CS5054NI Advanced Programming and Technologies

50% Group Coursework

2023 Spring

Group Name:			
SN	**Student Name**	**College ID**	**University ID**
1.	**Himanshu Yadav**	**NP01CP4A210166**	**21049513**
2.	**Anju Kumari Yadav**	**NP01CP4S220194**	**22015649**
3.	**Dipana Sharma**	**NP01CP4A210088**	**21039875**
4.	**Sunayana Shrestha**	**NP01CP4A210074**	**21040037**

Assignment Due Date: Monday, May 8, 2023

Assignment Submission Date: Monday, May 8, 2023

Word Count: 2200

Project File Links:

Google Drive Link:	https://drive.google.com/file/d/16ffAqnDxNsnv2hFyL1tdrcDPMI-_SICC/view?usp=share_link

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of Contents

Table of Tables

Table of Tables

1. Introduction

E-commerce websites have become an essential part of our everyday lives in the current technological era. People may now purchase products and services without leaving their houses thanks to online shopping. The development of a clothes e-commerce website is the main goal of this group coursework. Model, View, and Controller are the three components that make up the MVC-style website. The project's user-friendly layout, simple navigation, and effective search and filter tools are all designed to provide users a smooth purchasing experience. Additionally, the website needs to have a safe login process, an admin area, and a system for validation and handling exceptions.

1.1 Aim

The project's aim is to create an e-commerce website (Clothes Website) that follows the MVC pattern, allowing users to register, login, browse products, and make purchases, as well as providing an admin panel for managing products and orders, while ensuring security and dependability through validation and exception handling.

1.2 Objectives

a. Develop website with MVC architecture, Servlets, Model methods, and View files.
b. Implement secure login system with password encryption and user registration (including image upload).
c. Create admin panel for product database management and order tracking.
d. Design homepage with product display, search, and filters.
e. Include user-specific features like profile editing, password change, and product viewing.
f. Ensure website reliability and security through validation and exception handling.

2. User Interface Design

The user interface is the application's front-end view with which the user interacts to use the software. User interface is a component of software and is intended to provide the user with insight into the software. The user interface provides the foundation for human-computer interaction. (Tutorialspoint.com, 2023)

