Project Report

on

Aditi's Collection

Submitted in partial fulfillment of Semester-I of

Master of Computer Application (MCA)

at

University of Mumbai

Submitted by

Akanksha Anil Jagtap

Under the guidance of

Prof. Nidhi Poonia

Bharati Vidyapeeth's

Institute of Management and Information Technology

Sector 8, CBD Belapur

Navi Mumbai

2023 - 2025

Bharati Vidyapeeth's

Institute of Management and Information Technology

Navi Mumbai

Certificate of Approval

This is to certify that the Project titled ' **Aditi's Collection** ' is successfully done by **Akanksha Anil Jagtap** during Mini Project of her course in partial fulfillment of **Masters of Computer Application** under the **University of Mumbai, Mumbai**, through the Bharati Vidypeeths Institute of Management and Information Technology, Navi Mumbai carried out by her under our guidance and supervision.

Sign & Date External Examiner
Guide Signature and Date

Principal
Dr. Suhasini Vijaykumar

College Seal

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction to Project

In a world increasingly shaped by digital experiences, the ”Aditi’s Collection Website” project emerges as a response to the evolving needs of a home-based business specializing in artisanal handcrafted items. This project aims to establish a vibrant online presence for Aditi’s Collection, providing a dynamic platform for showcasing and selling a diverse range of products, including wall hangings, door hangings, macram mirror wall hangings, and key holders.

Aditi’s Collection, driven by a passion for craftsmanship, has thrived in traditional markets. However, recognizing the transformative potential of an online presence, this project was conceived to bring the business into the digital realm. The project seeks to overcome the limitations of a localized market, enhance customer interactions, and streamline operational processes

1.2 Domain Knowledge

Domain knowledge for the ”Aditi’s Collection Website” project refers to understanding the specific industry or field in which the project operates. In this case, the domain revolves around e-commerce for a home-based business specializing in handcrafted items, including wall hangings, door hangings, macram mirror wall hangings, and key holders. Here’s an overview of the domain knowledge:

1. Artisanal Craftsmanship:
   - Understanding the art and skill involved in creating handcrafted items.
   - Knowledge of various techniques, materials, and styles used in crafting wall hangings, door hangings, and macram mirror wall hangings.
2. Home-Based Business:
   - Awareness of the unique challenges and opportunities associated with running a business from home.

- Knowledge of inventory management, order processing, and customer interactions in a home-based setting.

3. E-Commerce:

- Understanding the fundamentals of e-commerce, including online product catalogs, shopping carts, and secure checkout processes.
- Awareness of industry standards and best practices for creating a user-friendly online shopping experience.

1.3 Problem Description

The business landscape is evolving, with digital transformation playing a pivotal role in reshaping consumer behavior and market dynamics. Aditi's Collection, a home-based business specializing in handcrafted items, faced inherent challenges within the traditional, localized market. Recognizing the need to adapt to the changing times and expand its reach, the "Aditi's Collection Website" project was initiated.

1. Limited Market Reach:
   - Aditi's Collection primarily operated within a local market, constraining its potential customer base and revenue growth.
2. Manual Business Operations:
   - The absence of an online platform resulted in manual order processing, making it prone to errors and limiting efficiency.
3. Lack of Product Visibility:
   - The business struggled to showcase its diverse range of handcrafted products to a broader audience, missing out on potential sales opportunities
4. Limited Customer Interaction:
   - Customers had limited ways to interact with Aditi's Collection, hindering the establishment of a loyal customer base.

Chapter 2

System Study

2.1 Existing system

Before the development of the new website, Aditi's Collection operated without a dedicated online platform. The business primarily relied on traditional methods of sales, such as in-person interactions, word of mouth, and limited social media presence. The absence of an online store restricted the reach of Aditi's Collection to a broader customer base and hindered the ability to showcase and sell products beyond local networks.

2.2 Limitations

1. The absence of an online presence restricted the business to local markets, missing out on potential customers beyond geographical boundaries.

2. Orders were primarily processed manually through direct customer interactions or over the phone, leading to potential errors and inefficiencies.

3. Products were not easily accessible to a wider audience, hindering the potential for increased sales and brand exposure.

4. The lack of an online system meant there was no real-time tracking of inventory levels, leading to potential stockouts or overstocking issues.

5. Customers had limited ways to interact with Aditi's Collection, making it challenging to build a loyal customer base and receive feedback.

2.3 Proposed system

The proposed system for Aditi's Collection is a comprehensive e-commerce platform designed to transform the business into a dynamic online marketplace. Leveraging cutting-edge technologies, the new system aims to address the limitations of the existing model and unlock opportunities for growth, improved customer engagement, and streamlined business operations.

2.4 Objectives

1. Create an Intuitive User Interface:

   Develop a user-friendly website design to enhance the overall user experience. Prioritize easy navigation and an intuitive layout for customers to browse products effortlessly.

2. E-commerce Functionality:

   Implement a fully functional e-commerce platform allowing customers to view products, add items to their cart, and proceed to checkout. Enable user to order product.

3. Product Management:

   Implement a dynamic product management system allowing the business owner to easily add new products. Include high-quality images and detailed descriptions for each product.

2.5 Feasibility Studies

A feasibility study is a crucial step in determining the viability and potential success of a project. It assesses various aspects, including technical, economic, operational, and scheduling feasibility. Here is an outline for the feasibility study of the "Aditi's Collection Website" project:

2.5.1 Technical

- **Technology Stack:** The project utilizes React.js for the frontend and Firestore for the backend. Development expertise in React.js and Firebase technologies is available.
- **Scalability:** The chosen technologies offer scalability, allowing the website to handle potential future growth in terms of user traffic and product catalog expansion.
- **Integration Capabilities:** Integration with Firebase services ensures seamless data storage, retrieval, and real-time updates. The technology stack supports potential future integrations, such as online payment gateways.

2.5.2 Economical

- **Cost-Benefit Analysis:** Initial development costs, including technology, tools, and human resources, have been assessed. A projected increase in sales and customer reach, facilitated by the online platform, is expected to outweigh development and operational costs.
- **Return on Investment (ROI):** The ROI is estimated based on projected revenue generation through increased sales, expanded customer base, and enhanced operational efficiency. Projections indicate a positive ROI within the anticipated timeframe.

2.5.3 Operational

- **Business Process Impact:** The project significantly impacts and streamlines business processes, automating order processing, inventory management, and customer interactions. Operational efficiency is expected to improve with the shift to online business operations.
- **Training and Adaptation:** Staff training programs have been implemented to ensure a smooth transition to the new online system. The user-friendly interface minimizes the learning curve for both customers and internal staff.

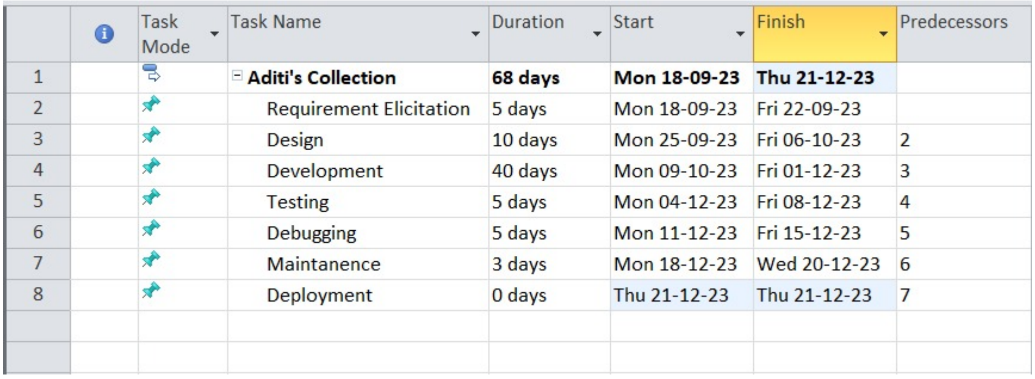
Chapter 3

System Analysis

3.1 Gantt chart

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity. This allows you to see at a glance:

- What the various activities are
- When each activity begins and ends
- How long each activity is scheduled to last
- Where activities overlap with other activities, and by how much.
- The start and end date of the whole project.



		Task Mode	Task Name	Duration	Start	Finish	Predecessors
1			**Aditi's Collection**	**68 days**	**Mon 18-09-23**	**Thu 21-12-23**	
2			Requirement Elicitation	5 days	Mon 18-09-23	Fri 22-09-23	
3			Design	10 days	Mon 25-09-23	Fri 06-10-23	2
4			Development	40 days	Mon 09-10-23	Fri 01-12-23	3
5			Testing	5 days	Mon 04-12-23	Fri 08-12-23	4
6			Debugging	5 days	Mon 11-12-23	Fri 15-12-23	5
7			Maintanence	3 days	Mon 18-12-23	Wed 20-12-23	6
8			Deployment	0 days	Thu 21-12-23	Thu 21-12-23	7

Figure 3.1: Gantt Chart Table

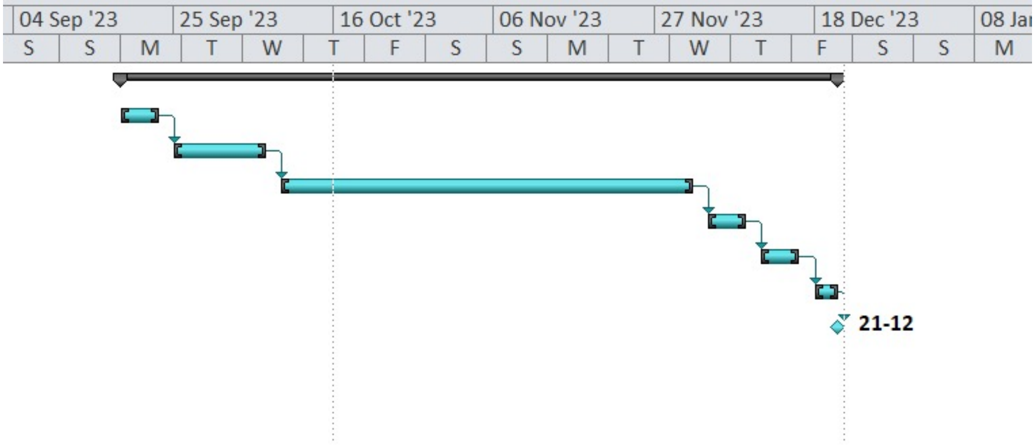


Figure 3.2: Gantt Chart

3.2 Use Case Diagrams

Creating a Use Case Diagram for a Weather Forecasting Project involves identifying the different actors and the interactions they have with the system.

- User: Interacts with the system ,Login to the website, Register on the website, view products, add product to the cart, delete product from the cart, Checkout, logout from the website

- Admin: Interacts with the system, Login to the website through admin dashboard,add new product(product image,description,price,etc.), can see the orders placed by the customers.

