SMART FASHION RECOMMENDER APPLICATION WITH CHATBOT

PROJECT REPORT

Submitted by

HEMA S – AC19UEC057

MANISHA V P - AC19UEC076

MONISHA K- AC19UEC079

RAAGAMALIGA B - AC19UEC098

In partial fulfilment of the award of the degree

of

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATION ENGINEERING

ADHIYAMAAN COLLEGE OF ENGINEERING

DR.M.G.R NAGAR, HOSUR-635130

ANNA UNIVERSITY: CHENNAI 600025

NOVEMBER 2022

SMART FASHION RECOMMENDER APPLICATION

CONTENTS

ABSTRACT

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. 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. With technological advancements, this branch of artificial intelligence exhibits a tremendous amount of potential in image processing, parsing, classification, and segmentation. Despite its huge potential, the number of academic articles on this topic is limited. The available studies do not provide a rigorous review of fashion recommendation systems and the corresponding filtering techniques. To the best of the authors' knowledge, this is the first scholarly article to review the state-of-the-art fashion recommendation systems and the corresponding filtering techniques. In addition, this review also explores various potential models that could be implemented to develop fashion recommendation systems in the future. This paper will help researchers, academics, and practitioners who are interested in machine learning, computer vision, and fashion retailing to understand the characteristics of the different fashion recommendation systems.

CHAPTER 1

INTRODUCTION

1.1 Project Overview

Humans are inevitably drawn toward something that is visually more attractive. This tendency of humans has led to the development of the fashion industry over time. With the introduction of recommender systems in multiple domains, retail industries are coming forward with investments in the latest technology to improve their business. Fashion has been in existence for centuries and will be prevalent in the coming days as well. Women are more correlated with fashion and style, and they have a larger product base to deal with making it difficult to take decisions. It has become an important aspect of life for modern families since a person is more often than not judged based on his attire. Moreover, apparel providers need their customers to explore their entire product line so they can choose what they like the most which are not possible by simply going
into a cloth store

Analytics is important to every business as they dissect sales into meaningful variables. Customer segmentation is a key aspect of the personalization of the product. How the customer interacts with offers, discounts, new product launches, pricing, coupons, trend, and season plays a huge role in segmentation. Here consumers are grouped based on preferences and likes. Products can also be grouped and mapped based on consumer behavior thanks to the
vectorization of words

Chatbots are great at communicating with consumers at scale. They redirect consumers to appropriate locations to solve their queries. A sales bot converts a potential customer into a sale by patronizing the customers. They provide a customized solution to end consumers from previous sales analytics.

By coupling product recommenders with Chatbots, customers can get a personalized response, they can also aid data collection that can be deployed in the analysis of consumer behavior, chats can be used to analyze the sentiment, collect customer
feedback on the product.

1.2 PROPOSE

This is an e-commerce website project, the backend was written by Flask (Python) and the frontend used Bootstrap. In addition to an easy-to-use and robust web interface, we also integrated some AI technologies to make the project unique. For example, there is a recommendation system to provide useful feedback on what a user might potentially want to buy, based on their previous choices. A chatbot is also available to provide a sense of personalized shopping and seamless service.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing problem

In shopping apps navigating to the desired product is very difficult and no proper assistance will be provided.

● Which you can directly do your online shopping based on your choice without any search.

● Instead of navigating to several screens for booking products online the user can directly talk to Chatbot regarding the products.

2.2 References

Cristiana Stan et al. Creating an outfit is a problem based on the preferences of each person and it can be difficult even for the best experts. This paper presents an automation system that can recommend a full outfit based on fashion recommendations. there are two convolutional neural networks based on the Alex Net model that is used to identify cloth items and attributes associated with each item they are continuously updated to obtain a recommendation that is more suitable for each user

WEIQIAN LI et al [1]. Nowadays rapid growth of fashion e-commerce, fashion recommendation has become a main digital marketing tool built for customer reviews, ratings, and feedback. online feedback and review is a powerful source and this is useful for enhancing the product of fashion recommendations for future purchases. the effectiveness in customer rating predictions as compared red several states of fashion recommenders.

Hindi et al. Nowadays people are higher shopping attractive and increased demand for garments. People are facing too many garments and consumers need to try them on repeatedly, thus fashion evaluation methods based on appearance in the paper and the experimental results show that consumer's fashion level can be accurately described basic on the index of appearance of the approach higher level application value for the fashion recommendation system.

Tariq Hussain et al. In recent days larger amount of information and rules of the internet service, phenomenon leads to extremely low utilization of information also known as the information overload problem, such the images are used to traditionally & keywords, any picture object can be represented by its specific colour, shape and texture, this experimental model shows and achieve better than the ones of the previous techniques

Pankaj Agarwal et al. In the fashion e-commerce platforms, there are many components to use the platform for the user. Product discovery is one of the key components of the user’s good experience and also collaborative filtering-based approach is popular now a day's, it is for recommending non- personalizing the product and also for q

Peng Peng Zhao et al. Online shopping is now a well-developed platform to purchase a product. For fashion recommendations using the image is getting more attention now a day's, so the visual and textual jointly enhanced interpretable(VTJEI) model is used for these product recommendations based on the image and review

Victoria Organotin et al. Chatbots are used in many fields, for the e-commerce chatbot we aim at developing a web-based chatbot called Covenant university community mall(CUSM). The e-commerce chatbot application for CUSM seeks to provide an easy, smart and comfortable shopping experience

Vitaliy Husak et al. the work is devoted to the development of creating a list of recommendations for the style of clothing. It provides practical use of chatbots as virtual assistants involve natural language processing, so that will helps to archive a high level of natural language recognition that the communication is connected

Yashar Deldjoo et al. The recommender systems help users navigate the collection of products to find the items and followed or ignored webpage visits to determine, Due to Market dynamics and customer preferences this is the challenge the usage of traditional recommender syste

Neera Sanjay Agashe et al. Data has no research and they include e-commerce and the system has significance. It is one of the large scales of makes a scope and domains and improves the semantic analysis of provider outputs and tools, websites, etc. If the analysis is taking a minimal use

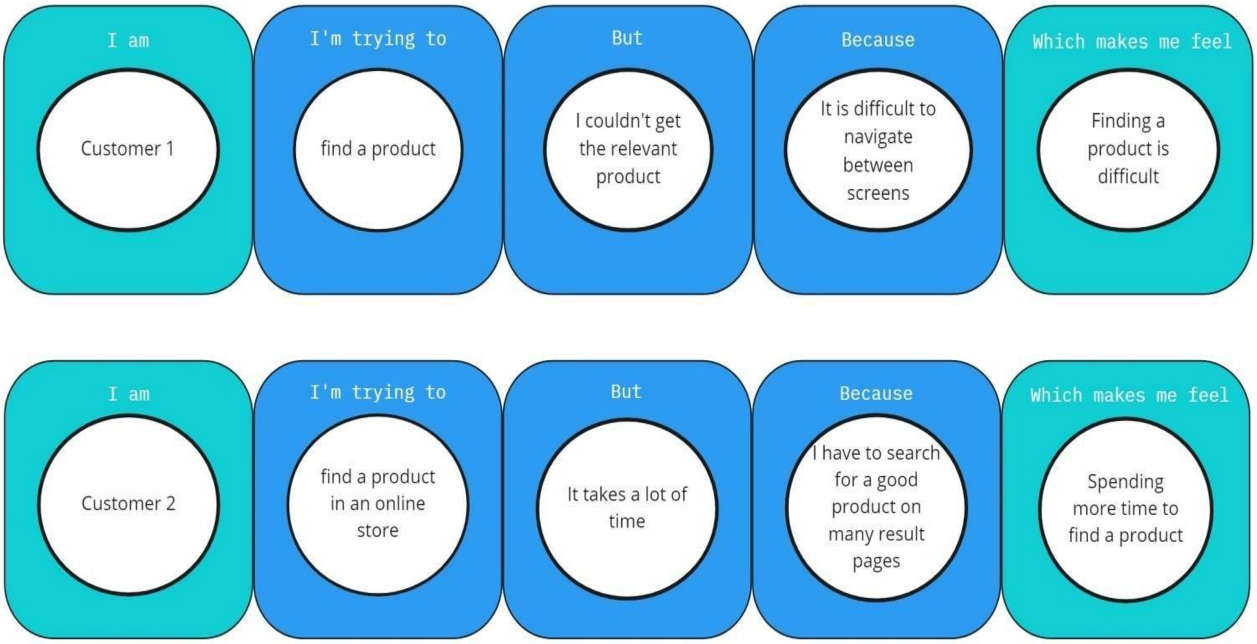
2.3 PROBLEM STATEMENT DEFINITION

Customer Problem Statement Template:

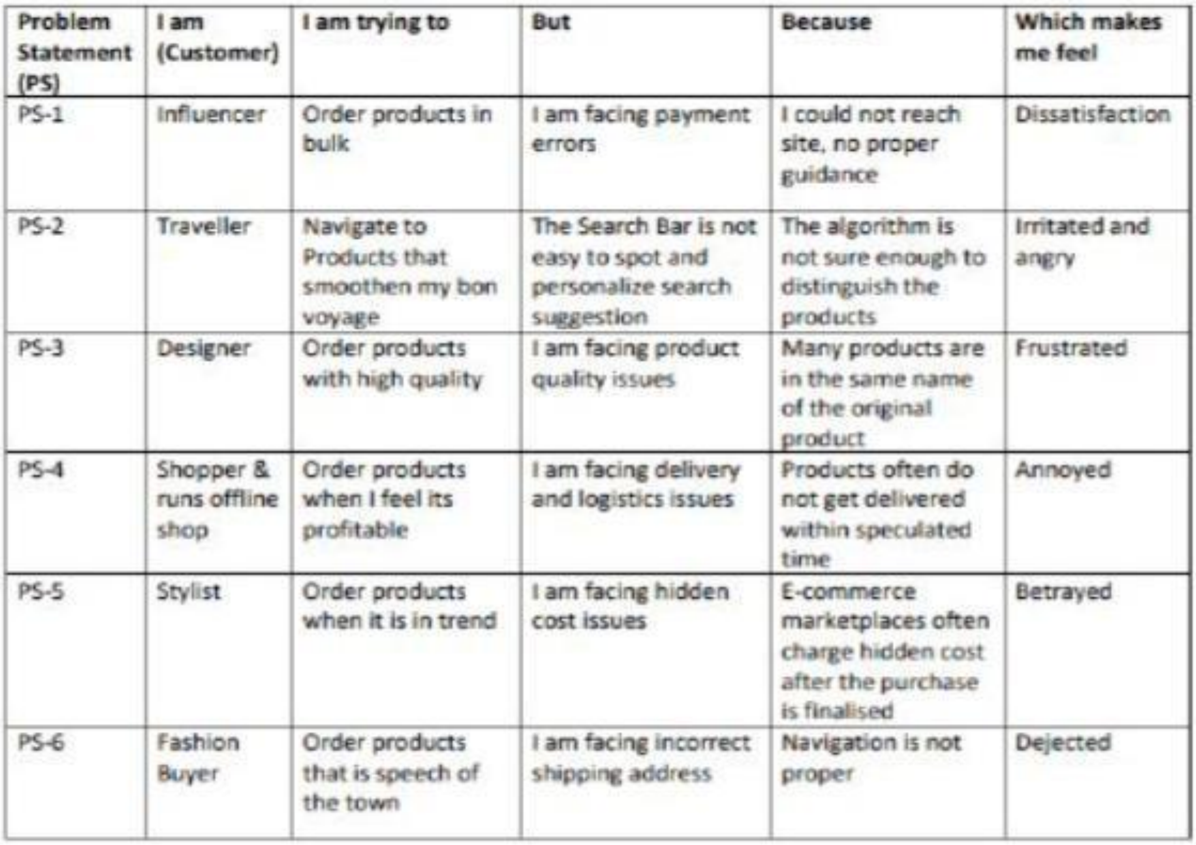
Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

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

Example:



I am	I'm trying to	But	Because	Which makes me feel
Customer 1	find a product	I couldn't get the relevant product	It is difficult to navigate between screens	Finding a product is difficult
Customer 2	find a product in an online store	It takes a lot of time	I have to search for a good product on many result pages	Spending more time to find a product



Problem Statement (PS)	I am (Customer)	I am trying to	But	Because	Which makes me feel
PS-1	Influencer	Order products in bulk	I am facing payment errors	I could not reach site, no proper guidance	Dissatisfaction
PS-2	Traveller	Navigate to Products that smoothen my bon voyage	The Search Bar is not easy to spot and personalize search suggestion	The algorithm is not sure enough to distinguish the products	Irritated and angry
PS-3	Designer	Order products with high quality	I am facing product quality issues	Many products are in the same name of the original product	Frustrated
PS-4	Shopper & runs offline shop	Order products when I feel its profitable	I am facing delivery and logistics issues	Products often do not get delivered within speculated time	Annoyed
PS-5	Stylist	Order products when it is in trend	I am facing hidden cost issues	E-commerce marketplaces often charge hidden cost after the purchase is finalised	Betrayed
PS-6	Fashion Buyer	Order products that is speech of the town	I am facing incorrect shipping address	Navigation is not proper	Dejected

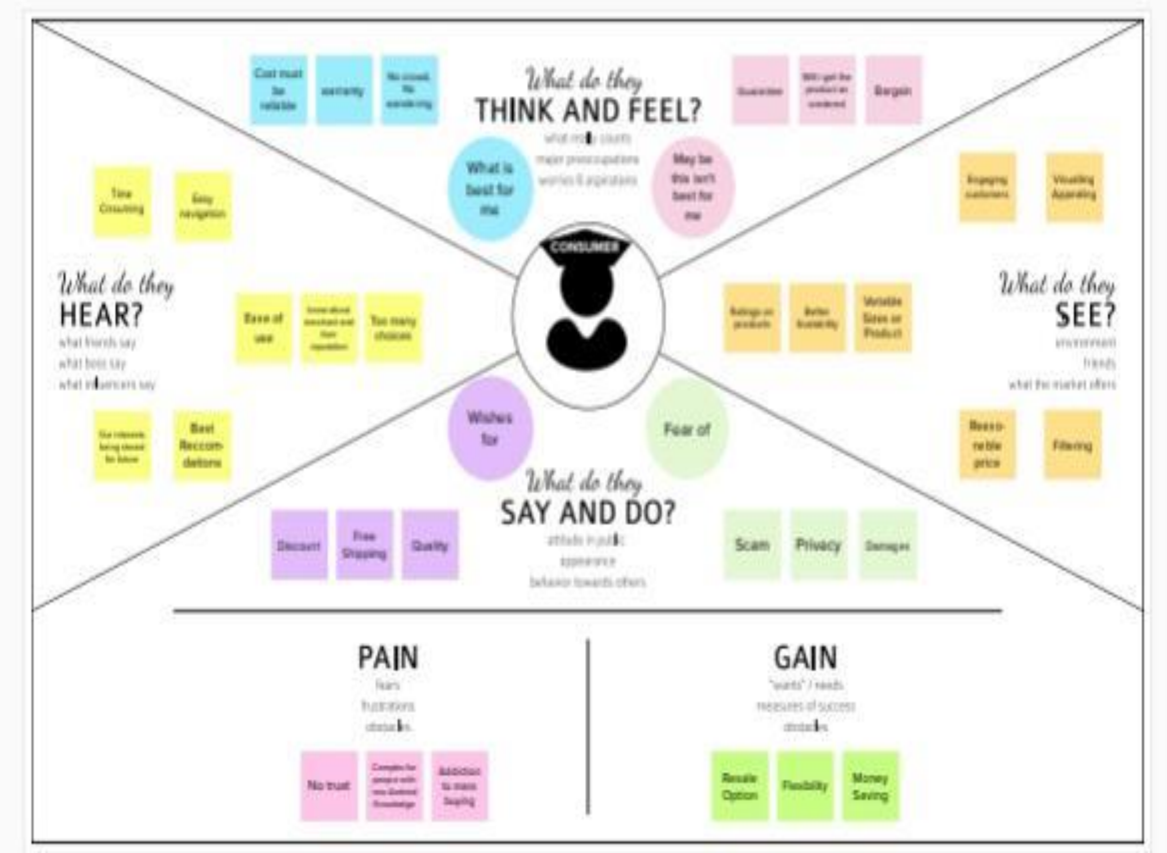
CHAPTER 3

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 behaviors 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.