2.1 Wireframe

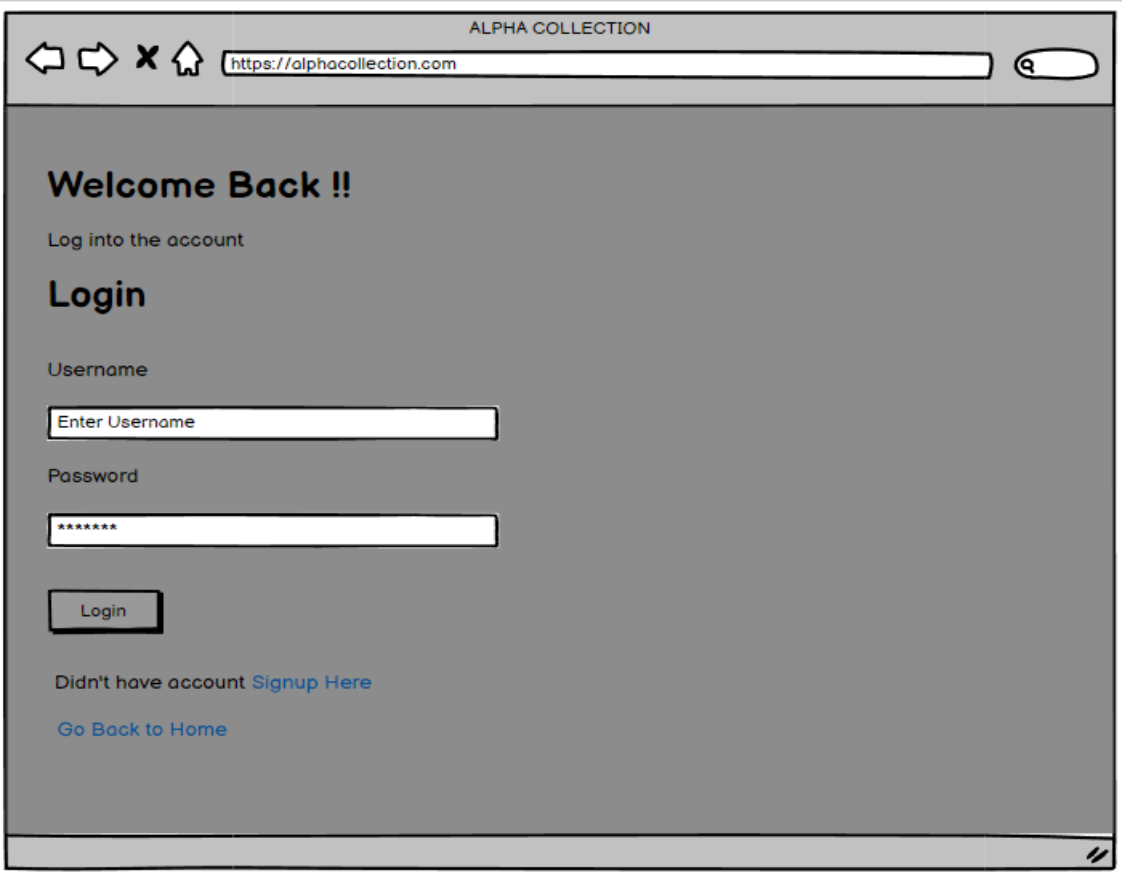

Figure 1: Wireframe of Login page

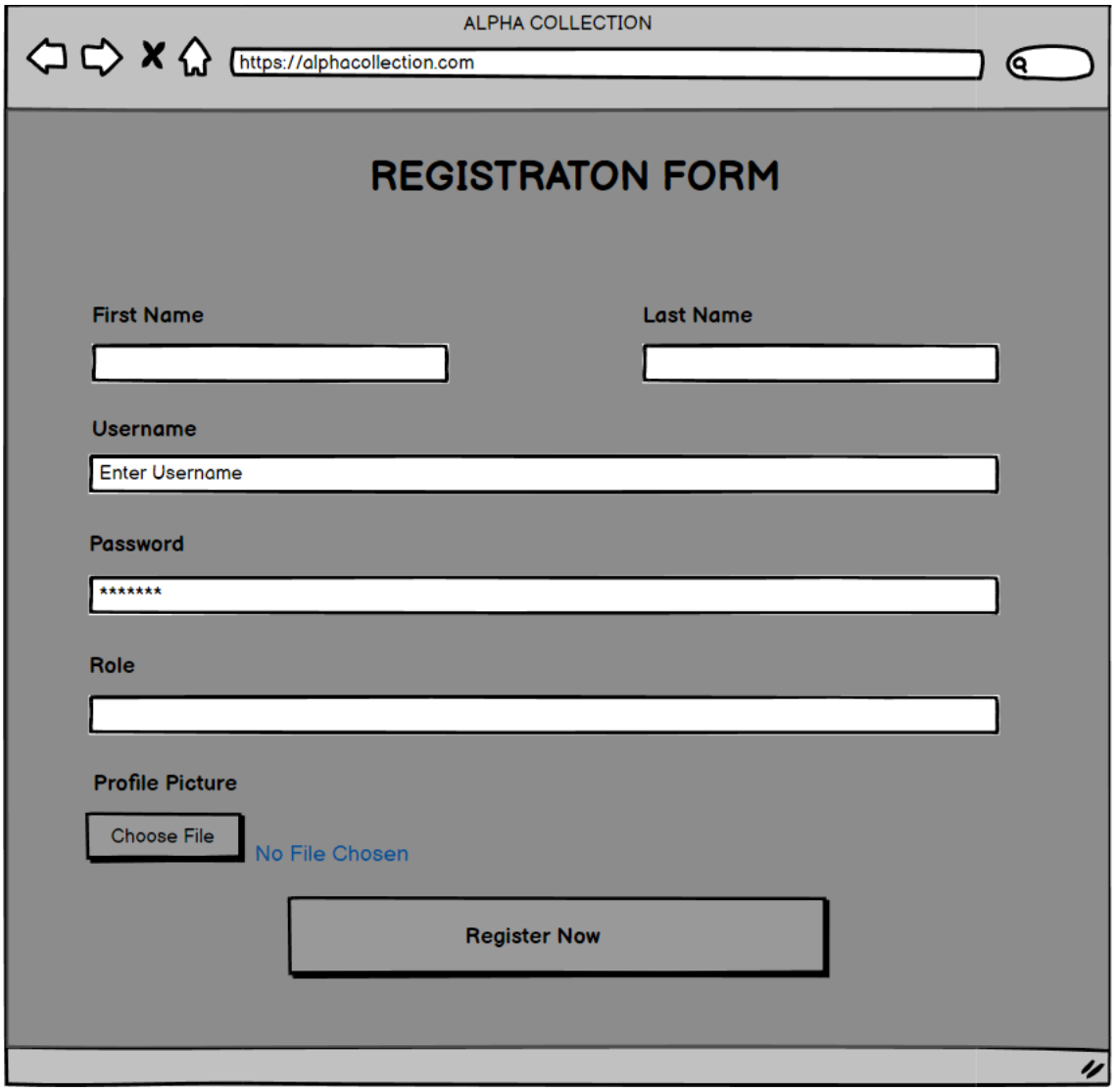

Figure 2: Wireframe of Registration form

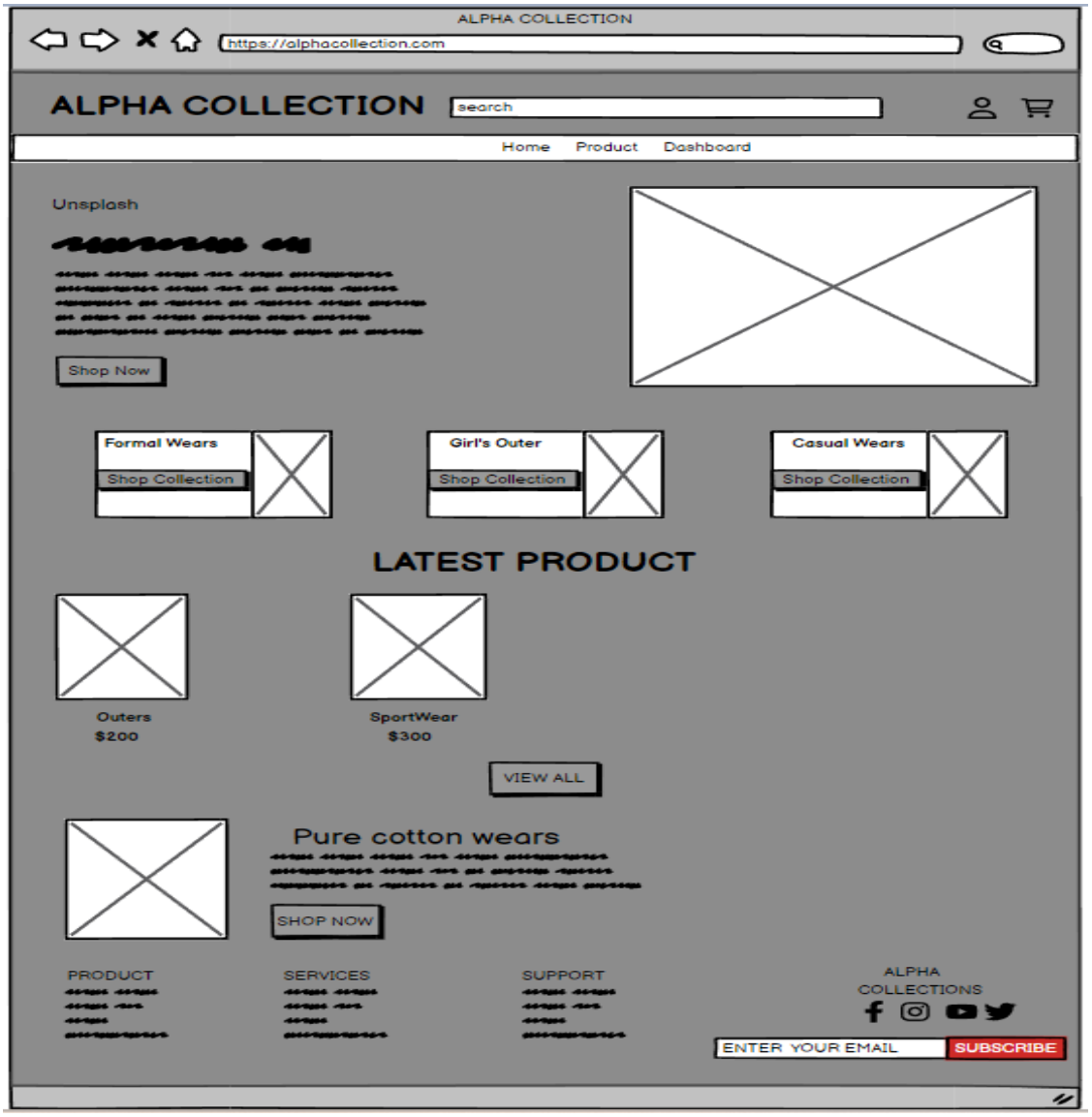

Figure 3: Wireframe of Home page

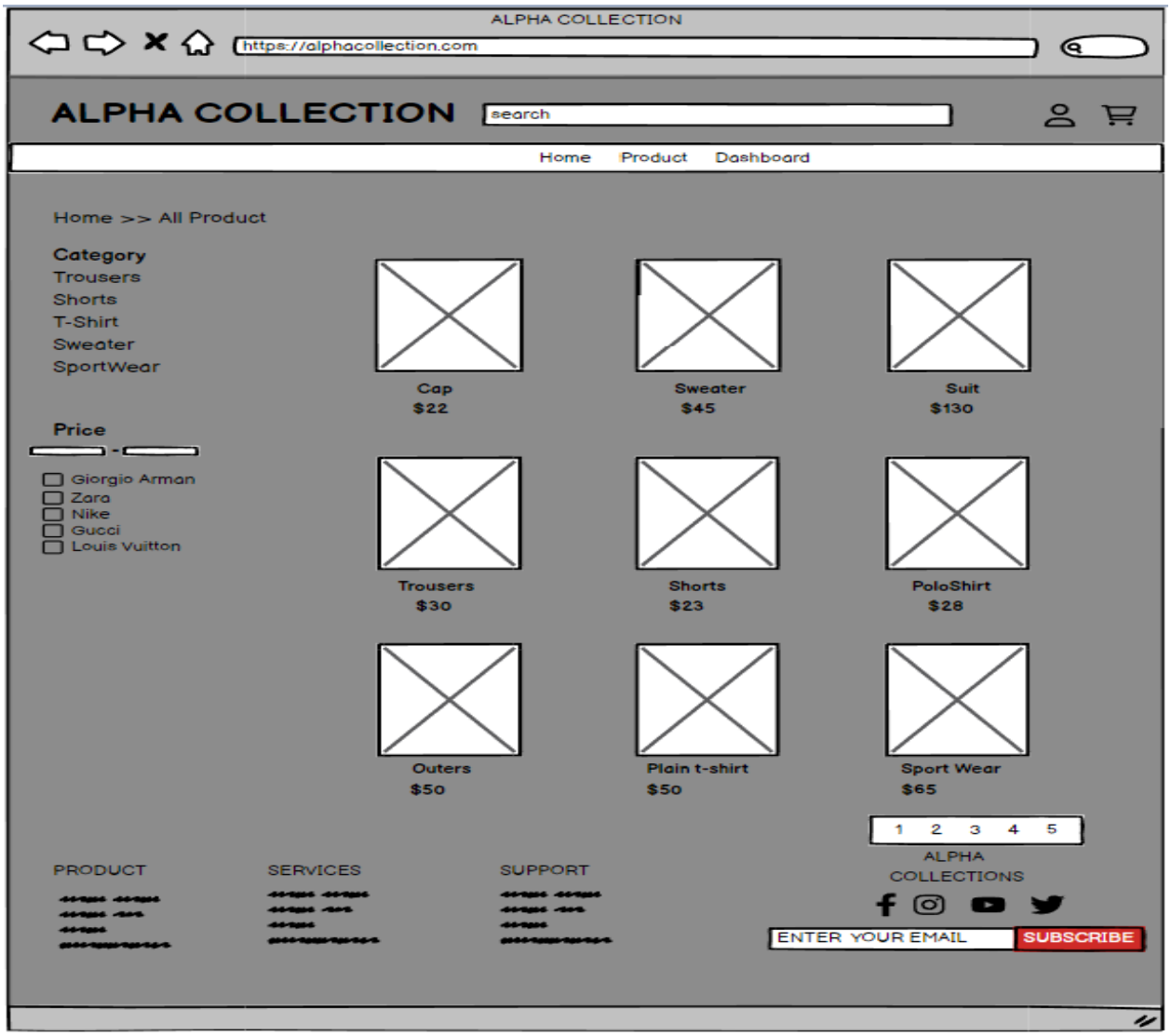

Figure 4: Wireframe of Product page

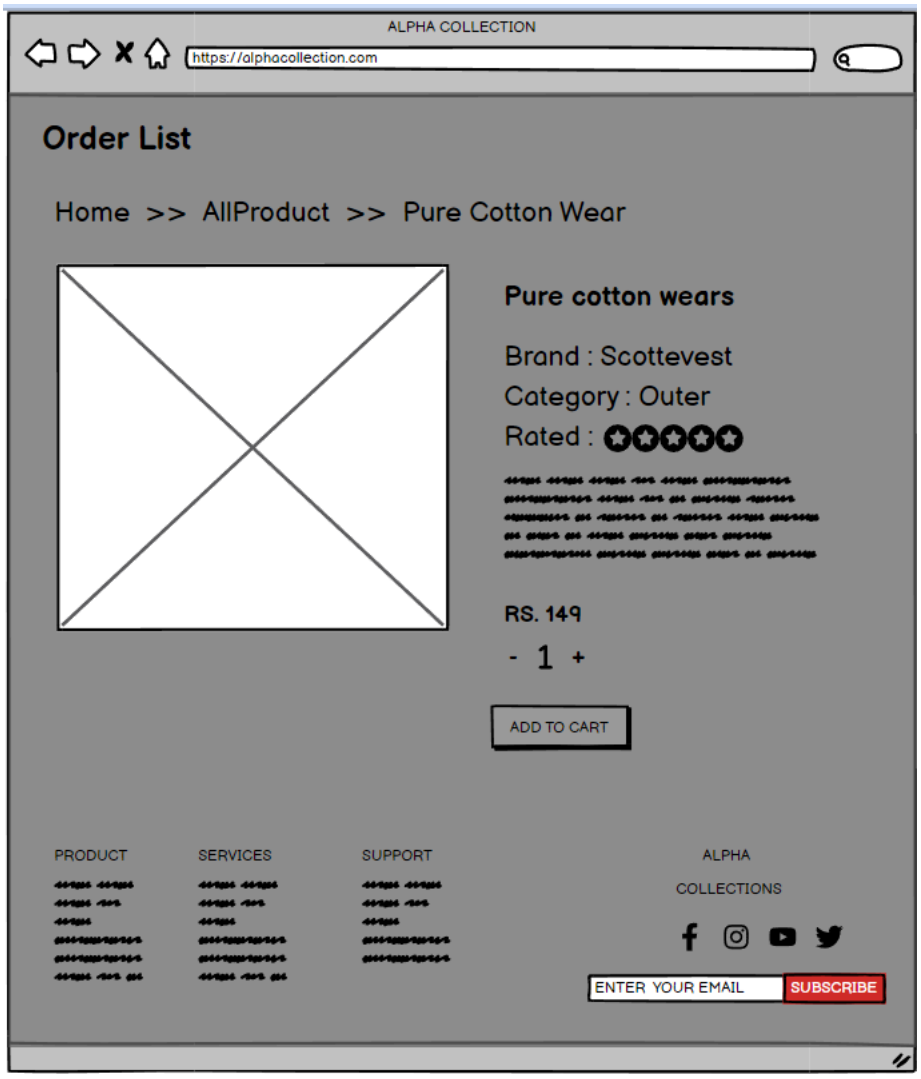

Figure 5: Wireframe of Product's details page

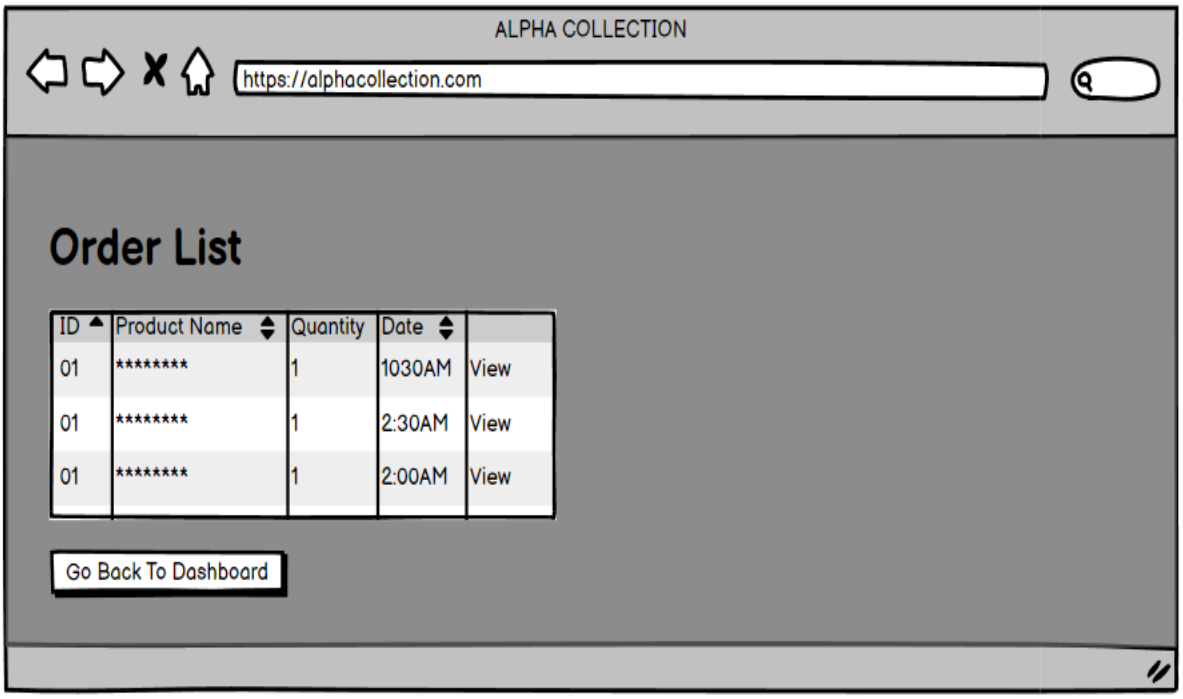

Figure 6: Wireframe of Total Order page

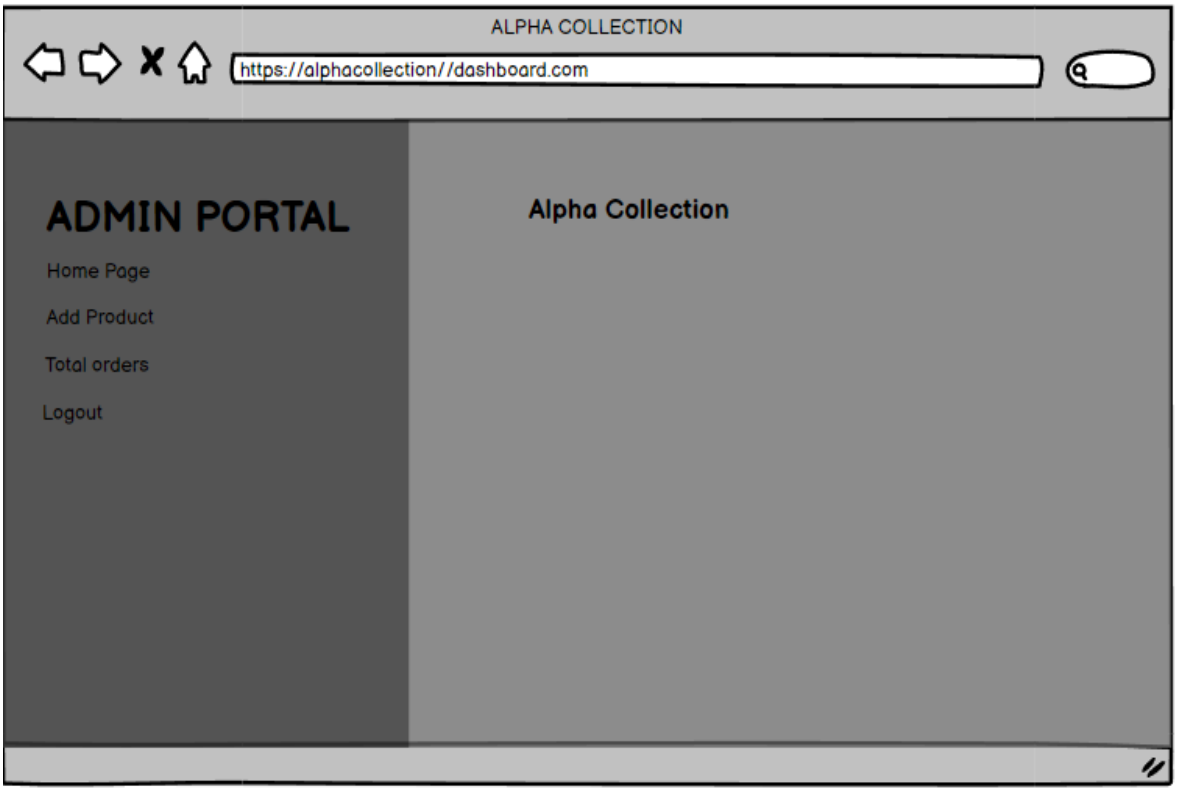

