

---

# **Software Design Specification**

**for**

## **eCommerce Website 354**

**Version 1.0 approved**

**Prepared by Group 9**

**November 8th, 2019**

**Team members:**

Alexander Cohen	26847692 (Coordinator)
Konstantin Hristev	40008099 (Co-Coordinator)
Maude Braunstein	27545967
Jesse Silber	26993702
Hubert Alarie	40077197
Tarek Deghayli	40022064
Lin Tian	40077354
Gagandeep Singh	40037231
Dongdong Zhang	40043995
Hao Bo Yuan	40013341
Houssen Kassir	40015126

## Course Information

The report is prepared as part of the team project for the course of COMP 354 - Software Engineering at Concordia University in Fall 2019.

## Revision History

Name	Date	Reason For Changes	Version
Group 9	Oct. 6, 2019	The first version	1.0
Group 9	Nov. 8, 2019	Second revision; Adding more use cases,E/R Diagram for database, frontend object diagram, and screenshots.	2.0

## Table of Contents

<b>Introduction</b>	<b>Error! Bookmark not defined.</b>
<b>Design Considerations</b>	5
Functional and Non-functional Requirements	5
Goals and Guidelines	5
Development Methods	6
<b>Entity Relationship Diagram</b>	6
<b>Use Case Diagrams</b>	11
<b>Class Diagram</b>	17
<b>External Interfaces</b>	24
<b>Appendix: Changes in Requirements</b>	35
<b>Bibliography</b>	36

# 1. INTRODUCTION

This document outlines the software design and specification, especially, entity relationship diagram, use case diagrams, class diagram and external interfaces (screenshots of front pages) with all its inputs and outputs are presented to illustrate key points in the design stage.

Intended readers of the document include both stakeholders and website developers. However, the technical parts aim mostly to development side; project managers, designers, front-end/back-end developers, testers, documentation writers and other contributors of this project. For information related to Requirement Specification, please refer to the first SRS document version 1.0.

The ecommerce web platform is a relatively new, self-contained product, whose main function is to provide consumers with a business platform to buy and sell products in an international market.

This software is a web-based application. Stakeholders are interested in having a platform that can be used to sell and purchase various products online. 8% commission rate attained from every item sale by the platform for company profit. The development team aims to provide a user friendly environment, whereby customers are able to easily buy and sell their products. Increasing the ease of use for end users will contribute to the website's growth, and in turn generate higher company profit, and ultimately greater shareholder profit.

## 2. DESIGN CONSIDERATIONS

### 2.1. *Functional and Non-functional Requirements*

The design of e-Commerce Website 354 is based on a series of functional and non-functional requirements, as listed in the previous document. Specifically, functional requirements include requirements for the transactions, the sellers, the UI interaction, site features, account settings, etc.. Non-functional requirements deal with product requirements, organizational requirements, and external requirements. Details could be found in corresponding sections in SRS document version 1.0.

### 2.2. *Goals and Guidelines*

The goal of having a design document is to have a measure of our progress and easy assessment of any change in requirements.

November 08, 2019 - Submission of Design Document

November 15, 2019 - First prototype

December 03, 2019 - Final presentation

### **2.3. *Development Methods***

The front-end of the online platform will be developed using Vue.js and the Bootstrap framework, in order to create a responsive, user friendly, and overall aesthetically pleasing user interface. The amount of animations will be limited, and the components of the website will be structured in a logical hierarchy. Developers will follow ADA regulations to keep the website accessible to users with impairments.

The back end will be developed using MySQL, a flexible, fast, and very scalable database used to handle e-commerce that supports a large user base. The middle ware will consist of a python Django framework, which is easy to use and can encompass several modules that will facilitate communication with the database and frontend.

## **3. Entity Relationship Diagram**

The entity relationship diagram of the database is shown in Figure 3.1. Fourteen entities are defined. Detailed descriptions are shown below.



**goods\_goods class(main):** This entity describe all the attributes of a single product. The 'category\_id' is a foreign key that is bound to the product category class.

goods\_sn: Unique serial number id for the listed good.  
name: name of the good as described by the manufacturer.  
click\_num: number of times the specific good was clicked.  
sold\_num: number of times the specific good was sold.  
fav\_num: number of times the specific good was favorited.  
goods\_num: quantity of good available in stock(it decreases every time a valid purchase is made).  
market\_price: the listed price at which the product is available.  
shop\_brief: a short description of the seller that sells this product.  
goods\_brief: a brief description of the good.  
goods\_desc: a detailed description of the good.  
ship\_free: a flag that describes if the shipping is free the item.  
goods\_front\_image: the first image that a customer on the website will see.  
is\_new: flag to indicate if the good is a new product or not.  
is\_hot: the good is considered of higher importance, due to popularity in sales.  
add\_time: date and time when the good was made available.  
category\_id: the id of the category the good belongs to.

**goods\_goodscategory:** This entity mainly describes the category that a product belongs to. The category has a unique identifier called to reference the category item. Furthermore, the product class has a foreign key 'parent\_category\_id' which stands for the category of good from its superclass, it matches the category\_id of the superclass.

id: primary key that uniquely identifies each category.  
name: name of the category.  
code: code assigned to a specific category.  
desc: a description about the types of product this category contains.  
category\_type: since categories can have multiple levels, like mens->boots->sneaker or women->handbag->shoulder\_bag  
is\_tab: if the goods has tab.  
add\_time: time when category is added.  
parent\_category\_id: general category the item belongs to.

**goods\_goodsbrand:** This entity is a subclass of category, it describes the product brand name under the category, it has a unique identifier to reference the brand. The class contains a foreign key for reference by category parent using 'category\_id'.

id: primary key that assigns unique id to the brand  
name: name of the brand  
desc: description about the brand that can include type of product sold, international/local and other details



image: brand image  
add\_time: time when brand is added.  
category\_id: foreign key that relates it to the parent class goods\_goodcategory

**goods\_goodsimage:** This entity is a subclass of product class. Can be referenced to main class using foreign key goods\_id.

id: image id.  
image: goods image.  
add\_time: time when image is added.  
goods\_id: goods id.

**goods\_banner:**

id: primary key to  
image: image corresponding to the banner  
index: index that corresponds to the banner  
add\_time: time when banner is added.  
goods\_id: to identify which good it refers to

**user\_operation\_userfav:** This entity keep track of users favorite operations so that later can be used to analyze and pop up personalized advertisement.

**id : favorite user operation id.**  
add\_time: time when operation is added.  
goods\_id: goods id.  
user\_id: user id.

**trade\_shoppingcart:** This entity describe the shopping cart, including type of goods that were saved and the quantity of the goods.

id: primary key to identify unique trade id.  
nums: trade numbers.  
add\_time: time when operation is added.  
goods\_id: id that is used to reference the good  
user\_id: foreign key to identify the user uniquely

**user\_operation\_useraddress:** This entity simply describe user's address. This class is a subclass of user class and it has foreign key 'user\_id' to bind super class.

id: primary key to identify users uniquely  
province: user province location.  
city: user's city location.  
address: user's home address including apartment and street name.  
signer\_name: user signature.

add\_mobile: user mobile number.  
add\_time: time when operation is added.  
user\_id: user id.  
district: user district.

**users\_userprofile:** This entity describes the user's personal information. It is the super class in user category, and it has 5 subclasses.

id: primary key to identify users uniquely  
password: user password which is encrypted before storing in the database  
last\_login: time when the user last logged in.  
is\_superuser: flag to indicate if the specified user is admin or not.  
username: string that identifies the username of the customer/seller, it is unique  
is\_staff: flag to identify if the user belongs to the staff or not  
is\_active: flag to tag online users  
date\_joined: date on which the account was created  
first\_name: user's first name.  
last\_name: user's last name.  
birthday: user birthday(day/month/year).  
gender: gender of the user for statistical purposes  
mobile: mobile number of the user  
email: email address of the user used for communication purposes

**users\_userprofile\_group:** This entity is used to describe user category types and is one of the subclasses of user. User can be seller or buyer.

id: user group id.  
userprofile\_id: user profile id.  
permission\_id: permission id.

**users\_userpermission:** This entity control user access to certain functionalities.

id: primary key for the user  
userprofile\_id: user profile id.  
group\_id: user group id.

**user\_operation\_userleavingmessage:** This entity keep track of the comments and reviews for each product, as well as the rank for seller.

id: message id.  
message\_type: message type.  
subject: subject of the message.  
message: message.  
file: file path.  
add\_time: time when message is added.  
user\_id: user id.

**trade\_orderinfo:** This entity keeps track of each payment transaction. It contains all billing attributes.

id: primary key to identify each order uniquely.  
order\_sn: serial number of the order.  
pay\_status: payment status (accept, decline, pending...)  
post\_script: postal information.  
order\_mount: this is the total amount that the customer needs to pay for the order  
pay\_time: time at which the payment was processed.  
address: address where the invoice will be sent i.e. billing address.  
signer\_name: name of the user who placed the order.  
signer\_mobile: billing phone number.  
add\_time: time when the order was placed.  
user\_id: foreign id that identifies the user who placed the order.

**trade\_ordergoods:** This entity describes that in each order, the category and brand of a single good and the quantity.

id: primary key for id of ordergoods.  
goods\_num: quantity of goods ordered.  
add\_time: time when the order was placed.  
goods\_id: foreign key to identify the good.  
order\_id: foreign key to identify the order info.