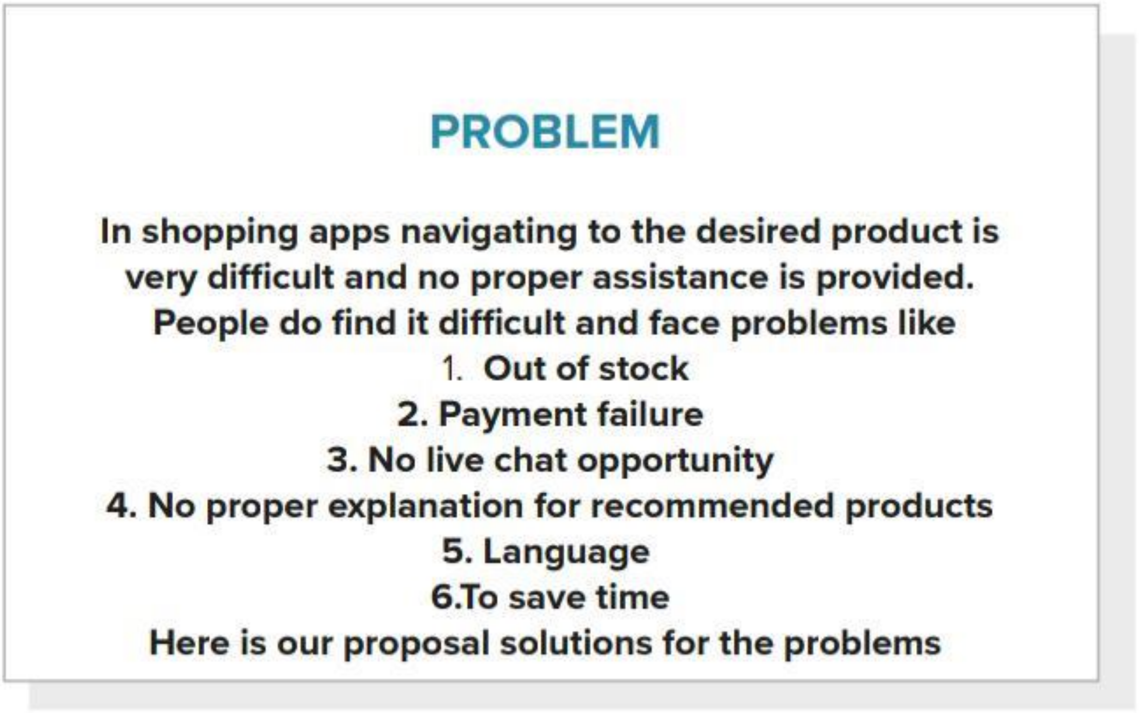

What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Cost must be reliable
warranty
Bargain
What is best for me
May be this isn't best for me
CONSUMER
What do they
HEAR?
what friends say
what boss say
what influencers say
Time Consuming
Easy navigation
Ease of use
Too many choices
Best Reccom-dations
What do they
SEE?
environment
friends
what the market offers
Better Availability
Variable Sizes or Product
Reaso-nable price
Filtering
Wishes for
Fear of
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Discount
Free Shipping
Quality
Scam
Privacy
Damages
PAIN
fears
frustrations
obstacles
No trust
Addiction to more buying
GAIN
"wants" / needs
measures of success
obstacles
Resale Option
Flexibility
Money Saving

3.2 Ideation & Brainstorming:

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

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.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



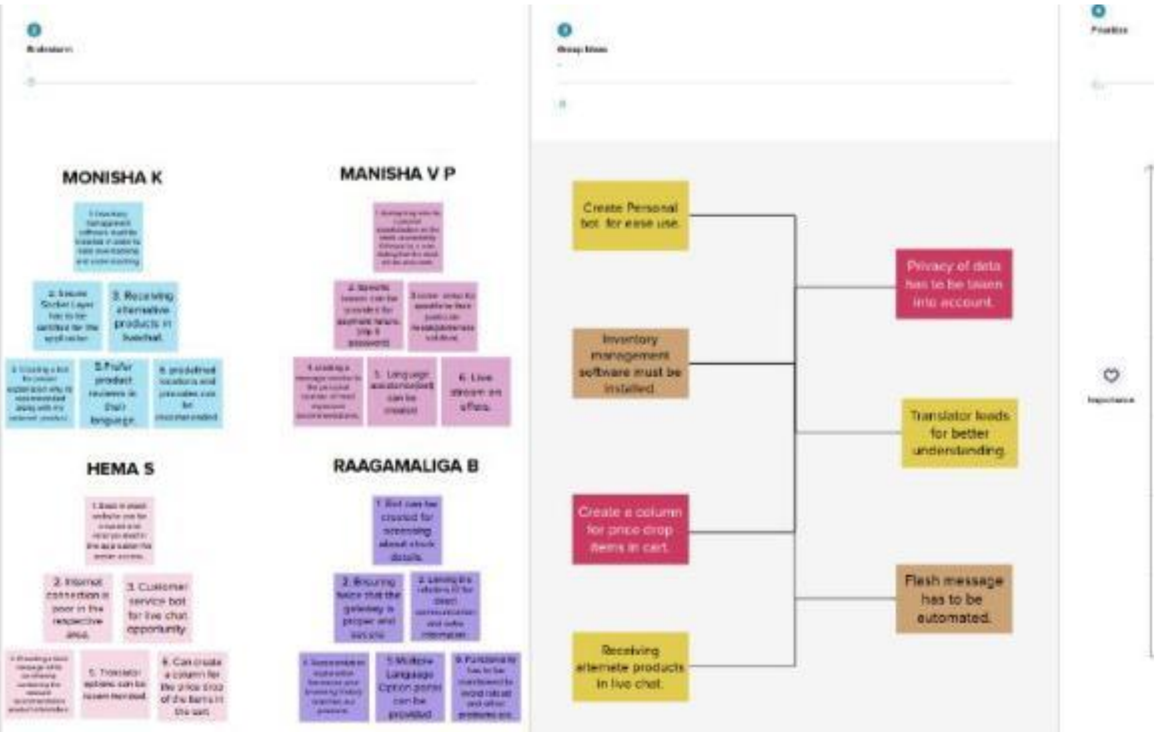
PROBLEM

In shopping apps navigating to the desired product is very difficult and no proper assistance is provided. People do find it difficult and face problems like

1. Out of stock
2. Payment failure
3. No live chat opportunity
4. No proper explanation for recommended products
5. Language
6.To save time

Here is our proposal solutions for the problems

Step-2: Brainstorm, Idea Listing and Grouping



MONISHA K

1. [illegible]

2. Secure Socket Layer has to be certified for the application.

3. Receiving alternative products in livechat.

4. [illegible]

5.Prefer product reviews in their language.

6. predefined locations and pincodes can be recommended

MANISHA V P

1. [illegible]

2. Specific tokens can be provided for payment failure (otp & password)

3. [illegible]

4. [illegible]

5. Language assistance/bot can be created

6. Live stream on offers.

HEMA S

1. [illegible]

2. Internet connection is poor in the respective area.

3. Customer service bot for live chat opportunity.

4. [illegible]

5. Translator options can be recommended.

6. Can create a column for the price drop of the items in the cart.

RAAGAMALIGA B

1. Bot can be created for screening about stock details.

2. Ensuring twice that the gateway is proper and secure.

3. [illegible]

4. [illegible]

5. Multiple Language Option panel can be provided

6. Functionality has to be maintained to [illegible]

Create Personal bot for ease use.

Inventory management software must be installed.

Create a column for price drop items in cart.

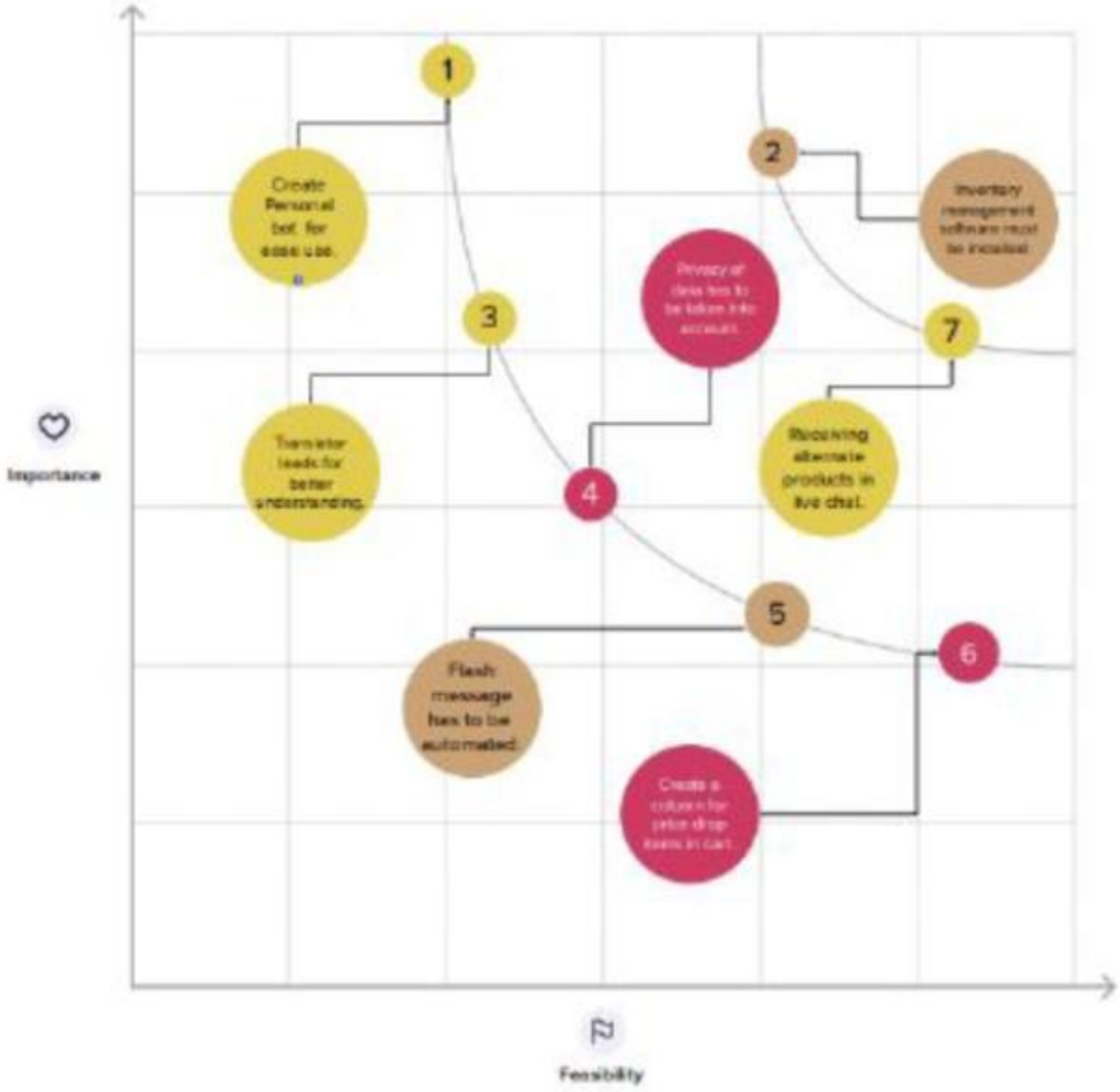
Receiving alternate products in live chat.

Privacy of data has to be taken into account.

Translator leads for better understanding.

Flash message has to be automated.

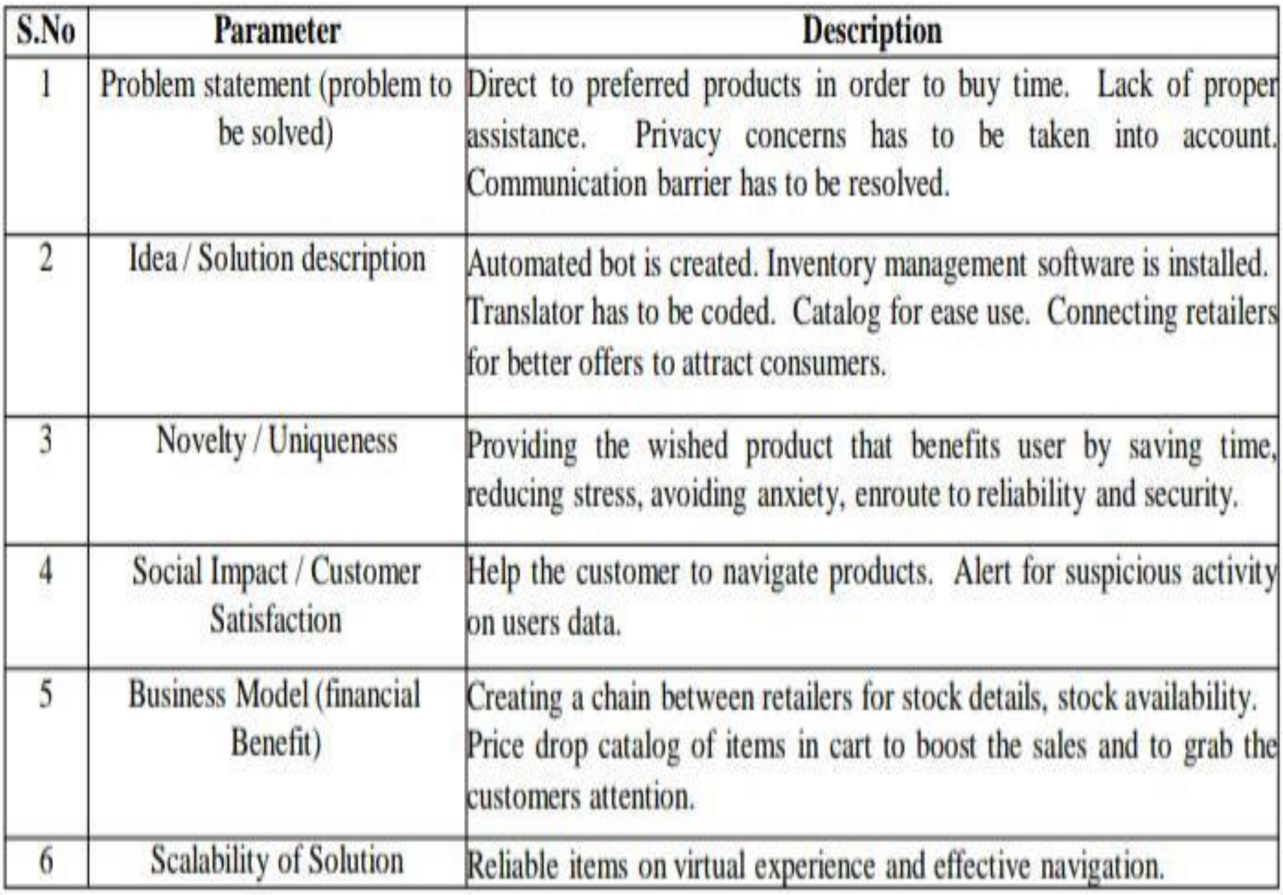
Step-3: Idea Prioritization


1
Create Personal bot for ease use.
2
Inventory management software must be installed
Privacy of data has to be taken into account.
3
Translator leads for better understanding.
7
Receiving alternate products in live chat.
4
5
6
Flash message has to be automated.
Create a column for price drop items in cart.
Importance
Feasibility

3.3 Proposed Solution:

Proposed Solution Template:

The project team shall fill in the following information in the proposed solution template.



