SportyShoes.com

Kartikaya Srivastav

v (0.1)

1. **Introduction** 
   1. Purpose
   2. Requirement Overview

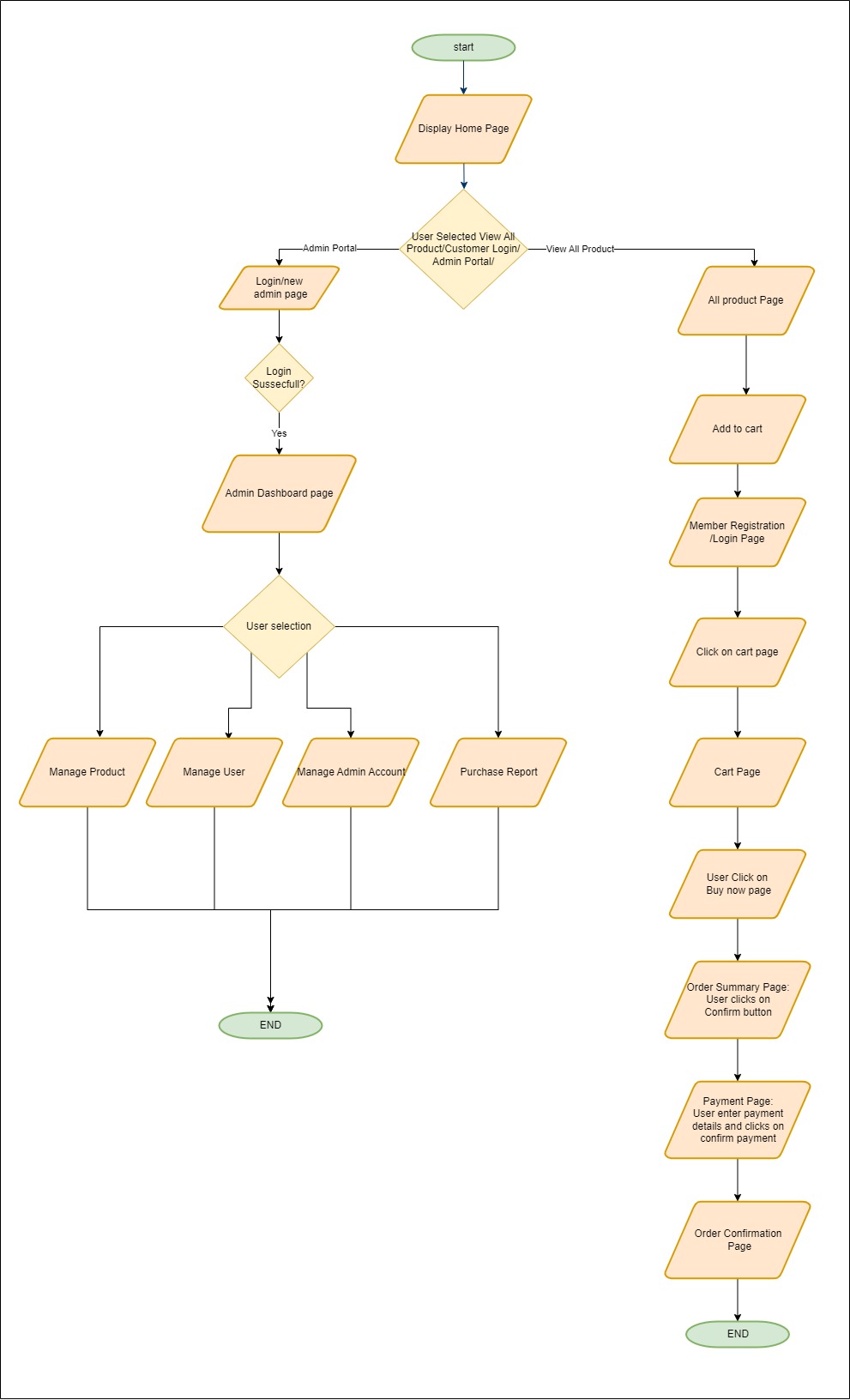
Sporty Shoes is a company that manufactures and sells sports shoes. They have a walk-in store, and now, they wish to launch their ecommerce portal sportyshoes.com.

**Code can be found at : https://github.com/ksrivastav/FlyawayWeb.git**

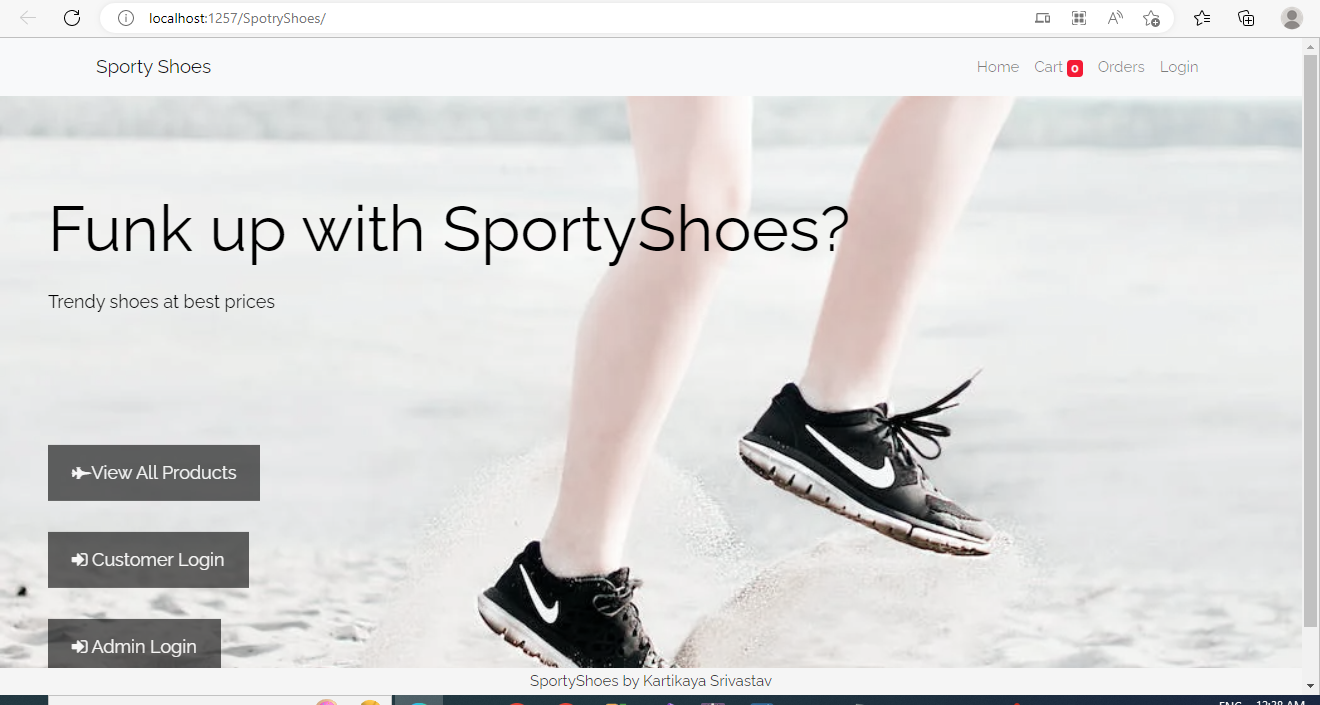
1. **Product Specifications.**
   1. **Product Specifications**

SportyShoes.com Website will have the following features.

* + 1. A homepage which will be listing all available products. Users will be able search for a particular product on the homepage.
    2. Users will be able to filter products based on specified criteria.
    3. User will be able to Order by clicking on the product and purchase it.
    4. There will be an admin page with below functionalities
       1. The admin should be able to change his password if he wants, he should be able to:
       2. Manage the products in the store including categorizing them  
          Browse the list of users who have signed up and be able to search users  
          See purchase reports filtered by date and category
  1. **Applicaiton Flow Diagram**

