SMART FASHION RECOMMENDER APPLICATION

IBM – NALAIYA THIRAN PROJECT

INDUSTRY MENTOR **:** Krishna Chaitanya

FACULTY MENTOR **:** Arun Raj V

A PROJECT REPORT

Submitted by

TEAM ID	PNT2022TMID18407
TEAM LEAD	S . V . Sree Priya Dharshni
TEAM MEMBER	C . Akshaya
TEAM MEMBER	P . Bhavadharani
TEAM MEMBER	P . Kalavathi

In partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

MEPCO SCHLENK ENGINEERING COLLEGE,SIVAKASI

TABLE OF CONTENTS

ABSTRACT

Creating an outfit is a problem that is based on the preferences of each person and it can be difficult even for the best experts. Most previous work on outfit recommendation focuses on designing visual features to enhance recommendations. Existing work neglects user comments of fashion items, which have been proven to be effective in generating explanations along with better recommendation results. 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, a recommendation system is required to sort, order, and efficiently convey relevant product content or information to users. Image-based fashion recommendation systems (FRSs) have attracted a huge amount of attention from fast fashion retailers as they provide a personalized shopping experience to consumers.

The rapid progress of computer vision, cloud computing and artificial intelligence combined with the current growing urge for online shopping systems opened an excellent opportunity for the fashion industry. As a result, many studies worldwide are dedicated to modern fashion related applications such as virtual try-on and fashion synthesis. However, the accelerated evolution speed of the field makes it hard to track these many research branches in a structured framework. Such hierarchical application based multi-label classification of studies increases the visibility of current research, promotes the field, provides research directions, and facilitates access to related studies.

1 INTRODUCTION

1.1.PROJECT OVERVIEW

During the last few years, online shopping has been growing. In 2013, the total turnover for 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 easily discover applicable products or 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.

1.2.PURPOSE

Recommendaton systems make Recommendations based on the informaton they are provided with and in the manner in which they are programmed. Going into details, most of the evaluation applied is independent coming up with a brand-new recommendation algorithm, system, or model. However, different researchers use already existing work as researchers use an already existing current piece of work to come up with a new diagram or to truly improve the current one. The present analysis model focuses on the use of a current algorithmic program and, consequently, the use of a new research concept comes up with a recommender system. Existing researchand fashions have given us some inspirations of how to design fashion recommendation systems. Nevertheless, they also involve some common drawbacks. Therefore, in this study, our aim to suggest a new method to assist personal choice making through supplying images and get suggestions based on provided contents

The contribution of the research are follows:

- To design and implement a web-based clothing fashion style recommender system based on deep learning;
- A scheme for improving a person’s clothing style by removing thefeatures he/she doesn’t like.
- These attributes served to a similar model to retrieve similar images as Recommendations.
- Combined with more common content-based recommendation systems, our model can help to extend robustness and performance.

2 . LITERATURE SURVEY

1.**Title: An Intelligent Personalized Fashion Recommendation System**

Publisher :IEEE

Year:2019

Authors: Cristiana Stan et.al

Abstract: Creating an outfit is a problem that is based on the preferences of each person and it can be difficult even for the best experts. This paper presents an automated system that can recommend a full outfit based on a cloth item considering also user's preference. Two convolutional neural networks based on the AlexNet model are used to identify cloth items and attributes associated with each item. After that, two types of scores are used in order to evaluate the user's preference for combination of different items, that are continuously updated in order to obtain recommendations that are more suitable for each user.

Inference: This paper presents an automated system that can recommend a full outfit based on a cloth item considering also user's preference and to identify cloth items and attributes associated with each item.

2.Title: Explainable Outfit Recommendation with Joint Outfit Matching and Comment Generation

Publisher: IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING

Year:2020

Authors: Yujie Lin et.al

Abstract: Most previous work on outfit recommendation focuses on designing visual features to enhance recommendations. Existing work neglects user comments of fashion items, which have been proven to be effective in generating explanations along with better recommendation results. This paper propose a novel neural network framework, neural outfit recommendation (NOR), that simultaneously provides outfit recommendations and generates abstractive comments. Neural outfit recommendation (NOR) consists of two parts: outfit matching and comment generation. For outfit matching, they propose a convolutional neural network with a mutual attention mechanism to extract visual features. The visual features are then decoded into a rating score for the matching prediction. For abstractive comment generation, we propose a gated recurrent neural network with a cross-modality attention mechanism to transform visual features into a concise sentence. The two parts are jointly trained based on a multi-task learning framework in an end-to-end back-propagation paradigm. Extensive experiments conducted on an existing dataset and a collected real-world dataset show NOR achieves significant improvements over state-of-the-art baselines for outfit recommendation. Meanwhile, our generated comments achieve impressive ROUGE and BLEU scores in comparison to human-written comments. The generated comments can be regarded as explanations for the recommendation results. We release the dataset and code to facilitate future research.

Inference: This paper proposed a deep learning-based framework, called NOR, which simultaneously gives outfit recommendations and generates abstractive comments as explanations. We have released a large real-world dataset, ExpFashion, including images, contextual metadata of items, and user comments.

3.Title: A Conversation-Driven Approach for Chatbot Management

Publisher: IEEE

Year:2021

Author: Guilherme Guy de Andrade et.al

Abstract: Managing and evolving a chatbot's content is a laborious process and there is still a lack of standardization. In this context of standardization, the absence of a management process can lead to bad user experiences with a chatbot. This work proposes the Chatbot Management Process, a methodology for content management on chatbot systems. The proposed methodology is based on the experiences acquired with the development of Evatalk, the chatbot for the Brazilian Virtual School of Government. The focus of this methodology is to evolve the chatbot content through the analysis of user interactions, allowing a cyclic and human-supervised process. We divided the proposed methodology into three distinct phases, namely, manage, build, and analyze. Moreover, the proposed methodology presents a clear definition of the roles of the chatbot team. They validate the proposed methodology along with the creation of the Evatalk chatbot, whose amount of interactions was of 22,771 for the 1,698,957 enrolled attendees in the Brazillian Virtual School of Government in 2020. The application of the methodology on Evatalk's chatbot brought positive results: we reduced the chatbot's human hand-off rate from 44.43% to 30.16%, the chatbot's knowledge base examples increased by 160% whilst maintaining a high percentage of confidence in its responses and keeping the user satisfaction collected in conversations stable.

Inference: Managing and evolving a chatbot's content is a laborious process and there is still a lack of standardization. This work proposes the Chatbot Management Process, a methodology for content management on chatbot systems

4.Title: Color Navigation by Qualitative Attributes for Fashion Recommendation

Publisher: IEEE

Year:2018

Authors: Yeongnam Chae, Jiu Xu, Bjorn Stenger and Soh Masuko

Abstract: This paper proposes a novel method to navigate a color palette using attributes recognized from speech input. Our target application is a fashion recommender system for mobile e-commerce. Starting with a selected color, a user can request to show items of a different color by qualitative attributes (e.g. 'a little cuter'). These attributes are mapped to a query vector within the Lab color space in order to select the next color. The system distinguishes 85 attributes, each with three different possible magnitudes. This color navigation by speech was demonstrated in a mobile fashion recommender system. The proposed model is validated in a user study with 196 subjects.

Inference: This paper proposed a novel method to navigate a color palette via an attributebased query. By interpolating the path between input color and target color in polar coordinates, the attribute-based query is converted to a query vector in the Lab color space.

5.Title: Recommending Outfits from Personal Closet

Publisher: IEEE

Year:2018

Authors: Pongsate Tangseng, Kota Yamaguchi, and Takayuki Okatani

Abstract: In this paper,they consider grading a fashion outfit for recommenda tion, where we assume that users have a closet of items and we aim at producing a score for an arbitrary combination of items in the closet. The challenge in outfit grading is that the input to the system is a bag of item pictures that are unordered and vary in size. They build a deep neural network-based system that can take variable-length items and predict a score. They collect a large number of outfits from a popular fashion sharing website, Polyvore, and evaluate the performance of our grading system. They compare our model with a random-choice baseline, both on the traditional classification evaluation and on people's judgment using a crowdsourcing platform. With over 84% in classification accuracy and 91% matching ratio to human annotators, their model can reliably grade the quality of an outfit. They also build an outfit recommender on top of their grader to demonstrate the practical application of their model for a personal closet assistant

Inference: In this paper, they study outfits as combinations of items by developing outfit graders and outfit recommenders. Given a combination of items as an outfit, their best model can judge if the outfit looks good or not at over 84% accu racy on testing samples, and at 91% matching ratio on evaluations by human annotators.

2.1 EXISTING PROBLEM

In existing system only simple web application and their rating has been implemented in existing system, A ecommerce product recommendation engine is a piece of technology that displays recommended products to shoppers throughout your store. It uses machine learning to get smarter and show increasingly relevant products to shoppers based on their interests and previous browsing behavior.

Multiple number of online marketing companies around world-wide has been practicing the naive method for apparel marketing website. This paper aims to simulate this recommendation system on real world data set taken from the marketing giant, Amazon's Product Advertising API, in a policy compliant manner by following the procedure in three steps : Analyzing the data to select the pivot for the recommendation system, Data preprocessing to remove invalid sections and to implement and find appropriate choices among the techniques. Clothing is a kind of symbol that represents people's internal perceptions through their outer appearance. It conveys information about their choices, faith, personality, profession, social status, and attitude towards life. In this work we are interested in recommender systems that operate in one particular vertical market, garments and fashion products. This setting introduces a particular set of challenges and sub-problems, that are relevant for developing effective recommender systems.

2.2.REFERENCES:

1. W.-H. Cheng, S. Song, C.-Y. Chen, S. C. Hidayati, and J. Liu, "Fashion 1Meets Computer Vision: A Survey," arXiv, Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.13988.

2. J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Image-Based Recommendations on Styles and Substitutes," in Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Aug. 2015, pp. 43–52, doi: 10.1145/2766462.2767755.

3. H. Lee, J. Seol, and S. Lee, "Style2Vec: Representation Learning for Fashion Items from Style Sets," arXiv, Aug. 2017, [Online]. Available: http://arxiv.org/abs/1708.04014.

4. X. Gu, F. Gao, M. Tan, and P. Peng, "Fashion analysis and understanding with artificial intelligence," Inf. Process. Manag., vol. 57, no. 5, p. 102276, Sep. 2020, doi: 10.1016/j.ipm.2020.102276.

5.Prevayl Holdings Limited. Available online: https://www.prevayl.com (accessed on October 19, 2021

2.3 PROBLEM STATEMENT SOLUTION

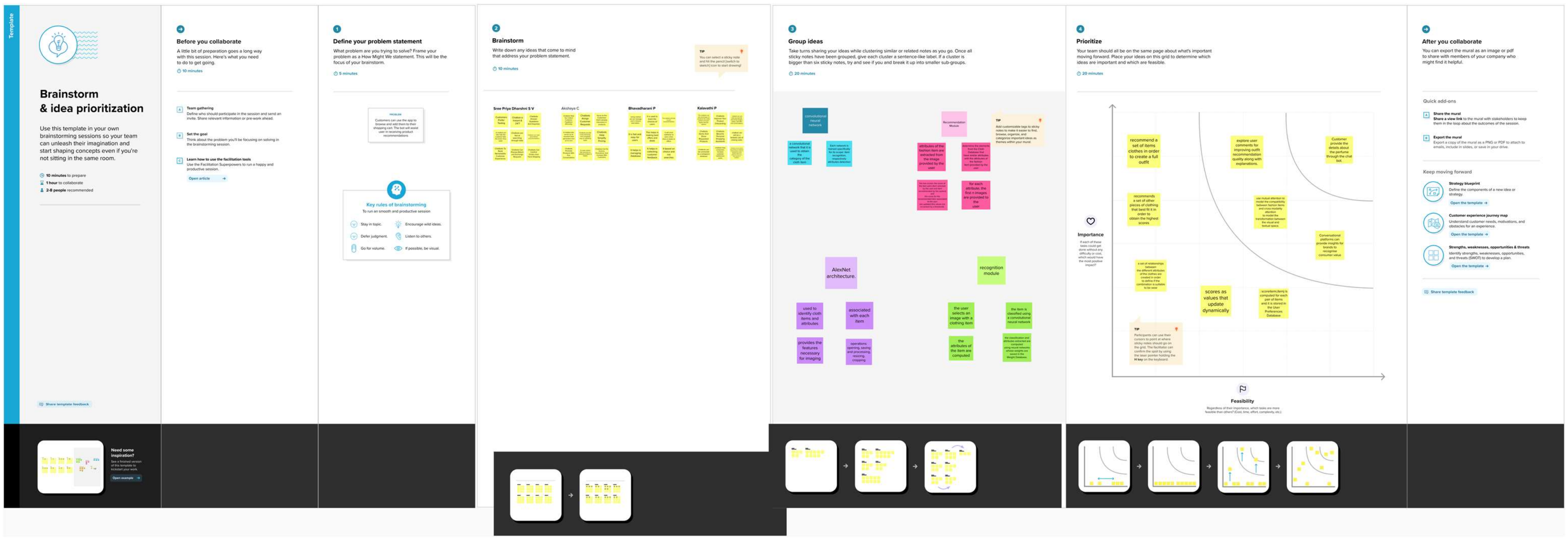
The personal information collected by recommenders raises the risk of unwanted exposure of that information. Also, malicious users can bias or sabotage the recommendations that are provided to other users. In recent years, the textile and fashion industries have witnessed an enormous amount of growth 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.

- The problem of the work is to design static web applications deployments with customer deployment
- Lack of interaction between application and user
- User need to navigate across multiple pages to choose right product
- Confusion in choosing product
- Lack of sales
- Complex User Interface.

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 behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges. An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers.



Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

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

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

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

Open article →

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM
Customers can use the app to browse and add them to their shopping cart. The bot will assist user in receiving product recommendations

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

2 Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Sree Priya Dharshni S V

Akshaya C

Bhavadharani P

Kalavathi P

3 Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

convolutional neural network

a convolutional network that it is used to obtain the category of the cloth item

Each network is trained specifically for its scope: item recognition, respectively attributes detection

Recommendation Module

TIP
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

attributes of the fashion item are extracted from the image provided by the user

for each attribute, the first n images are provided to the user

AlexNet architecture.

recognition module

4 Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

recommend a set of items clothes in order to create a full outfit

explore user comments for improving outfit recommendation quality along with explanations.