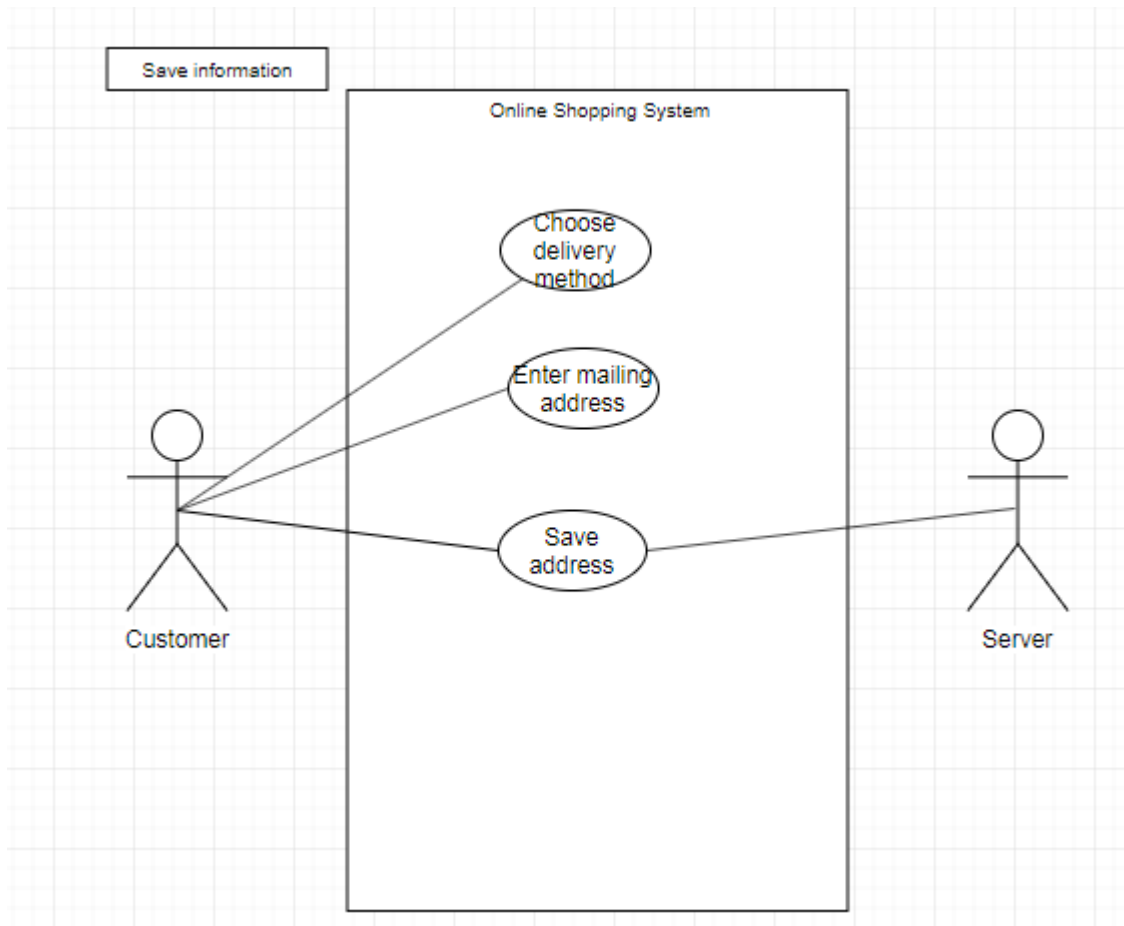
## 4. Use Case Diagrams

The use case diagrams below illustrate the relationship between the user and the server during several types of actions on the website.

### 4.1 Fast Checkout

This is a behavioral UML diagram between Customer and Server. The server stores information related to the customers address, which facilitate his/her checkout for future

purchases. The Delivery methods such as “One-Day Shipping” or “Standard Shipping” will have to be chosen in every delivery.



**Figure 4.1.** Customer interaction with the eCommerce site for selecting user preferences.

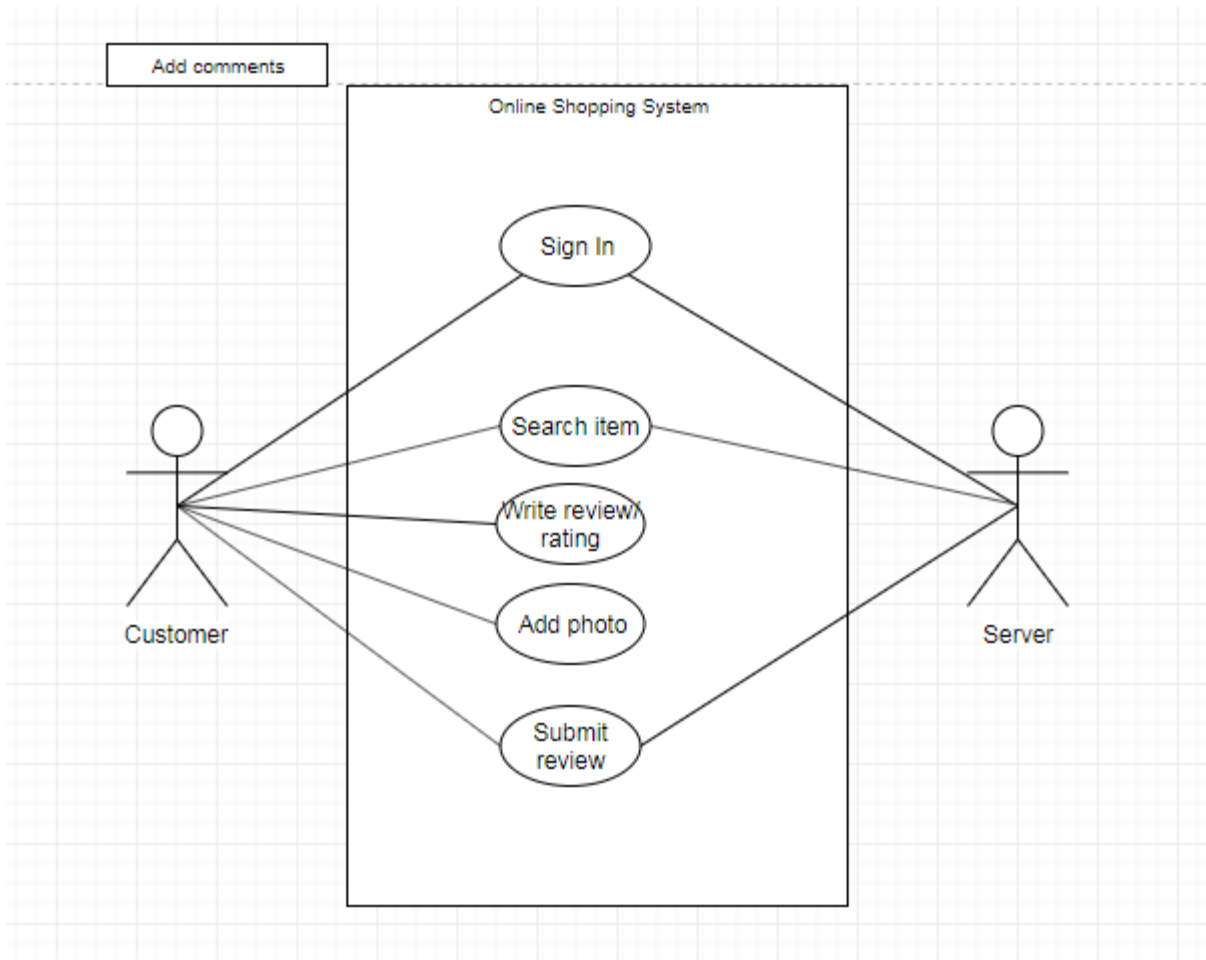
## 4.2 Adding Comments

This UML Diagram shows the involvement of the Customer and the Server in terms of adding comments to a listing.

Only customers are allowed to add a comment and they need to be logged in to do so.

Customer can search for a listing, by making a request to the server, and then add comments to it. The limit for adding a comment is 4 lines maximum. To make it more user friendly they have the option of uploading a picture of the product they are reviewing. A maximum of 5 images can be uploaded.

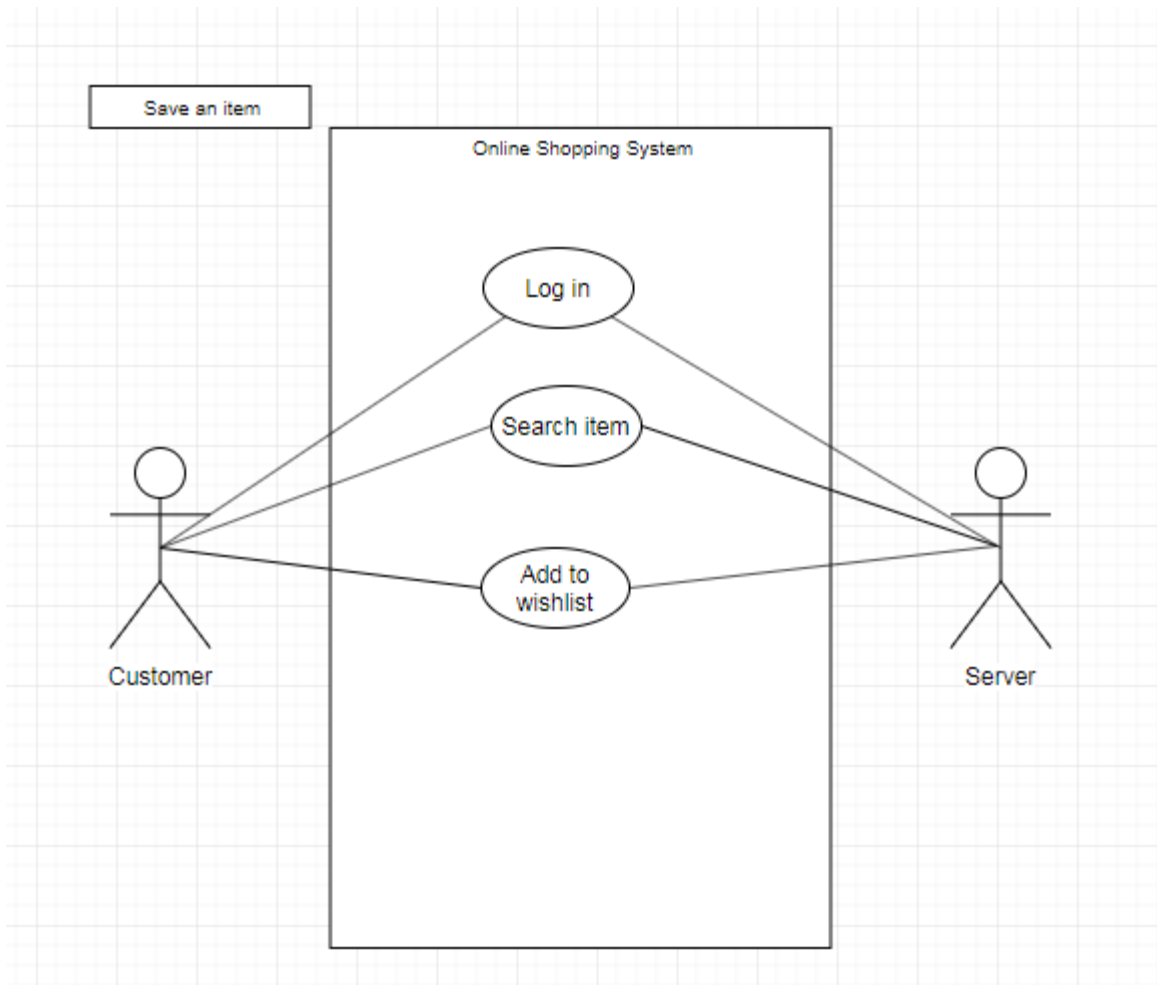
Then the user can submit a review once they have entered all the information.



