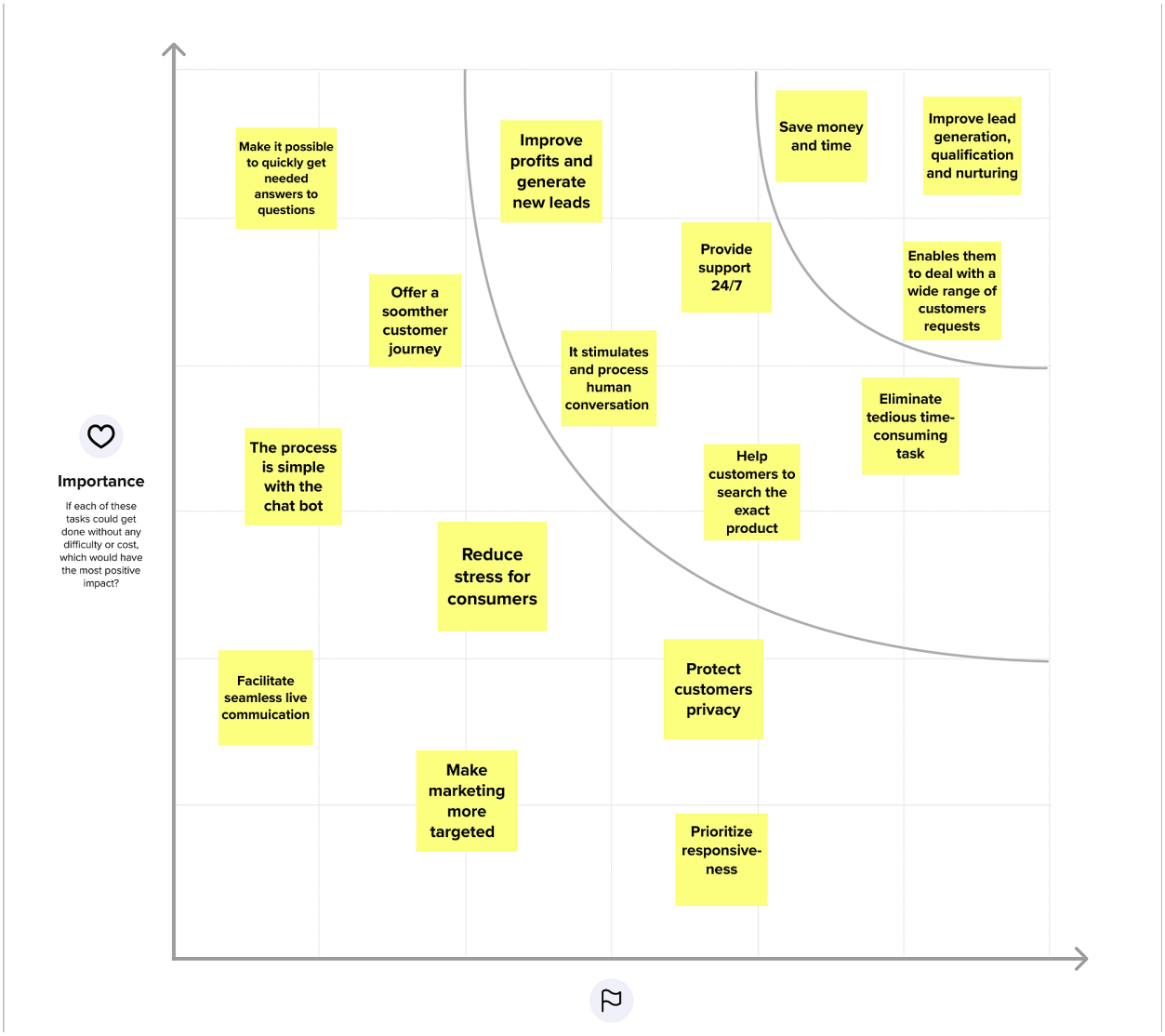
An empathy map is **a collaborative tool teams can use to gain a deeper insight into their customers**. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.



3.2 IDEATION & BRAINSTORMING:

Ideation is the **process of forming ideas from conception to implementation, most often in a business setting.** Ideation is expressed via graphical, written, or verbal methods, and arises from past or present knowledge, influences, opinions, experiences, and personal convictions.

Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between **ideation and brainstorming is that ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity**



3.3 PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	In normal Online Shopping: ● Poor customer service due to lack of service teams. ● There are many human error can occur. ● Negative feedback from customer ● There is no sufficient way to guide the customers properly. ● Lack of sales. ● Poor relationship between customer
2.	Idea / Solution description	● By using Smart fashion recommender application: ● Improve the effectiveness of customer service teams. ● Reduce the potential for human error. ● Collect candid and meaningful customer feedback. ● Guide customers along the path to purchase. ● Turns leads into sales (aka, boost conversion). ● Build stronger customer relationships
3.	Novelty / Uniqueness	● Improve your call/chat containment rate with the latest BERT-based natural language understanding (NLU) models that are capable of recognizing intent and context accurately and efficiently in more complex use cases.
4.	Social Impact / Customer Satisfaction	● Complaint handling might be one of the most critical factors to gain customer experience, from minor to big complaints. It is crucial to handle complaints as fast as possible to keep the customer satisfied.
5.	Business Model (Revenue Model)	● This application can be developed with minimum cost and at the same time it can provide high performance
6.	Scalability of the Solution	● This can be developed to a scalable product by using sensors and transmitting the data through Wireless Sensor Network and Analysing the data in cloud and operation is performed using chat bots.

3.4 PROBLEM SOLUTION FIT:

A. CUSTOMER SEGMENTS

- ➤ Demographic segmentation looks at single variables that make your customers who they are on a societal level and then groups them together

B. JOBS-TO-BE-DONE / PROBLEMS

- ➤ Message Interpreting.
- ➤ Machine to human transition.
- ➤ Personalization.
- ➤ Chatbot style.
- ➤ Data gathering

C. TRIGGERS

- ➤ Triggers are subscribers' actions or keywords sent in their messages.

D. EMOTIONS: BEFORE / AFTER

- ➤ Emotional chatbots are computer programs designed to simulate conversation with human users. Research shows that most customers still prefer human support over AI

E. AVAILABLE SOLUTIONS

- ➤ They help you get to know your customers.
- ➤ Customer service at any time.
- ➤ They help optimize costs.
- ➤ They improve customer satisfaction

F. CUSTOMER CONSTRAINTS

- ➤ Unclear scope of the chatbot and/or too broad purposes of its utilization.
- ➤ Lack of customer perspective in building the chatbot.

G. BEHAVIOUR

- ➤ Automate your website support.
- ➤ Chatbots handle refunds & exchange requests efficiently.
- ➤ Bots help in order confirmation & track shipping.
- ➤ Handle internal help desk.

H. CHANNELS OF BEHAVIOUR

- **ONLINE:** Chatbots use conversational marketing to engage customers.
- **OFFLINE:** They can quickly qualify leads before redirecting them to the appropriate next steps.

I. PROBLEM ROOT CAUSE

- Chatbots have limited responses.
- They're not often able to answer multi-part questions or questions that require decisions.
- This often means your customers are left without a solution, and have to go through more steps to contact your support team
- Hence increases initial installation cost unlike human beings.

J. SOLUTION

- At the most basic level, a chatbot is a computer program that simulates and processes human conversation.
- Allowing humans to interact with digital devices as if they were communicating with a real person.
- Automate interactions with prospects and customers, either on your website or in your app, for the purpose of generating sales.

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS:

FR No.	FR No. Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	1. Registration through Form 2. Registration through Gmail 3. Registration through LinkedIN
FR-2	User Confirmation	4. Confirmation via Email 5. Confirmation via OTP
FR-3	User Login	6. Login through email and password
FR-4	User purchase	7. They can search and buy any particular product through chatbot .