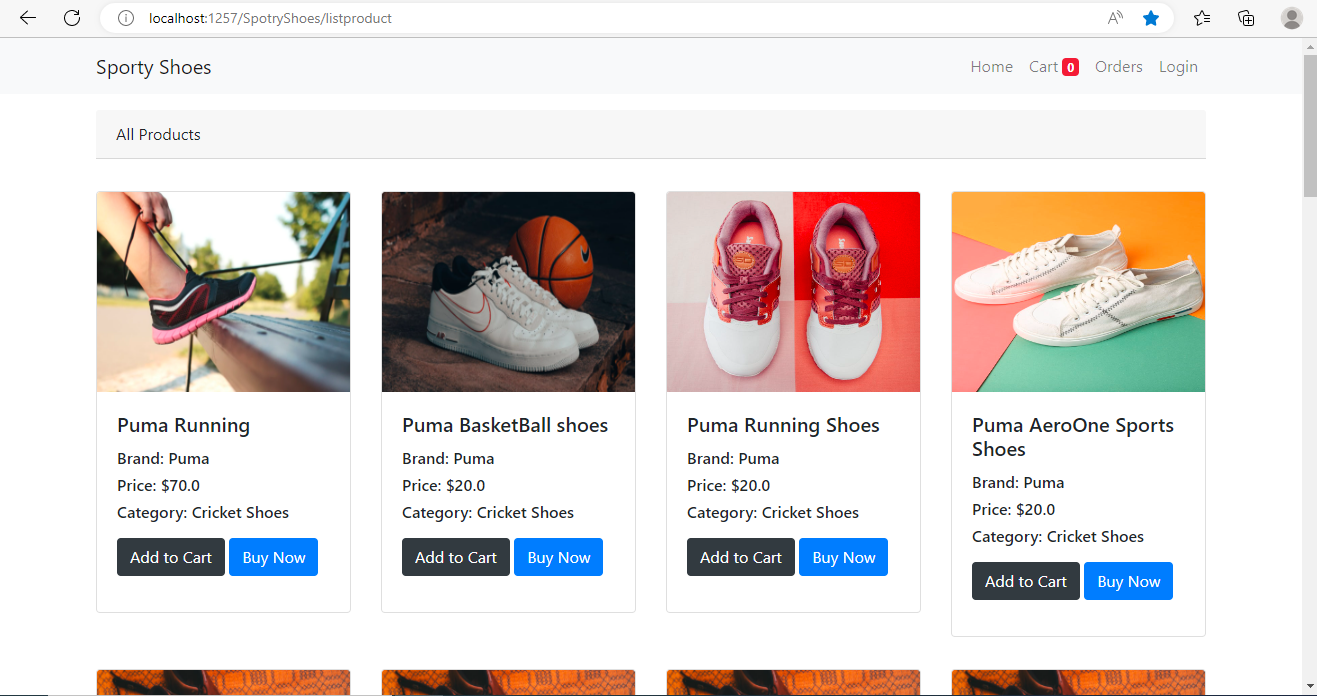


* 1. **User experience and user manual.**
     1. SportyShoes is Spring MVC based web application.
     2. When the Application starts, you are landed on the home page. Below is the screenshot of the home page.

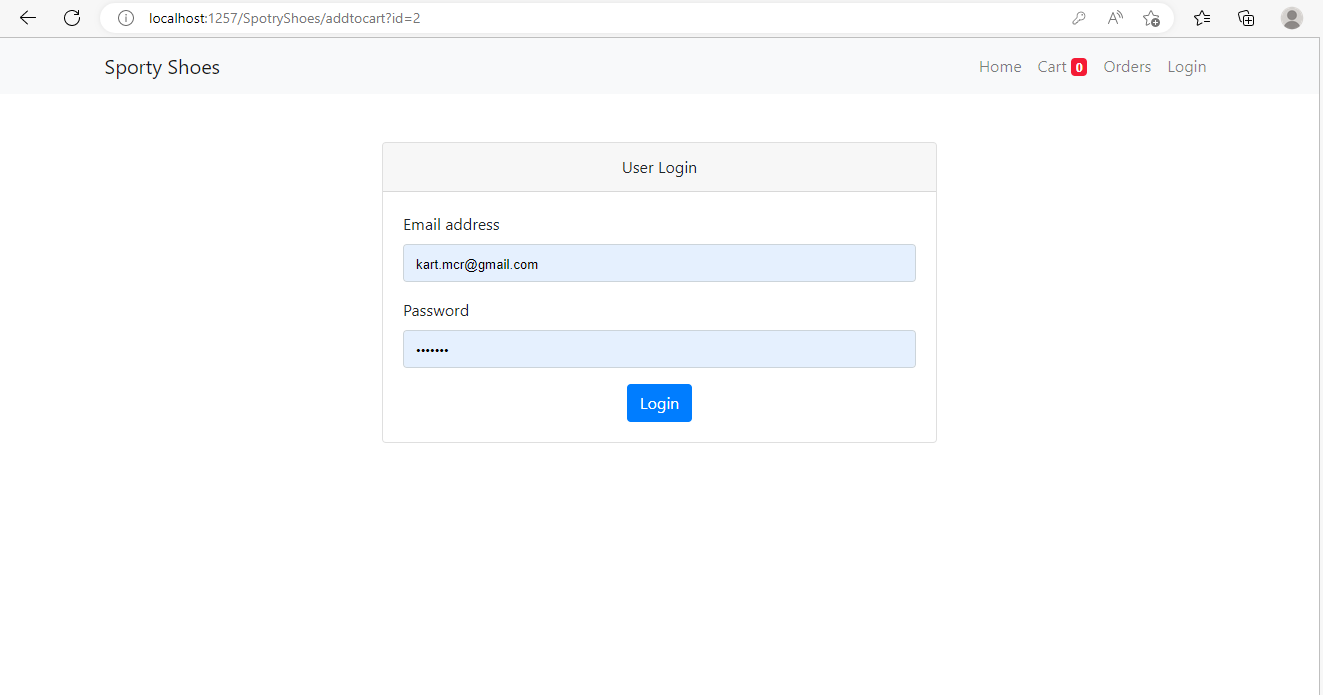


There are three menus in Flyaway web application. Users can choose from the menu simply by clicking on them.

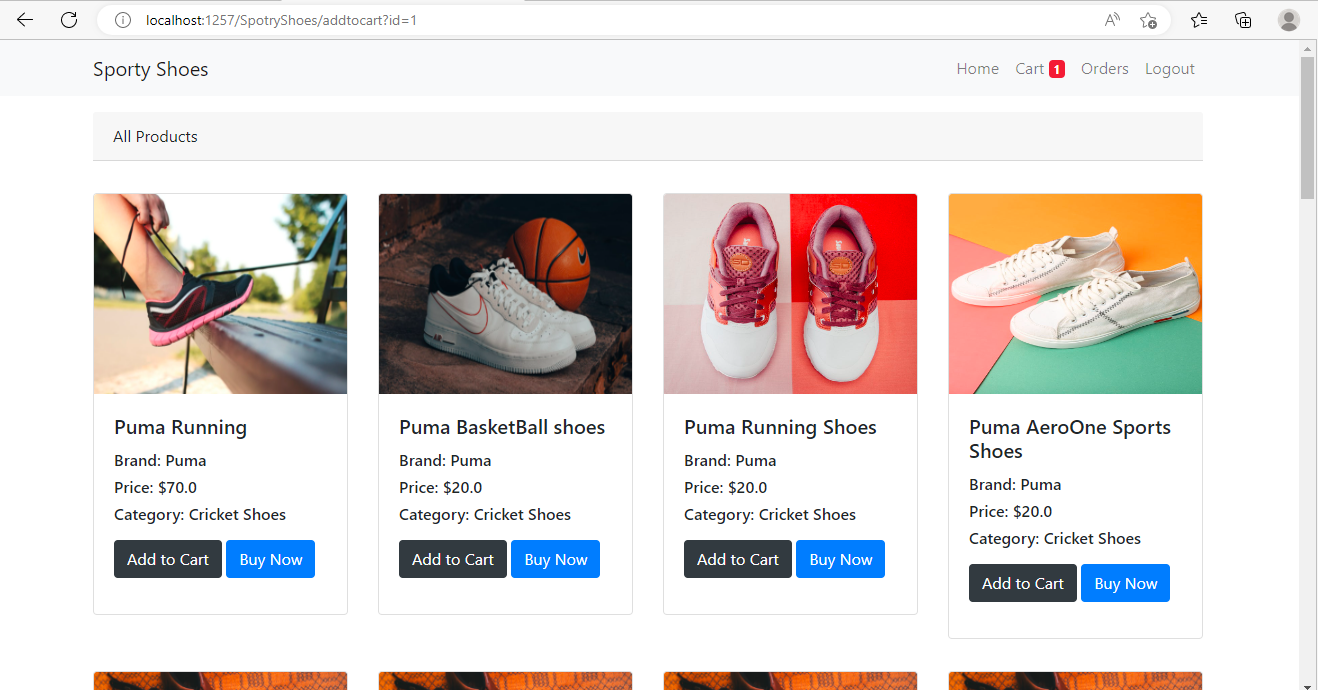
* + 1. **Working with All Product**
       1. After clocking on View All Product page from the main screen you will be landed on "View All Product” page. Here you can see list of all product available.