Customer provide the details about the perfume through the chat bot.

recommends a set of other pieces of clothing that best fit it in order to obtain the highest scores

Conversational platforms can provide insights for brands to recognise consumer value

scores as values that update dynamically

Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

TIP
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.

Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

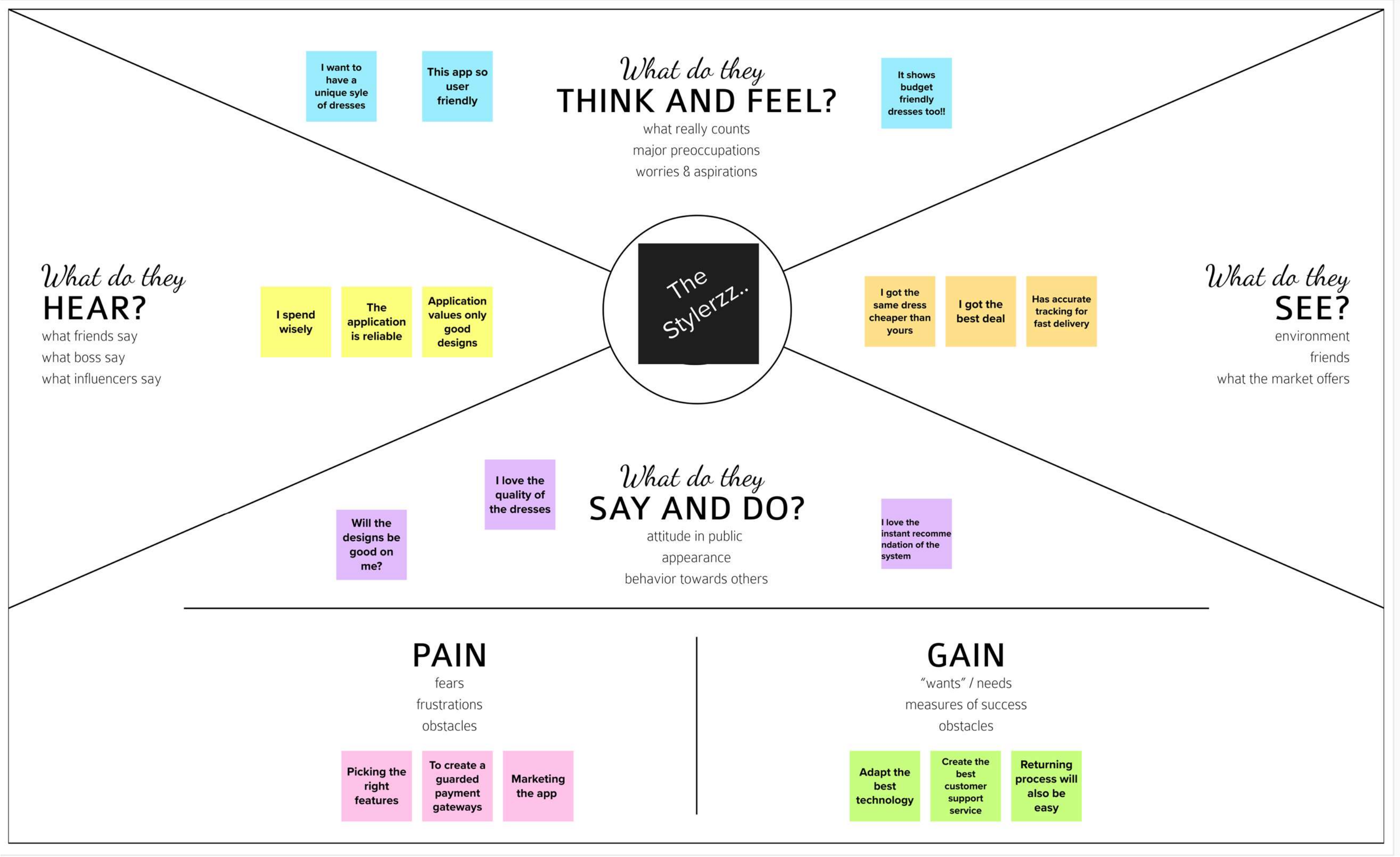
Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

Share template feedback

3.2 IDEATION & BRAINSTROMING

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


I want to have a unique syle of dresses
This app so user friendly
What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
It shows budget friendly dresses too!!
What do they
HEAR?
what friends say
what boss say
what influencers say
I spend wisely
The application is reliable
Application values only good designs
The Stylerzz..
I got the same dress cheaper than yours
I got the best deal
Has accurate tracking for fast delivery
What do they
SEE?
environment
friends
what the market offers
Will the designs be good on me?
I love the quality of the dresses
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
I love the instant recommendation of the system
PAIN
fears
frustrations
obstacles
Picking the right features
To create a guarded payment gateways
Marketing the app
GAIN
"wants" / needs
measures of success
obstacles
Adapt the best technology
Create the best customer support service
Returning process will also be easy

3.3 PROPOSED SOLUTION:

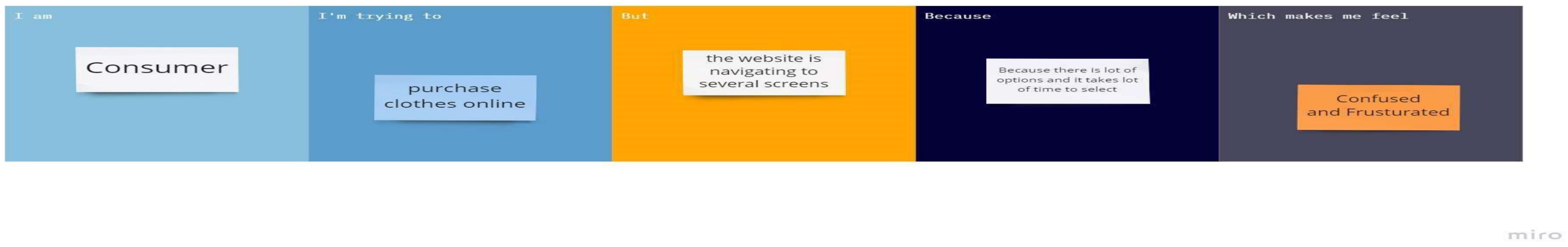
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Customers can use the app to browse and add them to their shopping cart. The bot will assist user in receiving product recommendations.
2.	Idea / Solution description	We have come up with a new innovative solution through which you can directly do your online shopping based on your choice without any search. It can be done by using the chatbot.
3.	Novelty / Uniqueness	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.
4.	Social Impact / Customer Satisfaction	Can recommend products that are suitable and can also save lot of time
5.	Business Model (Revenue Model)	Conversion of lead to customer through better experience and feasibility
6.	Scalability of the Solution	The customer service chatbots help businesses grow and scale with ease, especially when web traffic volume increases.

3.4 PROBLEM SOLUTION FIT

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

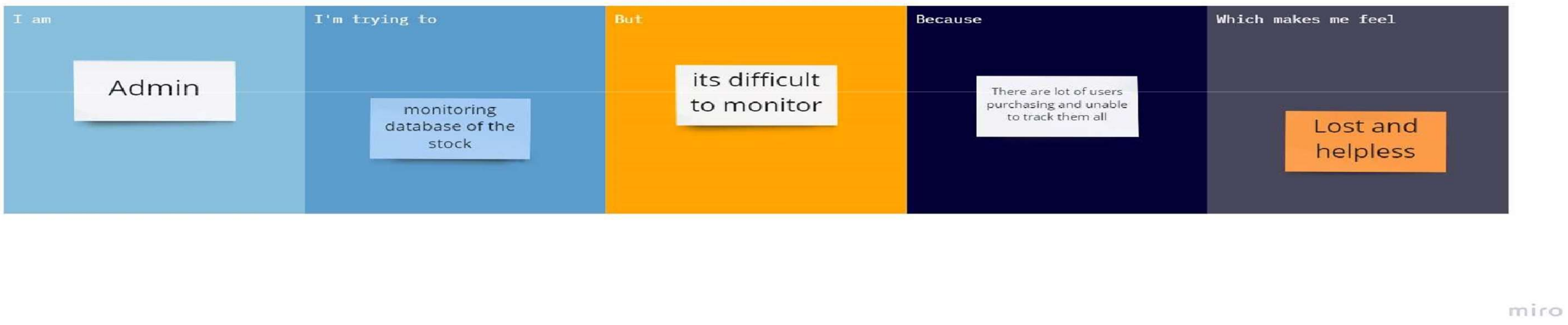
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1(based on user)	Consumer	Purchase Clothes Online	The website is navigating to several screens	Because there is lot of options and it takes a lot of time	Confused and frustrated
PS-2(based on Admin)	Admin	Monitoring database of the stock	It's difficult to monitor	There are lot of users purchasing and unable to track them all	Lost and helpless

PS-1:



I am	I'm trying to	But	Because	Which makes me feel
Consumer	purchase clothes online	the website is navigating to several screens	Because there is lot of options and it takes lot of time to select	Confused and Frusturated

PS-2:



I am	I'm trying to	But	Because	Which makes me feel
Admin	monitoring database of the stock	its difficult to monitor	There are lot of users purchasing and unable to track them all	Lost and helpless

4. REQUIREMENT ANALYSIS

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 Gmail Registration through Mobile number Registration through Facebook
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Browsing of Product	Browsing of product can be made easy by filtering it by price, colour, discount ,gender, brands etc
FR-4	Chatbot Assistance	Chatbot support to put users at ease. It speeds up response time and provide service 24/7
FR-5	Shopping Cart	Shopping Cart consists of, □ Add to Wishlist □ Add to bag □ Remove from Wishlist □ Remove from bag
FR-6	Payment details	Payment can be using debit card, credit card, net banking, UPI and cash on delivery
FR-7	Confirmation	Confirmation can be done by sending a mail or a SMS to registered phone number.
FR-8	Return Policy	If product is not satisfied by customers it can be returned up to 10 days after purchase and money will be returned.

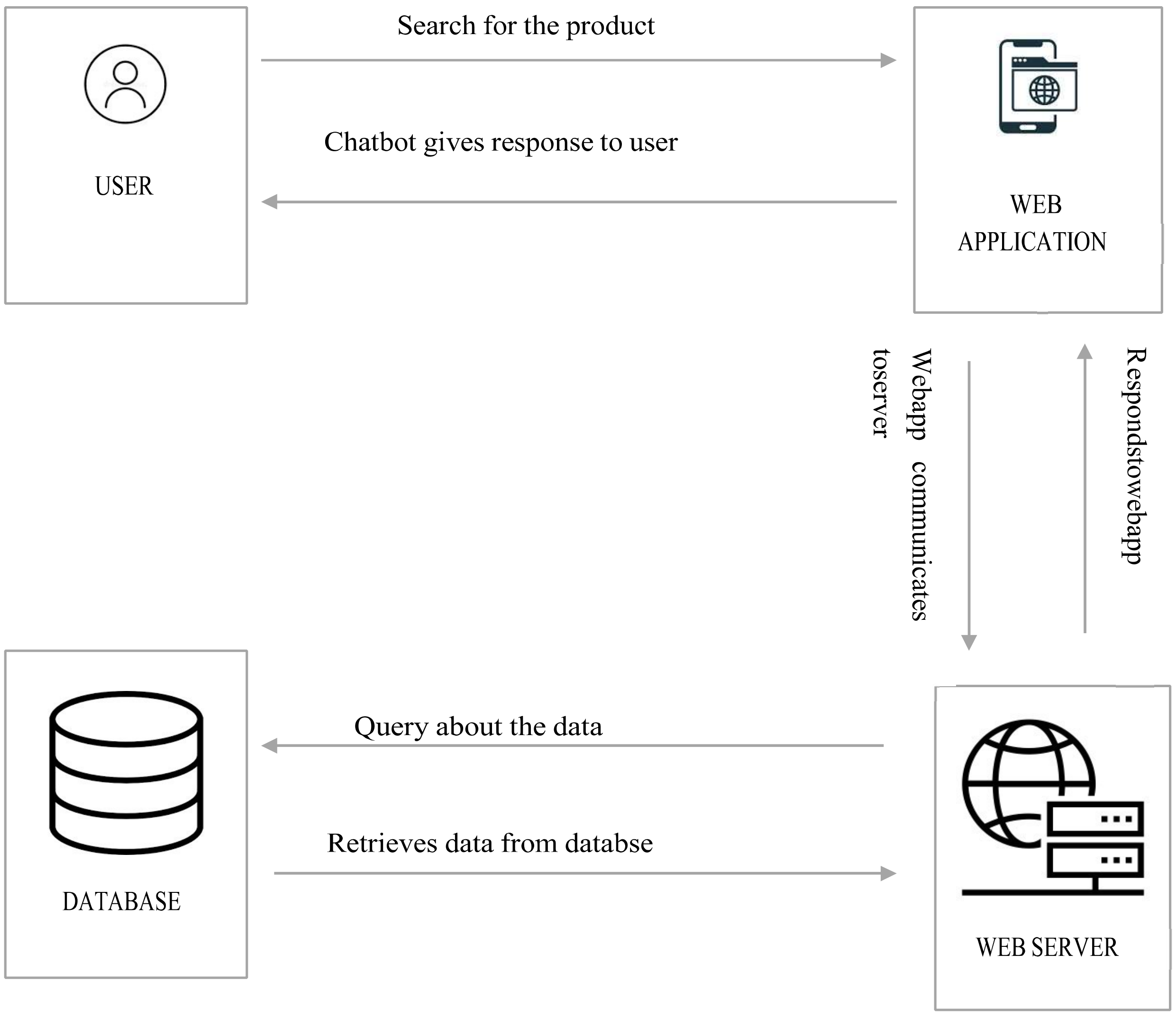
4.2.Non-functional Requirements:

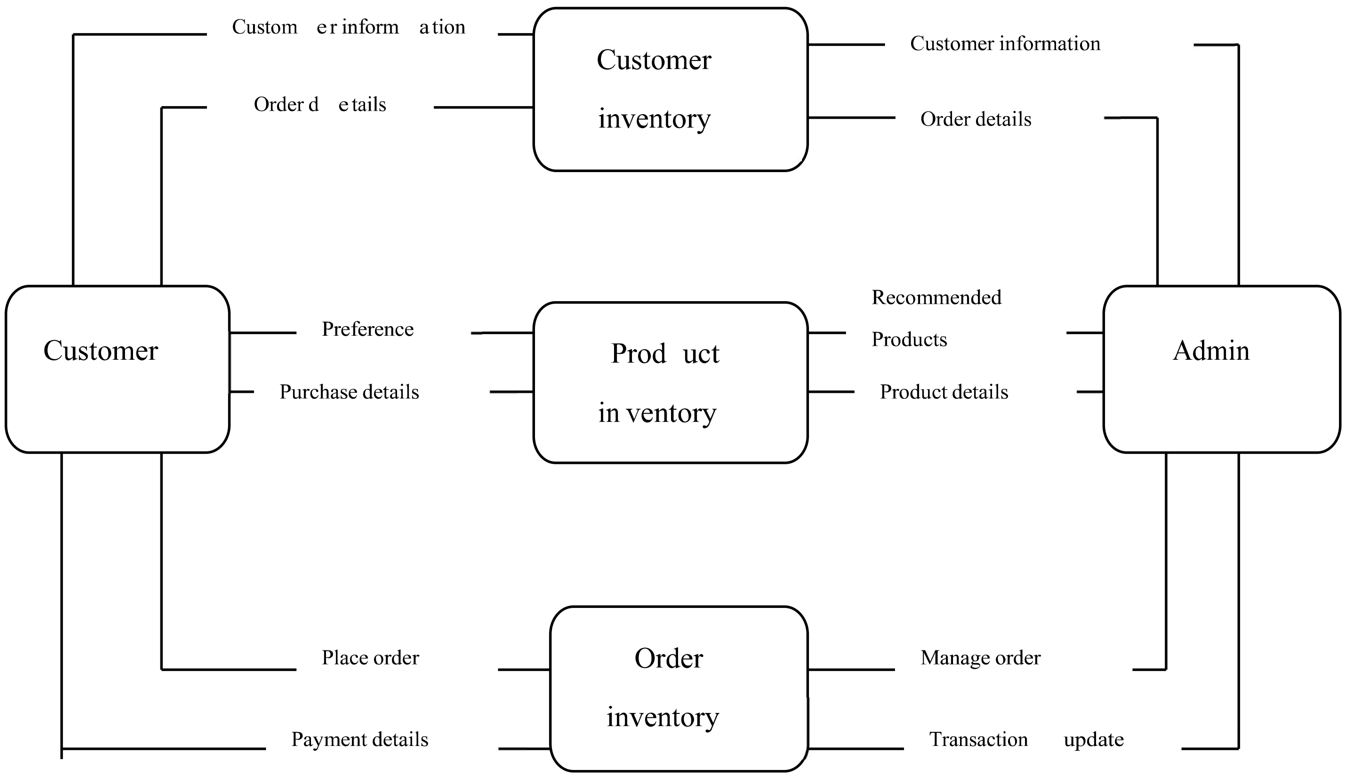
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Usability of app will be more because of chatbothuman interaction. It will be useful t make customers at ease while shopping and have friendly interactions.
NFR-2	Security	Every Features will be encrypted especially payment will be secure
NFR-3	Reliability	Since cloud service is used, it will be more reliable and it can handle high traffic which makes the performance efficient.
NFR-4	Performance	A good-performing application provides flawless user experience to the customers. Cloud application performance management ultimately take actions to resolve issues and maintain optimal performance.
NFR-5	Availability	To meet customer requirements anytime and anywhere, application and chatbot will be available for 24/7.
NFR-6	Scalability	The customer service chatbots help businesses grow and scale with ease, especially when web traffic volume increases

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAM:





5.2 SOLUTION & TECHNICAL ARCHITECTURE

Table-1 : Components & Technologies:

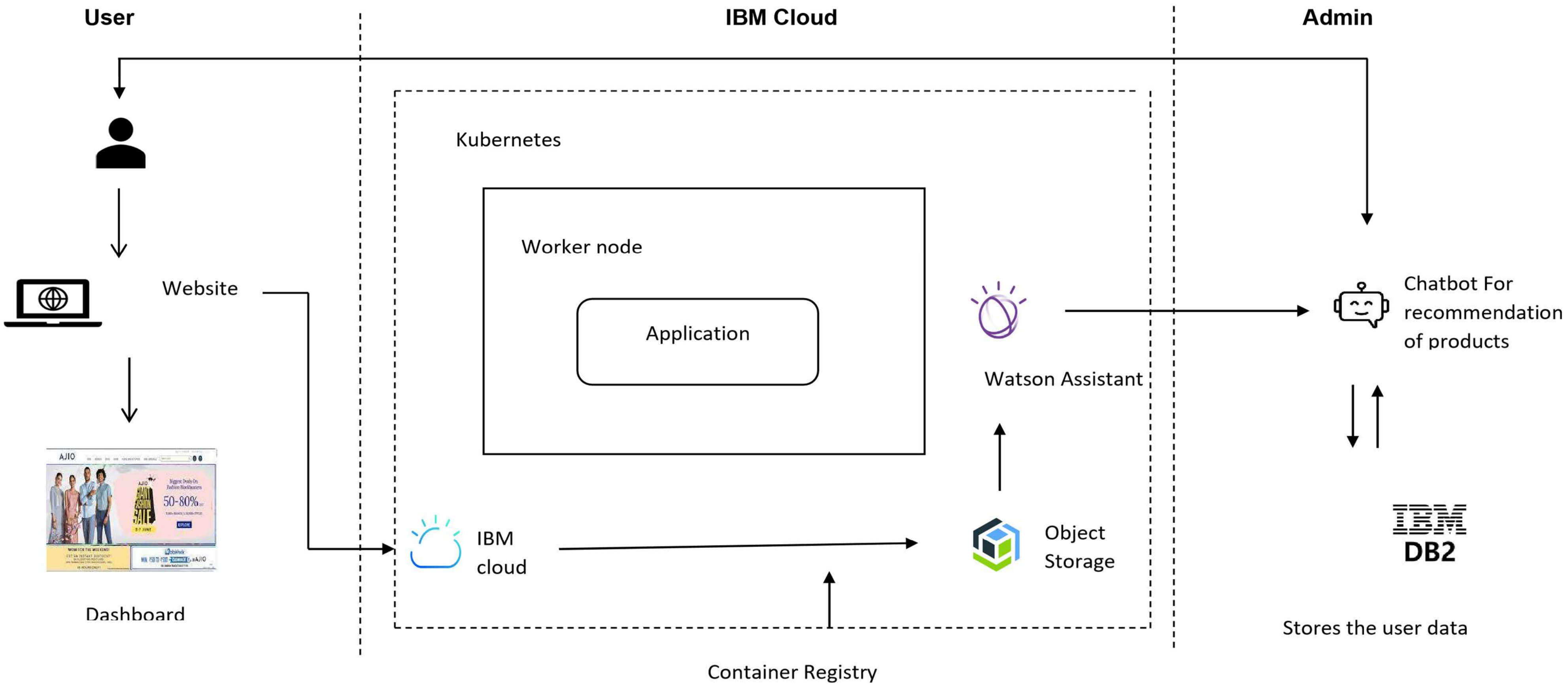
S.No	Component	Description	Technology
1.	Website	To get desired product, customer can chat with chatbot.	HTML, Watson Chatbot, CSS, JavaScript
2.	Kubernetes	It monitors application and make it easier to manage it. If any software crash it automatically restart the work	Kubernetes
3.	IBM Object Storage	It's delivered on demand with just-in-time capacity and costs, and eliminates buying and managing your own data storage infrastructure.	Bucket
4.	Docker	Docker is a tool designed to make it easier for developers to develop, ship, and run applications by using containers.	Container
5.	DB2	The Db2 DBMS operates as the server to manage data in databases across a multiuser environment, enabling many concurrent users to access the same data simultaneously.	MySQL, NoSQL, etc.
6.	Cloud Database	IBM Db2 on Cloud is a fully managed SQL cloud database that offers a dedicated operations team, point-in-time recovery.	IBM DB2, IBM Cloudant etc.
7.	Container registry	Container Registry is a service for storing private container images	Kubernetes
8.	Watson Assistant	Watson Assistant lets you build conversational interfaces into any application, device, or channel	IBM Watson Assistant
9.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Anaconda Cloud Server Configuration : IBM cloud	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Cloud Stack, Eucalyptus. Open Nebula, App Scale, Docker	Docker
2.	Security Implementations	Authentication and password management Accountability to authorize and monitor the use anonymous accounts and to remove	Encryptions, Secured Authorization.
3.	Scalable Architecture	The customer service chatbots help businesses grow and scale with ease, especially when web traffic volume increases.	DB2,Watson Chatbot
4.	Availability	To meet customer requirements anytime and anywhere, application and chatbot will be available for 24/7.The stock will be frequently checked.	Docker

5.	Performance	A good-performing application provides flawless user experience to the customers. Cloud application performance management ultimately take actions to resolve issues and maintain optimal performance.	Kubernetes

Technical Architecture:



5.3 USER STORIES:

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	I can login through Gmail account which is signed up already.	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login by entering email ID and password.	High	Sprint-1

	Dashboard	USN-6	Instead of browsing through different products and selecting them, customer can chat with chatbot. Chatbot helps in recommending products according to user’s preferences.	Instead of navigation through different tabs, chatbot can be used.	High	Sprint-1
	Chatbot	USN-7	Chatbot helps in choosing best deals and offers of the day	It helps me to reduce the cost.	High	Sprint-3
		USN-8	Chatbot helps in payment and confirming the order.	I receive confirmation mail for verifying	High	Sprint-2
Customer (Web user)	Login	USN-1	Not only the app is required to install, it can be logged into by knowing its link.	I can log into different systems by knowing the ID and password	Medium	Sprint-1
Customer Care Executive	Action	USN-1	Chatbot is available 24/7.So whenever a problem is faced it can be rectified.	I can rectify my doubts with chatbot	High	Sprint-3
Administrator	Admin	USN-1	They maintain a record of everything that users are purchasing.	I can store the information	High	Sprint-1
		USN-2	The administrator's job is to search the stock database.	I can approve products and purchases	High	Sprint-2

6 PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION:

Remaining Task:

MILESTONES	ACTIVITY	DESCRIPTION
Project development phase	Delivery of Sprint-1,2, 3, 4	To develop the code and submit the develop the code after completion of testing
Setting-up app environment	Create IBM cloud account	Sign up IBM cloud account
	Create flask project	Getting started with the flask to create project
	Install IBM cloud cli	Install IBM command line interface (CLI)
	Docker CLI installation	Installing docker CLI
	Create an account in sendgrid	Create an account in sendgrid Use service as e-mail integration to the application for sending emails
Implementing web application	Create UI to interact with the application	Create UI □ registration page □ login page □ view products page □ add products page
	Create IBM DB2 and connect with the Python	Create IBM DB2 in IBM cloud and link with the Python
Integrating sendgrid service	Sendgrid integration with the Python	To send emails from the application we need to integrate the sendgrid services
Developing a chat bot	Building a chat bot and integrate with the application	Build the chat bot and integrate it to the flask application
Deployment of app in IBM cloud	Containerise the app	Create a docker image of the application in addition to push it to the IBM container registry
	Upload image to IBM container registry	Upload the image to IBM container registry
	Deploy in in kubernetes cluster	Once the image is uploaded to IBM container registry deploy the image toward IBM kubernetes cluster

Completed Tasks:

MILESTONES	ACTIVITY	DESCRIPTION
Ideation phase	Literature survey	Literature survey on the selected project and information gathering
	Empathy map	Prepare empathy map to capture the user pains and gains, prepare a list of problem statement
	Ideation	Organising the brainstorming session and prioritise the top three ideas based on feasibility hand importance
Project design phase 1	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 documents
	Solution architecture	Prepare solution architecture document
Project design phase 2	Customer journey map	Prepare customer journey map to understand the user interactions and experience with the application
	Functional requirements	Prepare functional and non- functional necessity document
	Data flow diagram	Prepare data flow diagram and user stories
	Technology architecture	Draw technology architecture diagram
Project planning phase	Milestones and activity list	Prepare milestones and activity list of the project
	Sprint delivery plan	Prepare spring delivery plan

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story points	Priority	Team Members
Sprint-1	Setting up App environment	USN-1	As a user, I can register in ICTA Academy and create IBM cloud account.	2	High	Kalavathi.P Bhavadharani.P
Sprint-1		USN-2	As a user, I will create a flask project	1	Low	Akshaya.C Sree Priya dharshni.S.V
Sprint-1		USN-3	As a user, I will install IBM Cloud CLI	2	Medium	Kalavathi.P Akshaya.C
Sprint-2	Setting up App environment	USN-4	As a user, I can install Docker CLI	1	Low	Bhavadharani.P Sree Priya dharshni.S.V
Sprint-2		USN-5	As a user, I will Create an account in sendgrid	2	Medium	Akshaya.C Sree Priya dharshni.S.V Kalavathi.P Bhavadharani.P
Sprint-3	Implementing web application	USN-6	As a user, I Create UI to interact with the application	1	High	Akshaya.C Sree Priya dharshni.S.V Kalavathi.P Bhavadharani.P
Sprint-3		USN-7	As a user, I Create IBM DB2 and connect with Python	3	High	Bhavadharani.P Sree Priya dharshni.S.V
Sprint-3	Integrating sendgrid service	USN-8	As a user, I will be integrating sendgrid with python code	2	High	Akshaya.C Sree Priya dharshni.S.V Kalavathi.P Bhavadharani.P

Sprint-3	Developing a chatbot	USN-9	As a user, I must build a chatbot and integrate to application	1	Medium	Akshaya.C Sree Priya dharshni.S.V Kalavathi.P Bhavadharani.P
Sprint-4	Development of App in IBM Cloud	USN-10	As a user, I will Containerize the App	1	Low	Kalavathi.P Akshaya.C
Sprint-4		USN-11	As a user, I will upload image to IBM Container registry	2	Medium	Bhavadharani.P Sree Priya dharshni.S.V
Sprint-4		USN-12	As a user, I will deploy App in Kubernetes cluster	3	High	Kalavathi.P Bhavadharani.P
Sprint-4	User panel		As a user ● Register, Login, Email, Verification ● Manual Search ● Order placement, Order Details	3	High	Akshaya.C Sree Priya dharshni.S.V Kalavathi.P Bhavadharani.P