Figure 7: Wireframe of Admin's dashboard page

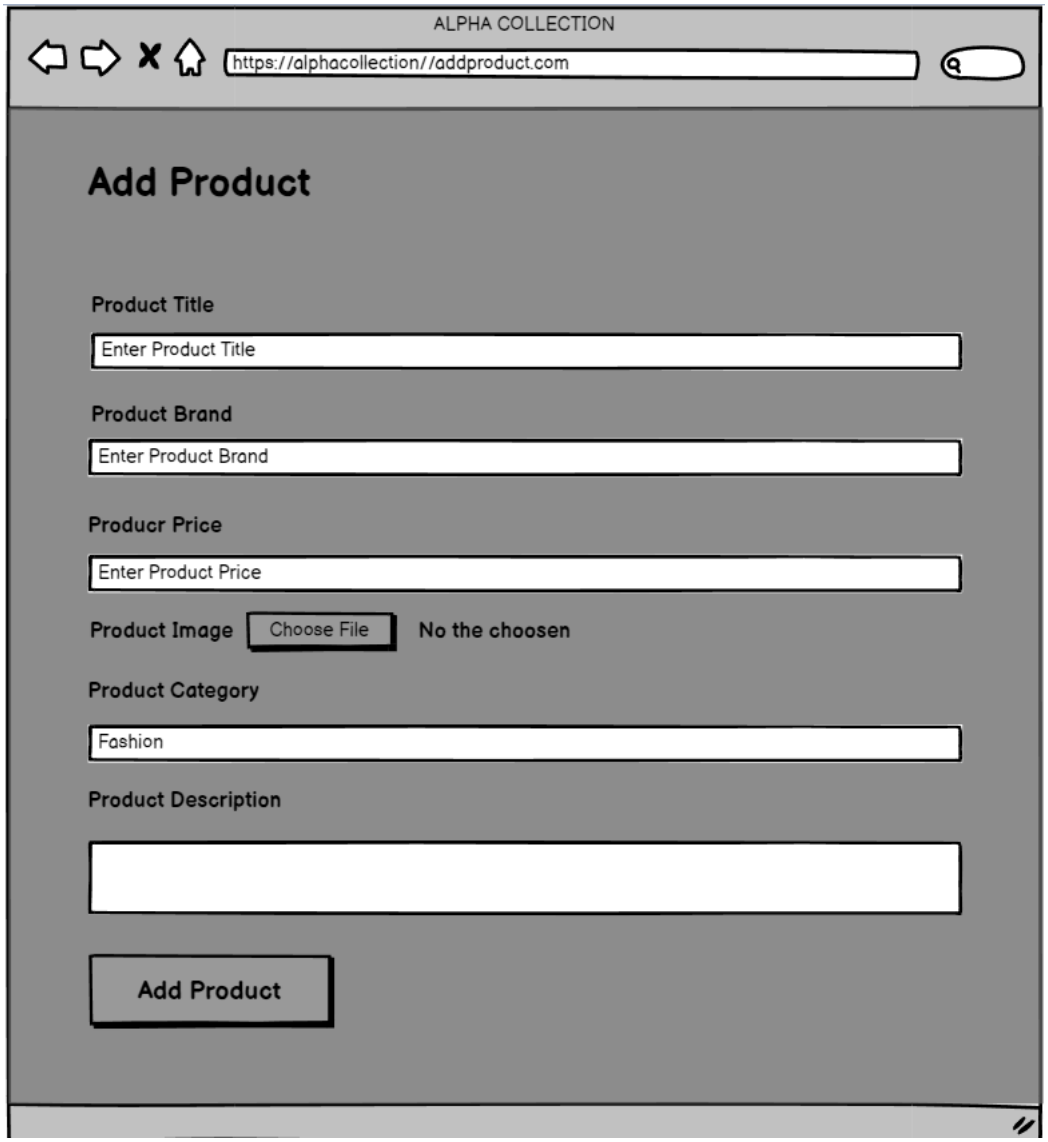

Figure 8: Wireframe of Add Products page

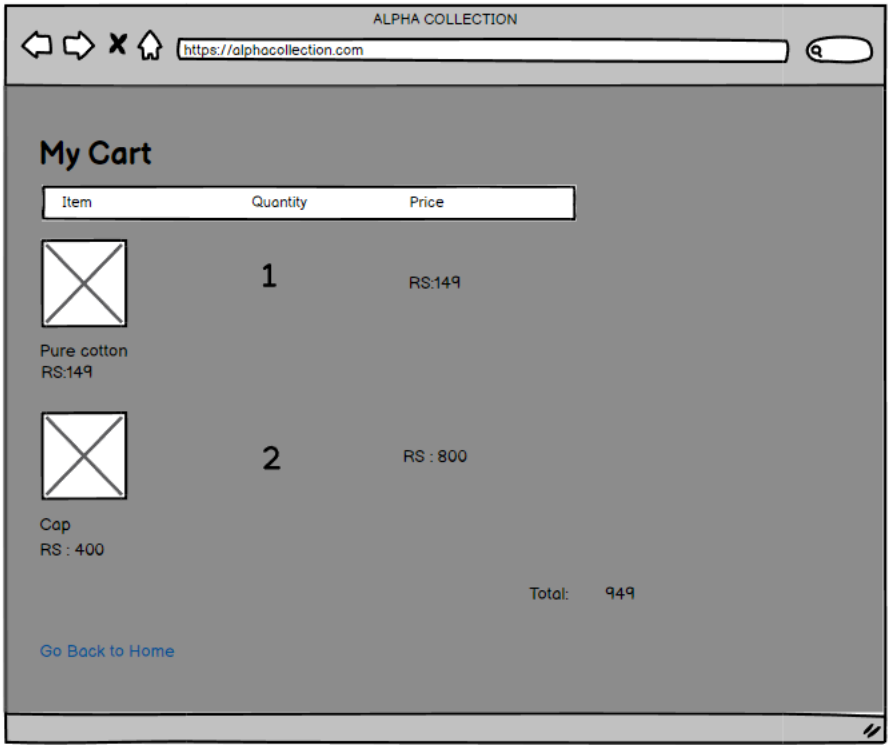

Figure 9: Wireframe of Cart page

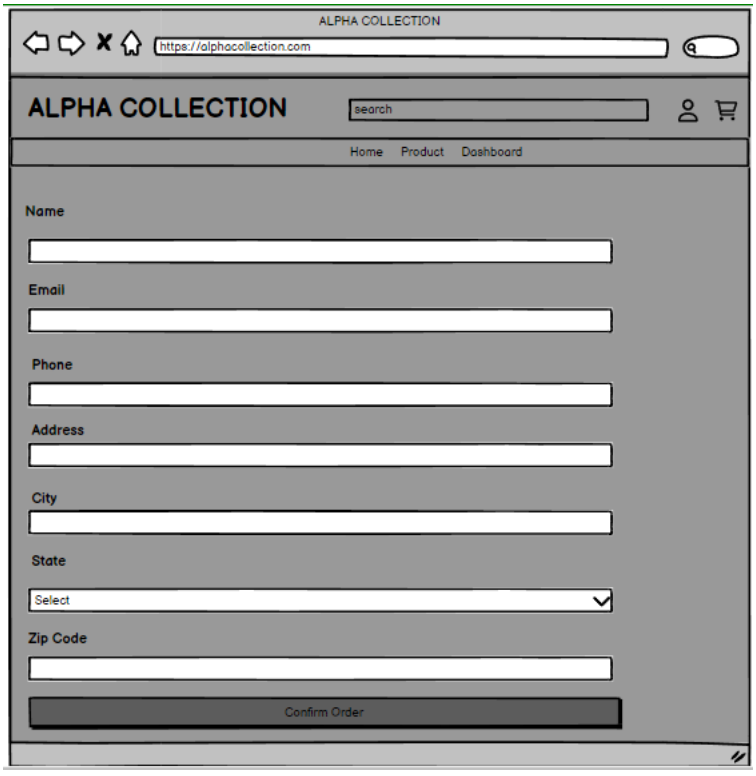

Figure 10: Wireframe of Checkout page

2.2 Actual design

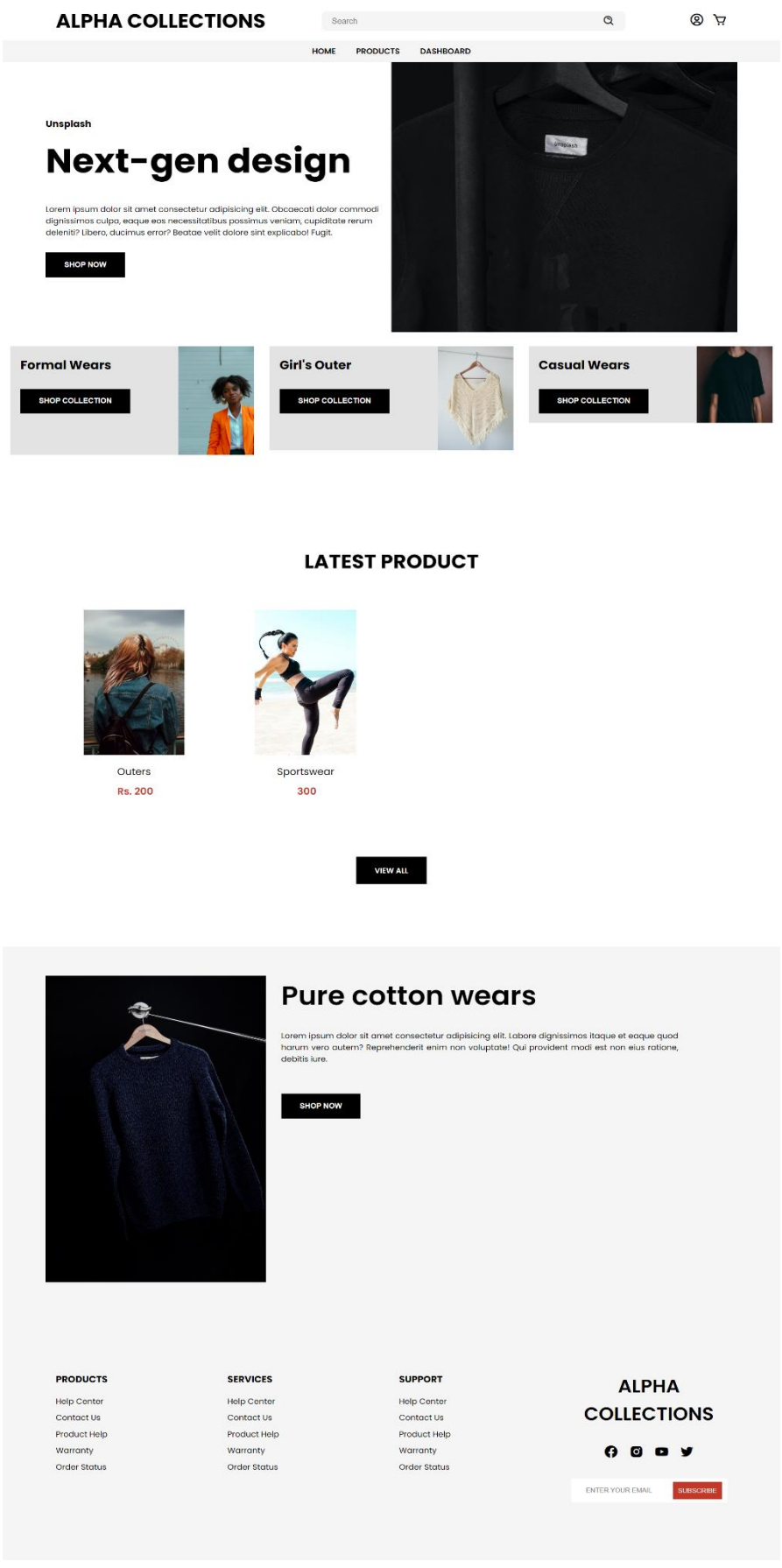

Figure 11: Actual design of Home page

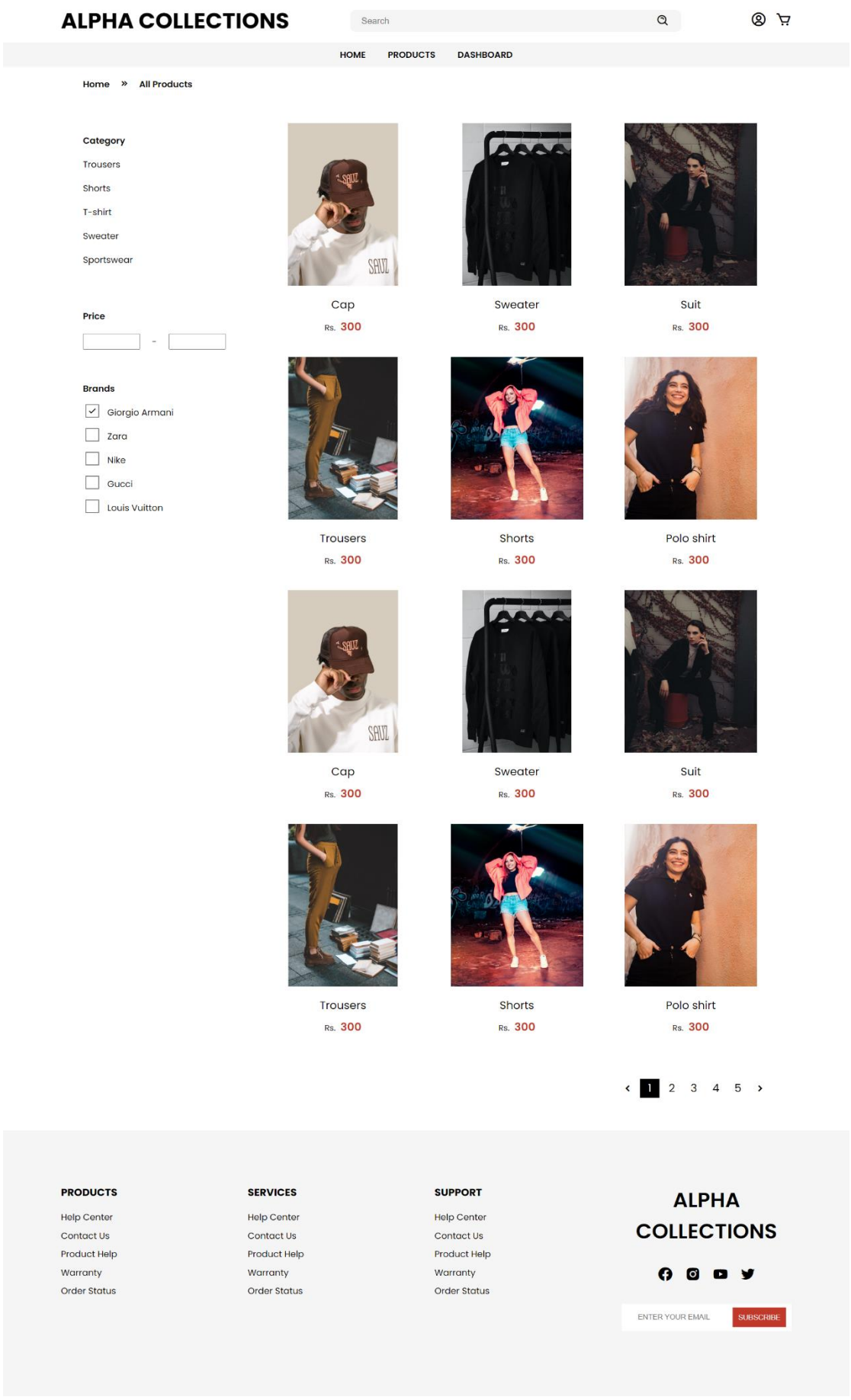
ALPHA COLLECTIONS
Search
HOME
PRODUCTS
DASHBOARD
Home » All Products
Category
Trousers
Shorts
T-shirt
Sweater
Sportswear
Price
Brands
Giorgio Armani
Zara
Nike
Gucci
Louis Vuitton
Cap
Rs. 300
Sweater
Rs. 300
Suit
Rs. 300
Trousers
Rs. 300
Shorts
Rs. 300
Polo shirt
Rs. 300
Cap
Rs. 300
Sweater
Rs. 300
Suit
Rs. 300
Trousers
Rs. 300
Shorts
Rs. 300
Polo shirt
Rs. 300
1 2 3 4 5
PRODUCTS
Help Center
Contact Us
Product Help
Warranty
Order Status
SERVICES
Help Center
Contact Us
Product Help
Warranty
Order Status
SUPPORT
Help Center
Contact Us
Product Help
Warranty
Order Status
ALPHA COLLECTIONS
ENTER YOUR EMAIL
SUBSCRIBE