**Figure 4.2.** Customer interaction with the site for overall comments

### 4.3 Saving Items

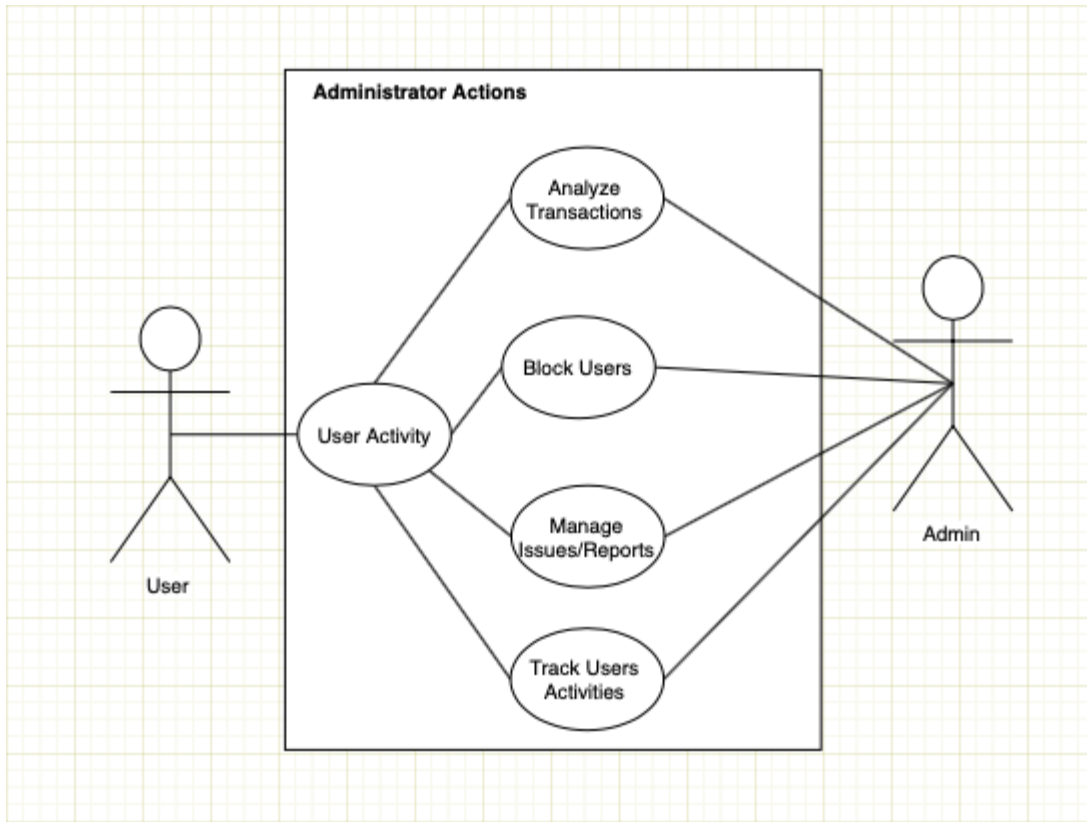
To save an item a customer needs to be logged into the website. They have an option to save their searches in a wish list and purchase it on a later date.



**Figure 4.2.** User interaction for search and adding items to wishlist

#### 4.4 Admin Roles

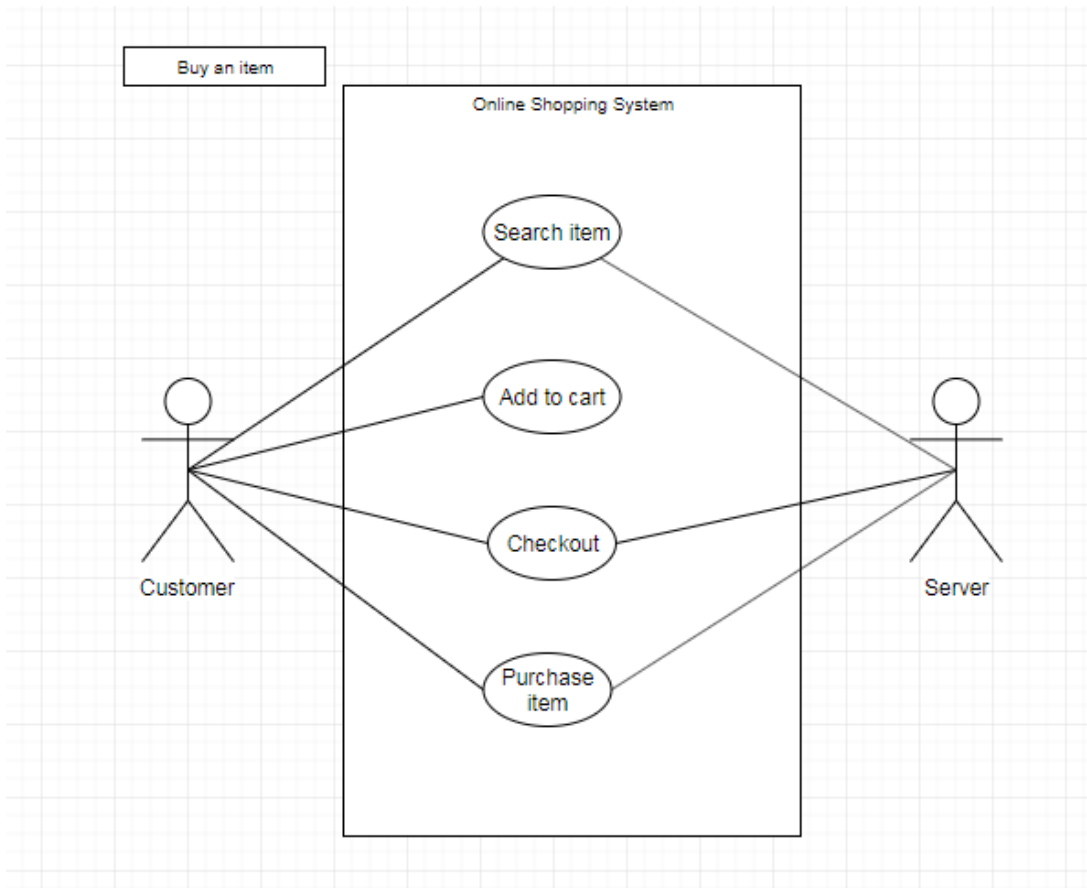
As an administrator of the website, one can analyze the transactions which are part of the revenue for the business. If some seller is receiving negative comments, the admin of the website can block them from selling products using this platform. Admin is provided with a view to manage reports and resolve issues. They can also track the activities of seller to keep track of best sellers in terms of quantity sold or amount earned.



**Figure 4.3.** Administrator actions for performing administrative tasks

#### 4.5 Buying an Item

For a customer to buy an item, they have to be logged in to the website. We provide the option of searching which helps the customer to look for specific items. Once the customer has made up his mind to purchase an item, they can choose the option "Add to Cart". Multiple products from multiple sellers can be added to the same cart. When the customer wants to pay for those items, they can choose the "Checkout" option which leads to payment gateway.

