

## ASSIGNMENT 2 FRONT SHEET

<b>Qualification</b>	<b>BTEC Level 5 HND Diploma in Computing</b>		
<b>Unit number and title</b>	10: Website Design & Development		
<b>Submission date</b>		<b>Date Received 1st submission</b>	
<b>Re-submission Date</b>		<b>Date Received 2nd submission</b>	
<b>Student Name</b>	Le Minh Nghia	<b>Student ID</b>	GCD201527
<b>Class</b>	GCD0904	<b>Assessor name</b>	Phan Thanh Tra
<b>Student declaration</b>  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.			
		<b>Student's signature</b>	Nghia

### Grading grid

P5	P6	P7	M4	M5	D2	D3

☐ **Summative Feedback:**

☐ **Resubmission Feedback:**

**Grade:**

**Assessor Signature:**

**Date:**

**Signature & Date:**

## TABLE OF CONTENTS

I.	Creating a design document for a branded, multipage website supported with medium fidelity wireframes and a full set of client and user requirements (P5).....	10
1.	User's requirements by using User Story.....	10
2.	Accessibility standards and guidelines for website.....	11
3.	Use case diagrams.....	13
4.	Wireframes.....	15
5.	Site map.....	20
II.	Using your design document with appropriate principles, standards and guidelines to produce a branded, multipage website supported with realistic content (P6).....	22
1.	Functional screen shot after implementation.....	22
2.	Create/ Read/ Update/ Delete Function.....	69
III.	Comparison and contrast the multipage website created to the design document (M4).....	77
IV.	Critically evaluation the design and development process against your design document and analyze any technical challenges (D2).....	82
1.	Evaluation the design document.....	82
2.	Evaluation the process of development.....	83
3.	The technical challenges/ difficulties in web development.....	84
V.	Creating a suitable Test Plan identifying key performance area and use it to review the functionality and performance of your website. User Experience (UX) and User Interface (UI) (P7).....	86
1.	Test plan.....	86
2.	Test schedule.....	87
3.	Test case.....	88
VI.	Evaluation the Quality Assurance (QA) process and review how it was implemented during your design and development stages (M5)...	95

1. Quality Assurance.....	95
2. Applying QA in my website.....	97
VII. Critically evaluation the results of your Test Plan and include a review of the overall success of your multipage website; use this evaluation to explain any areas of success and provide justified recommendations for areas that require improvement (D3).....	102
1. Evaluation the web.....	102
2. Recommendations for improvements.....	103
3. Advantages, disadvantages of my web project.....	104

## TABLE OF FIGURES

Figure 1: Use case diagram.....	14
Figure 2: Landing page.....	15
Figure 3: Login page.....	16
Figure 4: Register page.....	17
Figure 5: Profile page.....	18
Figure 6: Cart page.....	19
Figure 7: Site map.....	21
Figure 8: Home page.....	22
Figure 9: Code home page – unregistered customer.....	23
Figure 10: Code home page – registered customer.....	24
Figure 11: CSS Home page.....	25
Figure 12: Categories section.....	26
Figure 13: Code category.....	27
Figure 14: CSS Categories.....	28
Figure 15: Introduction section.....	29
Figure 16: Code for 3 books in Introduction section.....	30
Figure 17: Code for slide show.....	31
Figure 18: CSS code for slide show.....	32
Figure 19: Products section.....	33
Figure 20: Code for Products section.....	34
Figure 21: Code for Product section.....	35

Figure 22: Some CSS code for Products section.....	36
Figure 23: Price list section.....	37
Figure 24: Example code of a category of Price list section.....	38
Figure 25: CSS code for Price list section.....	39
Figure 26: Contact section.....	40
Figure 27: A part of code of Contact section.....	41
Figure 28: CSS code for Contact section.....	42
Figure 29: Search book by keyword.....	42
Figure 30: Code for Search function.....	43
Figure 31: PHP code for Search function.....	44
Figure 32: CSS code for search function.....	45
Figure 33: Newsletter section.....	46
Figure 34: Code for Newsletter section.....	46
Figure 35: CSS code for Newsletter section.....	47
Figure 36: Footer.....	48
Figure 37: Code for Footer.....	49
Figure 38: SCSS code for Footer.....	50
Figure 39: Register page.....	51
Figure 40: Code for Register page.....	52
Figure 41: CSS code for Register page.....	53
Figure 42: PHP code for Register page.....	55
Figure 43: Log in page.....	56
Figure 44: Code for Log in page.....	57

Figure 45: CSS code for Log in page.....	58
Figure 46: PHP code for Log in page.....	59
Figure 47: Profile page.....	60
Figure 48: Code for Profile page.....	61
Figure 49: CSS code for Profile page.....	62
Figure 50: PHP code for Profile Page.....	63
Figure 51: Cart page.....	64
Figure 52: Code for Cart page.....	65
Figure 53: CSS code for Cart page.....	66
Figure 54: PHP code for Cart page.....	67
Figure 55: PHP code for Cart page.....	68
Figure 56: Create Account.....	69
Figure 57: Check account.....	70
Figure 58: Pressing Cart logo.....	70
Figure 59: Cart.....	71
Figure 60: Cart before removing the book “Scary Stories” .....	72
Figure 61: Cart after removing the book “Scary Stories” .....	73
Figure 62: Total price before removing the book “Scary Stories” .....	74
Figure 63: Total price after removing the book “Scary Stories” .....	75
Figure 64: Read Profile.....	76
Figure 65: Wireframe for web.....	77
Figure 66: Landing page.....	78
Figure 67: Wireframe for Cart page.....	79

Figure 68: Cart page.....	80
Figure 69: Quality Assurance.....	95
Figure 70: Wireframe and Prototype.....	97
Figure 71: Images.....	98
Figure 72: Content.....	99
Figure 73: Navigation bar.....	100
Figure 74: Performance by Chrome.....	101



## INTRODUCTION

After the introduction of technologies and approaches in Assignment 1, it's time to move on to Assignment 2. Now I'll explain the ideas involved in creating an online book store. The project test plan is presented, as well as the project challenge.

### **I. Creating a design document for a branded, multipage website supported with medium fidelity wireframes and a full set of client and user requirements (P5)**

#### **1. User's requirements by using User Story**

An established book business with a branch in Da Nang wishes to create a website that can showcase items for the purpose of exposing the shop to new customers and accept online orders. Because the business is small, it focuses on selling a limited number of things in small quantities. Based on Facebook and Instagram fan sites, the website has a consumer attraction feature. Furthermore, using the website's map, consumers may be directed to the store and register to become a member of the store.

- As an administrator
  - I want to add products to the website so that customers can choose more.
  - I want to delete low-order products so that update new products.
  - I want to delete and update low-order items to update new categories for customers.
- As a customer
  - I want to see the list of dishes so that know what products I need.
  - I want to know the amount of money to pay for a product so that pay for that product.
  - I want to order online so that I don't have to go to the store in person.
  - I want to register as a member so that receive the latest notifications.

## 2. Accessibility standards and guidelines for website

The Web Content Accessibility Guidelines (WCAG) are produced by web developers, writing tools, and accessibility assessment tools and apply to web content. This implies that websites, tools, and technology are created and built with individuals with disabilities in mind. It's for individuals to perceive, comprehend, navigate, and engage with the Web, as well as contribute to it. It's also good for individuals who aren't disabled, such as those who have "situational restrictions," such as being in direct sunlight or being in a place where they can't hear sound.

### Website design principle

- Perceivable
  - Adaptive: Create material that can be presented in a number of ways without losing information or structure (e.g., a simpler layout). This includes things like correct labeling, taking into account varied screen sizes, and so forth.
  - Distinguishable: Separate the foreground and backdrop to make it easier for people to see and hear content. This covers things like contrast and font size in the user interface, as well as aural issues like background noise.
- Operable
  - Navigation: Provides users with options to browse, find, and locate material. Information architecture, sequence of attention, and proper headers, titles, and labels are all examples of essential user interface considerations.
- Understandable
  - Readable: Make the text simple to read and comprehend. This might be as basic as recognizing human language on a page, or it can be as complex as detecting and interpreting idioms or abbreviations.
  - Predictable: Make Web pages look and function in a consistent manner. This includes component interactions that are predictable and navigation forms that are consistent.
- Robust
  - Compatibility: Compatibility with present and future user agents should be maximized. It is essentially the responsibility of the developers to ensure that the product is constructed in such a way that assistive technologies are supported.
- ❖ Guidelines

- Set up web page format: You'll need to figure out what kind of website you're making. If the website is an e-commerce one. It will focus on product introductions, have a lot of engagement with customers when it comes to purchasing, and visitors will be able to contact the Shop using the information on the landing page.
- Set goals: Make a list of the website's most significant and vital goals. Make a particular list or list the most fundamental items, and if anything else comes up while you're developing a website, you'll need to add it to the list.
- Website requirements: Ask questions like a customer
  - How many pages should the website be built?
  - Does the website need to link to social networking sites like Facebook or Instagram?
  - Can the site be registered as a member?
  - Is it possible to order online from the site?
- Building basic content: One of the most crucial aspects that distinguishes your website from others is its content. The content of a website is what determines its success. Many individuals believe that simply creating a website with a pleasing layout, improving loading speed, and offering quality items is sufficient, but these are only auxiliary elements. The most essential thing for a new website is to create content, but most of us still don't grasp the true value of content, therefore producing content for the website is still a bit of a grey area.
- Website promotion: Businesses may use media to market their products and services on their websites, bringing their pictures and branding closer to customers.
- Development Website: Once your website is up and running. If you believe it simply functions properly, you are mistaken. You should continually research and enhance your website so that customers are pleased and comfortable when they visit it. You should also supply your consumers with content that is really useful. These are the aspects that will help your website attract and retain a significant number of visitors.

### 3. Use case diagrams

- Definition: The basic approach for modeling the functional needs of a software system is the use case. Text or illustrations can be used to explain use cases. When a Use Case model represents a system in the most intuitive and intelligible way for all users, it is effective.
- Project use case: A use case diagram will be provided below to explain the relationship and function between the client and the administrator (web owner). Customers are those who come to browse the items and place orders, while the admin is the website administrator. The admin's job is to replace low-order-volume goods and categories with new products and categories. After successful registration, check the order details on the client side and view the customer's information. On the client side, there will be two categories: registered and unregistered customers. Registered customers may see and modify profiles, view goods, place orders, and make payments, while unregistered customers can only view products.

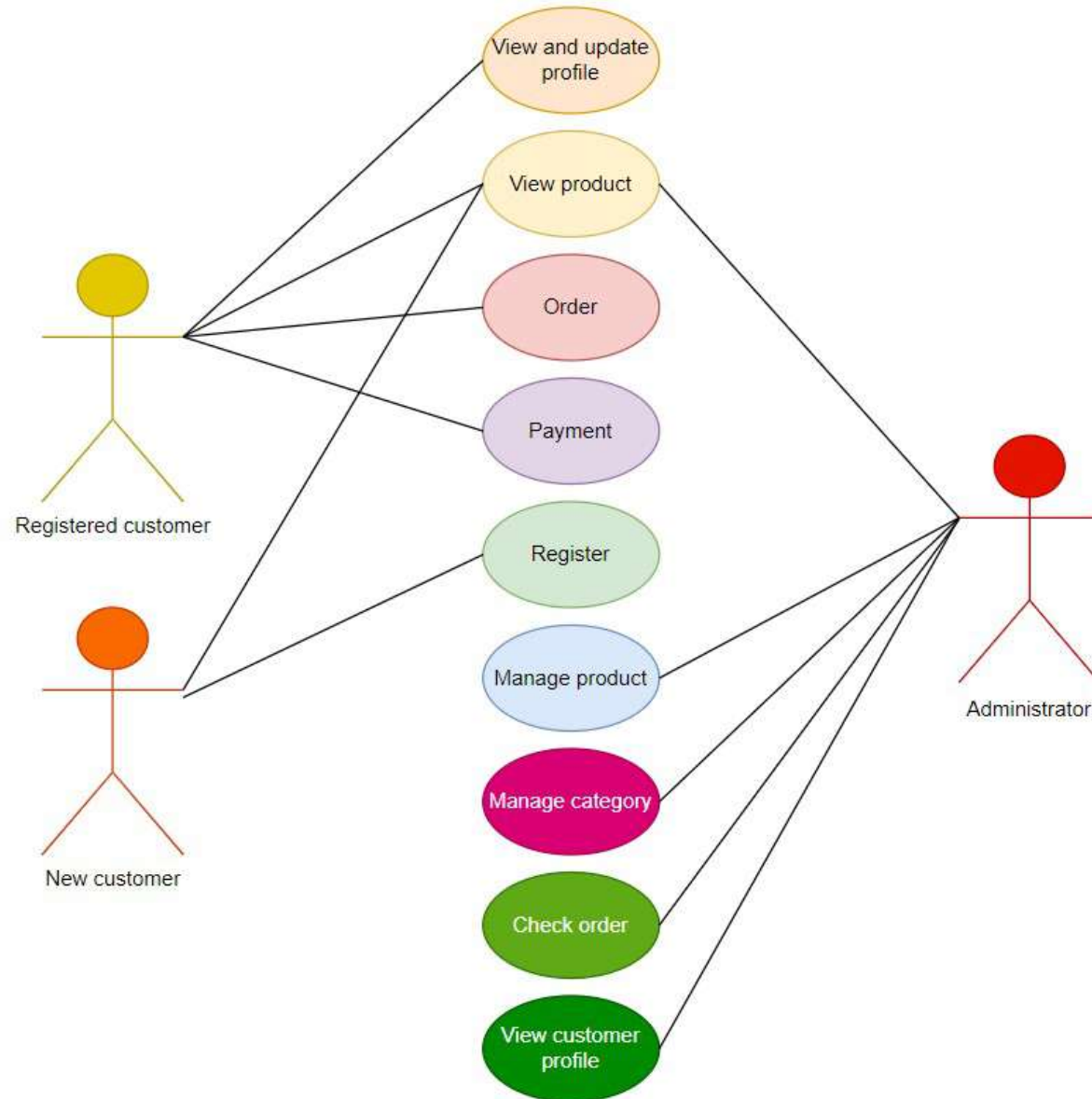


Figure 1: Use case diagram

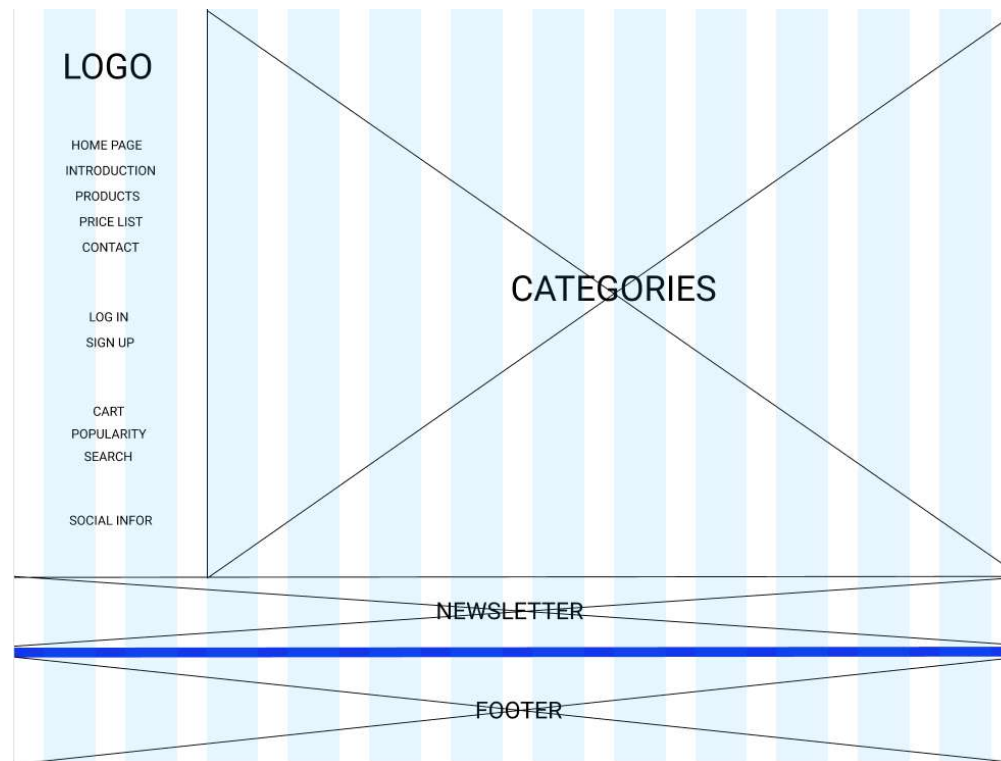
## 4. Wireframes

### a. Definition

A wireframe is a two-dimensional illustration of a page's interface that specifically focuses on space allocation and prioritization of content, functionalities available, and intended behaviors. For these reasons, wireframes typically do not include any styling, color, or graphics. Wireframes also help establish relationships between a website's various templates.