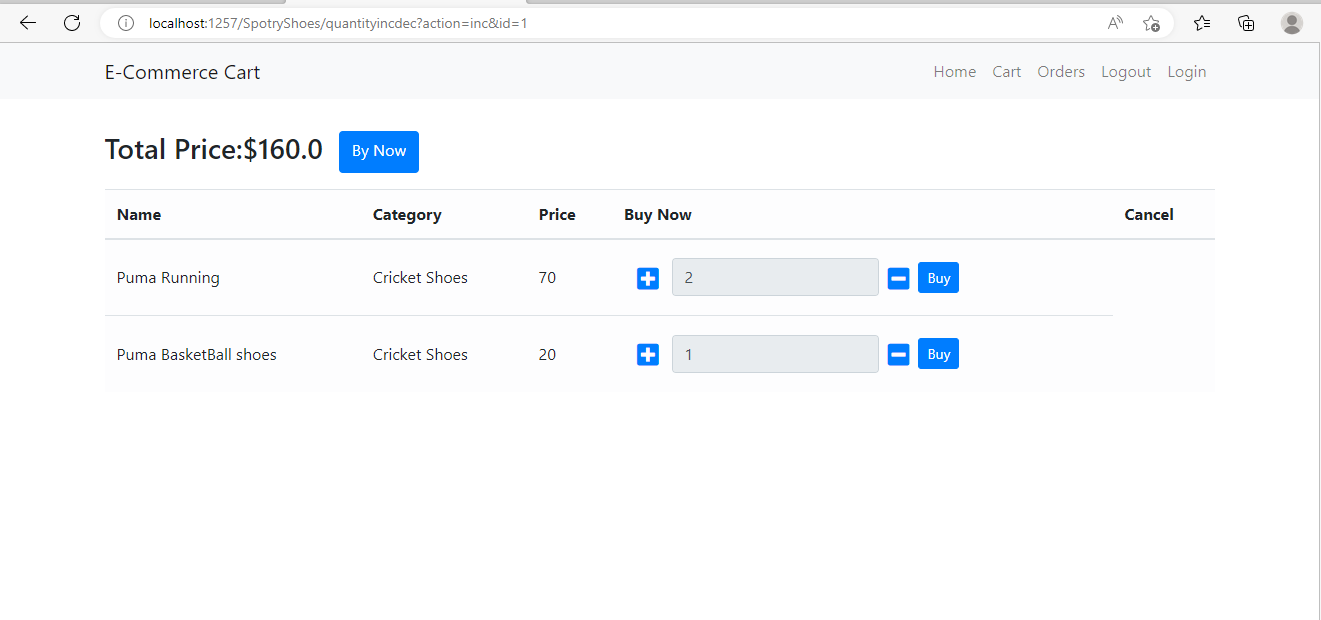
* + - 1. Click on “Add to Cart” on which ever product you want to purchase. You can add multiple products to cart.
      2. Whenever you click on add to cart, if you are not logged in already, you would be redirected to user login page.
      3. Log into



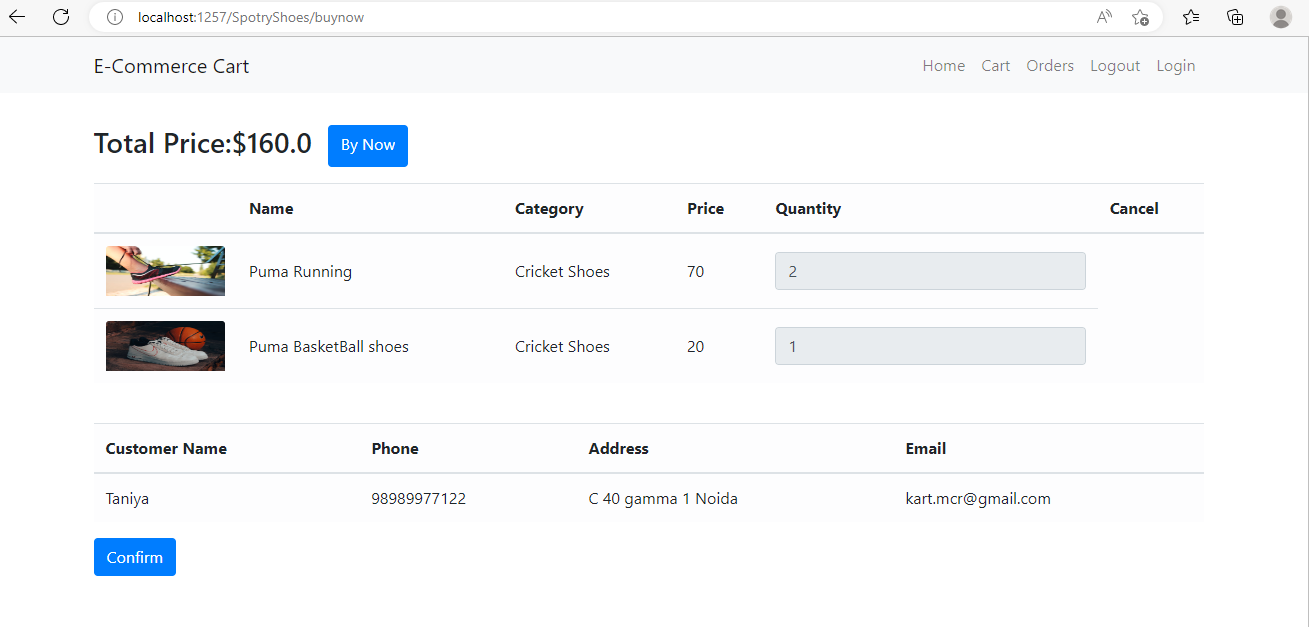
* + - * 1. Here you have to enter your sign in in details and you would be redirected to all product page after successfully sign in. You would land back in product page. If you add product to cart, you would be able to see count in red text updating with count (adjacent to cart).



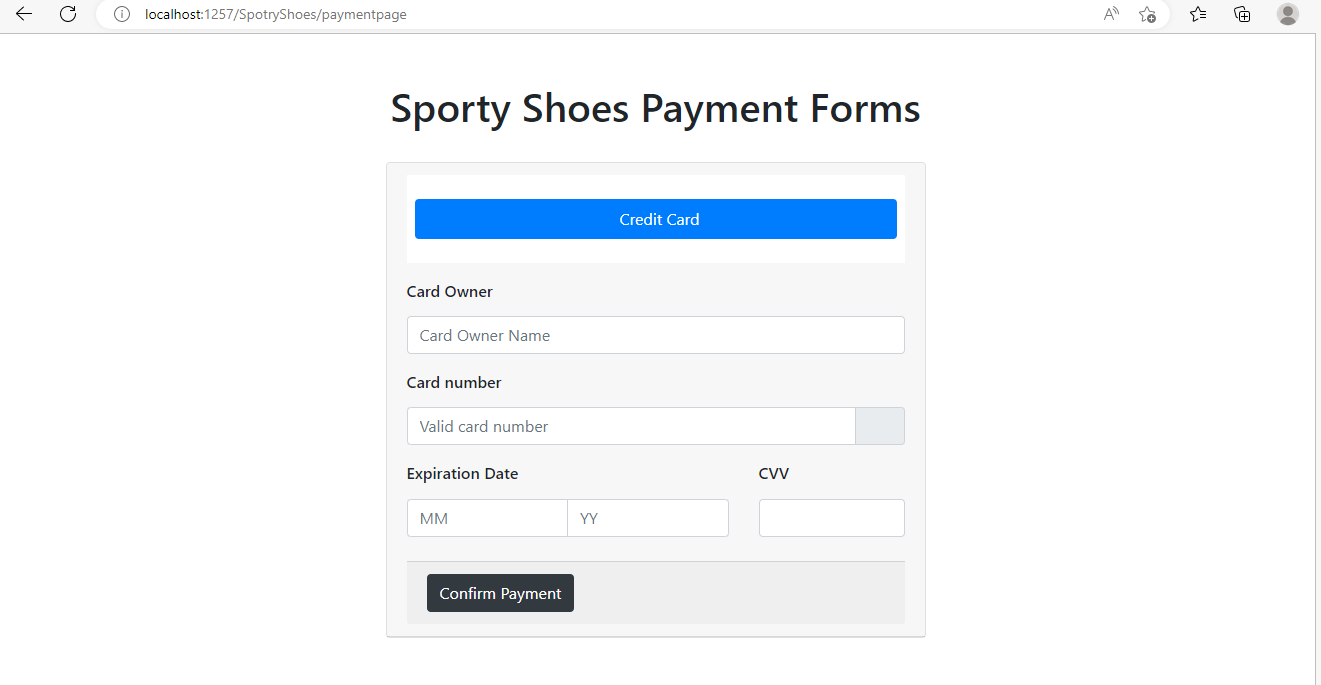
* + - * 1. Click on cart and you will be redirected to cart page. here you would be able to see the item you have added in cart along with quantities.