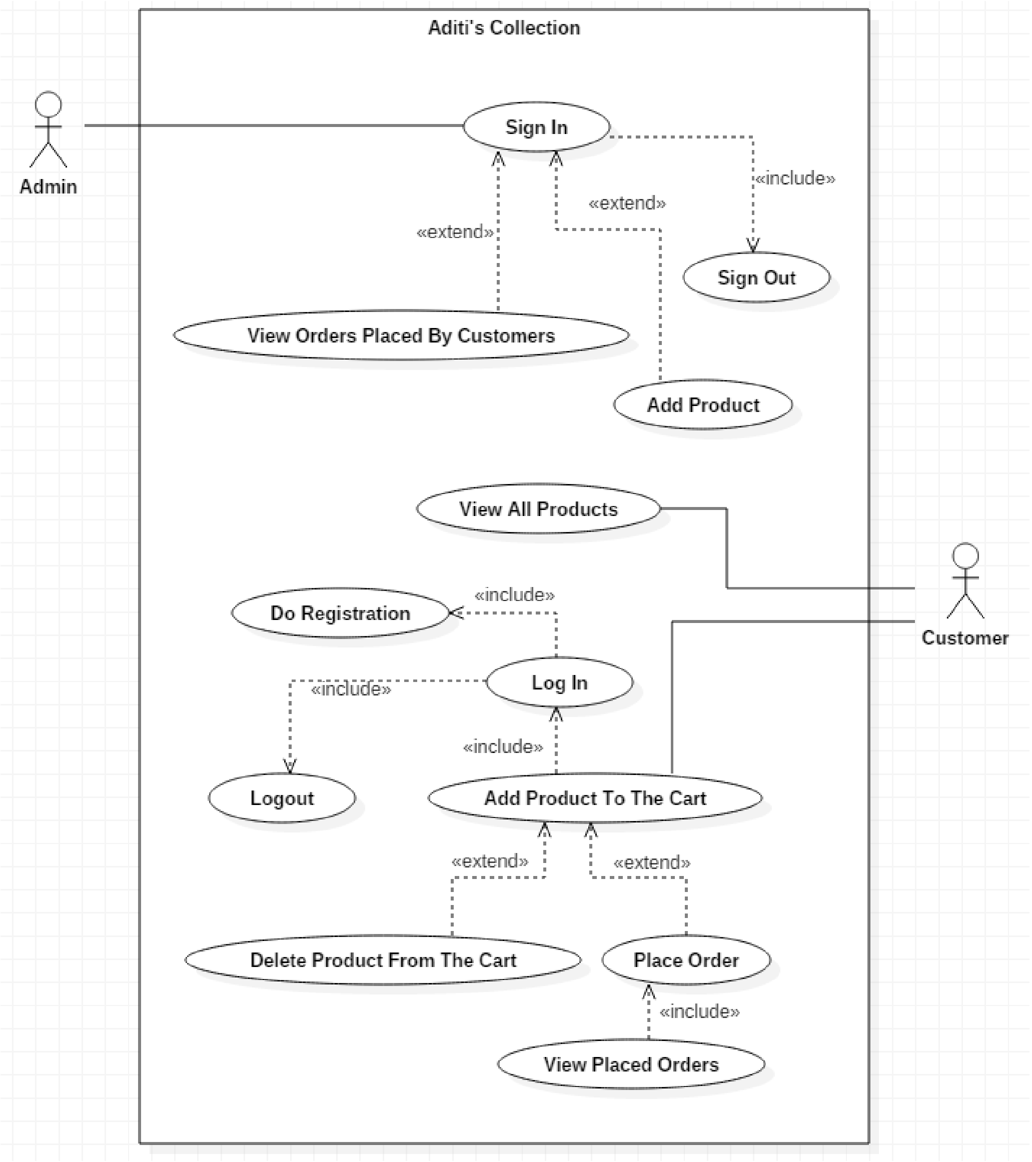


Figure 3.3: Use case

3.3 Operating tools and technology

3.3.1 Technologies Used

1. **Frontend Development:**

   React.js:

   - React.js is an open-source JavaScript library developed and maintained by Facebook.
   - It is widely used for building user interfaces, especially for single-page applications where the user interacts with the page without reloading it.
   - React.js follows a component-based architecture, making it a powerful and efficient tool for creating dynamic and interactive web applications.
   - It encourages the creation of reusable UI components, which present data that changes over time.
   - React can also render on the server using Node, and it can power native apps using React Native.
   - React implements one-way data flow which makes it easy to reason about our app.

2. **Backend Development:**

   Firebase:

   - Firebase is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit.
   - It is built on Googles infrastructure.
   - Firebase is categorized as a NoSQL database program, which stores data in JSON-like documents.
   - In Firebase, a document is a set of key-value pairs defined by a schema. A group of documents makes up a collection.

   Firestore:

   - **Firestore:** A NoSQL cloud database for efficient data storage and retrieval.
   - **Firebase Authentication:** To secure user accounts and manage access controls.

3. **Styling:**

   - **CSS3:** For styling and layout design.
   - **Styled Components:** For component-level styling.

4. **Development Tool:**

    - **Visual Studio Code:** The primary integrated development environment (IDE). Optimized for building and debugging modern web and cloud applications.

5. **Hardware Requirements:**

    - Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system.
    - As for e.g. Modem, WAN LAN, Ethernet Cross-Cable.

Chapter 4

System Design

4.1 Database Design

In Firebase Firestore, the database design is centered around collections and documents. Below is a simplified example of how you can structure your data for the ”Aditi’s Collection Website” project in Firebase Firestore:

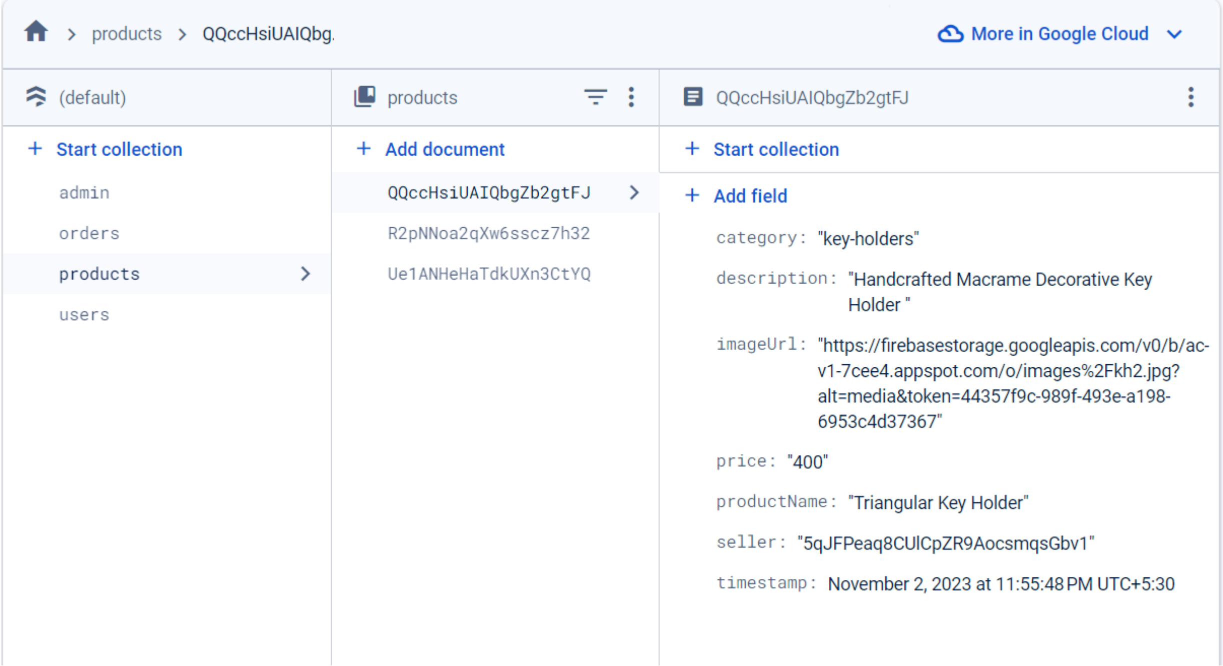
Firestore Collections:

- Products Collection:

  Code:

  Collection: products
  Document: productId
  - name: ”Product Name”
  - description: ”Product Description”
  - price: 29.99
  - category: ”Wall Hangings”
  - imageURL: ”https://example.com/product$_{i}mage.jpg$”

4.1.1 Collection Example



admin
orders
products
users

QQccHsiUAIQbgZb2gtFJ
R2pNNoa2qXw6sscz7h32
Ue1ANHeHaTdkUXn3CtYQ

category: "key-holders"

description: "Handcrafted Macrame Decorative Key Holder "

imageUrl: "https://firebasestorage.googleapis.com/v0/b/ac-v1-7cee4.appspot.com/o/images%2Fkh2.jpg?alt=media&token=44357f9c-989f-493e-a198-6953c4d37367"

price: "400"

productName: "Triangular Key Holder"

seller: "5qJFPeaq8CUlCpZR9AocsmqsGbv1"

timestamp: November 2, 2023 at 11:55:48 PM UTC+5:30

4.2 Class Diagram

The class diagram depicts a static view of an application. It represents the types of objects residing in the system and the relationships between them.
A class consists of its objects, and also it may inherit from other classes.
A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

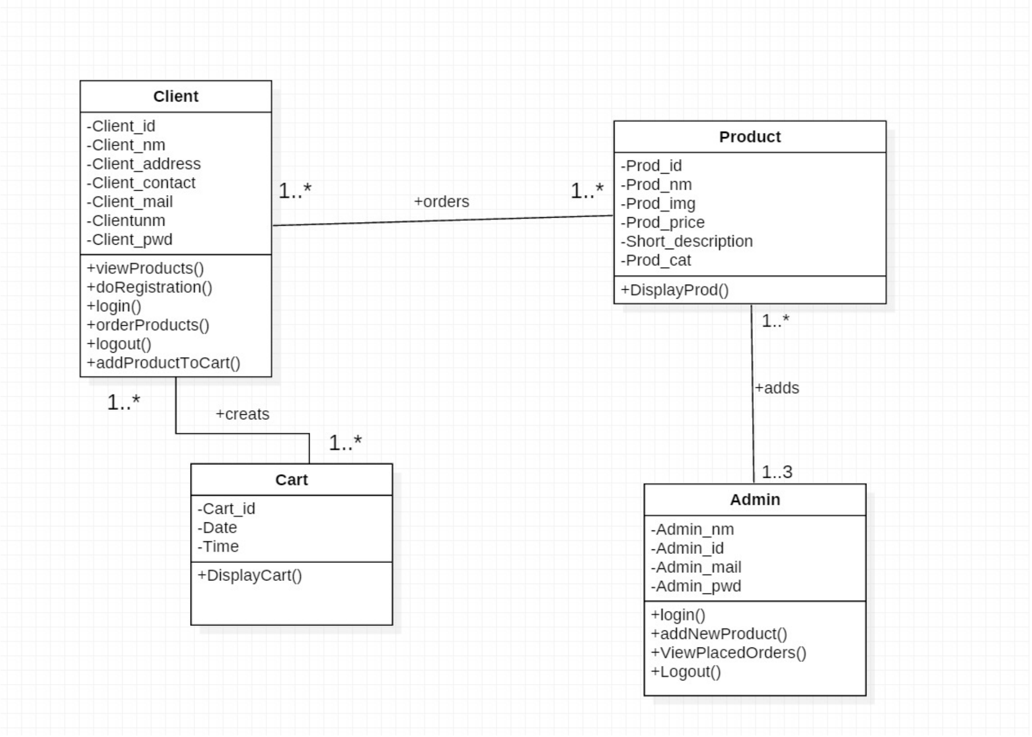


Figure 4.1: Class Diagram

4.3 Activity Diagram

The activity diagram helps in envisioning the workflow from one activity to another. It put emphasis on the condition of flow and the order in which it occurs. The flow can be sequential, branched, or concurrent, and to deal with such kinds of flows, the activity diagram has come up with a fork, join, etc.