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

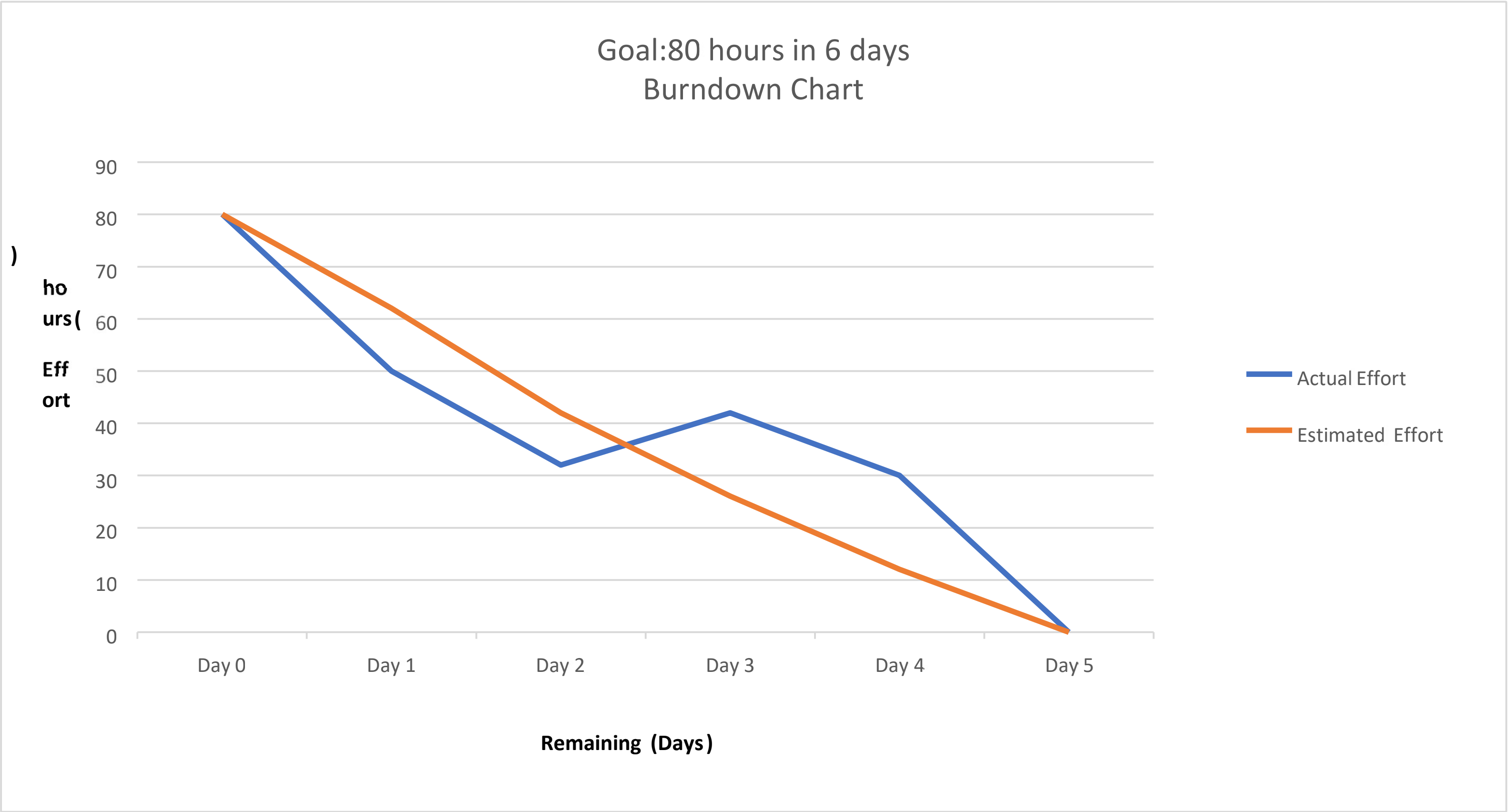
Velocity

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

Burndown Chart

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



7.CODING AND SOLUTION

7.1 FEATURE 1

```
<html>


<head>

    <title> JMTG </title>


</head>

<style>

    * {         margin: 0;
padding: 0;
```

```
      font-family: "Century Gothic", CenturyGothic, AppleGothic, sans-serif;

   }


      .main {        width: 100%;        background: linear-gradient(to top, rgba(219, 18, 18, 0.5), rgba(0, 0, 0, 0.5)50%);
    backgroundposition: center;        background-size: cover;        height: 100%;        font-family: "Century Gothic",
                                                    CenturyGothic, AppleGothic, sans-
serif;

   }


   .navbar {        width: 100%;        height: 75px;        margin: auto;

   }


   .icon {         width: 200px;
float: left;         height: 70px;
   }


   .logo {        color: rgb(98, 246, 152);        font-size: 35px;        padding-left:

20px;

      float: left;

      padding-top: 10px;

   }


   .menu {        width: 400px;
float: left;        height: 70px;
   }        ul {        float: left;        display: flex;        justify-content: center;

align-items: center;

   }        ul li {

      list-style: none;        margin-left: 62px;        margin-top: 27px;        font-size:

14px;

   }        ul li a {        text-decoration: none;        color: #ffff;
font-weight: bold;        transition: 0.4s ease-in-out;
   }


   ul li a:hover {        color: rgb(98, 246, 152);

   }


   .search {        width: 330px;

      float: left;
margin-left: 270px;
   }
```

```
    .srch {

        width: 200px;         height: 40px;         background: transparent;         border:
1px solid rgb(98, 246, 152);         margin-top: 13px;
        color: #ffff;         border-right: none;         font-size: 16px;         float: left;
padding: 10px;         border-bottom-left-radius: 5px;         border-top-left-radius:
5px;

    }


    .btn {

        width: 100px;         height: 40px;         background: rgb(246, 98, 221);
border:

2px rgb(246, 98, 221);         margin-top: 13px;         color: #ffff;         font-size:
15px;         border-bottom-right-radius: 5px;         border-
bottom-right-radius: 5px;

    }


    .btn:focus {         outline: none;

    }


    .srch:focus {         outline: none;

    }


    .content {         width: 1200px;         height: auto;         margin: auto;         color:
rgb(161, 144, 144);         position: relative;
    }


    .content.par {         padding-left: 20px;         padding-bottom: 25px;         letter-
spacing: 1.2px;         line-height: 30px;
    }


    .content h1 {         font-size: 50px;         padding-left: 20px;
margintop: 9%;         letter-spacing: 2px;
    }


    .content .cn {         width: 160px;         height: 40px;         background: rgb(246,
98, 221);         border: none;         margin-bottom: 10px;         margin-left: 20px;
fontsize: 18px;         border-radius: 10px;         cursor: pointer;         transition: .4s
ease;
    }
```

```
   .content .cn a {         text-decoration: none;         color: #000;
transition: .3s ease;
   }


   .cn:hover {        background-color: #000;

   }


   .content span {        color: rgb(246, 98, 221);        font-size: 60px;

   }


   .form {        width: 250px;        height: 380px;        background: linear-gradient(to top, rgba(243, 241, 239, 0.8), rgba(252, 252, 252,
0.8)50%);        position: absolute;        top: -20px;        left: 870px;        border-radius: 10px;        padding: 25px;
   }


   .form h2 {        width: 220px;        text-align: center;        color: rgb(235,
226, 100);        font-size: 22px;        border-radius: 10px;        margin: 2px;
padding: 8px;

   }
   .form input {        width: 240px;        height: 35px;        background:

transparent;

   }


   .form input {        width: 240px;        height: 35px;        background: transparent;
borderbottom: 1px solid rgb(246, 98, 221);        border-top: none;        border-right: none;
borderleft: none;        color: rgb(14, 12, 12);        font-size: 15px;        letter-spacing: 1px;
margintop: 30px;
   }


   .form input:focus {        outline: none;

   }


    ::placeholder {        color: #000;

   }


   .btnn {        width: 240px;        height: 40px;        background: rgb(22, 21, 22);
border: none;        margin-top: 30px;        font-size: 18px;        border-radius:
10px;        cursor: pointer;        color: rgb(105, 100, 100);        transition: 0.4s ease;

   }


   .btnn:hover {        background: #000;        color: rgb(235, 201, 229);
```

```
    }
    .btnn a {         text-decoration: none;         color: white;         font-weight:
bold;

    }


    .form .link {         font-size: 17px;         padding-top: 20px;
textalign: center;
    }


    .form .link a {         text-decoration: none;         color: rgb(246, 98, 221);

    }


    .liw {

        padding-top: 15px;         padding-bottom: 10px;         text-align: center;

    }

</style>





<body>


    <div class="main">

        <div class="navbar">

            <div class="icon">

                <h2 class="logo">SMART FASHION</h2>

            </div>


            <div class="menu">

                <ul>

                    <li><a href="#">HOME</a></li>

           <li><a href=”#”>CONTACT</a></li>

                    <li><a href="#">ABOUT US</a></li>



                </ul>

            </div>


            <div class="search">
        <input class="srch" type="search" name="" placeholder="Type to
search">

        <a href="#"><button class="btn">Search</button></a>

      </div>


    </div>


    <div class="content">

      <h1>Smart Fashion <br><span>Application</span></h1>


      <div class="form">

        <h2>SMART REGISTER</h2>

        <form action="/Register" method="post">

          <input type="text" name="username" placeholder="Username">

          <input type="email" name="email" placeholder="Email id">

          <input type="password" name="password" placeholder="Password">

          <button type="submit" class="btnn">REGISTER</button>


          <p class="link">Already have an account<br>

            <a href="/Login">Log in</a> here</a>

          </p>

        </form>



      </div>

    </div>

  </div>


</body>


</html>
```