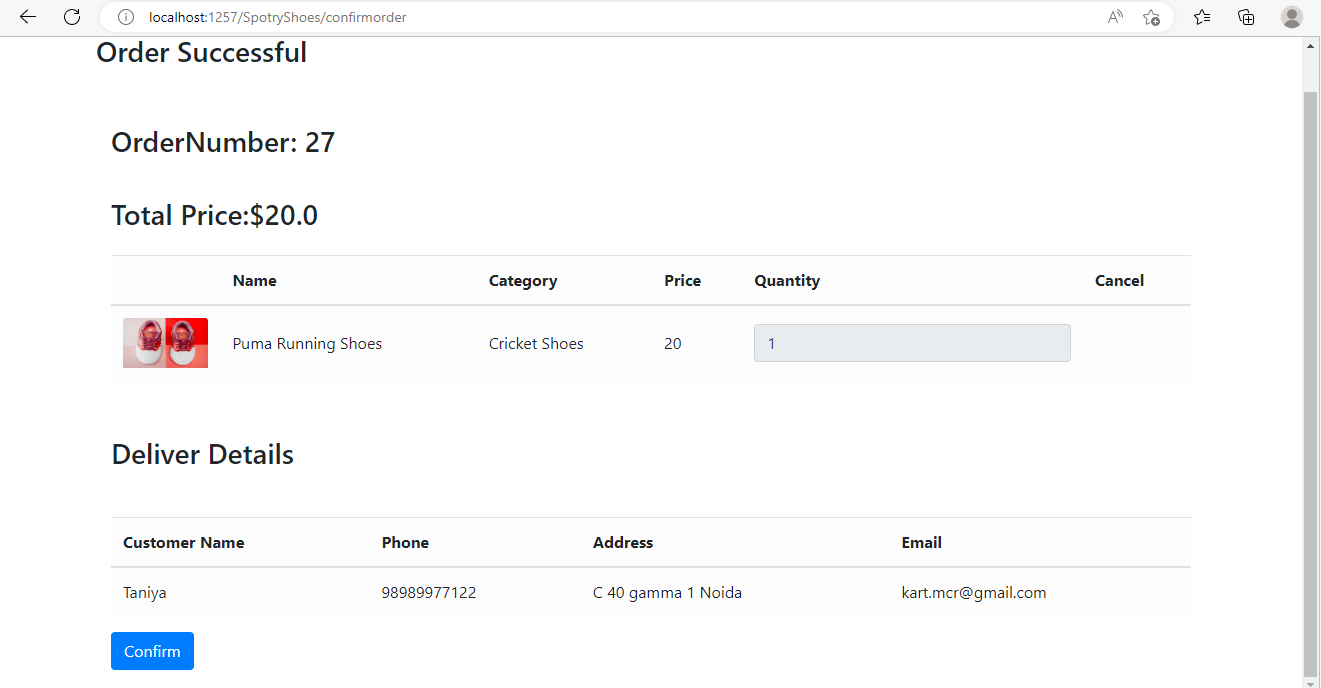
* + - * 1. Click on buy now item you would be redirected to order summary page.



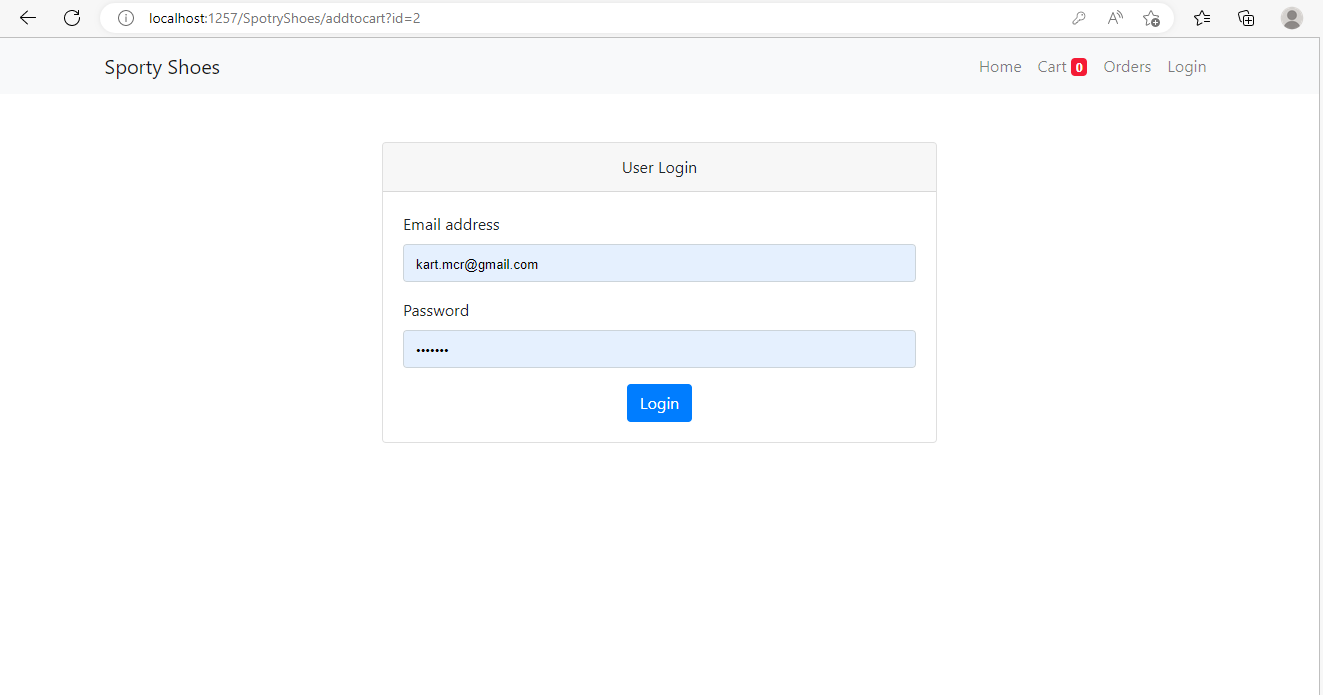
* + - 1. Once you click on confirm button, you are redirected to payment page. Here you can simply enter card details and. Click on Confirm Payment



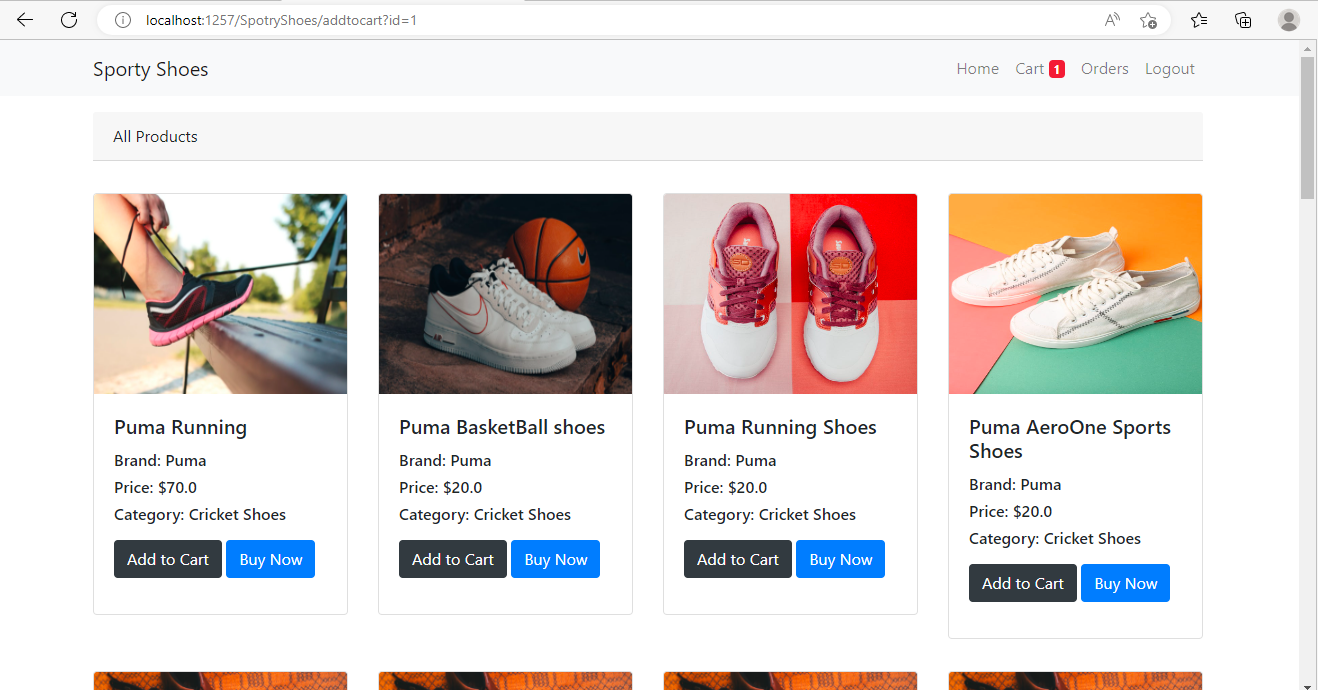
* + - 1. Once click on Confirm Payment, you will be redirected to order Confirmation page..