Figure 12: Actual design of Product page

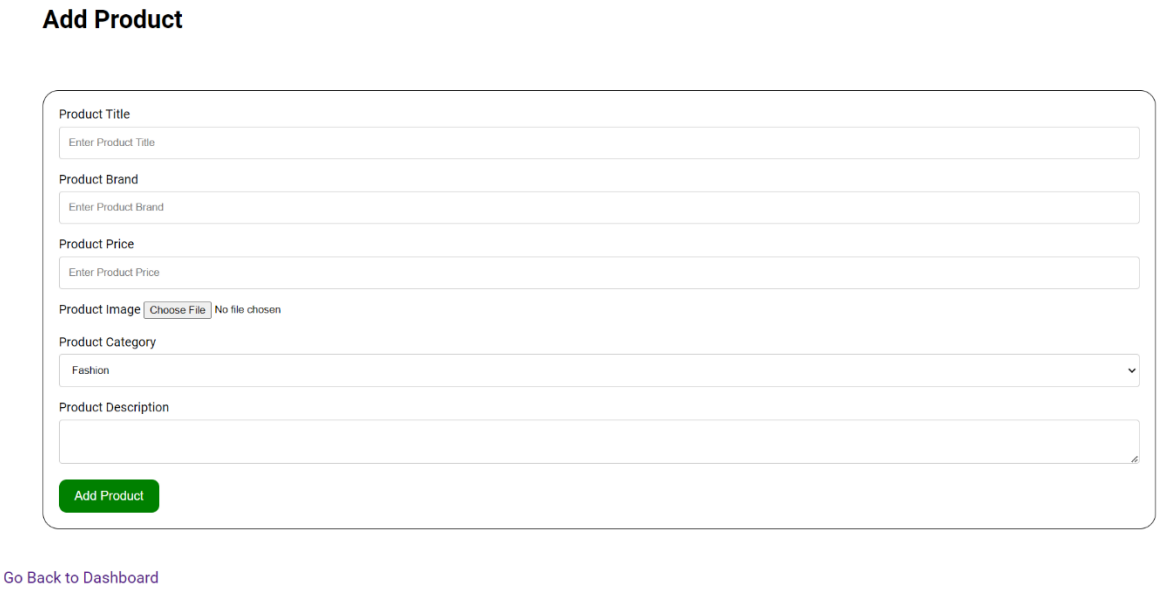

Figure 13: Actual design of Add Product page

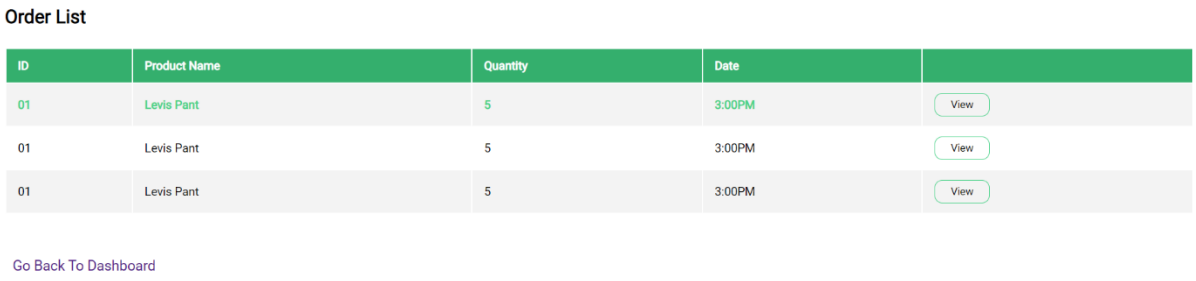

Figure 14: Actual design of Total order page

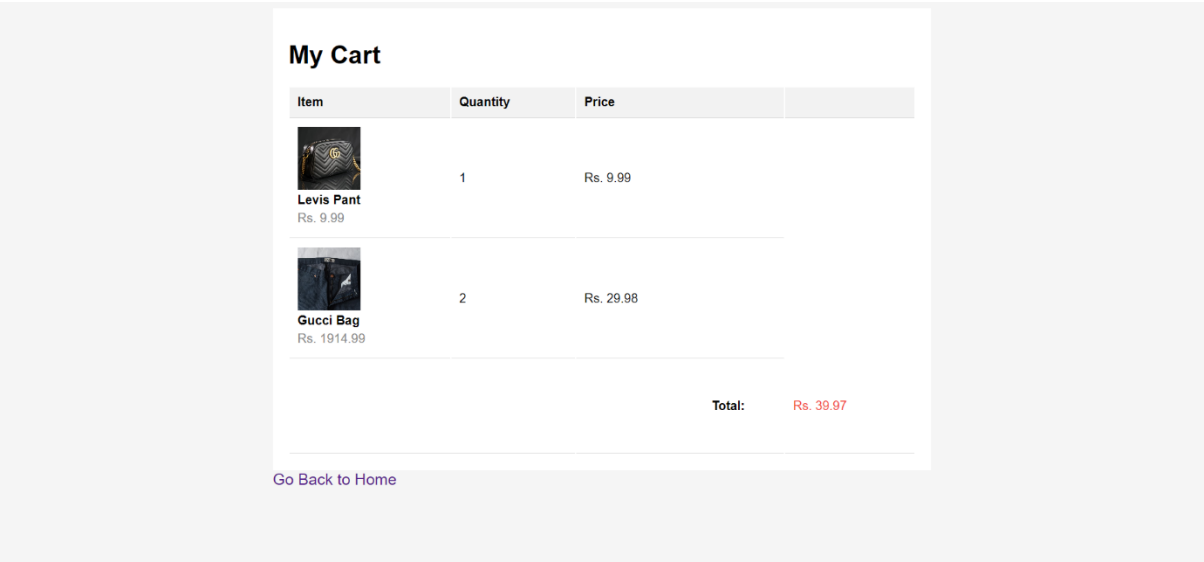

Figure 15: Actual design of Cart page

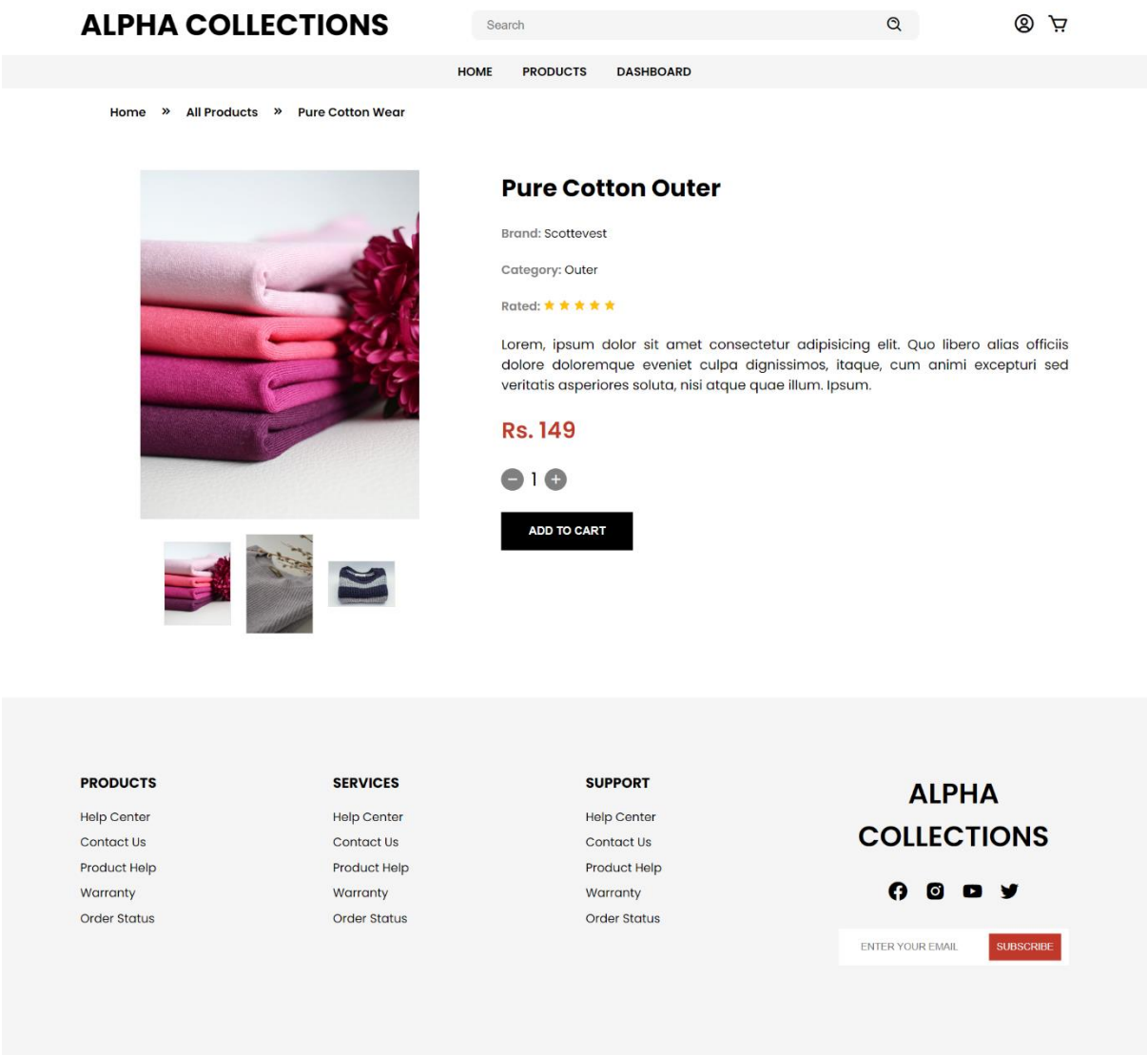

Figure 16: Actual design of Product's details page

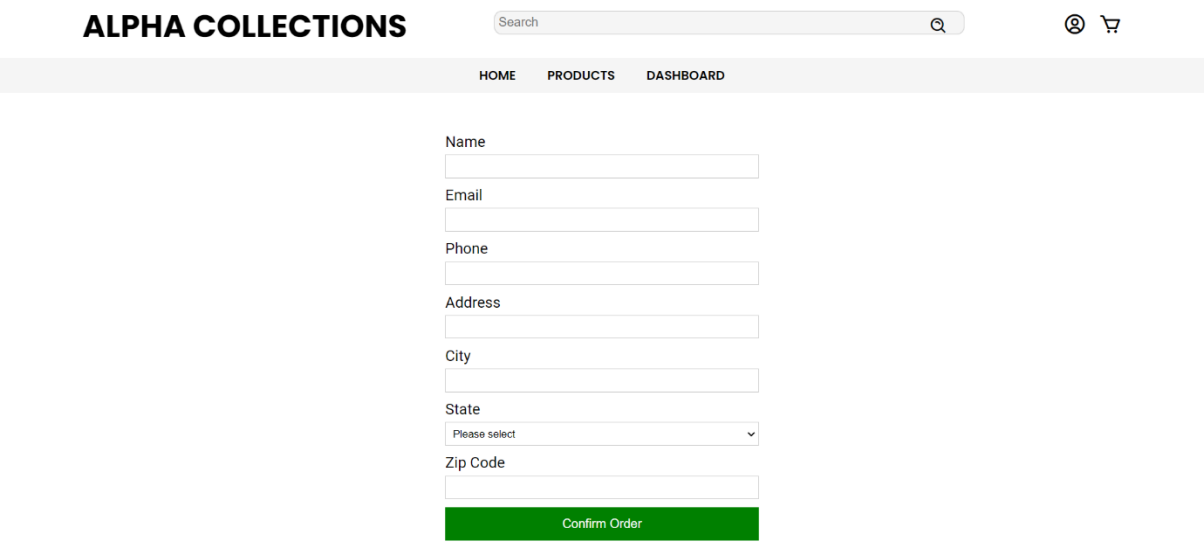

Figure 17: Actual design of Checkout page

Figure 18: Actual design of Admin's dashboard

Figure 19: Actual design of Registration Form page

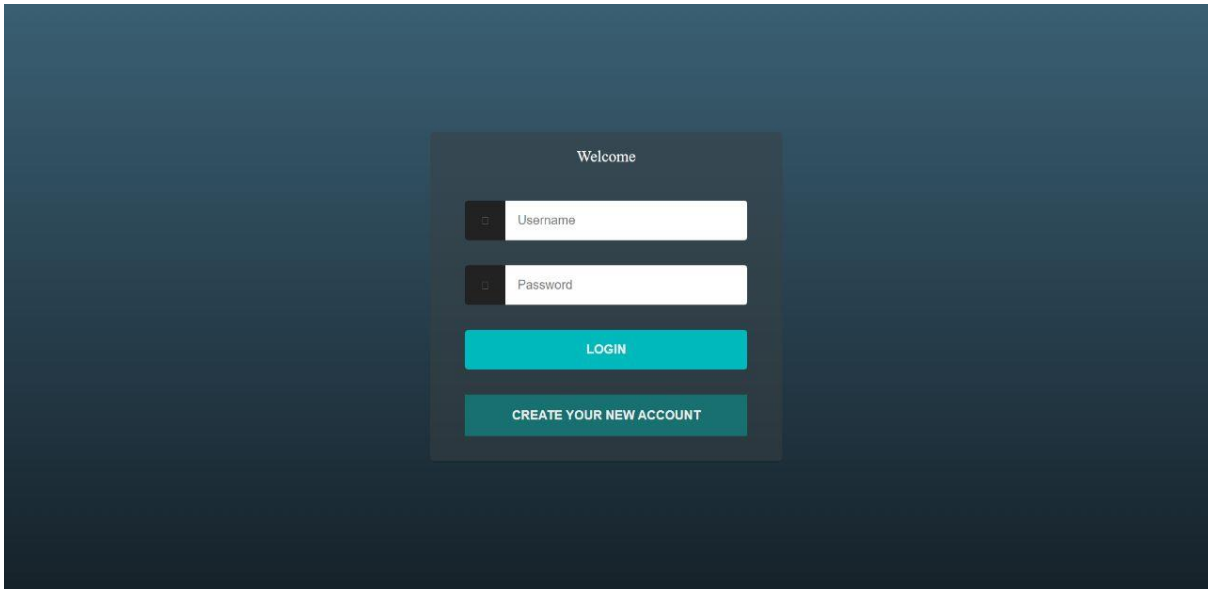

Figure 20: Actual design of Login page

3. Class Diagram

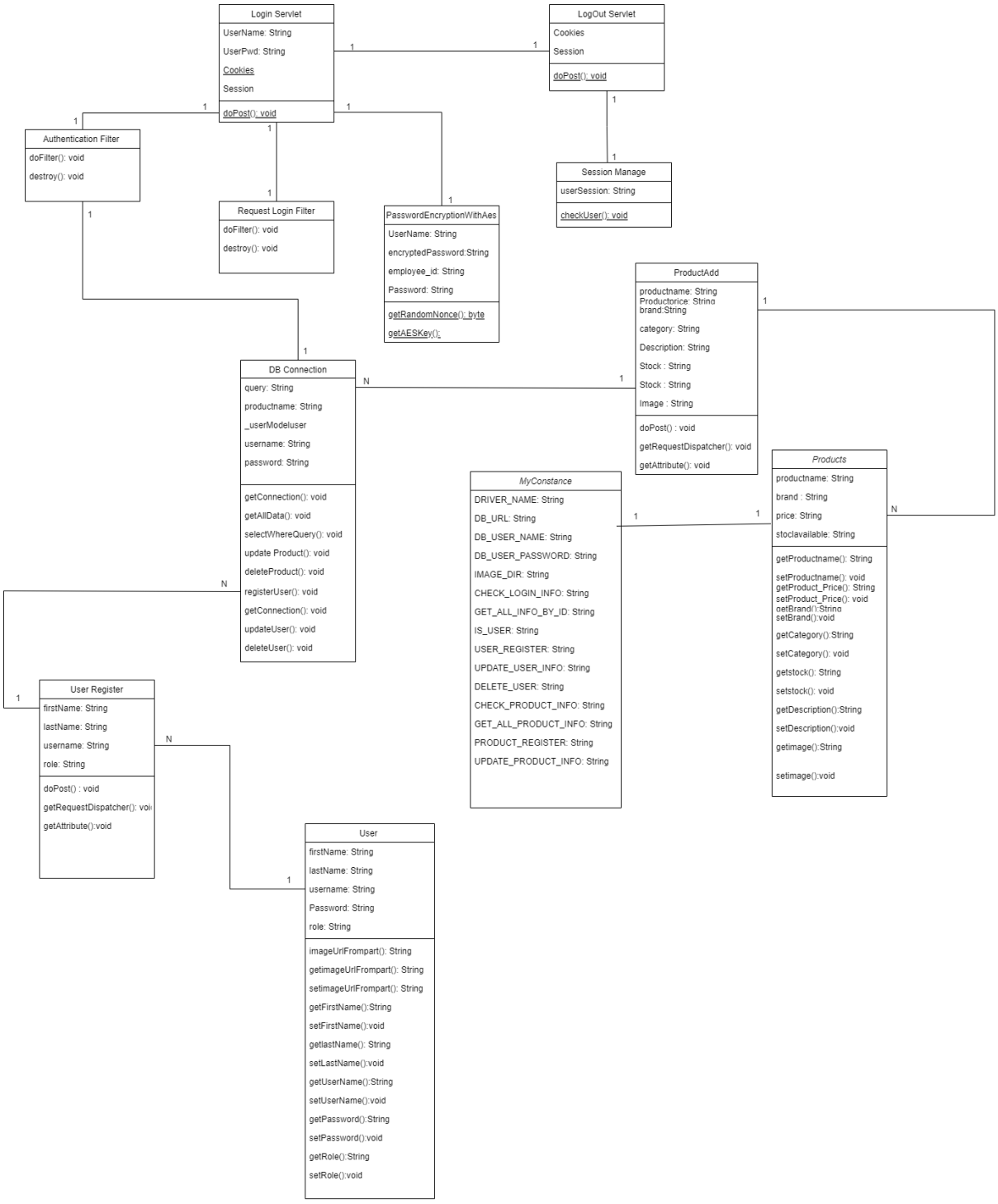

Figure 21: Class Diagram of Alfa Collection

4. Method description

4.1 Db Connection

- **selectWhereQuery ():** This method is used to execute a database SELECT query with a WHERE clause. It accepts the query as input and retrieves the matching database records.
- **updateProduct ():** This method is used to refresh a database product. It modifies the existing product data with the specified updated information.
- **deleteProduct ():** This method removes a product from the database. It eliminates the specified item from the database.
- **registerProduct ():** This method is used to add a new product to the database. It adds the product details as a new entry to the database.
- **registerUser ():** This method is used to add a new user to the system. It generates a new user account by storing the user's details in the database.
- **updateUser ():** This method is used to modify database subscriber information. It permits updating the existing user's information with the provided new information.
- **deleteUser ():** This method removes a user from the system. It deletes the user's account from the database, along with all associated data.