7.2 FEATURES

```
@import url('https://fonts.googleapis.com/css2?family=Spartan:wght@100;200;300;400;500;6 00;700;800;900&display=swap');

*{    margin:0;    padding:0;    box-sizing:border-box;
font-family: 'Spartan', sans-serif;
}

h1{     font-size:50px;     line-height:64px;
color:#222;
}

h2{     font-size:46px;     line-height:54px;
color:#222;
}

h4{     font-size:20px;
color:#222;
}

h6{     font-weight:700;     font-
size:12px;
}

p{    font-size:16px;     color:#465b52;     margin:15px 0 20px 0;

}

.section-p1{     padding:40px 80px;

}

.section-m1{     padding:40px 0;

}

button.normal{     font-size:14px;     font-weight:600;     padding:15px 30px;
color:#000;     background-color:#fff;     border-radius:4px;     cursor:pointer;
border:none;     outline:none;     transition:0.2s;

}

button.white{     font-size:13px;     font-weight:600;
padding:11px 18px;
    color:#fff;     background-color:transparent;     cursor:pointer;
border:1px solid #fff;     outline:none;     transition:0.2s;

}

body{

    width:100%;

}

/* Header start E3E6F3*/
```

```
#header{

   display:flex;     align-items:center;     justify-content:space-between;
padding:6px 60px;     background:#ffffff;     box-shadow:5px 10px #800080;     z-
index:999;     position:sticky;     top:0;     left:0;

}


#navbar{     display:flex;     align-items:center;
justify-content:center;
}


#navbar li{     list-style:none;     padding:0 20px;     position:relative;

}


#navbar li a{     text-decoration:none;     font-size:16px;     font-weight:600;
color:#1a1a1a;     transition:0.3s ease;

}


#navbar li a:hover, #navbar li a.active{     color:#800080;

}


#navbar li a.active::after,
#navbar li a:hover::after{
   content:"";     width:30%;     height:2px;     background:#800080;
position:absolute;     bottom:-4px;     left:20px;

}

/* Home page */


#hero{     background-image:url("https://unik.s3.jp-tok.cloud-objectstorage.appdomain.cloud/home.jpg");

   height:90vh;     width:100%;     background-size:cover;     background-position:
top 25% right 0;     padding:0 80px;     display: flex;     flex-direction:column;
align-items:flex-start;     justify-content:center;

}


#hero h4{     padding-
bottom:15px;
}
#hero h1{     color:#800080;

}
```

```
#feature{     display:flex;     align-items:center;     justify-content:space-between;
flex-wrap:wrap;
}

#feature .fe-box{     width:180px;     text-align:center;
padding:25px 15px;     box-shadow:20px 20px 34px
rgba(0,0,0,0.03);     border:1px solid #cce7d0;     border-radius:4px;
margin:15px 0;
}

#feature .fe-box:hover{     box-shadow:20px
10px 54px rgba(70,62,221,0.1);
}

#feature .fe-box img{     width:100%;     margin-bottom:10px;
}

#feature .fe-box h6{     display:inline-block;     padding:9px 8px 6px 8px;
lineheight:1;     border-radius:4px;     color:#088178;     background-
color:#fddde4;
}

#feature .fe-box:nth-child(2) h6{     background-color: #cdebbc;
}

#feature .fe-box:nth-child(3) h6{     background-color: #d1edf2;
}

#feature .fe-box:nth-child(4) h6{     background-color: #cdd4f8;
}

#feature .fe-box:nth-child(5) h6{ background-color: #f6dbf6;
}

#feature .fe-box:nth-child(6) h6{     background-color: #fff2e5;
}

#product1{     text-
align:center;
}

#product1 .pro-container{     display:flex;     justify-content:space-between;
paddingtop:20px;     flex-wrap:wrap;
```

```
}

#product1 .pro{     width:23%;     min-width:250px;     padding:10px 12px;
border:1px solid #cce7d0;     border-radius:25px;     cursor:pointer;      box-
shadow:20px 20px 30px rgba(0,0,0,0.02);
   margin:15px 0;     transition: 0.2s ease;     position:relative;
}

#product1 .pro:hover{       box-shadow:20px
20px 30px rgba(0,0,0,0.06);
}

#product1 .pro img{     width:100%;     border-radius:20px;

}

#product1 .pro .des{ text-align:start; padding:10px 0;

}

#product1 .pro .des span{     color:#606063;     font-size:14px;

}

#product1 .pro .des h5{     padding-top:7px;     color:#1a1a1a;
fontsize:14px;

}

#product1 .pro .des i{     font-size:12px;     color:rgb(243,181,25);

}

#product1 .pro .des h4{     padding-top:7px;     color:#800080;
fontsize:15px;     font-weight:700;

}

#product1 .pro .bag{     width:40px;     height:40px     line-height:40px
borderradius:50px     background-color:#800080;     font-weight:500;
color:#800080;


   position:absolute;     bottom:20px;     right:10px;

}

#banner{ display:flex; flex-direction:column;     justify-content:center;     align-
    items:center;     text-align:center;

   background-image:url("https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/unik/banner.jpg");
width:100%;     height:40vh;     background-size:cover;     background-position:center;
```

```
}

#banner h4{     color:#fff;     font-size:16px;
}

#banner h2{     color:#fff;     font-size:30px;     padding:10px 0;
}

#banner h2 span{     color:#ef3636;
}

#banner button:hover, #hero button:hover{     background:#800080;
color:#fff;
}

#sm-banner{
   display:flex;     justify-content:space-between;     flex-wrap:wrap
}
#sm-banner .banner-box{     display:flex;
    flex-direction:column; justify-content:center;     align-items:flex-start;
    background-image:url("https://s3.jp-tok.cloud-
    objectstorage.appdomain.cloud/unik/sm.jpg");     min-width:580px;
    height:50vh;     background-size:cover;     background-position:center;
    padding:30px;
}

#sm-banner .banner-box2{     background-image:url("https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/unik/sm1.jpg");
}


#sm-banner h4{     color:#000;     font-size:20px;     font-weight:300
}

#sm-banner h2{     color:#000;     font-size:28px;     font-weight:800
}

#sm-banner span{     color:#000;     font-size:14px;     font-weight:500;     padding-
bottom:15px;
}

#sm-banner button{     color:#000;
```

```
}

#sm-banner .banner-box:hover button{     background:#800080;     color:#000;

}


#banner3{

   display:flex;     justify-content:space-between;     flex-wrap:wrap;
padding:0 80px;

}


#banner3 .banner-box{

      display:flex;       flex-direction:column;       justify-content:center;       align-items:flex-start;       backgroundimage:url("https://s3.jp-
tok.cloud-objectstorage.appdomain.cloud/unik/banner31.jpg");

      min-width:30%;       height:30vh;       background-size:cover;
backgroundposition:center;       padding:8px;       margin-bottom:20px;

}


#banner3 .banner-box2{      background-image:url("https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/smart3/5.jpg");

}


#banner3 .banner-box3{      background-image:url("https://s3.jp-tok.cloud-
objectstorage.appdomain.cloud/unik/banner33.jpg");
}


#banner3 h2{     color:#fff;     font-weight:000;     font-size:22px;

}


#banner3 h3{     color:#ec544e;     font-weight:800;     font-size:15px;

}


#banner3 .banner-box:hover{

   box-shadow:20px 10px 54px rgba(70,62,221,0.1);

}



#newsletter{     display:flex;     justify-content:space-between;     flex-wrap:wrap;
alignitems:center;     background-image:url("img/purple.jpg");     background-
repeat:no-repeat;     background-position:20% 30%;     background-color:#800080;

}


#newsletter h4{     font-size:22px;     font-weight:700;     color:#000;

}
```

```
#newsletter p{     font-size:14px;     font-weight:600;     color:#000;
}

#newsletter p span{     color:#ffbd27;
}

#newsletter .form{     display:flex;     width:40%;
}


#newsletter input{
   height:3.125rem;     padding: 0 1.25em;     font-size:14px;     width:100%;
border:1px solid transparent;     border-radius:4px;     outline:none;     bordertop-
right-radius:0;     border-bottom-right-radius:0;
}

#newsletter button{     background-color:#000;     color:#fff;     white-space:nowrap;
border-top-left-radius:0;     border-bottom-left-radius:0;
}

footer{     display:flex;     justify-content:space-between;     flex-wrap:wrap;
}

footer .col{     display:flex;     flex-direction:column;     align-
items:flexstart;     margin-bottom:20px;     padding-bottom:20px;
}

footer .logo{     margin-bottom:2px;
}

footer h4{     font-size:14px;     padding-bottom:20px;
}

footer p{     font-size:13px;     margin:0 0 8px 0;
}

footer a{     font-size:13px;     text-direction:none;     color:#222;
margin:0 0 8px 0;
}

footer a:hover{     width=100%;     text-align:center;     color:#800080;
}

/* Products Page */

#page-header{
    background-image:url("https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/unik/pro-banner.jpg");
width:100%;     height:40vh;     background-size:cover;     display:flex;     justify-content:center;
textalign:center;     flex-direction:column;     padding:14px;
}

#page-header h2, #page-header p{     color:#000;
}

#pagination{     text-align:center;
}

#pagination a{
    text-decoration:none;     background-color:#DDA0DD;     padding:15px 20px;
border-radius:4px;     color:#000;     font-weight:600;
}

#pagination a i{     font-size:16px;     font-weight:600;
}

#pagination a:hover{     background-color:#800080;     color:#fff;
}

/* Single Product */

#prodetails{     display:flex;     margin-top:20px;
}

#prodetails .single-pro-image{     width:40%;     margin-right:50px;
}

.small-img-group{     display:flex;     justify-content:space-between;
}

.small-img-col{     flex-basis:24%;     cursor:pointer;
}
```

```
#prodetails .single-pro-details{ width:50%;     padding-
top:100px;
}

#prodetails .single-pro-details h4{     padding: 40px 0 20px 0;

}

#prodetails .single-pro-details h2{     font-size:26px;

}

#prodetails .single-pro-details select{     display:block;     padding:5px 10px;
marginbottom:10px;

}

#prodetails .single-pro-details input{     width:50px;     height:47px;
paddingleft:10px;     font-size:16px;     margin-right:10px;

}

#prodetails .single-pro-details input:focus{
outline:none;
}

#prodetails .single-pro-details button{     background-
color:#800080;     color:#fff;
}

#prodetails .single-pro-details span{     line-height:25px;

}



/* Blog Page */

#page-header.blog-header{     background-image:url("https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/unik/blog.jpeg");

}

#blog{     padding :150px
150px 0 150px;
}

#blog .blog-box{     display:flex;     align-items:center;     width:100%;
position:relative;     padding-bottom:90px;
}
```

```
#blog .blog-img{     width:50%;     margin-right:40px;

}


#blog img{     width:100%;     height:300px;     object-fit:cover;

}


#blog .blog-details{     width:50%;

}


#blog .blog-details a{     text-decoration:none;     font-size:11px;
color:#000;     font-weight:700;     position:relative;     transition:0.3s;

}


#blog .blog-details a::after{

     content:"";

    width:50px;     height:1px;     background-color:#000;
position:absolute;     top:4px;     right:-60px;

}


#blog .blog-details a:hover{     color:#800080;

}


#blog .blog-details a:hover::after{      background-color:#800080;

}


#blog .blog-box h1{     position:absolute;     top:-40px;

    left:0;     font-size:70px;     font-weight:700;     color:3c9cbce;
zindex:-9;

}



/* About Page */


#page-header.about-header{     background-image:url(https://s3.jp-tok.cloud-objectstorage.appdomain.cloud/unik/about.jpg);

}


#about-head{

    display:flex;     align-items:center;

}


#about-head img{     width:50%;     height:auto;
}

#about-head div{     padding-left:40px;
}

#about-app{
    text-align:center;
}

#about-app .video{     width:70%;     height:100%;     margin:30px auto 0 0;
}

#about-app .video video{     width:100%;     height:100%;     margin-top:20px;
}

#contact-details{     display:flex;     justify-content:space-between;
flexwrap:wrap;
}

#contact-details .details{     width:40%;
}

#contact-details .details span, #form-details form span{     font-size:12px;
}

#contact-details .details h2, #form-details form h2{     font-size:26px;
lineheight:35px; padding:20px 0;
}

#contact-details .details h3{     font-size:16px;     padding-bottom:15px;
}

#contact-details .details li{     list-style:none;     display:flex;     padding:10px;
}

#contact-details .details li i{     font-size:14px;     padding-right:22px;
}

#contact-details .details li p{     font-size:14px;     margin:0;
}
```

```
#contact-details .map{     width:55%;     height:400px;

}


#contact-details .map iframe{     width:100%;     height:100%;

}


#form-details{     display:flex;     justify-content:space-between;     margin:30px;
padding:80px;     border:1px solid #800080;

}


#form-details form{     width:65%;     display:flex;     flex-direction:column;
align-items:flex-start;

}


#form-details form input, #form-details form textarea{     width:100%;
padding:12px 15px;     outline:none;     margin-bottom:20px;
border:1px solid #e1e1e1;

}


#form-details form button{     background-color:#800080;     color:#fff;

}


#form-details .people div{     padding-bottom:25px;

}


#form-details .people div img{     width:65px;     height:65px;     object-fit:cover;
margin-right:15px;

}


#form-details .people div p{     margin:0;     font-size:13px;
lineheight:25px;

}


#form-details .people div p span{     display:block;     font-
size:16px; fontweight:600;  color:#000;
}



/* Cart Page */


#cart table{     width:100%;     border-collapse:collapse;
tablelayout:fixed;      white-space:nowrap;
```

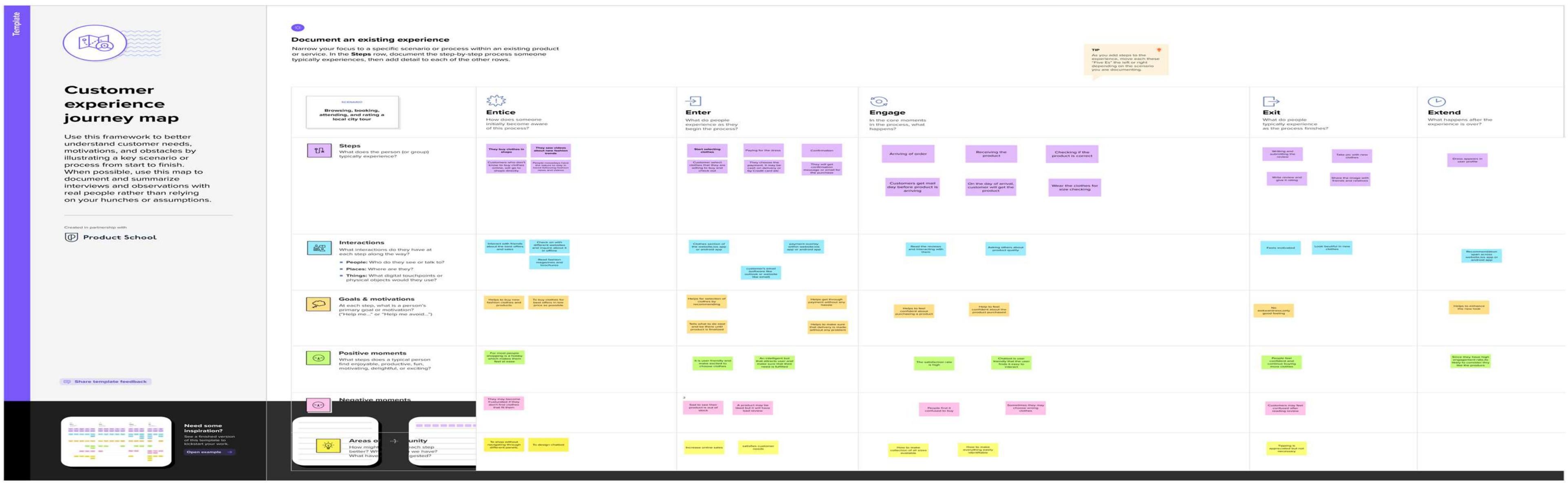
```
}

#cart table img{     width:70px;
}

#cart table td:nth-child(1){     width:100px;     text-align:40px 40px;
}

#cart table td:nth-child(2){     width:150px;     text-align:40px 40px;
}

#cart table td:nth-child(3){     width:250px;     text-align:40px 40px;
}

#cart table td:nth-child(4),
#cart table td:nth-child(5), #cart table td:nth-child(6){     width:150px;
textalign:40px 40px;
}

#cart table td:nth-child(5) input{ width:70px;
                                                        padding:10px 5px 10px 15px;
}

#cart table thead{     border:1px solid #e2e9e1;     border-left:none;     border-
right:none;
}

#cart table thead td{     font-weight:700;     text-transform:uppercase;
fontsize:13px;     padding:18px 0;
}

#cart table tbody tr td{     padding-top:15px;
}

#cart table tbody t{     font-size:13px;
}

#cart-add{     display:flex;     flex-wrap:wrap;     justify-content:space-between;
}

#coupon{     width:50%;     margin-bottom:30px;
```

```
}

#coupon h3, #subtotal h3{     padding-bottom:15px;

}

#coupon input{ padding:10px 20px; outline:none;
width:60%;     margin-right:10px;     border:1px solid
#800080;

}

#coupon button,

#subtotal button, #newsletter button{     background-color:#800080;
color:#fff;     padding:12px 20px;

}


#subtotal{     width:50%;     margin-bottom:30px;     border:1px solid
#e2e9e1;     padding:30px;

}

#subtotal table{     border-collapse:collapse;     width:100%
marginbottom:20px;

}

#subtotal table td{     width:100%;     border:1px solid #800080;
padding:10px;     font-size:13px;

}



/* Register */

/* Add padding to containers */

.container,
.imgcontainer {
padding: 16px;
background-color:
white;

}

#register, #login{   border: 60px solid ;   padding: 15px;

}
```

```
/* Full-width input fields */ input[type=text], input[type=password] {   width:
100%;   padding: 15px;   margin: 5px 0 22px 0;   display: inline-block;   border:
none;   background: #f1f1f1;

}


input[type=text]:focus, input[type=password]:focus {   background-color: #fff;   outline: none;
}


/* Overwrite default styles of hr */ hr {   border: 1px solid #800080;
marginbottom: 25px;

}


/* Set a style for the submit button */
.registerbtn {   background-color: #800080;
  color: white;   padding: 16px 20px;   margin: 8px 0;   border: none;
cursor: pointer;   width: 100%;   opacity: 0.9;

}

.registerbtn:hover {   opacity: 1;

}


/* Add a blue text color to links */

#register a {   color:#800080 ;

}


/* Set a grey background color and center the text of the "sign in" section */

.signin {

  background-color: #f1f1f1;    text-
align: center;
}


/* Login Page */


#login input[type=text], input[type=password] {   width: 100%;   padding: 12px 20px;   margin: 8px
0;   display: inline-block;   border: 1px solid #ccc;   box-sizing: border-box;

}


#login{   border: 60px solid ;   padding: 15px;
border-image: url(img1.jpg) 30 stretch;
}

#login button {   background-color: #800080;

  color: white;   padding: 14px 20px;   margin: 8px 0;   border: none;
cursor: pointer;   width: 100%;
}

button:hover {   opacity: 0.8;
}

.imgcontainer {   text-align: center;   margin: 24px 0 12px 0;
}

img.avatar {   width: 40%;   border-radius: 50%;
}

.container {   padding: 16px;
}

span.psw {    float: right;
padding-top: 16px;
}

/* Blog 1 */

#blog1{
    text-align:center;
}

#blog1 .pro-container{     display:flex;     justify-content:space-between;
paddingtop:20px;     flex-wrap:wrap;
}

#blog1 .pro{     width:23%;     min-width:250px;     padding:10px 12px;
border:1px solid #cce7d0;     border-radius:25px;     cursor:pointer;
    box-shadow:20px 20px 30px rgba(0,0,0,0.02);     margin:15px 0;     transition:
0.2s ease;     position:relative;
}


#blog1 .pro:hover{       box-shadow:20px
20px 30px rgba(0,0,0,0.06);
}

#blog1 .pro img{     width:100%;     border-radius:20px;
}
```

/* Media query*/

7.3 DATABASE SCHEMA



Template

Customer experience journey map

Use this framework to better understand customer needs, motivations, and obstacles by illustrating a key scenario or process from start to finish. When possible, use this map to document and summarize interviews and observations with real people rather than relying on your hunches or assumptions.