4.3.1 Admin Activity Diagram

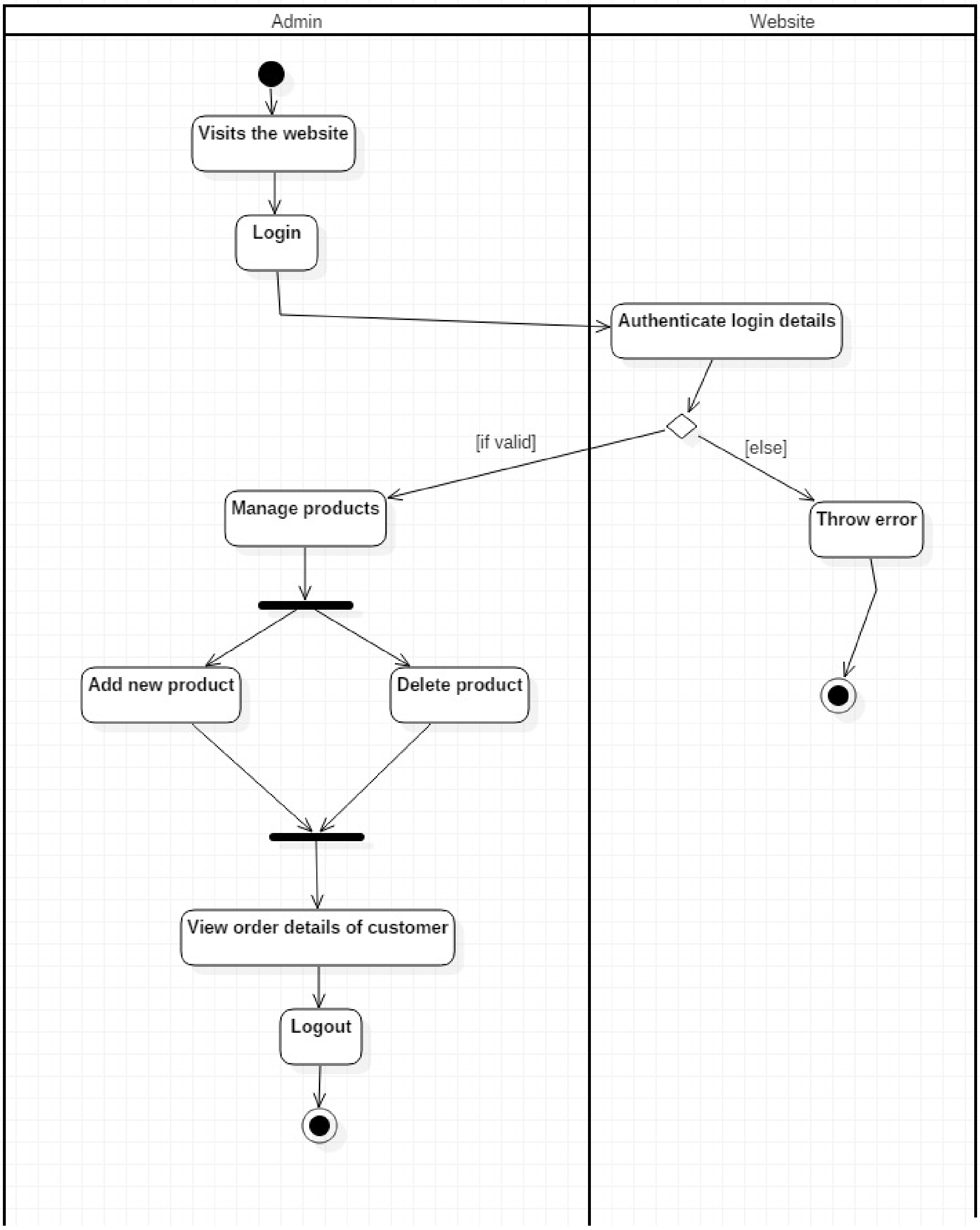


Figure 4.2: Admin Activity Diagram

4.3.2 User Activity Diagram

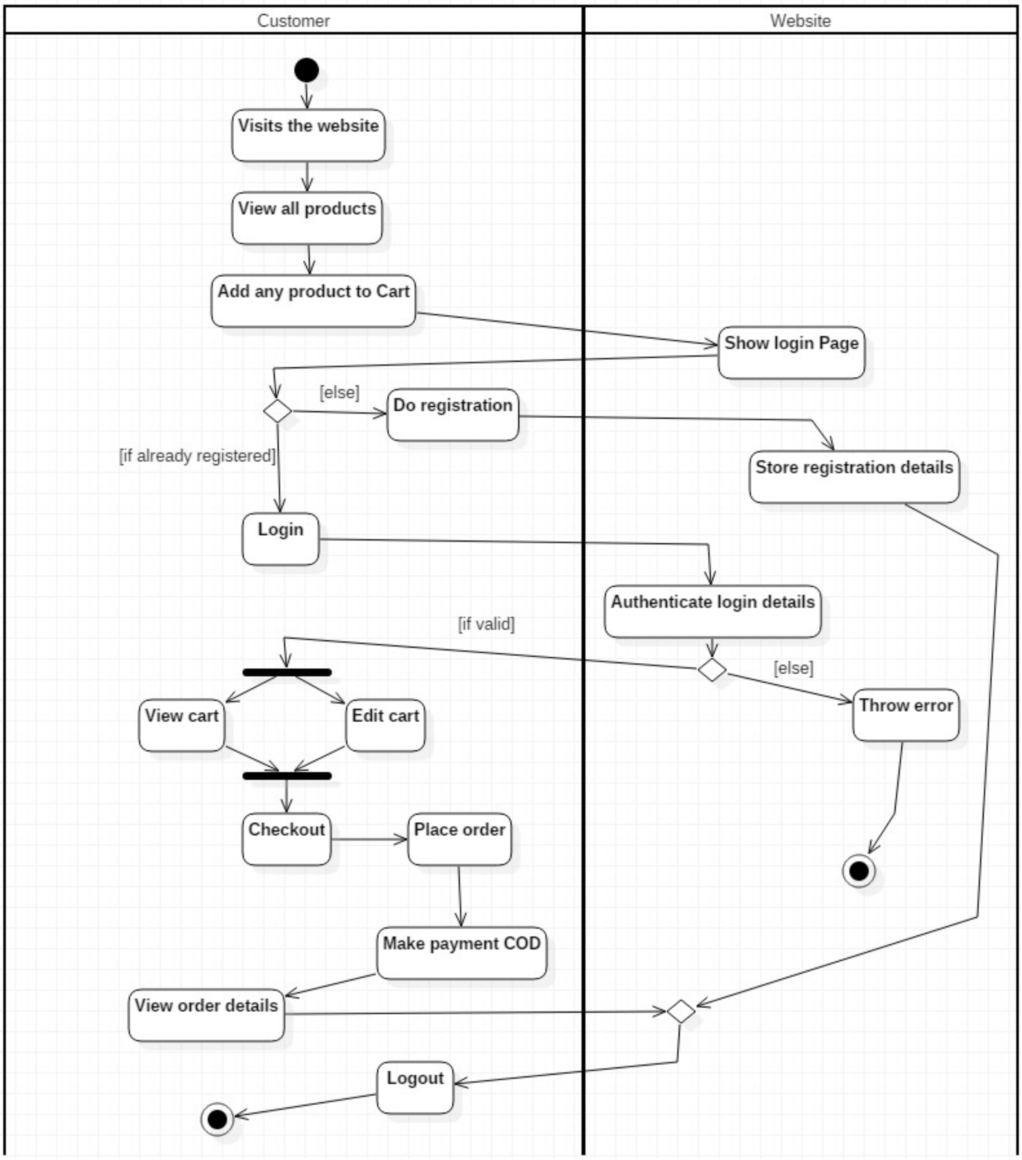
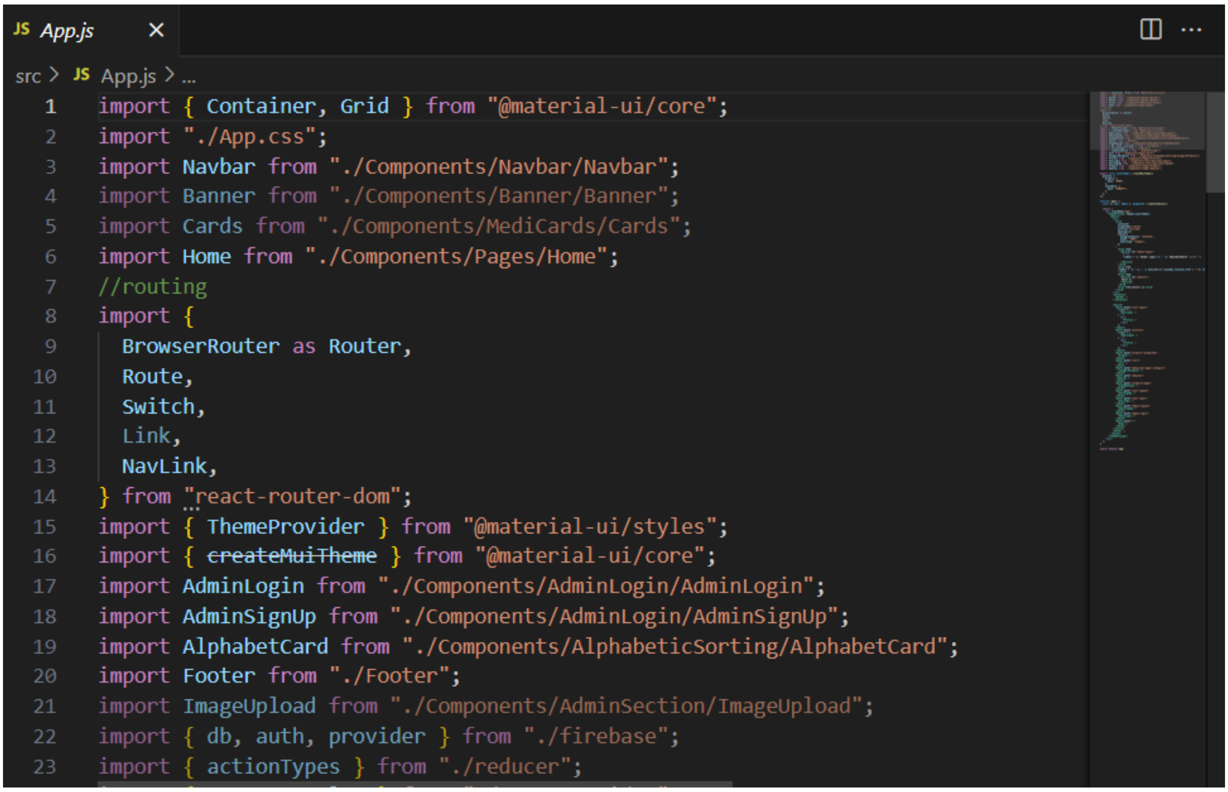