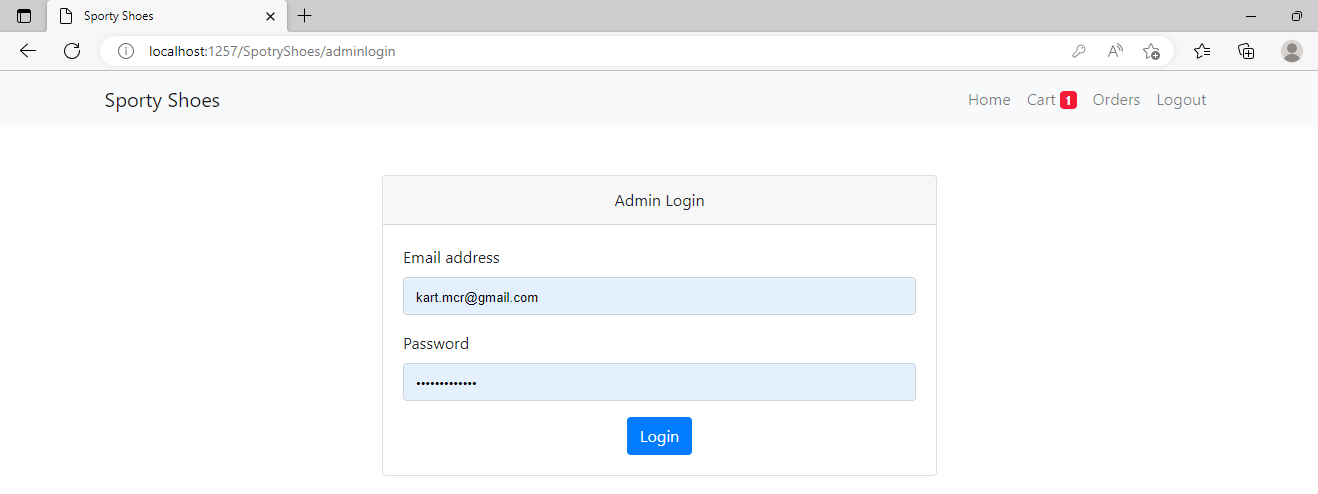
* + 1. Working with Customer Login.
       1. If you click on, you will be redirected to user log in page.

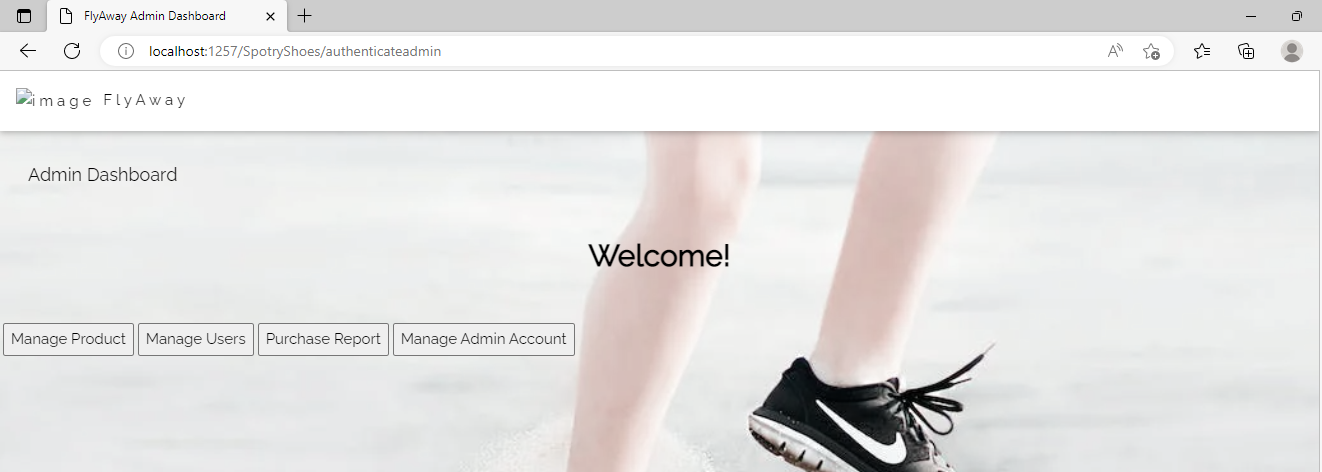


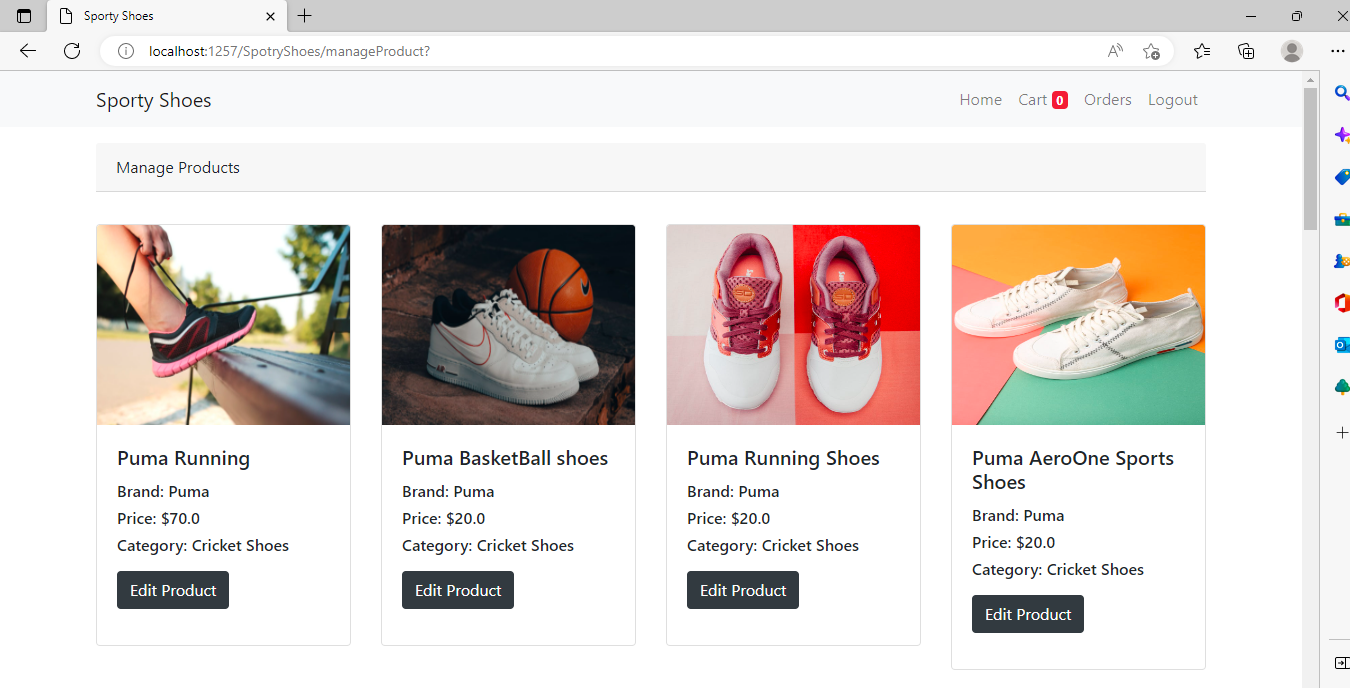
* + - 1. Here you have to enter your sign in in details and you would be redirected to all product page after successfully sign in. You would land back in product page.



