**Shoe Ecommerce**

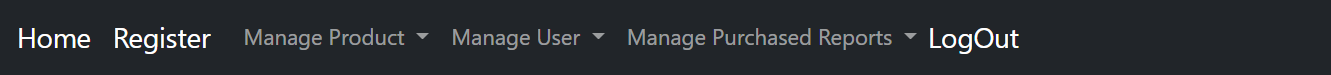
**Index:**

**Git URL:** [**https://github.com/RajaRajeswari12/Phase3ShoeEcommerce.git**](https://github.com/RajaRajeswari12/Phase3ShoeEcommerce.git)

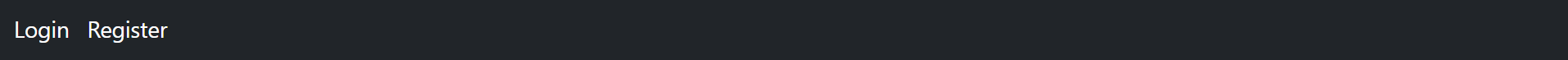
**Jsp Pages:**

1. header.html
2. addProduct.html
3. adminLogin.html
4. adminRegistration.html
5. deleteProduct.html
6. displayUserDetail.html
7. homePage.html
8. searchByUser.html
9. searchPurchaseReportsByCategory.html
10. searchPurchaseReportsByDate.html
11. updateProduct.html
12. updateProductPage.html
13. viewCustomer.html
14. viewProduct.html
15. viewPurchaseReport.html
16. welcome.html
17. **Servlets [com.sportyshoes.ecommerce.controller]**
    1. AdminUserController.java
    2. CustomerController.java
    3. ProductCategoryController.java
    4. ProductController.java
    5. PurchaseReportController.java
18. **Repositories [com.sportyshoes.ecommerce.repository]**
    1. AdminUserRepository.java
    2. CustomerRepository.java
    3. ProductCategoryRepository.java
    4. ProductRepository.java
    5. PurchaseReportRepository.java
19. **Models[com.sportyshoes.ecommerce.models]**
    1. Airline.java
    2. AdminUser.java
    3. Customer.java
    4. Product.java
    5. ProductCategory.java
    6. PurchaseReport.java
20. **Service[com.sportyshoes.ecommerce.service]**
    1. AdminUserService.java
    2. CustomerService.java
    3. ProductCategoryService.java
    4. ProductService.java
    5. PurchasedReportService.java
21. **Service Implementation [com.sportyshoes.ecommerce.service.implementation]**
    1. AdminUserServiceImpl.java
    2. CustomerServiceImpl.java
    3. ProductCategoryServiceImpl.java
    4. ProductServiceImpl.java
    5. PurchasedReportServiceImpl.java
22. **POJOS[com.sportyshoes.ecommerce.POJO]**
    1. CustomerNamePOJO.java
    2. GetProductIdPOJO.java
    3. PurchaseReportByDatePOJO.java
23. **Validators[com.sportyshoes.ecommerce.validator]**
    1. AdminUserValidator.java
24. Jsp Pages
25. **header.html**

Header Html contains the link to all the pages. It checks whether the user has logged in or not. If the Admin logs in then it allows access to the following header links.