Created in partnership with

Product School

Share template feedback

Need some inspiration?

See a finished version of this template to kickstart your work.

Open example →

Document an existing experience

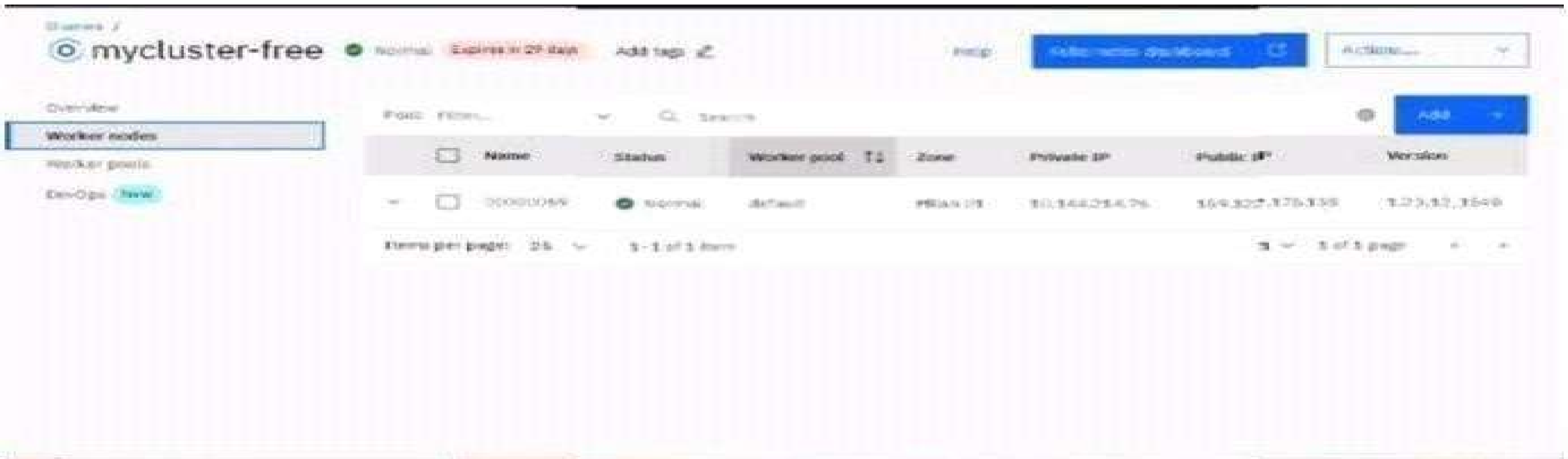
Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.

TIP

As you add steps to the experience, move each these "Five Es" the left or right depending on the scenario you are documenting.

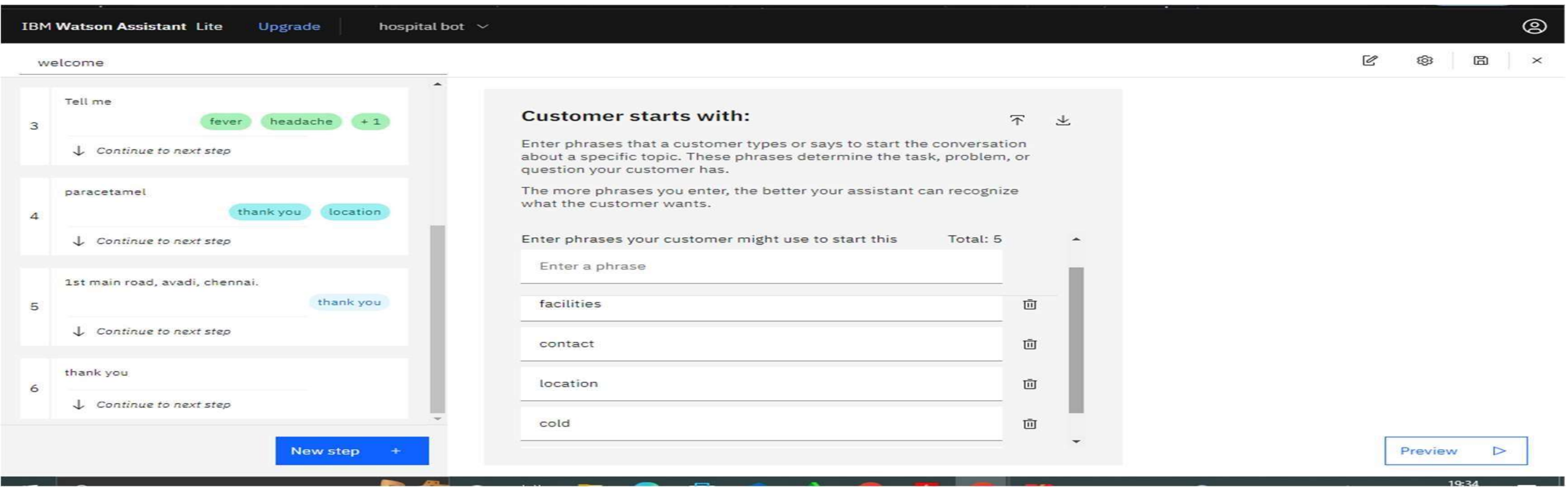
SCENARIO Browsing, booking, attending, and rating a local city tour	Entice How does someone initially become aware of this process?	Enter What do people experience as they begin the process?	Engage In the core moments in the process, what happens?	Exit What do people typically experience as the process finishes?	Extend What happens after the experience is over?
Steps What does the person (or group) typically experience?	They buy clothes in shops They saw videos about new fashion trends Customers who don't know to buy clothes online, will go to shops directly People nowadays have the nature to stay in trend following fashion news and videos	Start selecting clothes Customer select clothes that they are willing to buy and check out Paying for the dress They choose the payment. It may be cash on delivery or by Credit card etc Confirmation They will get confirmation message or email for the purchase	Arriving of order Receiving the product Checking if the product is correct Customers get mail day before product is arriving On the day of arrival, customer will get the product Wear the clothes for size checking	Writing and submitting the review Take pic with new clothes Write review and give it rating Share the image with friends and relatives	Dress appears in user profile
Interactions What interactions do they have at each step along the way? ▪ **People:** Who do they see or talk to? ▪ **Places:** Where are they? ▪ **Things:** What digital touchpoints or physical objects would they use?	Interact with friends about the best offers and sales Check on with different websites and inquire about it in offline Read fashion magazines and brochures	Clothes section of the website,ios app or android app payment overlay within website,ios app or android app customer's email (software like outlook or website like email)	Read the reviews and interacting with them Asking others about product quality	Feels motivated Look beutiful in new clothes	Recommendation span across website,ios app or android app
Goals & motivations At each step, what is a person's primary goal or motivation? ("Help me..." or "Help me avoid...")	Helps to buy new fashion clothes and products To buy clothes for best offers in low price as possible	Helps for selection of clothes by recommending Helps get through payment without any hassle Tells what to do next and be there until product is finalized Helps to make sure that delivery is made without any problem	Helps to feel confident about purchasing a product Help to feel confident about the product purchased	No awkwardness,only good feeling	Helps to enhance the new look
Positive moments What steps does a typical person find enjoyable, productive, fun, motivating, delightful, or exciting?	For most people shopping is a hobby which makes them feel at ease	It is user friendly and make excited to choose clothes An intelligent bot that attracts user and make sure that their need is fulfilled	The satisfaction rate is high Chatbot is user friendly that the user finds it easy to interact	People feel confident and continue buying more clothes	Since they have high engagement rate,its likely to consider they like the products
Negative moments	They may become frustrated if they don't find clothes that fit them	Sad to see their product is out of stock A product may be liked but it will have bad review	People find it confused to buy Sometimes they may choose wrong clothes	Customers may feel confused after reading review	
Areas o unity How might better? Wh What have each step we have? gested?	To shop without navigating through different panels. To design chatbot	Increase online sales satisfies customer needs	How to make collection of all sizes available How to make everything easily identifiable	Tipping is appreciated but not necessary	

INTEGRATING CHATBOT WITH HTML PAGE:



	Name	Status	Worker pool	Zone	Private IP	Public IP	Version
☐	[illegible]	● Normal	default	[illegible]	10.144.214.76	169.[illegible].175.135	1.23.12_1549

Items per page: 25 1 - 1 of 1 items 1 1 of 1 page



3 Tell me — fever, headache, + 1 — Continue to next step

4 paracetamel — thank you, location — Continue to next step

5 1st main road, avadi, chennai. — thank you — Continue to next step

6 thank you — Continue to next step

Customer starts with:

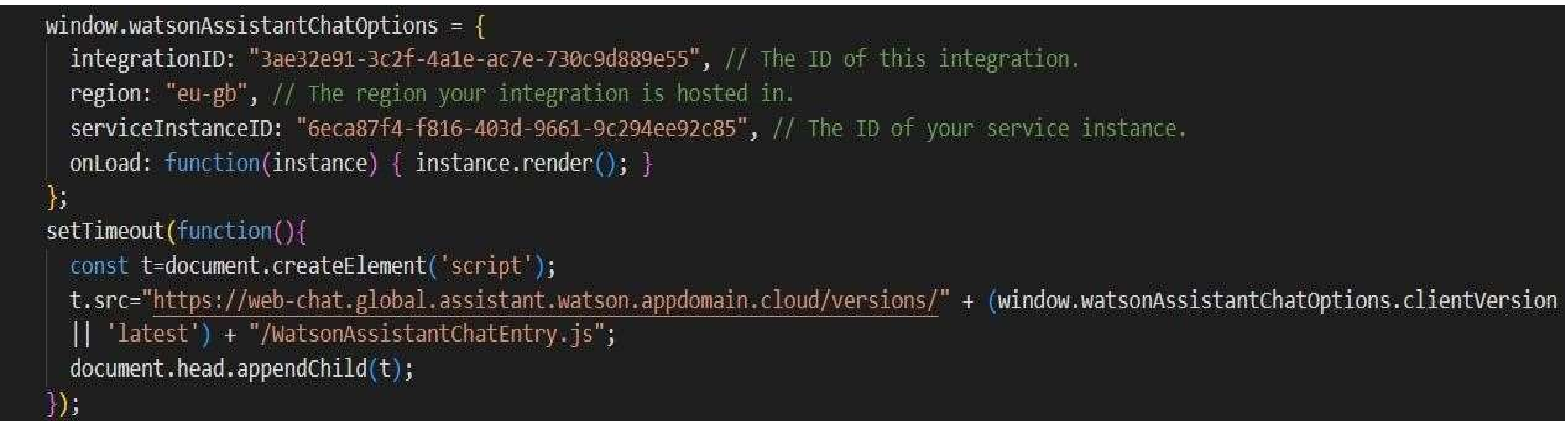
Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

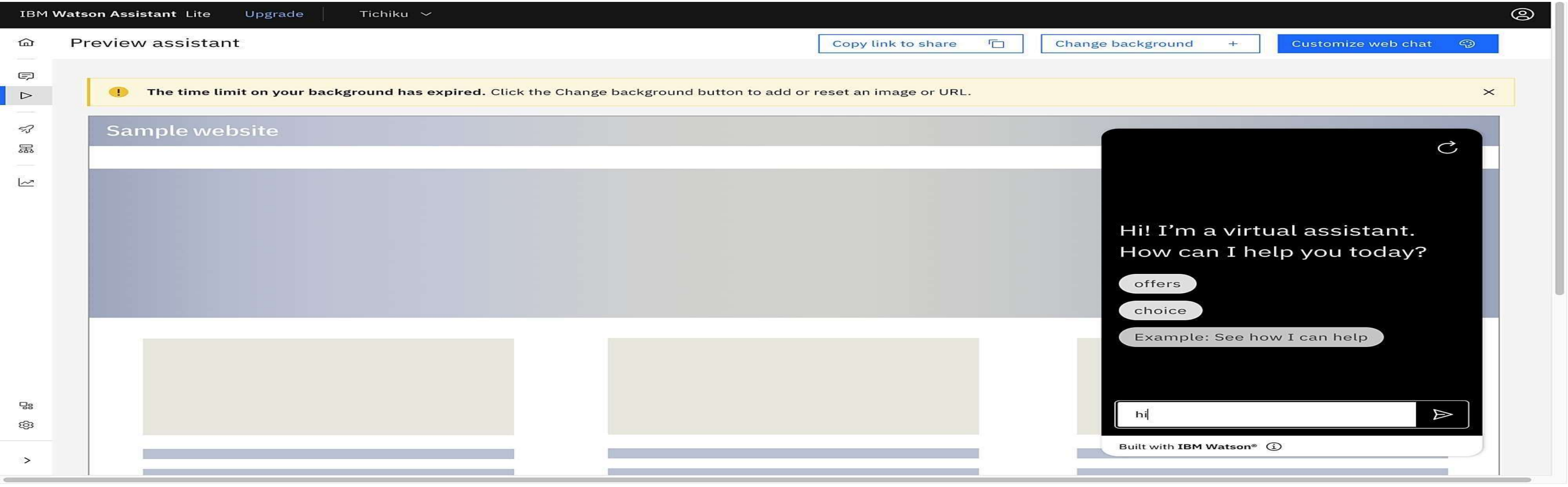
Enter phrases your customer might use to start this — Total: 5