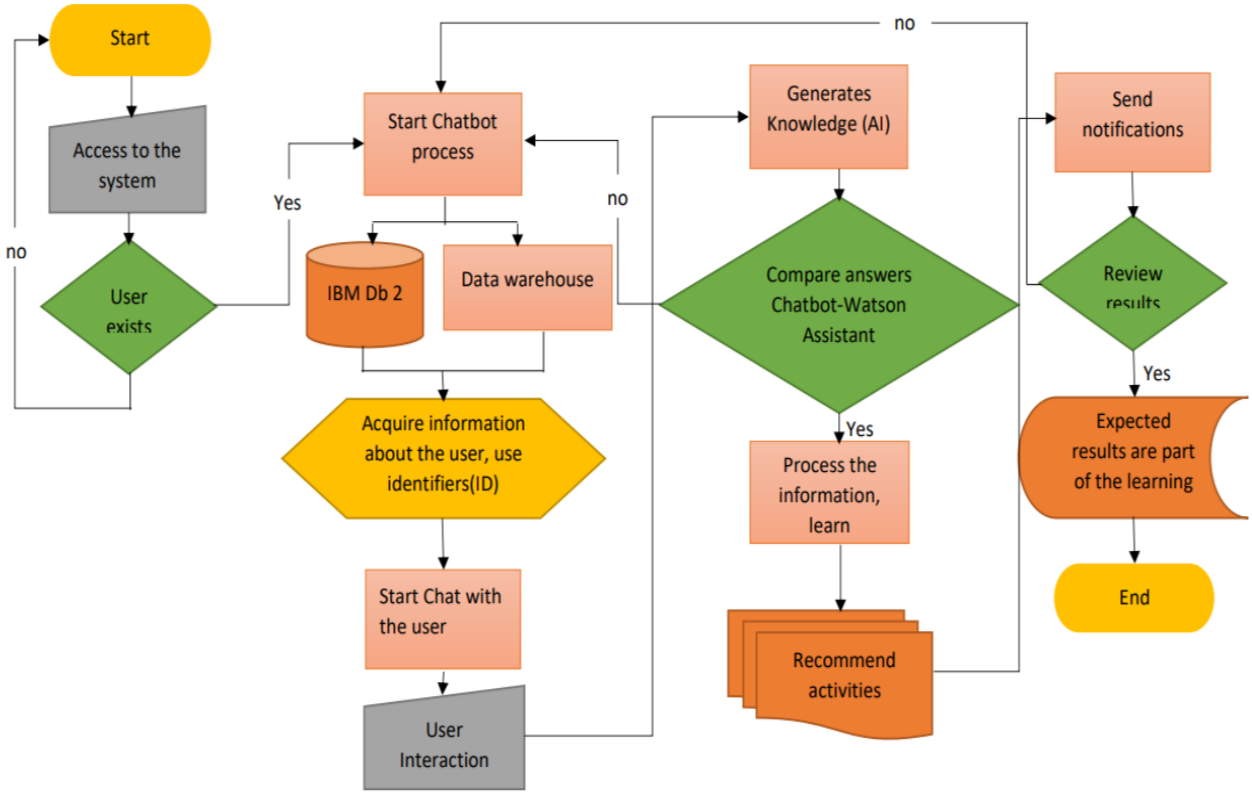
4.2 NON-FUNCTIONAL REQUIREMENTS:

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Can automate tasks performed frequently and at specific times.
NFR-2	**Security**	Authentication, with which chatbots verify the user identity and authorisation, which grants the user access to information or function
NFR-3	**Reliability**	87 percent of customers find chatbots effective or very effective at resolving their issues
NFR-4	**Performance**	Make more money, save time, shorten sales cycles, boost conversion, and aid in crossplatform performance
NFR-5	**Availability**	They are available for 24/7.
NFR-6	**Scalability**	The ability to meet consumer demands at scale is what makes customer service chatbots so successful for businesses

5.PROJECT DESIGN

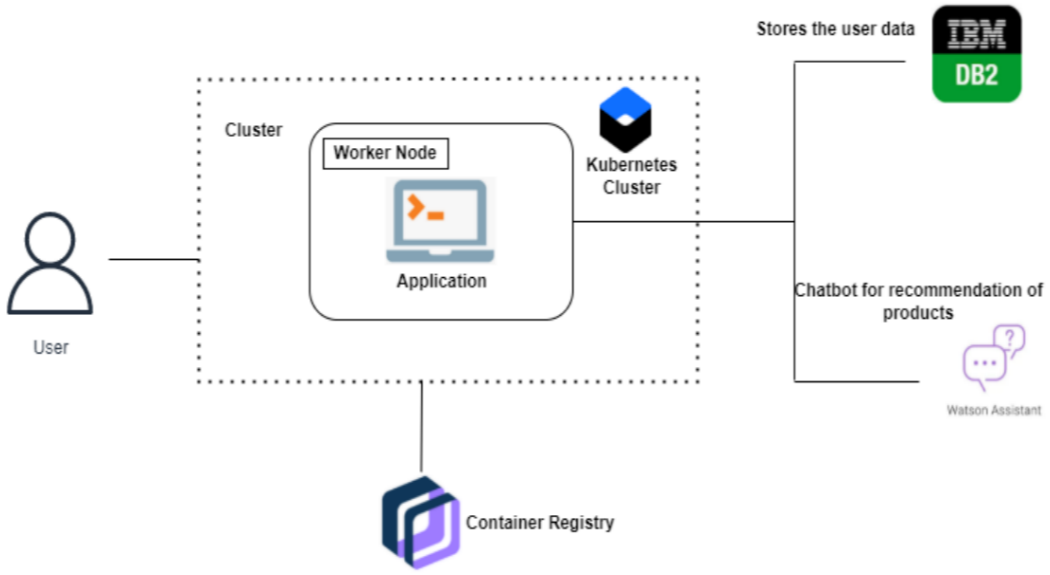
5.1 DATAFLOW DIAGRAMS:

A data-flow diagram is a way of representing a flow of data through a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops



5.2 SOLUTION &TECHNICAL ARCHITECTURE:

Based on the complexity of the deployment, a solution architecture diagram may actually be a **set of diagrams documenting various levels of the architecture**. The diagram relates the information that you gather on the environment to both physical and logical choices for your architecture in an easily understood manner.



5.3 USER STORIES:

1. CUSTOMER (MOBILE USER):

- ➤ As a user, I can register for the application by entering my username, email and password
- ➤ As a user, I will receive confirmation email once I have registered for the application
- ➤ As a user, I can log into the application by entering email and password

2. CUSTOMER (DISCOUNT SEEKERS):

- ➤ As a user, I shop around searching multiple sites or stores trying to find best price
- ➤ As a user, I will always base my buying decision on how much I can save

3. CUSTOMER(IMPULSE BUYERS):

- ➤ As a user, I buy dress and costumes without planning in advance

4. CUSTOMER(LOYAL):

- ➤ As a user, I purchase willingly and repeatedly

6.PROJECT PLANNING AND SCHEDULING

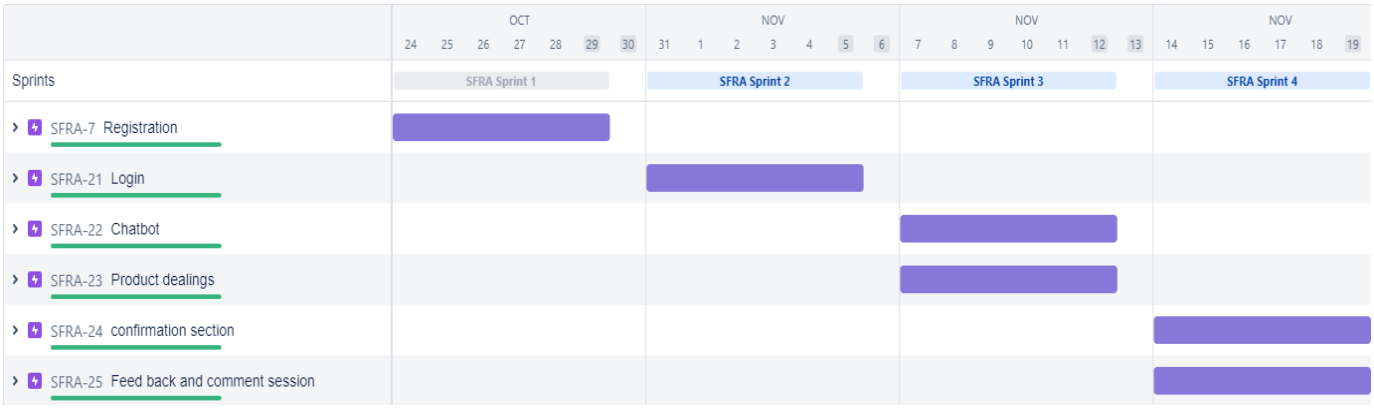
6.1 SPRINT PLANNING & ESTIMATION:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	2	High	V.Mohana Priya V.Sangeetha S.Sindhu
Sprint-2	Login	USN-2	As a user I can login into the application by entering email &password.	2	Medium	S.Sindhu V.Mohana Priya
Sprint-3	Chatbot	USN-3	As a user, I can contact with chatbot for selecting my products	3	High	V.Sangeetha G.Harish
Sprint-3	Product dealings	USN-4	As a user, I receive recommendation with best deals and offers from chatbot	2	Medium	V.Sangeetha V.Mohana Priya G.Harish
Sprint-4	Confirmation section	USN-5	As a user, I received the confirmation message from the chatbot	3	High	V.Sangeetha V.Mohana Priya
Sprint-4	Feedback & comment section.	USN-6	As a user I can write a fashion review as both positive and negative.	2	High	S.Sindhu G.Harish