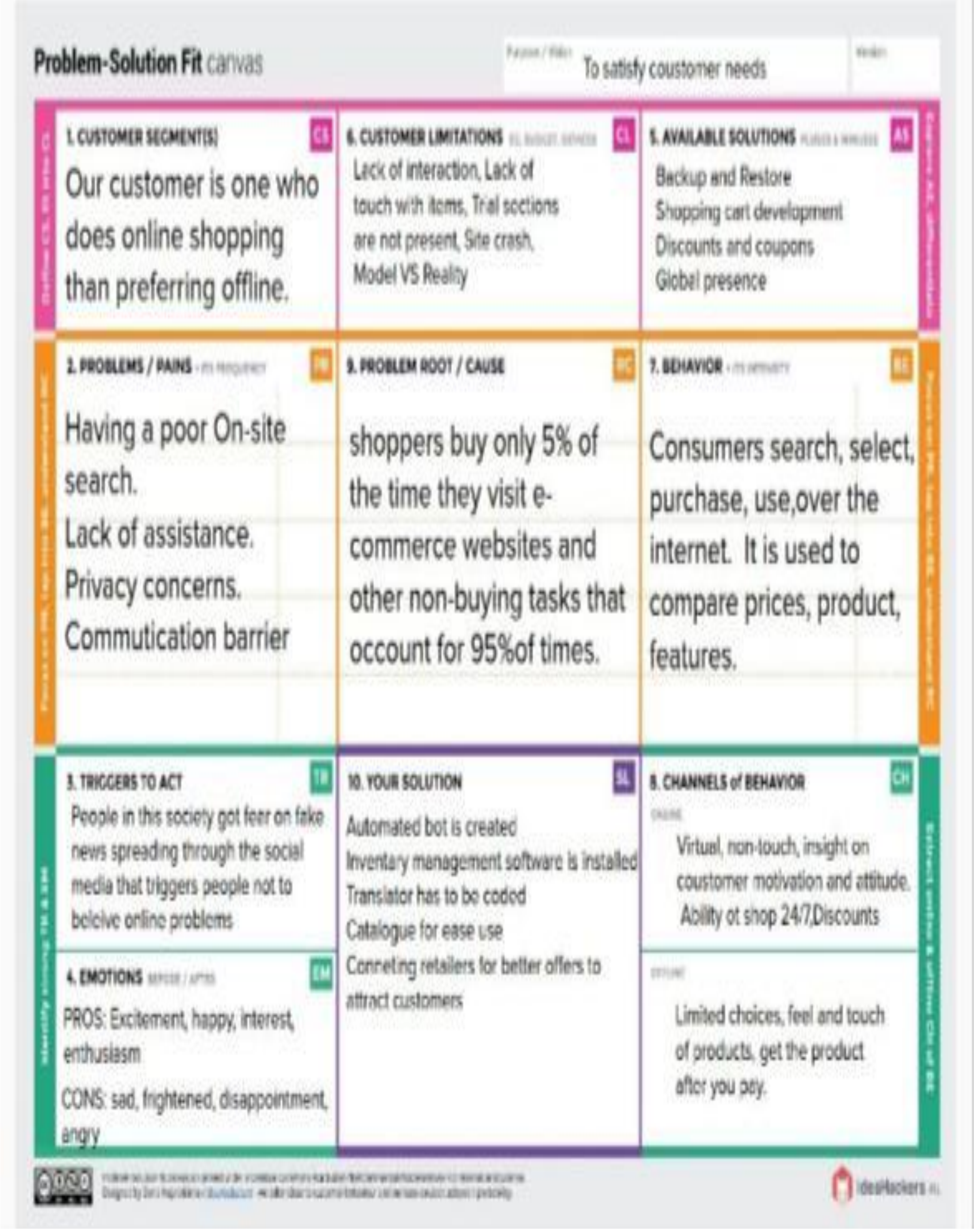
S.No	Parameter	Description
1	Problem statement (problem to be solved)	Direct to preferred products in order to buy time. Lack of proper assistance. Privacy concerns has to be taken into account. Communication barrier has to be resolved.
2	Idea / Solution description	Automated bot is created. Inventory management software is installed. Translator has to be coded. Catalog for ease use. Connecting retailers for better offers to attract consumers.
3	Novelty / Uniqueness	Providing the wished product that benefits user by saving time, reducing stress, avoiding anxiety, enroute to reliability and security.
4	Social Impact / Customer Satisfaction	Help the customer to navigate products. Alert for suspicious activity on users data.
5	Business Model (financial Benefit)	Creating a chain between retailers for stock details, stock availability. Price drop catalog of items in cart to boost the sales and to grab the customers attention.
6	Scalability of Solution	Reliable items on virtual experience and effective navigation.

3.4 Problem Solution fit:

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

Purpose:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem behaviour fit and building trust by solving frequent annoyances, or urgent or costly problems.
- Understand the existing situation in order to improve it for your target group.



Problem-Solution Fit canvas

To satisfy coustomer needs

1. CUSTOMER SEGMENT(S) — CS

Our customer is one who does online shopping than preferring offline.

6. CUSTOMER LIMITATIONS — CL

Lack of interaction, Lack of touch with items, Trial sections are not present, Site crash, Model VS Reality

5. AVAILABLE SOLUTIONS — AS

Backup and Restore
Shopping cart development
Discounts and coupons
Global presence

2. PROBLEMS / PAINS — PR

Having a poor On-site search.
Lack of assistance.
Privacy concerns.
Commutication barrier

9. PROBLEM ROOT / CAUSE — RC

shoppers buy only 5% of the time they visit e-commerce websites and other non-buying tasks that occount for 95%of times.

7. BEHAVIOR — BE

Consumers search, select, purchase, use,over the internet. It is used to compare prices, product, features.

3. TRIGGERS TO ACT — TR

People in this society got fear on fake news spreading through the social media that triggers people not to beleive online problems

4. EMOTIONS — EM

PROS: Excitement, happy, interest, enthusiasm

CONS: sad, frightened, disappointment, angry

10. YOUR SOLUTION — SL

Automated bot is created
Inventary management software is installed
Translator has to be coded
Catalogue for ease use
Conneting retailers for better offers to attract customers

8. CHANNELS of BEHAVIOR — CH

ONLINE

Virtual, non-touch, insight on coustomer motivation and attitude. Ability ot shop 24/7,Discounts

OFFLINE

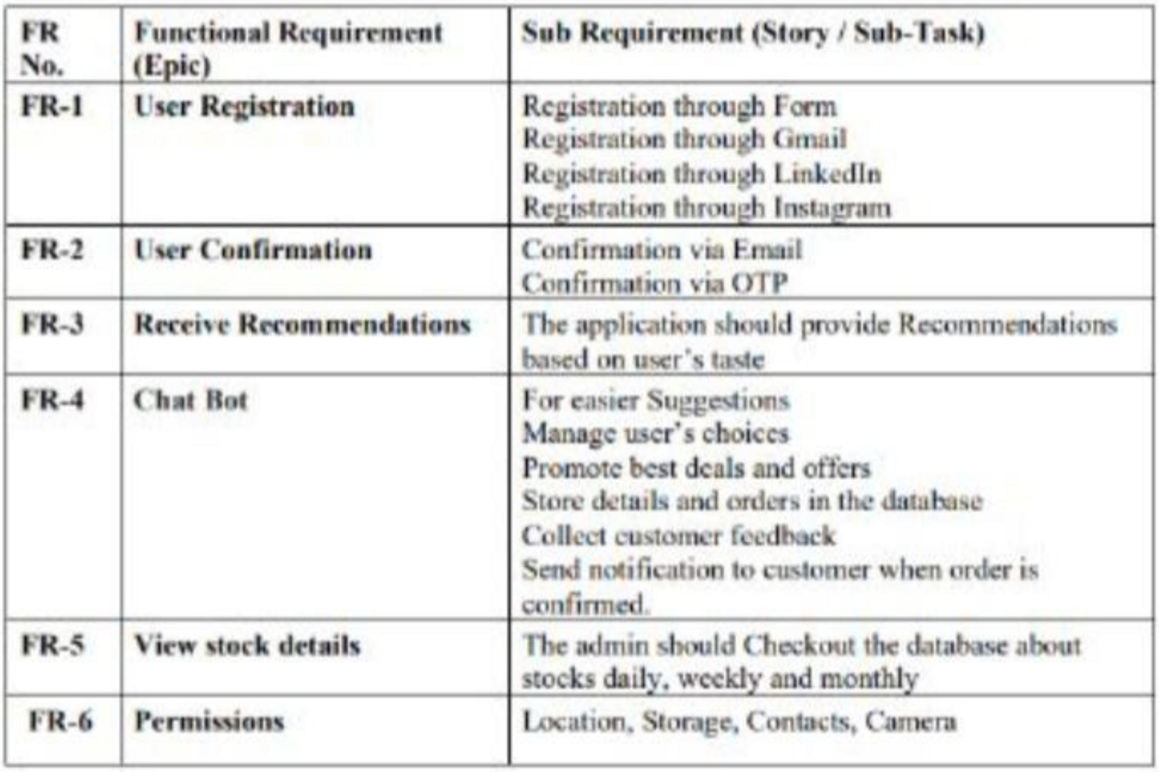
Limited choices, feel and touch of products, get the product after you pay.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional requirement:

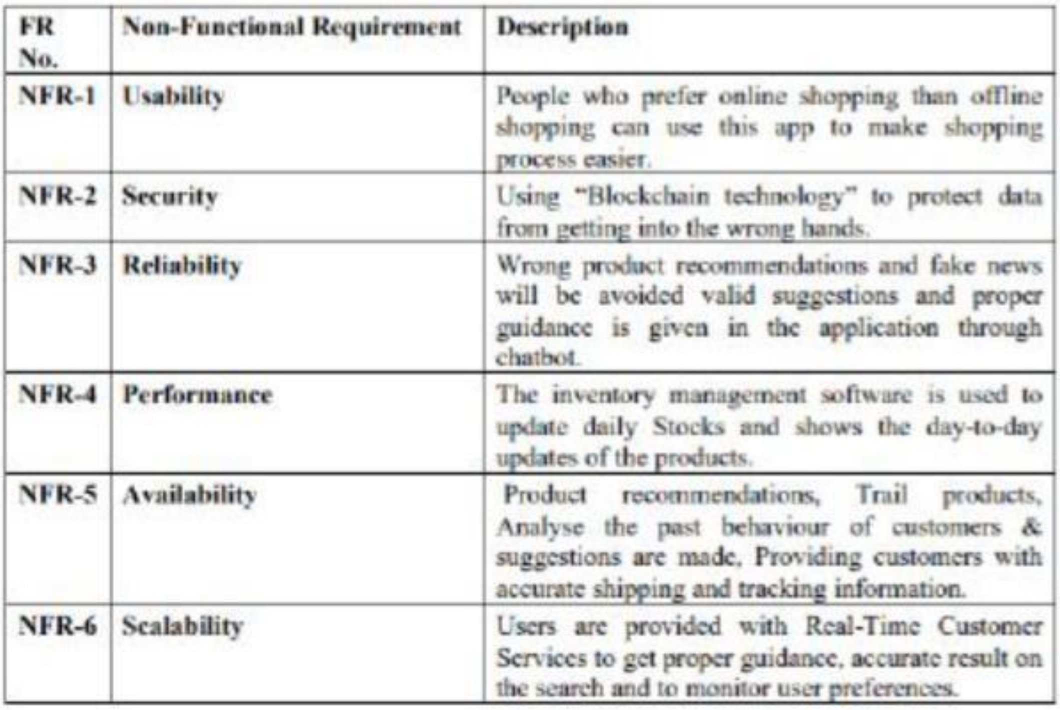
Following are the functional requirements of the proposed solution.



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn Registration through Instagram
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Receive Recommendations	The application should provide Recommendations based on user's taste
FR-4	Chat Bot	For easier Suggestions Manage user's choices Promote best deals and offers Store details and orders in the database Collect customer feedback Send notification to customer when order is confirmed.
FR-5	View stock details	The admin should Checkout the database about stocks daily, weekly and monthly
FR-6	Permissions	Location, Storage, Contacts, Camera

4.2 Non-functional Requirements:

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



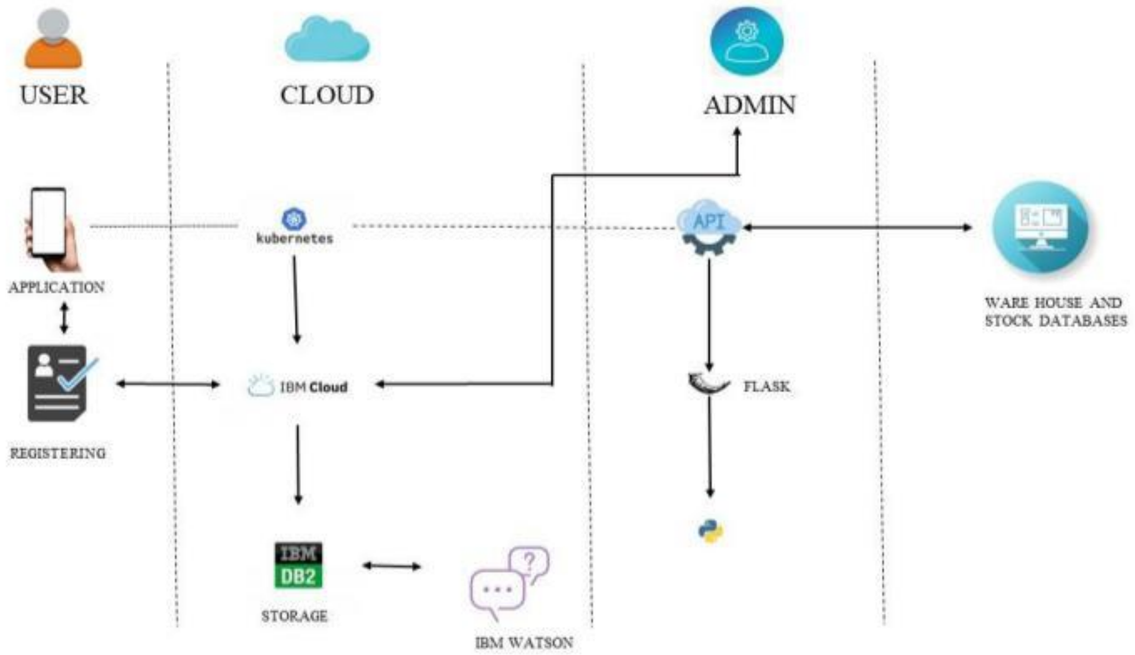
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	People who prefer online shopping than offline shopping can use this app to make shopping process easier.
NFR-2	Security	Using "Blockchain technology" to protect data from getting into the wrong hands.
NFR-3	Reliability	Wrong product recommendations and fake news will be avoided valid suggestions and proper guidance is given in the application through chatbot.
NFR-4	Performance	The inventory management software is used to update daily Stocks and shows the day-to-day updates of the products.
NFR-5	Availability	Product recommendations, Trail products, Analyse the past behaviour of customers & suggestions are made, Providing customers with accurate shipping and tracking information.
NFR-6	Scalability	Users are provided with Real-Time Customer Services to get proper guidance, accurate result on the search and to monitor user preferences.

CHAPTER 5

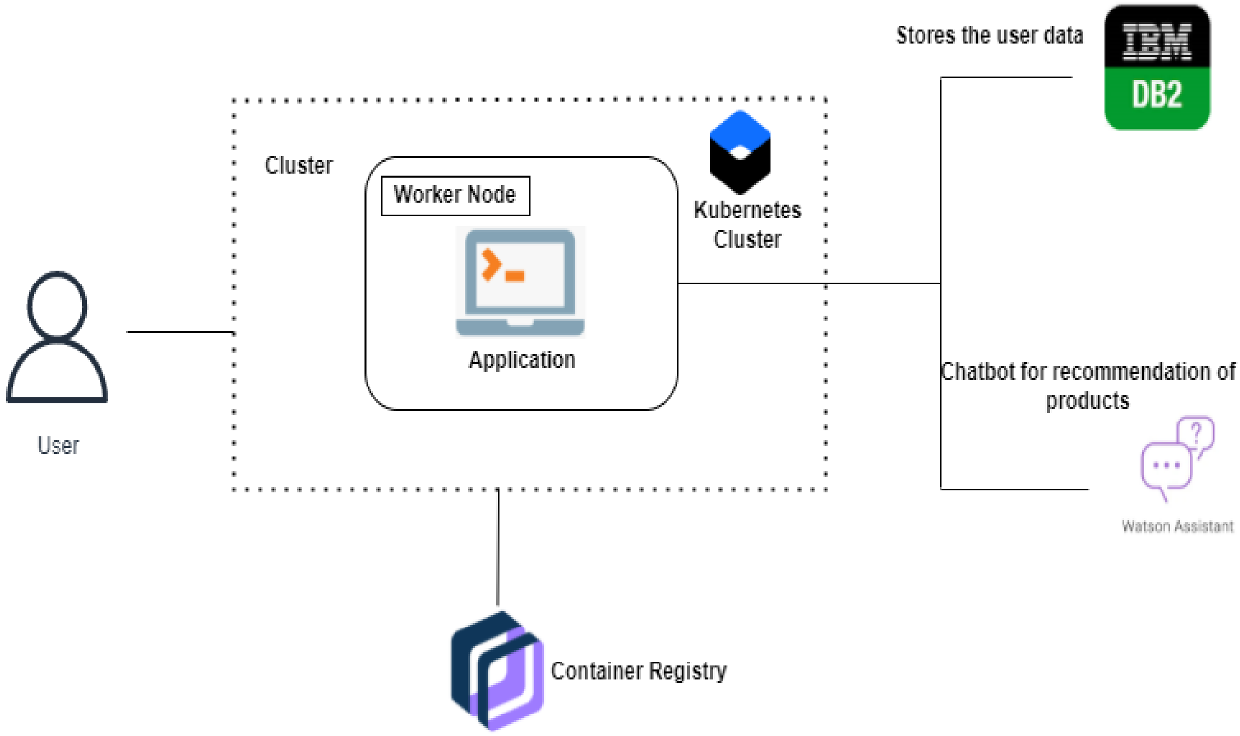
PROJECT DESIGN

5.1 Data Flow Diagrams:

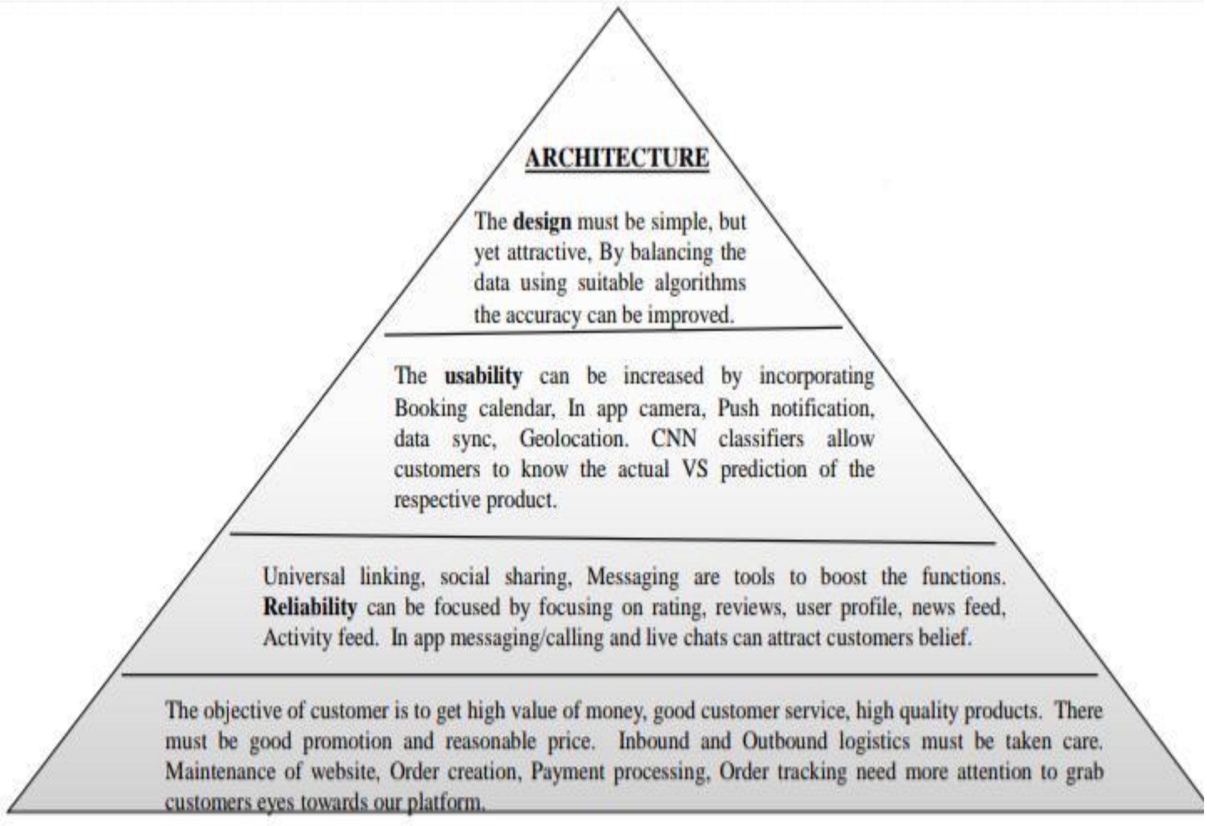
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can graphically depict the right amount of the system requirement. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Technical Architecture:



5.3 Solution Architecture:



ARCHITECTURE

The **design** must be simple, but yet attractive, By balancing the data using suitable algorithms the accuracy can be improved.

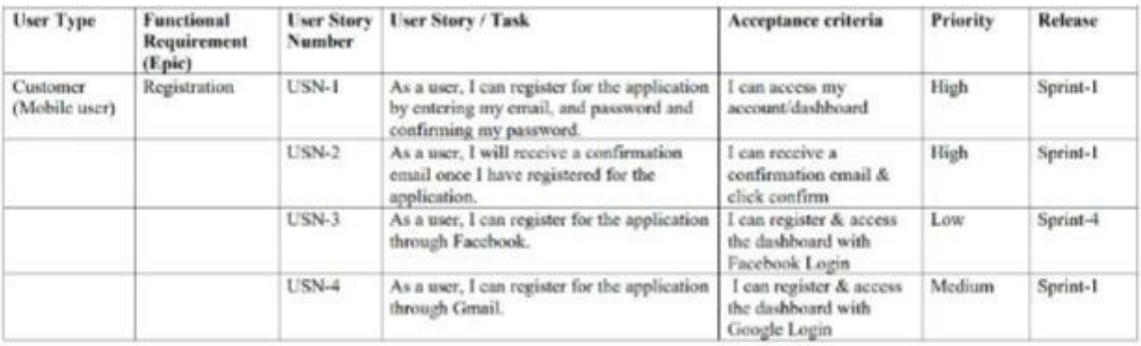
The **usability** can be increased by incorporating Booking calendar, In app camera, Push notification, data sync, Geolocation. CNN classifiers allow customers to know the actual VS prediction of the respective product.

Universal linking, social sharing, Messaging are tools to boost the functions. **Reliability** can be focused by focusing on rating, reviews, user profile, news feed, Activity feed. In app messaging/calling and live chats can attract customers belief.

The objective of customer is to get high value of money, good customer service, high quality products. There must be good promotion and reasonable price. Inbound and Outbound logistics must be taken care. Maintenance of website, Order creation, Payment processing, Order tracking need more attention to grab customers eyes towards our platform.

5.4 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, and password and confirming my password.	I can access my account/dashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application.	I can receive a 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-4
		USN-4	As a user, I can register for the application through Gmail.	I can register & access the dashboard with Google Login	Medium	Sprint-1

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

Use the below template to create a product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	UI Creation Creating Registration page, Login page	10	Medium	HEMA S MANISHA V P MONISHA K RAAGAMALIGA B
Sprint-1	Database Connectivity	USN-2	Viewing and applying jobs Connecting UI with Database	10	High	HEMA S MANISHA V P MONISHA K RAAGAMALIGA B
Sprint-2	SendGrid Integration	USN-3	SendGrid Integration with Python Code	10	Low	HEMA S MANISHA V P MONISHA K RAAGAMALIGA B
Sprint-2	Chatbot Development	USN-4	Building a chatbot	10	High	HEMA S MANISHA V P MONISHA K

						RAAGAMALIGA B
Sprint-3	Integration and Containerization	USN-5	Integrating the chatbot to the e-HTML page and containerizing the app	20	Medium	HEMA S MANISHA V P MONISHA K RAAGAMALIGA B
Sprint-4	Upload Image and deployment	USN-6	Upload the image to the IBM Registry and deploy it in the Kubernetes Cluster.	20	High	HEMA S MANISHA V P MONISHA K RAAGAMALIGA B

6.2 Sprint Delivery Schedule:

Project Tracker, Velocity:

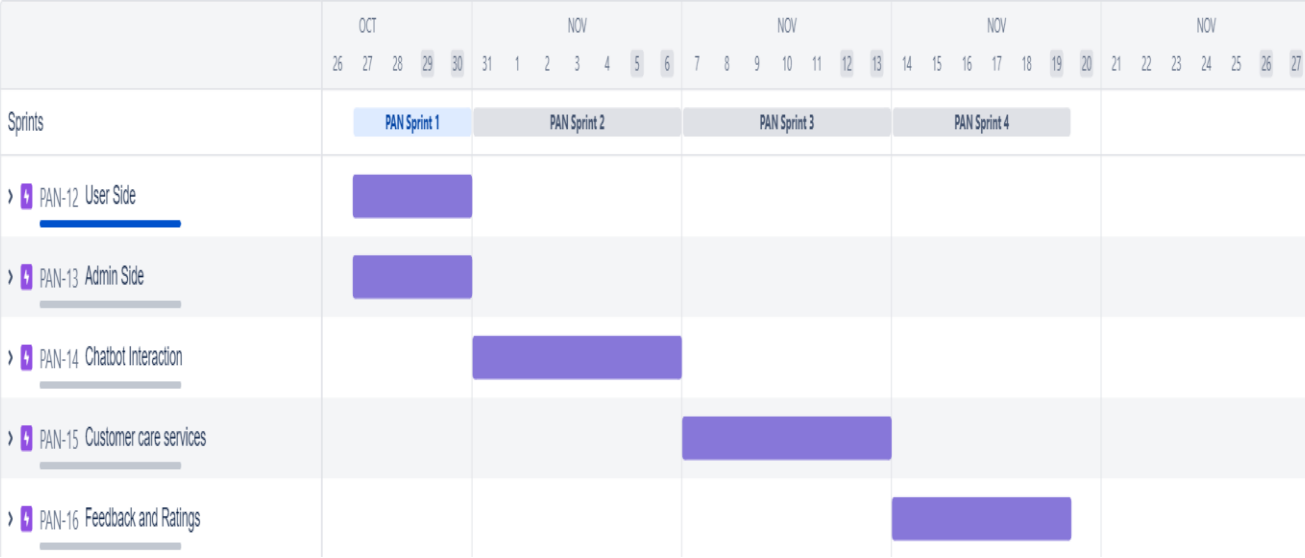
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	06 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

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

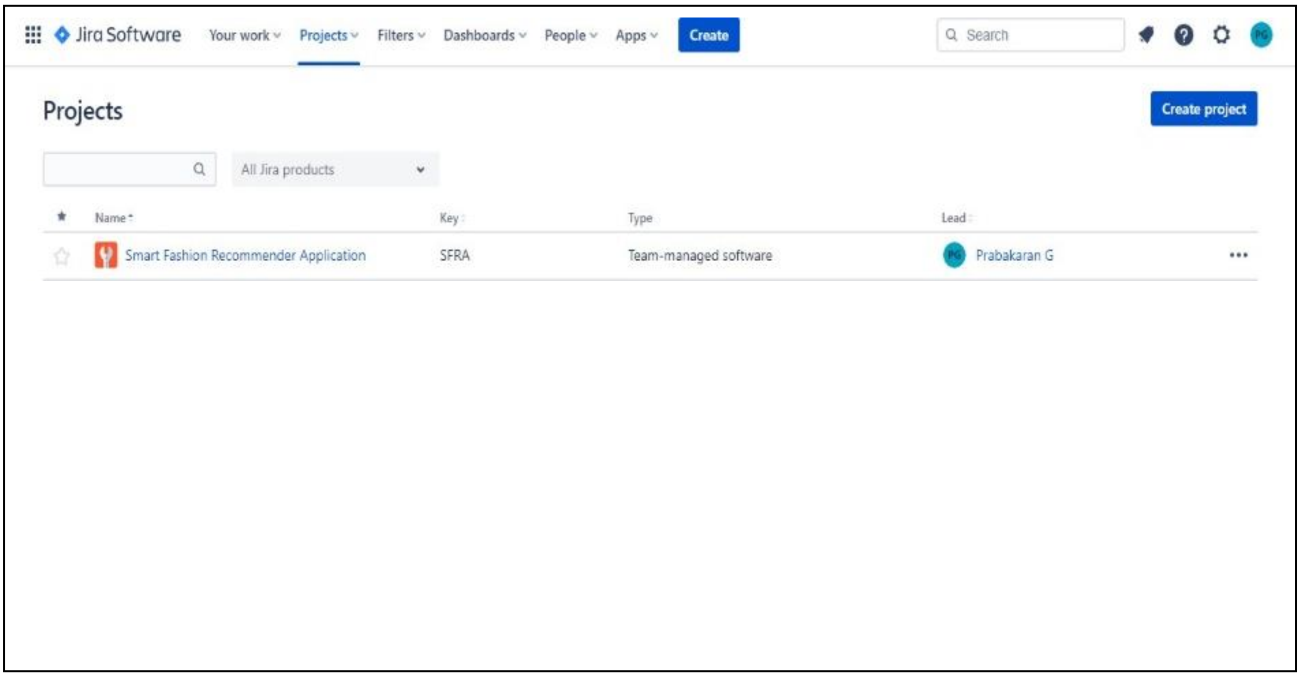
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

Burndown Chart :



6.3 Reports from JIRA

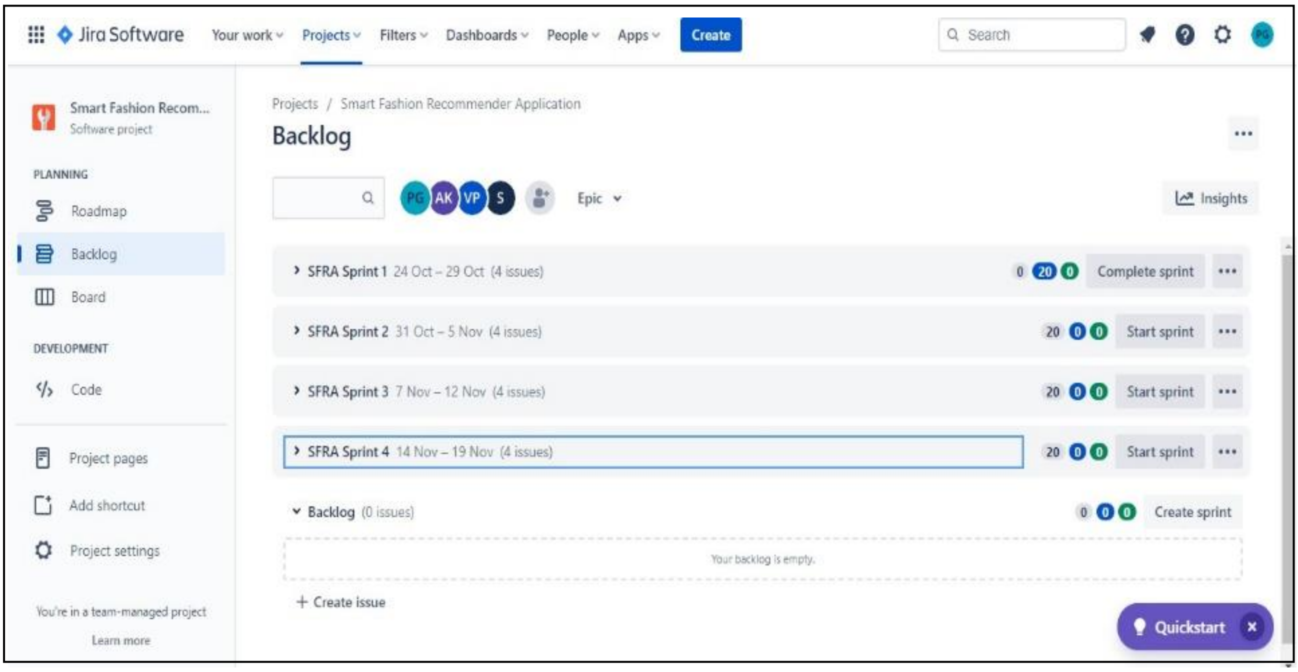
1.Image showing the project created in the JIRA software



Projects

★	Name	Key	Type	Lead
☆	Smart Fashion Recommender Application	SFRA	Team-managed software	Prabakaran G

2. Image showing the List of Sprints and their time duration in the project



SFRA Sprint 1 24 Oct – 29 Oct (4 issues) 0 20 0 Complete sprint

SFRA Sprint 2 31 Oct – 5 Nov (4 issues) 20 0 0 Start sprint

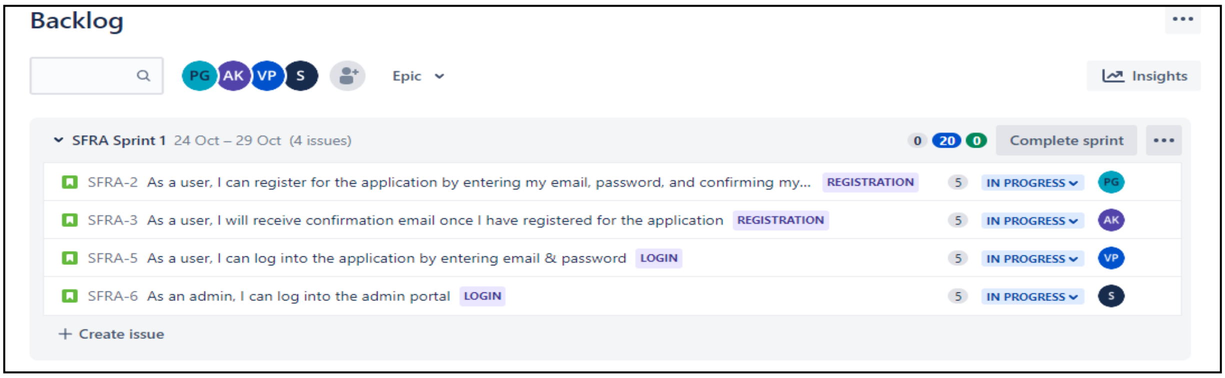
SFRA Sprint 3 7 Nov – 12 Nov (4 issues) 20 0 0 Start sprint

SFRA Sprint 4 14 Nov – 19 Nov (4 issues) 20 0 0 Start sprint

Backlog (0 issues) 0 0 0 Create sprint

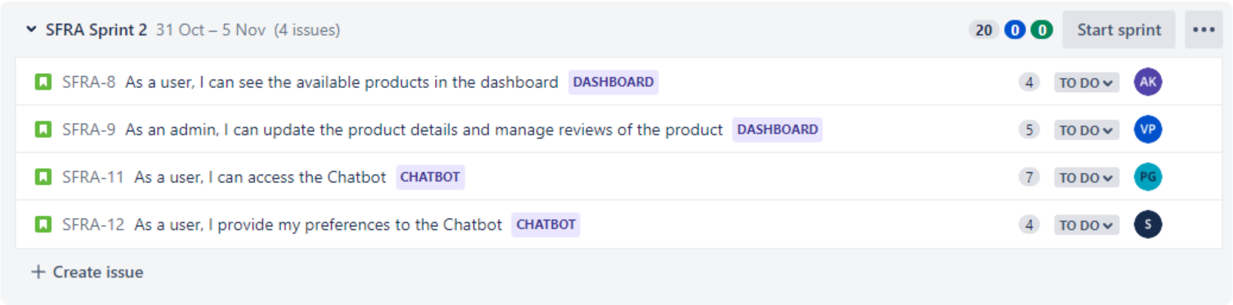
Your backlog is empty.

