Name: Chandana Chowdary Potu

Project: Developing an E-commerce Website for Sporty Shoes.

Project objective: As a Full Stack Developer, complete the features of the application by planning the development and pushing the source code to the GitHub repository.

This document contains sections for:

- Sprint planning and Task completion ● Core concepts used in project ● Flow of the Application.
- Demonstrating the product capabilities, appearance, and user interactions.
- Unique Selling Points of the Application
- Conclusions

Sprints planning and Task completion:

The project is planned to be completed in 2 sprint. Tasks assumed to be completed in the sprints are:

- Creating the flow of the application
- Initializing git repository to track changes as development progresses.
- Writing the Java program to fulfill the requirements of the project. Testing the Java program with different kinds of User input Pushing code to GitHub.
- Creating this specification document highlighting application capabilities, appearance, and user interactions.

Core concepts used in project:

• Object-Oriented: used to create and model objects for users and their credentials.

• Data Access Object: to abstract and encapsulate all access to the data source.

• Object–Relational Mapping: to map the objects to the database.

• Databases: used to store and retrieve data.

• Data Sources: used to define a set of properties required to identify and access the database.

• Collections: used some collections such arraylist to store collection of data.

• Exception Handling: used to catch problems that arises in the code especially in I/O blocks

Technologies Used:

• Spring MVC: to build web applications as it follows the Model-View-Controller design pattern.

• JSP: to handle the presentation view.

• Hibernate: to simplify the development and the interaction with the database.

• CSS: to format the contents.

• Bootstrap: to use some CSS and JavaScript designs.

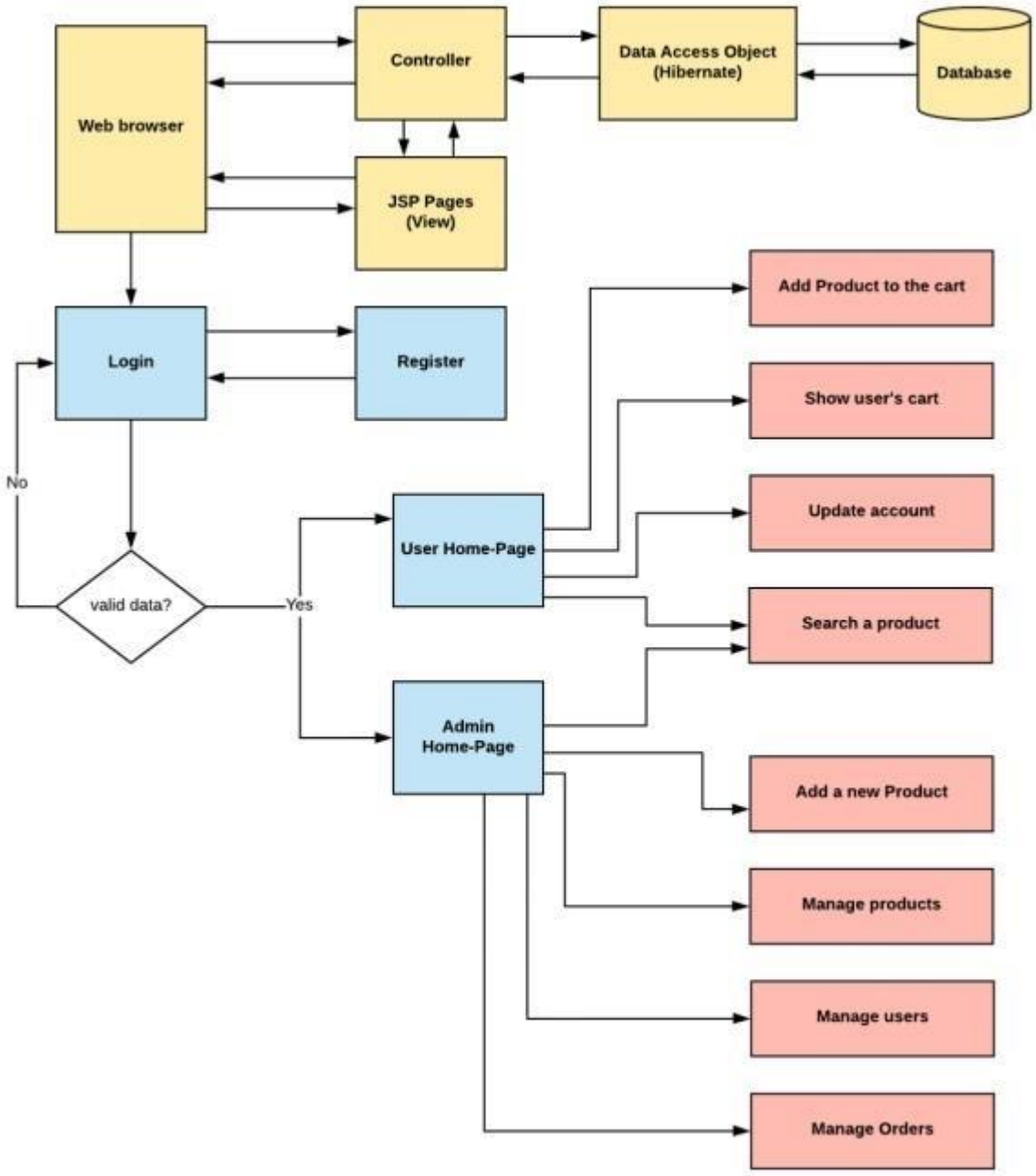
• Maven: to manage the project.