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	9	6 Days	24 Oct 2022	29 Oct 2022	9	29 Oct 2022
Sprint-2	5	6 Days	31 Oct 2022	05 Nov 2022	5	06 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	8	6 Days	14 Nov 2022	19 Nov 2022	8	19 Nov 2022

6.3 REPORT FROM JIRA:



7.CODING AND SOLUTIONING

7.1 FEATURE:

Features of Chatbot:

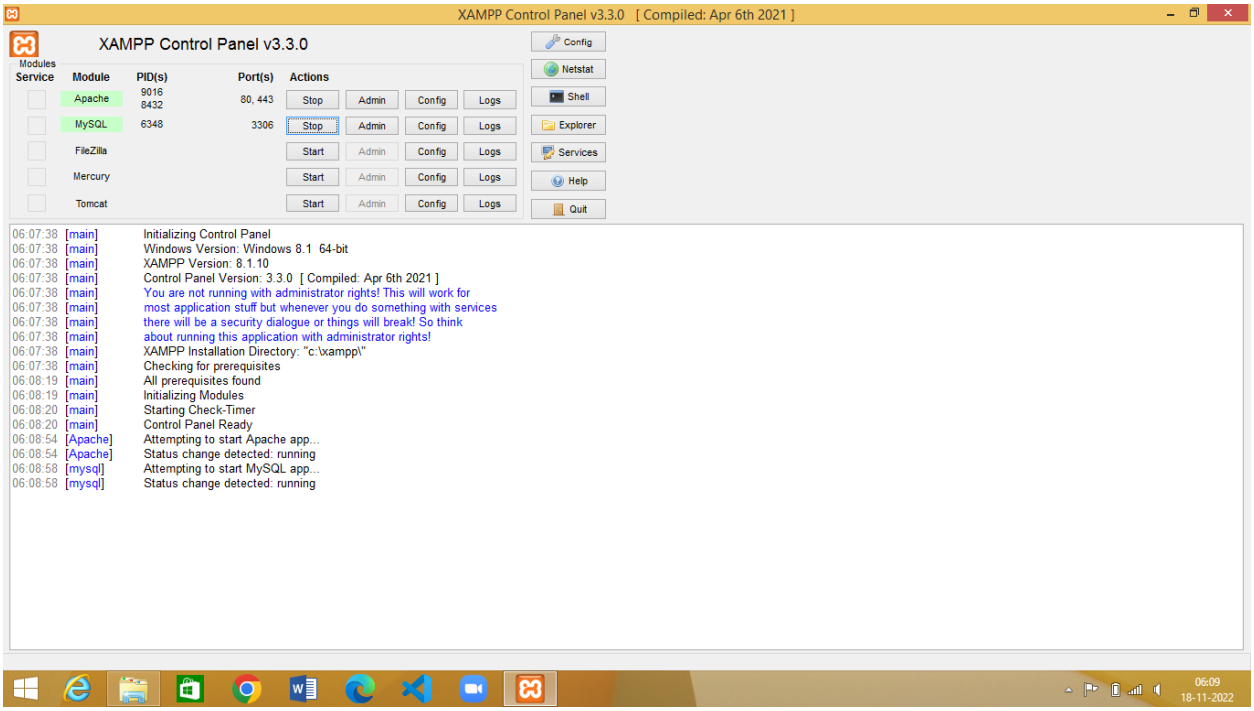
- Using chatbot we can manage user's choices and orders.
- The chatbot can give recommendations to the users based on their interests.
- It can promote the best deals and offers on that day.
- It will store the customer's details and orders in the database.
- The chatbot will send a notification to customers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