### b. Wireframes for website

#### Landing page



*Figure 2: Landing page*

## Login page

LOG IN

USER NAME

PASSWORD

LOG IN

*Figure 3: Login page*

## Register page

~~SIGN UP~~

USER NAME

PASSWORD

CONFIRM PASSWORD

FULL NAME

PHONE NUMBER

ADDRESS

~~SIGN UP~~ ~~CLEAR~~

*Figure 4: Register page*



## Profile page

UPDATE INFORMATION

FULL NAME

PHONE NUMBER

ADDRESS

UPDATE CLEAR

*Figure 5: Profile page*

NAME	PRICE	QUANTITY

CART SUBTOTAL

TOTAL PRICE

TOTAL

DELIVERY

**PAYMENT**

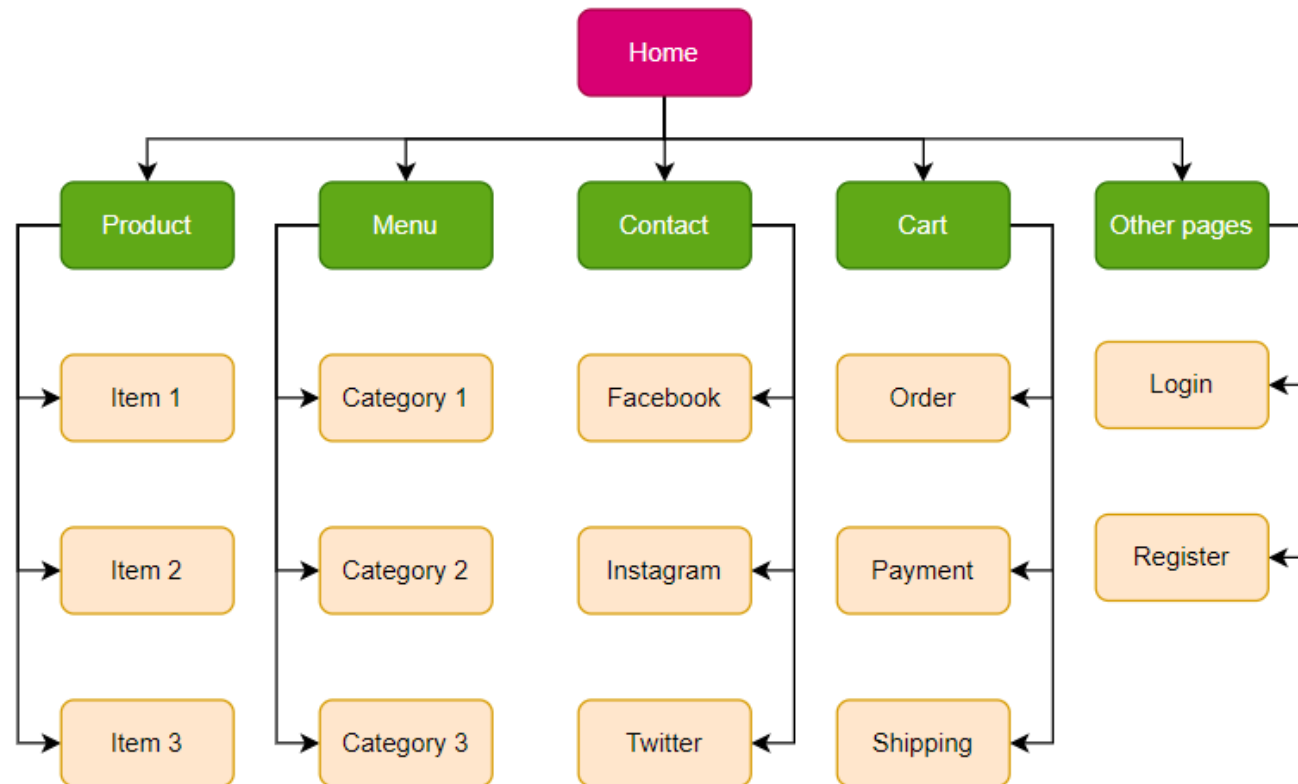
## 5. Site map

### a. Definition

The term "sitemap" refers to a map of the website. Sitemaps identify and succinctly define the many types of information on your website. A sitemap is essential for every website since it allows users and crawlers to navigate through the website's links. The user-facing sitemap is known as an HTML site map, whereas the spider-facing sitemap is known as an XML site map. Sitemaps are highly vital, and search engines would appreciate websites with a navigation map, which is very handy for bots to index and is quite advantageous in SEO.

- Advantages
  - Improved website planning.
  - Forward-thinking development.
  - A Sitemap has the ability of listing all of the URLs that you have in your website

**b. Site map for website**



*Figure 7: Site map*

Customers initially come to our home page, where they may find information about special events, product categories, and product details. The user then travels to the home page, where all product information is recorded, by clicking on the picture of each product. The user then picks the product, as well as the product page information. The location that most accurately represents your goods. If the user is happy with the purchase, he or she can add it to their shopping cart; otherwise, they can delete it. Return to the home page for additional information. If this is not the case, direct the user to the check-out information. We will dispatch the product to you after the information has been completed.

## II. Using your design document with appropriate principles, standards and guidelines to produce a branded, multipage website supported with realistic content (P6)

### 1. Functional screen shot after implementation

#### Home page

The homepage is the introductory and web page that includes the store's logo name and a navigation bar at left side containing categories of book, cart, login, register, introduction, contact information, homepage. It is designed in HTML, CSS, SCSS, PHP.

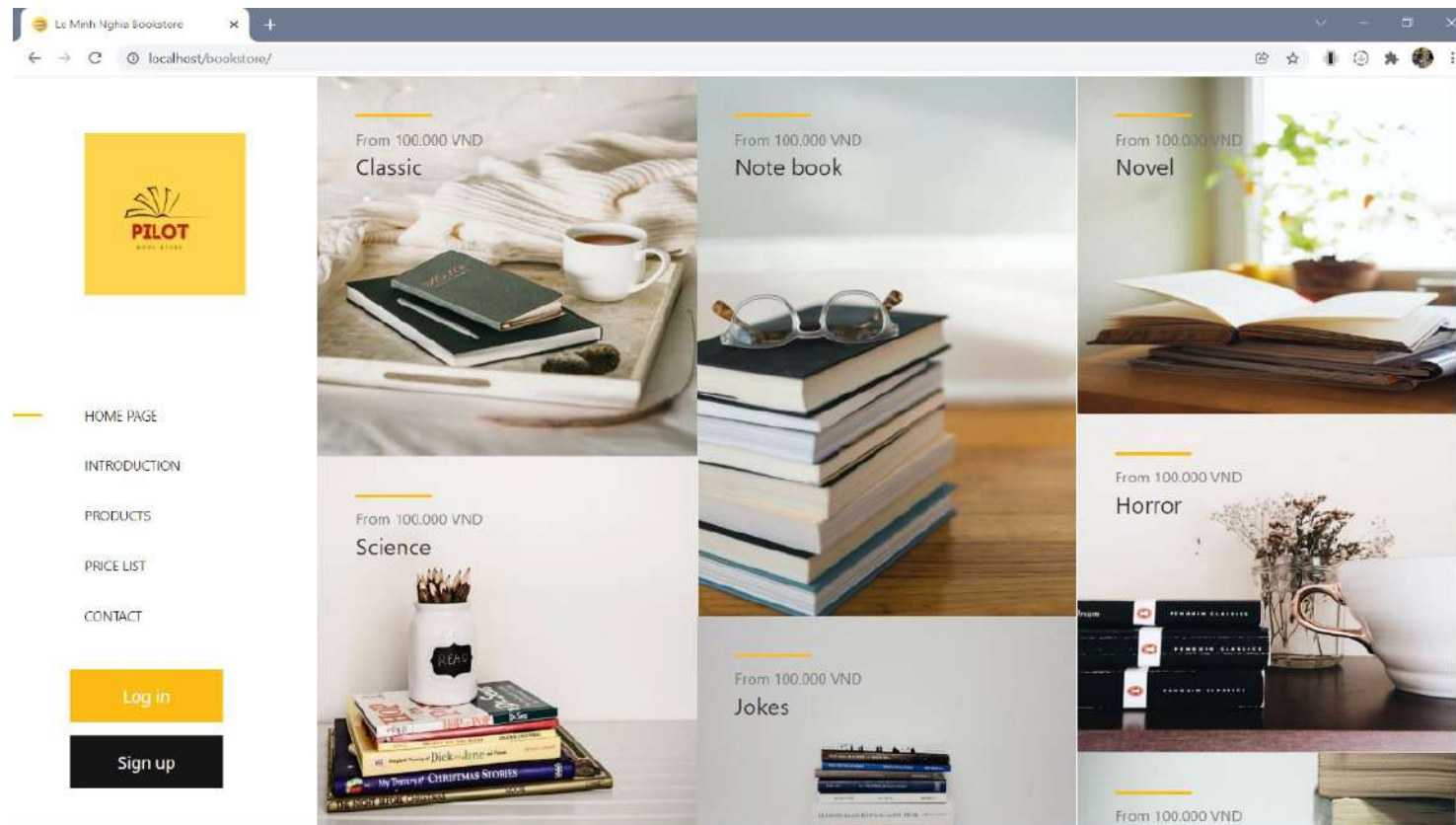


Figure 8: Home page

```
<!-- Logo -->
<div class="logo">
  <a href="index.php"></a>
</div>
<!-- Amado Nav -->
<nav class="amado-nav">
  <ul>
    <li class="active"><a href="index.php">Home page</a></li>
    <li><a href="intro.php">Introduction</a></li>
    <li><a href="product.php?category=horror">Products</a></li>
    <li><a href="price-list.php">Price list</a></li>
    <li><a href="contact.php">Contact</a></li>
  </ul>
</nav>
<!-- Button Group -->
<div class="amado-btn-group mt-30 mb-100">
  <div class="row">
    <a href="login.php" class="btn amado-btn mb-15">
      <?php if (isLoggedIn()) { echo 'style="display:none;"'; } ?>Log in</a>
    <a href="register.php" class="btn amado-btn active">
      <?php if (isLoggedIn()) { echo 'style="display:none;"'; } ?>Sign up</a>
  </div>
  <div class="row">...
</div>
<!-- Cart Menu -->
<div class="cart-fav-search mb-100">
  <a href="cart.php" class="cart-nav"> Cart
  <span></span></a>
  <a href="#" class="fav-nav"> Popularity</a>
  <a href="#" class="search-nav"> Search</a>
</div>
<!-- Social Button -->
<div class="social-info d-flex justify-content-between">
  <a href="#"><i class="fa fa-pinterest" aria-hidden="true"></i></a>
  <a href="#"><i class="fa fa-instagram" aria-hidden="true"></i></a>
  <a href="#"><i class="fa fa-facebook" aria-hidden="true"></i></a>
  <a href="#"><i class="fa fa-twitter" aria-hidden="true"></i></a>
</div>
```

Figure 9: Code home page – unregistered customer

This is the code to display each component of the home page depending on the status/ type of customer, the registered and unregistered one. I used function named `isNotLoggedIn` in PHP to check type of customer, then screen will show interface according to it. Above picture is showing that user has not registered account yet.

```
<div class="row">
  <div class="col-2">
    <i class="fa fa-3x fa-user" <?php if (isNotLoggedIn()){ echo 'style="display:none;"; } ?></i>
  </div>
  <div class="col-10">
    <a class="dropdown-item" href="reset-password.php"
      <?php if (isNotLoggedIn()){ echo 'style="display:none;"; } ?>>Welcome,
      <b><?php echo htmlspecialchars($_SESSION["username"]); ?></b></a>
    <div class="dropdown" <?php if (isNotLoggedIn()){ echo 'style="display:none;"; } ?>>
      <a class="dropdown-item dropdown-toggle" href="#" id="dropdownMenuLink"
        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" <?php if (isNotLoggedIn()){ echo 'style="display:none;"; } ?>>
        Account
      </a>

      <div class="dropdown-menu" aria-labelledby="dropdownMenuLink">
        <a class="dropdown-item" href="edit-user.php"
          <?php if (isAdmin()){ echo 'style="display:none;"; } ?>>Edit profile</a>
        <a class="dropdown-item" href="changepw.php"
          <?php if (isAdmin()){ echo 'style="display:none;"; } ?>>Change password</a>
        <a class="dropdown-item" href="/admin/dashboard.php"
          <?php if (!isAdmin()){ echo 'style="display:none;"; } ?>>Management</a>
        <div class="dropdown-divider"></div>
        <a class="dropdown-item" href="logout.php"
          <?php if (isNotLoggedIn()){ echo 'style="display:none;"; } ?>>Log out</a>
      </div>
    </div>
  </div>
</div>
```

*Figure 10: Code home page – registered customer*

This is code for user who registered account. Besides that, I will compare to admin in order to provide reasonable function for user.

Function `isAdmin` will check user (admin or customer).

This is some of the CSS code to style the home page

```
.main-content-wrapper {
  position: relative;
  z-index: 1;
  @include trans-duration(500ms);
  @include flex-wrap(wrap);
  .header-area {
    @include trans-duration(500ms);
    position: relative;
    z-index: 1;
    @include flex(0 0 320px);
    width: 320px;
    max-width: 320px;
    background-color: #fff;
    padding: 60px 75px;
    .nav-close {
      @include trans-duration(500ms);
      width: 40px;
      height: 40px;
      position: absolute;
      top: 0;
      right: 15px;
      background-color: red;
      text-align: center;
      color: #fff;
      line-height: 40px;
      font-size: 12px;
      cursor: pointer;
      background-color: $bg-default;
      display: none;
      &:hover {
        background-color: $dark-color;
      }
    }
  }
  .logo {
    margin-bottom: 100px;
    display: block;
  }
}
```

Figure 11: CSS Home page



## Categories

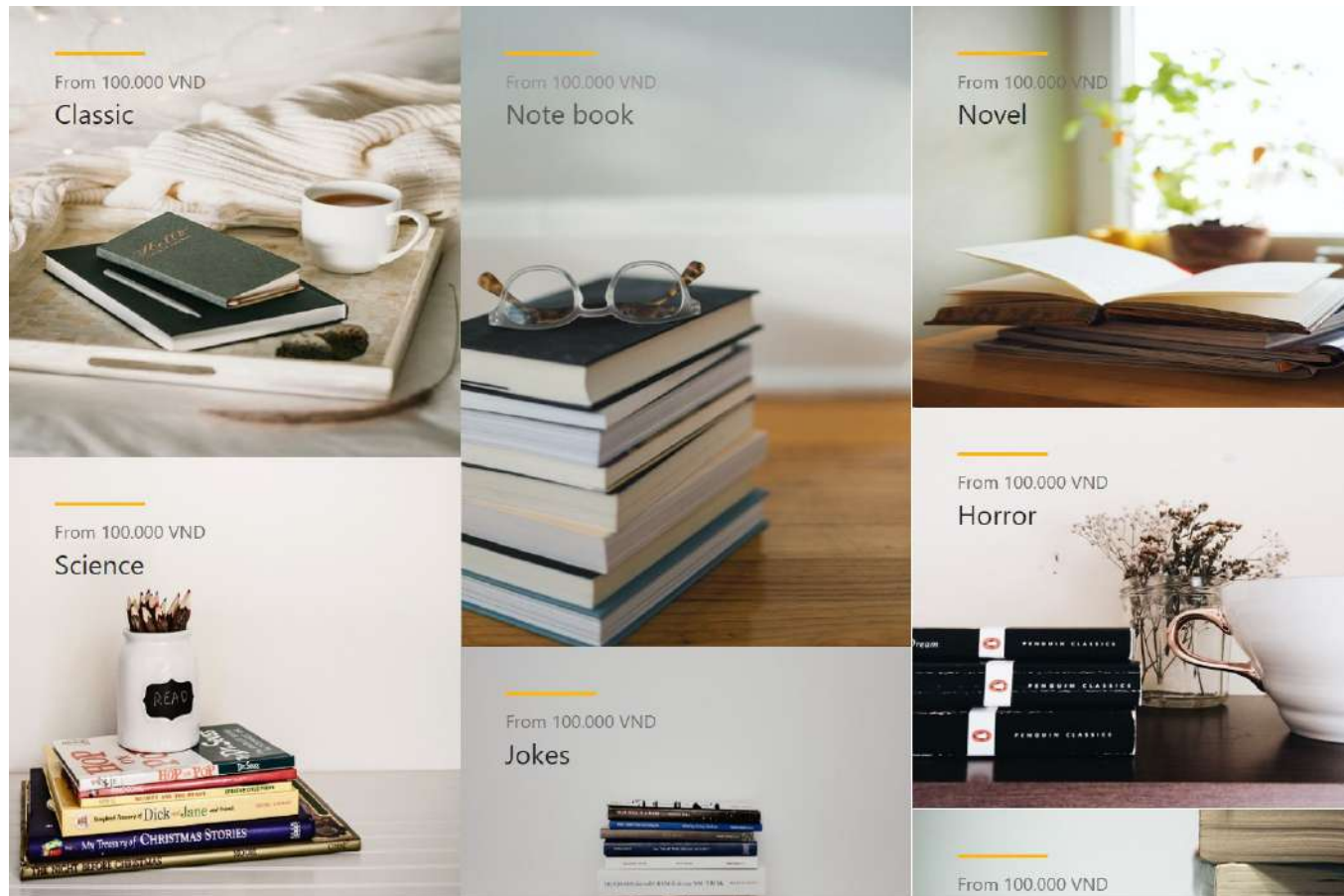


Figure 12: Categories section

In the Categories section, I divided into 9 single categories: Classic, Note book, Novel, Science, Jokes, Horror, Fiction, Formula, Research belong to price. Customer clicks one of categories to choose book. Single category displays information including name, price and image

```
<!-- Single Catagory -->
<div class="single-products-catagory clearfix">
  <a href="product.php?category=horror">
    
    <div class="hover-content">
      <div class="line"></div>
      <p>From 100.000 VND</p>
      <h4>Classic</h4>
    </div>
  </a>
</div>

<!-- Single Catagory -->
<div class="single-products-catagory clearfix">
  <a href="product.php?category=horror">
    
    <div class="hover-content">
      <div class="line"></div>
      <p>From 100.000 VND</p>
      <h4>Note book</h4>
    </div>
  </a>
</div>

<!-- Single Catagory -->
<div class="single-products-catagory clearfix">
  <a href="product.php?category=novel">
    
    <div class="hover-content">
      <div class="line"></div>
      <p>From 100.000 VND</p>
      <h4>Novel</h4>
    </div>
  </a>
</div>
```

Figure 13: Code category

This is some of the CSS code to style the categories

```
.single-products-category {  
  position: relative;  
  z-index: 1;  
  width: 33.33333%;  
  top: 0;  
  a {  
    display: block;  
    width: 100%;  
    height: 100%;  
    position: relative;  
    z-index: 10;  
    &::after {  
      @include trans-duration(250ms);  
      position: absolute;  
      width: 100%;  
      height: 100%;  
      top: 0;  
      left: 0;  
      content: '';  
      background-color: rgba(60, 60, 60, 0.7);  
      z-index: 1;  
      opacity: 0;  
      visibility: visible;  
    }  
    img {  
      width: 100%;  
    }  
  }  
}
```

Figure 14: CSS Categories

## Introduction

Introduction section will display some interested experiences when customer uses our book store. Besides that, it shows some popular book in the world and details.