• Eclipse: to write and run the code.

• phpMyAdmin: to administrate and manage the database manually.

• Tomcat: to run and deploy servlet application.

Flow of the Application:


Web browser
Controller
Data Access Object (Hibernate)
Database
JSP Pages (View)
Login
Register
No
valid data?
Yes
User Home-Page
Admin Home-Page
Add Product to the cart
Show user's cart
Update account
Search a product
Add a new Product
Manage products
Manage users
Manage Orders

Project Users Stories : (Agile and Scrum)

The project is planned to be completed in 2 sprints. Tasks assumed to be completed in the sprint are:

- Creating the flow of the application
- Initializing git repository to track changes as development progresses.
- Writing the Java program to fulfill the requirements of the project. Testing the Java program with different kinds of User input Pushing code to GitHub.

1) As an admin I can Set up a master list of all the products.

2) As an admin I can Set up a master list of all the users.

3) As an admin I can Set up a master list of all the orders.

4)Manage the products in the store including categorizing them
5) Browse the list of users who have signed up and be able to search users
6) See purchase reports filtered by date and category

Demonstrating the product capabilities, appearance, and user interactions:

To demonstrate the product capabilities, below are the sub-sections configured to highlight appearance and user interactions for the project:

Step 1: Creating a new project in Eclipse

- Open Eclipse
- Go to File -> New -> Project -> Maven Project -> Next.
- Type in any project name and click on “Finish.” ● Select your project and go to File -> New -> Class.

Step 2:

Java files

AdminController.java

HomePageController.java

LoginController.java

LoginController.java

OrderDAOImpl.java

ProductDAO.java

ProductDAOImpl.java

UserDAO.java

Order.java

Product.java

User.java

Jsp files

Addproduct.jsp

Left-list.jsp

Login.jsp

ManageOrders.jsp

manageUsers.jsp

manage-products.jsp

myaccount.jsp

mycart.jsp register.jsp

update-products.jsp

user-home.jsp

CSS files

login.css

style.css

Shoesdata.sql

All above mentioned file are attached via zip file all these are

Step 3:

1. **Creating the project in eclipse.**

2 Import the “database\database.sql” file to your database administration tool.

-- phpMyAdmin SQL Dump

-- version 5.1.0

-- https://www.phpmyadmin.net/

-- --

--
Host: 127.0.0.1:3307

Generation Time: April 13, 2022 at 03:43 PM

-- Server version: 10.4.18-MariaDB

-- PHP Version: 8.0.3

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

START TRANSACTION;

SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8mb4 */;

--

-- Database: `sporty-shoes`

--

-- --

-- --

--

Table structure for table `orders`

```
CREATE TABLE `orders` (
  `id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL,
  `product_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `orders`

--

```
INSERT INTO `orders` (`id`, `user_id`, `product_id`) VALUES
(7, 3, 1),
(10, 20, 3),
(11,  20,  1),  (12,
20, 4); drop tables
orders;
```

-- ----------

```
-- --

--

    Table structure for table `products`



CREATE TABLE `products` (

  `id` int(11) NOT NULL,

  `name` varchar(50) NOT NULL,

  `company` varchar(50) NOT NULL,

  `size` int(11) DEFAULT NULL,

  `price` double NOT NULL,

  `image_link` varchar(2555) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;



--

-- Dumping data for table `products`

--



INSERT INTO `products` (`id`, `name`, `company`, `size`, `price`, `image_link`) VALUES

(1, 'Reebok Originals', 'Reebok', 34, 654, 'https://i.imgur.com/NYeJGJw.jpg'),

(2, 'The Puma 750', 'Puma', 24, 543, 'https://i.imgur.com/JzC8UwF.jpg'),
```

-- --

--

(3, 'Adidas original', 'Adidas', 45, 678, 'https://imgur.com/0GXitLh.jpg'),

(4, 'Bata Sk8-Hi Flame', 'Bata', 28, 897, 'https://imgur.com/xa1NmjZ.jpg');

drop tables products;

```
-- ------------------------------------------------------


--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `type` int(10) NOT NULL DEFAULT 0,
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `age` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `users`
--

INSERT INTO `users` (`id`, `type`, `username`, `password`, `age`) VALUES
(2, 0, 'Anu', '1234', 24),
(4, 0, 'anju', '1234', 34),
(6, 0, 'karthi', '1234', 1),
```