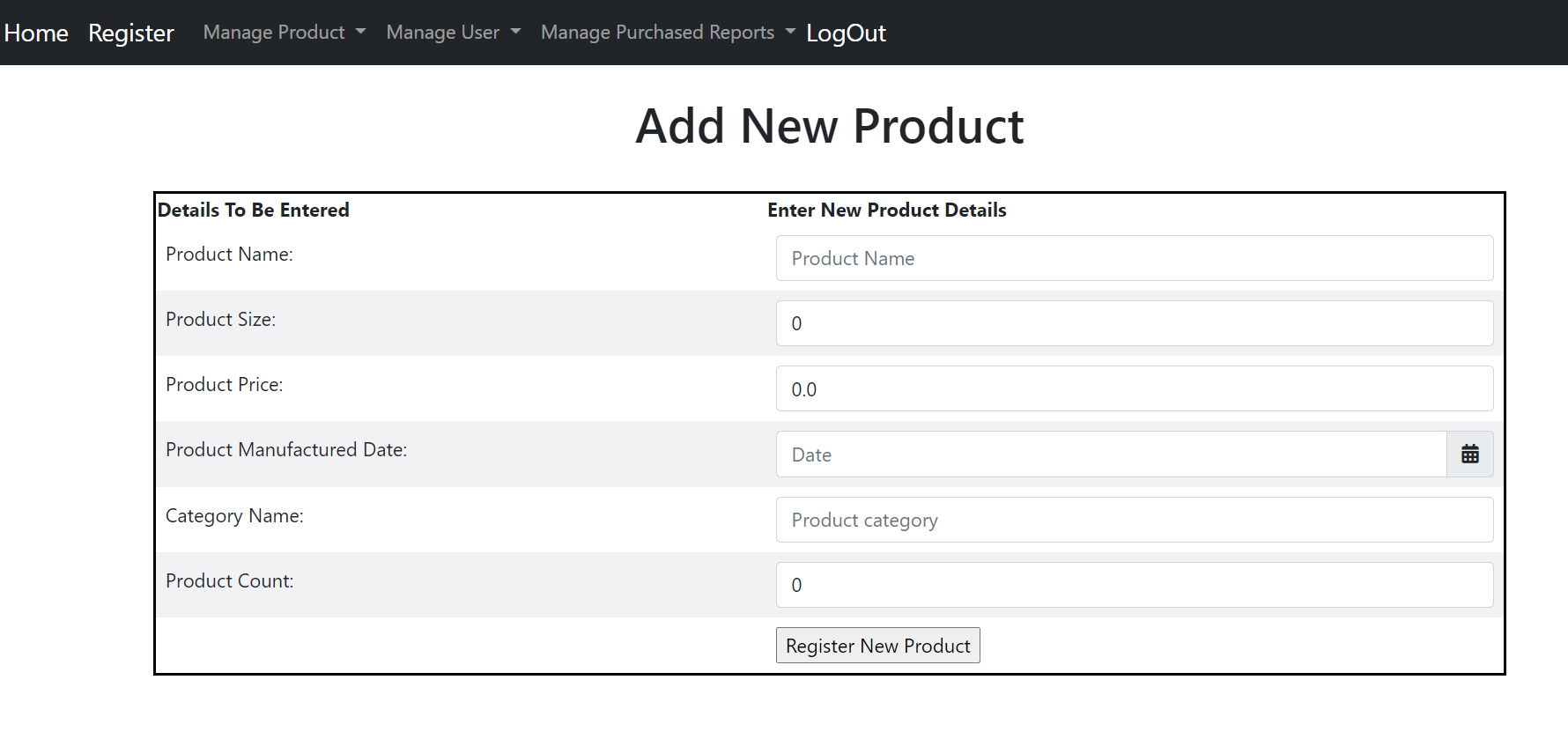
****

Or else only the below header links are active.

****

1. **addProduct.html**

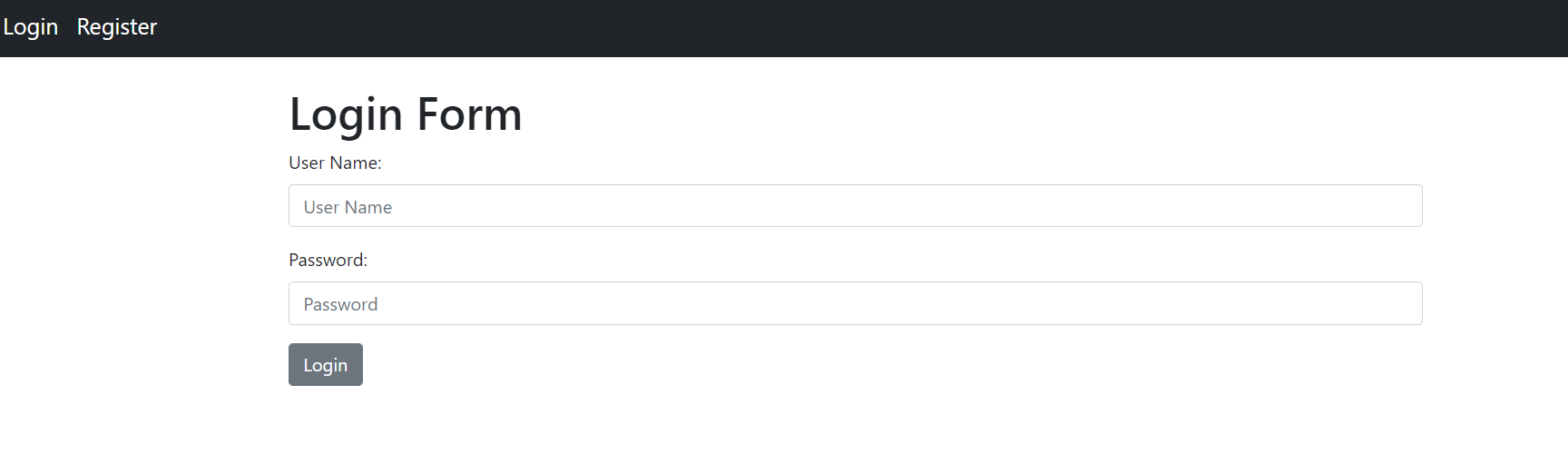
Add Product Page is used to add new product details along with Category to the backend. Once successfully inserted the data without any error. The page navigates to the viewProduct page where all the products are displayed.

****

1. **adminLogin.html**

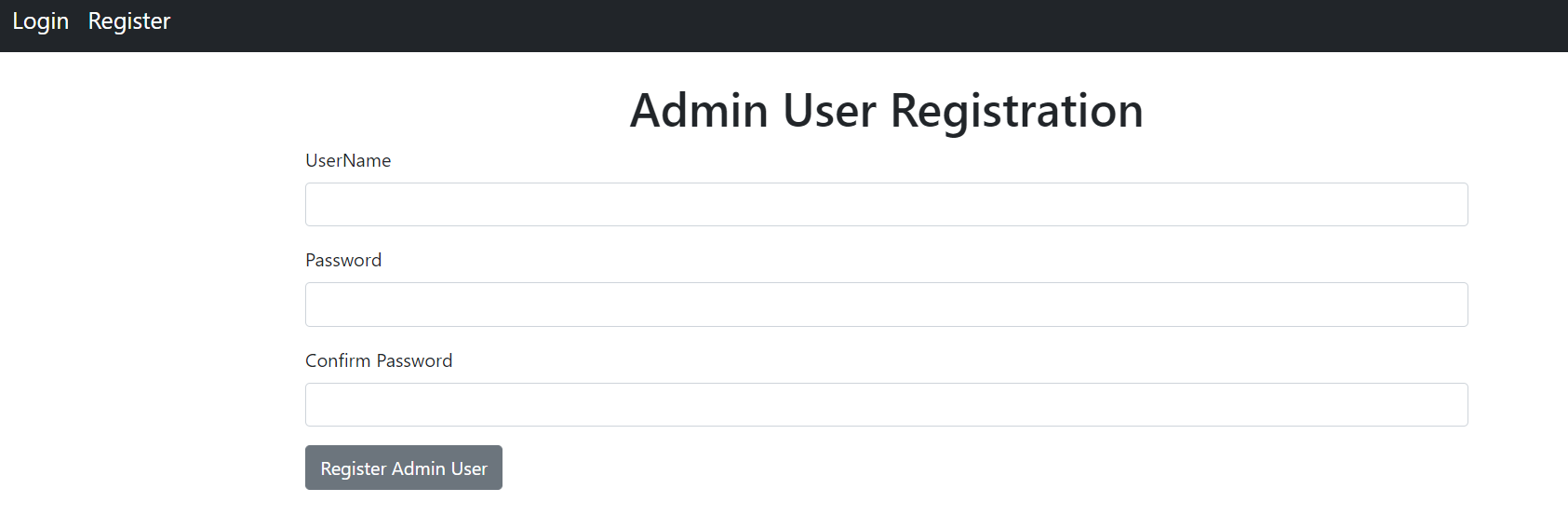
Login Page is used by the Admin to Login into shoe’s E-commerce portal.