The Old Man and the



A Brief History of Time

Figure 15: Introduction section

This is CSS code for 3 books in Introduction including image, name and summarize of book

```
<div class="row" style="padding-top: 50px;">
  <div class="col-lg-4">
    
    <h2>The Old Man and the Sea</h2>
    <p>Bringing great value to readers, including extremely useful tips on how to behave in daily life. The spread of the book is extremely wide - with 1
    <p><a class="btn btn-secondary" href="product-detail.php?id=14" role="button">Details... </a></p>
  </div><!-- /.col-lg-4 -->
  <div class="col-lg-4">
    The Thorn Birds</h2>
    <p>Do you want to take a shortcut?, Do you want to be on the road to success?, The recipe for this, according to connection guru Keith, is getting t
    <p><a class="btn btn-secondary" href="product-detail.php?id=13" role="button">Details... </a></p>
  </div><!-- /.col-lg-4 -->
  <div class="col-lg-4">
    
    <h2>A Brief History of Time</h2>
    <p>Along with the Universe in a Chestnut Shell, A Brief History of Time is considered the most famous and popular book on cosmology by Stephen Hawki
    <p><a class="btn btn-secondary" href="product-detail.php?id=19" role="button">Details... </a></p>
  </div><!-- /.col-lg-4 -->
</div>
```

Figure 16: Code for 3 books in Introduction section

```
<div id="product_details_slider" class="carousel slide" data-ride="carousel">
  <ol class="carousel-indicators">
    <li class="active" data-target="#product_details_slider" data-slide-to="0" style="background-image: url(img/product-img/pro-big-1.jpg);">
    </li>
    <li data-target="#product_details_slider" data-slide-to="1" style="background-image: url(img/product-img/pro-big-2.jpg);">
    </li>
    <li data-target="#product_details_slider" data-slide-to="2" style="background-image: url(img/product-img/pro-big-3.jpg);">
    </li>
    <li data-target="#product_details_slider" data-slide-to="3" style="background-image: url(img/product-img/pro-big-4.jpg);">
    </li>
  </ol>
  <div class="carousel-inner">
    <div class="carousel-item active">
      <a class="gallery_img" href="img/intro-img/1.jpg">
        
        <div class="carousel-caption">
          <h3 style="color:white">Welcome to our bookstore!</h3>
        </div>
      </a>
    </div>
    <div class="carousel-item">
      <a class="gallery_img" href="img/intro-img/2.jpg">
        
        <div class="carousel-caption">
          <h3 style="color:white">Experience sensations that have never been there before</h3>
        </div>
      </a>
    </div>
    <div class="carousel-item">
      <a class="gallery_img" href="img/intro-img/3.jpg">
        
        <div class="carousel-caption">
          <h3 style="color:white">Read books online on PC, on phone</h3>
        </div>
      </a>
    </div>
  </div>
</div>
```

Figure 17: Code for slide show

Slide show web is where we promote the importance roles of book store. Designed by Bootstrap Carousel Plugin

This is CSS code for slide show

```
.carousel {  
  position: relative;  
}  
  
.carousel-inner {  
  position: relative;  
  width: 100%;  
  overflow: hidden;  
}  
  
.carousel-item {  
  position: relative;  
  display: none;  
  -ms-flex-align: center;  
  align-items: center;  
  width: 100%;  
  transition: -webkit-transform 0.6s ease;  
  transition: transform 0.6s ease;  
  transition: transform 0.6s ease, -webkit-transform 0.6s ease;  
  -webkit-backface-visibility: hidden;  
  backface-visibility: hidden;  
  -webkit-perspective: 1000px;  
  perspective: 1000px;  
}  
  
@media screen and (prefers-reduced-motion: reduce) {  
  .carousel-item {  
    transition: none;  
  }  
}
```

Figure 18: CSS code for slide show



## Products

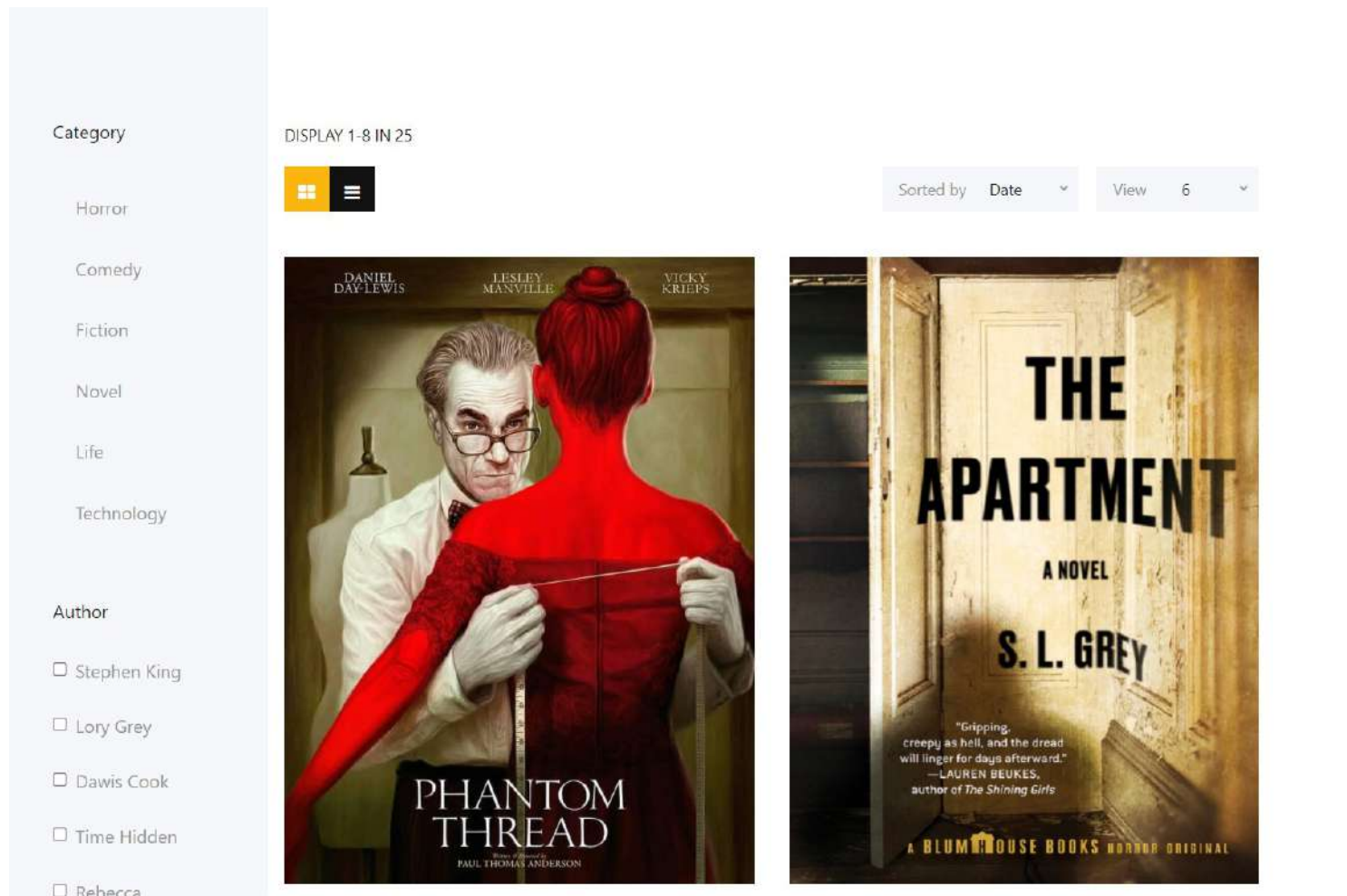


Figure 19: Products section

The product section will contain pictures, names of books and prices. It is designed in HTML, CSS and PHP



```
<div class="categories-menu">
  <ul>
    <li><a href="product.php?category=horror">Horror</a></li>
    <li><a href="product.php?category=comedy">Comedy</a></li>
    <li><a href="product.php?category=fiction">Fiction</a></li>
    <li><a href="product.php?category=novel">Novel</a></li>
    <li><a href="product.php?category=life">Life</a></li>
    <li><a href="product.php?category=tech">Technology</a></li>
  </ul>
</div>
</div>

<!-- Single widget -->
<div class="widget brands mb-50">
  <!-- Widget Title -->
  <h6 class="widget-title mb-30">Author</h6>

  <div class="widget-desc">
    <!-- Single Form Check -->
    <div class="form-check">
      <input class="form-check-input" type="checkbox" value="" id="amado">
      <label class="form-check-label" for="amado">Stephen King</label>
    </div>
    <!-- Single Form Check -->
    <div class="form-check">
      <input class="form-check-input" type="checkbox" value="" id="ikea">
      <label class="form-check-label" for="ikea">Lory Grey</label>
    </div>
    <!-- Single Form Check -->
    <div class="form-check">
      <input class="form-check-input" type="checkbox" value="" id="furniture">
      <label class="form-check-label" for="furniture">Dawis Cook</label>
    </div>
    <!-- Single Form Check -->
    <div class="form-check">
      <input class="form-check-input" type="checkbox" value="" id="factory">
      <label class="form-check-label" for="factory">Time Hidden</label>
    </div>
  </div>
</div>
```

Figure 20: Code for Products section

I used input type checkbox for customer to choose one or many authors in the same time. It is convenient for customer to buy books

Here is code for price option of book. And sorted book by date, earliest, popularity.

```
<h6 class="widget-title mb-30">Price</h6>

<div class="widget-desc">
  <div class="slider-range">
    <div data-min="10" data-max="1000" data-unit="VND"
      class="slider-range-price ui-slider ui-slider-horizontal ui-widget ui-widget-content ui-corner-all"
      data-value-min="10" data-value-max="1000" data-label-result="">
    <div class="ui-slider-range ui-widget-header ui-corner-all"></div>
    <span class="ui-slider-handle ui-state-default ui-corner-all" tabindex="0"></span>
    <span class="ui-slider-handle ui-state-default ui-corner-all" tabindex="0"></span>
    </div>
    <div class="range-price">10.000 VND - 1.0000.000 VND</div>
  </div>
</div>
</div>
</div>

<div class="amado_product_area section-padding-100">
  <div class="container-fluid">
    <div class="row">
      <div class="col-12">
        <div class="product-topbar d-xl-flex align-items-end justify-content-between">
          <!-- Total Products -->
          <div class="total-products">
            <p>Display 1-8 in 25</p>
            <div class="view d-flex">
              <a href="#"><i class="fa fa-th-large" aria-hidden="true"></i></a>
              <a href="#"><i class="fa fa-bars" aria-hidden="true"></i></a>
            </div>
          </div>
          <!-- Sorting -->
          <div class="product-sorting d-flex">
            <div class="sort-by-date d-flex align-items-center mr-15">
              <p>Sorted by</p>
              <form action="#" method="get">
                <select name="select" id="sortBydate">
                  <option value="value">Date</option>
                  <option value="value">New</option>
                </select>
              </form>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

Figure 21: Code for Product section

```
.widget {  
  position: relative;  
  z-index: 1;  
}  
  
.widget .widget-title {  
  font-size: 16px;  
  text-transform: capitalize;  
  color: #242424;  
}  
  
.widget .catagories-menu li a {  
  text-transform: capitalize;  
  font-size: 16px;  
  padding: 15px 0 15px 20px;  
  display: block;  
  font-weight: 400;  
  color: #959595;  
}  
  
.widget .catagories-menu li a:hover,  
.widget .catagories-menu li a:focus {  
  color: #fbb710;  
}  
  
.widget .catagories-menu li.active a {  
  color: #fbb710;  
}  
  
.widget.price .slider-range-price {  
  background-color: #c7d2d8 !important;  
  height: 2px;  
  border: none;  
}  
  
.widget.price .ui-slider-handle {  
  background-color: #fff;  
  top: -7px;  
  width: 18px;
```

Figure 22: Some CSS code for Products section

## Price list

RETAIL CUSTOMER	USER CUSTOMER	WHOLESALE CUSTOMERS
~ 100.000 VND /1 book	~ 90.000 VND /1 book	~ 60.000 VND /1 book
3 PRODUCTS Each day	5 PRODUCTS Each day	Unlimited PRODUCTS Each day
~50.000 VND Shipping fee	~20.000 VND Shipping fee	Free Shipping fee
No Sales promotion	Yes Sales promotion	Yes Sales promotion
3 months Insurance	6 months Insurance	1 year Insurance
24/7 Support	24/7 Support	24/7 Support
<a href="#">BUY NOW</a>	<a href="#">REGISTER</a>	<a href="#">REGISTER</a>

Figure 23: Price list section

There are 3 options for customer choosing. Including price of a book and benefits of each option.

```
<div class="head_bg"></div>
<div class="head">
  <span>Retail customer</span>
</div>
<!--//HEAD END-->

</div>
<!--//HEAD CONTENT END-->

<!--PRICE START-->
<div class="generic_price_tag clearfix">
  <span class="price">
    <span class="sign">~</span>
    <span class="currency">100</span>
    <span class="cent">.000 VND</span>
    <span class="month">/1 book</span>
  </span>
</div>
<!--//PRICE END-->

</div>
<!--//HEAD PRICE DETAIL END-->

<!--FEATURE LIST START-->
<div class="generic_feature_list">
  <ul>
    <li><span>3 PRODUCTS</span> Each day</li>
    <li><span>~50.000 VND</span> Shipping fee</li>
    <li><span>No</span> Sales promotion</li>
    <li><span>3 months</span> Insurance</li>
    <li><span>24/7</span> Support</li>
  </ul>
</div>
```

Figure 24: Example code of a category of Price list section

This is CSS for Price list section

```
#generic_price_table .generic_content .generic_head_price .generic_price_tag{
    padding: 0 0 20px;
}

#generic_price_table .generic_content .generic_head_price .generic_price_tag .price{
    display: block;
}

#generic_price_table .generic_content .generic_head_price .generic_price_tag .price .sign{
    display: inline-block;
    font-family: "Lato",sans-serif;
    font-size: 28px;
    font-weight: 400;
    vertical-align: middle;
}

#generic_price_table .generic_content .generic_head_price .generic_price_tag .price .currency{
    font-family: "Lato",sans-serif;
    font-size: 60px;
    font-weight: 300;
    letter-spacing: -2px;
    line-height: 60px;
    padding: 0;
    vertical-align: middle;
}

#generic_price_table .generic_content .generic_head_price .generic_price_tag .price .cent{
    display: inline-block;
    font-family: "Lato",sans-serif;
    font-size: 24px;
    font-weight: 400;
    vertical-align: bottom;
}
```

Figure 25: CSS code for Price list section

## Contact section

### Contact

Name

Email

Topic

Message

Send

### Contact information

Do not hesitate to contact us immediately via:





-  Address: LeMinhNghia - Ba Trieu Street, Xuan Phu Ward, Hue City
-  Phone: +8429072002 (Free)
-  Email: [nghialmgcd201527@fpt.edu.vn](mailto:nghialmgcd201527@fpt.edu.vn)
-  Website: <https://holidaywithnghia.herokuapp.com/index.html>

Figure 26: Contact section

In order to knowing more details about our book store or giving comment about our quality, I create Contact page for customer to enter their information, our book store will contact them by email.