Figure 4.3: User Activity Diagram

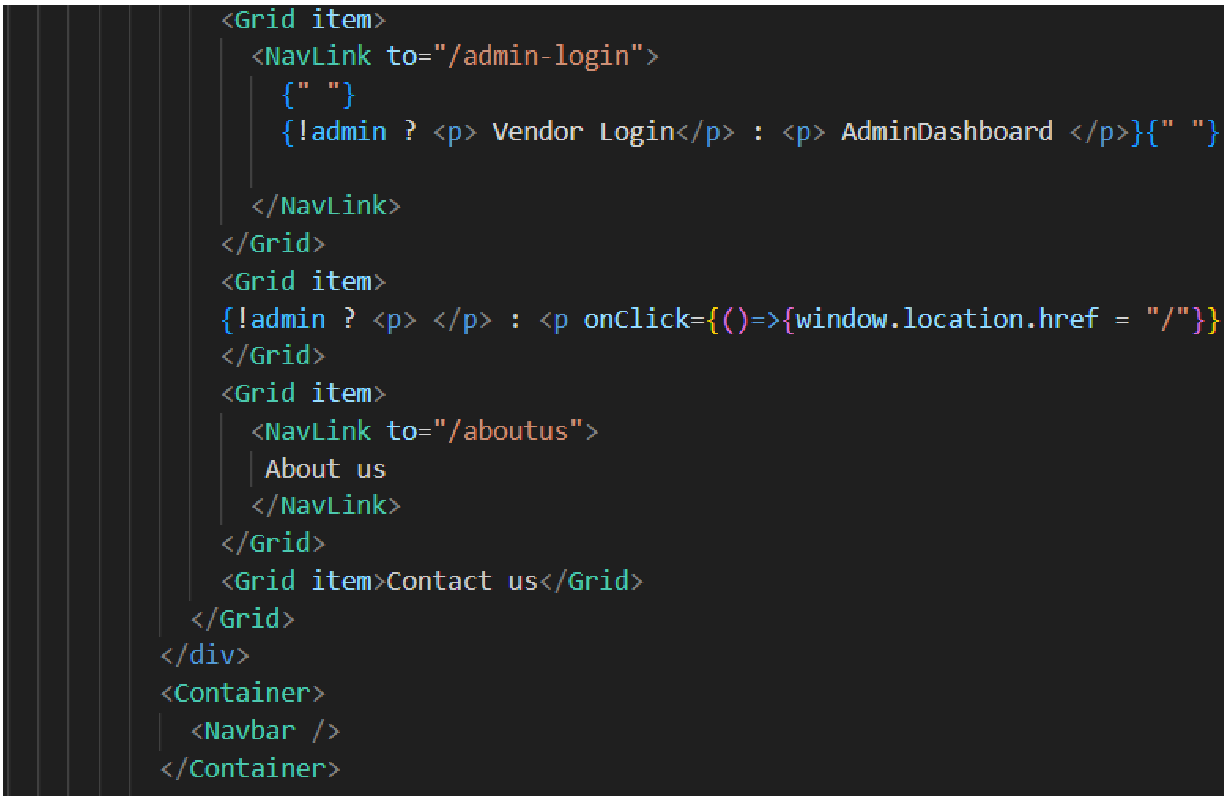
Chapter 5

Coding

5.1 App.js

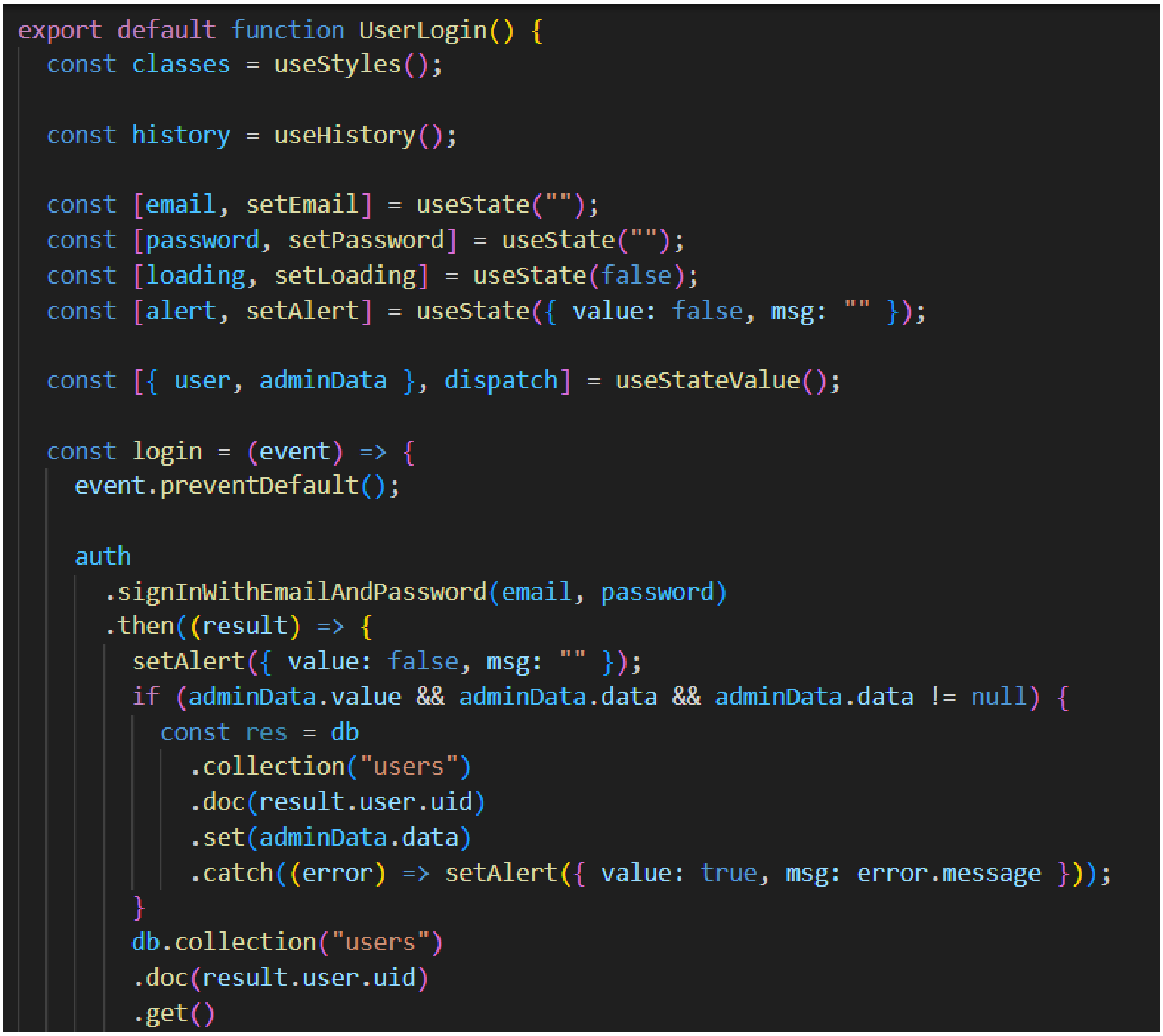


```javascript
import { Container, Grid } from "@material-ui/core";
import "./App.css";
import Navbar from "./Components/Navbar/Navbar";
import Banner from "./Components/Banner/Banner";
import Cards from "./Components/MediCards/Cards";
import Home from "./Components/Pages/Home";
//routing
import {
  BrowserRouter as Router,
  Route,
  Switch,
  Link,
  NavLink,
} from "react-router-dom";
import { ThemeProvider } from "@material-ui/styles";
import { ~~createMuiTheme~~ } from "@material-ui/core";
import AdminLogin from "./Components/AdminLogin/AdminLogin";
import AdminSignUp from "./Components/AdminLogin/AdminSignUp";
import AlphabetCard from "./Components/AlphabeticSorting/AlphabetCard";
import Footer from "./Footer";
import ImageUpload from "./Components/AdminSection/ImageUpload";
import { db, auth, provider } from "./firebase";
import { actionTypes } from "./reducer";
```



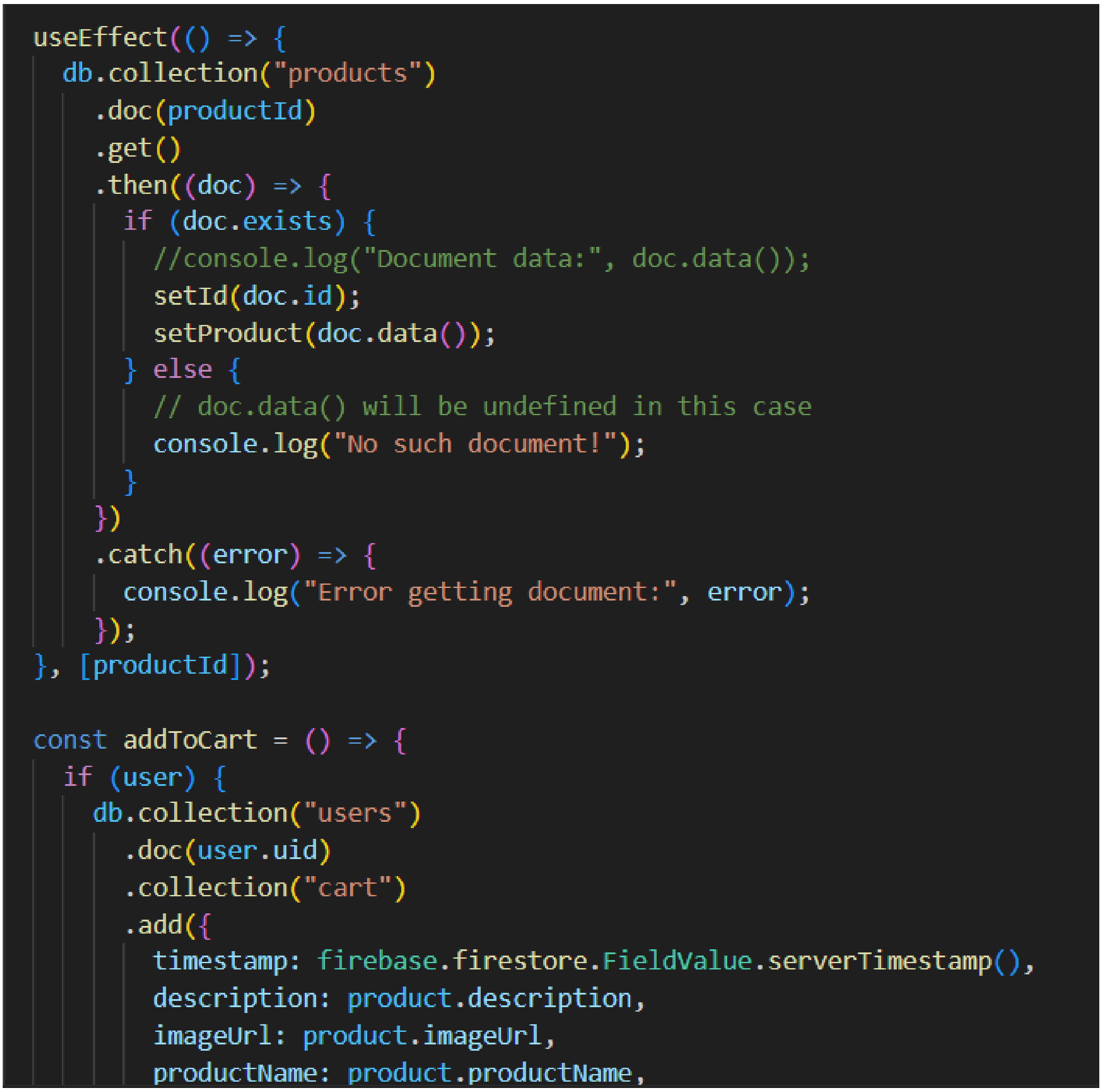
```
    <Grid item>
      <NavLink to="/admin-login">
        {" "}
        {!admin ? <p> Vendor Login</p> : <p> AdminDashboard </p>}{" "}

      </NavLink>
    </Grid>
    <Grid item>
    {!admin ? <p> </p> : <p onClick={()=>{window.location.href = "/"}}
    </Grid>
    <Grid item>
      <NavLink to="/aboutus">
       About us
      </NavLink>
    </Grid>
    <Grid item>Contact us</Grid>
  </Grid>
</div>
<Container>
  <Navbar />
</Container>
```

5.2 UserLogin.js



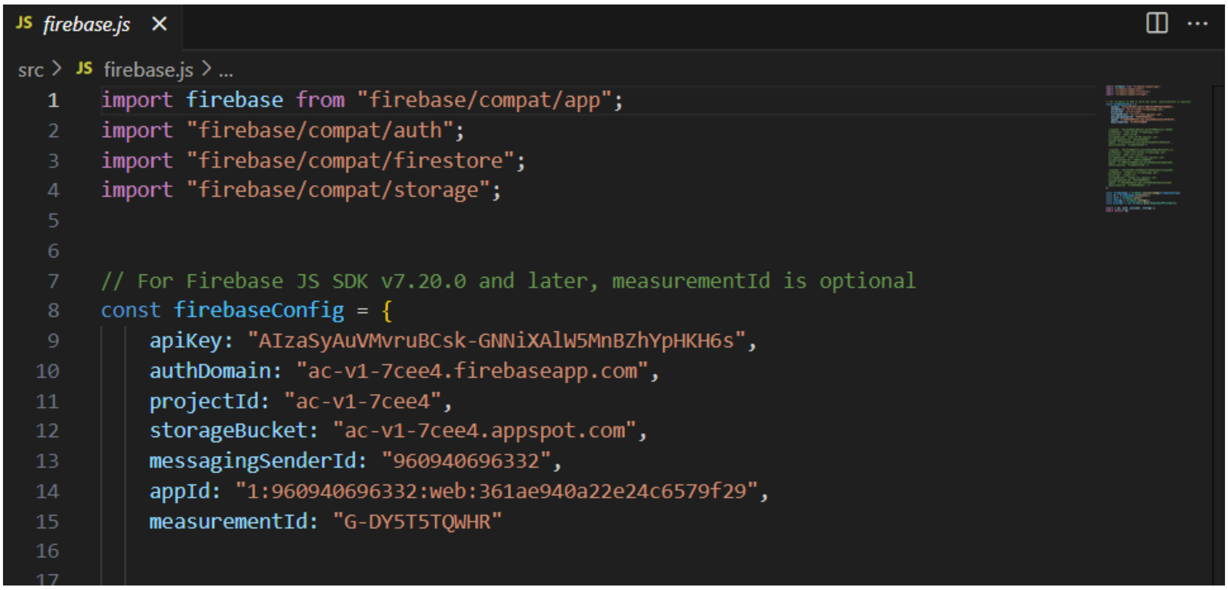
```
export default function UserLogin() {
  const classes = useStyles();

  const history = useHistory();

  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [loading, setLoading] = useState(false);
  const [alert, setAlert] = useState({ value: false, msg: "" });

  const [{ user, adminData }, dispatch] = useStateValue();

  const login = (event) => {
    event.preventDefault();

    auth
      .signInWithEmailAndPassword(email, password)
      .then((result) => {
        setAlert({ value: false, msg: "" });
        if (adminData.value && adminData.data && adminData.data != null) {
          const res = db
            .collection("users")
            .doc(result.user.uid)
            .set(adminData.data)
            .catch((error) => setAlert({ value: true, msg: error.message }));
        }
        db.collection("users")
        .doc(result.user.uid)
        .get()
```