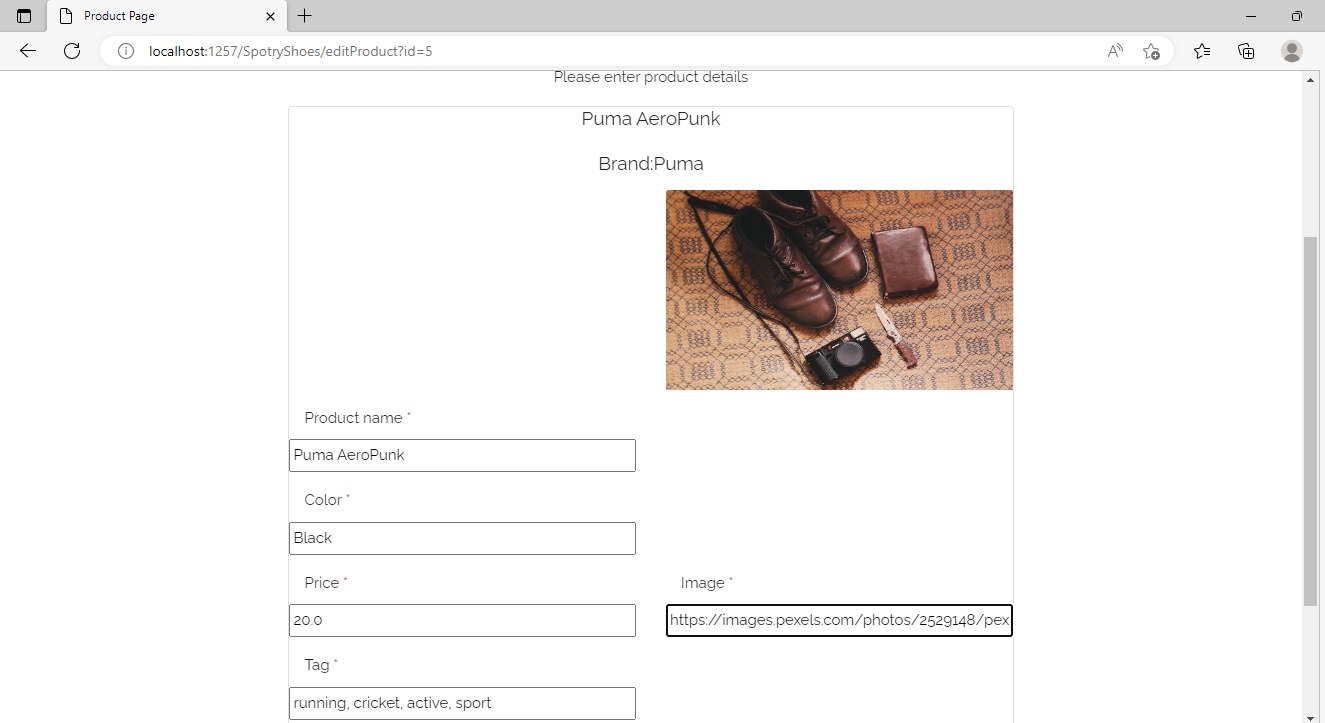
* + 1. Working with Admin Portal.
       1. If you clock on Admin Portal, you would be redirected to admin login



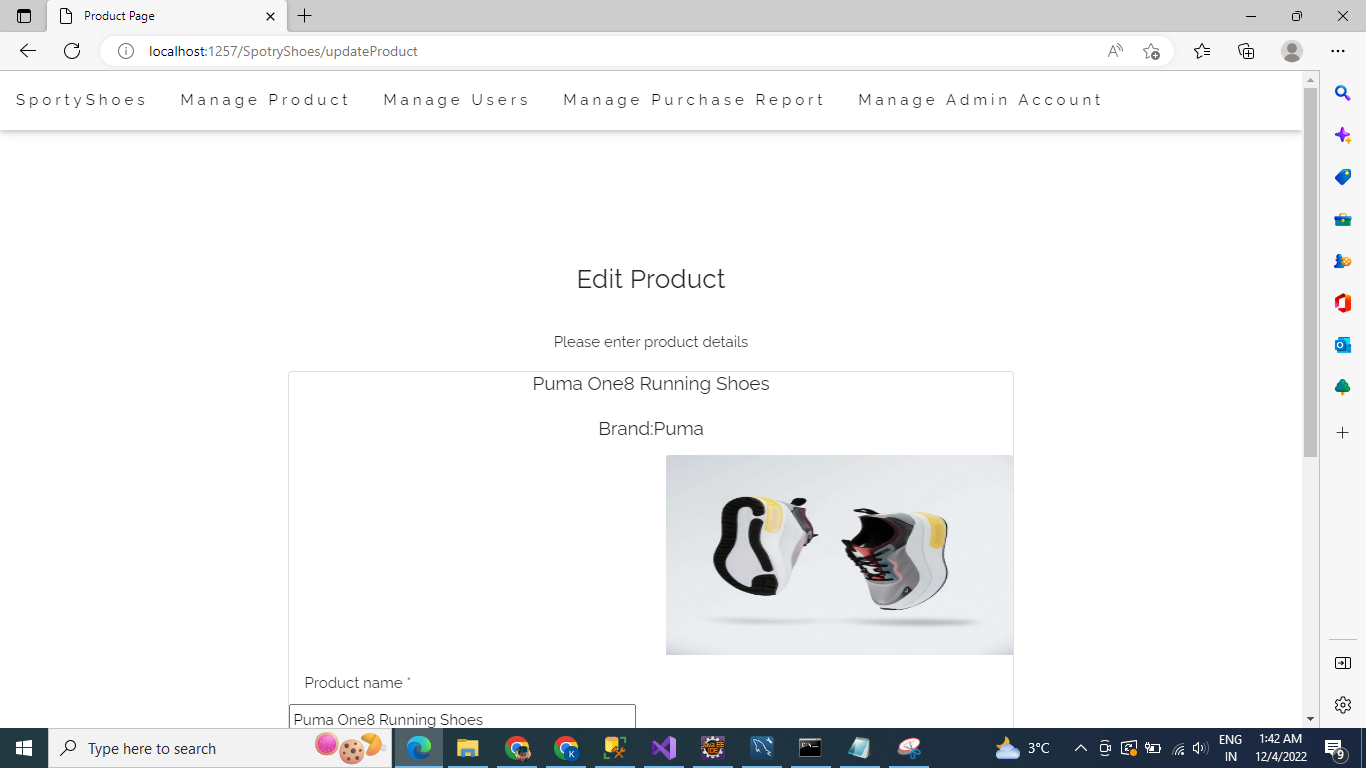
* + 1. Once you successfully log in, you are redirected to admin dashboard
    2. Working with Manage Product.
       1. You will be redirected to Manage Product Page.



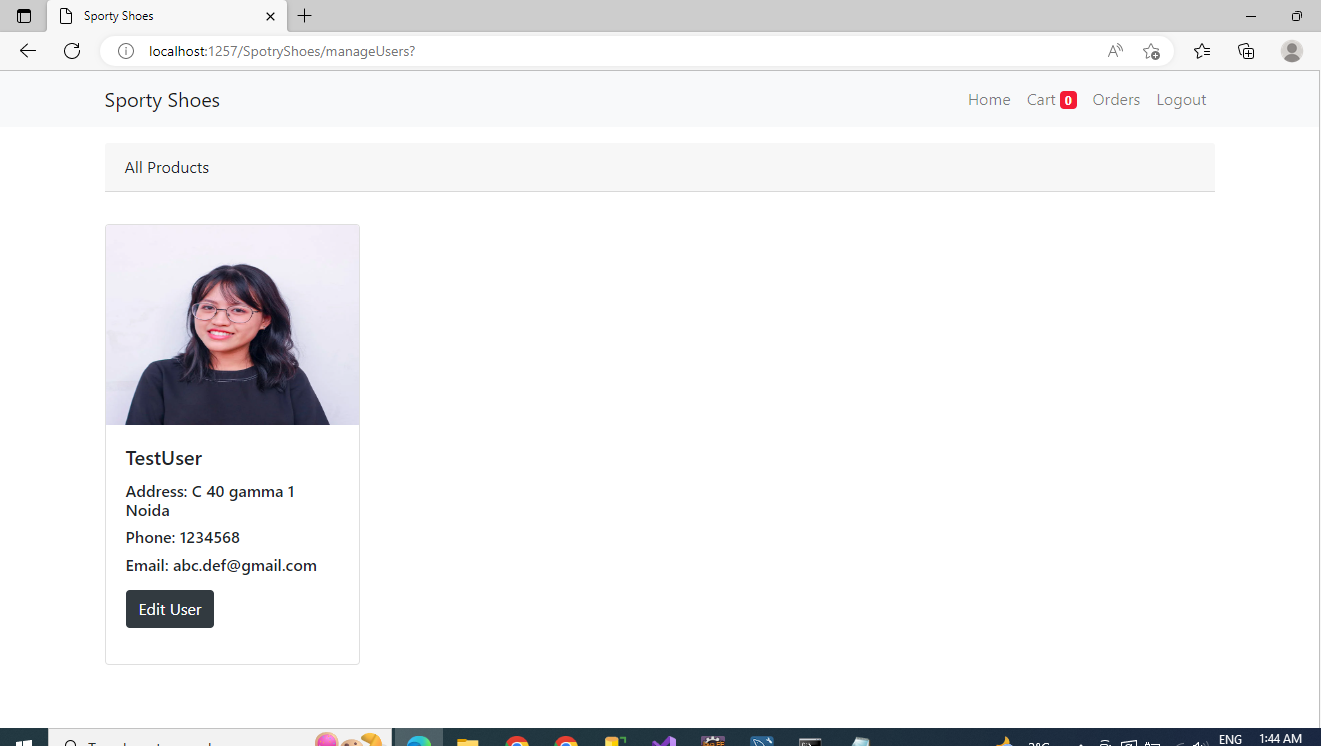
* + - 1. Click on edit product to edit any product details. You will be redirected to edit product page.