```
<div class="col-md-6 d-flex align-items-stretch">
  <div class="info-wrap w-100 p-md-5 p-4 py-5 img">
    <h3>Contact information</h3>
    <p class="mb-4">Do not hesitate to contact us immediately via:</p>
    <div class="dbox w-100 d-flex align-items-start">
      <div class="icon d-flex align-items-center justify-content-center">
        <span class="fa fa-map-marker"></span>
      </div>
      <div class="text pl-3">
        <p style="color:black;"><span>Address:</span> LeMinhNghia - Ba Trieu Street, Xuan Phu Ward, Hue City</p>
      </div>
    </div>
    <div class="dbox w-100 d-flex align-items-center">
      <div class="icon d-flex align-items-center justify-content-center">
        <span class="fa fa-phone"></span>
      </div>
      <div class="text pl-3">
        <p><span>Phone:</span> <a href="tel://1234567920">+8429072002 (Free)</a></p>
      </div>
    </div>
    <div class="dbox w-100 d-flex align-items-center">
      <div class="icon d-flex align-items-center justify-content-center">
        <span class="fa fa-paper-plane"></span>
      </div>
      <div class="text pl-3">
        <p><span>Email:</span> <a href="mailto:info@yoursite.com">nghialmgcd201527@fpt.edu.vn</a></p>
      </div>
    </div>
    <div class="dbox w-100 d-flex align-items-center">
      <div class="icon d-flex align-items-center justify-content-center">
        <span class="fa fa-globe"></span>
      </div>
      <div class="text pl-3">
        <p><span>Website:</span> <a href="#">https://holidaywithnghia.herokuapp.com/index.html</a></p>
      </div>
    </div>
  </div>
```

Figure 27: A part of code of Contact section



```
.wrapper {  
  width: 100%;  
  overflow: hidden;  
  border-radius: 10px;  
  -webkit-box-shadow: 0px 21px 41px -13px rgba(0, 0, 0, 0.18);  
  -moz-box-shadow: 0px 21px 41px -13px rgba(0, 0, 0, 0.18);  
  box-shadow: 0px 21px 41px -13px rgba(0, 0, 0, 0.18); }  
  
.contact-wrap {  
  background: #e7e9eb; }  
  .contact-wrap h3 {  
    color: black; }
```

Figure 28: CSS code for Contact section

## Search function



A search bar interface with a light blue background. In the top right corner, there is a small orange square button with a white 'x' icon. Below this, a white search input field is centered. Inside the input field, on the left, is a magnifying glass icon, followed by the placeholder text "Enter keyword for book".

Figure 29: Search book by keyword

There is a form for customer entering keyword to searching book. I put a logo of search in the top-right corner.

```
<div class="search-wrapper section-padding-100">
  <div class="search-close">
    <i class="fa fa-close" aria-hidden="true"></i>
  </div>
  <div class="container">
    <div class="row">
      <div class="col-12">
        <div class="search-content">
          <form action="search.php" method="post">
            <input class="form-control" type="text" name="search" id="search"
              placeholder="Enter keyword for book">
            <button class="form-control" type="submit"></button>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
```

Figure 30: Code for Search function

```
<?php
    $set=$_POST['search'];
    if($set){
        $result = mysqli_query($con,"SELECT * FROM products WHERE name LIKE '%$set%' ");
        $result1 = mysqli_query($con,"SELECT * FROM products WHERE author LIKE '%$set%' ");
        $result2 = mysqli_query($con,"SELECT * FROM products WHERE category LIKE '%$set%' ");
        while($row = mysqli_fetch_assoc($result)){
            echo
```

*Figure 31: PHP code for Search function*

Customer can search some book by keywords which are contained in name or author or category of book

I used SCSS to write a code for Search function. This is a faster way to design Search function. Because I want to display Search function at the top of page, I put a fixed position for it.

```
.search-wrapper {  
  position: fixed;  
  z-index: 1;  
  width: 100%;  
  height: 260px;  
  z-index: 5000;  
  top: -350px;  
  left: 0;  
  background-color: #f5f7fa;  
  @include trans-duration(500ms);  
  .search-content {  
    form {  
      position: relative;  
      z-index: 1;  
      box-shadow: 0 0 5px rgba(0, 0, 0, 0.075);  
      input {  
        width: 100%;  
        height: 60px;  
        border: none;  
        padding-left: 50px;  
        font-size: 14px;  
        color: #6b6b6b;  
      }  
      button {  
        position: absolute;  
        top: 0;  
        left: 0;  
        border: none;  
        width: 50px;  
        height: 60px;  
        z-index: 10;  
        cursor: pointer;  
        background-color: transparent;  
        text-align: center;  
      }  
    }  
  }  
}
```

Figure 32: CSS code for search function

## Newsletter section

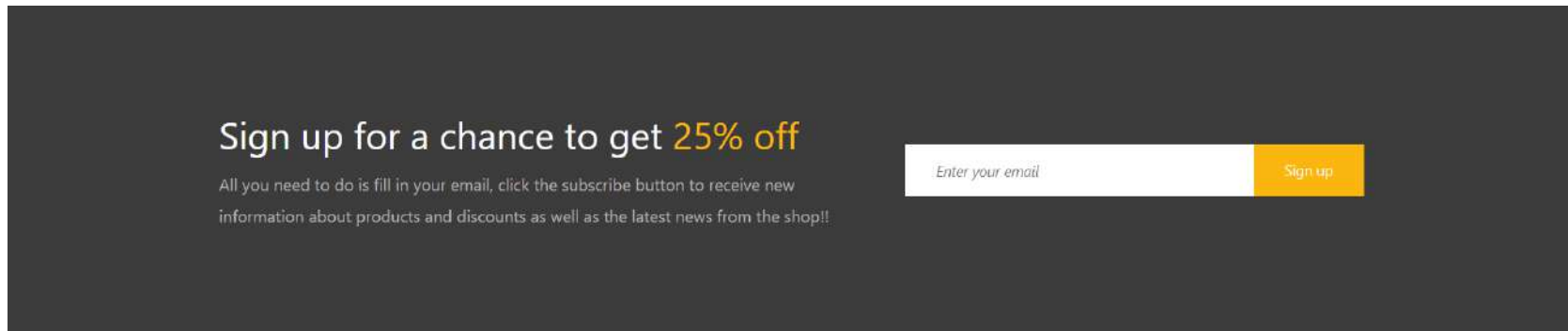


Figure 33: Newsletter section

In order to making customer be impressed by our book store, I put a newsletter section to attract more customer to sign up account in our book store.

```
<section class="newsletter-area section-padding-100-0">
  <div class="container">
    <div class="row align-items-center">
      <!-- Newsletter text -->
      <div class="col-12 col-lg-6 col-xl-7">
        <div class="newsletter-text mb-100">
          <h2>Sign up for a chance to get <span>25% off </span></h2>
          <p>All you need to do is fill in your email, click the subscribe button to receive new information
            about products and discounts as well as the latest news from the shop!!</p>
        </div>
      </div>
      <!-- Newsletter form -->
      <div class="col-12 col-lg-6 col-xl-5">
        <div class="newsletter-form mb-100">
          <form action="#" method="post">
            <input id="mail-check" type="email" name="email" class="nl-email"
              placeholder="Enter your email">
            <input type="submit" onclick="showMess()" value="sign up">
          </form>
        </div>
      </div>
    </div>
  </div>
</section>
```

Figure 34: Code for Newsletter section

This is SCSS code for Newsletter section

```
.newsletter-area {
  position: relative;
  z-index: 1;
  background-color: #3c3c3c;
  .newsletter-text {
    position: relative;
    z-index: 1;
    h2 {
      font-size: 36px;
      margin-bottom: 10px;
      color: #fff;
      span {
        color: $bg-default;
      }
    }
    p {
      color: #b3b1b1;
      margin-bottom: 0;
    }
  }
  .newsletter-form {
    form {
      position: relative;
      z-index: 1;
      input[type="email"] {
        width: 100%;
        height: 50px;
        background-color: #fff;
        color: #afafaf;
        font-size: 14px;
        font-style: italic;
        border: none;
        padding: 0 30px;
      }
      input[type="submit"] {
        @include trans-duration(500ms);
        height: 50px;
        background-color: $bg-default;
        color: #fff;
      }
    }
  }
}
```

Figure 35: CSS code for Newsletter section

## Footer

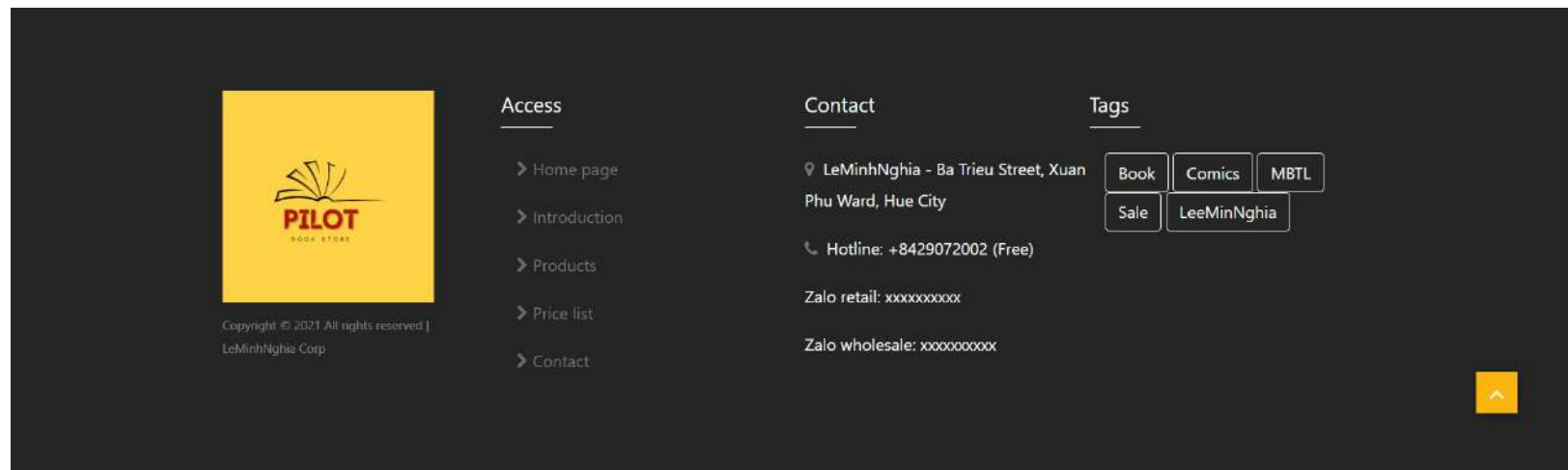


Figure 36: Footer

```
<footer class="footer_area clearfix">
  <div class="container">
    <div class="row">
      <div class="col-md-3">
        <div class="single_widget_area">
          <!-- Logo -->
          <div class="footer-logo mr-50">
            <a href="index.php"></a>
          </div>
          <!-- Copywrite text -->
          <p class="copywrite">
            Copyright &copy;
            <script>
              document.write(new Date().getFullYear());
            </script> All rights reserved | LeMinhNghia Corp
          </p>
        </div>
      </div>
      <div class="col-md-3">
        <div class="row">
          <h5 style="color: white;">Access</h5>
        </div>
        <div class="row mb-4">
          <div class="underline bg-light" style="width: 50px;"></div>
        </div>

        <p><i class="fa fa-chevron-right" aria-hidden="true"></i> <a style="color:#777; font-size:100%;"
          href="index.php">Home page</a></p>
        <p><i class="fa fa-chevron-right" aria-hidden="true"></i> <a style="color:#777; font-size:100%;"
          href="intro.php">Introduction</a></p>
        <p><i class="fa fa-chevron-right" aria-hidden="true"></i> <a style="color:#777; font-size:100%;"
          href="product.php?category=horror">Products</a></p>
        <p><i class="fa fa-chevron-right" aria-hidden="true"></i> <a style="color:#777; font-size:100%;"
          href="price-list.php">Price list</a></p>
        <p><i class="fa fa-chevron-right" aria-hidden="true"></i> <a style="color:#777; font-size:100%;"
          href="contact.php">Contact</a></p>
      </div>
    </div>
  </div>
</div>
```

Figure 37: Code for Footer

In the Footer section, displaying our copyright and all information about me, book store and there are some tags for customer to search our book store by fastest way.



This is SCSS and CSS code for Footer section

```
.footer_area {
  position: relative;
  z-index: 1;
  background-color: #252525;
  padding: 80px 0;
  .single_widget_area {
    .copywrite {
      font-size: 12px;
      color: rgba(255, 255, 255, 0.4);
      margin-top: 10px;
      a {
        font-size: 12px;
        color: rgba(255, 255, 255, 0.4);
        &:hover {
          color: #fff;
        }
      }
    }
  }
  .footer_menu {
    position: relative;
    z-index: 1;
    margin-left: auto;
    .navbar {
      padding: 0;
    }
    .navbar-nav {
      .nav-item {
        .nav-link {
          padding: 10px 30px;
          color: #fff;
          text-transform: uppercase;
          font-size: 14px;
          @media #{$breakpoint-md} {
            padding: 10px 0;
          }
          @media #{$breakpoint-xs} {
            padding: 10px 0;
          }
          &:hover,
```

Figure 38: SCSS code for Footer

## Register page

### SIGN UP

Username

Password

Confirm password

Full name

Phone number

Address

Sign up

Clear

What if you already have an account? [Log in now.](#)

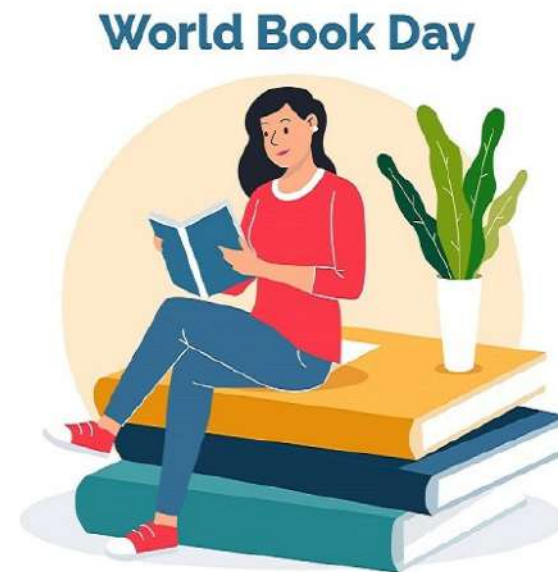


Figure 39: Register page

The register page will include boxes to fill in the account registration information: Username, Password, Confirm password, Full name, Phone number, Address. It is designed in HTML, CSS, PHP.

```
<form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
  <div class="form-group <?php echo (!empty($username_err)) ? 'has-error' : ''; ?>">
    <label>Username</label>
    <input type="text" id="username" name="username" class="form-control"
      value="<?php echo $username; ?>">
    <span class="help-block"><?php echo $username_err; ?></span>
  </div>
  <div class="form-group <?php echo (!empty($password_err)) ? 'has-error' : ''; ?>">
    <label>Password</label>
    <input type="password" id="password" name="password" class="form-control"
      value="<?php echo $password; ?>">
    <span class="help-block"><?php echo $password_err; ?></span>
  </div>
  <div class="form-group <?php echo (!empty($confirm_password_err)) ? 'has-error' : ''; ?>">
    <label>Confirm password</label>
    <input type="password" id="repassword" name="confirm_password" class="form-control"
      value="<?php echo $confirm_password; ?>">
    <span class="help-block"><?php echo $confirm_password_err; ?></span>
  </div>
  <div class="form-group <?php echo (!empty($name_err)) ? 'has-error' : ''; ?>">
    <label>Full name</label>
    <input type="text" id="name" name="name" class="form-control"
      value="<?php echo $name; ?>">
    <span class="help-block"><?php echo $name_err; ?></span>
  </div>
  <div class="form-group <?php echo (!empty($phone_err)) ? 'has-error' : ''; ?>">
    <label>Phone number</label>
    <input type="number" id="phone" name="phone" class="form-control"
      value="<?php echo $phone; ?>">
    <span class="help-block"><?php echo $phone_err; ?></span>
  </div>
  <div class="form-group <?php echo (!empty($address_err)) ? 'has-error' : ''; ?>">
    <label>Address</label>
    <input type="text" id="address" name="address" class="form-control"
      value="<?php echo $address; ?>">
    <span class="help-block"><?php echo $address_err; ?></span>
  </div>
```

Figure 40: Code for Register page

This is CSS code for Register page

```
.cart-table-area {  
  position: relative;  
  z-index: 1;  
  @include flex(0 0 calc(95% - 320px));  
  width: calc(95% - 320px);  
  max-width: calc(95% - 320px);  
  @media #{$breakpoint-lg} {  
    @include flex(0 0 calc(100% - 320px));  
    width: calc(100% - 320px);  
    max-width: calc(100% - 320px);  
  }  
  @media #{$breakpoint-md} {  
    @include flex(0 0 calc(100% - 280px));  
    width: calc(100% - 280px);  
    max-width: calc(100% - 280px);  
  }  
  @media #{$breakpoint-xs} {  
    @include flex(0 0 100%);  
    width: 100%;  
    max-width: 100%;  
    padding-top: 0;  
  }  
}  
.cart-title {  
  h2 {  
    font-size: 30px;  
    margin-bottom: 30px;  
    color: #242424;  
  }  
}  
table {  
  thead {  
    width: 100%;  
    display: block;  
    tr {  
      @include display-flex(flex);  
      width: 100%;  
      height: 40px;  
      background-color: #f5f7fa;  
    }  
  }  
}
```

Figure 41: CSS code for Register page

First, I have connected to the database so that after the customer successfully registers, the information will be saved to the database. Here, I used function `isNotLoggedIn` which means that I want to check the customer logged or not yet, the program will automatically declare the variable names of the columns to fill in the information that will correspond to the columns in the database. If customer has not entered user then it will notify customer to enter user and after successfully entering user it will check if Password, ConfirmPassword and other section are the same.