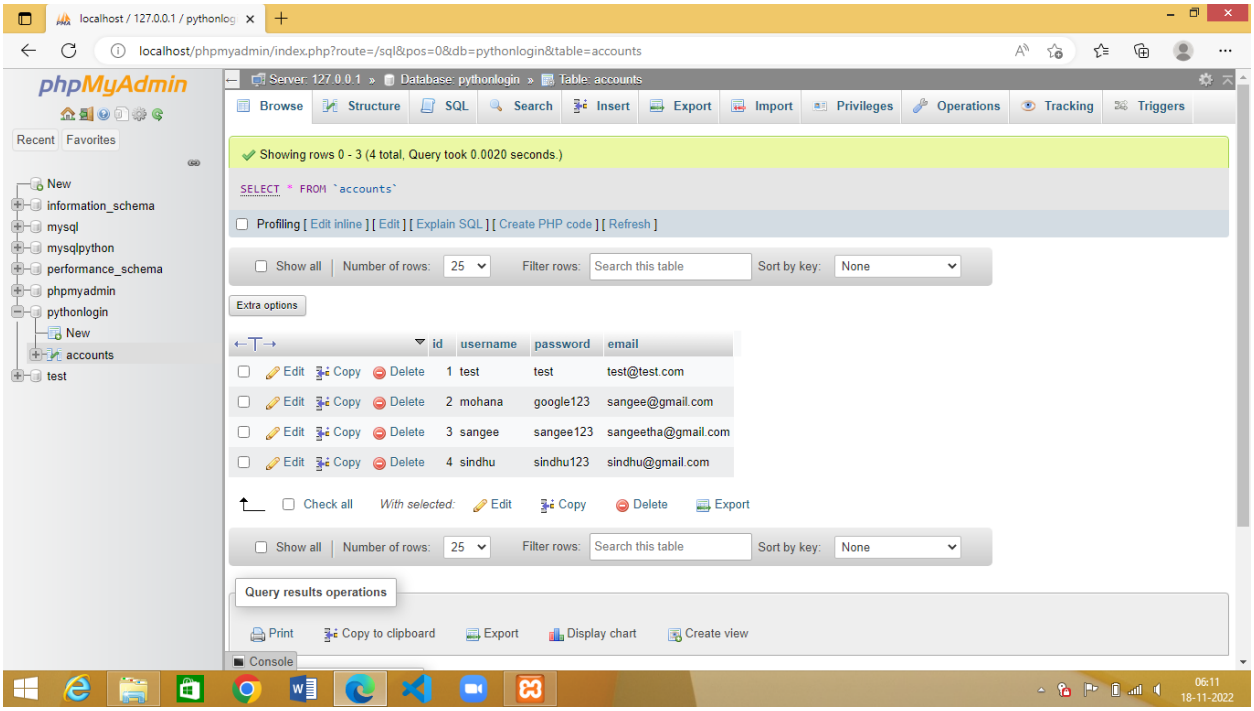
Codes used to connect chatbot:

```
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "b3f46917-ef71-4bbd-8ba9-4290d403e588", // The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "36ba6bd5-0f0a-4e4e-b698-91de6fdf83c8", // 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>
```

7.2 DATABASE SCHEMA:

The database used in this project is **xammp**





8.TESTING

8.1 TESTCASES:

A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly.A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario.

Test case ID	Feature Type	Component	Test Scenario
RegisterPage_TC_OO1	Functional	Register Page	Verify the details in register page and move the details to the database.
LoginPage_TC_OO2	Functional	Login Page	Verify the UI elements in Login
Dashboard_TC_OO3	Functional	Home Page	User can view the user dashboard
Product_TC_OO4	Functional	Product Page	Verified user can view the Product details.
Chatbot_TC_OO5	Functional	Product Page	Chatbot will appear after pressing the products

8.2 USER ACCEPTANCE TESTING:

Sr. No.	Security test cases	Type- Negative/ Positive Test Case
1	Verify if a user cannot enter the characters more than the specified range in each field (Username and Password).	Negative
2	Verify if a user cannot enter the characters more than the specified range in each field (Username and Password).	Positive
3	Verify the login page by pressing 'Back button' of the browser. It should not allow you to enter into the system once you log out.	Negative
4	Verify the timeout functionality of the login session.	Positive
5	Verify if a user should not be allowed to log in with different credentials from the same browser at the same time.	Negative
6	Verify if a user should be able to login with the same credentials in different browsers at the same time.	Positive
7	Verify the Login page against SQL injection attack.	Negative
8	Verify the implementation of SSL certificate.	Positive

9.RESULTS

9.1 PERFORMANCE METRICS:

S.No	Parameter	Performance
1.	Response Time	0.2s (Average of 10 trails)
2.	Workload	500 users (Calculated based on Cloud Space)
3.	Revenue	Individual users and pharmaceutical industries.
4.	Efficirncy	Simple and straightforward workflow, which makes the process efficient.
5.	Down Time	Almost no down time due to IBM CLoud enabled solution.