If the Admin logins with his/her credentials then it is navigated to the Admin page where the admin can manage the products, view the list of customers who have signed up and see their details. Also view the purchase reports.

****

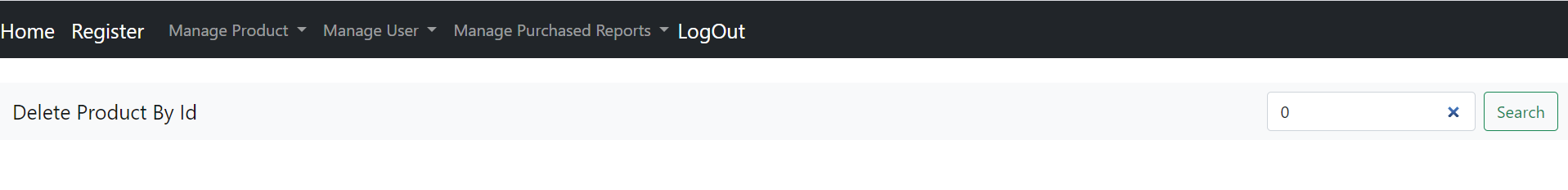
1. **adminRegistration.html**

Admin Registration page is used to register new user as Admin. And the Admin details are stored on the admin\_user.



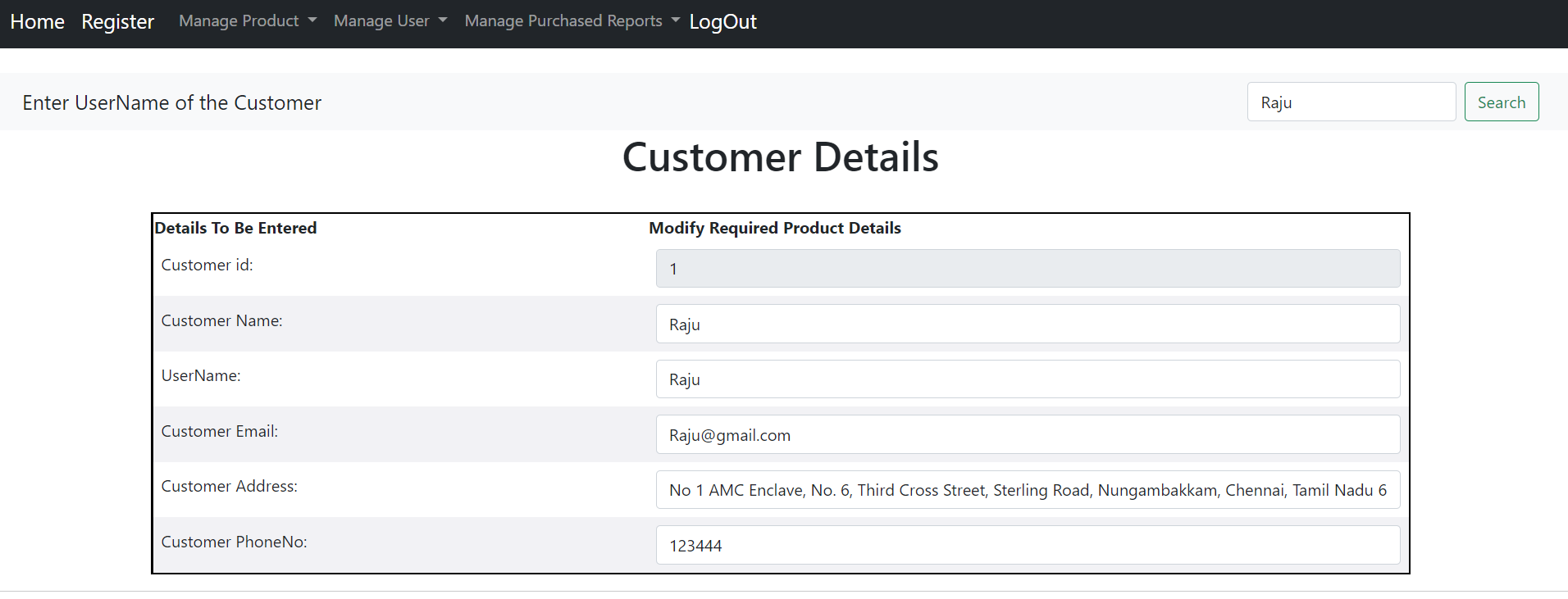
1. **deleteProduct.html**

Delete Product page is used to delete the product details by using the Product Id.