```
function isNotLoggedIn() {
    if(!isset($_SESSION["loggedin"]) || $_SESSION["loggedin"] !== true){
        return true;
    } else {
        return false;
    }
}

$username = $password = $confirm_password = $name = $phone = $address = "";
$username_err = $password_err = $confirm_password_err = $name_err = $phone_err = $address_err = "";

if($_SERVER["REQUEST_METHOD"] == "POST"){

    if(empty(trim($_POST["username"]))){
        $username_err = "You have not entered username yet!";
    } else{

        $sql = "SELECT id FROM users WHERE username = ?";

        if($stmt = $mysqli->prepare($sql)){

            $stmt->bind_param("s", $param_username);

            $param_username = trim($_POST["username"]);

            if($stmt->execute()){

                $stmt->store_result();

                if($stmt->num_rows == 1){
                    $username_err = "This username is no longer available!";
                } else{
                    $username = trim($_POST["username"]);
                }
            }
        }
    }
}
```

```
if(empty(trim($_POST["password"]))) {
    $password_err = "You have not entered password!";
} elseif(strlen(trim($_POST["password"])) < 5) {
    $password_err = "The password you entered is too short!";
} else {
    $password = trim($_POST["password"]);
}

if(empty(trim($_POST["confirm_password"]))) {
    $confirm_password_err = "You have not confirmed your password!";
} else {
    $confirm_password = trim($_POST["confirm_password"]);
    if(empty($password_err) && ($password != $confirm_password)) {
        $confirm_password_err = "Confirm password does not match!";
    }
}

if(empty(trim($_POST["name"]))) {
    $name_err = "You did not enter a first and last name!";
} elseif(strlen(trim($_POST["name"])) > 25) {
    $name_err = "The name you just entered is too long!";
} else {
    $name = trim($_POST["name"]);
}

if(empty(trim($_POST["phone"]))) {
    $phone_err = "You did not enter a phone number!";
} elseif(strlen(trim($_POST["phone"])) > 11) {
    $phone_err = "Incorrect phone number!";
} elseif(strlen(trim($_POST["phone"])) < 10) {
    $phone_err = "Incorrect phone number!";
} else {
    $phone = trim($_POST["phone"]);
}

if(empty(trim($_POST["address"]))) {
    $address_err = "You do not enter an address!";
} elseif(strlen(trim($_POST["address"])) > 1000) {
```

Figure 42: PHP code for Register page

## Log in page

In order to log in our book store, customer must register an account by Register page, then that account will be saved in database, which allow customer logging in. Customer will enter username and password according to their input in the Register page.

### LOG IN

Username

Password

Log in

What if you don't have an account? [Register now.](#)

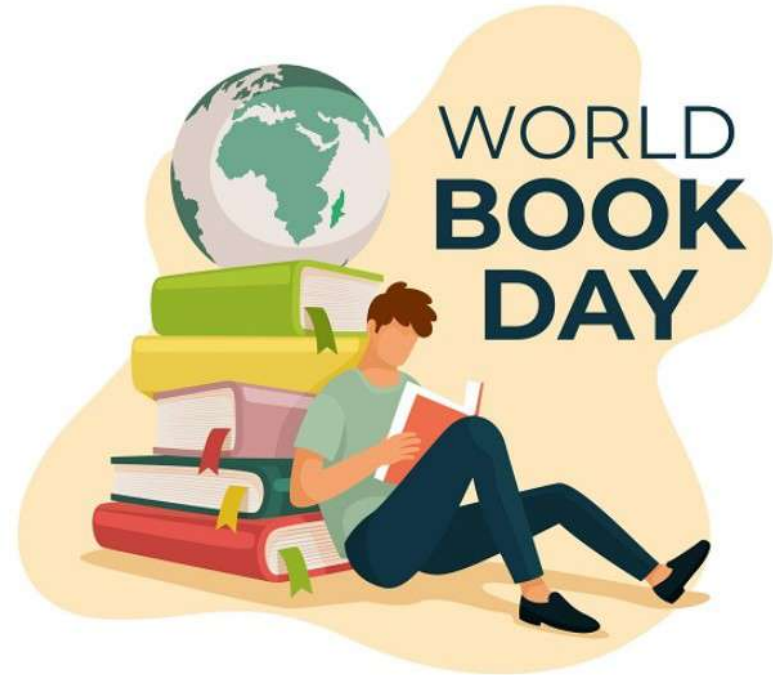


Figure 43: Log in page

```
<div class="cart-table-area section-padding-100">
  <div class="container pt-5 justify-content-center">
    <div class="row">
      <div class="col-lg-6 col-md-12 col-xs-12">
        <div class="cart-title mt-50">
          <h2>LOG IN</h2>
        </div>
        <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
          <div class="form-group <?php echo (!empty($username_err)) ? 'has-error' : ''; ?>">
            <i class="fa fa-user" aria-hidden="true"> </i><label style="margin-left: 5px;">Username</label>
            <input type="text" id="username" name="username" size=30 class="form-control"
              value="<?php echo $username; ?>"
            <span class="help-block"><?php echo $username_err; ?></span>
          </div>
          <div class="form-group <?php echo (!empty($password_err)) ? 'has-error' : ''; ?>">
            <i class="fa fa-lock" aria-hidden="true"> </i><label style="margin-left: 5px;">Password</label>
            <input type="password" id="password" name="password" size=30 class="form-control">
            <span class="help-block"><?php echo $password_err; ?></span>
          </div>
          <div class="form-group">
            <input type="submit" class="btn btn-warning" value="Log in"
              onclick="javascript:check()">
          </div>
          <p>What if you don't have an account? <a href="register.php" style="color: #fbb710; font-weight: bold;">Register now</a>
        </form>
      </div>
      <div class="col-lg-6 col-md-12 col-xs-12">
        
      </div>
    </div>
  </div>
</div>
```

Figure 44: Code for Log in page

I used htmlspecialchars to replace characters with special meaning in HTML which is safe to display it without modification. Besides that, there are serious risks of cross-site scripting, I have a good reason that I need to use action="<?php echo htmlspecialchars(\$\_SERVER["PHP\_SELF"]); ?>". Function !empty(\$username\_err) will check username that entering by customer. It is similar to !empty(\$password\_err) and so on.



This is CSS code for Log in page

```

table {
  head {
    width: 100%;
    display: block;
    tr {
      @include display-flex(flex);
      width: 100%;
      height: 40px;
      background-color: #f5f7fa;
      th {
        border-top: none;
        border-bottom: none;
        @include flex(0 0 25%);
        width: 25%;
        max-width: 25%;
        font-size: 12px;
        color: #242424;
        text-transform: capitalize;
        font-weight: 400;
      }
    }
  }
  tbody {
    width: 100%;
    display: block;
    tr {
      @include display-flex(flex);
      @include align-items(center);
      width: 100%;
      td {
        border: none;
        @include flex(0 0 25%);
        width: 25%;
        max-width: 25%;
        h5 {
          font-size: 16px;
          color: #252525;
        }
      }
    }
  }
}
  
```

Figure 45: CSS code for Log in page

```
function isNotLoggedIn() {
    if(!isset($_SESSION["loggedin"]) || $_SESSION["loggedin"] !== true){
        return true;
    } else {
        return false;
    }
}

if(isset($_SESSION["loggedin"]) && $_SESSION["loggedin"] === true){
    header("location: index.php");
    exit;
}

require_once "config.php";

$username = $password = $user_level = $name = $phone = $address = "";
$username_err = $password_err = "";

if($_SERVER["REQUEST_METHOD"] == "POST"){

    if(empty(trim($_POST["username"]))){
        $username_err = "You have not signed up yet!";
    } else{
        $username = trim($_POST["username"]);
    }

    if(empty(trim($_POST["password"]))){
        $password_err = "You have not entered a password!";
    } else{
        $password = trim($_POST["password"]);
    }
}
```

*Figure 46: PHP code for Log in page*

require\_once "config.php": connect to MySQL

Isset REQUEST\_METHOD: Notify to customer if error may happen on the page that customer is accessing.

## Profile page

I created a Profile page, which shows information of customer and also allows customer editing their profile.

### UPDATE INFORMATION

Full name

Lê Minh Nghĩa

Phone number

0824063391

Address

Huế

Update

Clear

What if you want to change your password? [Click here!](#)

## World Book Day



Figure 47: Profile page

```
<div class="cart-title mt-50">
    <h2>UPDATE INFORMATION</h2>
</div>
<div><?php if(isset($message)) { echo $message; } ?></div>
<form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
    <div class="form-group">
        <label>Full name</label>
        <input type="text" id="name" name="name" class="form-control"
            value="<?php echo $row["name"] ?>">
    </div>
    <div class="form-group">
        <label>Phone number</label>
        <input type="number" id="phone" name="phone" class="form-control"
            value="<?php echo $row["phone"] ?>">
    </div>
    <div class="form-group">
        <label>Address</label>
        <input type="text" id="address" name="address" class="form-control"
            value="<?php echo $row["address"] ?>">
    </div>
    <div class="form-group">
        <input type="submit" name="update" class="btn btn-warning" value="Update"
            onclick="javascript:check()">
        <input type="reset" class="btn btn-default" value="Clear">
    </div>
    <p>What if you want to change your password? <a href="changepw.php" style="color: #fbb710; font-weight: bold;">
</form>
</div>
<div class="col-lg-5 col-md-12 col-xs-12">
    <div class="row h-25">
    </div>
    <div class="row">
        
    </div>
</div>
```

Figure 48: Code for Profile page

```
table {
  width: 100%;
  border-collapse: collapse;
  text-align: center;
  font-family: sans-serif;
  font-size: 12px;
  color: #242424;
  font-weight: 400;

  tr {
    height: 40px;
    background-color: #f5f7fa;
  }

  th {
    width: 25%;
    max-width: 25%;
    font-size: 12px;
    color: #242424;
    text-transform: capitalize;
    font-weight: 400;
    border-top: none;
    border-bottom: none;
    @include flex(0 0 25%);
  }

  tbody {
    width: 100%;
    display: block;
    tr {
      @include display-flex(flex);
      @include align-items(center);
      width: 100%;
      td {
        border: none;
        @include flex(0 0 25%);
        width: 25%;
        max-width: 25%;
        h5 {
          font-size: 16px;
          color: #252525;
        }
      }
    }
  }
}
```

This is SCSS code for Profile page

```
session_start();

include 'config.php';

include 'function.php';
$id = $_SESSION["id"];
$qry = mysqli_query($con,"select * from users where id='$id'"); // select query
$row = mysqli_fetch_array($qry);

if(isset($_POST['update'])){
    if(count($_POST)>0) {
        if($_POST['name'] != '' && $_POST['phone'] != '' && $_POST['address'] != '') {
            $name = $_POST['name'];
            $phone = $_POST['phone'];
            $address = $_POST['address'];
            mysqli_query($con,"UPDATE users set name = '$name', phone = '$phone', address = '$address' WHERE id= '" . $id. "'");
            $row['name'] = $name;
            $row['phone'] = $phone;
            $row['address'] = $address;
            $message = "Successfully updated";
        }
        elseif($_POST['name'] == ''){
            $message = "You have not entered a new name yet!";
        }
        elseif($_POST['phone'] == ''){
            $message = "You have not entered a new phone number!";
        }
        elseif($_POST['address'] == ''){
            $message = "You have not entered a new address!";
        }
    }
    mysqli_close($con);
}
```


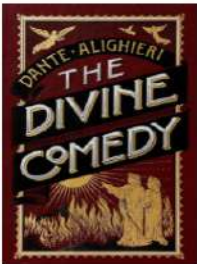
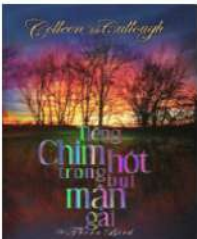
Figure 50: PHP code for Profile Page

Firstly I will connect to file function.php, each customer will have hide ID in MySQL, when customer wants to update information, MySQL will select all information from users table by ID. If customer has not entered information yet, system will send notify text.

## Cart page

Cart page will include a table that will contain information of the product after ordering from outside Product page: Image, Name, Price, Quantity, Remove. It is designed in HTML, CSS, SCSS and PHP.

### Cart

Name	Price	Quantity
 <div>Phantom Thread</div> <div>Remove</div>	200000 VND	1
 <div>The Divine Comedy</div> <div>Remove</div>	110000 VND	1
 <div>Tiếng chim hót trong bụi mận gai</div> <div>Remove</div>	36000 VND	1

Cart subtotal

Total Price: 512000 VND

Delivery: Free

Total: 512000 VND

Payment

Figure 51: Cart page

```
<div class="cart-table clearfix">
    <?php
    $total_price = 0;
    if(isset($_SESSION["shopping_cart"])){
    ?>
    <table class="table table-responsive">
        <thead>
            <tr>
                <th></th>
                <th>Name</th>
                <th>Price</th>
                <th>Quantity</th>
            </tr>
        </thead>
        <tbody>
            <?php
            foreach ($_SESSION["shopping_cart"] as $product){
            ?>
            <tr>
                <td class="cart_product_img">
                    <img src='<?php echo $product["image"]; ?>' alt="Product">
                </td>
                <td class="cart_product_desc">
                    <h5><?php echo $product["name"]; ?></h5>
                    <form method='post' action=''>
                        <input type='hidden' name='code' value='<?php echo $product["code"]; ?>' />
                        <input type='hidden' name='action' value="remove" />
                        <button type='submit' class='btn btn-danger remove'>Remove</button>
                    </form>
                </td>
            </tr>
        </tbody>
    </table>
    }
    ?>
    </div>
```

Figure 52: Code for Cart page

Foreach loop is used in this page to display all books in cart with its information. I created Remove button to delete book.



```
.cart-table-area {
  position: relative;
  z-index: 1;
  @include flex(0 0 calc(95% - 320px));
  width: calc(95% - 320px);
  max-width: calc(95% - 320px);
  @media #{ $breakpoint-lg } {
    @include flex(0 0 calc(100% - 320px));
    width: calc(100% - 320px);
    max-width: calc(100% - 320px);
  }
  @media #{ $breakpoint-md } {
    @include flex(0 0 calc(100% - 280px));
    width: calc(100% - 280px);
    max-width: calc(100% - 280px);
  }
  @media #{ $breakpoint-xs } {
    @include flex(0 0 100%);
    width: 100%;
    max-width: 100%;
    padding-top: 0;
  }
}

.cart-title {
  h2 {
    font-size: 30px;
    margin-bottom: 30px;
    color: #242424;
  }
}

table {
  thead {
    width: 100%;
    display: block;
    tr {
      @include display-flex(flex);
      width: 100%;
      height: 40px;
      background-color: #f5f7fa;
    }
  }
}
```

Figure 53: SCSS code for Cart page

This is SCSS code for Cart page.

```
if (isset($_POST['code']) && $_POST['code']!=""){
    $code = $_POST['code'];
    $result = mysqli_query(
        $con,
        "SELECT * FROM `products` WHERE `code`='$_POST['code']'"
    );
    $row = mysqli_fetch_assoc($result);
    $name = $row['name'];
    $code = $row['code'];
    $price = $row['price'];
    $image = $row['image'];

    $cartArray = array(
        $code=>array(
            'name'=>$name,
            'code'=>$code,
            'price'=>$price,
            'quantity'=>1,
            'image'=>$image)
    );

    if(empty($_SESSION["shopping_cart"])) {
        $_SESSION["shopping_cart"] = $cartArray;
        $status = "<div class='box'>Product is added to your cart!</div>";
    }else{
        $array_keys = array_keys($_SESSION["shopping_cart"]);
        if(in_array($code,$array_keys)) {
            $status = "<div class='box' style='color:red;'>
            Product is already added to your cart!</div>";
        } else {
            $_SESSION["shopping_cart"] = array_merge(
                $_SESSION["shopping_cart"],
                $cartArray
            );
            $status = "<div class='box'>Product is added to your cart!</div>";
        }
    }
}
```

Figure 54: PHP code for Cart page

When the customer presses the Cart logo, the program will automatically understand `isset($_POST['code'])`, and then get the product information corresponding to the code of product.