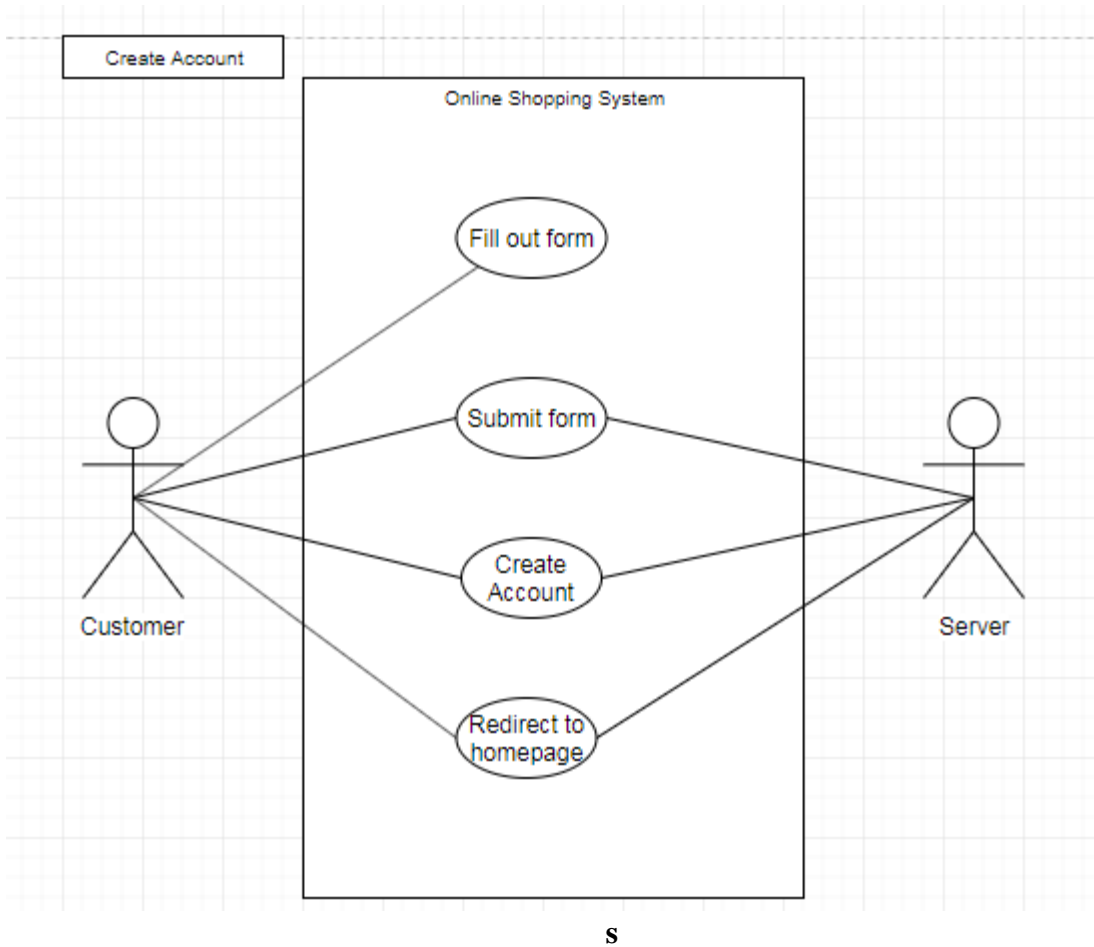


**Figure 4.4.** Figure demonstrating purchase of an item

#### 4.6 Account Creation

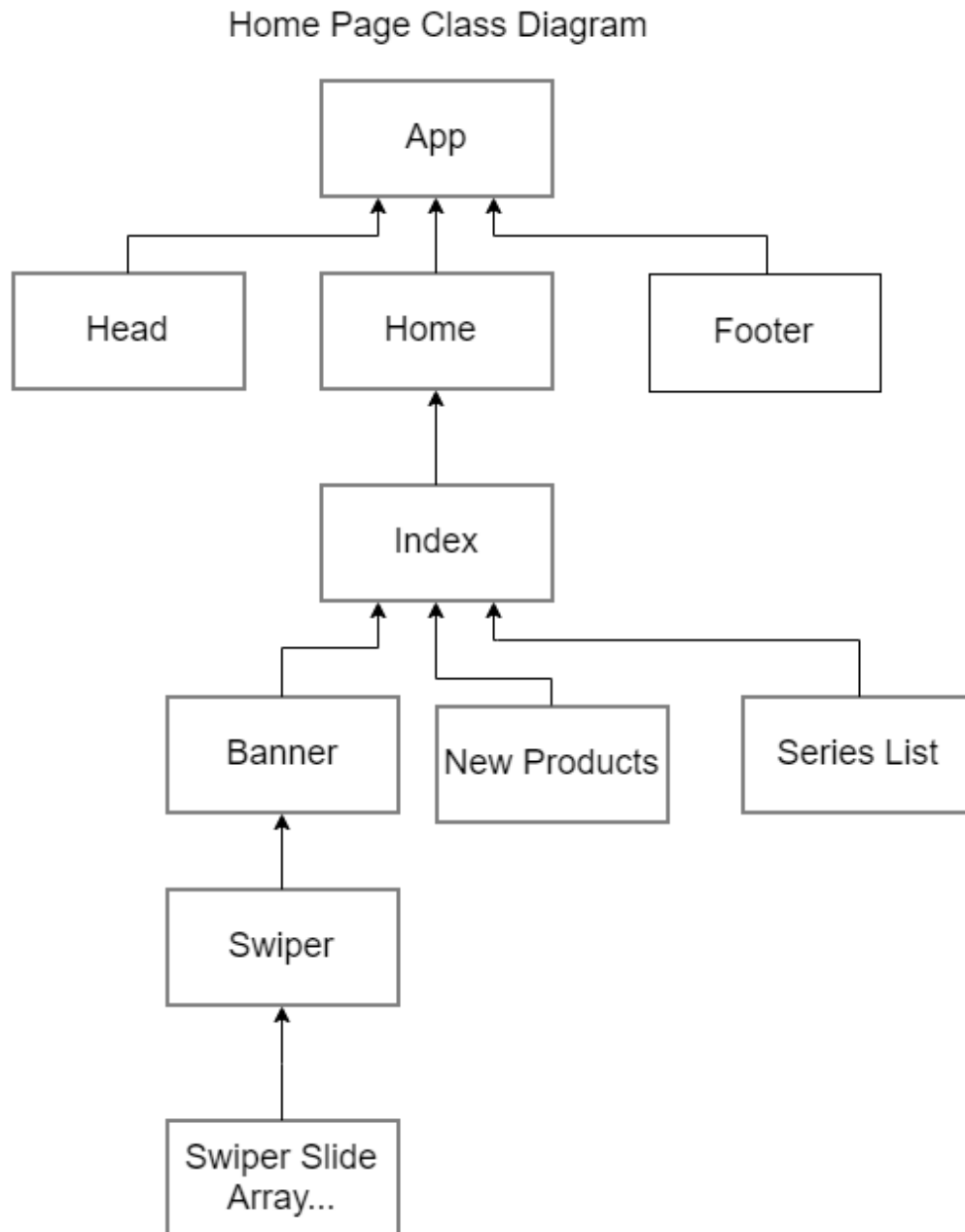
This UML diagram is to represent the functionalities of a login page. The login page contains a form to be filled by the user. It contains username and password that needs to be entered by the user. There is a "Register" button that needs to be clicked which displays a message once the account is created successfully. The user is then redirected to the home page of the website.





## 5. Class Diagram

Five class diagrams are made to illustrate the class relationship in the design of the front end. Details are provided in each diagram.



**Figure 5.1.** Home Page Class Diagram, this displays the front banner alongside the new products featured in the store.

**App:** provides the main stage for displaying the webpage. It contains all the child components inside.

**Head:** Component provides the TopBar for Searching, Sign In/Sign Up, including the logo and shopping cart.

**Home:** Contains all the main content for the web page.

**Index:** Contains the Banner and recent content from the page.

**Banner:** Front Banner for loading images of promotional items on display

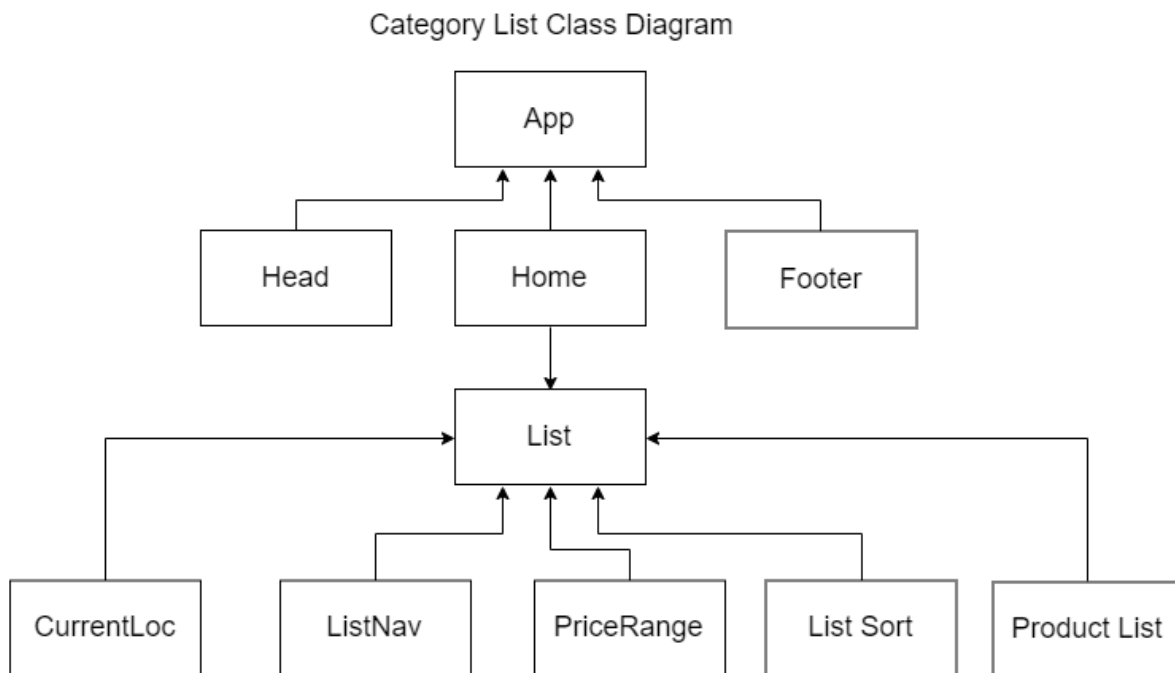
**Swiper:** Component for providing animations for sliding between images

**Swiper Slide Array:** collection of each individual image with description for the banner

**New Products:** Dynamic feature for displaying new items available in the store

**Series List:** This component outlines the guarantees the website provides to the users

**Footer:** Provides additional information concerning website and contact info.



**Figure 5.2** Displays the collection page of items based on the category they belong to, alongside with breadcrumbs and filtering based on price and name of product.