****

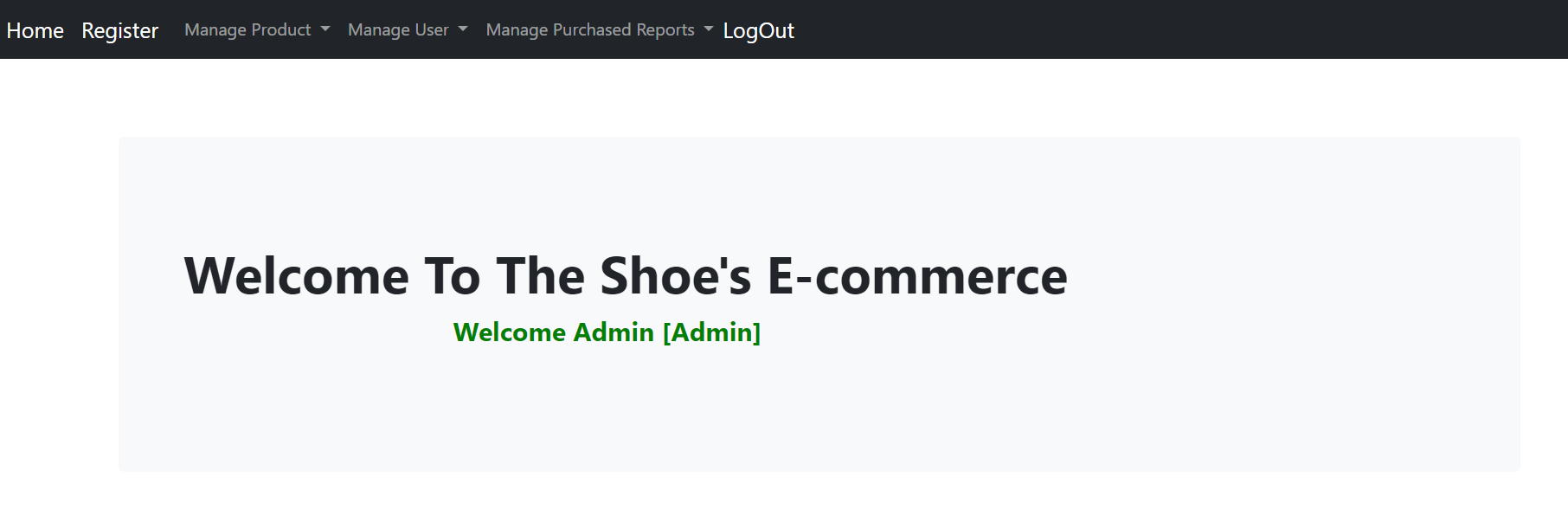
1. **displayUserDetail.html**

DisplayUserDetail page is used to display the details of the Customer by giving the customers username.



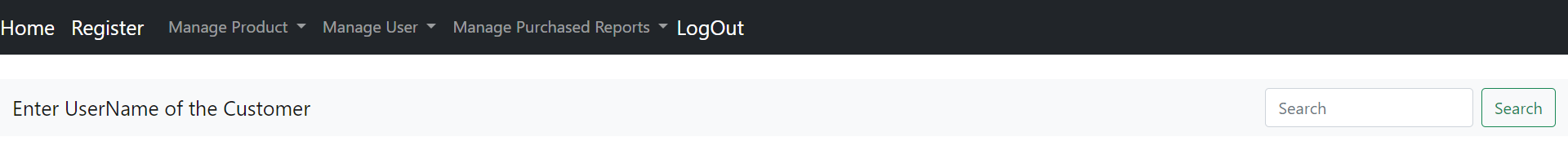
1. **homePage.html**

Home Page is the first page comes when you access the link <http://localhost:8081/home>



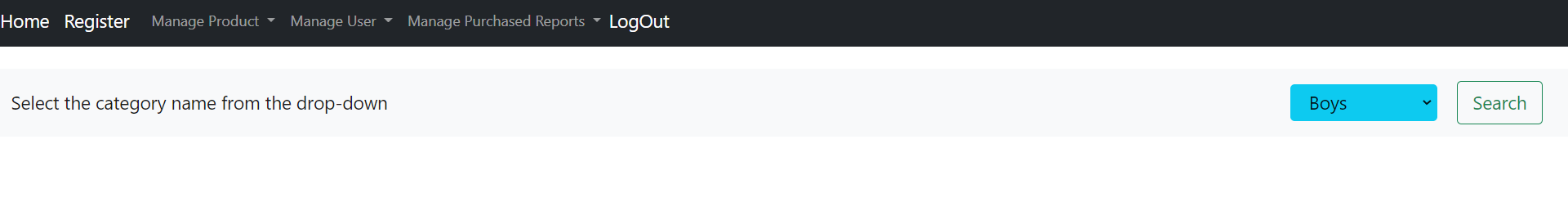
1. **searchByUser.html**

This Page is used to search the customer by giving username.

****

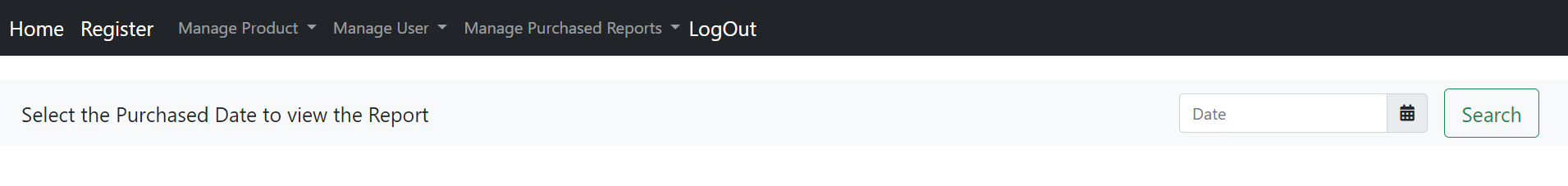
1. **searchPurchaseReportsByCategory.html**

This page is used to get all the Purchase Reports by choosing the shoe’s category from the drop down list.