```
<?php
    session_start();

    include 'config.php';

    include 'function.php';

    $status="";
    if (isset($_POST['action']) && $_POST['action']=="remove"){
        if(!empty($_SESSION["shopping_cart"])) {
            foreach($_SESSION["shopping_cart"] as $key => $value) {
                if($_POST["code"] == $key){
                    unset($_SESSION["shopping_cart"][$key]);
                    $status = "<div class='box' style='color:red;'>
                        1 product has just been removed from the cart!</div>";
                }
            }
            if(empty($_SESSION["shopping_cart"])){
                unset($_SESSION["shopping_cart"]);
            }
        }
    }

    if (isset($_POST['action']) && $_POST['action']=="change"){
        foreach($_SESSION["shopping_cart"] as &$value){
            if($value['code'] === $_POST["code"]){
                $value['quantity'] = $_POST["quantity"];
                break; // Stop the loop after we've found the product
            }
        }
    }
}
```

*Figure 55: PHP code for Cart page*

To be able to remove or change quantity of product in Cart, I must also connect to the database to get product information. If (!empty(\$\_SESSION["shopping\_cart"])) then the program will run a foreach loop to get the data of product corresponding to the product that the customer wants to remove or change.

## 2. Create/ Read/ Update/ Delete Function

### 1) Create

#### a. Create account

The first step would be register a new user account by clicking Sign up and then filling in the user information in the boxes provided

### SIGN UP

Username

nghiatest

Password

.....

Confirm password

.....

Full name

Lee Min Nghia

Phone number

0824063333

Address

Da Nang, Viet Nam

Sign up

Clear

What if you already have an account? [Log in now.](#)

*Figure 56: Create Account*

If the account has successfully logged in, the account will transfer to Log in page. If the Password and Confirm Password are not same, the program will ask to re-enter it. Then we will go to MySQL database to check the newly create account information. The newly created account has an ID of 11 and the information just entered is displayed correctly in the database

id	username	password	created_at	name	phone	address
10	nghialmgcd201527	\$2y\$10\$ubmpTAOKFQvTOtpVenTOG.YnTDZUJh/OnYSSCuUkW39...	2021-12-17 16:07:01	Lê Minh Nghĩa	0824063391	Huế
11	nghiatest	\$2y\$10\$PcWnvITTXCmsnmunYESJV.1e5y9UI4Tm1MagpCS762H...	2021-12-24 09:37:40	Lee Min Nghia	0824063333	Da Nang, Viet Nam

Figure 57: Check account

## b. Create order

After pressing Cart logo, product will be added to cart



Figure 58: Pressing Cart logo


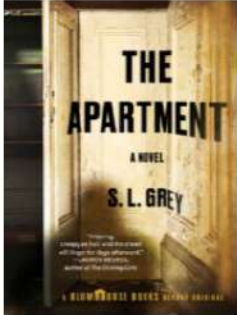

	Name	Price	Quantity
	Phantom Thread <a href="#">Remove</a>	200000 VND	1 ▾
	The Apartment <a href="#">Remove</a>	120000 VND	1 ▾
	Scary Stories <a href="#">Remove</a>	120000 VND	1 ▾

Figure 59: Cart

## 2) Delete

### Deleting order

I have added to my cart some books and now I will remove those products from the cart


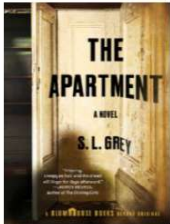

	Name	Price	Quantity
	Phantom Thread <a href="#">Remove</a>	200000 VND	1 ▾
	The Apartment <a href="#">Remove</a>	120000 VND	1 ▾
	Scary Stories <a href="#">Remove</a>	120000 VND	1 ▾

Figure 60: Cart before removing the book “Scary Stories”

Below is the result after removing the book “Scary Stories”


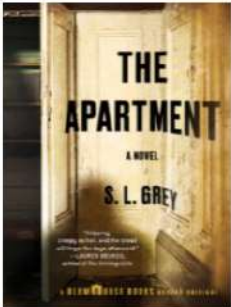



	Name	Price	Quantity
	Phantom Thread <a href="#">Remove</a>	200000 VND	1 ▾
	The Apartment <a href="#">Remove</a>	120000 VND	1 ▾

Figure 61: Cart after removing the book “Scary Stories”



### 3) Update

I will show update of total price. The total price of order is shown below.

Name	Price	Quantity
 <div>Phantom Thread</div> <div>Remove</div>	200000 VND	1
 <div>The Apartment</div> <div>Remove</div>	120000 VND	1
 <div>Scary Stories</div> <div>Remove</div>	120000 VND	1

Cart subtotal


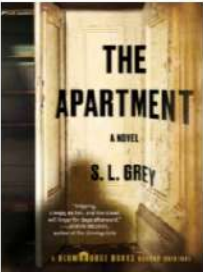
Total Price: 440000 VND

Delivery: Free

Total: 440000 VND

Payment

Figure 62: Total price before removing the book “Scary Stories”

Name	Price	Quantity
 <div>Phantom Thread</div> <div>Remove</div>	200000 VND	1
 <div>The Apartment</div> <div>Remove</div>	120000 VND	1

Cart subtotal

Total Price: 320000 VND

Delivery: Free

Total: 320000 VND

Payment

Figure 63: Total price after removing the book “Scary Stories”

#### 4) Read

Continuing we will go to the Profile page to see the information just registered. The Profile page displayed the correct information just registered.

## UPDATE INFORMATION

Full name

Lê Minh Nghĩa

Phone number

0824063391

Address

Huế

Update

Clear

*Figure 64: Read Profile*

III. Comparison and contrast the multipage website created to the design document (M4)

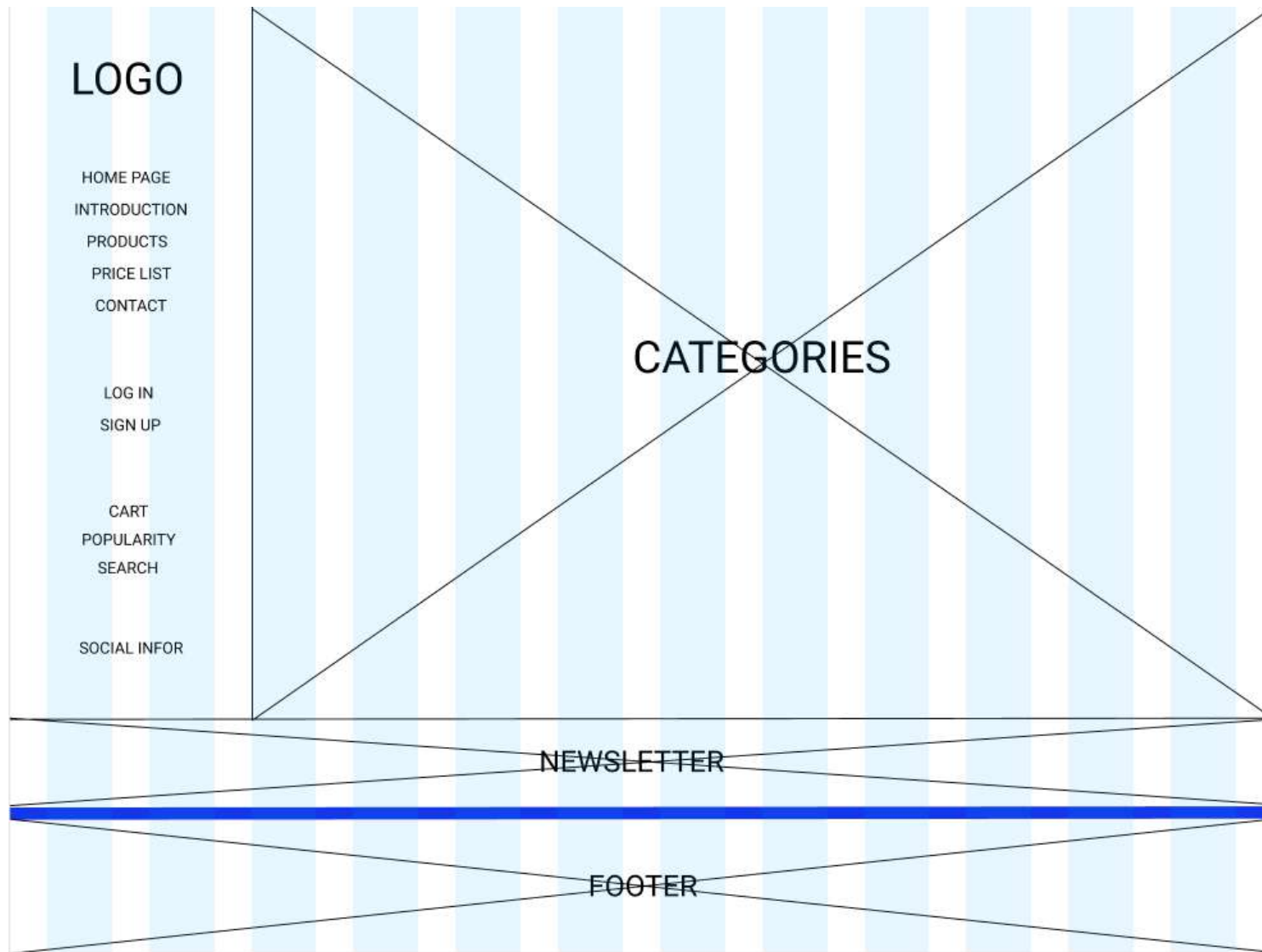


Figure 65: Wireframe for web

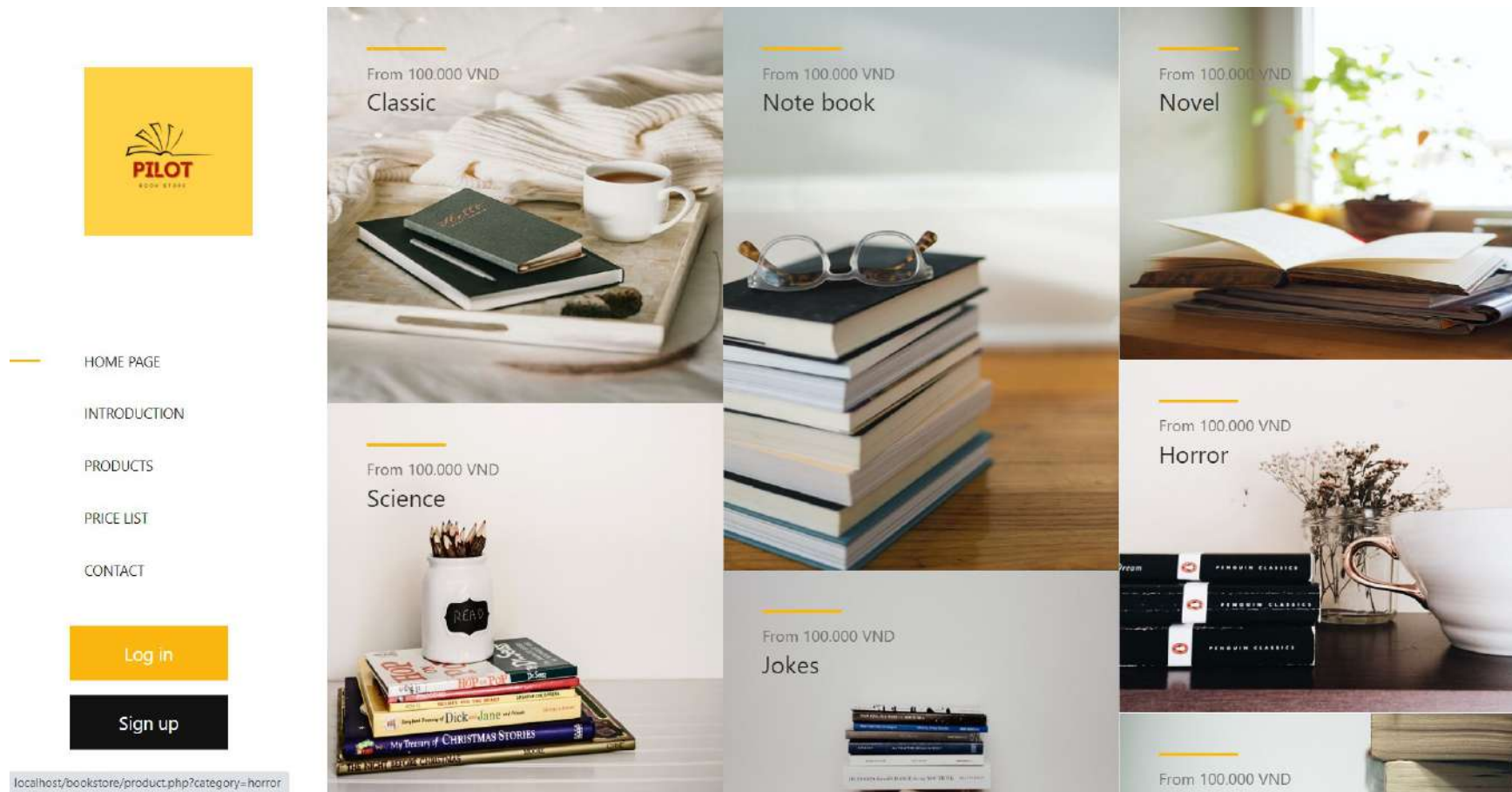


Figure 66: Landing page

[illegible]

Figure 67: Wireframe for Cart page



Name	Price	Quantity	Cart subtotal	
	Lược Sử Thời Gian	152000 VND	1	Total Price: 227000 VND
<a href="#">Remove</a>				Delivery: Free
				Total: 227000 VND
	Tế Bào Gốc: Khám Phá Cùng Nhà Khoa Học	75000 VND	1	<a href="#">Payment</a>
<a href="#">Remove</a>				

Figure 68: Cart page

## Evaluation

- Drawing wireframes is one of the most important tasks in the web design process. However, if we do not draw them before to web construction, unpleasant things will happen. Let's imagine I designed a website that fits the customer's criteria based on the client's request, but the customer doesn't like the way I layout the navigation bar; the logo should be on the left, and the functionality should be on the left. Menus, shopping carts, and login and registration functionalities, among other things, should all be on the right. And it took a lot of time and work on my part. So, if you don't sketch wireframes before designing, you'll have a major problem; this should be enough to convince you of the need of wireframes.
- As a first stage, I simply lay out the client's needs and the functionality that they desire. Only in this easy approach am I able to easily alter the location, add or remove a few components, and suit the customer. This enables me to understand and evaluate the situation more thoroughly before to creating and constructing the website.
- The color element is the most noticeable distinction between wireframe and web. You may see what you consider to be vital and required in the wireframe. However, not only is the style of a website crucial, but the colors are just as significant.
- The home page, registration, login, profile, and cart pages are mainly clearly built according to the original wireframe when compared to the original design. As a result, I am correct in claiming that my site is 90% complete in accordance with the wireframe.
- However, there are a few aspects that I need to address during the design and development phase to ensure that the website is consistent. The distance between the Logo and the function buttons, for example, is fairly close in the design of the wireframe's navigation bar. However, if we do it like a wireframe on the main page, it will make the homepage appear awful since their distance is pretty close together and loses the balance compared to alignment, therefore we need to do something to make it more balanced. As a result, the remaining 10% will be used to change the navigation bar rhyme.
- I create the majority of the elements in the wireframe on the website. So there's nothing I can't do on the website in terms of wireframes.



	Wireframe	Prototype
What	A sketch/ basic rendering to convey the high-level concept of new product functionality	Interactive simulation of new product functionality
Purpose	To gain consensus and collect internal feedback on how new functionality will work	To collect feedback by user testing the real experience
Design fidelity	Low	High
Included elements	The format and structure of content	Final interactive elements and navigation
Time invested	Low	High
Creator	PM or UX Designer	UX Designer

#### IV. Critically evaluation the design and development process against your design document and analyze any technical challenges (D2)

##### 1. Evaluation the design document

Design documentation is critical to our business because it explains the design, the reasoning for our decisions, and the tools for customers to carry on until the project is completed.

Not only does good interface documentation identify all of the pixel dimensions, text kinds, and interaction nuances, but it also explains how to use them (which it must). Internal stakeholders can also tell the high-level story, with the broad picture sewn together and the future.