3. Images showing the User Stories involved in each sprint



SFRA Sprint 1 24 Oct – 29 Oct (4 issues)

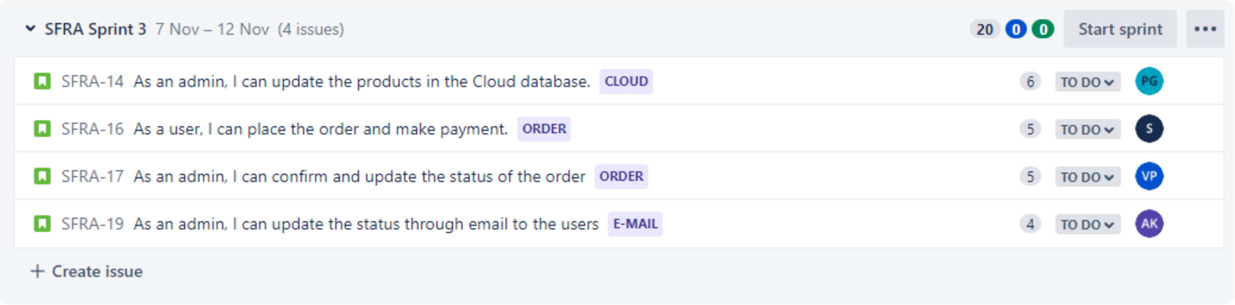
SFRA-2	As a user, I can register for the application by entering my email, password, and confirming my...	REGISTRATION	5	IN PROGRESS	PG
SFRA-3	As a user, I will receive confirmation email once I have registered for the application	REGISTRATION	5	IN PROGRESS	AK
SFRA-5	As a user, I can log into the application by entering email & password	LOGIN	5	IN PROGRESS	VP
SFRA-6	As an admin, I can log into the admin portal	LOGIN	5	IN PROGRESS	S



SFRA Sprint 2 31 Oct – 5 Nov (4 issues)

SFRA-8	As a user, I can see the available products in the dashboard	DASHBOARD	4	TO DO	AK
SFRA-9	As an admin, I can update the product details and manage reviews of the product	DASHBOARD	5	TO DO	VP
SFRA-11	As a user, I can access the Chatbot	CHATBOT	7	TO DO	PG
SFRA-12	As a user, I provide my preferences to the Chatbot	CHATBOT	4	TO DO	S

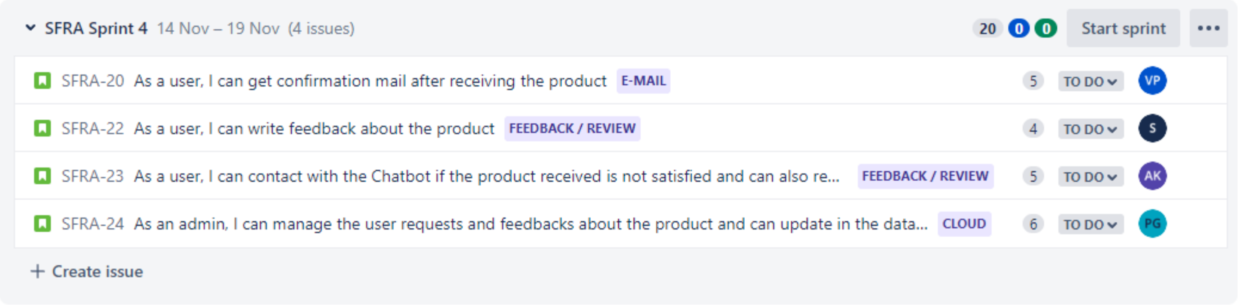
3. Images showing the User Stories involved in each sprint (cont.)



SFRA Sprint 3 7 Nov – 12 Nov (4 issues) 20 0 0 Start sprint •••

Key	Summary	Label	Points	Status	Assignee
SFRA-14	As an admin, I can update the products in the Cloud database.	CLOUD	6	TO DO	PG
SFRA-16	As a user, I can place the order and make payment.	ORDER	5	TO DO	S
SFRA-17	As an admin, I can confirm and update the status of the order	ORDER	5	TO DO	VP
SFRA-19	As an admin, I can update the status through email to the users	E-MAIL	4	TO DO	AK

+ Create issue

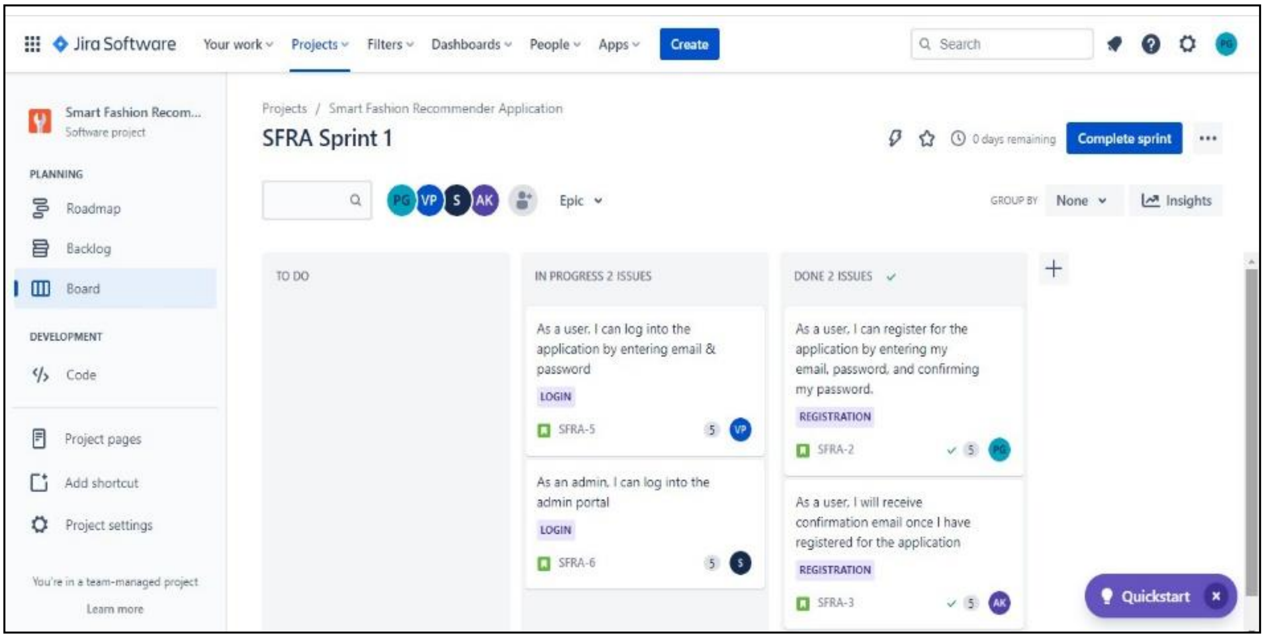


SFRA Sprint 4 14 Nov – 19 Nov (4 issues) 20 0 0 Start sprint •••

Key	Summary	Label	Points	Status	Assignee
SFRA-20	As a user, I can get confirmation mail after receiving the product	E-MAIL	5	TO DO	VP
SFRA-22	As a user, I can write feedback about the product	FEEDBACK / REVIEW	4	TO DO	S
SFRA-23	As a user, I can contact with the Chatbot if the product received is not satisfied and can also re...	FEEDBACK / REVIEW	5	TO DO	AK
SFRA-24	As an admin, I can manage the user requests and feedbacks about the product and can update in the data...	CLOUD	6	TO DO	PG

+ Create issue

4. Image showing the working progress of the tasks in the selected Spring



TO DO

IN PROGRESS 2 ISSUES

As a user, I can log into the application by entering email & password

LOGIN

SFRA-5 5 VP

As an admin, I can log into the admin portal

LOGIN

SFRA-6 5 S

DONE 2 ISSUES ✓

As a user, I can register for the application by entering my email, password, and confirming my password.

REGISTRATION

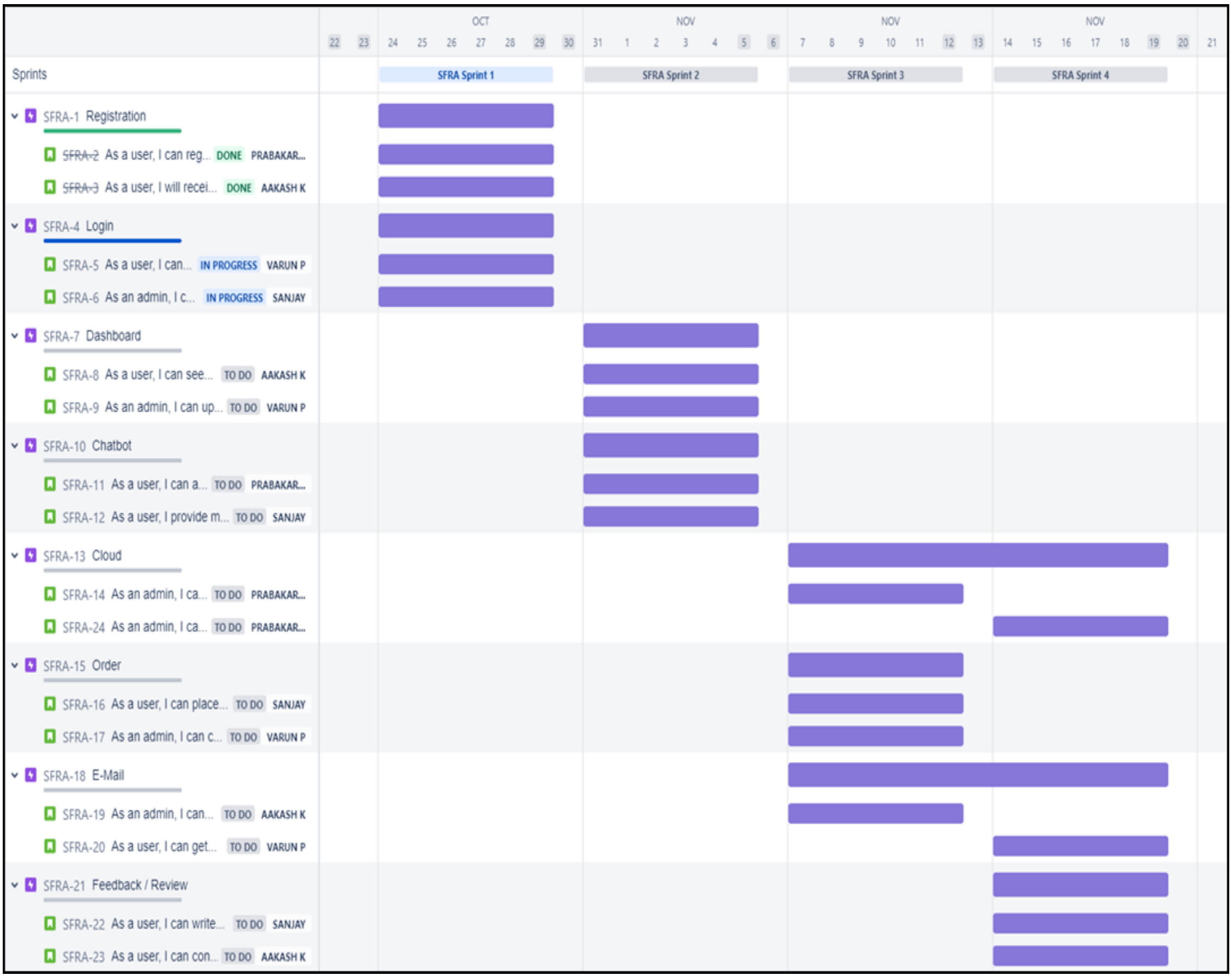
SFRA-2 ✓ 5 PG

As a user, I will receive confirmation email once I have registered for the application

REGISTRATION

SFRA-3 ✓ 5 AK

5. Image showing the Roadmap of the Project



SFRA-1	Registration		
~~SFRA-2~~	As a user, I can reg...	DONE	PRABAKAR...
~~SFRA-3~~	As a user, I will recei...	DONE	AAKASH K
SFRA-4	Login		
SFRA-5	As a user, I can...	IN PROGRESS	VARUN P
SFRA-6	As an admin, I c...	IN PROGRESS	SANJAY
SFRA-7	Dashboard		
SFRA-8	As a user, I can see...	TO DO	AAKASH K
SFRA-9	As an admin, I can up...	TO DO	VARUN P
SFRA-10	Chatbot		
SFRA-11	As a user, I can a...	TO DO	PRABAKAR...
SFRA-12	As a user, I provide m...	TO DO	SANJAY
SFRA-13	Cloud		
SFRA-14	As an admin, I ca...	TO DO	PRABAKAR...
SFRA-24	As an admin, I ca...	TO DO	PRABAKAR...
SFRA-15	Order		
SFRA-16	As a user, I can place...	TO DO	SANJAY
SFRA-17	As an admin, I can c...	TO DO	VARUN P
SFRA-18	E-Mail		
SFRA-19	As an admin, I can...	TO DO	AAKASH K
SFRA-20	As a user, I can get...	TO DO	VARUN P
SFRA-21	Feedback / Review		
SFRA-22	As a user, I can write...	TO DO	SANJAY
SFRA-23	As a user, I can con...	TO DO	AAKASH K

CHAPTER 7

CODING & SOLUTIONING:

index.html

```
<DOCTYPE html>
    <html lang=”en”>
        <head>
            <meta charset=”UTF-8”>
            <meta http-equi=”X-UA-Compatible” content=”IE-edge”>
            <meta name=”viewport” content=”width=device-width, initial-scale=1.0”>
            <title>
                Clothing – Best Apparels Online Store
            </title>

            <link rel=”stylesheet” href=”CSS/home.css”>

        </head>
        <body>
            <nav class=”navbar”>
                <div class=”nav”>
                    <img src=”img/dark-logo.png” class=”brand-logo” alt=””>
                    <div class=”nav-items”>
                        <div class=”search”>
                            <input type=”text” class=”search-box” placeholder=”searchbrand, product”>
                            <button class=”search-btn”>search
                            </button>

                        </div>
                        <ul class=”links=container”>
                            <li class=”link-item”><a href=”#”
class=”link”>home</a></li>
                            <li class=”link-item”><a href=”#”
class=”link”>women</a></li>
                            <li class=”link-item”><a href=”#”
class=”link”>men</a></li>
                            <li class=”link-item”><a href=”#”
class=”link”>kids</a></li>
                            <li class=”link-item”><a href=”#”
class=”link”>accessories</a></li>
                        </ul>
                        <a href=”#”>
                            <img src=”img/user.png” alt=””>
                        </a>
                        <a href=”#”>
                            <img src=”img/cart.png” alt=””>
                        </a>

                    </div>
                </div>

            </nav>

            <!—hero section →
            <header class=”hero-section”>
                <div class=”content”>
                    <img src=”img/light-logo.png” class=”logo”
alt=””>
                    <p class=”sub=heading”>
                        Best Fashion Collection Of All Time
```

```
          </p>
        </div>
      </header>
      <section class=”product”>
        <h2 class=”product-category”>best selling</h2>
        <button class=”pre-btn”><img
src=”img/arrow6.png” alt=””></button>
        <button class=”nxt-btn”><img
src=”img/arrow6.png” alt=””></button>

        <div class=”product-container”>
          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>50% off</span>
              <img src=”img/card1.png” class=”product-
thumb” alt=””>
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>20% off</span>
              <img src=”img/card2.png” class=”product-
thumb” alt=””>
```

```
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>30% off</span>
              <img src=”img/card3.png” class=”product-
thumb” alt=””>
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>40% off</span>
              <img src=”img/card4.png” class=”product-
thumb” alt=””>
```

```
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>50% off</span>
              <img src=”img/product image 1.png”
class=”product-thumb” alt=””>
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>50% off</span>
              <img src=”img/product image 2.png”
class=”product-thumb” alt=””>
```

```
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>

          <div class=”product-card”>
            <div class=”product-image”>
              <span class=”discount-tag”>50% off</span>
              <img src=”img/product image 3.png”
class=”product-thumb” alt=””>
              <button class=”card-btn”>add to
whislist</button>
            </div>
            <div class=”product-info”>
              <h2 class=”product-brand”>BRAND</h2>
              <p class=”product-short-des”>a short line
about the cloth..</p>
              <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
          </div>


        </div>
      </section>
<!—collections →
<section class=”collection-container”>
  <a href=”#” class=”collection”>
```

```
    <img src=”img/women collection.png” alt=””>
    <p class=”collection-title”>Women Apparels</p>
  </a>
  <a href=”#” class=”collection”>
    <img src=”img/men.png” alt=””>
    <p class=”collection-title”>Men Apparels</p>
  </a>
  <a href=”#” class=”collection”>
    <img src=”img/pngegg.png” alt=””>
    <p class=”collection-title”>Accessories</p>
  </a>
</section>
<section class=”product”>
  <h2 class=”product-category”>shoes</h2>
  <button class=”pre-btn”><img src=”img/arrow6.png”
alt=””></button>
  <button class=”nxt-btn”><img src=”img/arrow6.png”
alt=””></button>

  <div class=”product-container”>
    <div class=”product-card”>
      <div class=”product-image”>
        <span class=”discount-tag”>50% off</span>
        <img src=”img/pic1.jpg” class=”product-thumb”
alt=””>
        <button class=”card-btn”>add to whislist</button>
      </div>
      <div class=”product-info”>
        <h2 class=”product-brand”>BRAND</h2>
        <p class=”product-short-des”>a short line about the
cloth..</p>
        <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
      </div>
```

```
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>20% off</span>
                <img src=”img/pic2.webp” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>30% off</span>
                <img src=”img/pic3.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>
        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>40% off</span>
                <img src=”img/pic4.webp” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>50% off</span>
                <img src=”img/pic5.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
        <span class=”discount-tag”>50% off</span>
        <img src=”img/pic6.webp” class=”product-thumb”
alt=”””>
        <button class=”card-btn”>add to whislist</button>
      </div>
      <div class=”product-info”>
        <h2 class=”product-brand”>BRAND</h2>
        <p class=”product-short-des”>a short line about the
cloth..</p>
        <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
      </div>
    </div>

    <div class=”product-card”>
      <div class=”product-image”>
        <span class=”discount-tag”>50% off</span>
        <img src=”img/pic7.jpeg.jpg” class=”product-
thumb” alt=”””>
        <button class=”card-btn”>add to whislist</button>
      </div>
      <div class=”product-info”>
        <h2 class=”product-brand”>BRAND</h2>
        <p class=”product-short-des”>a short line about the
cloth..</p>
        <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
      </div>
    </div>

  </div>
</section>
<footer>
  <div class=”footer-content”>
```

```
        <img src=”img/light-logo.png” class=”logo” alt=””>
        <div class=”footer-ul-container”>
            <ul class=”category”>
                <li class=”category-title”>men</li>
                <li><a href=”#” class=”footer-link”>t-
shirts</a></li>
                <li><a href=”#” class=”footer-
link”>sweatshirts</a></li>
                <li><a href=”#” class=”footer-link”>shirts</a></li>
                <li><a href=”#” class=”footer-link”>jeans</a></li>
                <li><a href=”#” class=”footer-
link”>trousers</a></li>
                <li><a href=”#” class=”footer-link”>shoes</a></li>
                <li><a href=”#” class=”footer-
link”>casuals</a></li>
                <li><a href=”#” class=”footer-
link”>formals</a></li>
                <li><a href=”#” class=”footer-
link”>sports</a></li>
                <li><a href=”#” class=”footer-
link”>watch</a></li>
            </ul>
            <ul class=”category”>
                <li class=”category-title”>women</li>
                <li><a href=”#” class=”footer-link”>t-
shirts</a></li>
                <li><a href=”#” class=”footer-
link”>sweatshirts</a></li>
                <li><a href=”#” class=”footer-link”>shirts</a></li>
                <li><a href=”#” class=”footer-link”>jeans</a></li>
                <li><a href=”#” class=”footer-
link”>trousers</a></li>
                <li><a href=”#” class=”footer-link”>shoes</a></li>
                <li><a href=”#” class=”footer-
link”>casuals</a></li>
                <li><a href=”#” class=”footer-
link”>formals</a></li>
                <li><a href=”#” class=”footer-
link”>sports</a></li>
                <li><a href=”#” class=”footer-
link”>watch</a></li>
            </ul>
        </div>
    </div>
    <p class=”footer-title”>about company</p>
    <p class=”info”>Lorem ipsum, dolor sit amet consectetur
adipisicing elit.
    Repellat tempore ad suscipit, eos eius quisquam sed optio
nisi quaerat
    Fugiat ratione et vero maxime praesentium, architecto
minima reiciendis
    Iste quo deserunt assumenda alias ducimus. Ullam odit
maxime id voluptates
    Rerum tenetur corporis laboriosam! Cum error ipsum
laborum tempore in rerum
    Necessitatibus nostrum nobis modi! Debitis adipisci illum
nemo aperiam sed,
    Et accusamus ut officiis. Laborum illo exercitationem quo
culpa
    Reprehenderit excepturi distinctio tempore cupiditate
praesentium nisi ut
    Iusto, assumenda perferendis facilis voluptas autem fuga
sunt ab debitis
    Voluptatum harum eum. Asperiores, natus! Est deserunt
incidunt quasi
    Placeat omnis, itaque harum?</p>
    <p class=”info”>support emails – help@clothing.com,
    customersupport@clothing.com</p>
    <p class=”info”>telephone – 180 00 00 001, 180 00 00
002</p>
    <div class=”footer-social-container”>
        <div>
            <a href=”#” class=”social-link”>terms & services</a>
            <a href=”#” class=”social-link”>privacy page</a>
        </div>
        <div>
            <a href=”#” class=”social-link”>instagram</a>
            <a href=”#” class=”social-link”>facebook</a>
            <a href=”#” class=”social-link”>twitter</a>
        </div>
    </div>
    <p class=”footer-credit”>Clothing, Best apparels online
store</p>
        </footer>
            <script src=”js/nav.js”></script>
            <script src=”js/footer.js”></script>
            <script src=”js/home.js”></script>
        </body>
    </html>
</DOCTYPE>
```

Home.css

```
@import ‘nav.css’;
@import ‘footer.css’;
```

```
.hero-section{
   Width: 100%;
   Height: calc(100vh – 120px);
   Background-image: url(‘../img/accessories-collection.png’);
   Background-size: cover;
   Display: flex;
   Justify-content: center;
   Align-items: center;
}

.hero-section .logo{
   Height: 150px;
   Display: block;
   Margin: auto;
}

.hero-section .sub-heading{
   Margin-top: 10px;
   Text-align: center;
   Color:rgb(237, 230, 230);
   Text-transform: capitalize;
   Font-size: 35px;
   Font-weight: 300;
}

/* product */

.product{
   Position: relative;
   Overflow: hidden;
   Padding: 20px 0;
}
```

```
.product-category{
   Padding: 0 10vw;
   Font-size: 30px;
   Font-weight: 500;
   Margin-bottom: 40px;
   Text-transform: capitalize;
}

.product-container{
   Padding: 0 10vw;
   Display: flex;
   Overflow-x: auto;
   Scroll-behavior: smooth;
}

.product-container::-webkit-scrollbar{
   Display: none;
}

.product-card{
   Flex: 0 0 auto;
   Width: 250px;
   Height: 450px;
   Margin-right: 40px;
}

.product-image{
   Position: relative;
   Width: 100%;
   Height: 350px;
   Overflow: hidden;
}

.product-thumb{
    Width: 100%;
    Height: 350px;
    Object-fit: cover;
}
.discount-tag{
    Position: absolute;
    Background: rgb(226, 230, 231);
    Padding: 5px;
    Border-radius: 5px;
    Color: #c3bdbd;
    Right: 10px;
    Top: 20px;
    Text-transform: capitalize;
}

.card-btn{
    Position: absolute;
    Bottom: 10px;
    Left: 50%;
    Transform: translateX(-50%);
    Padding: 10px;
    Width: 90%;
    Text-transform: capitalize;
    Border: none;
    Outline: none;
    Background: rgb(174, 189, 165);
    Border-radius: 5px;
    Transition: 0.5s;
    Cursor: pointer;
    Opacity: 0;
}

.product-card:hover .card-btn{
    Opacity: 1;
```

```
}

.card-btn:hover{
   Background: #adb5b9;
}

.product-info{
   Width: 100%;
   Height: 100px;
   Padding-top: 10px;
}
.product-brand{
   Text-transform: uppercase;
}

.product-short-des{
   Width: 100%;
   Height: 20px;
   Line-height: 20px;
   Overflow: hidden;
   Opacity: 0.5;
   Text-transform: capitalize;
   Margin: 5px 0;
}
.price{
   Font-weight: 900;
   Font-size: 20px;
}

.actual-price{
   Margin-left: 20px;
   Opacity: 0.5;
   Text-decoration: line-through;
}
```

```
.pre-btn, .nxt-btn{
   Border: none;
   Width: 10vw;
   Height: 100%;
   Position: absolute;
   Top: 0;
   Display: flex;
   Justify-content: center;
   Align-items: center;
   Background: linear-gradient(90deg, rgba(255, 255, 255, 0) 0%, rgb(250, 247, 247) 100%);
   Cursor: pointer;
   z-index: 8;
}

.pre-btn{
   Left: 0;
   Transform: rotate(180deg);
}

.nxt-btn{
   Right: 0;
}

.pre-btn img, .nxt-btn img{
   Opacity: 0.2;
}

.pre-btn:hover img, .nxt-btn:hover img{
   Opacity: 1;
}

.collection-container{
   Width: 100%;
   Display: grid;
   Grid-template-columns: repeat(2, 1fr);
   Grid-gap: 10px;
}

.collection{
   Position: relative;
}

/* collections */


.collection img{
   Width: 100%;
   Height: 100%;
   Object-fit: cover;
}

.collection p{
   Position: absolute;
   Top: 50%;
   Left: 50%;
   Transform: translate(-50%, -50%);
   Text-align: center;
   Color: #fff;
   Font-size: 50px;
   Text-transform: capitalize;
}

.collection:nth-child(3){
   Grid-column: span 2;
   Margin-bottom: 10px;
}
```

Nav.css

```
*{
   Margin: 0;
   Padding: 0;
   Box-sizing: border-box;
   Font-family: ‘roboto’, sans-serif;
}

.navbar{
   Position: sticky;
   Top: 0;
   Left: 0;
   Width: 100%;
   Background: #f5f5f5;
   z-index: 9;
}

.nav{
   Padding: 10px 10vw;
   Display: flex;
   Justify-content: space-between;
}

.brand-logo{
   Height: 60px;
```

```
}

.nav-items{
   Display: flex;
   Align-items: center;
}

.search{
   Width: 500px;
   Display: flex;
}

.search-box{
   Width: 80%;
   Height: 40px;
   Padding: 10px;
   Border-top-left-radius: 10px;
   Border-bottom-left-radius: 10px;
   Border: 1px solid #d1d1d1;
   Text-transform: capitalize;
   Background: none;
   Color: #a9a9a9;
   Outline: none;
}

.search-btn{
   Width: 20%;
   Height: 40px;
   Padding: 10px 20px;
   Border: none;
   Outline: none;
   Cursor: pointer;
   Background: #383838;
   Color: #fff;
```

```
    Text-transform: capitalize;
    Font-size: 15px;
    Border-top-right-radius: 10px;
    Border-bottom-right-radius: 10px;
}

.search-box::placeholder{
    Color: #a9a9a9;
}

.nav-items a{
    Margin-left: 20px;
}

.nav-items a img{
    Width: 30px;
}

.links-container{
    Width: 100%;
    Display: flex;
    Padding: 10px 10vw;
    Justify-content: center;
    List-style: none;
    Border-top: 1px solid #d1d1d1;
}

.link{
    Text-transform: capitalize;
    Padding: 0 10px;
    Margin: 0 5px;
    Text-decoration: none;
    Color: #383838;
    Opacity: 0.5;
    Transition: .5s;
}

.link:hover{
    Opacity: 1;
}
```

Footer.css

```
Footer{
    Position: relative;
    Width: 100%;
    Padding: 40px 10vw;
    Padding-bottom: 80px;
    Background: #383838;
}

.footer-content{
    Display: flex;
    Width: 100%;
    Justify-content: space-between;
}

.footer-content .logo{
    Height: 160px;
}

.footer-ul-container{
    Width: 45%;
    Display: flex;
    Justify-content: space-between;
}

.category{
    Width: 200px;
    Display: grid;
    Grid-template-columns: repeat(2, 1fr);
    Grid-gap: 10px;
    List-style: none;
}

.category-title{
    Grid-column: span 2;
    Text-transform: capitalize;
    Color: #fff;
    Font-size: 20px;
    Margin-bottom: 20px;
}

.category .footer-link{
    Text-decoration: none;
    Text-transform: capitalize;
    Color: rgba(255, 255, 255, 0.75);
}

.footer-link:hover{
    Color: #fff;
}



.footer-title, .info{
    Color: rgba(255, 255, 255, 0.75);
    Margin: 20px 0;
    Text-transform: capitalize;
}

.footer-title{
    Margin-top: 80px;
    Color: #fff;
}

.footer-social-container{
    Margin-top: 40px;
    Display: flex;
    Justify-content: space-between;
}

.social-link{
    Color: #fff;
    Margin-left: 20px;
    Text-transform: capitalize;
}

.social-link:nth-child(1){
    Margin-left: 0;
}

.footer-credit{
    Width: 100%;
    Padding: 10px;
    Position: absolute;
    Left: 0;
    Bottom: 0;
    Text-align: center;
    Color: #fff;
    Background: rgba(0, 0, 0, 0.2);
```

```
}
```

Home.js

```
Const productContainers =
[…document.querySelectorAll(‘.product-container’)];
Const nxtBtn = […document.querySelectorAll(‘.nxt-btn’)];
Const preBtn = […document.querySelectorAll(‘.pre-btn’)];

productContainers.forEach((item, i) => {
   let containerDimenstions = item.getBoundingClientRect();
   let containerWidth = containerDimenstions.width;

   nxtBtn[i].addEventListener(‘click’, () => {
      item.scrollLeft += containerWidth;
   })

   preBtn[i].addEventListener(‘click’, () => {
      item.scrollLeft -= containerWidth;
   })
})
```

Nav.js

```
Const createNav = () => {
    Let nav = document.querySelector(‘.navbar’);

    Nav.innerHTML = `
        <div class=”nav”>
            <img src=”img/dark-logo.png” class=”brand-logo”
alt=””>
            <div class=”nav-items”>
                <div class=”search”>
                    <input type=”text” class=”search-box”
placeholder=”search brand, product”>
                    <button class=”search-btn”>search</button>
                </div>
                <a href=”#”><img src=”img/user.png” alt=””></a>
                <a href=”#”><img src=”img/cart.png” alt=””></a>
            </div>
        </div>
        <ul class=”links-container”>
            <li class=”link-item”><a href=”#”
class=”link”>home</a></li>
            <li class=”link-item”><a href=”#”
class=”link”>women</a></li>
            <li class=”link-item”><a href=”#”
class=”link”>men</a></li>
            <li class=”link-item”><a href=”#”
class=”link”>kids</a></li>
            <li class=”link-item”><a href=”#”
class=”link”>accessories</a></li>
        </ul>
    `;
}
```

```
createNav();
```

footer.js

```
const createFooter = () => {
    let footer = document.querySelector(‘footer’);

    footer.innerHTML = `
    <div class=”footer-content”>
        <img src=”img/light-logo.png” class=”logo” alt=””>
        <div class=”footer-ul-container”>
            <ul class=”category”>
                <li class=”category-title”>men</li>
                <li><a href=”#” class=”footer-link”>t-shirts</a></li>
                <li><a href=”#” class=”footer-link”>sweatshirts</a></li>
                <li><a href=”#” class=”footer-link”>shirts</a></li>
                <li><a href=”#” class=”footer-link”>jeans</a></li>
                <li><a href=”#” class=”footer-link”>trousers</a></li>
                <li><a href=”#” class=”footer-link”>shoes</a></li>
                <li><a href=”#” class=”footer-link”>casuals</a></li>
                <li><a href=”#” class=”footer-link”>formals</a></li>
                <li><a href=”#” class=”footer-link”>sports</a></li>
                <li><a href=”#” class=”footer-
link”>watch</a></li>
            </ul>
            <ul class=”category”>
                <li class=”category-title”>women</li>
                <li><a href=”#” class=”footer-link”>t-
shirts</a></li>
                <li><a href=”#” class=”footer-
link”>sweatshirts</a></li>
                <li><a href=”#” class=”footer-link”>shirts</a></li>
                <li><a href=”#” class=”footer-link”>jeans</a></li>
                <li><a href=”#” class=”footer-
link”>trousers</a></li>
                <li><a href=”#” class=”footer-link”>shoes</a></li>
                <li><a href=”#” class=”footer-
link”>casuals</a></li>
                <li><a href=”#” class=”footer-
link”>formals</a></li>
                <li><a href=”#” class=”footer-
link”>sports</a></li>
                <li><a href=”#” class=”footer-
link”>watch</a></li>
            </ul>
        </div>
    </div>
    <p class=”footer-title”>about company</p>
    <p class=”info”>Lorem ipsum, dolor sit amet consectetur
adipisicing elit. Repellat tempore ad suscipit, eos eius
quisquam sed optio nisi quaerat fugiat ratione et vero maxime
praesentium, architecto minima reiciendis iste quo deserunt
assumenda alias ducimus. Ullam odit maxime id voluptates
rerum tenetur corporis laboriosam! Cum error ipsum laborum
tempore in rerum necessitatibus nostrum nobis modi! Debitis
adipisci illum nemo aperiam sed, et accusamus ut officiis.
```

```
Laborum illo exercitationem quo culpa reprehenderit
excepturi distinctio tempore cupiditate praesentium nisi ut
iusto, assumenda perferendis facilis voluptas autem fuga sunt
ab debitis voluptatum harum eum. Asperiores, natus! Est
deserunt incidunt quasi placeat omnis, itaque harum?</p>
    <p class=”info”>support emails – help@clothing.com,
customersupport@clothing.com</p>
    <p class=”info”>telephone – 180 00 00 001, 180 00 00
002</p>
    <div class=”footer-social-container”>
        <div>
            <a href=”#” class=”social-link”>terms & services</a>
            <a href=”#” class=”social-link”>privacy page</a>
        </div>
        <div>
            <a href=”#” class=”social-link”>instagram</a>
            <a href=”#” class=”social-link”>facebook</a>
            <a href=”#” class=”social-link”>twitter</a>
        </div>
    </div>
    <p class=”footer-credit”>Clothing, Best apparels online
store</p>
    `;
}

createFooter();
```