5.3 Product.js



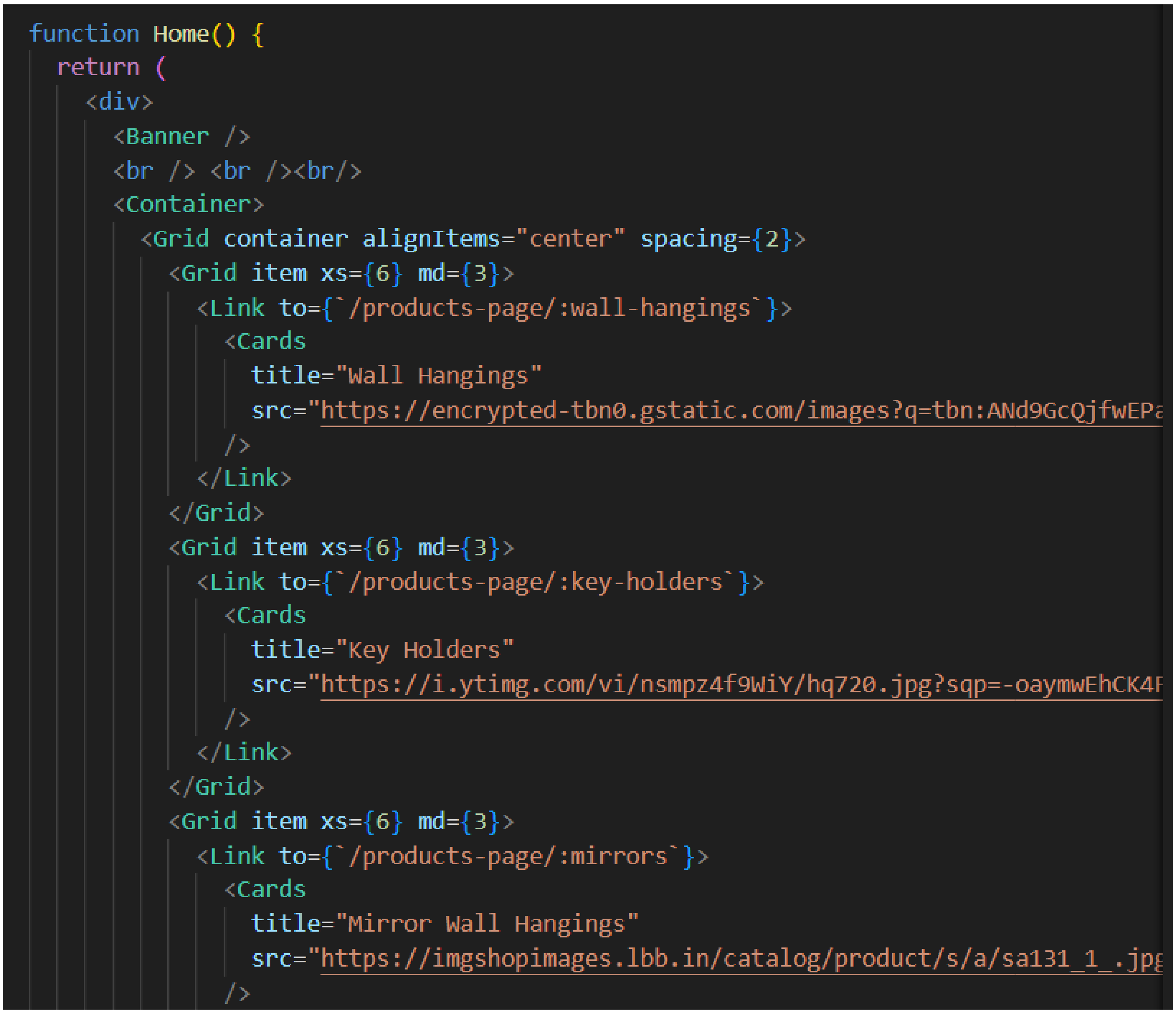
```
useEffect(() => {
  db.collection("products")
    .doc(productId)
    .get()
    .then((doc) => {
      if (doc.exists) {
        //console.log("Document data:", doc.data());
        setId(doc.id);
        setProduct(doc.data());
      } else {
        // doc.data() will be undefined in this case
        console.log("No such document!");
      }
    })
    .catch((error) => {
      console.log("Error getting document:", error);
    });
}, [productId]);

const addToCart = () => {
  if (user) {
    db.collection("users")
      .doc(user.uid)
      .collection("cart")
      .add({
        timestamp: firebase.firestore.FieldValue.serverTimestamp(),
        description: product.description,
        imageUrl: product.imageUrl,
        productName: product.productName,
```

5.4 Firebase.js



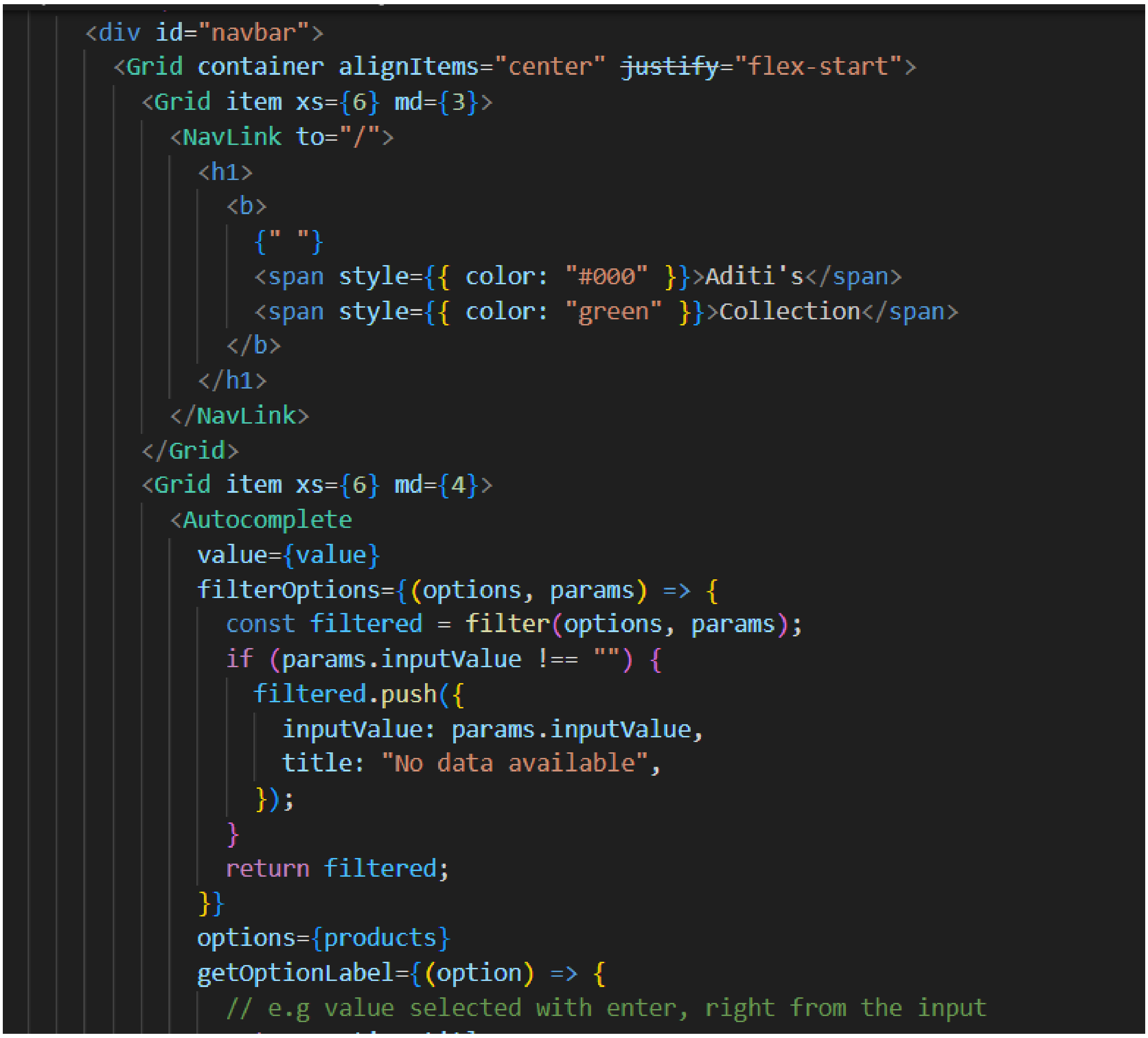
```
import firebase from "firebase/compat/app";
import "firebase/compat/auth";
import "firebase/compat/firestore";
import "firebase/compat/storage";


// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
    apiKey: "AIzaSyAuVMvruBCsk-GNNiXAlW5MnBZhYpHKH6s",
    authDomain: "ac-v1-7cee4.firebaseapp.com",
    projectId: "ac-v1-7cee4",
    storageBucket: "ac-v1-7cee4.appspot.com",
    messagingSenderId: "960940696332",
    appId: "1:960940696332:web:361ae940a22e24c6579f29",
    measurementId: "G-DY5T5TQWHR"

```

5.5 Home.js



```
function Home() {
  return (
    <div>
      <Banner />
      <br /> <br /><br/>
      <Container>
        <Grid container alignItems="center" spacing={2}>
          <Grid item xs={6} md={3}>
            <Link to={`/products-page/:wall-hangings`}>
              <Cards
                title="Wall Hangings"
                src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQjfwEPa
              />
            </Link>
          </Grid>
          <Grid item xs={6} md={3}>
            <Link to={`/products-page/:key-holders`}>
              <Cards
                title="Key Holders"
                src="https://i.ytimg.com/vi/nsmpz4f9WiY/hq720.jpg?sqp=-oaymwEhCK4F
              />
            </Link>
          </Grid>
          <Grid item xs={6} md={3}>
            <Link to={`/products-page/:mirrors`}>
              <Cards
                title="Mirror Wall Hangings"
                src="https://imgshopimages.lbb.in/catalog/product/s/a/sa131_1_.jpg
              />
```