**App:** See the 'Home page' class diagram for description

**Head:** See the 'Home page' class diagram for description

**List:** Wrapper component for the display view of all components on the page

**CurrentLoc:** the PATH of categories, and listings to the current collection page.

**ListNav:** The navigation sidebar for displaying items under the current type of brand.

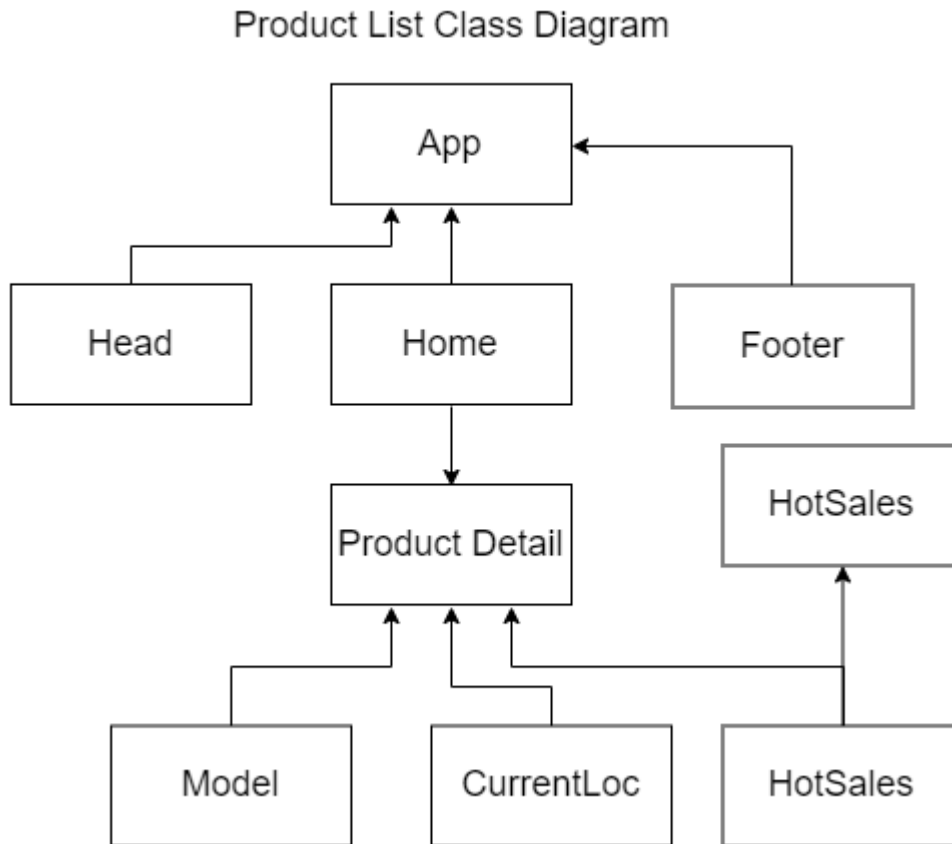
**PriceRange:** component for filtering the collection page of items, based on pricing

**ListSort:** Sort items based on pricing and quantity sold.

**ProductList:** Wrapper component containing a list of product elements.

**RouterLink:** Item component containing header image, description and pricing.

**Footer:** See the 'Home page' class diagram for description



**Figure 5.3** Details the item itself. Provides information with images, description and pricing.

#### Product List Class Diagram

**App:** See the 'Home page' class diagram for description

**Head:** See the ‘Home page’ class diagram for description

**Home:** See the ‘Home page’ class diagram for description

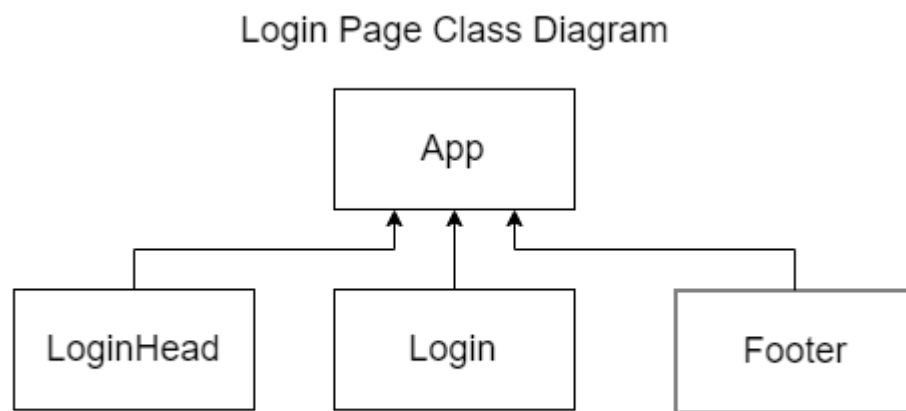
**Product Detail:** Entire component for wrapping the product description

**Model:** Displays the item with pictures and description, quantity, add to cart,etc.

**CurrentLoc:** Describes the PATH for the categories to the product.

**Hotsales:** Advertising sidebar for describing new sales of items

**Footer:** See the ‘Home page’ class diagram for description



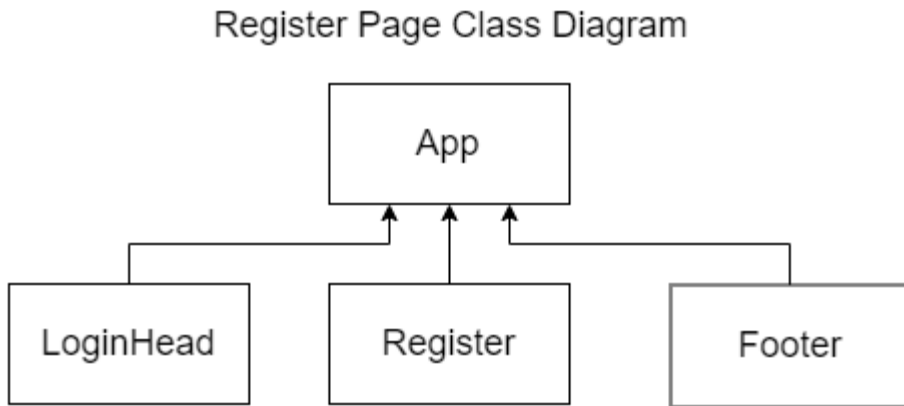
**Figure 5.4** Diagram for the login page.

**App:** See the ‘Home page’ class diagram for description

**LoginHead:** Component provides the TopBar (with the logo), specifically accommodating for login/signup page

**Login:** TextBox component that receives username and password credentials for login. Also provides register feature for users with no registered account.

**Footer:** See the Home page class diagram for description



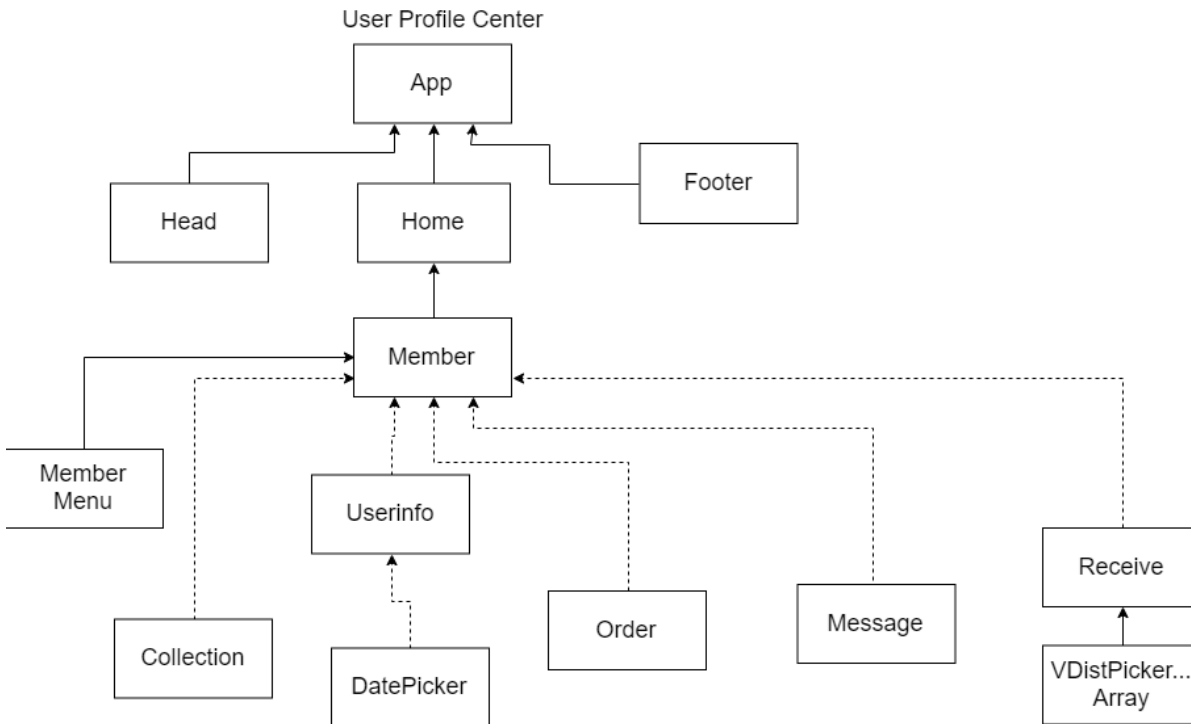
**Figure 5.5** Class diagram new users wanting to register to the eCommerce site

**App:** See the 'Home page' class diagram for description

**LoginHead:** See the 'Login Page Class Diagram' for more information

**Register:** Textbox component to enter username and password for registration.

**Footer:** See the Home page class diagram for description



**Figure 5.6** User Profile Center for viewing/editing profile information related to personal credentials, shipping, orders, and messages.

**Home:** See the ‘Home page’ class diagram for description

**Member:** Subclass containing the account credentials

**Member Menu:** The sidebar used for selecting user details and preferences to view, and edit such as order history, personal information, current orders and status, message center, and shipping.

**Collection:** Component represented as a table that is a backtrace of all the previous orders have been made.

**User Info:** Text boxes to change current personal information for update

**DatePicker:** responsible for selecting date of birth

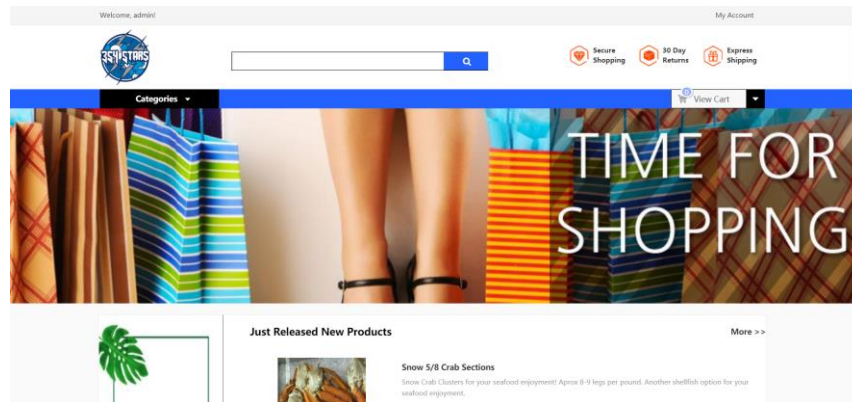
**Order:** Table component that looks at the current orders and their status

**Message:** message center for looking at communication between buyer and sellers

## 6. External Interfaces

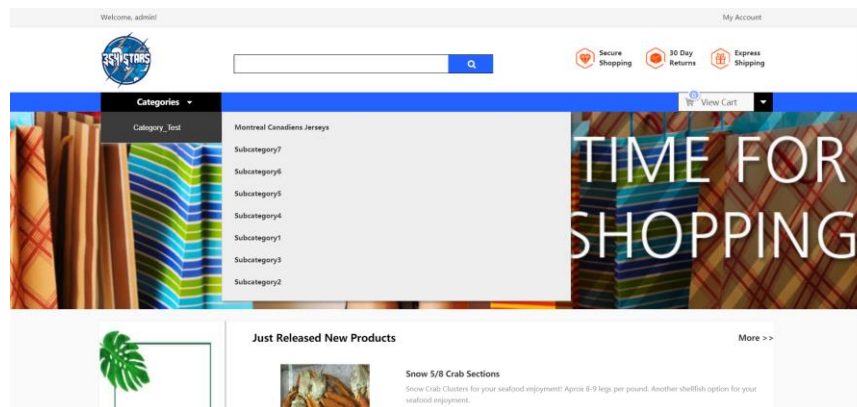
Screenshots of external surfaces with inputs and outputs are provided.