Enter a phrase

- facilities
- contact
- location
- cold



```
window.watsonAssistantChatOptions = {
  integrationID: "3ae32e91-3c2f-4a1e-ac7e-730c9d889e55", // The ID of this integration.
  region: "eu-gb", // The region your integration is hosted in.
  serviceInstanceID: "6eca87f4-f816-403d-9661-9c294ee92c85", // 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);
});
```



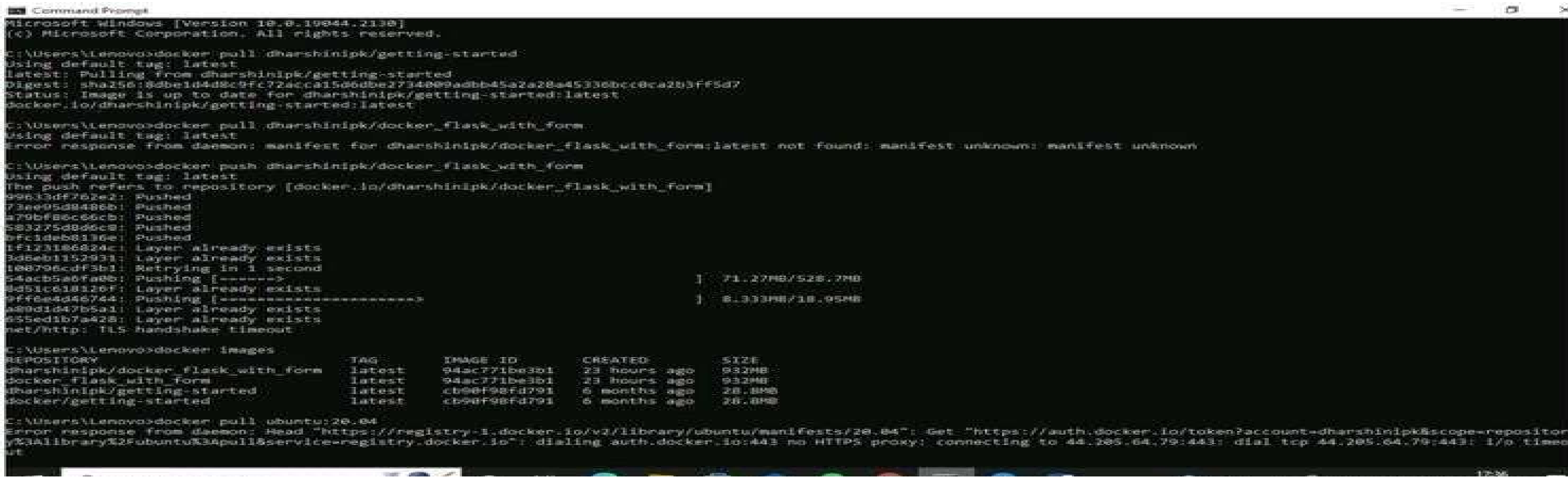
The time limit on your background has expired. Click the Change background button to add or reset an image or URL.

Hi! I’m a virtual assistant. How can I help you today?

- offers
- choice
- Example: See how I can help

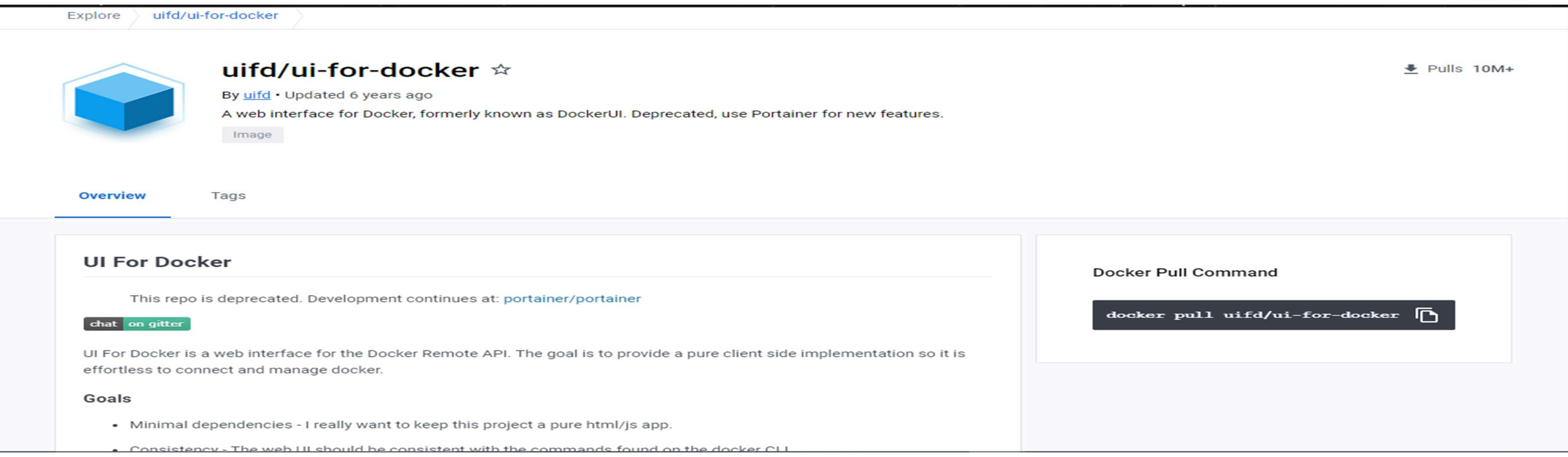
hi

DOCKER PUSH:



```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Lenovo>docker pull dharshinipk/getting-started
Using default tag: latest
latest: Pulling from dharshinipk/getting-started
Digest: sha256:8dbe1d4d8c9fc72acca15d6dbe2734009adbb45a2a28a45336bcc0ca2b3ff5d7
Status: Image is up to date for dharshinipk/getting-started:latest
docker.io/dharshinipk/getting-started:latest

C:\Users\Lenovo>docker pull dharshinipk/docker_flask_with_form
Using default tag: latest
Error response from daemon: manifest for dharshinipk/docker_flask_with_form:latest not found: manifest unknown: manifest unknown

C:\Users\Lenovo>docker push dharshinipk/docker_flask_with_form
Using default tag: latest
The push refers to repository [docker.io/dharshinipk/docker_flask_with_form]
99633df762e2: Pushed
73ee95d84866b: Pushed
a79bf86c66cb: Pushed
583275d8d6c0: Pushed
bfc1deb8136e: Pushed
1f123186824c: Layer already exists
3d6eb1152931: Layer already exists
100796cdf3b1: Retrying in 1 second
54acb5a6fa0b: Pushing [------->                                              ]  71.27MB/528.7MB
8d51c618126f: Layer already exists
9ff6e4d46744: Pushing [------------------------>                             ]  8.333MB/18.95MB
a09d1d47b5a1: Layer already exists
655ed1b7a428: Layer already exists
net/http: TLS handshake timeout

C:\Users\Lenovo>docker images
REPOSITORY                          TAG       IMAGE ID       CREATED          SIZE
dharshinipk/docker_flask_with_form  latest    94ac771be3b1   23 hours ago     932MB
docker_flask_with_form              latest    94ac771be3b1   23 hours ago     932MB
dharshinipk/getting-started         latest    cb90f98fd791   6 months ago     28.8MB
docker/getting-started              latest    cb90f98fd791   6 months ago     28.8MB

C:\Users\Lenovo>docker pull ubuntu:20.04
Error response from daemon: Head "https://registry-1.docker.io/v2/library/ubuntu/manifests/20.04": Get "https://auth.docker.io/token?account=dharshinipk&scope=repositor
y%3Alibrary%2Fubuntu%3Apull&service=registry.docker.io": dialing auth.docker.io:443 no HTTPS proxy: connecting to 44.205.64.79:443: dial tcp 44.205.64.79:443: i/o timeo
ut
```

CONTAINERIZING THE APPLICATION:



uifd/ui-for-docker ☆

By uifd • Updated 6 years ago

A web interface for Docker, formerly known as DockerUI. Deprecated, use Portainer for new features.

UI For Docker

This repo is deprecated. Development continues at: portainer/portainer

chat on gitter

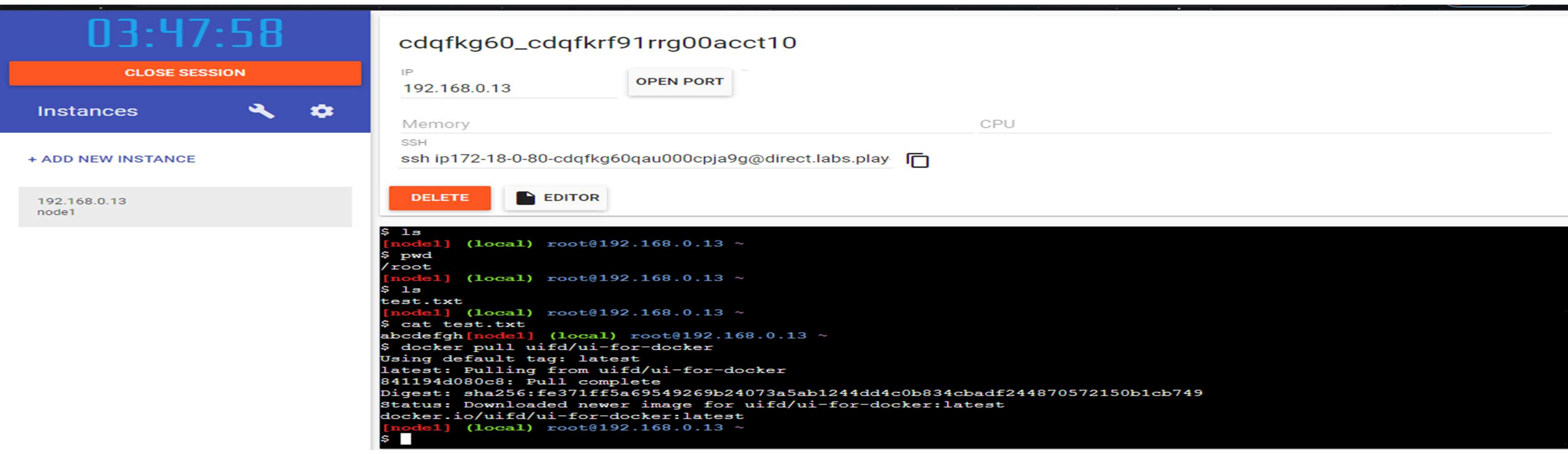
UI For Docker is a web interface for the Docker Remote API. The goal is to provide a pure client side implementation so it is effortless to connect and manage docker.

Goals

- Minimal dependencies - I really want to keep this project a pure html/js app.
- Consistency - The web UI should be consistent with the commands found on the docker CLI

Docker Pull Command

```
docker pull uifd/ui-for-docker
```

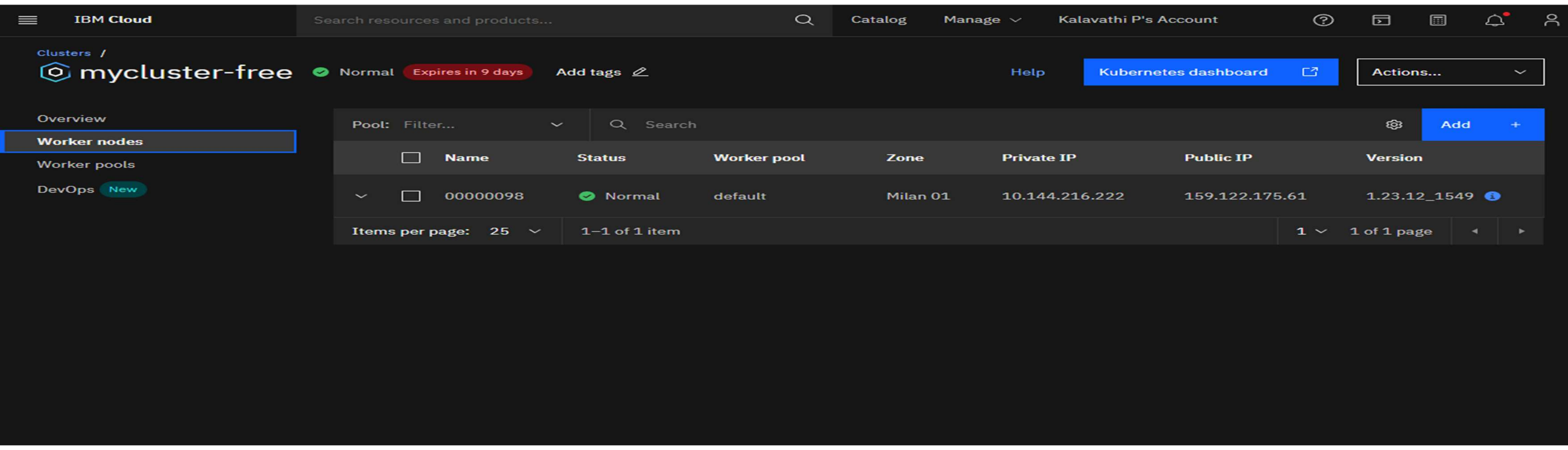


cdqfkg60_cdqfkrf91rrg00acct10

IP: 192.168.0.13

SSH: ssh ip172-18-0-80-cdqfkg60qau000cpja9g@direct.labs.play

```
$ ls
[node1] (local) root@192.168.0.13 ~
$ pwd
/root
[node1] (local) root@192.168.0.13 ~
$ ls
test.txt
[node1] (local) root@192.168.0.13 ~
$ cat test.txt
abcdefgh[node1] (local) root@192.168.0.13 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
latest: Pulling from uifd/ui-for-docker
841194d080c8: Pull complete
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[node1] (local) root@192.168.0.13 ~
$ 
```



□	Name	Status	Worker pool	Zone	Private IP	Public IP	Version
□	00000098	Normal	default	Milan 01	10.144.216.222	159.122.175.61	1.23.12_1549

8.TESTING

8.1 TEST CASE

Test cases define what must be done to test a system, including the steps executed in the system, the input data values that are entered into the system and the results that are expected throughout test case execution. Using test cases allows developers and testers to discover errors that may have occurred during development or defects that were missed during ad hoc tests. The benefits of an effective test case include

- Guaranteed good test coverage.