5.6 Navbar.js

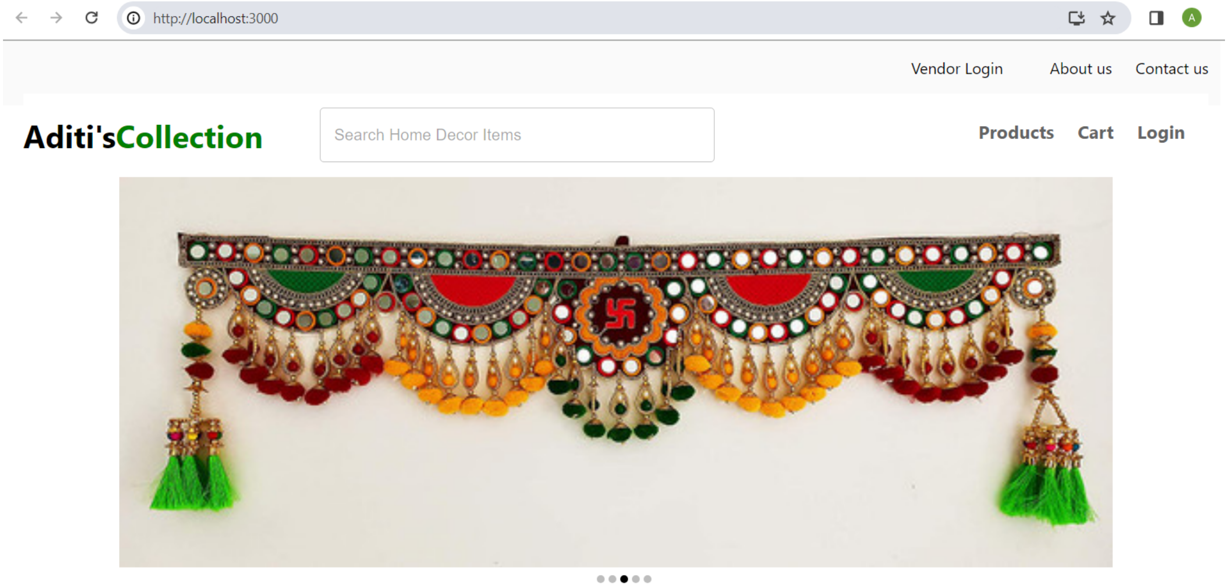


```
      <div id="navbar">
        <Grid container alignItems="center" ~~justify~~="flex-start">
          <Grid item xs={6} md={3}>
            <NavLink to="/">
              <h1>
                <b>
                  {" "}
                  <span style={{ color: "#000" }}>Aditi's</span>
                  <span style={{ color: "green" }}>Collection</span>
                </b>
              </h1>
            </NavLink>
          </Grid>
          <Grid item xs={6} md={4}>
            <Autocomplete
              value={value}
              filterOptions={(options, params) => {
                const filtered = filter(options, params);
                if (params.inputValue !== "") {
                  filtered.push({
                    inputValue: params.inputValue,
                    title: "No data available",
                  });
                }
                return filtered;
              }}
              options={products}
              getOptionLabel={(option) => {
                // e.g value selected with enter, right from the input
```