```
(8, 0, 'Sam', '1234', 23),

(15, 1, 'admin', 'admin', 0),

(16,  0,  'Hasin',  '4242',  24),

(13, 0, 'Manasa', '12345', 33);

drop tables users;



--

-- Indexes for dumped tables

--



--

-- Indexes for table `orders`

--

ALTER TABLE `orders`

  ADD PRIMARY KEY (`id`),

  ADD KEY `product_id` (`product_id`),

ADD KEY `user_id` (`user_id`);   commit;



--

-- Indexes for table `products`

--
```

```
ALTER TABLE `products`
ADD PRIMARY KEY (`id`);
commit;

--
-- Indexes for table `users`
--
ALTER TABLE `users`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `orders`
--
ALTER TABLE `orders`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=13;
  commit;

--
-- AUTO_INCREMENT for table `products`
```

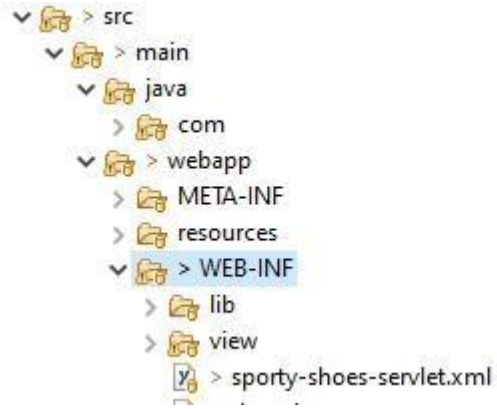
```
--

ALTER TABLE `products`

  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;

  commit;


--

-- AUTO_INCREMENT for table `users`

--

ALTER TABLE `users`

  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=21;


--

-- Constraints for dumped tables

--


--

-- Constraints for table `orders`

--

ALTER TABLE `orders`

  ADD CONSTRAINT `product_id` FOREIGN KEY (`product_id`) REFERENCES `products` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,

  ADD CONSTRAINT `user_id` FOREIGN KEY (`user_id`) REFERENCES `users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

COMMIT;
```

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

3. Go to "main\webapp\WEB-INF\sporty-shoes-servlet.xml" file, open it.

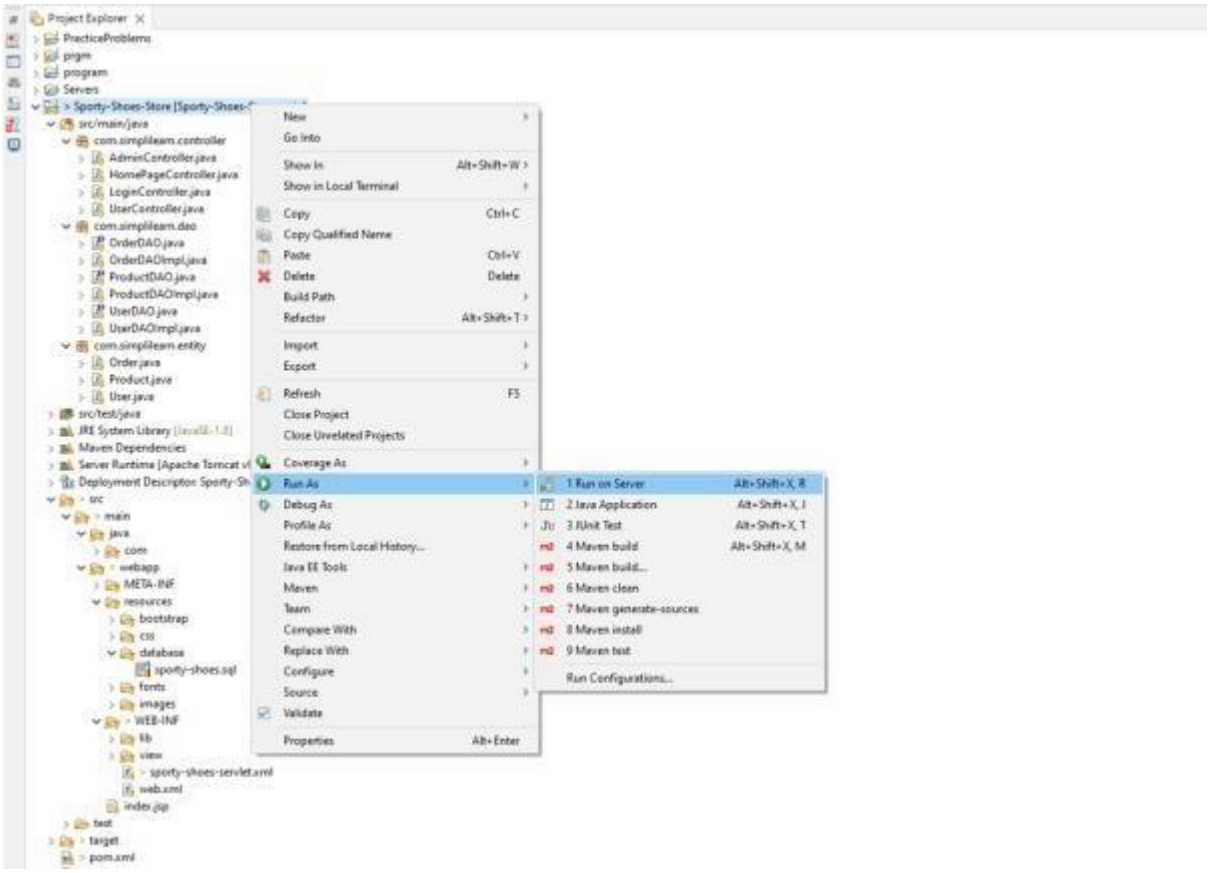


4. Edit the database' properties such as username, password and driverClassName to be suit to your database administration tool.