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- For customers, recommender systems can help them find items which they are interested in. For enterprises, recommender systems can improve the loyalty of their customers by enhancing the user experience and further convert more browsers to consumers.

- chatbots are bringing about a transformation and enabling retailers to foster better shopping experiences.

- The rise of AI chatbots in the business world is enormous.

- **24×7 Support:** Most customers expect businesses to be available 24 hours a day, 7 days a week. While having a customer service team around-the-clock is an (expensive) option, with chatbots you can eliminate that cost and still ensure your customers are catered to immediately by chatbots – irrespective of what time of the day it is. Offering 24-hour support is a great way to ensure customer satisfaction.

- **Personalization:** Chatbots can also be used to collect data about your visitors and leverage it to make better product suggestions and recommendations. Understanding customer inquiries, their needs, and preferences can allow you to personalize product pages and build customer loyalty and affinity. In addition, chatbots can also notify customers when items are out of stock, and suggest appropriate alternate products based on their preferences while informing them about their expected delivery date and time.

- **Reduced Costs:** Having chatbots do most (or probably all) of your customer service activities can help save a substantial amount of money on customer service. Efficient customer assistance by chatbots requires less human support, allowing you to drive your focus on more critical aspects of your e-commerce site, such as page layout or checkout. You can also dramatically reduce human error and enable efficient customer service with minimal resource costs.

- **Product Guidance:**Very often, e-commerce visitors get lost in the maze of millions of products. Chatbots can help such customers find the exact product they are looking for in a huge catalog and directly jump to the checkout page, or obtain information on current sales. By providing answers or advice to specific customer inquiries, chatbots can guide clients and enable them to make purchases on the fly. For example, eBay's ShopBot guides customers through their products, asks questions to understand their needs, and offers recommendations like a real sales associate.

- **Cart Recovery:** Contrary to popular belief, an abandoned cart can also be a great source of revenue. Chatbots can remind users of items in their abandoned shopping cart and ask them if they are willing to proceed towards checkout or if they would like to clean their cart. On most occasions, such reminders push customers to revisit their cart and enable them to purchase some if not all of the items in their cart.

DISADVANTAGES:

- **Not identifying the customer's use case:** A chatbot may not be a one-size-fits-all solution for every business. Businesses often make the mistake of choosing the wrong type of chatbot without considering their needs and use case. In these cases, the chatbot does not prove to be advantageous to the business, causes repeated dead-ends for customers, and results in hemorrhaged money.

- **Not understanding customer emotion and intent:** It is as important to express empathy via conversational AI to customers as it is to solve their problems. Users may be approaching the chatbot in a frustrated state, so when the chatbot fails to understand the customer queries, the situation is bound to get worse.

- **The chatbot lacks transparency:** Bot failure and customer frustration are often brought about by not making it clear to a customer that they are interacting with a bot and not a human agent. This can skew customer expectations and cause a poor user experience.

- **When customers prefer human agents:** Chatbots are making great advancements and we are more and more likely to use them. However, most of us still feel most comfortable with human agents over a bot. 54% of US customers choose to communicate more with human customer support agents than with chatbots. You can fix it by redirecting customers to a human agent when required.

- **Not able to address personalized customer issues:** Chatbots are mostly trained to answer customer FAQs and function based on what information they have been provided using artificial intelligence (AI) and ML. But they are often at a loss when it comes to resolving specific personalized queries

- **Lacking data collection and analysis functions:** One major reason why chatbots fail is that we forget to regularly analyze them to improve their functioning. The work is not done once it's launched

- **Not aligning with the brand :** Often, the chatbot's look and conversational personality do not match that of the brand. This can make the customers feel disconnected from the business and ultimately lead to a bad customer experience.