****

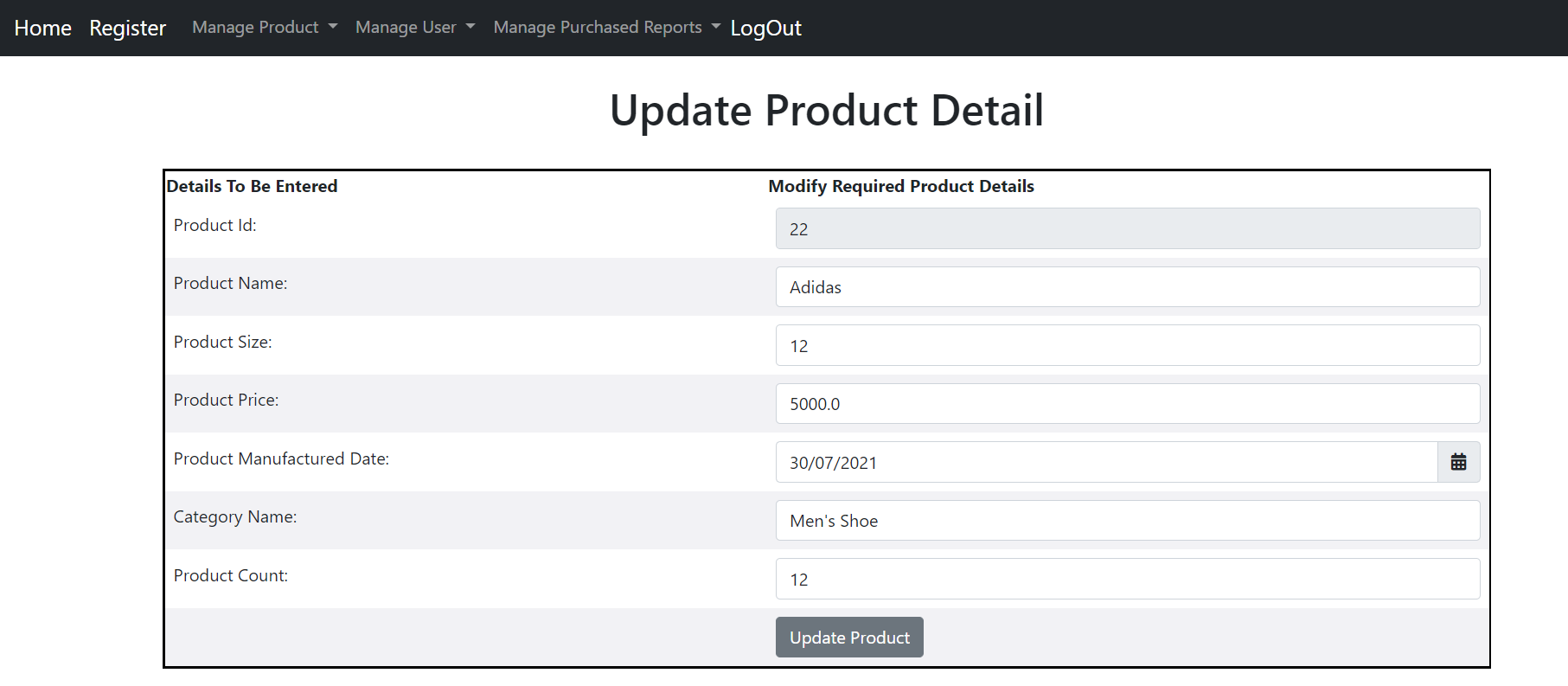
1. **searchPurchaseReportsByDate.html**

This page is used to get all the Purchase Reports by selecting the purchased Date.

****

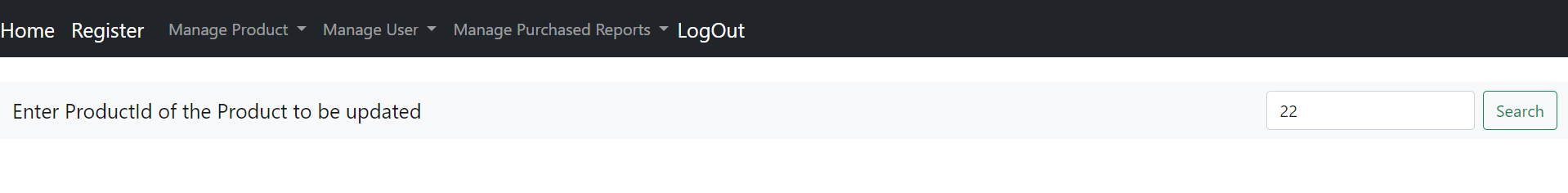
1. **updateProduct.html**

This Page is used to update the details of the added products.