4.2 Authentication Filter and Request Login Filter

- **Destroy ()**

  This method is a component of the lifecycle of particular programme components or objects. It is responsible for executing cleansing operations and relinquishing resources when the component or object is no longer required or is being terminated.

- **dofilter()**

  In web applications, the "dofilter()" method handles incoming queries by using specified operations or filters. It executes authentication, data

validation, recording, and modification of request parameters. Its functionality is tailored to the needs and configuration of the application.

4.3 Session Manage

- **userSession ()**

  The userSession () method manages and maintains the user account in a web application. It offers the ability to initiate, administer, and terminate the user session. The user session class is responsible for managing the user's interaction and data persistence throughout their session in the web application.

- **Checkuser ()**

  The "Checkuser ()" method verifies a user's credentials for authorization purposes. It compares the supplied username and password against stored information to ensure that only authorised users can access resources. To protect user information, additional security measures, such as encryption or password hashing, can be implemented.

5. Test Cases

i. Testing jsp webpage

Test Name	Testing jsp webpage
Expected Output	Webpage should run and display while running the jsp file
Actual Output	jsp webpage runs and displays.
Test Result	The test was successful.

Table 1: Testing jsp webpage

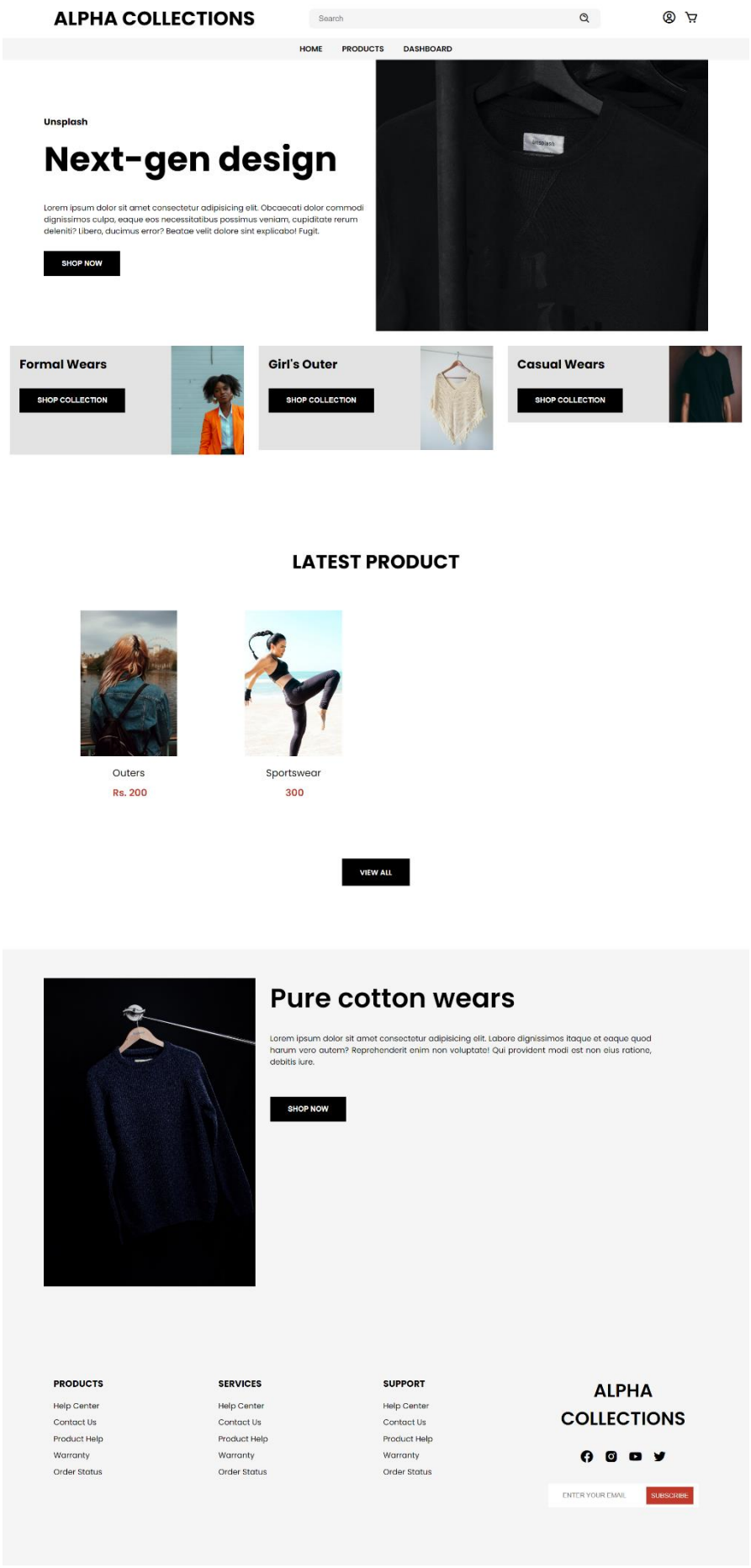
ALPHA COLLECTIONS
Search
HOME
PRODUCTS
DASHBOARD
Unsplash
Next-gen design
Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati dolor commodi dignissimos culpa, eaque eos necessitatibus possimus veniam, cupiditate rerum deleniti? Libero, ducimus error? Beatae velit dolore sint explicabo! Fugit.
SHOP NOW
Formal Wears
SHOP COLLECTION
Girl's Outer
SHOP COLLECTION
Casual Wears
SHOP COLLECTION
LATEST PRODUCT
Outers
Rs. 200
Sportswear
300
VIEW ALL
Pure cotton wears
Lorem ipsum dolor sit amet consectetur adipisicing elit. Labore dignissimos itaque et eaque quod harum vero autem? Reprehenderit enim non voluptate! Qui provident modi est non eius ratione, debitis iure.
SHOP NOW
PRODUCTS
Help Center
Contact Us
Product Help
Warranty
Order Status
SERVICES
Help Center
Contact Us
Product Help
Warranty
Order Status
SUPPORT
Help Center
Contact Us
Product Help
Warranty
Order Status
ALPHA COLLECTIONS
ENTER YOUR EMAIL
SUBSCRIBE

Figure 22: Screenshot of jsp page

ii. Testing register data is stored in database or not

Testing Name	Testing Register page
Objectives	To prove register data is stored in database
Action	➢ Filling up the Registration form with appropriate data type ➢ Clicking the "REGISTER NOW" button.
Expected Result	The form refreshes and the data are stored in the register table of the alfa database.
Result	The register table of alfa database was updated with the data filled.
Conclusion	The test was successful.

Table 2: Testing register page

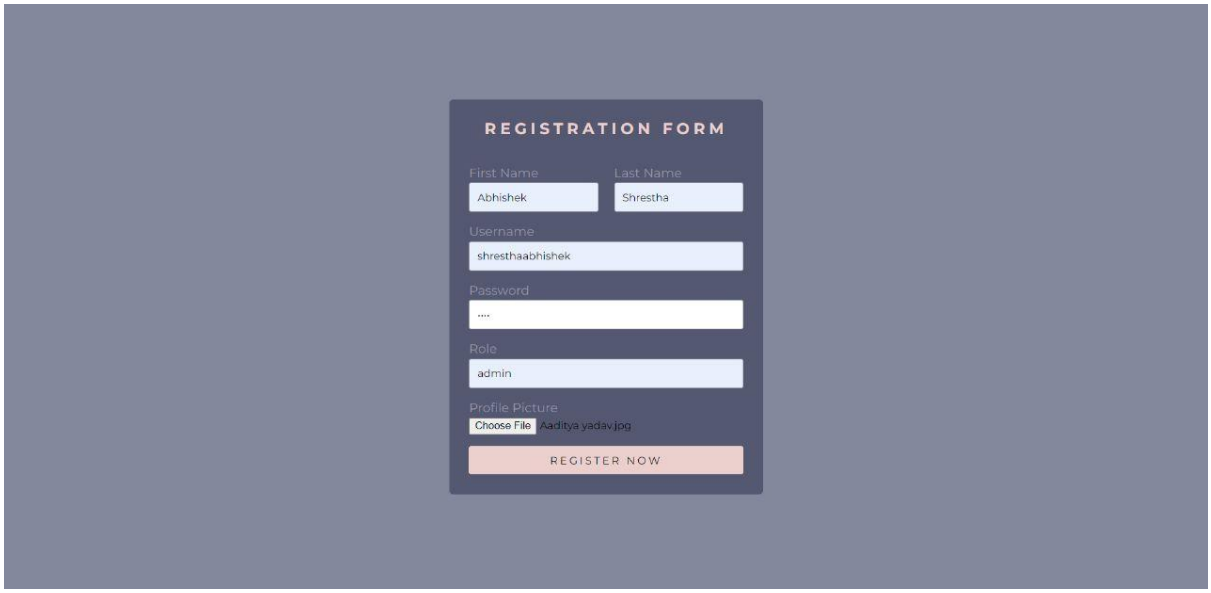

Figure 23: Inserting values in registration form

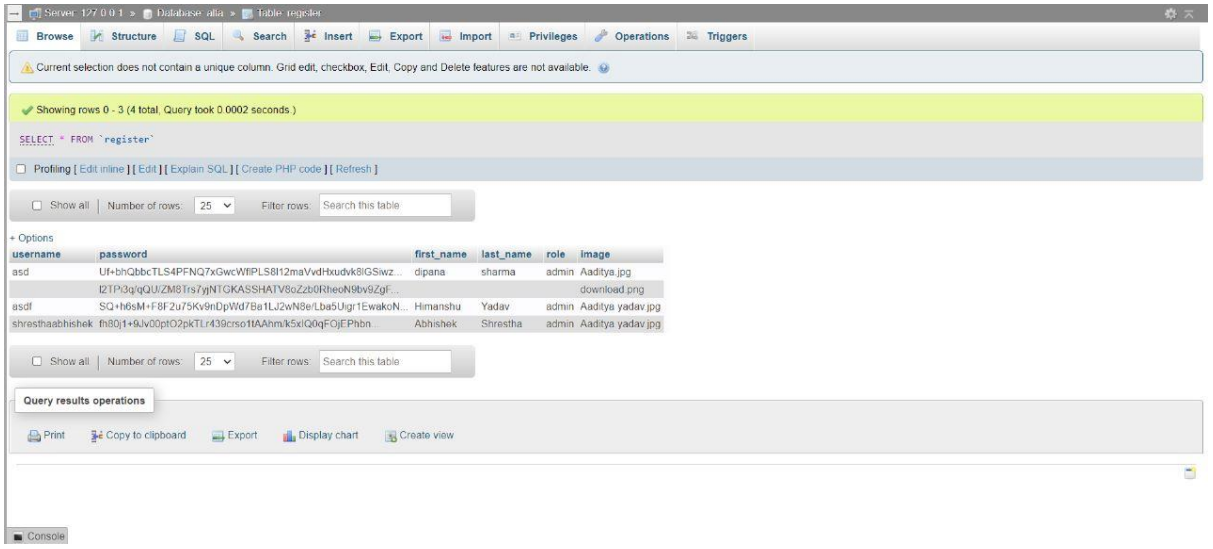

Figure 24: Checking the values in the server

iii. Testing Login page

Testing Name	Testing Login page
Objectives	To prove login data is stored in database
Action	➢ Filling up the Login form with appropriate data type ➢ Clicking the “LOGIN” button.
Expected Result	The user can go to home page if he/she has entered correct credentials.
Result	The user is directed to the home page.
Conclusion	The test was successful.