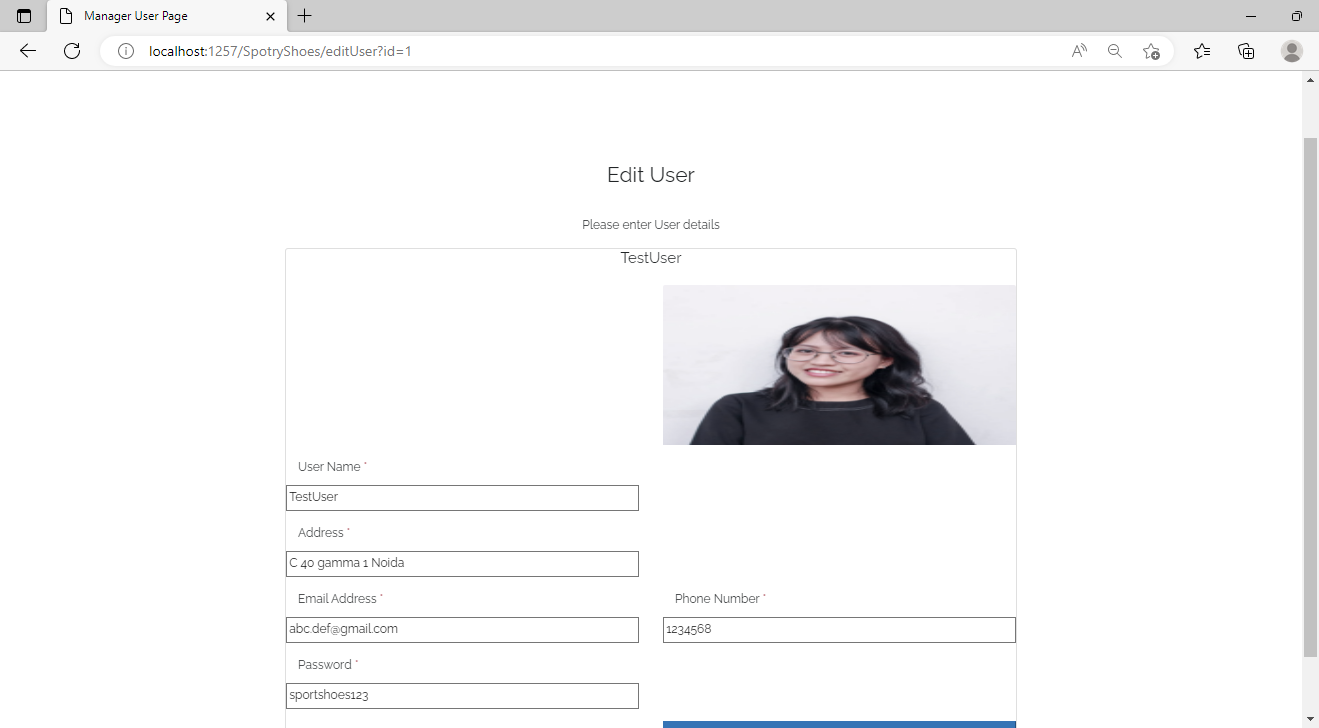
* + - 1. Enter your product details you want to change and click on submit details. You’re pare will reload and display latest updated details.



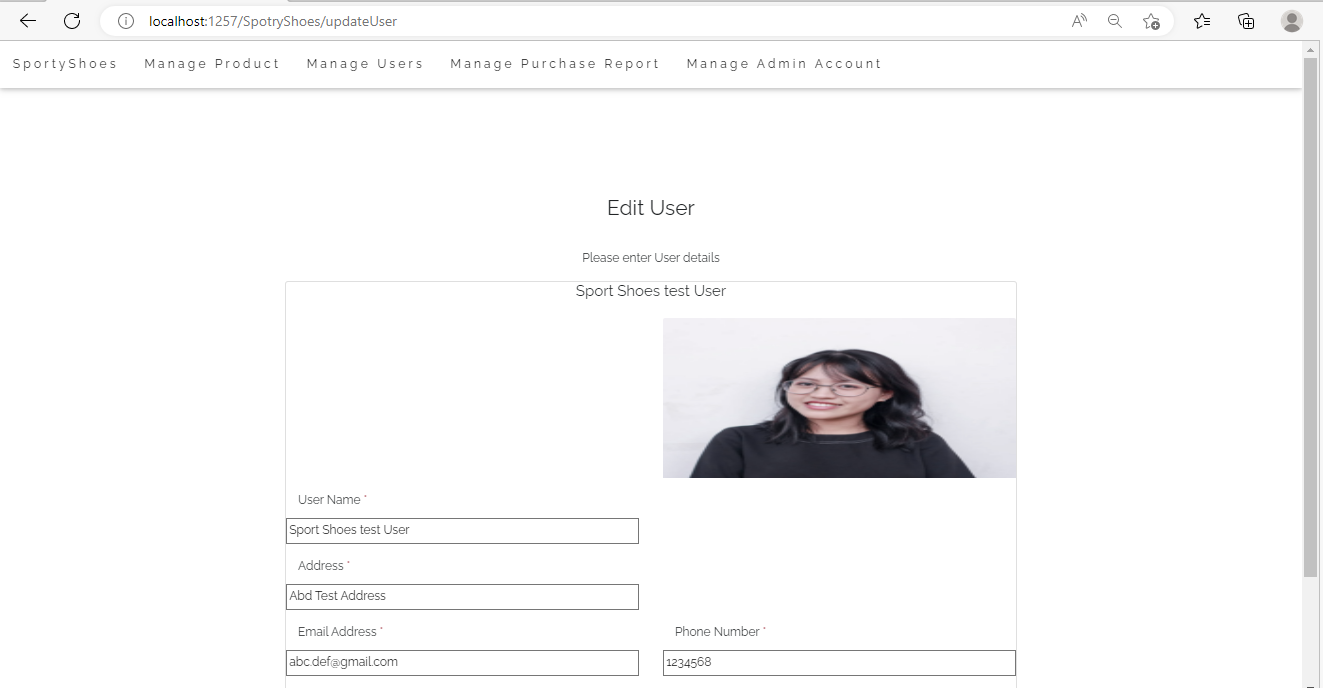
* + 1. Manage User.
       1. Click on Manage User, you will be redirected to Manage User Page.



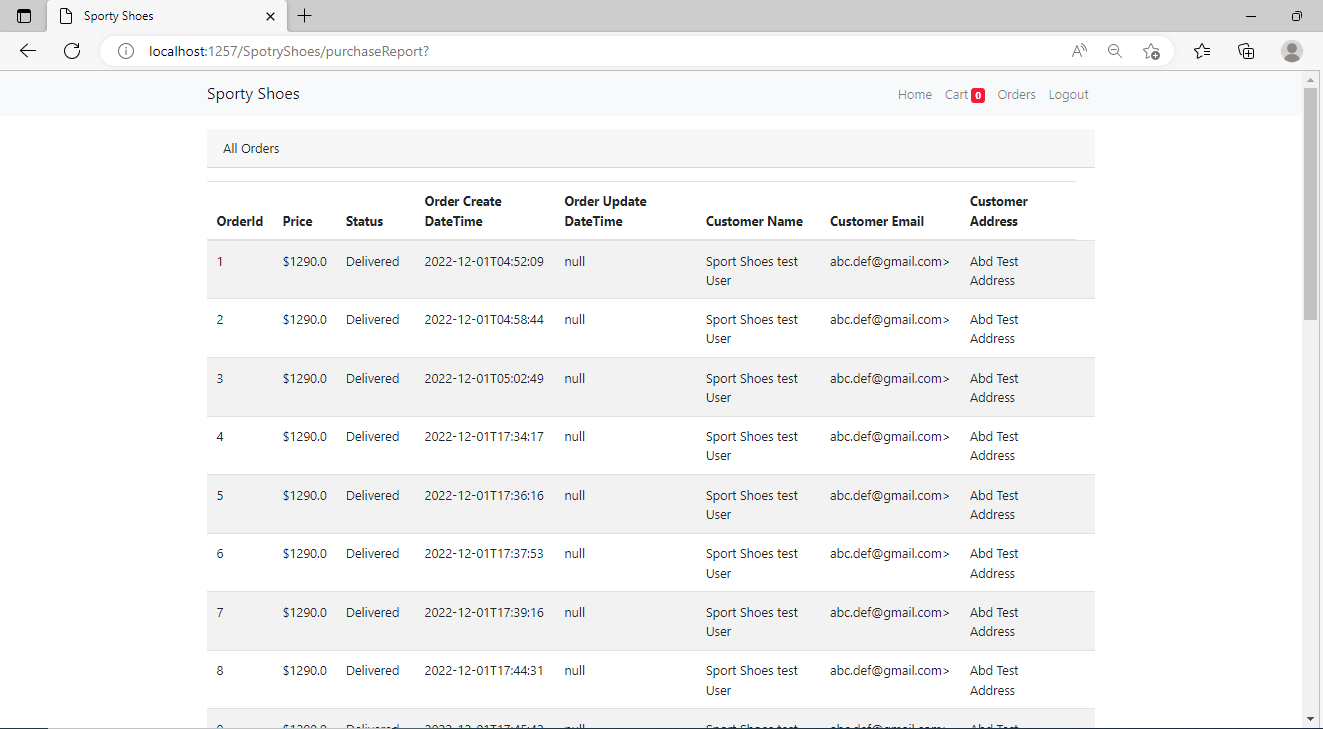
* + - 1. Click on Edit User. You will be redirected to edit user page.