****

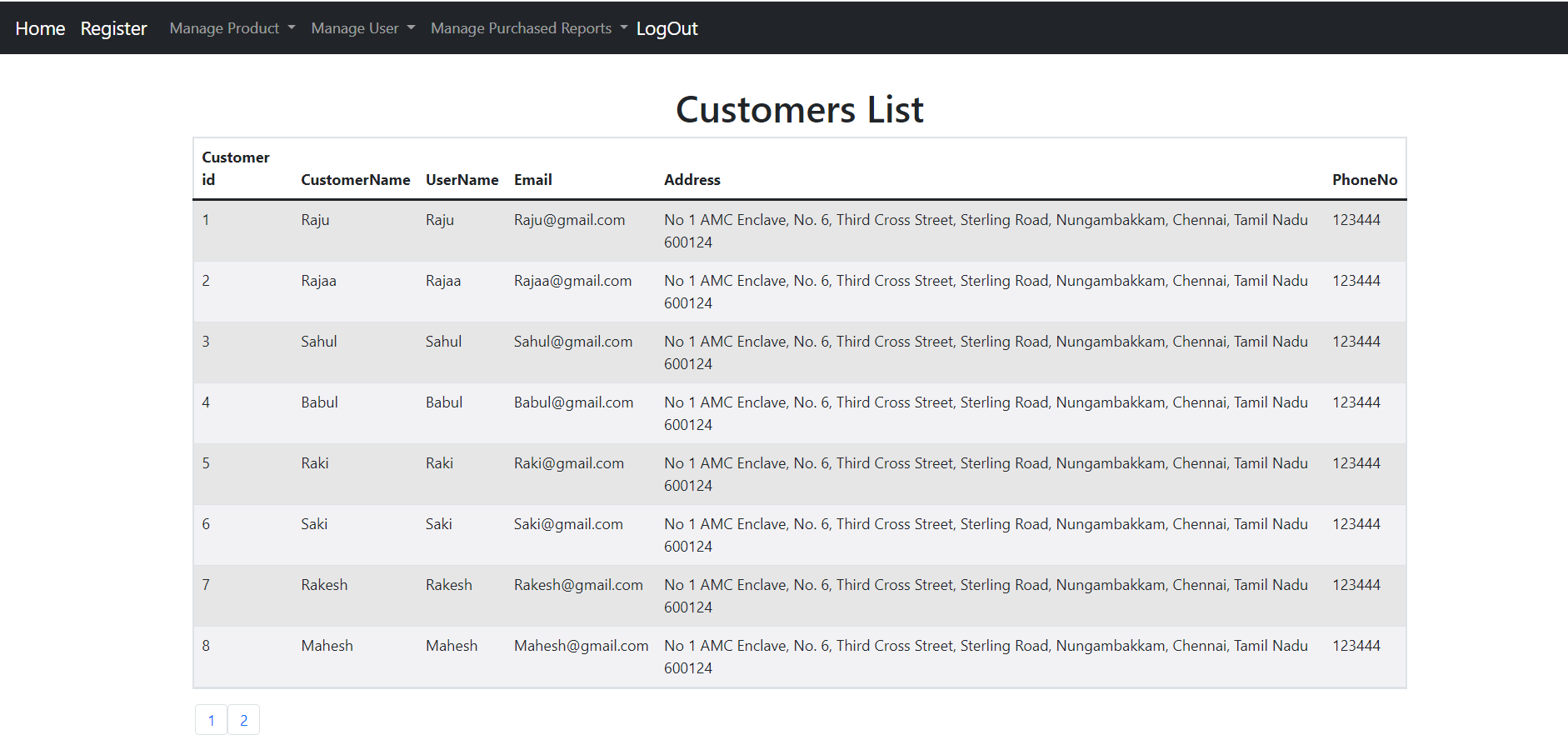
1. **updateProductPage.html**

Delete Product page is used to delete the product details by using the Product Id.