### 6.1 Home Page



**Figure 6.1.** Homepage view of the site

The home page allows the user to browse for products to buy. They can search for a specific product in the search bar, find a product using the categories, or browse and click a product to be redirected to its product page. The user can also log in or register for an account, as well as view their cart if they are logged in.



**Figure 6.2.** Homepage showing the drop down categories

Hovering over the categories button will allow the user to choose a category of products they want to browse. Hovering over a category will bring up its subcategories, which the user can click to bring browse the items of that category.

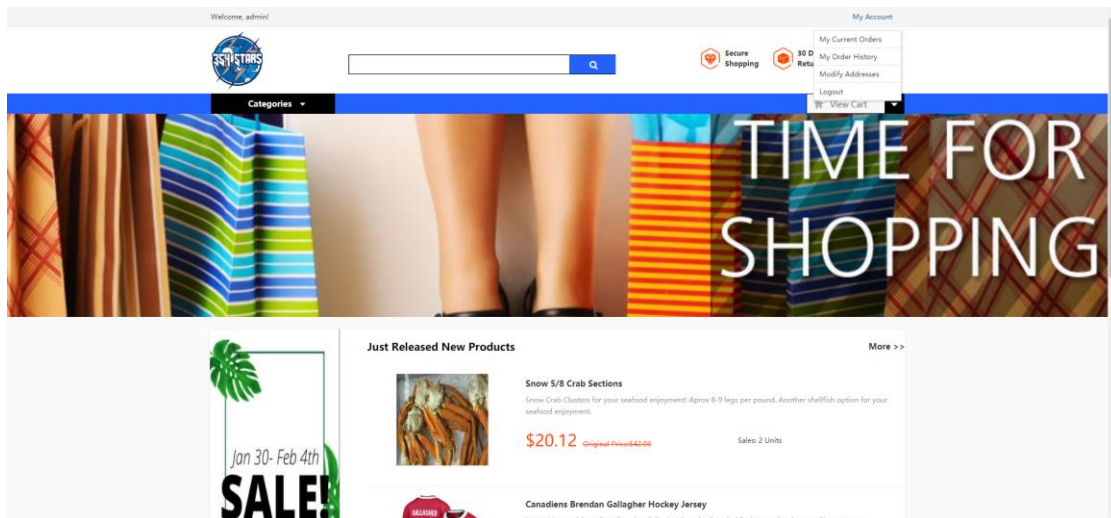


ing Returns Shipping



**Figure 6.3.** Zooming in the shopping cart to demonstrate that is logged in therefore cart will not work.

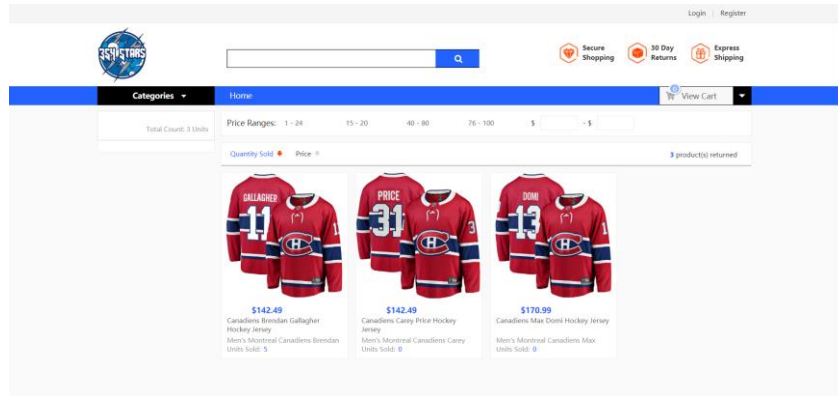
If the user is not signed into their account, hovering or pressing view cart will notify the user that they must be signed in to use the cart feature.



**Figure 6.4.** Demonstrates user preferences for logged in members.

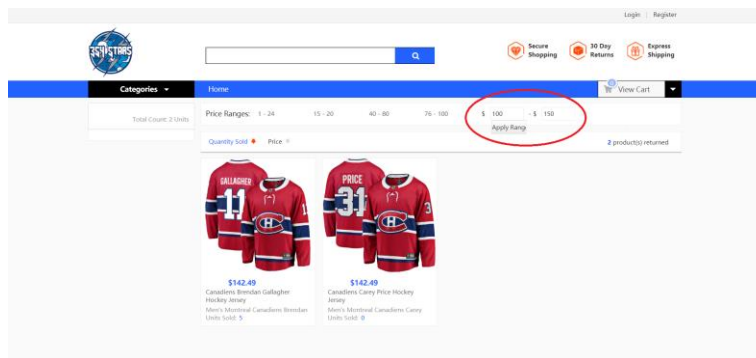
If the user is logged in, hovering over the 'my account' button will bring up the options to redirect to the different account pages.

## 6.2 Category Page



**Figure 6.5.** Collection page of items on display

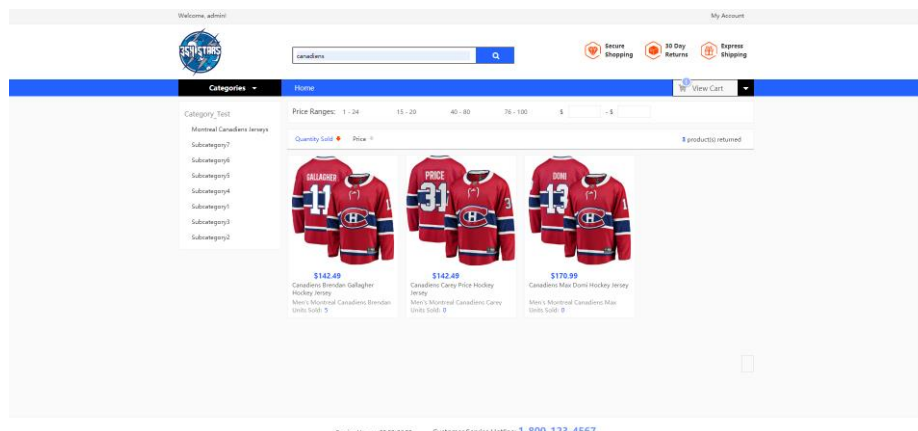
When clicking on a certain category, the user will be directed to the category page, which will display all of the items in that category or subcategory. The user can then click on their desired item to be redirected to its product page. The user can choose how many items they would like to have displayed on the page, and the price range they would like to have displayed.



**Figure 6.6.** Demonstrating filtering options for collection page based on price

Applying the range of prices will display only the items within that price range.

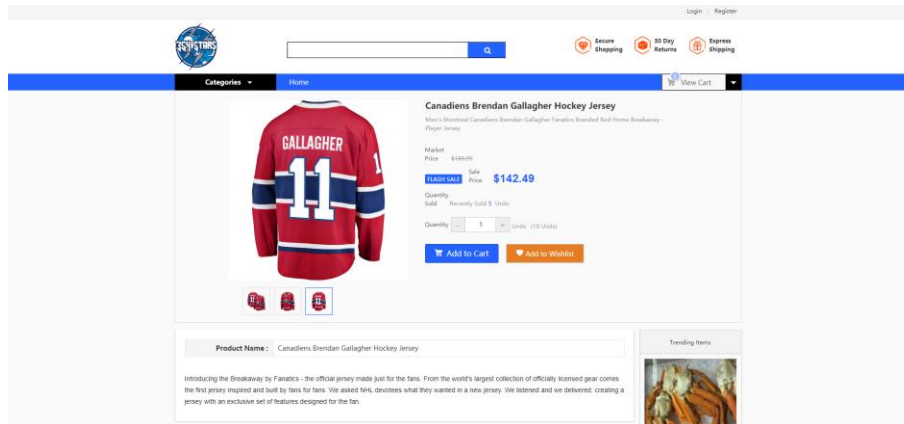
## 6.3 Search Page



**Figure 6.7.** Demonstrating filtering options for collection page based on quantity sold

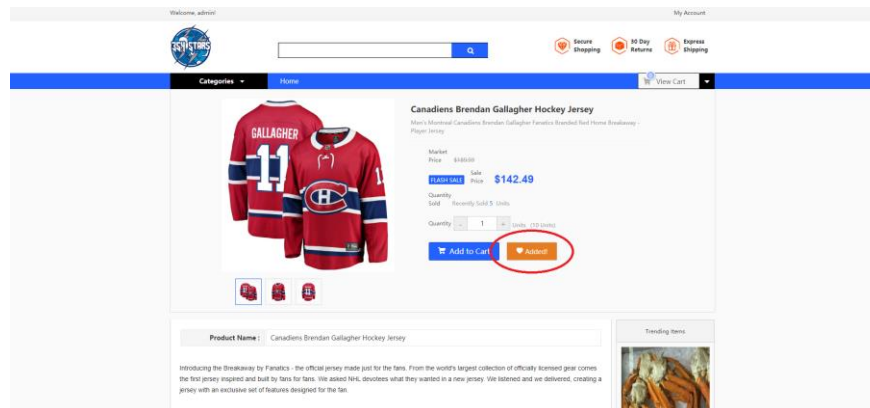
The search page will return all the items pertaining to the search. The user can choose how many items to display, and a price range for the items displayed, just like the category page. The user can click on a category in the sidebar to navigate to that category.