As outside contractors, one of our top goals is to build confidence and traction around the design. Both managers and developers will be able to hear from the paperwork. After all, when it comes to executing design, we usually leave a lot of the hard work at the customer's door, and developing amazing apps begins with bringing all of the stakeholders on board. The finest product documentation provides a single design language, a design framework, and a venue for the customer to improve the design.

For designs of Online Book Store and above, design documentation is required. It provides a way to communicate a Website's design to developers, allowing them to grasp the Web's design principles and update or upgrade the system without jeopardizing the Web's stability. The data that is contained within the design.

The document provides a reason for why a Web is designed so and whether changing the Web will change the intent of the design.

Document design is critical to the Web maintenance lifecycle and should not be viewed as a burden on the development system. This paper will assist developers in comprehending the significance of design documents throughout the life of a computer Web site, as well as ensuring site stability. The necessity of satisfying the objective and purpose of ShopOnline design should be recognized by developers. Complying with document requirements is the polar opposite of rigidly adhering to them.

## **2. Evaluation the process of development**

### **a. Aesthetics**

All of the hues should work nicely together. When supporting people with color blindness or visual disabilities, it's important to use colors that are bright enough. Make sure the typefaces you use on your website are simple to read regardless of the user's screen resolution.

### **b. Content and accessibility**

The copywriting style should represent the website's purpose and communicate to the targeted target audience. We do this through our tone, design, and arrangement choices. When we choose, we must ensure that the content throughout the whole platform is consistent.

### **c. Usability**

One of the most important aspects of accessibility is the ability to be customized for multiple browser kinds and online devices. And, in order to maintain that accessibility, you must keep in mind how quickly the website loads. This is critical since travelers will exit your website if they become impatient while waiting.

The search feature of a website is a crucial component of any website since it allows visitors to find the content they are looking for more quickly. This would improve their entire experience and encourage them to return to the web.

<b>Robustness</b>	Responsiveness, When the Web is moved to a customer or into a system, it refers to the system's capacity to suffer disruptions that might impact the system's functioning or performance. Our Web Shop performs well in terms of billing, order processing, and shopping cart operations.
<b>Reliability</b>	The design, construction, and usage of a product or system determine its actual reliability performance. The inherent reliability capability performance is mostly determined by design decisions.
<b>Portability</b>	The website is simple to set up and run on any host that is compatible (s) Multiple dependencies might exist in complex web applications. It's probable that the code contains settings or customised routines that will fail if the program is relocated to a new computer if it's been on the same server for a long time. You'll need scripts to create the server, install the necessary packages, and install the code. It's straightforward to load your website onto any identical server once that's all set up.
<b>Efficiency</b>	The website satisfies the demands, the performance should improve, the technical process should be met, and efficiency should be ensured through test cases.
<b>Maintainability</b>	During the challenging maintenance phase, you must labor continually to maintain the website, its content, product pricing, and promotions.

### 3. The technical challenges/ difficulties in web development

#### User interface and user experience (UX/ UI)

The web used to be a whole different realm, depending on the size of the screen that the client was using. Websites, for example, should be responsive enough on smaller displays in the age of Smartphones. It's sometimes the smallest UI components that have the most impact. It's tough to keep my customers loyal to your website if my web application annoys or confuses them. Another aspect of a website that developers frequently overlook is navigation. The website visitor will have a better user experience if the navigation is intuitive. Without a learning curve, intuitive navigation leads my audience to the information they want. Customers can discover information without difficulty when the navigation is easy, resulting in a faultless experience that prevents them from visiting rivals.

## Scalability

Scalability has nothing to do with performance or making efficient use of computer power and bandwidth. It's all about load balancing across the servers, so when the demand grows (i.e. there's more traffic on the page), more servers may be added to help balance things out. I should not put all of the demand on a single server; instead, I should build the software to run in a cluster of servers. When more servers are added, service-oriented architecture (SOA) can aid with scalability. SOA allows me to adapt quickly and simply. Service-oriented architecture is a design in which application components deliver services to other components through a network using a communication protocol.

## Performance

Website speed is widely acknowledged as being critical to the success of a website. When it comes to my internet company, every second matters. Web apps that are too slow are a failure. As a result, customers leave my website, harming both your revenue and reputation. It is claimed that before designing a web application, think about performance. Poorly written code, unoptimized databases, unmanaged data growth, traffic surges, poor load distribution, default setup, troublesome third-party services, and so on are some of the performance difficulties. A content distribution network (CDN) is a worldwide network of proxy servers scattered across several data centers. It implies that instead of employing a single web server, a network of servers is used. The requests on the server will be routed to multiple servers to balance the traffic, and the contents will be separated on different CDNs so that there will be no queuing and waiting for different assets such as photos, videos, text, and so on.

## Front-end

Working with the web for two months has convinced me that web development has never been more exciting. Here are some of the obstacles that a front-end web developer will confront, in my opinion: Strongly encode JavaScript. With the help of mature MVC and CMS, the program was finally created and generated solid code that can be maintained pretty simple. I need to put in more effort to guarantee that all JavaScript works in all browsers and that new code is developed and maintained in a logical manner. Maintain a healthy balance between long-term and short-term design goals. If you make a modification that is too brief, the code will be broken. Your team will become too inflexible if you wait too long. I have to design the proper keywords, arrange the links, and create a site map in order to implement SEO principles. Register for a search engine sitemap. Create admin pages to make managing the Book Store's staff a breeze.

## Back-end

The backend developer's duty is to figure out what the website's goals are and how to achieve them. Keeping data safe while simultaneously ensuring that it is visible to the users who should have access to it. Responsible for the development of a payment processing system that includes taking data, securely storing it, and charging that payment. Manage API resources that are cross-platform. He or she might be involved in system architecture and data science analyses. Developers are in charge of organizing the logic of a system that works on several devices. To make it simpler to program against, backend developers should be engaged in the creation of frameworks or architecture. Back-end developers should be capable of creating algorithms and troubleshooting system issues.

### **V. Create a suitable Test Plan identifying key performance area and use it to review the functionality and performance of your website. User Experience (UX) and User Interface (UI) (P7)**

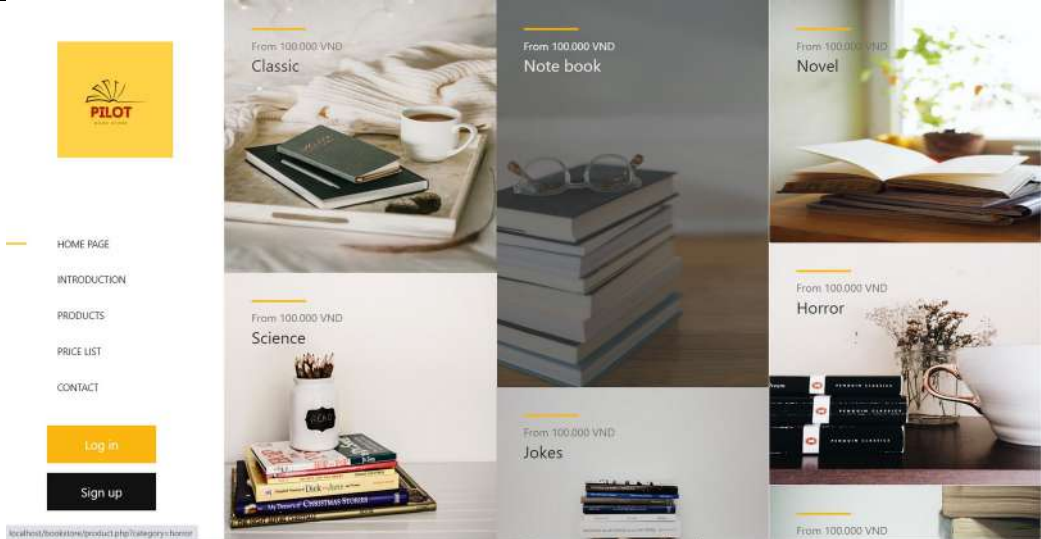
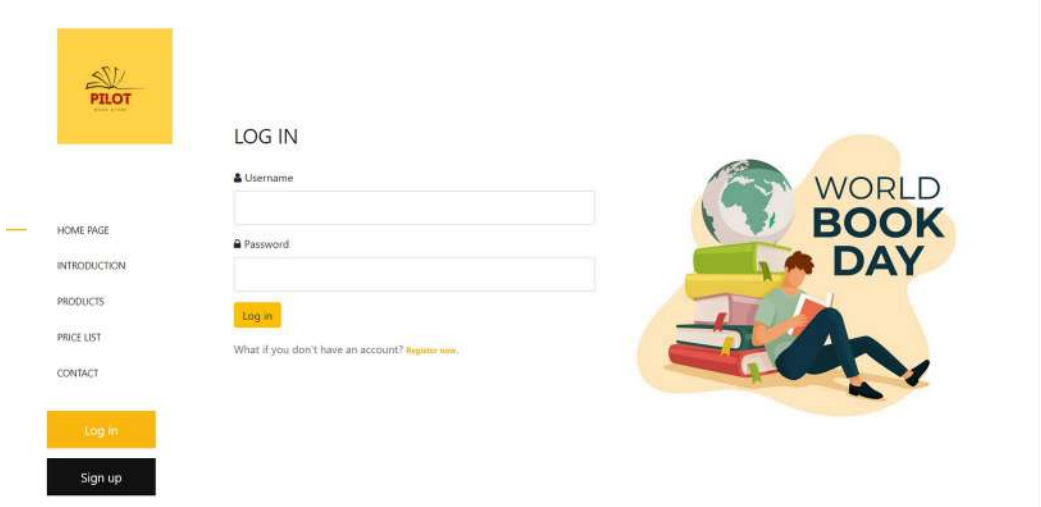
#### **1. Test plan**

- As a customer, I want the website to display all pages: Landing page, Login page, Register page, Profile page, Cart page.