product.html

```
<DOCTYPE html>
    <html lang=”en”>
```

```
        <head>
            <meta charset=”UTF-8”>
            <meta http-equiv=”X-UA-Compatible” content=”IE-
edge”>
            <meta name=”viewport” content=”width=device-
width, initial-scale=1.0”>
            <title>
                Clothing – Best Apparels Online Store
            </title>

            <link rel=”stylesheet” href=”CSS/home.css”>
            <link rel=”stylesheet” href=”CSS/product.css”>

        </head>
        <body>
            <nav class=”navbar”></nav>
            <section class=”product-details”>
                <div class=”image-slider”>
                    <div class=”product-images”>
                        <img src=”img/images.jpg” class=”active”
alt=””>
                        <img src=”img/images1.webp” alt=””>
                        <img src=”img/images2.jpg” alt=””>
                        <img src=”img/product image 1.png” alt=””>
                    </div>
                </div>
                <div class=”details”>
                    <h2 class=”product-brand”>calvin klein</h2>
                    <p class=”product-short-des”>Lorem ipsum dolor
sit, amet consectetur adipisicing elit.</p>
                    <span class=”product-price”>$99</span>
                    <span class=”product-actual-price”>$200</span>
                    <span class=”product-discount”>( 50% off
)</span>
```

```
               <p class=”product-sub-heading”>select size</p>

               <input type=”radio” name=”size” value=”s”
checked hidden id=”s-size”>
               <label for=”s-size” class=”size-radio-btn
check”>s</label>
               <input type=”radio” name=”size” value=”m”
hidden id=”m-size”>
               <label for=”m-size” class=”size-radio-
btn”>m</label>
               <input type=”radio” name=”size” value=”l”
hidden id=”l-size”>
               <label for=”l-size” class=”size-radio-
btn”>l</label>
               <input type=”radio” name=”size” value=”xl”
hidden id=”xl-size”>
               <label for=”xl-size” class=”size-radio-
btn”>xl</label>
               <input type=”radio” name=”size” value=”xxl”
hidden id=”xxl-size”>
               <label for=”xxl-size” class=”size-radio-
btn”>xxl</label>

               <button class=”btn cart-btn”>add to
cart</button>
               <button class=”btn”>add to wishlist</button>
            </div>
         </section>
<section class=”product”>
   <h2 class=”product-category”>shirts</h2>
   <button class=”pre-btn”><img src=”img/arrow6.png”
alt=””></button>
```

```
    <button class=”nxt-btn”><img src=”img/arrow6.png”
alt=””></button>

    <div class=”product-container”>
        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>50% off</span>
                <img src=”img/men1.webp” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>20% off</span>
                <img src=”img/men2.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>30% off</span>
                <img src=”img/men3.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>40% off</span>
                <img src=”img/men4.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>
        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>50% off</span>
                <img src=”img/men5.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
                <span class=”discount-tag”>50% off</span>
                <img src=”img/men6.jpg” class=”product-thumb”
alt=””>
                <button class=”card-btn”>add to whislist</button>
            </div>
            <div class=”product-info”>
                <h2 class=”product-brand”>BRAND</h2>
                <p class=”product-short-des”>a short line about the
cloth..</p>
                <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
            </div>
        </div>

        <div class=”product-card”>
            <div class=”product-image”>
        <span class=”discount-tag”>50% off</span>
        <img src=”img/men7.jpg” class=”product-thumb”
alt=””>
        <button class=”card-btn”>add to whislist</button>
      </div>
      <div class=”product-info”>
        <h2 class=”product-brand”>BRAND</h2>
        <p class=”product-short-des”>a short line about the
cloth..</p>
        <span class=”price”>$20</span><span
class=”actual-price”>$40</span>
      </div>
    </div>

  </div>
</section>
      <footer>

      </footer>
      <script src=”js/nav.js”></script>
      <script src=”js/footer.js”></script>
      <script src=”js/home.js”></script>
      <script src=”js/product.js”></script>
      </body>
  </html>
```

Product.css

```
.product-details{
  Width: 100%;
  Padding: 60px 10vw;
```

```
    Display: flex;
    Justify-content: space-between;
}

.image-slider{
    Width: 500px;
    Height: 500px;
    Position: relative;
    Background-image: url(‘../img/product\ image\ 1.png’);
    Background-size: cover;
}


.details{
    Width: 50%;
}

.details .product-brand{
    Text-transform: capitalize;
    Font-size: 30px;
}

.details .product-short-des{
    Font-size: 25px;
    Line-height: 30px;
    Height: auto;
    Margin: 15px 0 30px;
}

.product-price{
    Font-weight: 700;
    Font-size: 30px;
}
```

```css
.product-actual-price{
    Font-size: 30px;
    Opacity: 0.5;
    Text-decoration: line-through;
    Margin: 0 20px;
    Font-weight: 300;
}

.product-discount{
    Color: #ff7d7d;
    Font-size: 20px;
}

.product-sub-heading{
    Font-size: 30px;
    Text-transform: uppercase;
    Margin: 60px 0 10px;
    Font-weight: 300;
}

.size-radio-btn{
    Display: inline-block;
    Width: 80px;
    Height: 80px;
    Text-align: center;
    Font-size: 20px;
    Border: 1px solid #383838;
    Border-radius: 50%;
    Margin: 10px;
    Margin-left: 0;
    Line-height: 80px;
    Text-transform: uppercase;
    Color: #383838;
    Cursor: pointer;
```

```
}

.size-radio-btn.check{
   Background: #383838;
   Color: #fff;
}

.btn{
   Width: 48%;
   Padding: 20px;
   Border-radius: 5px;
   Background: none;
   Border: 1px solid #383838;
   Color: #383838;
   Font-size: 20px;
   Cursor: pointer;
   Margin: 20px 0;
   Text-transform: capitalize;
}

.cart-btn{
   Margin-right: 2%;
   Background: #383838;
   Color: #fff;
}


Detail-des{
   Padding: 0 10vw;
   Text-transform: capitalize;
}

.heading{
   Font-size: 30px;
```

```
    Margin-bottom: 30px;
}

.des{
    Color: #383838;
    Line-height: 25px;
}







.product-images{
    Position: absolute;
    Bottom: 20px;
    Left: 50%;
    Transform: translateX(-50%);
    Width: 90%;
    Background: #fff;
    Border-radius: 5px;
    Display: grid;
    Grid-template-columns: repeat(4, 1fr);
    Height: 100px;
    Grid-gap: 10px;
    Padding: 10px;
}

.product-images img{
    Width: 100%;
    Height: 80px;
    Object-fit: cover;
    Cursor: pointer;
}

.product-images img.active{
    Opacity: 0.5;
}
```

Product.js

```
Const productImages = document.querySelectorAll(“.product-
images img”); // selecting all image thumbs
Const productImageSlide = document.querySelector(“.image-
slider”); // seclecting image slider element

Let activeImageSlide = 0; // default slider image

productImages.forEach((item, i) => { // loopinh through each
image thumb
    item.addEventListener(‘click’, () => { // adding click event
to each image thumbnail

productImages[activeImageSlide].classList.remove(‘active’);
// removing active class from current image thumb
        item.classList.add(‘active’); // adding active class to the
current or clicked image thumb
        productImageSlide.style.backgroundImage =
`url(‘${item.src}’)`; // setting up image slider’s background
image
        activeImageSlide = I; // updating the image slider
variable to track current thumb
    })
})
```

```
Const sizeBtns = document.querySelectorAll(‘.size-radio-btn’); // selecting size buttons
Let checkedBtn = 0; // current selected button

sizeBtns.forEach((item, i) => { // looping through each button
   item.addEventListener(‘click’, () => { // adding click event to each
       sizeBtns[checkedBtn].classList.remove(‘check’); // removing check class from the current button
       item.classList.add(‘check’); // adding check class to clicked button
       checkedBtn = I; // upading the variable
   })
})
```

Search.html

```
<head>
  <link rel=”stylesheet” href=”css/home.css”>
  <link rel=”stylesheet” href=”css/search.css”>
</head>
<body>
   <nav class=”navbar”></nav>
   <footer></footer>

   <script src=”js/nav.js”></script>
   <script src=”js/footer.js”></script>
</body>
```

```
<section class=”search-results”>
   <h2 class=”heading”>search results for <span>product</span></h2>
```

```
</section>
<img src="img/404.png" class="four-0-four-image" alt="">
<p class="four-0-four-msg">look like you are lost. Head to
beack to our <a href="#">homepage</a></p>
```

Search.css

```
.search-results{
   Width: 100%;
   Padding: 60px 0;
}

.heading{
   Font-size: 20px;
   Text-transform: capitalize;
   Font-weight: 400;
   Margin-bottom: 40px;
   Padding: 0 10vw;
}

.heading span{
   Font-weight: 700;
}
```

Product.css

```
.product-container{
   Display: grid;
   Grid-template-columns: repeat(4, 1fr);
   Height: auto;
   Grid-row-gap: 40px;
}
```

Footer.css

```
.four-0-four-image{
   Display: block;
   Margin: 60px auto;
}

.four-0-four-msg{
   Text-align: center;
   Text-transform: capitalize;
   Color: #383838;
}

.four-0-four-msg a{
   Color: #383838;
}
```

Server page

Server.js

```
“scripts”: {
   “start”: “nodemon server.js”
}
Const express = require(‘express’);
Const admin = require(‘firebase-admin’);
Const bcrypt = require(‘bcrypt’);
Const path = require(‘path’);
// declare static path
Let staticPath = path.join(__dirname, “public”);
//intializing express.js
```

```
Const app = express();

//middlewares
App.use(express.static(staticPath));

App.listen(3000, () => {
    Console.log(‘listening on port 3000…….’);
})
//routes
//home route
App.get(“/”, (req, res) => {
    Res.sendFile(path.join(staticPath, “index.html”));
})

App.get(‘/404’, (req, res) => {
    Res.sendFile(path.join(staticPath, “404.html”));
})

App.use((req, res) => {
    Res.redirect(‘/404’);
})
//signup route
App.get(‘/signup’, (req, res) => {
    Res.sendFile(path.join(staticPath, “signup.html”));
})
// firebase admin setup
Let serviceAccount = require(“path of key file”);

Admin.initializeApp({
  Credential: admin.credential.cert(serviceAccount)
});

Let db = admin.firestore();
// store user in db
```

```
Db.collection(‘users’).doc(email).get()
.then(user => {
    If(user.exists){
        Return res.json({‘alert’: ‘email already exists’});
    } else{
        // encrypt the password before storing it.
        Bcrypt.genSalt(10, (err, salt) => {
            Bcrypt.hash(password, salt, (err, hash) => {
                Req.body.password = hash;
                Db.collection(‘users’).doc(email).set(req.body)
                .then(data => {
                    Res.json({
                        Name: req.body.name,
                        Email: req.body.email,
                        Seller: req.body.seller,
                    })
                })
            })
        })
    }
})
Db.collection(‘users’).doc(email).get()
.then(…)
Const processData = (data) => {
    Loader.style.display = null;
    If(data.alert){
        showAlert(data.alert);
    } else if(data.name){
        // create authToken
        Data.authToken = generateToken(data.email);
        sessionStorage.user = JSON.stringify(data);
        location.replace(‘/’);
    }
}
```

Signup page

Form.html

```
<img src=”img/loader.gif” class=”loader” alt=””>
<div class=”container”>
    <img src=”img/dark-logo.png” class=”logo” alt=””>
    <div>
        <input type=”text” autocomplete=”off” id=”name”
placeholder=”name”>
        <input type=”email” autocomplete=”off” id=”email”
placeholder=”email”>
        <input type=”password” autocomplete=”off”
id=”password” placeholder=”password”>
        <input type=”text” autocomplete=”off” id=”number”
placeholder=”number”>
        <input type=”checkbox” checked class=”checkbox”
id=”terms-and-cond”>
        <label for=”terms-and-cond”>agree to our <a
href=””>terms and conditions</a></label>
        <br>
        <input type=”checkbox” class=”checkbox”
id=”notification”>
        <label for=”notification”>receive upcoming offers and
events mails</a></label>
        <button class=”submit-btn”>create account</button>
    </div>
    <a href=”/login” class=”link”>already have an account?
Log in here</a>
</div>
<div class=”alert-box”>
    <img src=”img/error.png” class=”alert-img” alt=””>
    <p class=”alert-msg”>Error message</p>
</div>
```

```
<script src=”js/form.js”></script>
```

Form.css

```
*{
   Margin: 0;
   Padding: 0;
   Box-sizing: border-box;
}

Body{
   Width: 100%;
   Min-height: 100vh;
   Display: flex;
   Justify-content: center;
   Align-items: center;
   Background: #f5f5f5;
   Font-family: ‘roboto’, sans-serif;
}

.loader{
   Position: absolute;
   Top: 50%;
   Left: 50%;
   Transform: translate(-50%, -50%);
   Width: 100px;
}
.logo{
   Height: 80px;
   Display: block;
   Margin: 0 auto 50px;
}
```

```
Input[type=”text”],
Input[type=”password”],
Input[type=”email”],
Textarea{
    Display: block;
    Width: 300px;
    Height: 40px;
    Padding: 20px;
    Border-radius: 5px;
    Background: #fff;
    Border: none;
    Outline: none;
    Margin: 20px 0;
    Text-transform: capitalize;
    Color: #383838;
    Font-size: 14px;
    Box-shadow: 0 4px 10px rgba(0, 0, 0, 0.01);
    Font-family: ‘roboto’, sans-serif;
}

::placeholder{
    Color: #383838;
}

.submit-btn{
    Width: 300px;
    Height: 40px;
    Text-align: center;
    Line-height: 40px;
    Background: #383838;
    Color: #fff;
    Border-radius: 2px;
    Text-transform: capitalize;
    Border: none;
    Cursor: pointer;
    Display: block;
    Margin: 30px 0;
}

/* checkbox styles */

.checkbox{
    -webkit-appearance: none;
    Position: relative;
    Width: 15px;
    Height: 15px;
    Border-radius: 2px;
    Background: #fff;
    Border: 1px solid #383838;
    Cursor: pointer;
}

.checkbox:checked{
    Background: #383838;
}

.checkbox::after{
    Content: ‘’;
    Position: absolute;
    Top: 60%;
    Left: 50%;
    Transform: translate(-50%, -50%);
    Width: 80%;
    Height: 100%;
    Pointer-events: none;
    Background-image: url(../img/check.png);
    Background-size: contain;
    Background-repeat: no-repeat;
    Display: none;
}

.checkbox:checked::after{
    Display: block;
}

Label{
    Text-transform: capitalize;
    Display: inline-block;
    Margin-bottom: 10px;
    Font-size: 14px;
    Color: #383838;
}

Label a{
    Color: #383838;
}

.link{
    Color: #383838;
    Text-transform: capitalize;
    Text-align: center;
    Display: block;
}
/* alert */
.alert-box{
    Width: 300px;
    Min-height: 150px;
    Background: #fff;
    Border-radius: 10px;
    Box-shadow: 0 5px 100px rgba(0, 0, 0, 0.05);
    Position: absolute;
    Top: 60%;
    Left: 50%;
    Transform: translate(-50%, -50%);
    Padding: 20px;
    Opacity: 0;
    Pointer-events: none;
    Transition: 1s;
}

.alert-box.show{
    Opacity: 1;
    Pointer-events: all;
    Top: 50%;
}

.alert-img{
    Display: block;
    Margin: 10px auto 20px;
    Height: 60px;
}

.alert-msg{
    Color: #e24c4b;
    Font-size: 20px;
    Text-transform: capitalize;
    Text-align: center;
    Line-height: 30px;
    Font-weight: 500;
}
```