```
<!-- Step 1: Define Database DataSource / connection pool -->

      <bean id="myDataSource" class="com.mchange.v2.c3p0.ComboPooledDataSource"

destroy-method="close">

        <property name="driverClass" value="com.mysql.cj.jdbc.Driver" />

        <property name="jdbcUrl" value="jdbc:mysql://localhost:3306/shoesdata" />

        <property name="user" value="root" />
```

```
<property name="password" value="Chandu@2022" />
```

5.Now run program on a server.

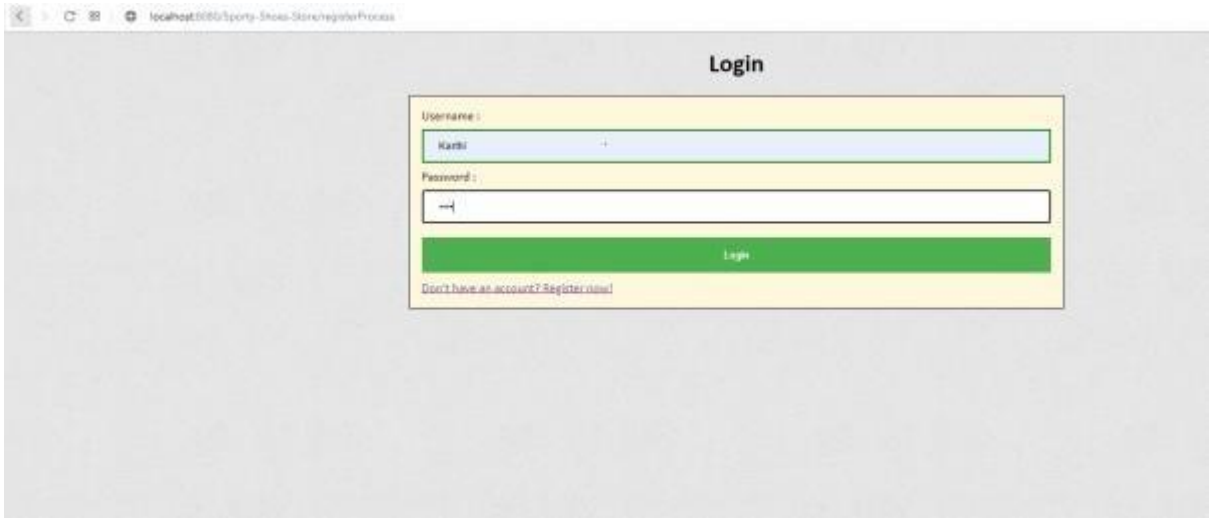


6. To login you must enter admin for both username and password.

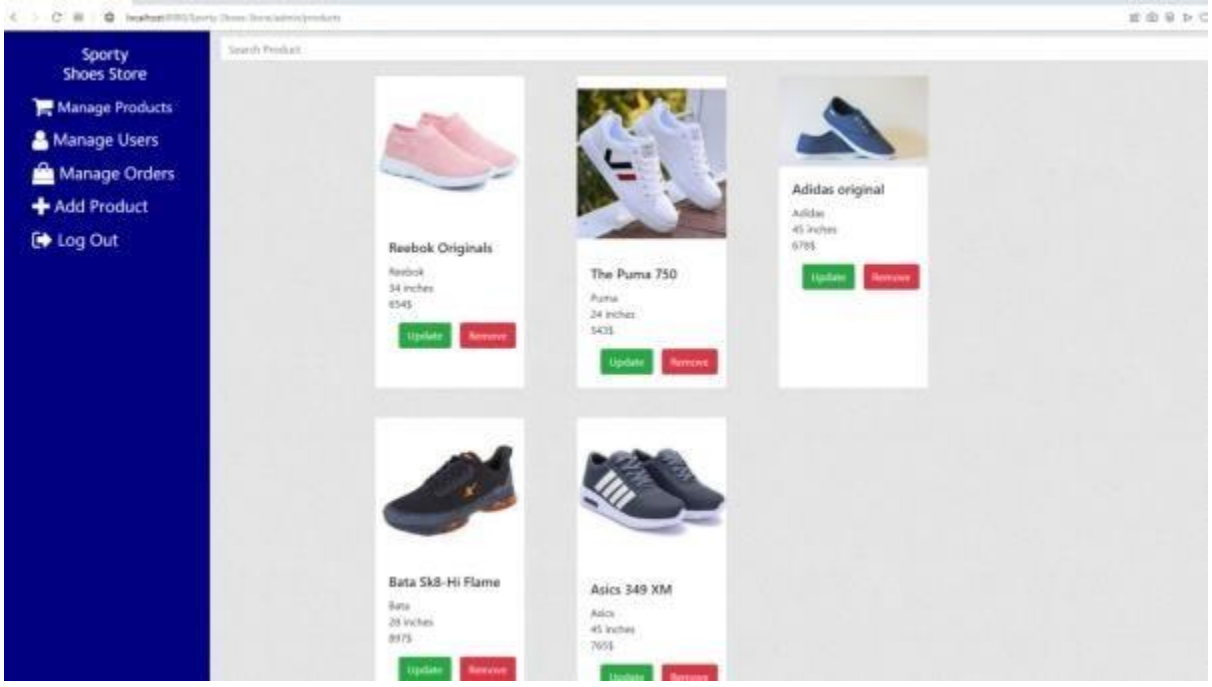
Screenshots:

1. **Register page:**

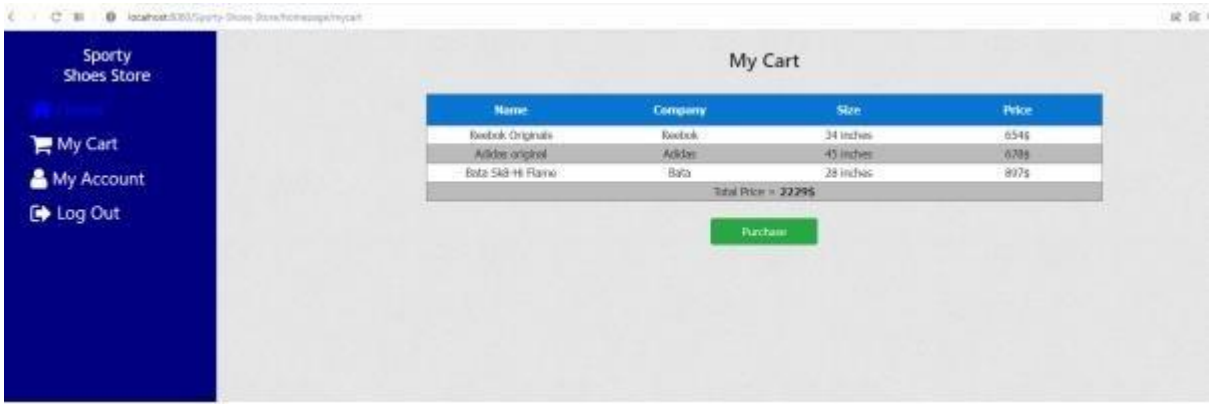
2. **Login page :**



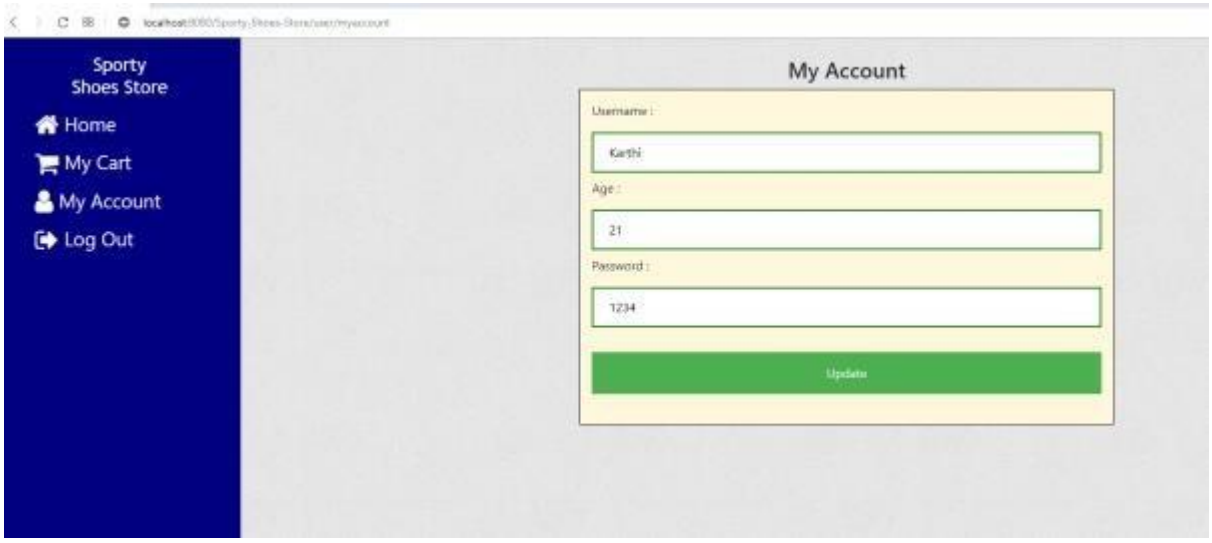
2. Users home page:



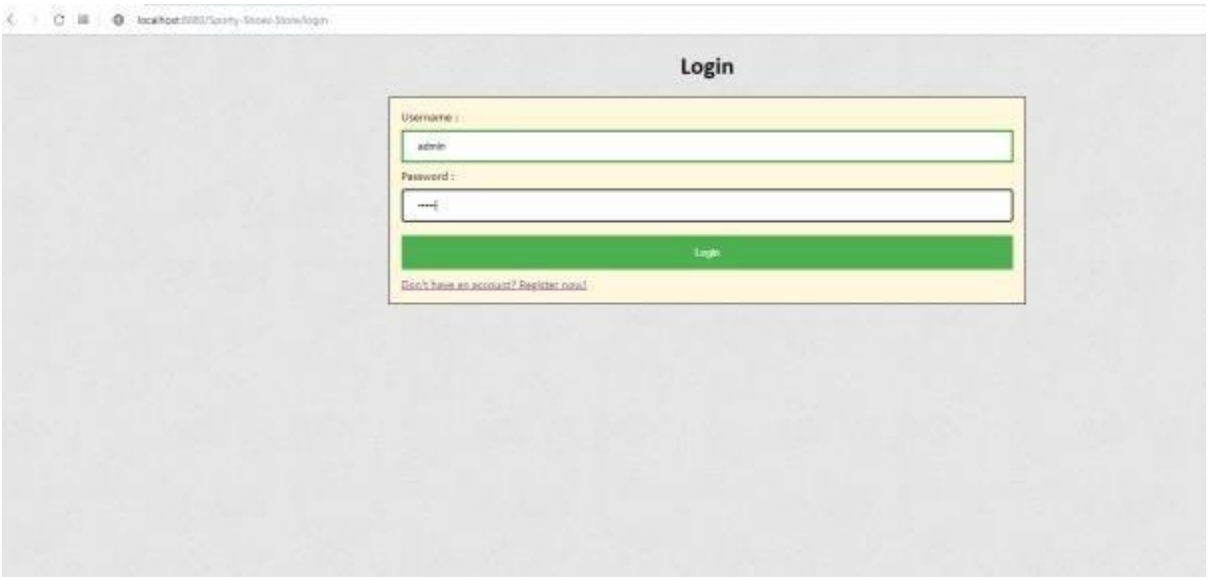
3. Users cart page:



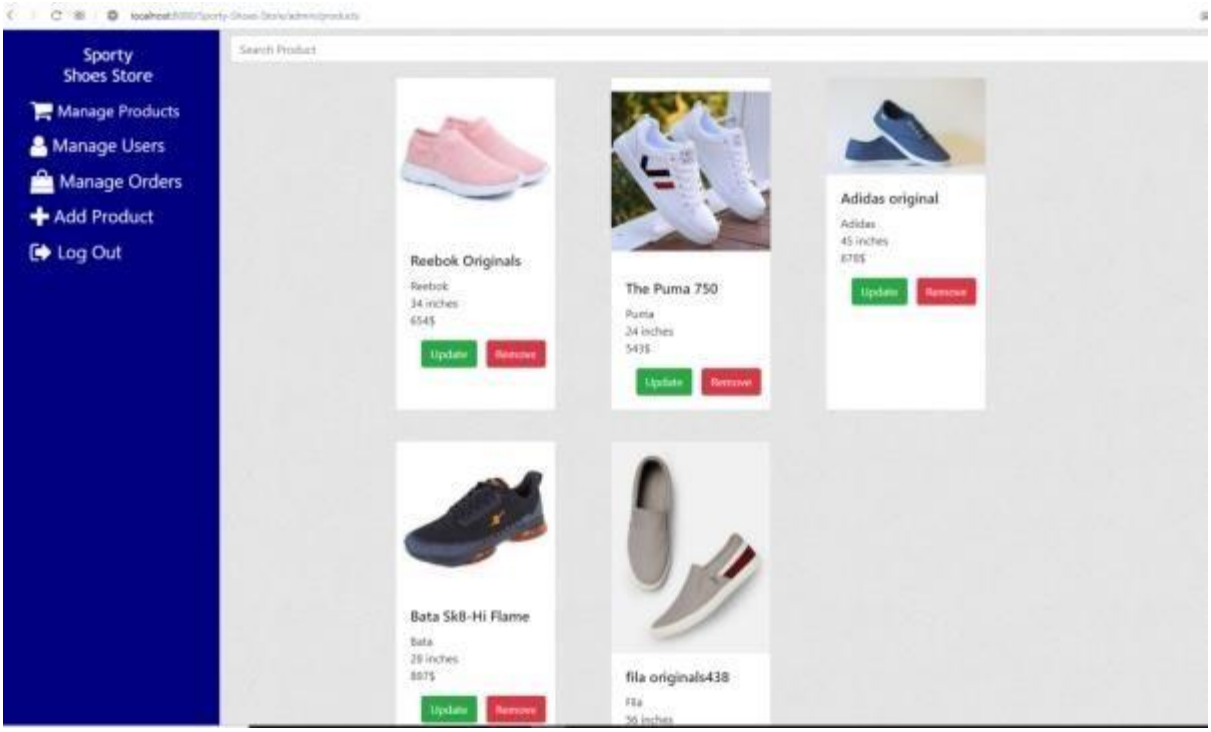
4. Users update account page:



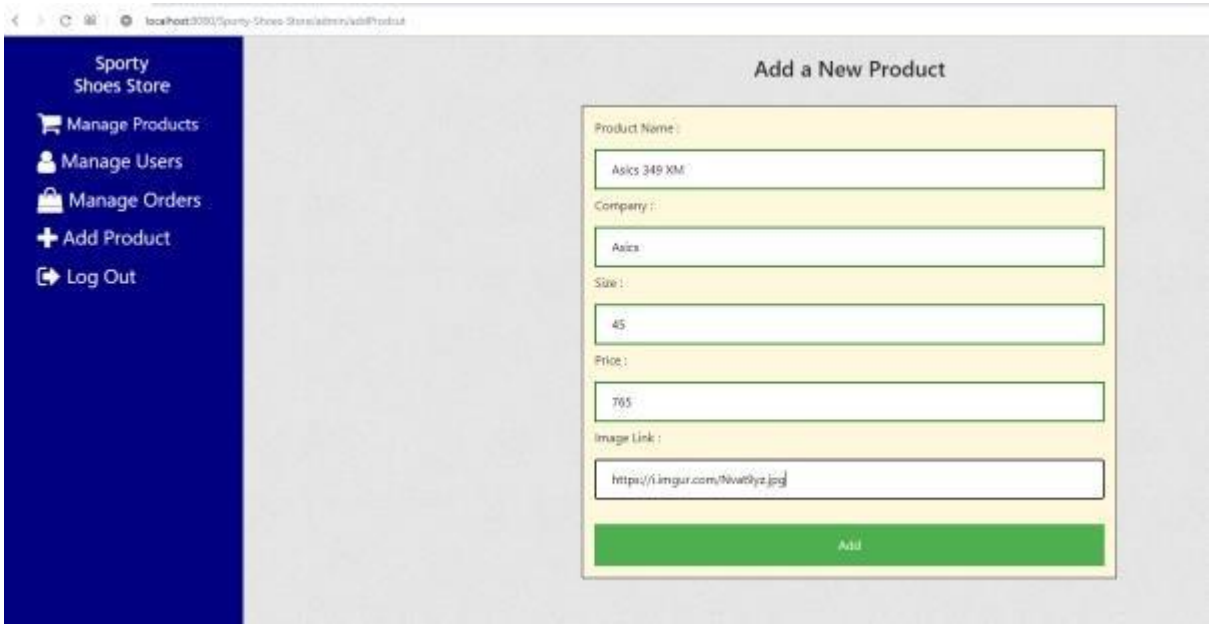