11.CONCLUSION

Chatbots are, first and foremost, customer self-service solutions. This basically means that they are there to help customers find information, learn about products and services, and get answers to simple questions on the fly.With a chatbot, your organization can easily offer high-quality support and conflict resolution any time of day, and for a large quantity of customers simultaneously.The use of chatbots can help businesses maintain a great level of consistency in answers and improve customer experience with the brand.

12.FUTURE SCOPE

Chatbots are changing the way businesses communicate and understand their customers. With AI, chatbots will have the ability to deliver a more personalized customer experience. It's also saving companies money through customer service, internal processes, and marketing efforts.

13.APPENDIX

SOURCE CODE:

app.py

```
from flask import Flask,render_template,request,session,url_for,redirect
from flask_mysqldb import MySQL
import MySQLdb.cursors
import mysql.connector
import re
app = Flask(__name__)

@app.route("/")
def home_page():
   return render_template("index.html")
@app.route("/website_user")
def website_user():
   return render_template("website_user.html")
@app.route("/register")
def register_user():
   return render_template("register_user.html")

app.secret_key = "d5fb8c4fa8bd46638dadc4e751e0d68d"
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = ''
app.config['MYSQL_DB'] ='pythonlogin'
mysql = MySQL(app)

@app.route('/login', methods=['GET', 'POST'])
def login():
   # Output message if something goes wrong...
   msg = ''
   # Check if "username" and "password" POST requests exist (user submitted form)
   if request.method == 'POST' and 'username' in request.form and 'password' in
request.form:
      username = request.form['username']
```

```
        password = request.form['password']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM accounts WHERE username = %s AND password =
%s', (username, password,))
        account = cursor.fetchone()
        if account:
            session['loggedin'] = True
            session['id'] = account['id']
            session['username'] = account['username']
            # Redirect to home page
            return redirect(url_for('website_user'))
        else:
            msg = 'Incorrect username/password!'
    return render_template('login.html', msg=msg)

@app.route('/register', methods=['GET', 'POST'])
def register():
    msg = ''
    if request.method == 'POST' and 'username' in request.form and 'password' in
request.form and 'email' in request.form:
        # Create variables for easy access
        username = request.form['username']
        password = request.form['password']
        email = request.form['email']
            # Check if account exists using MySQL
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM accounts WHERE username = %s', (username,))
        account = cursor.fetchone()
        # If account exists show error and validation checks
        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 = 'Username must contain only characters and numbers!'
        elif not username or not password or not email:
            msg = 'Please fill out the form!'
```

```
        else:
            cursor.execute('INSERT INTO accounts VALUES (NULL, %s, %s, %s)', (username,
password, email,))
            mysql.connection.commit()
            msg = 'You have successfully registered!'
    elif request.method == 'POST':
        msg = 'Please fill out the form!'
    return render_template('register_user.html', msg=msg)

@app.route("/frame")
def frame():
    return render_template("frame.html")
@app.route("/product")
def product():
    return render_template("product.html")
@app.route("/content")
def content():
    return render_template("content.html")
# womens
@app.route("/churidars")
def churidars_dress():
    return render_template("churidars.html")
@app.route("/sarees")
def sarees_dress():
    return render_template("sarees.html")
@app.route("/western")
def western_dress():
    return render_template("western.html")
# Mens
@app.route("/shorts")
def shorts_dress():
    return render_template("shorts.html")
@app.route("/t-shirts")
def tshirts_dress():
    return render_template("t-shirts.html")
@app.route("/jeans")
def jeans_dress():
```

```
    return render_template("jeans.html")
@app.route("/jackets")
def jacketss_dress():
    return render_template("jackets.html")
@app.route("/Pants")
def pants_dress():
    return render_template("Pants.html")
@app.route("/coarts")
def coats_dress():
    return render_template("coarts.html")
# kids dress
@app.route("/kids_frocks")
def kids_frocks_dress():
    return render_template("kids_frocks.html")
@app.route("/boys_shirts")
def boys_shirts_dress():
    return render_template("boys_shirts.html")
@app.route("/boys_pants")
def boys_pants_dress():
    return render_template("boys_pants.html")
if __name__=="__main__":
    app.run(debug=True)
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-40349-1660628507

PROJECT DEMO VIDEO LINK:

https://www.youtube.com/embed/OQdrsoxOIEw