Table 3: Testing Login Form

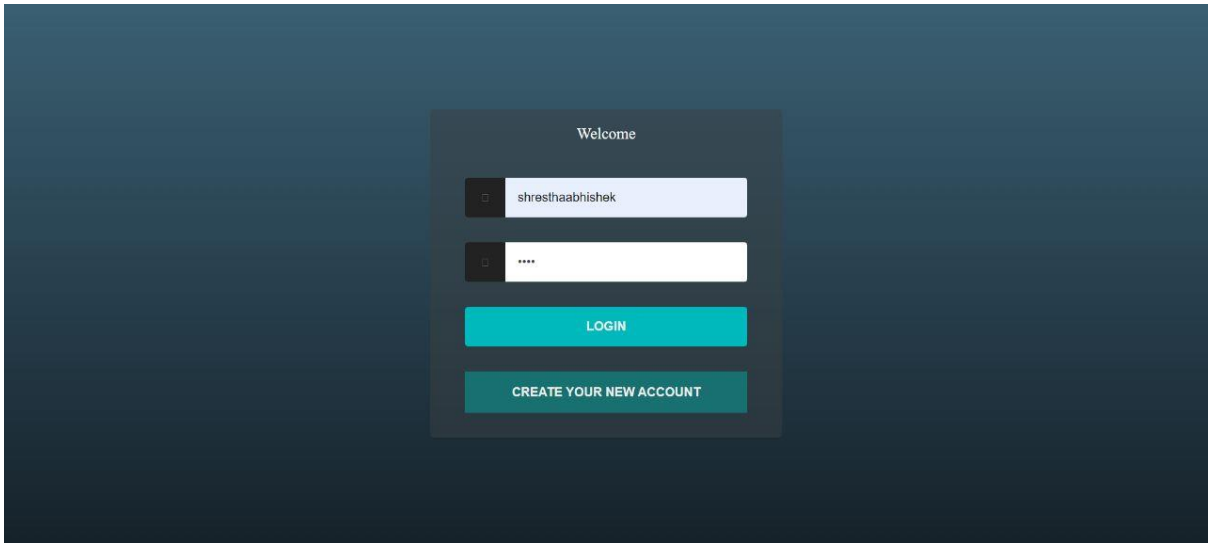

Figure 25: Screenshot of Login page

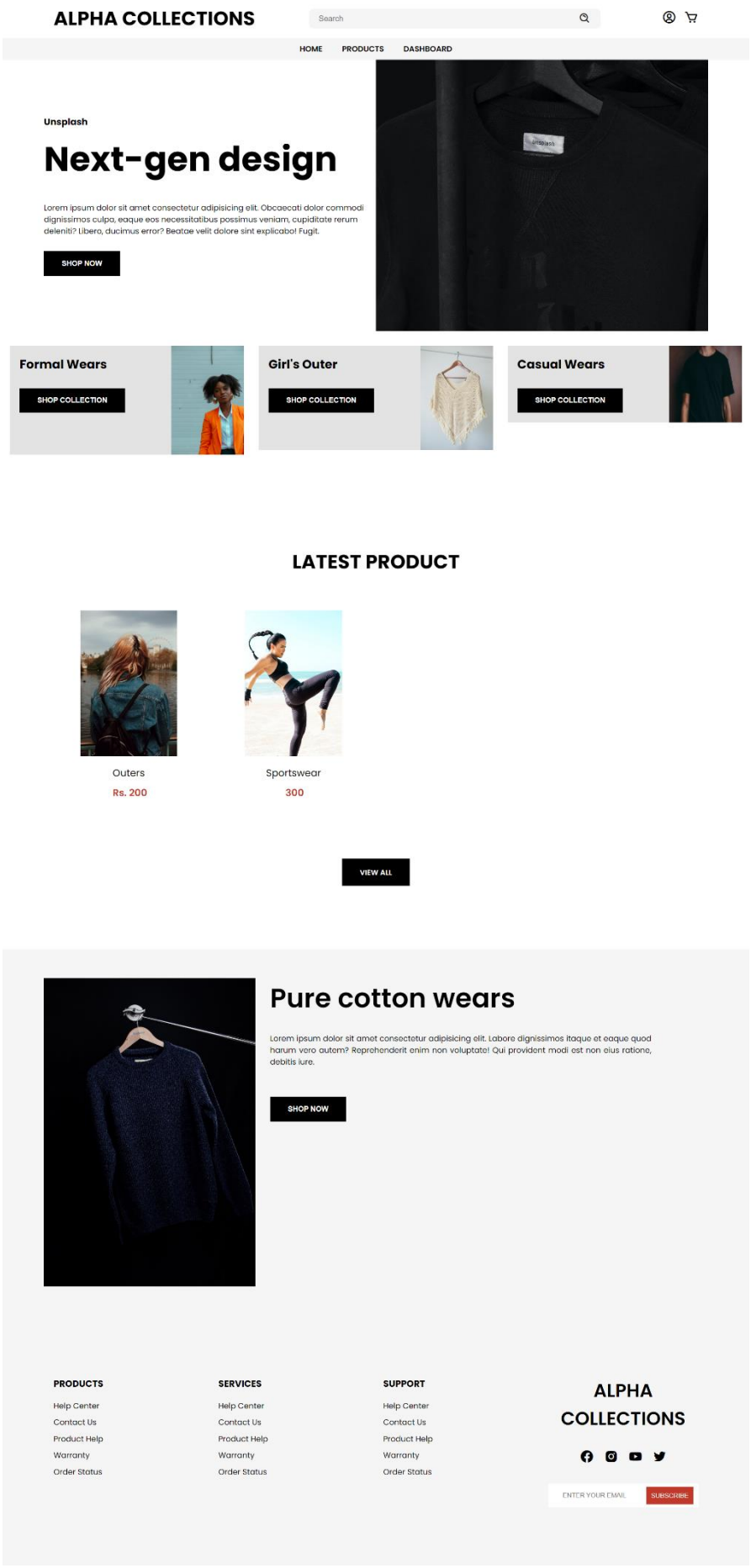
ALPHA COLLECTIONS
Search
HOME
PRODUCTS
DASHBOARD
Unsplash
Next-gen design
Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati dolor commodi dignissimos culpa, eaque eos necessitatibus possimus veniam, cupiditate rerum deleniti? Libero, ducimus error? Beatae velit dolore sint explicabo! Fugit.
SHOP NOW
Formal Wears
SHOP COLLECTION
Girl's Outer
SHOP COLLECTION
Casual Wears
SHOP COLLECTION
LATEST PRODUCT
Outers
Rs. 200
Sportswear
300
VIEW ALL
Pure cotton wears
Lorem ipsum dolor sit amet consectetur adipisicing elit. Labore dignissimos itaque et eaque quod harum vero autem? Reprehenderit enim non voluptate! Qui provident modi est non eius ratione, debitis iure.
SHOP NOW
PRODUCTS
Help Center
Contact Us
Product Help
Warranty
Order Status
SERVICES
Help Center
Contact Us
Product Help
Warranty
Order Status
SUPPORT
Help Center
Contact Us
Product Help
Warranty
Order Status
ALPHA COLLECTIONS
ENTER YOUR EMAIL
SUBSCRIBE

Figure 26: Screenshot of home page

iv. Testing hover

Testing Name	Testing Hover
Objectives	To test the hover feature of the webapp
Action	➢ Bring the mouse curser to the product images
Expected Result	The images change.
Result	The product image is changed.
Conclusion	The test was successful.

Table 4: Testing hover

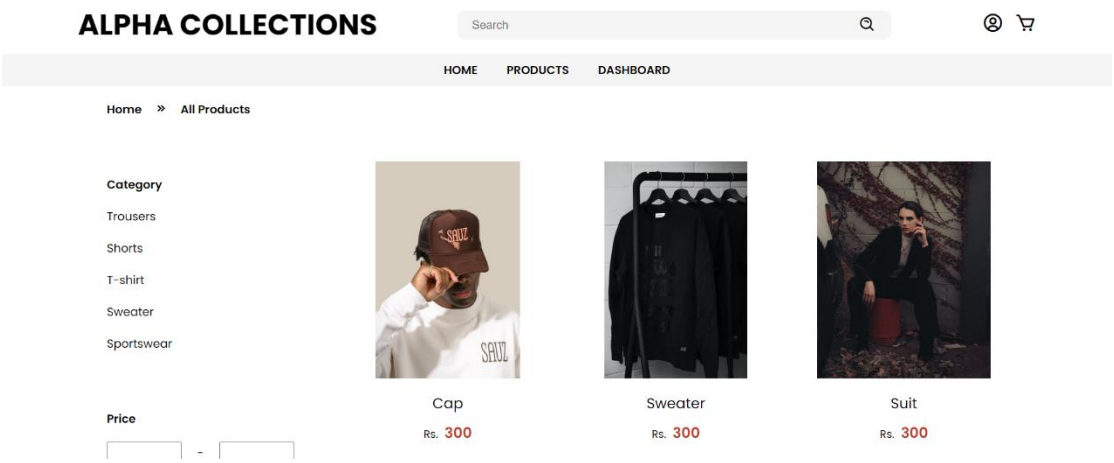

Figure 27: Screenshot of Product page without hover

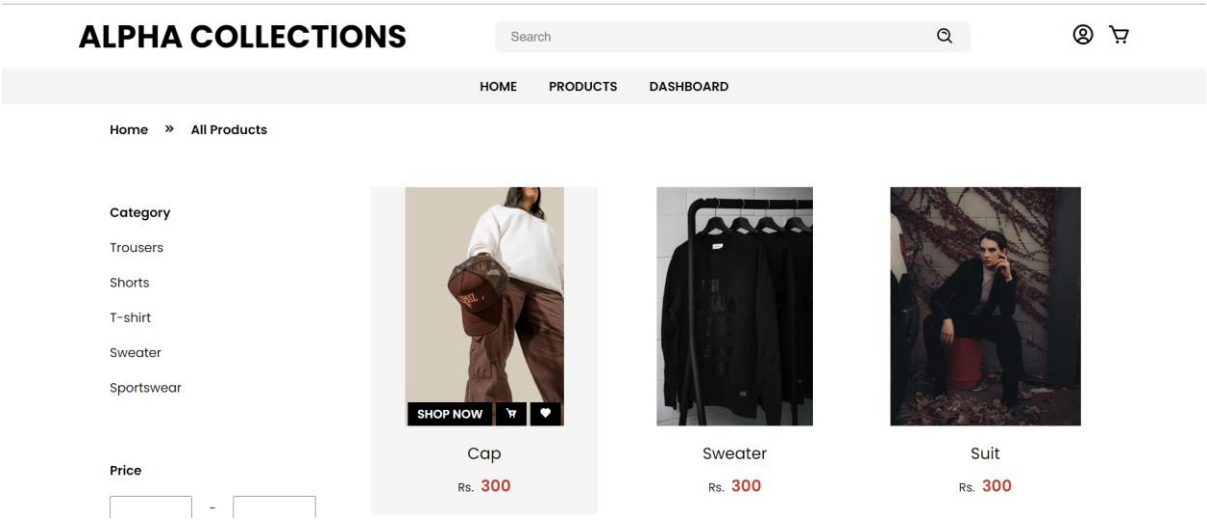

Figure 28: Screenshot of Product page with hover

v. Testing Add Product page

Testing Name	Testing Add product page
Objectives	To add data in the database while adding the product in add product page.
Action	➢ Fill the product details in the add product page.
Expected Result	The product details are stored in the product table of the alfa database.
Result	The product data is stored in the database
Conclusion	The test was successful.