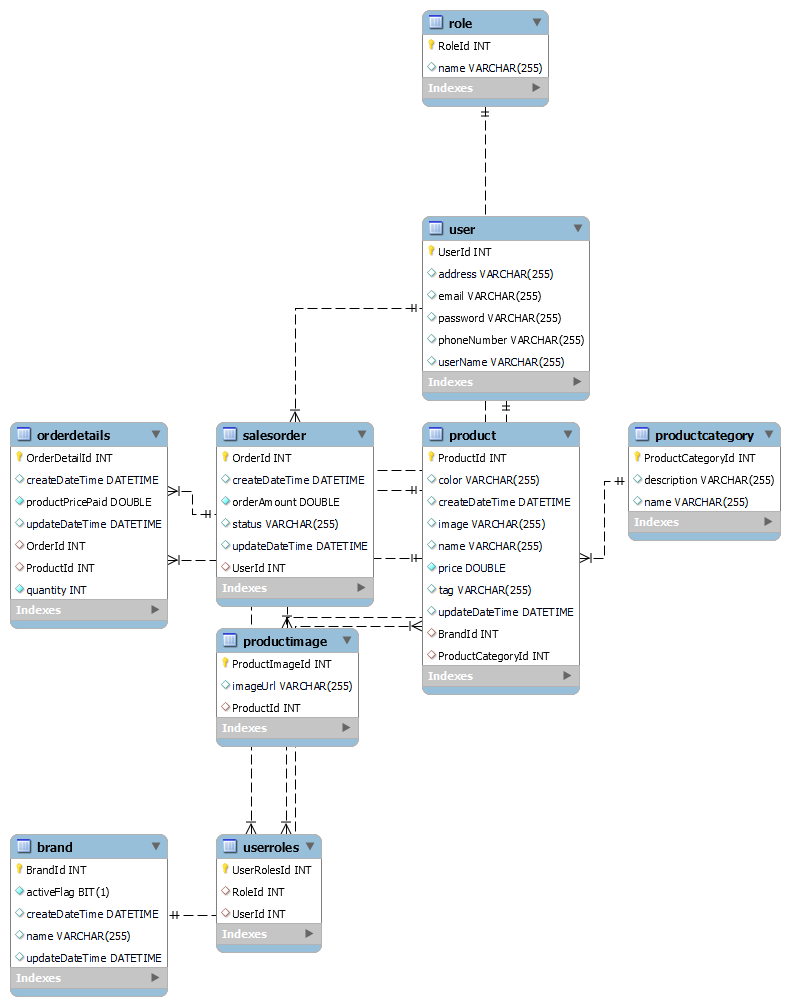
* + - 1. Enter details you want to update and lock on submit button.



* + 1. You can click on Manage Admin Account. You would be able edit manage account details in exactly same flow as Edit Users.
    2. Working with Purchase Report.
       1. You click on Purchase Report. You would be able to view purchase report.



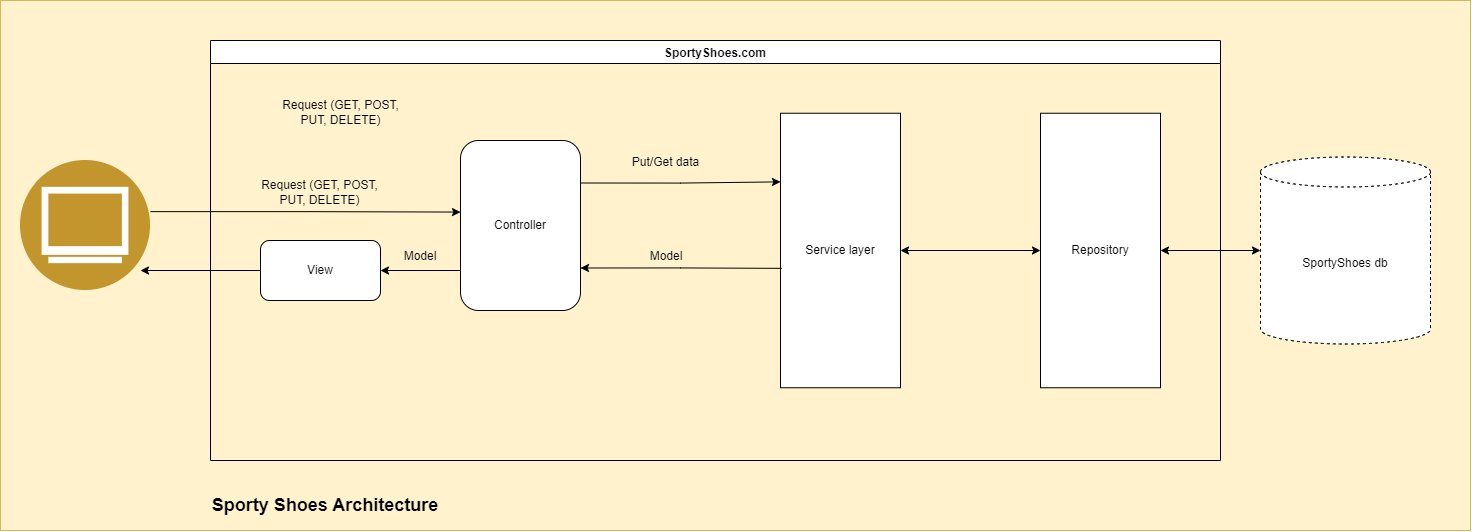
1. **System Design**
   1. **ER Diagram**



* 1. **Sporty Shoes System Architecture**

On high level Sporty is composed of 2 components

* + 1. **Sporty Shoes web app application:** This is a Spring MVC based web application which interacts with user and takes input and displays output to user. It consists of three components. It follows MVC+ Service + Repository Architecture.
       1. **Spring MVC:**  It consists of MVC based web application architecture.
       2. **Service Layer**: It is the service layer that interacts between MVC layer with DAL layer.
       3. **Repository Layer.** It is the data access layer which interacts between Service Layer and DB.
    2. **Sport Shoes Db.** it is the main database server which stores all business data. Sporty Shoes interacts with db via spring hibernate.
       1. It is created by Spring MVC using Code First Approach.



* 1. **Application Technical Specifications:**

Application has been designed to stick to SOLID principles. Application is designed to be easily scalable where new functionalities can be added without impacting existing ones. The following points have been taken into consideration while designing the application:

* + Application should be easily scalable.
  + Application should have the capability to be added with more menu options easily without impacting existing menu options.
  + Individual menu options can be added with new operations easily without impacting the existing ones.
  + High code reusability but with maximum loose coupling.
  + Thus, the application should be maximum closed for modification and open for extension.
    1. **Design Pattern Used:**
       1. MVC
       2. Service + Repository Layer.
       3. Dependency Injection
    2. **Data structures used**
       1. **ArrayList: To perform handle collection of files.**
       2. **HashMap: maps command with operations.**

1. **Sprint Planning:**

The Application has been implemented in two sprints

|  |  |  |
| --- | --- | --- |
| **Sprint Number** | **Days** | **Functionalities Delivered** |
| 1 | 15 | * Operations package * Helper package |
| 2 | 15 | * Menu package * Application integration and system testing |

1. **Unique Selling Points:**
   1. Application is extremely easy to use.
   2. Loose coupling between business and presentation logic.
   3. Highly scalable application due to implementation of Rest service.
   4. Web application implemented using Spring MVC provides extremely flexible and agile capabilities.
2. **Further improvements:**
   1. Allow users to create new account.
   2. Integration of resource file to configure repetitive config and data.