****

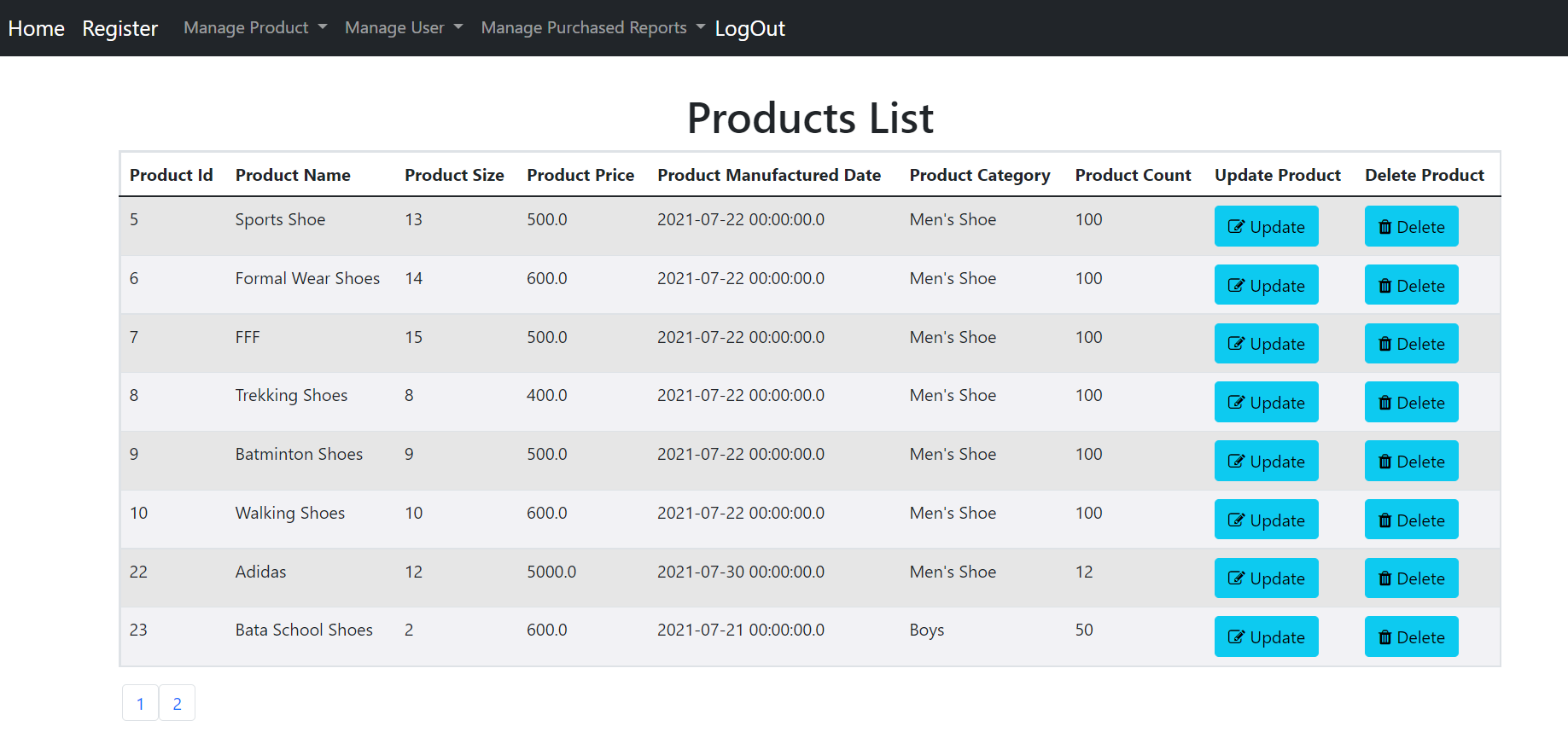
1. **viewCustomer.html**

List all the customers details who have signed up.



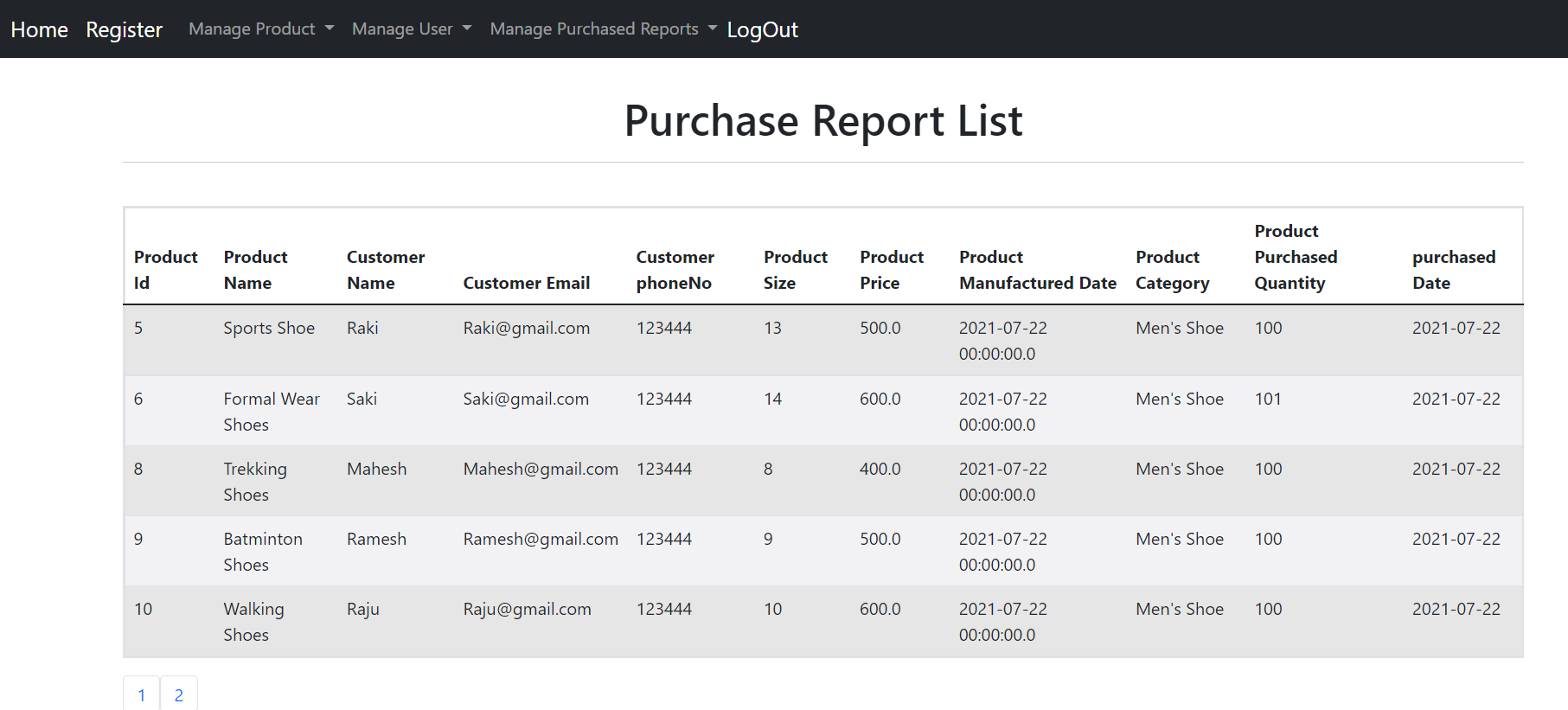
1. **viewProduct.html**

List all the Products added.