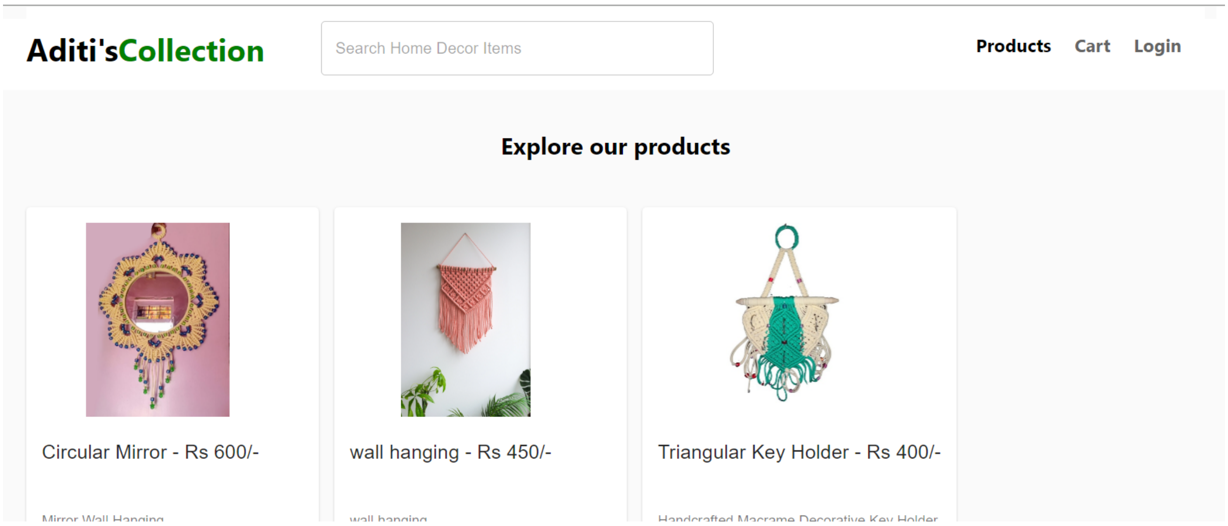
Chapter 6

System Implementation

6.1 Home Page



6.2 Product Page



Explore our products

Circular Mirror - Rs 600/-

Mirror Wall Hanging

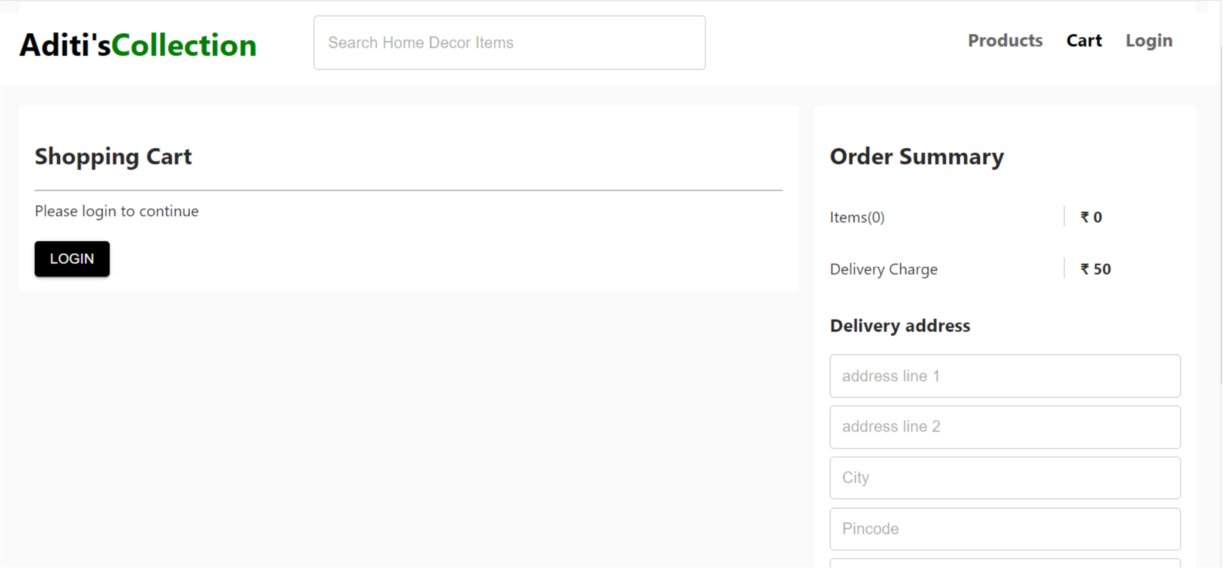
wall hanging - Rs 450/-

wall hanging

Triangular Key Holder - Rs 400/-

Handcrafted Macrame Decorative Key Holder

6.3 Cart Page



Shopping Cart

Please login to continue

Order Summary

Items(0)	₹ 0
Delivery Charge	₹ 50

Delivery address

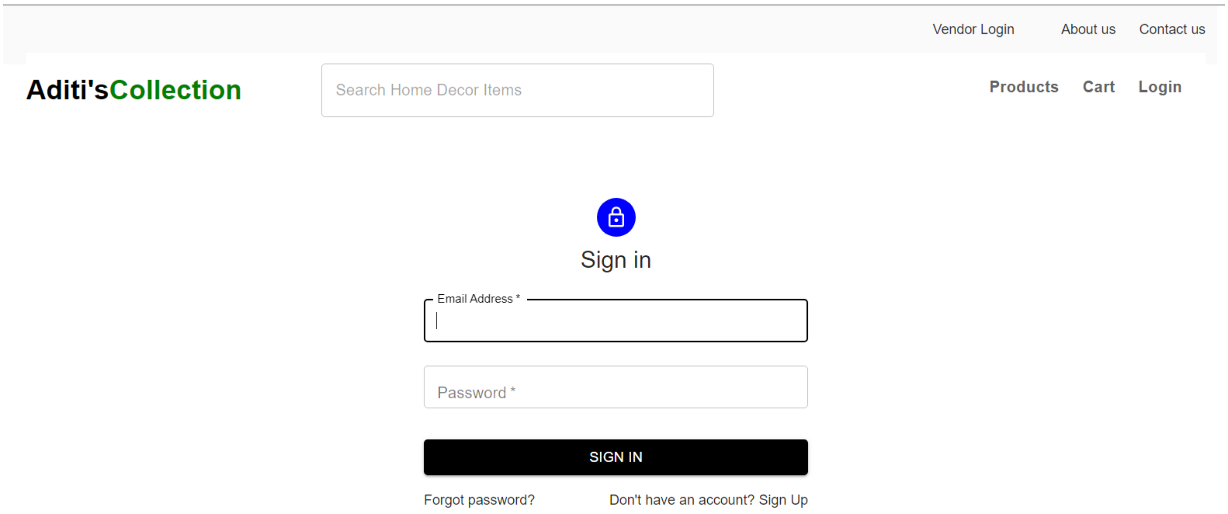
address line 1

address line 2

City

Pincode

6.4 Login Page

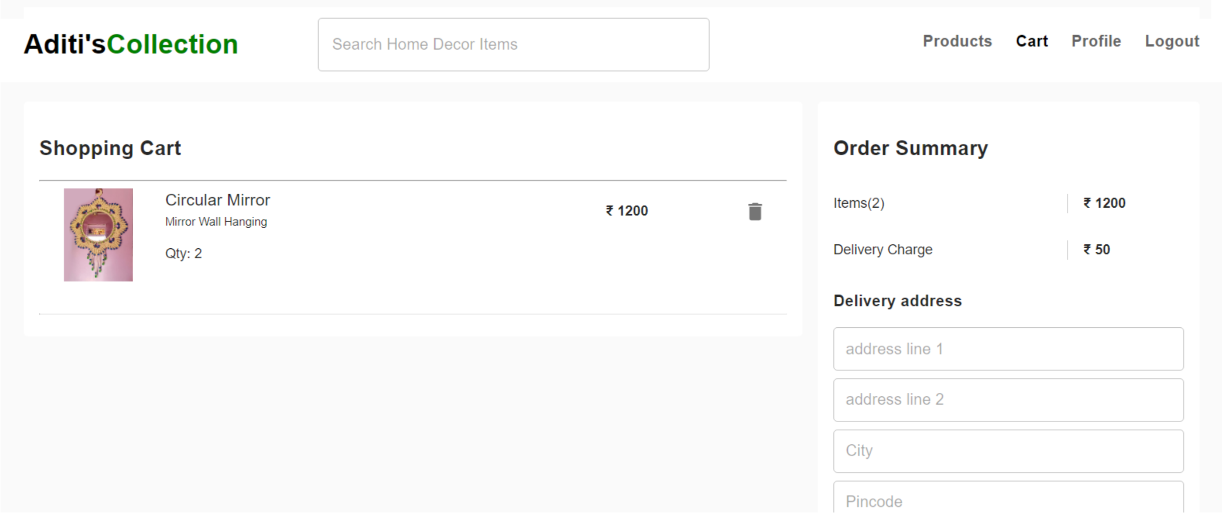


Sign in

Email Address *

Password *

6.5 My Cart



Aditi'sCollection

Search Home Decor Items

Products Cart Profile Logout

Shopping Cart

Circular Mirror
Mirror Wall Hanging
Qty: 2

₹ 1200

Order Summary

Items(2)	₹ 1200
Delivery Charge	₹ 50

Delivery address

address line 1

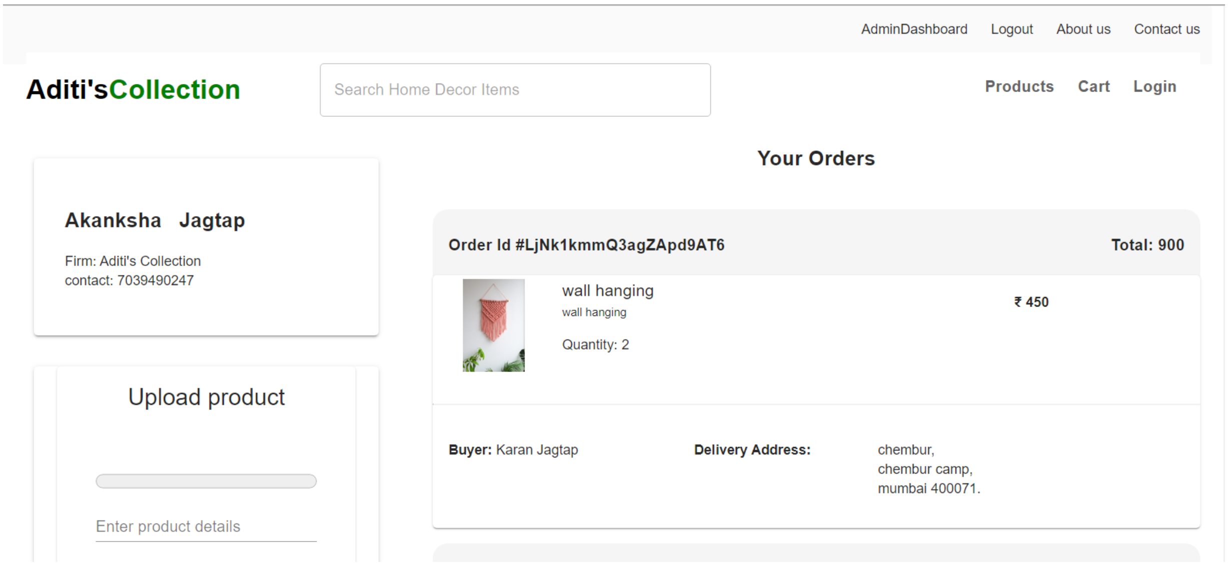
address line 2

City

Pincode

6.6 Admin Dashboard

6.6.1 Orders



AdminDashboard Logout About us Contact us

Aditi'sCollection

Search Home Decor Items

Products Cart Login

Your Orders

Akanksha Jagtap

Firm: Aditi's Collection
contact: 7039490247

Upload product

Enter product details

Order Id #LjNk1kmmQ3agZApd9AT6 **Total: 900**

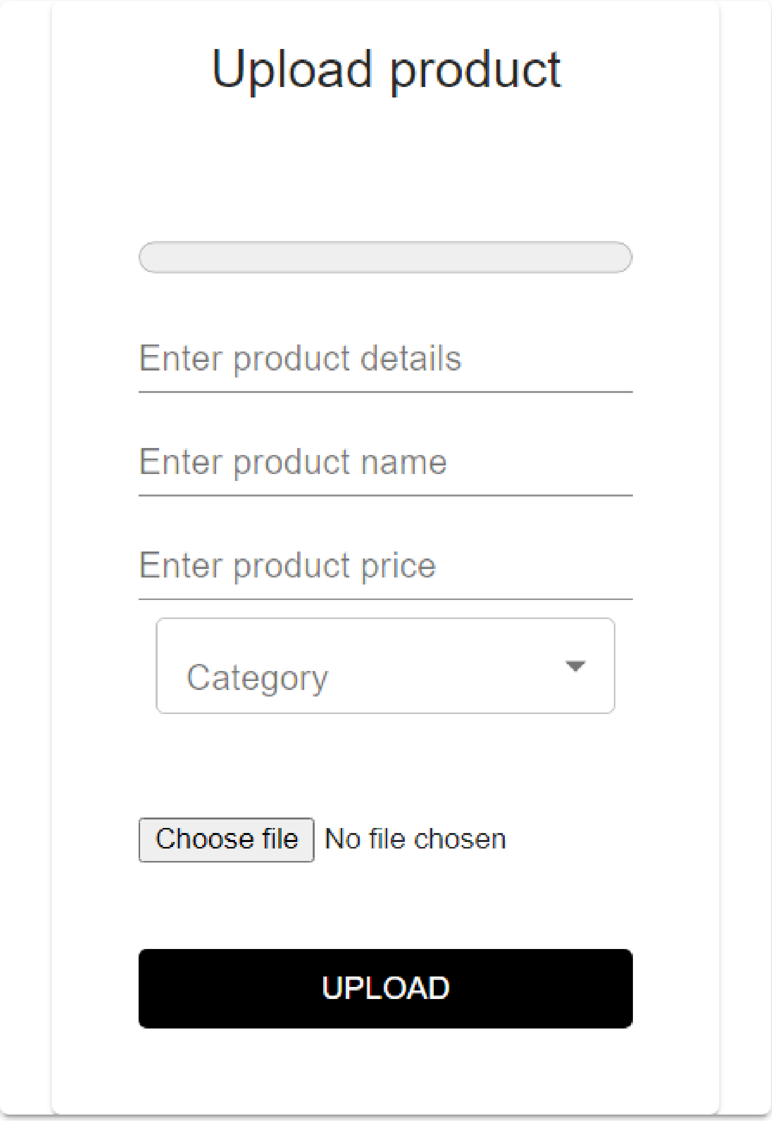
wall hanging
wall hanging
Quantity: 2

₹ 450

Buyer: Karan Jagtap

Delivery Address: chembur,
chembur camp,
mumbai 400071.