## 6.4 Product Page



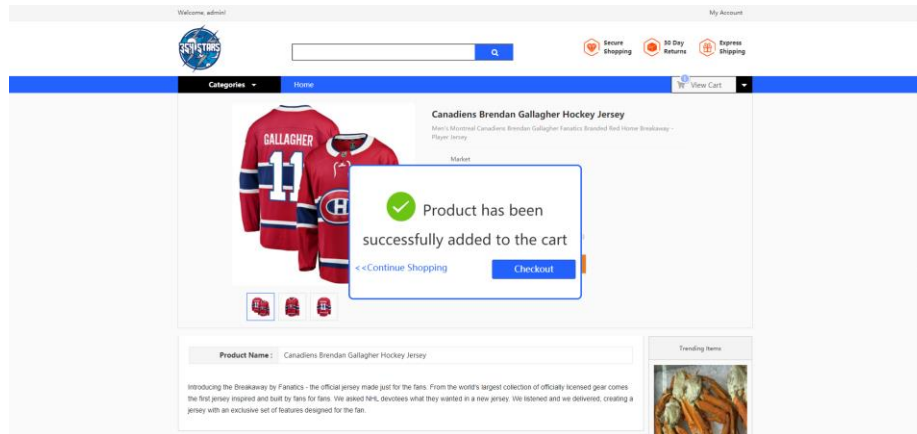
**Figure 6.8.** Displaying individual item for sale

The product page displays the products information, and allows the user to change the quantity they would like to purchase, as well as adding it to their cart and wishlist if they are logged in. The user can also click on the different pictures below the main image to see different views of the product.



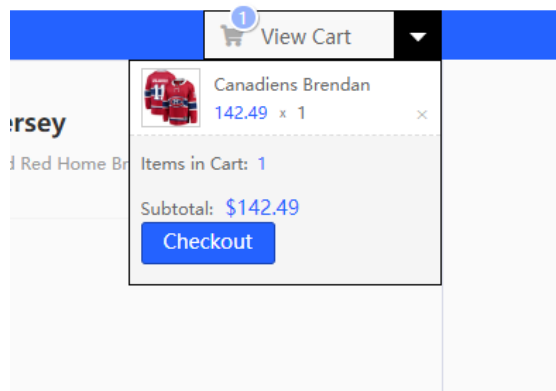
**Figure 6.9.** Adding item to wishlist option

When the user adds an item to their wishlist, they are notified that it has been added, and the add to wishlist button changes to added. If the user wants to remove the item from the wishlist, they can click the button again to revert it to its previous state and remove the item.



**Figure 6.10.** Adding item to cart

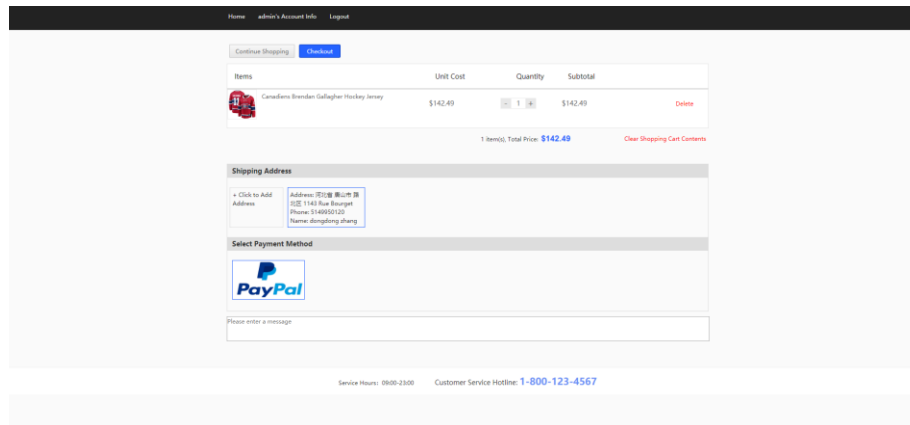
When a user adds an item to their cart, they get a pop-up to notify them that it has been added. From the pop-up, they can then choose to keep shopping, or go to checkout by clicking on the buttons in the pop-up.



**Figure 6.11.** Viewing the item on display in the shopping cart

When an item is added to the cart, hovering over the view cart button will display some product info, and will display a button to redirect the user to the cart.

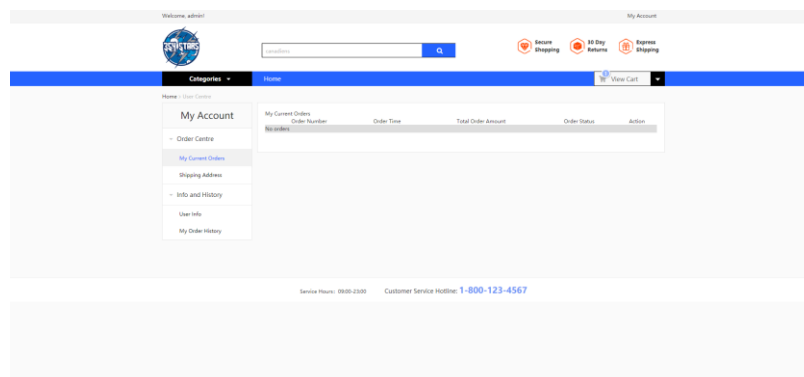
## 6.5 Checkout Page



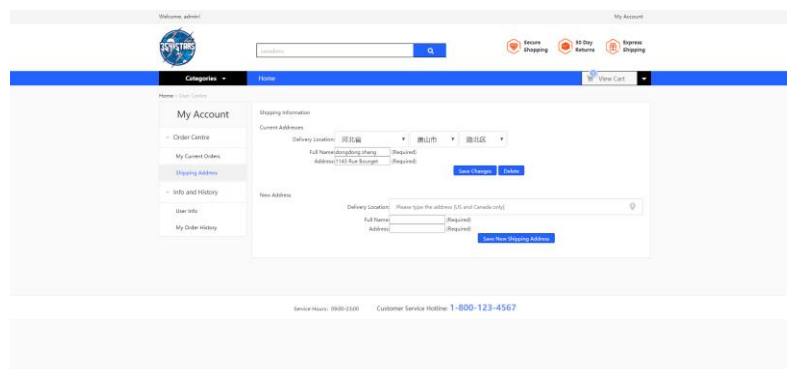
**Figure 6.12.** viewing the item in checkout

On the cart page, the products currently in the cart are displayed with their price, quantity, and subtotal. The user can click the delete button to remove an item from their cart. The purchase total is shown below the items in the cart. The user's shipping address is displayed below, and the user can click to add a new address. The user can also click to choose their method of payment. At the top, the user can click the buttons to redirect to the home page, their account info, or logout.

## 6.6 Account Page



The account page has four tabs. The first displays the items the user has ordered and is waiting for.



**Figure 6.13.** Viewing the user preference for shipping address.

The second page allows the user to edit their address, or new address by filling out all of the fields.

Personal Information

Name	
Date of Birth	
Gender	Male/Female
Email	email@domain.com

Confirm Changes

**Figure 6.14.** Viewing the user info for edit.

The third page allows the user to edit their personal information by filling out the fields provided.

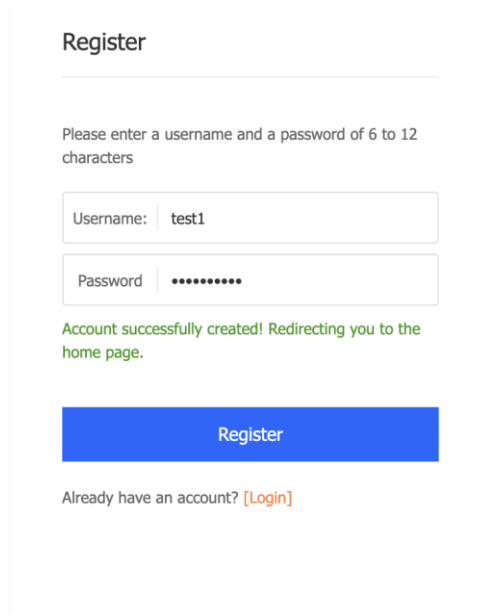
My Order History

Product Name	Price	Action
Canada 2010 Winter Olympic Games Hockey Jersey	\$20	Details
Canada 2010 Winter Olympic Games Hockey Jersey	\$19	Details
Canada 2010 Winter Olympic Games Hockey Jersey	\$142.49	Details

**Figure 6.15.** Viewing the order history,

The last page pertaining to the user's account displays the information for the user's previous orders.

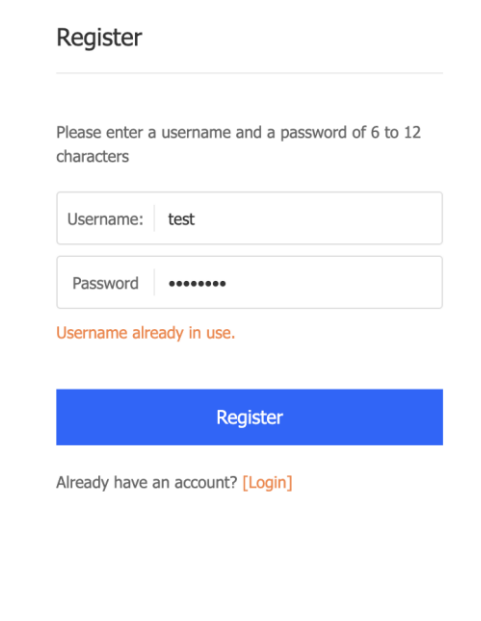
## 6.7 Registration Page



The screenshot shows a web form titled "Register". Below the title is a horizontal line. The text "Please enter a username and a password of 6 to 12 characters" is displayed. There are two input fields: "Username:" with the value "test1" and "Password:" with masked characters "••••••••". Below the password field is a green message: "Account successfully created! Redirecting you to the home page." At the bottom of the form is a blue button labeled "Register". Below the button is the text "Already have an account? [Login]" where "[Login]" is a link.