Form.js

```
Const loader = document.querySelector(‘.loader’);
```

```
// select inputs
Const submitBtn = document.querySelector(‘.submit-btn’);
Const name = document.querySelector(‘#name’);
Const email = document.querySelector(‘#email’);
Const password = document.querySelector(‘#password’);
Const number = document.querySelector(‘#number’);
Const tac = document.querySelector(‘#terms-and-cond’);
Const notification = document.querySelector(‘#notification’);
submitBtn.addEventListener(‘click’, () => {
        if(name.value.length < 3){
            showAlert(‘name must be 3 letters long’);
        } else if(!email.value.length){
            showAlert(‘enter your email’);
        } else if(password.value.length < 8){
            showAlert(‘password should be 8 letters long’);
        } else if(!number.value.length){
            showAlert(‘enter your phone number’);
        } else if(!Number(number.value) || number.value.length
< 10){
            showAlert(‘invalid number, please enter valid one’);
        } else if(!tac.checked){
            showAlert(‘you must agree to our terms and
conditions’);
        } else{
            // submit form
        }
})
If(name.value.length < 3){
    showAlert(‘name must be 3 letters long’);
}
showAlert(‘name must be 3 letters long’);
// alert function
Const showAlert = (msg) => {
    Let alertBox = document.querySelector(‘.alert-box’);
    Let alertMsg = document.querySelector(‘.alert-msg’);
    alertMsg.innerHTML = msg;
    alertBox.classList.add(‘show’);
    setTimeout(() => {
        alertBox.classList.remove(‘show’);
    }, 3000);
}
// send data function
Const sendData = (path, data) => {
    Fetch(path, {
        Method: ‘post’,
        Headers: new Headers({‘Content-Type’:
‘application/json’}),
        Body: JSON.stringify(data)
    }).then((res) => res.json())
    .then(response => {
        processData(response);
    })
}
Else{
    // submit form
    Loader.style.display = ‘block’;
    sendData(‘/signup’, {
        name: name.value,
        email: email.value,
        password: password.value,
        number: number.value,
        tac: tac.checked,
        notification: notification.checked,
        seller: false
    })
}
Const compareToken = (token, key) => {
    Let string = ‘’;
    For(let I = 0; I < token.length; i=i+2){
        Let index1 = char.indexOf(token[i]);
        Let index2 = char.indexOf(token[i+1]);
        String += char[index1 + index2];
    }
    If(string === key){
        Return true;
    }
    Return false;
}
```

Signup-post

```
App.use(express.json());
App.post('/signup', (req, res) => {
    Let { name, email, password, number, tac, notification } =
req.body;

    // form validations
    If(name.length < 3){
        Return res.json({'alert': 'name must be 3 letters long'});
    } else if(!email.length){
        Return res.json({'alert': 'enter your email'});
    } else if(password.length < 8){
        Return res.json({'alert': 'password should be 8 letters
long'});
    } else if(!number.length){
        Return res.json({'alert': 'enter your phone number'});
    } else if(!Number(number) || number.length < 10){
        Return res.json({'alert': 'invalid number, please enter
valid one'});
    } else if(!tac){
        Return res.json({‘alert’: ‘you must agree to our terms and conditions’});
    }
})
Let { name, email, password, number, tac, notification } = req.body;
If(name.length < 3){
    Return res.json({‘alert’: ‘name must be 3 letters long’});
} else if …..
JSON = {
   ‘key’: ‘value’
}
Const processData = (data) => {
    Loader.style.display = null;
    If(data.alert){
        showAlert(data.alert);
    }
}
```

Token.js

```
<script src=”js/token.js”></script>
Const compareToken = (token, key) => {
    Let string = ‘’;
    For(let I = 0; I < token.length; i=i+2){
        Let index1 = char.indexOf(token[i]);
        Let index2 = char.indexOf(token[i+1]);
        String += char[index1 + index2];
    }
    If(string === key){
        Return true;
    }
    Return false;
}
```

Chatbot

```
<script>
                Window.watsonAssistantChatOptions = {
                   integrationID: “c4969e38-901a-4c1b-a659-d054daf___”, // The ID of this integration.
                   Region: “eu-gb”, // The region your integration is hosted in.
                   serviceInstanceID: “9f76390e-b4b0-4b50-85d1-d22c61d___”, // The ID of your service instance.
                   onLoad: function(instance) {
                       instance.render();  // Renders the web chat widget on your page.
                       }
            };
                setTimeout(function(){
                   const t=document.createElement(‘script’);
               t.src=https://web-chat.global.assistant.watson.appdomain.cloud/loadWatsonAssistantChat.js;
                       document.head.appendChild(t);
                });
               </script>
```

Server.js

```
// add product
App.get(‘/add-product’, (req, res) => {
   Res.sendFile(path.join(staticPath, “addProduct.html”));
})
```

Add Product Page – Design

```
<head>
   // other head tags
   <link rel=”stylesheet” href=”css/signup.css”>
   <link rel=”stylesheet” href=”css/addProduct.css”>
</head>
<body>
   <img src=”img/loader.gif” class=”loader” alt=””>

   <div class=”alert-box”>
      <img src=”img/error.png” class=”alert-img” alt=””>
      <p class=”alert-msg”></p>
   </div>

   <script src=”js/token.js”></script>
   <script src=”js/addProduct.js”></script>
</body>
<img src=”img/dark-logo.png” class=”logo” alt=””>

<div class=”form”>
   <input type=”text” id=”product-name”
placeholder=”product name”>
   <input type=”text” id=”short-des” placeholder=”short line
about the product”>
   <textarea id=”des” placeholder=”detail
description”></textarea>

   <!—product image →
   <div class=”product-info”>
      <div class=”product-image”><p class=”text”>product
image</p></div>
      <div class=”upload-image-sec”>
         <!—upload inputs →
         <p class=”text”><img src=”img/camera.png”
alt=””>upload image</p>
```

```
            <div class=”upload-catalouge”>
                <input type=”file” class=”fileupload” id=”first-file-upload-btn” hidden>
                <label for=”first-file-upload-btn” class=”upload-image”></label>
                <input type=”file” class=”fileupload” id=”second-file-upload-btn” hidden>
                <label for=”second-file-upload-btn” class=”upload-image”></label>
                <input type=”file” class=”fileupload” id=”third-file-upload-btn” hidden>
                <label for=”third-file-upload-btn” class=”upload-image”></label>
                <input type=”file” class=”fileupload” id=”fourth-file-upload-btn” hidden>
                <label for=”fourth-file-upload-btn” class=”upload-image”></label>
            </div>
        </div>
        <div class=”select-sizes”>
            <p class=”text”>size available</p>
            <div class=”sizes”>
                <input type=”checkbox” class=”size-checkbox” id=”xs” value=”xs”>
                <input type=”checkbox” class=”size-checkbox” id=”s” value=”s”>
                <input type=”checkbox” class=”size-checkbox” id=”m” value=”m”>
                <input type=”checkbox” class=”size-checkbox” id=”l” value=”l”>
                <input type=”checkbox” class=”size-checkbox” id=”xl” value=”xl”>
                <input type=”checkbox” class=”size-checkbox” id=”xxl” value=”xxl”>
                <input type=”checkbox” class=”size-checkbox”
id=”xxxl” value=”xxxl”>
            </div>
        </div>
    </div>

    <div class=”product-price”>
        <input type=”number” id=”actual-price”
placeholder=”actual price”>
        <input type=”number” id=”discount”
placeholder=”discount percentage”>
        <input type=”number” id=”sell-price”
placeholder=”selling price”>
    </div>

    <input type=”number” id=”stock” min=”20”
placeholder=”item in sstocks (minimum 20)”>

    <textarea id=”tags” placeholder=”Enter categories here, for
example – Men, Jeans, Blue Jeans, Rough jeans (you sholud
add men or women at start)”></textarea>

    <input type=”checkbox” class=”checkbox” id=”tac”
checked>
    <label for=”tac”>clothing take 30% from your total
sell</label>

    <div class=”buttons”>
        <button class=”btn” id=”add-btn”>add product</button>
        <button class=”btn” id=”save-btn”>save draft</button>
    </div>
</div>
```

Addproduct.css

```
Body{
   Display: block;
   Padding: 0 10vw;
}

.logo{
   Margin: 20px auto 50px;
}

Input, textarea{
   Font-weight: 500;
}

Input:not(input[type=”checkbox”]){
   Width: 100%;
}

Textarea{
   Width: 100%;
   Height: 270px;
   Resize: none;
   Padding: 10px 20px;
}

.product-info{
   Width: 100%;
   Height: 500px;
   Display: grid;
   Grid-template-columns: .75fr 1.25fr;
   Grid-template-rows: repeat(2, 1fr);
   Grid-gap: 20px;
   Margin-bottom: 20px;
```

```
}

.product-image{
   Display: flex;
   Justify-content: center;
   Align-items: center;
   Background: #fff;
   Background-size: cover;
   Border-radius: 10px;
   Grid-row: span 2;
   Text-shadow: 0 0 10px #fff;
}

.text{
   Text-transform: capitalize;
   Color: #383838;
   Font-size: 20px;
   Font-weight: 500;
}

.upload-image-sec, .select-sizes{
   Background: #fff;
   Border-radius: 10px;
   Padding: 20px;
}

.text img{
   Height: 20px;
   Margin-right: 10px;
}

.upload-catalouge{
   Width: 100%;
   Margin: 20px 0;
   Display: grid;
   Grid-template-columns: repeat(4, 100px);
   Grid-gap: 10px;
}

.upload-image{
   Width: 100%;
   Height: 100px;
   Background: #f5f5f5;
   Cursor: pointer;
   Background-size: cover;
}

.upload-image:hover{
   Background: rgba(0, 0, 0, 0.2);
   Background-size: cover;
}


.sizes{
   Margin-top: 30px;
}

.size-checkbox{
   -webkit-appearance: none;
   Width: 100px;
   Height: 40px;
   Border-radius: 5px;
   Border: 1px solid #383838;
   Cursor: pointer;
   Margin-bottom: 10px;
   Margin-right: 10px;
   Position: relative;
   Color: #383838;
```

```
}

.size-checkbox::after{
   Content: attr(value);
   Position: absolute;
   Top: 50%;
   Left: 50%;
   Transform: translate(-50%, -50%);
   Font-size: 16px;
   Text-transform: uppercase;
}

.size-checkbox:checked{
   Background: #383838;
   Color: #fff;
}

Input::-webkit-outer-spin-button,
Input::-webkit-inner-spin-button{
   -webkit-appearance: none;
   Margin: 0;
}

.product-price{
   Display: grid;
   Grid-template-columns: repeat(3, 1fr);
   Grid-gap: 20px;
}

.product-price input{
   Margin: 0;
}

.buttons{
   Margin: 20px 0 50px;
}

.btn{
   Padding: 10px 30px;
   Text-transform: capitalize;
   Color: #fff;
   Background: #383838;
   Border-radius: 5px;
   Border: none;
   Outline: none;
   Margin-right: 10px;
   Cursor: pointer;
}

#save-btn{
   Background: #a9a9a9;
}
Input::-webkit-outer-spin-button,
Input::-webkit-inner-spin-button{
   -webkit-appearance: none;
   Margin: 0;
}
```

Signup.css

```
Input[type=”text”],
Input[type=”password”],
Input[type=”email”],
Input[type=”number”], // add this new line
Textarea{
    // properties
}
```

Form submission

```
Let user = JSON.parse(sessionStorage.user || null);
Let loader = document.querySelector('.loader');

// checknig user is logged in or not
Window.onload = () => {
    If(user){
        If(!compareToken(user.authToken, user.email)){
            Location.replace('/login');
        }
    } else{
        Location.replace('/login');
    }
}
```

Addproduct.js

```
// price inputs

Const actualPrice = document.querySelector('#actual-price');
Const discountPercentage =
document.querySelector('#discount');
Const sellingPrice = document.querySelector('#sell-price');
discountPercentage.addEventListener('input', () => {
    if(discountPercentage.value > 100){
        discountPercentage.value = 90;
    } else{
        Let discount = actualPrice.value *
discountPercentage.value / 100;
        sellingPrice.value = actualPrice.value – discount;
    }
})
sellingPrice.addEventListener(‘input’, () => {
    let discount = (sellingPrice.value / actualPrice.value) * 100;
    discountPercentage.value = discount;
})
```

Server.js

AWS Config

```
// aws config
Const aws = require(‘aws-sdk’);
Const dotenv = require(‘dotenv’);

Dotenv.config();

// aws parameters
Const region = “ap-south-1”;
Const bucketName = “ecom-website-tutorial-2”;
Const accessKeyId = process.env.AWS_ACCESS_KEY;
Const secretAccessKey = process.env.AWS_SECRET_KEY;

Aws.config.update({
    Region,
    accessKeyId,
    secretAccessKey
})

// init s3
Const s3 = new aws.S3();
// generate image upload link
```

```
Async function generateUrl(){
  Let date = new Date();
  Let id = parseInt(Math.random() * 10000000000);

  Const imageName = `${id}${date.getTime()}.jpg`;

  Const params = ({
    Bucket: bucketName,
    Key: imageName,
    Expires: 300, //300 ms
    ContentType: ‘image/jpeg’
  })
  Const uploadUrl = await
s3.getSignedUrlPromise(‘putObject’, params);
  Return uploadUrl;
}

// generate image upload link
Async function generateUrl(){
  Let date = new Date();
  Let id = parseInt(Math.random() * 10000000000);

  Const imageName = `${id}${date.getTime()}.jpg`;

  Const params = ({
    Bucket: bucketName,
    Key: imageName,
    Expires: 300, //300 ms
    ContentType: ‘image/jpeg’
  })
  Const uploadUrl = await
s3.getSignedUrlPromise(‘putObject’, params);
  Return uploadUrl;
}
```

```
// get the upload link
App.get(‘/s3url’, (req, res) => {
    generateUrl().then(url => res.json(url));
})
// upload image handle
Let uploadImages =
document.querySelectorAll(‘.fileupload’);
Let imagePaths = []; // will store all uploaded images paths;
uploadImages.forEach((fileupload, index) => {
    fileupload.addEventListener(‘change’, () => {
        const file = fileupload.files[0];
        let imageUrl;

        if(file.type.includes(‘image’)){
            // means user uploaded an image

        } else{
            showAlert(‘upload image only’);
        }
    })
})
If(file.type.includes(‘image’)){
    // means user uploaded an image
    Fetch(‘/s3url’).then(res => res.json())
    .then(url => {
        Fetch(url,{
            Method: ‘PUT’,
            Headers: new Headers({‘Content-Type’:
‘multipart/form-data’}),
            Body: file
        }).then(res => {
            Console.log(url)
        })
    })
}
If(file.type.includes(‘image’)){
    // means user uploaded an image
    Fetch(‘/s3url’).then(res => res.json())
    .then(url => {
        Fetch(url,{
            Method: ‘PUT’,
            Headers: new Headers({‘Content-Type’:
‘multipart/form-data’}),
            Body: file
        }).then(res => {
            imageUrl = url.split(“?”)[0];
            imagePaths[index] = imageUrl;
            let label =
document.querySelector(`label[for=${fileupload.id}]`);
            label.style.backgroundImage = `url(${imageUrl})`;
            let productImage = document.querySelector(‘.product-
image’);
            productImage.style.backgroundImage =
`url(${imageUrl})`;
        })
    })
}
```

CHAPTER 10

ADVANTAGES:

- ✓ Drive Traffic
- ✓ Provide Relevant Material
- ✓ Engage Customers
- ✓ Transform Shoppers to Clients
- ✓ Increase Average Order Value
- ✓ Boost Number of Items per Order
- ✓ Control Retailing and Inventory Rules
- ✓ Lower Work and Overhead
- ✓ Provide Reports
- ✓ Offer Recommendations and Direction

DISADVANTAGES:

- ☒ Significant investments required
- ☒ Too many choices
- ☒ The complex onboarding process
- ☒ Lack of data analytics capability
- ☒ The 'cold start' problem
- ☒ Inability to capture changes in user behavior
- ☒ Privacy concerns

CHAPTER 9

9.1 Performance Metrics

The concept of Transfer learning is used to overcome the issues of the small size Fashion dataset. Therefore we pre-train the classification models on the DeepFashion dataset that consists of 44,441 garment images. The networks are trained and validated on the dataset taken. The training results show a great accuracy of the model with low error, loss and good f-score

CONCLUSION 11

In this project, we have presented a novel framework for fashion recommendation that is driven by data, visually related, and simple effective recommendation systems for generating fashion product images. The proposed approach uses a two-stage phase. Initially, our proposed approach extracts the features of the image using CNN classifier ie., for instance allowing the customers to upload any random fashion image from any E-commerce website and later generating similar images to the uploaded image based on the features and texture of the input image. Such research must go forward to facilitate greater recommendation accuracy and improve the overall

experience of fashion exploration for direct and indirect
consumers alike

CHAPTER 12

FUTURE SCOPE:

The plan is to implement even more interesting features in future work. Scalable and adaptive, this website can be used by a wide range of e-commerce businesses.