- Reduced maintenance and software support costs.

- Reusable test cases.

- Confirmation that the software satisfies end-user requirements.

- Improved quality of software and user experience.

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

8.2 User Acceptance Testing

Following are the entry criteria for User Acceptance Testing: • Business Requirements must be available.

- Application Code should be fully developed

- Unit Testing, Integration Testing & System Testing should be completed

- No Showstoppers, High, Medium defects in System Integration Test Phase

- Only Cosmetic error is acceptable before UAT

- Regression Testing should be completed with no major defects

- All the reported defects should be fixed and tested before UAT

- Traceability matrix for all testing should be completed

- UAT Environment must be ready Sign off mail or communication from System Testing Team that the system is ready for UAT execution

8. RESULTS

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 filtering technique. Root Mean Square Error (RMSE), Receiver Operating Characteristics (ROC), Area Under Cover (AUC), Precision, Recall and F1 score is generally used to evaluate the performance or accuracy of the recommendation algorithms.

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

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

where, N_p is the total number of predictions, p_{ui} is the predicted rating that a user *u* will select an item *i* and r_{ui} 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 defined as the fraction of correct recommendations or predictions (known as True Positive) to the total number of correct relevant recommendations provided, which can be 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 number of relevant items expressed as percentages.

F1 Score. F1 score is an indicator of the accuracy of the model and ranges from 0 to 1, where a value close to 1 represents higher recommendation or prediction accuracy. It represents precision and recall as a single metric 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 which are recommended by the algorithm among all of the items.

Accuracy. Accuracy can be defined as the ratio of the number of total correct recommendations to the total recommendations provided.

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- Smart fashion recommender application is the user friendly.
- With the help of chatbot user can find the products very easily.
- This application used to discover the product based on the user’s choice, very easily and quickly.
- It has the ability to reduce transaction costs for consumers, and increase revenue for retailers.

DISADVANTAGES:

- It needs active internet connection.
- Privacy concerns.
- Too many choices.
- 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 completely on 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 provide the best recommendations based on the user’s wardrobe. Since the system is implemented as a website, it is very easy for the end users to access as well as use. The scope of this system can be expanded by including the ability to detect the various design 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**.** Users can have multiple shipping and billing information saved. During checkout they can use the drag and drop feature to select shipping and billing information.

13.APPENDIX

SOURCE CODE

```
.Py  code  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-
8bf9-
0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32459;Security=SSL;SSLServer
Certificate=DigiCertGlobalRootCA.crt;UID=dpk79343;PWD=29Jm7EbzOubtoerk"
,",")


@app.route('/')


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




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



   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']             mesg = 'Logged in
successfully !'


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





@app.route('/Register', methods =['GET', 'POST']) def registet():
    mesg = ''

    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:               mesg = 'Account already exists !'          elif
not re.match(r'[^@]+@[^@]+\.[^@]+', email):               mesg = 'Invalid email address !'
elif not re.match(r'[A-Za-z0-9]+', username):
            mesg = '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)            mesg = 'You have successfully registered !'     elif request.method
== 'POST':          mesg = '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():       mesg = ''        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:

            mesg = '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)          mesg = 'You have successfully applied
for job !'          session['loggedin'] = True

        TEXT = "Hello,a new application for job position" +jobs+"is requested"
```

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


@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('username',
None)    return render_template('home.html')


if __name__ == '__main__':    app.run(host='0.0.0.0')
```

HOME PAGE USING HTML

```
<html>


<head>
    <title> JMTG </title>


</head>
<style>
    * {        margin: 0;        padding: 0;        font-family: "Century Gothic", CenturyGothic, AppleGothic, sans-serif;
    }


    .main {        width: 100%;        background: linear-gradient(to top, rgba(219, 18, 18, 0.5), rgba(0, 0, 0, 0.5)50%); background-
position: center;  background-size: cover;        height: 100%;        font-family: "Century Gothic", CenturyGothic, AppleGothic,
sans-serif;
    }


    .navbar {        width: 100%;        height: 75px;        margin: auto;
    }
    .icon {          width: 200px;
float: left;          height: 70px;
    }


    .logo {        color: rgb(98, 246, 152);        font-size: 35px;        padding-left:

20px;

        float: left;        padding-top: 10px;

    }


    .menu {          width:
400px; float: left;
```

```
      height: 70px;



   ul {        float: left;        display: flex;        justify-content: center;
align-items: center;
   }
   ul li {        list-style: none;        margin-left:
62px;
margin-top: 27px;        font-size: 14px;
   }
   ul li a {
       text-decoration: none;
       color: #ffff;         font-weight: bold;
transition: 0.4s ease-in-out;
   }

   ul li a:hover {        color: rgb(98, 246, 152);
   }

   .search {          width: 330px;
float: left;        margin-left: 270px;
   }

   .srch {        width: 200px;        height: 40px;        background:
transparent;        border: 1px solid rgb(98, 246, 152);        margin-top: 13px;
color: #ffff;         border-right: none;        font-size: 16px;        float: left;
padding: 10px;        border-bottom-left-radius: 5px;        border-top-left-
radius:
5px;
   }

   .btn {        width: 100px;        height: 40px;        background:
rgb(98, 246, 152);        border: 2px solid rgb(98, 246, 152);
margin-top: 13px;          color: #ffff;        font-size: 15px; border-
bottom-right-radius: 5px; borderbottom-right-radius: 5px;



   .btn:focus {        outline: none;
   }

   .srch:focus {        outline: none;
   }


   .content {        width: 1200px;        height: auto;
margin: auto;

      color: #ffff;        position: relative;

   }


   .content.par {        padding-left: 20px;        padding-bottom: 25px;        letter-
spacing: 1.2px;        line-height: 30px;
   }


   .content h1 {        font-size: 50px; padding-left: 20px;
margintop: 9%; letter-spacing: 2px;

}
   .content .cn {        width: 160px;        height: 40px;
background: rgb(98, 246, 152);        border: none;        margin-
bottom: 10px;        margin-left: 20px;        font-size: 18px;
border-radius: 10px;        cursor: pointer;        transition: .4s ease;
   }


   .content .cn a {        text-decoration: none;        color: #000;
transition: .3s ease;
   }


   .cn:hover {        background-color: #fff;

   }


   .content span { color: rgb(98, 246, 152); font-size: 60px;




   .form {        width: 250px;        height: 380px;        background: linear-gradient(to top, rgba(236, 113, 113, 0.8)50%,
rgba(0, 0, 0,

0.8)50%);

      position: absolute;        top: -20px;        left:
870px;

border-radius: 10px;        padding: 25px;

   }
   .form h2 {        width: 220px;        text-align: center;        color:

rgb(206, 91, 187);        font-size: 22px;        border-radius: 10px;
margin: 2px;        padding: 8px;
   }
```

```
 .form input {         width: 240px;         height: 35px;         background:
transparent;
 }


 .form input {         width: 240px;         height: 35px;         background:
transparent;         border-bottom: 1px solid rgb(218, 91, 144);         border-top:
none;         border-right: none;         border-left: none;         color: #fff;         font-
size: 15px;         letter-spacing: 1px;         margin-top: 30px;
 }


 .form input:focus {         outline: none;
 }


  ::placeholder {
    color: #fff;
 }


 .btnn {         width: 240px;         height: 40px;         background: rgb(98,
246, 152);         border: none; margin-top: 30px;
```

```
        font-size: 18px;         border-radius: 10px;         cursor: pointer;
color:
#fff;         transition: 0.4s ease;
    }


    .btnn:hover {         background: #fff;         color: rgb(224, 113, 187);
    }


    .btnn a {         text-decoration: none;         color: rgb(15, 2, 2);         font-
weight: bold;
    }
    .form .link {         font-size: 17px;         padding-top: 20px;         text-align:
center;
    }


    .form .link a {         text-decoration: none;         color: rgb(98, 246, 152);
    }


    .liw {
        padding-top: 15px;         padding-bottom: 10px;
text-align: center;
    }
</style>


<body>


    <div class="main">
        <div class="navbar">
            <div class="icon">
                <h2 class="logo">Smart fashion</h2>
            </div>


            <div class="menu">
                <ul>
                    <li><a href="#">HOME</a></li>
                    <li>
                        <a href="#">CONTACT</a>
                    </li>
                    <li><a href="#">ABOUT US</a></li>


                </ul>

            </div>


            <div class="search">

                <input class="srch" type="search" name="" placeholder="Type to
search">

                <a href="#"><button class="btn">Search</button></a>

            </div>


        </div>




    </div>


</body>


</html>
```

INDEX PAGE:

```
 <html>


<head>

   <title> JMTG</title>


</head>

<style>

    * {         margin: 0;         padding: 0;         font-family: "Century Gothic", CenturyGothic, AppleGothic, sans-serif;

    }


    .main {         width: 100%;         background: linear-gradient(to top, rgba(219, 18, 18, 0.5), rgba(0, 0, 0, 0.5)50%);
backgroundposition: center;         background-size: cover;         height: 100%;         font-family: "Century Gothic",
CenturyGothic, AppleGothic, sans-
serif;

    }
    .navbar {         width: 100%;         height: 75px;         margin: auto;

    }


    .icon {         width: 200px;

        float: left;

        height: 70px;

    }


    .logo {         color: rgb(98, 246, 152);         font-size: 35px;         padding-
left:

20px;

        float: left;         padding-top: 10px;

    }


    .menu {         width: 400px;         float: left;
height: 70px;
    }

    ul {         float: left;         display: flex;         justify-content: center;
align-

items: center;

    }

    ul li {         list-style: none;         margin-left: 62px;         margin-top: 27px;

font-size: 14px;

    }

        ul
li a {

        text-decoration: none;

        color: #ffff;

        font-weight: bold;         transition: 0.4s ease-in-out;

    }


    ul li a:hover {         color: rgb(98, 246, 152);

    }


    .search {         width: 330px;
float: left;         margin-left: 270px;
    }
    .srch {         width: 200px;         height: 40px;         background: transparent;         border:
```

```
1px solid rgb(98, 246, 152);      margin-top: 13px;      color: #ffff;
border-right: none;      font-size: 16px;
      float: left;

      padding: 10px;      border-bottom-left-radius: 5px;      border-top-left-radius: 5px;

   }


   .btn {      width: 100px;      height: 40px;      background: rgb(246, 98,
221);      border: 2px rgb(246, 98, 221);      margin-top: 13px;      color:
#ffff;      font-size:
15px;      border-bottom-right-radius: 5px;      border-bottom-right-radius:
5px;

   }


   .btn:focus {      outline: none;

   }


   .srch:focus {      outline: none;

   }
   .content {      width: 1200px;      height: auto;      margin: auto;
color: rgb(161, 144, 144);      position: relative;

   }


   .content.par {      padding-left: 20px;      padding-bottom: 25px;
letterspacing: 1.2px;      line-height: 30px;
   }


   .content h1 {      font-size: 50px;      padding-left: 20px;
margintop: 9%;      letter-spacing: 2px;
   }


   .content .cn {      width: 160px;      height: 40px;      background:
rgb(246, 98, 221);      border: none;      margin-bottom: 10px;      margin-
left: 20px;      fontsize: 18px;      border-radius: 10px;      cursor: pointer;
transition: .4s ease;
   }


   .content .cn a {      text-decoration: none;      color: #000;
transition: .3s ease;
   }


   .cn:hover {      background-color: #000;
   }


   .content span {        color: rgb(246, 98, 221);        font-size: 60px;

   }


   .form {        width: 250px;        height: 380px;        background: linear-gradient(to top, rgba(243, 241, 239, 0.8), rgba(252, 252,
252, 0.8)50%);        position: absolute;        top: -20px;        left: 870px;        border-radius: 10px;        padding: 25px;
   }
   .form h2 {        width: 220px;        text-align: center;        color:

rgb(235, 226, 100);        font-size: 22px;        border-radius: 10px;
margin: 2px;        padding: 8px;
   }
   .form input {        width: 240px;        height: 35px;        background:

transparent;

   }


   .form input {        width: 240px;        height: 35px;        background: transparent;
borderbottom: 1px solid rgb(246, 98, 221);        border-top: none;        border-right: none;
borderleft: none;        color: rgb(14, 12, 12);        font-size: 15px;        letter-spacing: 1px;
margin-top: 30px;
   }


   .form input:focus {        outline: none;

   }


    ::placeholder {        color: #000;

   }


   .btnn {        width: 240px;        height: 40px;        background: rgb(22, 21,
22);        border: none;        margin-top: 30px;        font-size: 18px;
border-radius: 10px;        cursor: pointer;        color: rgb(105, 100, 100);
transition: 0.4s ease;
   }


   .btnn:hover {        background: #000;        color: rgb(235, 201, 229);

   }
   .btnn a {        text-decoration: none;        color: white;        font-weight:

bold;

   }
   .form .link {        font-size: 17px;        padding-top: 20px;        text-align:

center;
```

```
    }


    .form .link a {         text-decoration: none;          color: rgb(246, 98, 221);

    }


    .liw {

        padding-top: 15px;         padding-bottom: 10px;

        text-align: center;

    }

</style>


<body>


    <div class="main">

        <div class="navbar">

            <div class="icon">

                <h2 class="logo">Smart Fashion</h2>

            </div>


            <div class="menu">

                <ul>

                    <li><a href="#">HOME</a></li>

                    <li>

                        <a href="#">CONTACT</a>

                    </li>

                    <li><a href="#">ABOUT US</a></li>


                </ul>

            </div>


            <div class="search">

                <input class="srch" type="search" name="" placeholder="Type to
search">

                <a href="#"><button class="btn">Search</button></a>

            </div>


        </div>
    <div class="content">
      <h1>SMART FASHION <br><span>Application</span></h1>

      <div class="form">
        <h2>SMART LOGIN</h2>
        <form action="/Login" method="post">
          <input type="text" name="username" placeholder="Enter the Username">
          <input type="password" name="password" placeholder="Enter the Password ">
          <button type="submit" class="btnn"><a
href="#">Login</a></button>

          <p class="link">Don't have an account<br>
            <a href="/Register">Sign up </a> here</a>
          </p>

        </form>

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

</body>

</html>
```

GITHUB LINK: https://github.com/IBM-EPBL/IBM-Project-1591-1658401072

DEMO LINK: https://github.com/IBM-EPBL/IBM-Project-1591-1658401072