Table 5: Testing Add Product Page

Figure 29: Adding Product

Product_Name	Product_Price	Category	Brand	Stock	Description	Image
Cap	550	fashion	ghuci	30	this is a cap i want	Aaditya yadav.png
T-Shirt	1100	Causal	HM	10	I like this T-Shirt	Aaditya Yadav.png

Figure 30: Screenshot of database showing added products

vi. Testing password encryption

Testing Name	Testing password encryption
Objectives	To check the password that has been stored in the database is encrypted or not.
Action	➢ Fill all the credentials in the login and register page
Expected Result	The entered password and the saved password in the database are different and is encrypted in the database.
Result	The password is encrypted.
Conclusion	The test was successful.

Table 6: Password Encryption

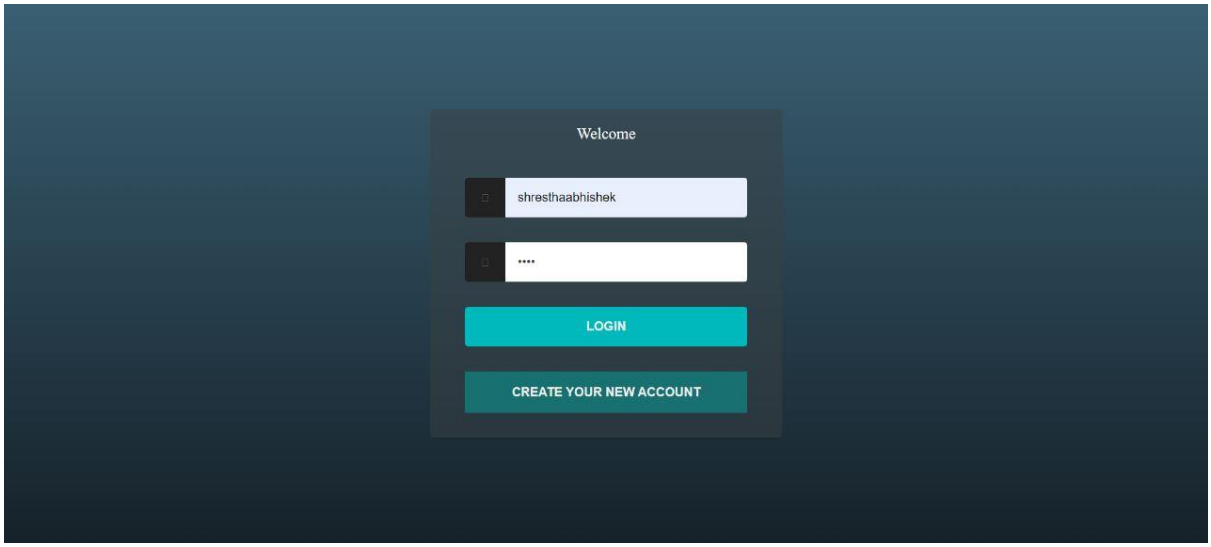

Figure 31: Screenshot while entering the unencrypted password

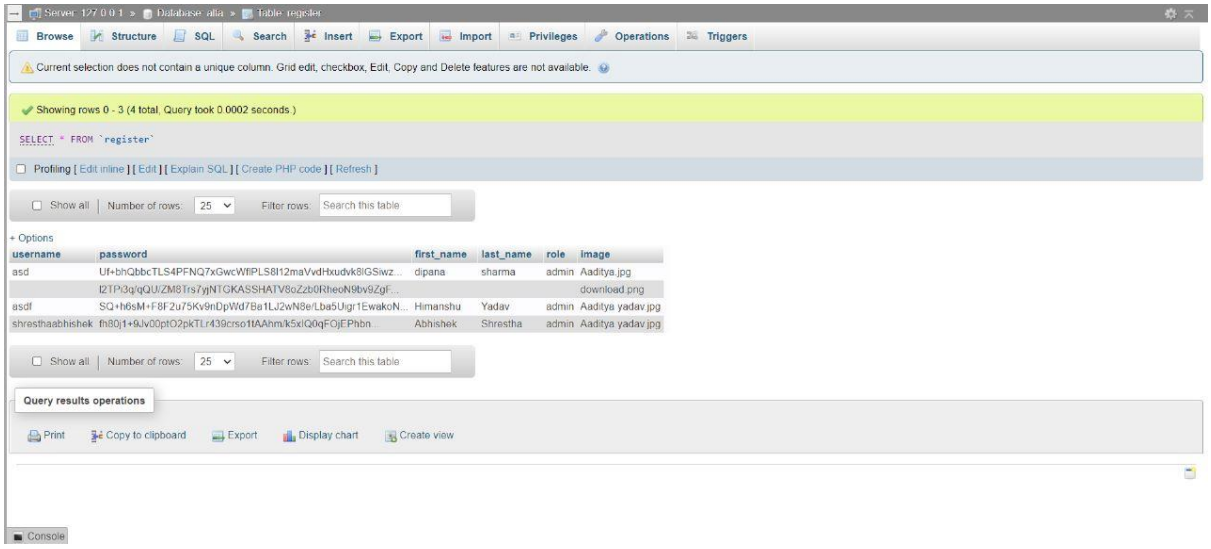

Figure 32: Screenshot of database that have password in encrypted form

6. Tools and libraries used

6.1 Eclipse

Eclipse, a Java-based IDE, supports multiple programming languages. Its plug-ins support Java, Python, C/C++, Scala, and PHP. Java developers use Eclipse with the Java Development Tools (JDT) project plug-in. (tutorialspoint, 2023)

Figure 33: Eclipse IDE

Eclipse's Java support and plugin library make Java web development popular. It has powerful Java-specific code editing, debugging, and refactoring tools. The large plugin ecosystem lets web developers customise and extend their work environment.

6.2 XAMPP

XAMPP is a free, open-source cross-platform web server. Cross-Platform, Apache, MySQL, PHP, and Perl is XAMPP. XAMPP, a popular cross-platform web server, lets developers test and develop code locally. (EDUCBA, 2021)

Figure 34: XAMPP

Eclipse uses XAMPP for web development and testing. It seamlessly integrates MySQL, Eclipse, and a comfortable local server environment. Developers can build and test full-stack web apps in this environment.

6.3 Balsamiq

Balsamiq Wireframes creates low-fidelity user interface wireframes quickly. This digital tool simulates sketching on a notepad or whiteboard. (TrustRadius, 2023)

Figure 35: Balsamiq Wireframes

Balsamiq supports iterative design, rapid prototyping for quick changes, drag-and-drop interface, and design concept communication. Balsamiq is useful for low-fidelity wireframes because designers can quickly iterate, collaborate, and improve their designs based on feedback.

7. Development Process

The development process for this coursework summary involves the following steps:

a. **Planning:** Understanding the course conditions and objectives, determining key features such as the login system, admin panel, homepage.

b. **Implementation of the MVC Pattern:** Create three MVC-compliant packages for the model, view, and controller. The controller package handles user requests with servlets, the model package manages database operations with model classes, and the view package responds with JSP and HTML files.

c. **Registration and Login System:** Implement user registration with image upload and give users and administrators the option to opt in. Ensure that credentials are encrypted and use login sessions to authenticate users.

d. **Admin Panel:** Make an admin-only JSP page. Add product information with images to the database, view product lists, edit and delete products, and view each client's order list.

e. **Home Page:** Create a homepage product display with image, price, and Add to Cart button. Allow product search. Allow non-logged-in users to search by category, price, and brand on the homepage.

f. **User Features:** Logged-in users can edit their profile (name, address, phone number, image), change their password, search by category, price, and brand, and view purchased or cart products. Implement a logout system that ends all sessions when the user logs out and restricts direct access to pages without logon.

g. **Validation and Exceptions Handling:** Validate data and manage exceptions throughout the application. Create a page with all necessary exception handling codes.

8. Critical Analysis

During the development of the e-commerce website, we encountered a number of challenges and roadblocks that required immediate resolution. Here are the issues we encountered and their respective resolutions:

When attempting to save user registration data and product information to the database, we encountered an error. We identified a configuration error in the database connection parameters as the root cause after conducting an investigation. To rectify the situation, we reviewed and modified the database configuration settings extensively. In addition, robust error management mechanisms were implemented to handle future database-related issues and provide meaningful error messages for debugging.

The inability to view uploaded images on the website's frontend was a second issue we encountered. Our project required that users upload images during user registration and product management, but we discovered that the images weren't displaying correctly on the homepage. We examined the code responsible for image uploads and formatting to address this issue. It became apparent that the database and front-end code did not store and reference image file paths correctly. By updating the file path handling logic and ensuring the correct storage and retrieval of images, we were able to successfully resolve the issue and ensure the correct display of images on the website.

Moreover, during the development of the project, organising the codebase and structuring the components according to the MVC pattern presented challenges. This hindered both code maintenance and team collaboration. To overcome this challenge, we conducted a thorough code review and implemented the necessary refactorings to conform to the MVC architectural pattern. The establishment of clear guidelines and best practises for code organisation and modularity resulted in a more structured and maintainable project.

This assignment provided valuable learning opportunities, as we encountered a variety of issues and sought our instructors' guidance. We utilised a combination of trial-and-error, careful examination of error messages, and comprehensive evaluations of configurations and code to identify and resolve issues. Despite the assignment's challenges, we were able to achieve our goals and adhere to the original project vision. In light of this experience, we are contemplating how to utilise VS Code and Eclipse for future projects.

9. Conclusion

In this coursework, we learned how to use the Model-View-Controller (MVC) pattern to create an e-commerce website for a clothing store. To ensure proper organization and separation of concerns, we began by developing three packages: model, view, and controller. We implemented a password-

encrypted login system with user registration and image uploading. Sessions were used to maintain user authentication and allowed both users and administrators to log in. Additionally, we created a page that only the administrator can access. This page allowed the administrator to add, edit, and delete products, as well as view the product list and order list for each customer.

Then, we designed a homepage with product information, including images, prices, stock, and a "Add to Cart" button. Users were able to search for and filter products by category, price, and rating. Users who were logged in could edit their profile, change their password, view their purchased or cart products, and add products to their shopping cart using the "Add to Cart" option.

We implemented proper validation and exception handling to ensure dependability and security. In addition, we maintained a consistent programming style through the use of appropriate comments and naming conventions.

Overall, this course provided practical training in developing a fully functional e-commerce website with essential features for an online clothing store. Utilising the MVC pattern enabled proper organisation and separation of concerns, thereby streamlining and optimising the development process.

References

EDUCBA, 2021. *What is XAMPP? | Complete Guide to What is XAMPP.* [Online]
Available at: https://www.educba.com/what-is-xampp/
[Accessed 02 08 2021].

TrustRadius, 2023. *Balsamiq Cloud Intro.* [Online]
Available at: https://www.trustradius.com/products/balsamiq/reviews?qs=pros-and-cons#reviews
[Accessed 07 05 2023].

Tutorialspoint.com, 2023. *Software User Interface Design.* [Online]
Available at:
https://www.tutorialspoint.com/software_engineering/software_user_interface_design.htm
[Accessed 07 04 2023].

tutorialspoint, 2023. *What is Eclipse?.* [Online]
Available at: https://www.tutorialspoint.com/eclipse/eclipse_overview.htm
[Accessed 27 03 2023].