****

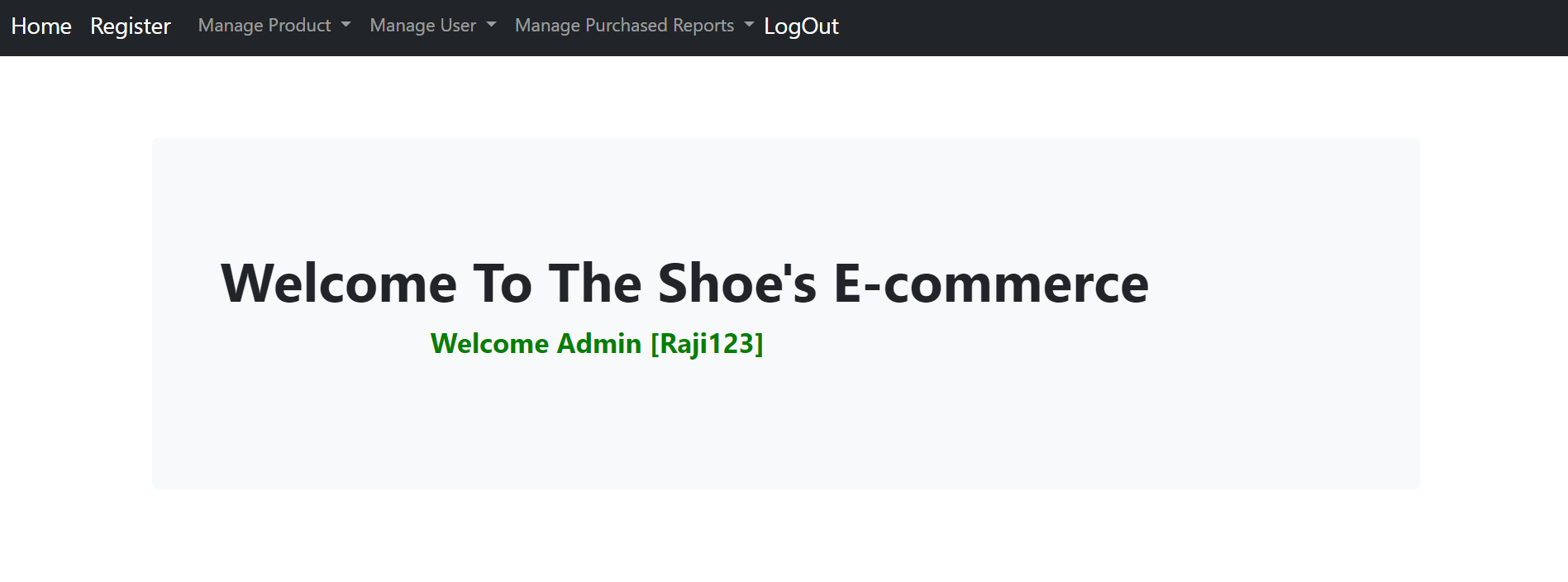
1. **viewPurchaseReport.html**

List all the details of the purchase made through this website.



1. **welcome.html**

This page is displayed, when the admin registers successfully.

****

1. **Controller[com.sportyshoes.ecommerce.controller]**
2. **AdminUserController.java**

AdminUserController receives the admin login, logout, register and home page URLs and takes the required actions.

1. **CustomerController.java**

CustomerController receives the list all customer URL and searchByUserName URL, then takes the required actions and navigate it to the appropriate pages.

1. **ProductController.java**

ProductController receives the URLs for list all product, SaveProduct, DeleteProduct, UpdateProduct , then takes the required actions and navigate it to the appropriate pages.

1. **PurchaseReportController.java**

PurchaseReportController receives the list all PurchaseReport URL, the list all PurchaseReport By Category URL, and the list all PurchaseReport By Date URL, then takes the required actions and navigate it to the appropriate pages.

1. **Repositories [com.sportyshoes.ecommerce.repository]**
2. AdminUserRepository.java
3. CustomerRepository.java
4. ProductCategoryRepository.java
5. ProductRepository.java
6. PurchaseReportRepository.java
7. **Models[com.sportyshoes.ecommerce.models]**
8. AdminUser.java
9. Customer.java
10. Product.java
11. ProductCategory.java
12. PurchaseReport.java
13. **Service[com.sportyshoes.ecommerce.service]**
14. AdminUserService.java

AdminUserService performs the backend operations for the admin login, logout, and register.

1. CustomerService.java

CustomerService performs the backend operations to retrieve the list of all customer and retrieve customer records by taking username as an input.

1. ProductService.java

ProductController performs the backend operations to SaveProduct, DeleteProduct, UpdateProduct and retrieves the list of all product.

1. PurchasedReportService.java

PurchaseReportController performs the backend operations to retrieve the list of all PurchaseReport, the list all PurchaseReport by category, and the list of all PurchaseReport by date.

1. **Service Implementation [com.sportyshoes.ecommerce.service.implementation]**
2. AdminUserServiceImpl.java
3. CustomerServiceImpl.java
4. ProductCategoryServiceImpl.java
5. ProductServiceImpl.java
6. PurchasedReportServiceImpl.java
7. **POJOS[com.sportyshoes.ecommerce.POJO]**
8. **CustomerNamePOJO.java**

This POJO is used to load the customer Name and page No on which that particular customer details listed.

1. **GetProductIdPOJO.java**

This POJO is used to load the Product ID and page No on which that particular product details are listed.

1. **PurchaseReportByDatePOJO.java**

This POJO is used to load the Purchased Date for which the purchase reports list are retrieved.

1. **Validators[com.sportyshoes.ecommerce.validator]**
2. **AdminUserValidator.java**

AdminUserValidator is used to validate the data entered by the Admin while registration.