5. **Login as admin:**



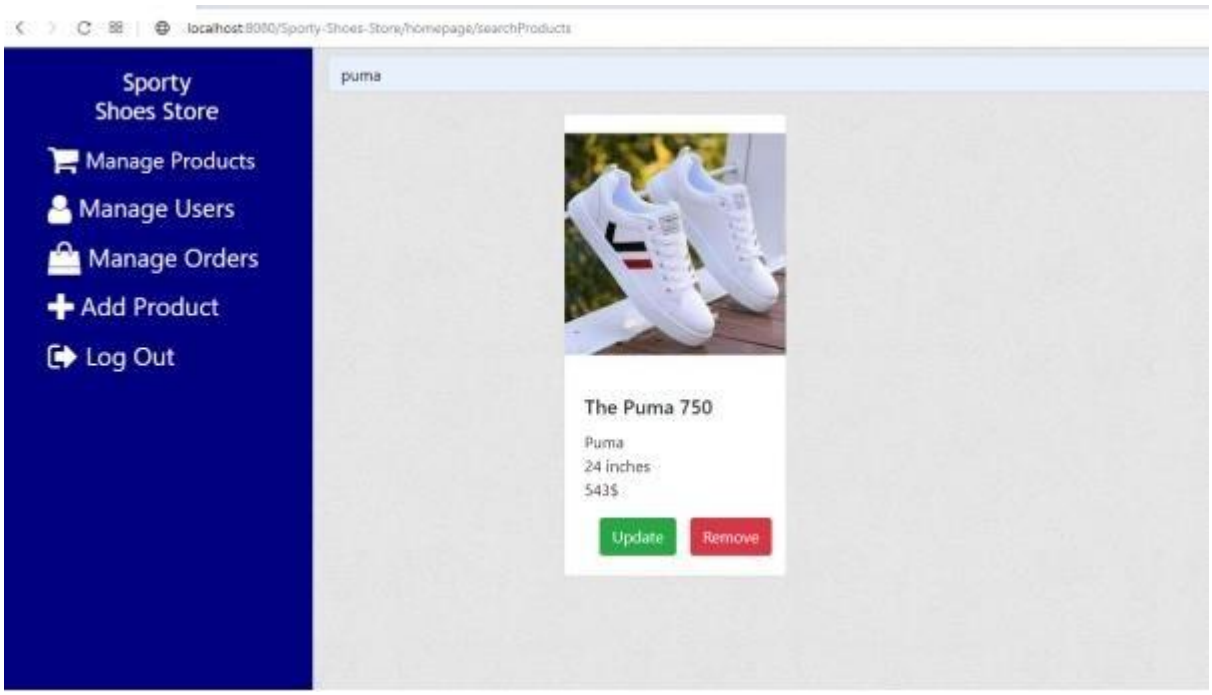
6. **Admins manage product page:**



7. Update products page:



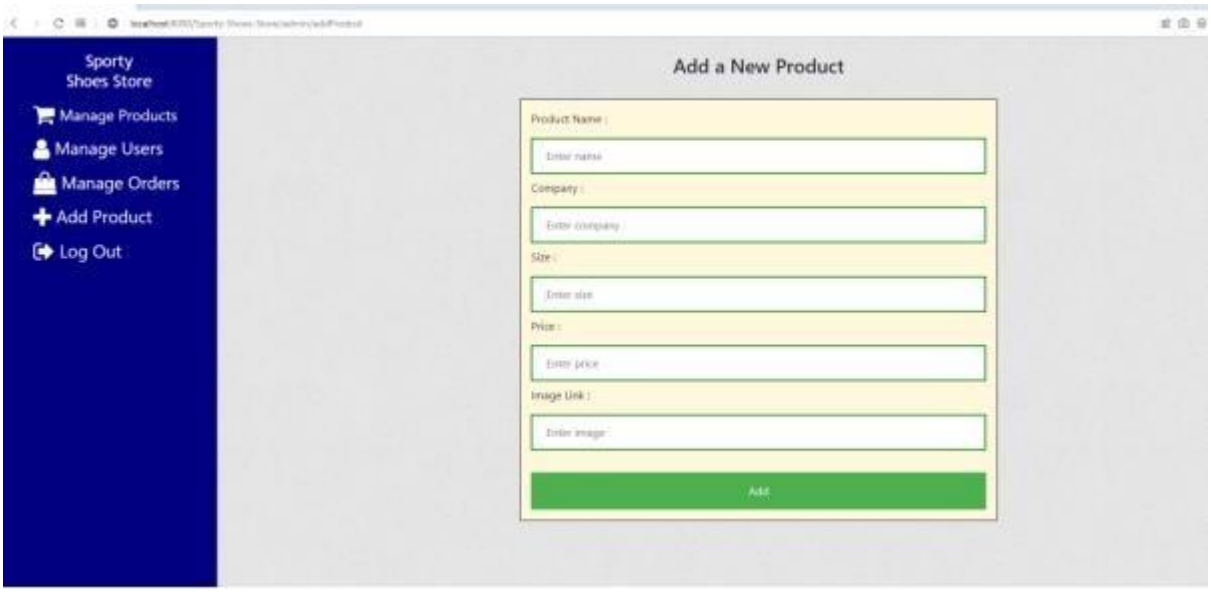
8. Search a product by name :



9. Manage users page:

10. Manage orders:

11. Add a new products to the system.



Step 4: Pushing the code to GitHub repository

- Open your command prompt and navigate to the folder where you have created your files.

  cd <folder path>

- Initialize repository using the following command:

  git init

- Add all the files to your git repository using the following command:

  git add .

- Commit the changes using the following command:

  git commit . -m <commit message>

- Push the files to the folder you initially created using the following command:

  git push -u origin master

Unique Selling Points of the Application

- Admin can manage the products in the store including categorizing them.
- Admin can browse the list of users who have signed up and be able to search users.
- Admin can be able to see purchase reports filtered by date and category.

Conclusions

In the program an application has been developed with a duration of two spirits. This application makes handling the data of the sport shoes company. All the data about the name, company, size, price, image link of the product are maintained. The admin can login through a User ID and password and manipulate the data.