**Figure 6.16.** Successful registration

If the user registers successfully, they will be given the message that their new account was successfully created, and will be redirected to the websites home page.



The screenshot shows the same "Register" form as Figure 6.16. The "Username:" field contains "test" and the "Password:" field contains masked characters "••••••••". Below the password field is an orange error message: "Username already in use." The blue "Register" button is at the bottom. Below the button is the text "Already have an account? [Login]" where "[Login]" is a link.

**Figure 6.17.** username already exists.

If the user tries to register with a username which is already in use, they will be taken back to the registration page, and will be told that the username has already been taken. They can now try to register again with a username that has not already been taken.

## Register

Please enter a username and a password of 6 to 12 characters

Username: test1

Password: ...

Invalid password format.

Register

Already have an account? [\[Login\]](#)

**Figure 6.18.** invalid password character limit

When the user enters a password which does not fit the criteria, they are taken back to the registration page, with a message below the password box notifying them that the password does not have a valid format. From there the user can try to register again.

## Register

Please enter a username and a password of 6 to 12 characters

Username:

Password

Cannot leave a field blank.

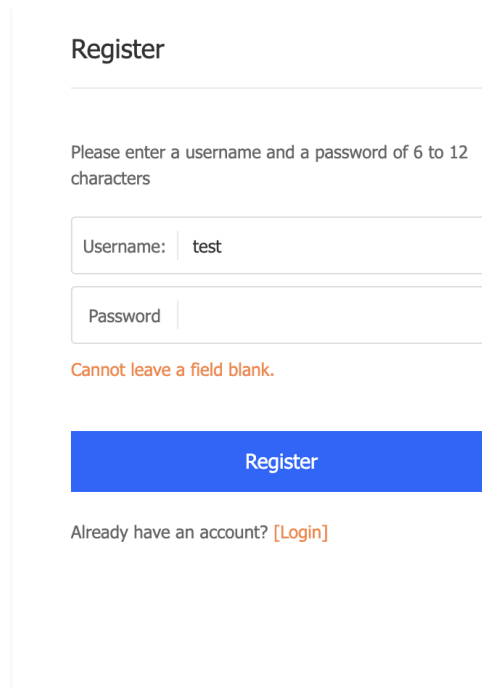
Register

Already have an account? [\[Login\]](#)

**Figure 6.19.** empty textbox for registration



If the user tries to register but leaves both fields blank, they are returned to the registration page, which displays a message below the password box notifying the user that the user cannot leave the field blank and must enter information into the blank fields.

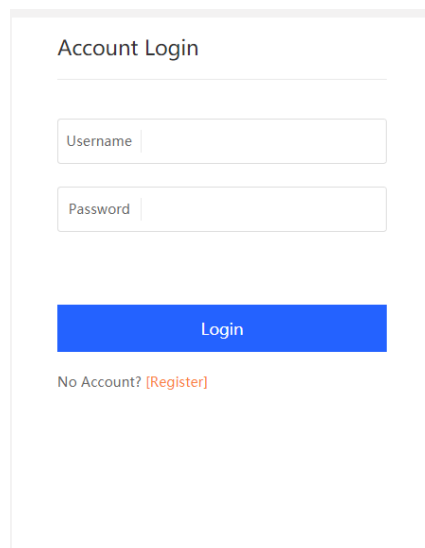


The screenshot shows a registration form titled "Register". Below the title is a horizontal line. The instructions read: "Please enter a username and a password of 6 to 12 characters". There are two input fields: "Username:" with the value "test" and "Password:" which is empty. Below the password field, a red error message states: "Cannot leave a field blank.". At the bottom of the form is a blue "Register" button. Below the button, it says "Already have an account? [Login]" where "[Login]" is a red link.

**Figure 6.20.** left password blank for registration

Even if the user fills one field, and leaves the other blank, they will be brought back to the registration page and shown the same message as in the screenshot above.

## 6.8 Login page

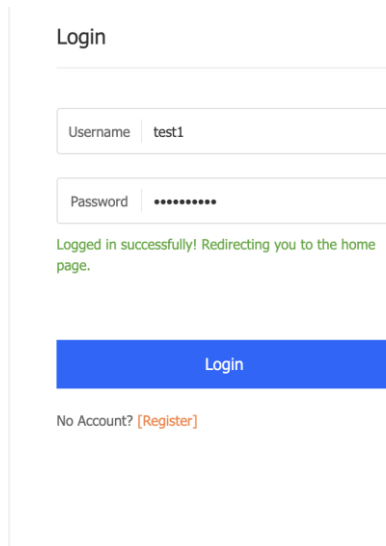


The screenshot shows a login form titled "Account Login". Below the title is a horizontal line. There are two input fields: "Username" and "Password", both of which are empty. Below the password field is a blue "Login" button. At the bottom of the form, it says "No Account? [Register]" where "[Register]" is a red link.

**Figure 6.21.** Login box

The login page allows the user to log into their account to purchase items from the website. The user needs to enter their username and password to log in. The user can also click on the

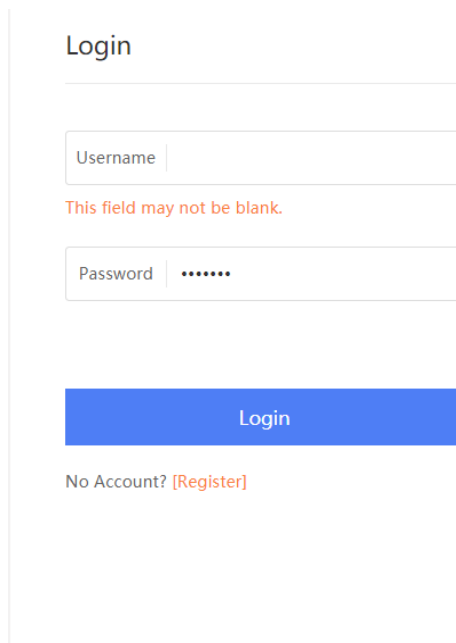
register button to be redirected to the registration page if they do not currently have an account.



The screenshot shows a login form titled "Login" with a horizontal line underneath. It contains two input fields: "Username" with the value "test1" and "Password" with masked characters "\*\*\*\*\*". Below the password field is a green message: "Logged in successfully! Redirecting you to the home page." At the bottom of the form is a blue "Login" button. Below the button is the text "No Account? [Register]" where "[Register]" is a red link.

**Figure 6.22.** Successful login

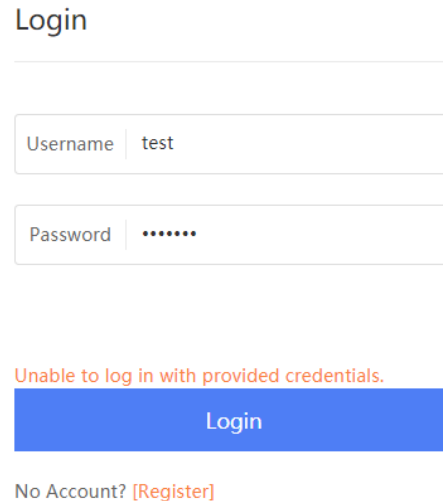
If the user logs in with a valid username and password, they will be notified that they have been logged in successfully, and will be redirected to the website’s home page.



The screenshot shows a login form titled "Login" with a horizontal line underneath. It contains two input fields: "Username" which is empty, and "Password" with masked characters "\*\*\*\*\*". Below the username field is a red error message: "This field may not be blank." At the bottom of the form is a blue "Login" button. Below the button is the text "No Account? [Register]" where "[Register]" is a red link.

**Figure 6.23.** Blank username for login

If the user tries to log in but leaves an empty field, they will be brought back to the login page, and will be notified that the fields cannot be left blank by a message below the password box.



The screenshot shows a login interface. At the top, the word "Login" is centered. Below it is a horizontal line. There are two input fields: the first is labeled "Username" and contains the text "test"; the second is labeled "Password" and contains seven dots. Below the password field, a red error message reads "Unable to log in with provided credentials." Underneath this message is a blue button with the text "Login" in white. At the bottom, the text "No Account?" is followed by a red link labeled "[Register]" in blue text.

**Figure 6.24.** invalid login for user

If the user enters the wrong login for either the username or password, they will be brought back to the login screen and told that they cannot login with invalid credentials by a message under the password box. They will be able to try to login again.

## 7. Appendix: Changes in Requirements

All the groups conducted a second review with Dr. Hanna Aiman on Oct. 24, 2019. Based on the interview, some requirements are modified and/or added. A list of new/modified requirements are listed below:

1. The Homepage should have an aesthetic appeal. Items that are new, on sale, or of interest to the user (based on user activity) should be displayed below the banner. Furthermore, advertising should be displayed using third party cookies.
2. Commission on sales: 3% commission rate will be taken by the web platform for the first 10 items sold by the vendors, to encourage new sellers to join the eCommerce site. Following the first 10 sold items, the commission rate is increased to 8% for every item purchase.
3. Sellers can reply to reviews (One reply per review) but cannot remove them.
4. Seller cannot sell anything for free.

5. Generate site activity reports (example: number of items sold listed by top sellers, etc)
6. Filters should be implemented to improve User Interface.
7. Password should be encrypted and then stored in database.

## 8. Bibliography

1. G. Alder, “free flowchart maker and diagrams online,” *Flowchart Maker & Online Diagram Software*. [Online]. Available: <https://www.draw.io/>. [Accessed: 08-Nov-2019].
2. G. Chau, A. Kyriakidis, E. Yerburch, S. Drasner, and E. You, “Vue.js,” *Vue.js*. [Online]. Available: <https://vuejs.org/>. [Accessed: 08-Nov-2019].
3. M. Otto and J. Thornton, “Bootstrap,” · *The most popular HTML, CSS, and JS library in the world*. [Online]. Available: <https://getbootstrap.com/>. [Accessed: 08-Nov-2019].