6.6.2 Image Upload



Upload product

Enter product details

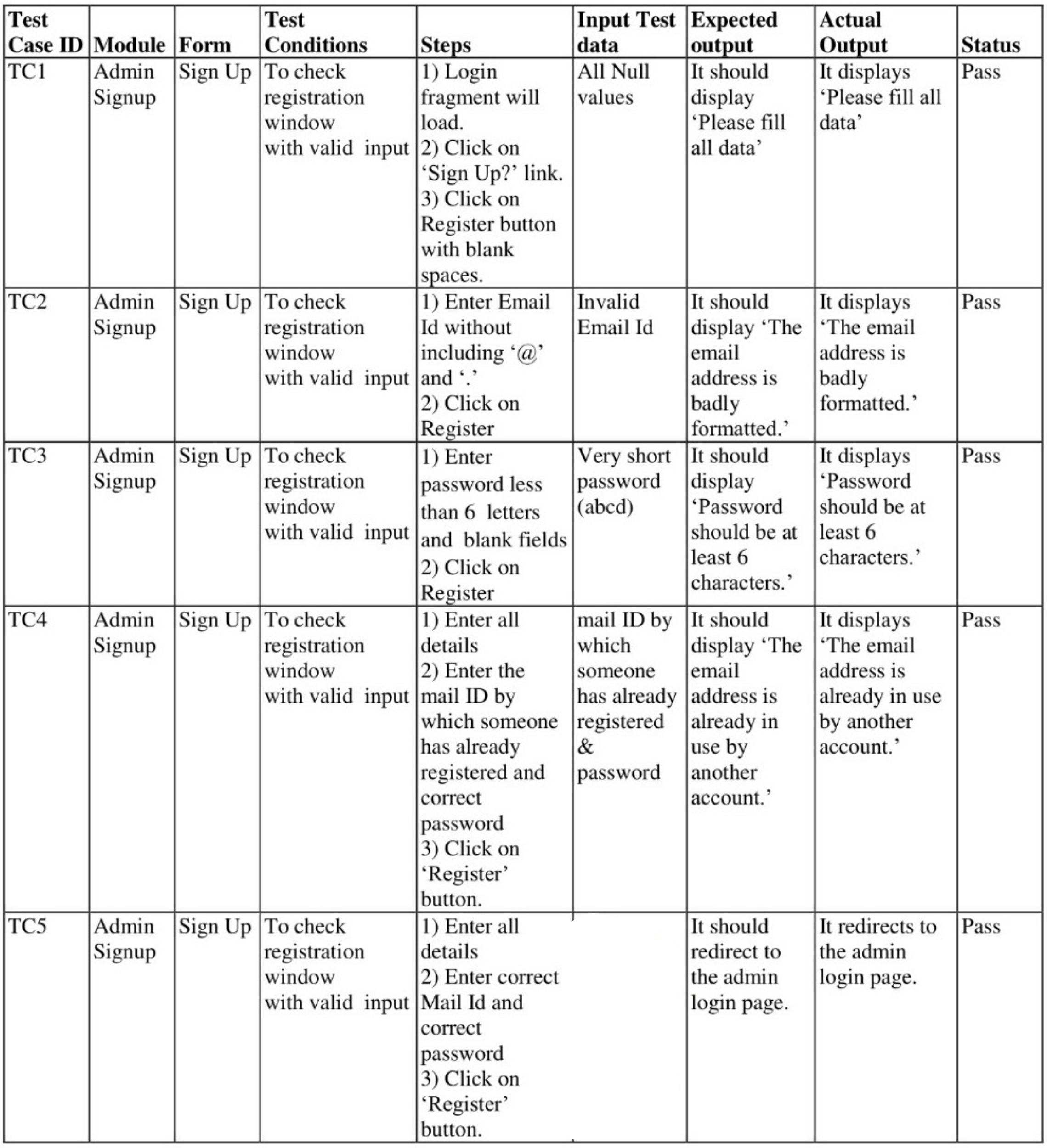
Enter product name

Enter product price

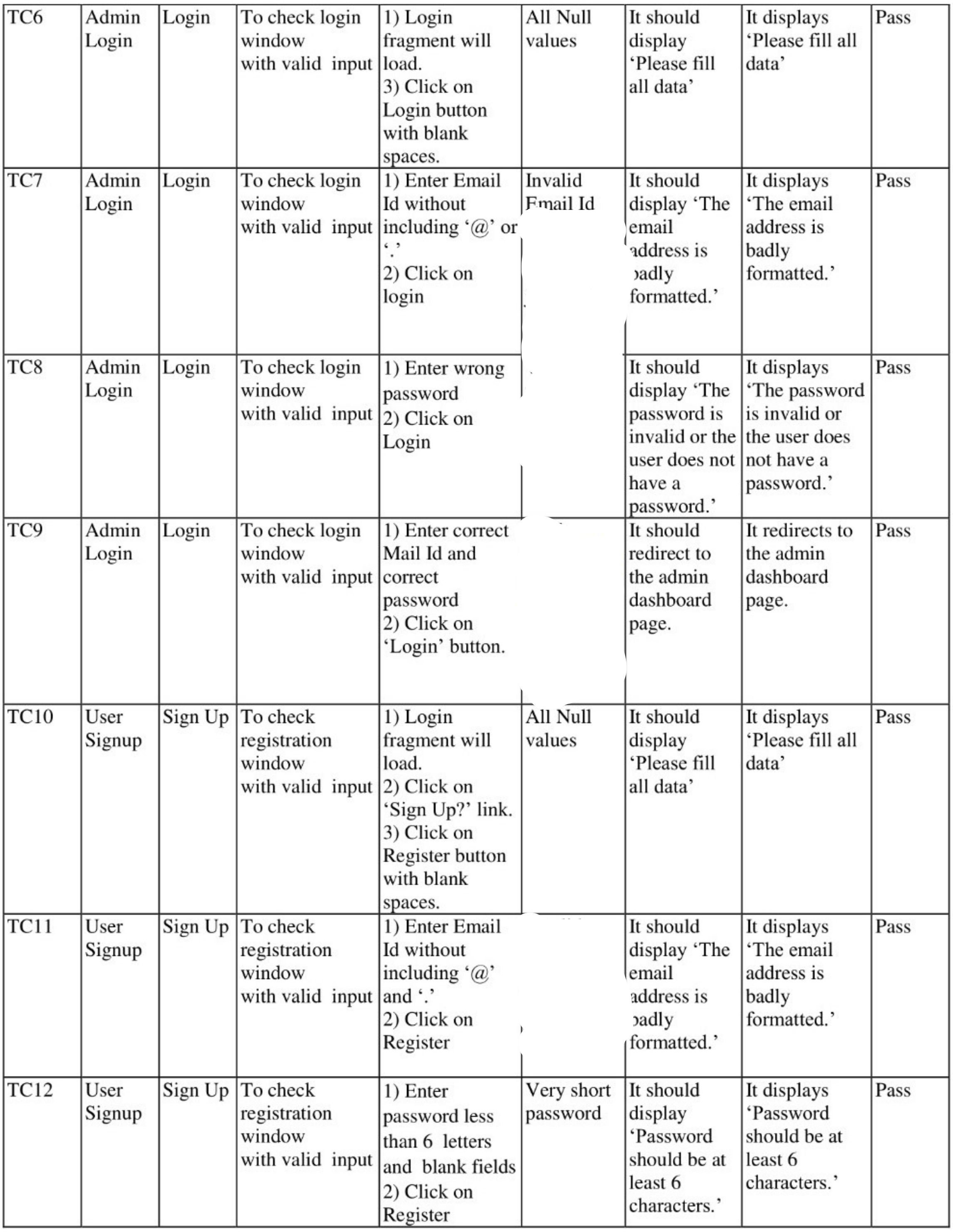
Chapter 7

System Testing

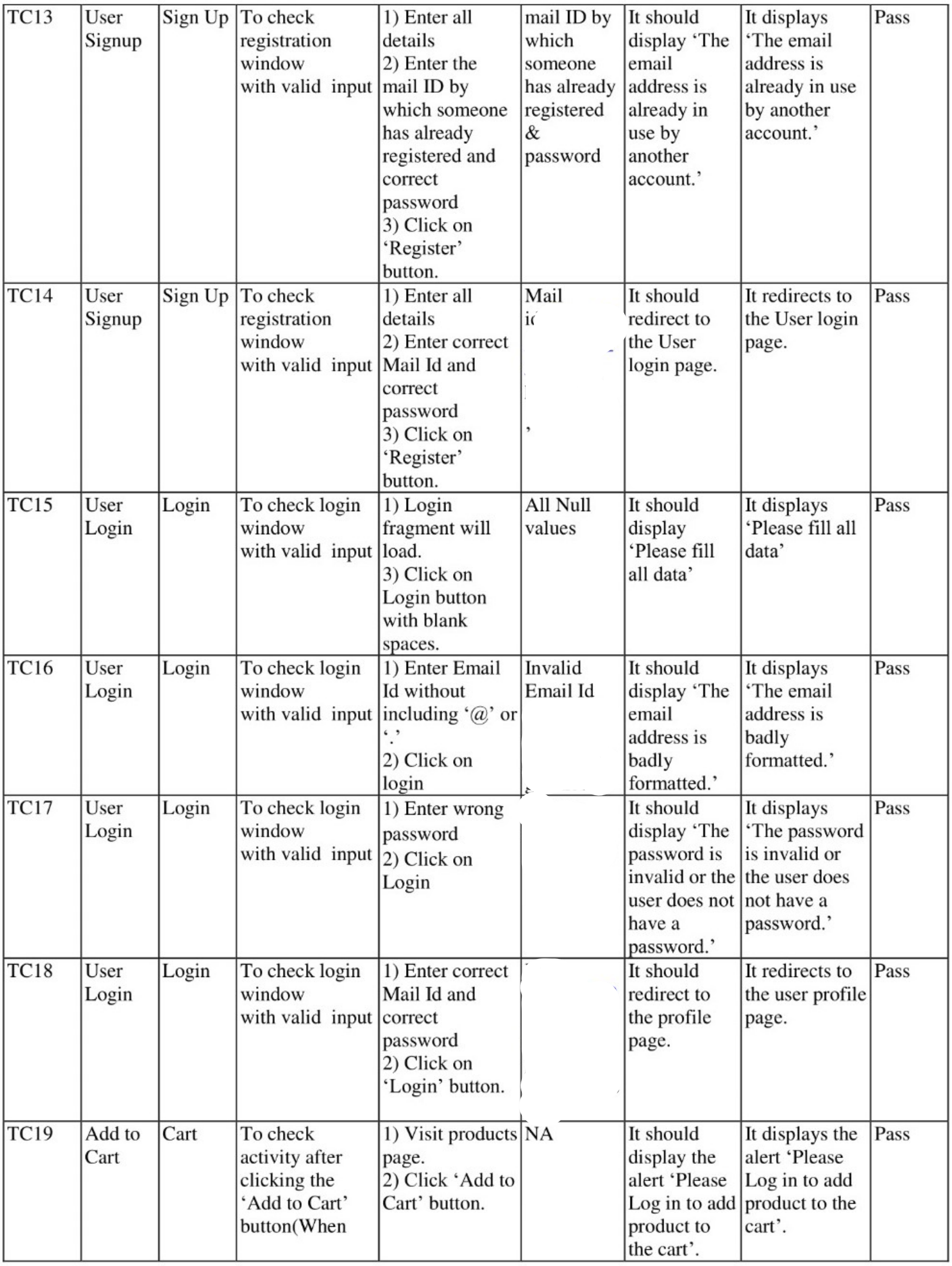
7.1 Test Cases



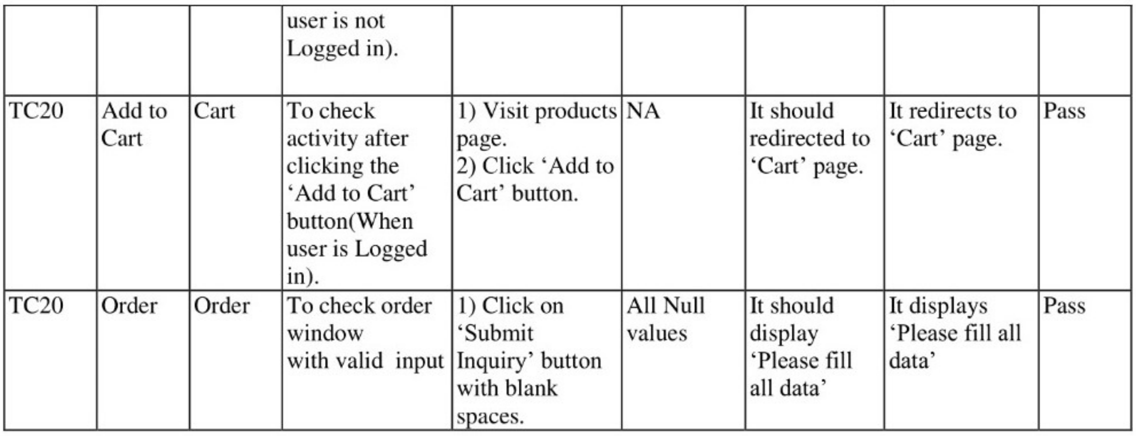
Test Case ID	Module	Form	Test Conditions	Steps	Input Test data	Expected output	Actual Output	Status
TC1	Admin Signup	Sign Up	To check registration window with valid input	1) Login fragment will load. 2) Click on ‘Sign Up?’ link. 3) Click on Register button with blank spaces.	All Null values	It should display ‘Please fill all data’	It displays ‘Please fill all data’	Pass
TC2	Admin Signup	Sign Up	To check registration window with valid input	1) Enter Email Id without including ‘@’ and ‘.’ 2) Click on Register	Invalid Email Id	It should display ‘The email address is badly formatted.’	It displays ‘The email address is badly formatted.’	Pass
TC3	Admin Signup	Sign Up	To check registration window with valid input	1) Enter password less than 6 letters and blank fields 2) Click on Register	Very short password (abcd)	It should display ‘Password should be at least 6 characters.’	It displays ‘Password should be at least 6 characters.’	Pass
TC4	Admin Signup	Sign Up	To check registration window with valid input	1) Enter all details 2) Enter the mail ID by which someone has already registered and correct password 3) Click on ‘Register’ button.	mail ID by which someone has already registered & password	It should display ‘The email address is already in use by another account.’	It displays ‘The email address is already in use by another account.’	Pass
TC5	Admin Signup	Sign Up	To check registration window with valid input	1) Enter all details 2) Enter correct Mail Id and correct password 3) Click on ‘Register’ button.		It should redirect to the admin login page.	It redirects to the admin login page.	Pass



TC6	Admin Login	Login	To check login window with valid input	1) Login fragment will load. 3) Click on Login button with blank spaces.	All Null values	It should display 'Please fill all data'	It displays 'Please fill all data'	Pass
TC7	Admin Login	Login	To check login window with valid input	1) Enter Email Id without including '@' or '.' 2) Click on login	Invalid Email Id	It should display 'The email address is badly formatted.'	It displays 'The email address is badly formatted.'	Pass
TC8	Admin Login	Login	To check login window with valid input	1) Enter wrong password 2) Click on Login		It should display 'The password is invalid or the user does not have a password.'	It displays 'The password is invalid or the user does not have a password.'	Pass
TC9	Admin Login	Login	To check login window with valid input	1) Enter correct Mail Id and correct password 2) Click on 'Login' button.		It should redirect to the admin dashboard page.	It redirects to the admin dashboard page.	Pass
TC10	User Signup	Sign Up	To check registration window with valid input	1) Login fragment will load. 2) Click on 'Sign Up?' link. 3) Click on Register button with blank spaces.	All Null values	It should display 'Please fill all data'	It displays 'Please fill all data'	Pass
TC11	User Signup	Sign Up	To check registration window with valid input	1) Enter Email Id without including '@' and '.' 2) Click on Register		It should display 'The email address is badly formatted.'	It displays 'The email address is badly formatted.'	Pass
TC12	User Signup	Sign Up	To check registration window with valid input	1) Enter password less than 6 letters and blank fields 2) Click on Register	Very short password	It should display 'Password should be at least 6 characters.'	It displays 'Password should be at least 6 characters.'	Pass



TC13	User Signup	Sign Up	To check registration window with valid input	1) Enter all details 2) Enter the mail ID by which someone has already registered and correct password 3) Click on 'Register' button.	mail ID by which someone has already registered & password	It should display 'The email address is already in use by another account.'	It displays 'The email address is already in use by another account.'	Pass
TC14	User Signup	Sign Up	To check registration window with valid input	1) Enter all details 2) Enter correct Mail Id and correct password 3) Click on 'Register' button.	Mail i(	It should redirect to the User login page.	It redirects to the User login page.	Pass
TC15	User Login	Login	To check login window with valid input	1) Login fragment will load. 3) Click on Login button with blank spaces.	All Null values	It should display 'Please fill all data'	It displays 'Please fill all data'	Pass
TC16	User Login	Login	To check login window with valid input	1) Enter Email Id without including '@' or '.' 2) Click on login	Invalid Email Id	It should display 'The email address is badly formatted.'	It displays 'The email address is badly formatted.'	Pass
TC17	User Login	Login	To check login window with valid input	1) Enter wrong password 2) Click on Login		It should display 'The password is invalid or the user does not have a password.'	It displays 'The password is invalid or the user does not have a password.'	Pass
TC18	User Login	Login	To check login window with valid input	1) Enter correct Mail Id and correct password 2) Click on 'Login' button.		It should redirect to the profile page.	It redirects to the user profile page.	Pass
TC19	Add to Cart	Cart	To check activity after clicking the 'Add to Cart' button(When	1) Visit products page. 2) Click 'Add to Cart' button.	NA	It should display the alert 'Please Log in to add product to the cart'.	It displays the alert 'Please Log in to add product to the cart'.	Pass



			user is not Logged in).					
TC20	Add to Cart	Cart	To check activity after clicking the ‘Add to Cart’ button(When user is Logged in).	1) Visit products page. 2) Click ‘Add to Cart’ button.	NA	It should redirected to ‘Cart’ page.	It redirects to ‘Cart’ page.	Pass
TC20	Order	Order	To check order window with valid input	1) Click on ‘Submit Inquiry’ button with blank spaces.	All Null values	It should display ‘Please fill all data’	It displays ‘Please fill all data’	Pass

Chapter 8

Limitation and Future Enhancement

8.1 Limitations

While the ”Aditi’s Collection Website” project has achieved notable success in transforming the business into a digital entity, it’s essential to acknowledge certain limitations that may impact its functionality and user experience.

- The current system only supports Cash on Delivery as a payment method.
- The project does not incorporate comprehensive online marketing features.
- While efforts have been made to ensure responsiveness, compatibility across all devices and browsers may vary.
- The absence of a dedicated customer feedback section limits the platform’s ability to collect and showcase customer reviews and ratings.

8.2 Future Enhancement of the Project

To address mentioned limitations and further elevate the ”Aditi’s Collection Website” project, the following future enhancements are listed below:

- Implement secure online payment gateways (e.g., credit/debit cards, digital wallets) to provide users with a wider range of payment choices.
- Implement a dedicated customer feedback section where users can leave reviews, ratings, and testimonials.
- Develop a system for promoting discounts, running special offers, and implementing a loyalty program to incentivize customer retention.

References

[1] https://blog.logrocket.com/user-authentication-firebase-react-apps/

[2] https://www.geeksforgeeks.org/how-to-implement-email-registration-and-login-using-firebase-in-react/

[3] https://mdbootstrap.com/docs/react/advanced/carousel/

[4] https://codedamn.com/news/reactjs/react-firebase-complete-tutorial

[5] https://www.makeuseof.com/react-app-firebase-connect/