- As a customer, I want the website to be successfully register as a new user.
- As a customer, I want the website to be successfully logged in
- As a customer, I want the website to have to see the products so I can choose to buy it.
- As a customer, I want the website to allow me to update my profile information so that I can change it as needed.
- As a customer, I want the website to allow me to place an order so I can buy it.
- As a customer, I want the website to allow me to remove items from my shopping cart when I no longer need them.

## 2. Test schedule

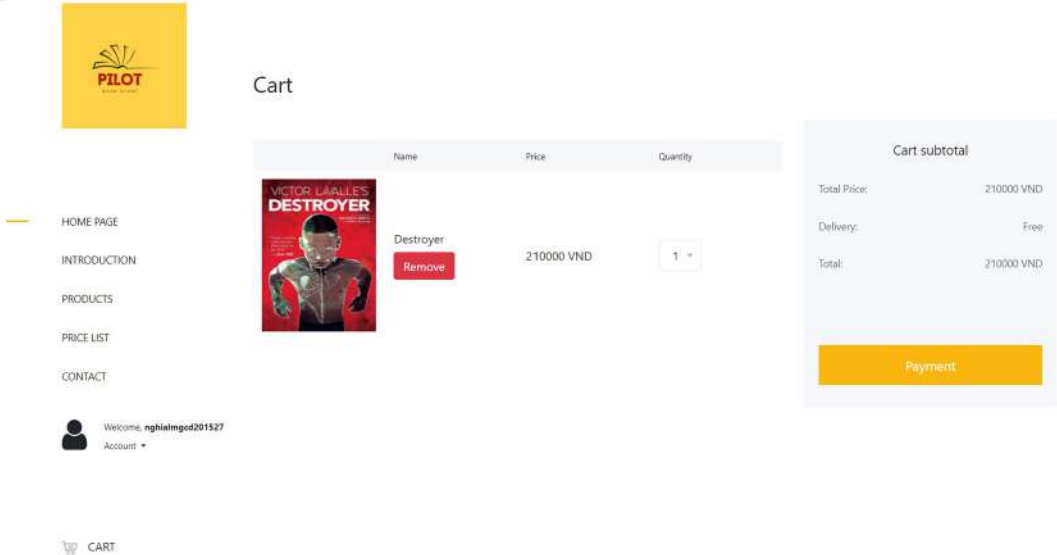
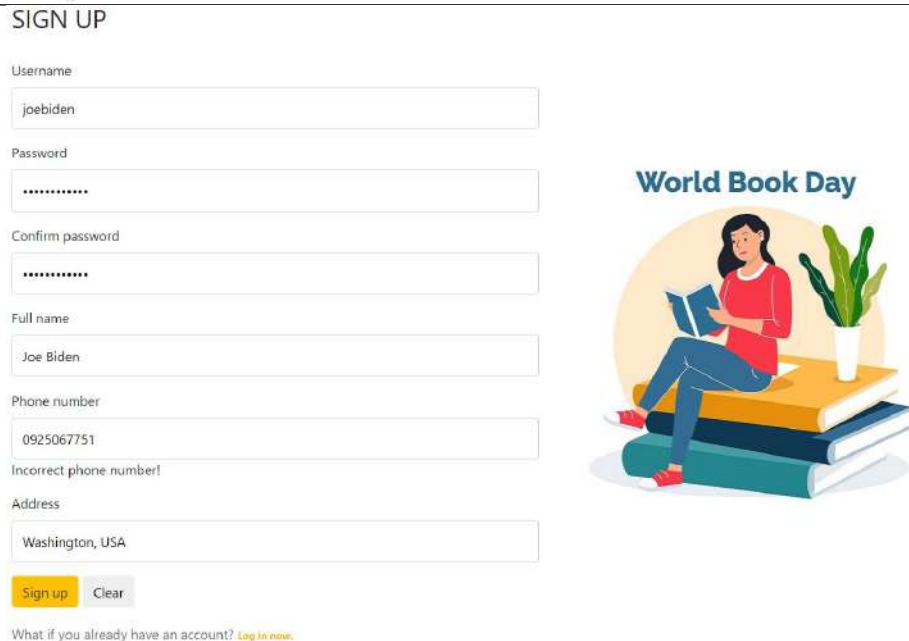
Test case	Start date	End date	Estimated effort (Hrs/Day)	Notes
1	24/12/2021	24/12/2021	1	
2	24/12/2021	24/12/2021	2	
3	24/12/2021	24/12/2021	1	
4	24/12/2021	24/12/2021	1	
5	24/12/2021	24/12/2021	1	
6	24/12/2021	24/12/2021	2	
7	24/12/2021	24/12/2021	1	
8	24/12/2021	24/12/2021	1	
9	24/12/2021	24/12/2021	1	
10	24/12/2021	24/12/2021	1	
11	24/12/2021	24/12/2021	2	
12	04/01/2022	04/01/2022	1	
13	04/01/2022	04/01/2022	1	
14	04/01/2022	04/01/2022	1	
15	04/01/2022	04/01/2022	1	
16	04/01/2022	04/01/2022	1	

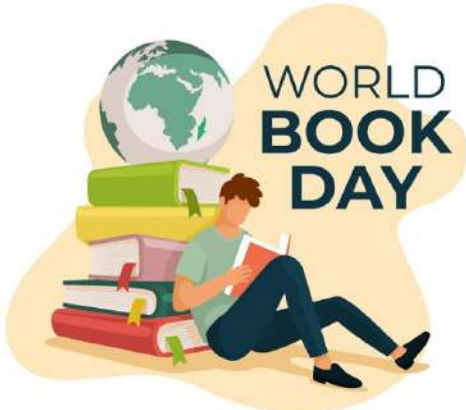

### 3. Test case

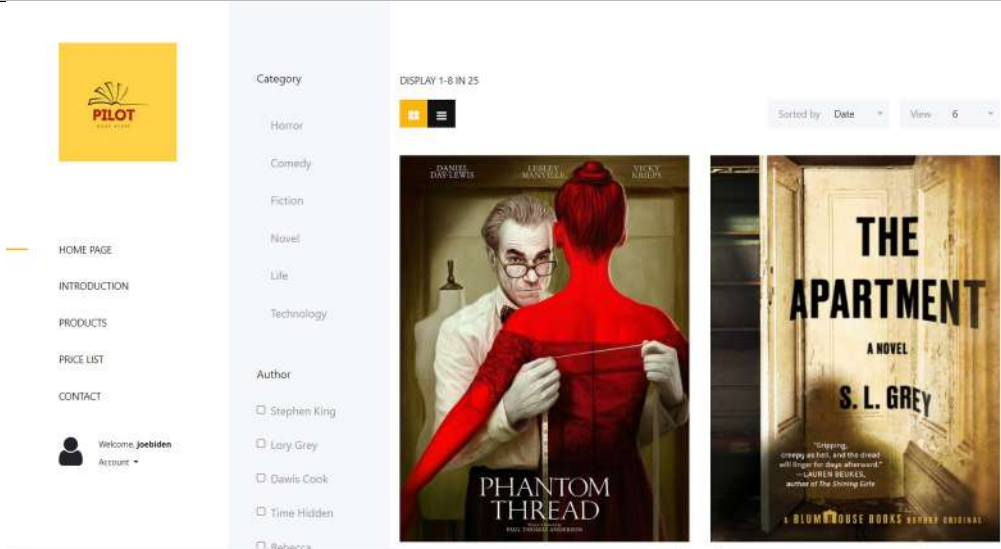
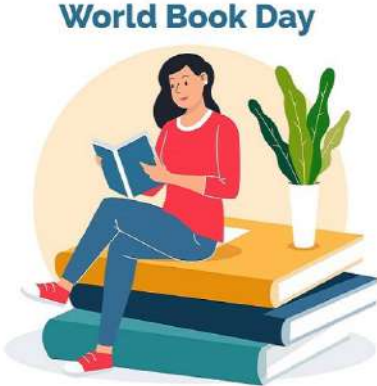
NO	TEST NAME	INPUT	EXPECT RESULT	ACTUAL RESULTS	ANALYZE TEST	NOTE
1	Test the interface of landing page	Web/ index.php	Show the interface of landing page		Correct test result	Pass
2	Test the interface of login page	Web/ login.php	Show the interface of login page		Correct test result	Pass

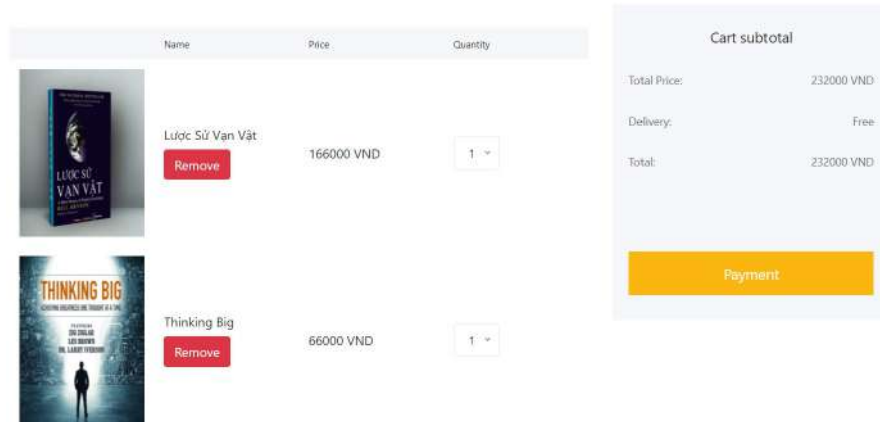
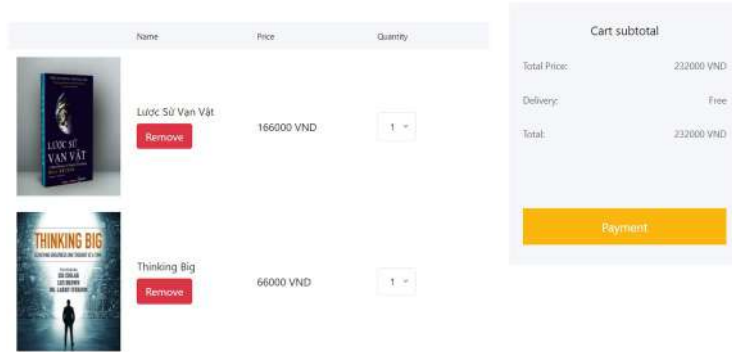
3	Test the interface of register page	Web/register.php	Show the interface of register page		Correct test result	Pass
4	Test the interface of profile page	Web/edit-user.php	Show the interface of profile page		Correct test result	Pass

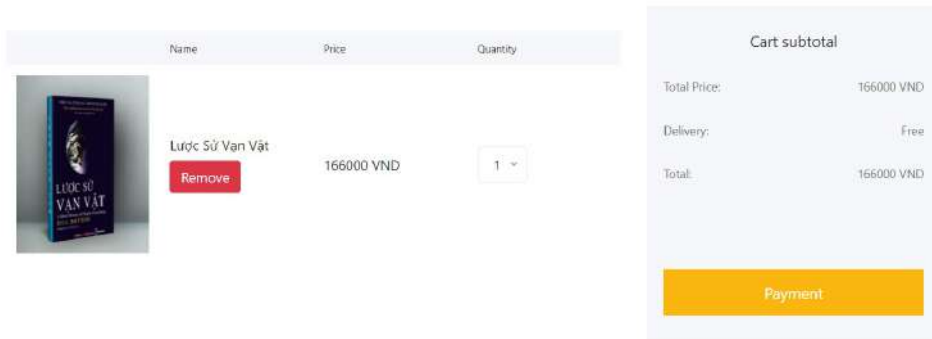
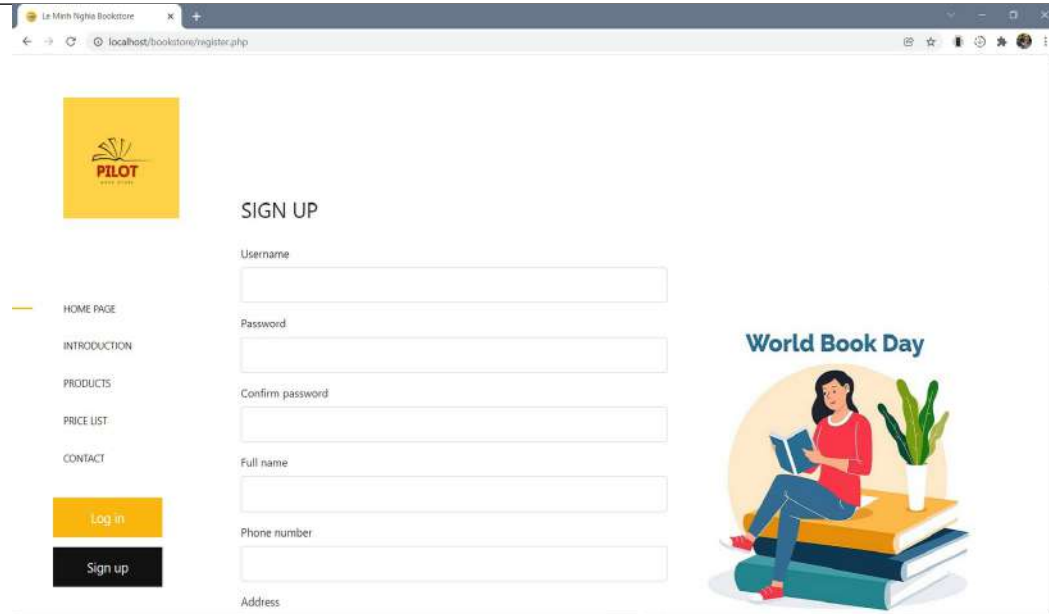


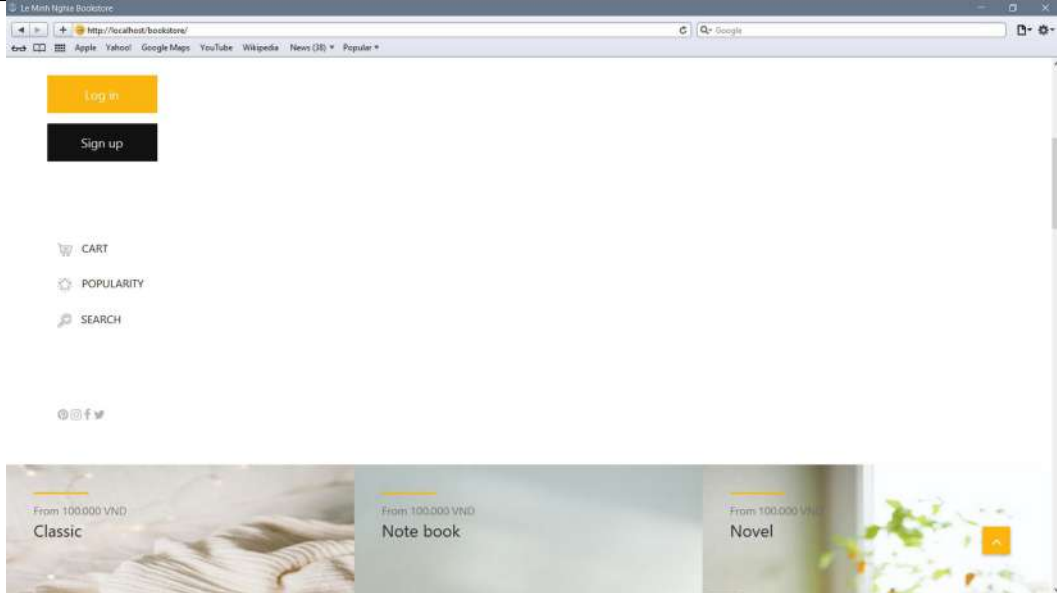
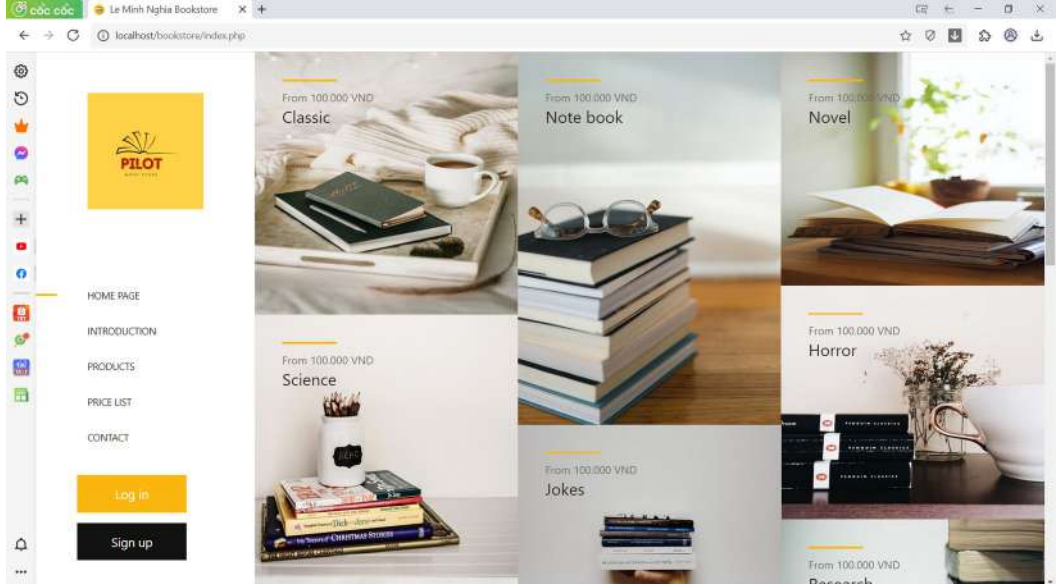
5	Test the interface of cart page	Web/ cart.php	Show the interface of cart page		Correct test result	Pass
6	Test register	Web/ register.php	Register successfully		Correct test result	Pass

				<table border="1"> <thead> <tr> <th>id</th><th>username</th><th>password</th><th>created_at</th><th>name</th><th>phone</th><th>address</th></tr> </thead> <tbody> <tr> <td>10</td><td>nghialmgcd201527</td><td>\$2y\$10\$ubmpTAOKFQvTOpVenTOG YnTDZUJh/OnYSSCuUkW39...</td><td>2021-12-17 16:07:01</td><td>Lê Minh Nghĩa</td><td>0824063391</td><td>Huế</td></tr> <tr> <td>11</td><td>nghiatest</td><td>\$2y\$10\$PcWnwITTXCmsnmunYEsJV 1e5y9UI4Tm1MagpCS762H...</td><td>2021-12-24 09:37:40</td><td>Lê Min Nghia</td><td>0824063333</td><td>Da Nang, Viet Nam</td></tr> <tr> <td>12</td><td>joe Biden</td><td>\$2y\$10\$7lrcyRxzn2XCzZL76QFeeuGi2cmW/UDesbXn8uVZwO...</td><td>2021-12-24 20:44:27</td><td>Joe Biden</td><td>0925067751</td><td>Washington, USA</td></tr> </tbody> </table>	id	username	password	created_at	name	phone	address	10	nghialmgcd201527	\$2y\$10\$ubmpTAOKFQvTOpVenTOG YnTDZUJh/OnYSSCuUkW39...	2021-12-17 16:07:01	Lê Minh Nghĩa	0824063391	Huế	11	nghiatest	\$2y\$10\$PcWnwITTXCmsnmunYEsJV 1e5y9UI4Tm1MagpCS762H...	2021-12-24 09:37:40	Lê Min Nghia	0824063333	Da Nang, Viet Nam	12	joe Biden	\$2y\$10\$7lrcyRxzn2XCzZL76QFeeuGi2cmW/UDesbXn8uVZwO...	2021-12-24 20:44:27	Joe Biden	0925067751	Washington, USA		
id	username	password	created_at	name	phone	address																												
10	nghialmgcd201527	\$2y\$10\$ubmpTAOKFQvTOpVenTOG YnTDZUJh/OnYSSCuUkW39...	2021-12-17 16:07:01	Lê Minh Nghĩa	0824063391	Huế																												
11	nghiatest	\$2y\$10\$PcWnwITTXCmsnmunYEsJV 1e5y9UI4Tm1MagpCS762H...	2021-12-24 09:37:40	Lê Min Nghia	0824063333	Da Nang, Viet Nam																												
12	joe Biden	\$2y\$10\$7lrcyRxzn2XCzZL76QFeeuGi2cmW/UDesbXn8uVZwO...	2021-12-24 20:44:27	Joe Biden	0925067751	Washington, USA																												
7	Test login	Web/ login/php	Logn successfully	<p><b>LOG IN</b></p> <p>Username</p> <input type="text" value="joe Biden"/> <p>Password</p> <input type="password" value="*****"/> <p><b>Log in</b></p> <p>What if you don't have an account? <a href="#">Register now.</a></p>  	Correct test result	Pass																												

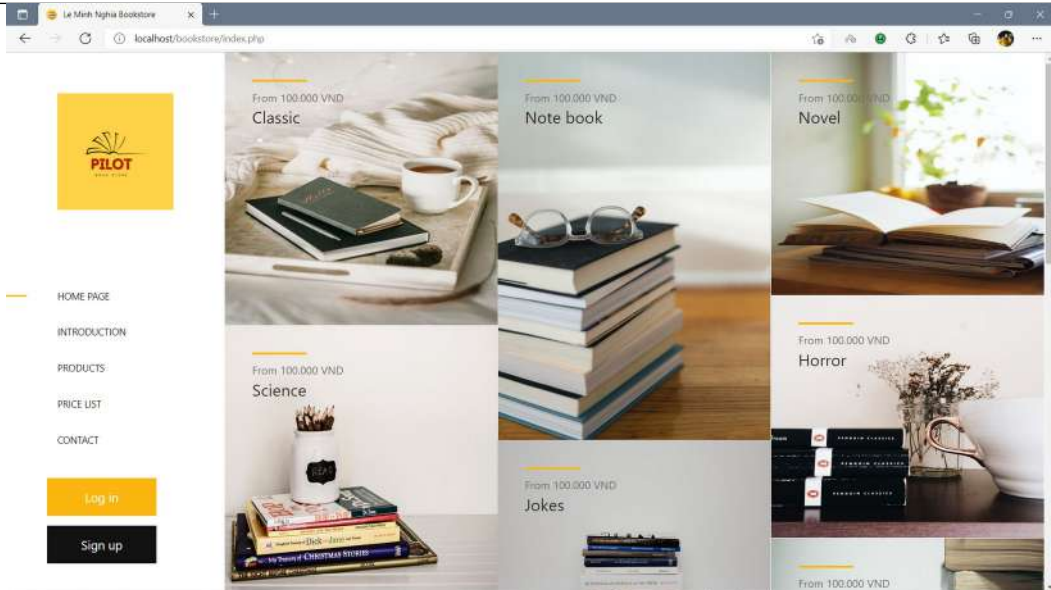
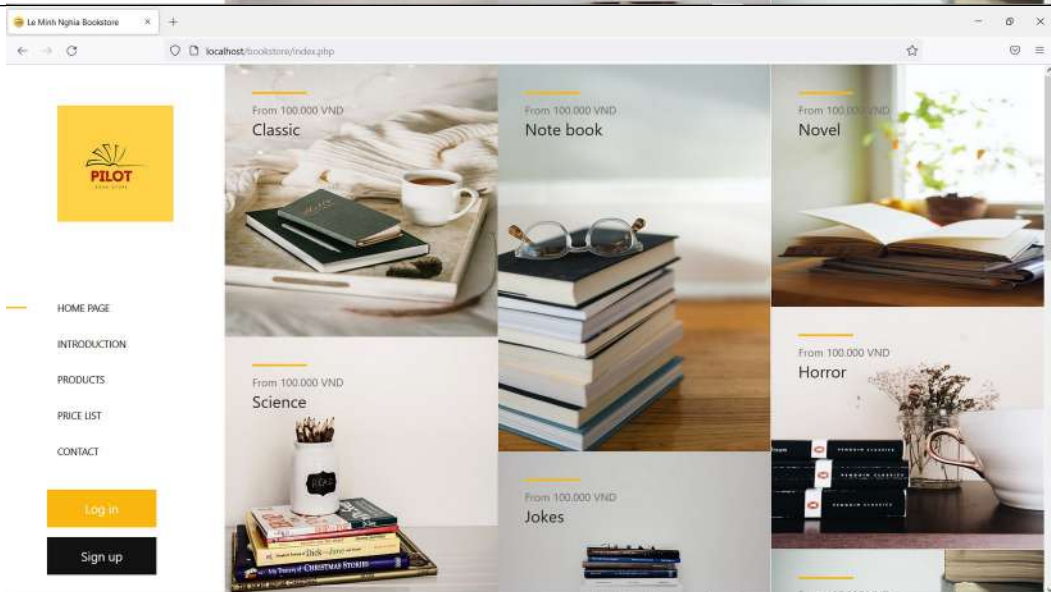
8	Test the interface of product	Web/product.php	Show the interface of product		Correct test result	Pass
9	Test update profile	Web/edit-user.php	Update successfully	<p>UPDATE INFORMATION</p> <p>Successfully updated</p> <p>Full name</p> <input type="text" value="Mr.Joe Biden"/> <p>Phone number</p> <input type="text" value="0925067751"/> <p>Address</p> <input type="text" value="Washington, USA"/> <p><input type="button" value="Update"/> <input type="button" value="Clear"/></p> <p>What if you want to change your password? <a href="#">Click here!</a></p> 	Correct test result	Pass

10	Test order	Web/ cart.php	Order successfully	<p>Cart</p> 	Correct test result	Pass
11	Remove product from cart	Web/ cart.php	Remove successfully	<p>Cart</p> 	Correct test result	Pass

				<p>Cart</p> 		
12	Open web by Google Chrome	Web/index.php	Open successfully		Correct test result	Pass

13	Open web by Safari	Web/index.php	Open successfully but the interface is not same as my design because I have not completed responsive for Safari browser		Correct test result	Pass
14	Open web by Coc Coc	Web/index.php	Open successfully		Correct test result	Pass



15	Open web by Microsoft Edge	Web/index.php	Open successfully		Correct test result	Pass
16	Open web by Firefox	Web/index.php	Open successfully		Correct test result	Pass

## Scheme and software

- Check data testing
- Server database
- Front-end operating system
- Customer operating system
- Browser
- Hardware includes the Operating System Registry
- Network
- Required documentation such as reference documents / setup manuals / installation guides / user manuals

## VI. Evaluation the Quality Assurance (QA) process and review how it was implemented during your design and development stages (M5)

### 1. Quality Assurance

Quality Assurance (QA) is a crucial phase in the design and development of a website or online application, since it ensures that the user experience is excellent (UX). QA is a procedure that controls the design and development of a website or online application in order to test its functioning. While testing your product's user interface (UI) and evaluating your user experience, the QA process exposes design problems and development defects. In Book Store, quality assurance and testing are the most important aspects of the web development process.



Figure 69: Quality Assurance



### **Quality assurance will include two principles**

- Purpose: The product must be fit for the intended purpose.
- Errors: Mistakes should be eliminated the first time.

### **The steps to follow to implement QA for the website are as follows**

- Step 1: Gather and define user needs for website functionality and user interface in the first step.
- Step 2: Create a user interface, a list of features that may be required in the website, and design options ranging from simple to complex to meet all of the given User Requirements.
- Step 3: Design and create the front-end to properly satisfy the user's user interface needs (the components on the interface must be clear, beautiful and avoid causing confusion between the users). components, and goods are shown in the most efficient and well-organized manner possible on the interface).
- Step 4: Deploy the Back-end database system depending on user needs, from simple operations like showing categories, goods, search engines, scrolling to other pages, and so on, to complex capabilities like login authentication, registration, and ordering through the shopping cart function.
- Step 5: Check and maintain the website on a regular basis to eliminate problems; if there are a lot of issues, you may collect user feedback and subsequently enhance the site to give visitors with a high-quality, completely functional shopping experience.

## 2. Applying QA in my website

**Match the original design:** To confirm that the coding result is accurate to the original website design, compare the original design you authorized with the customer with the final product. Sure, there will be some adjustments that have been discussed and debated along the process, but you should be aware of them and should not be surprised. Are you sure the font styles are correct? Is a necessary list the same as a bulleted list? If you have scrolling effects on your site, ensure sure they perform smoothly and elegantly.

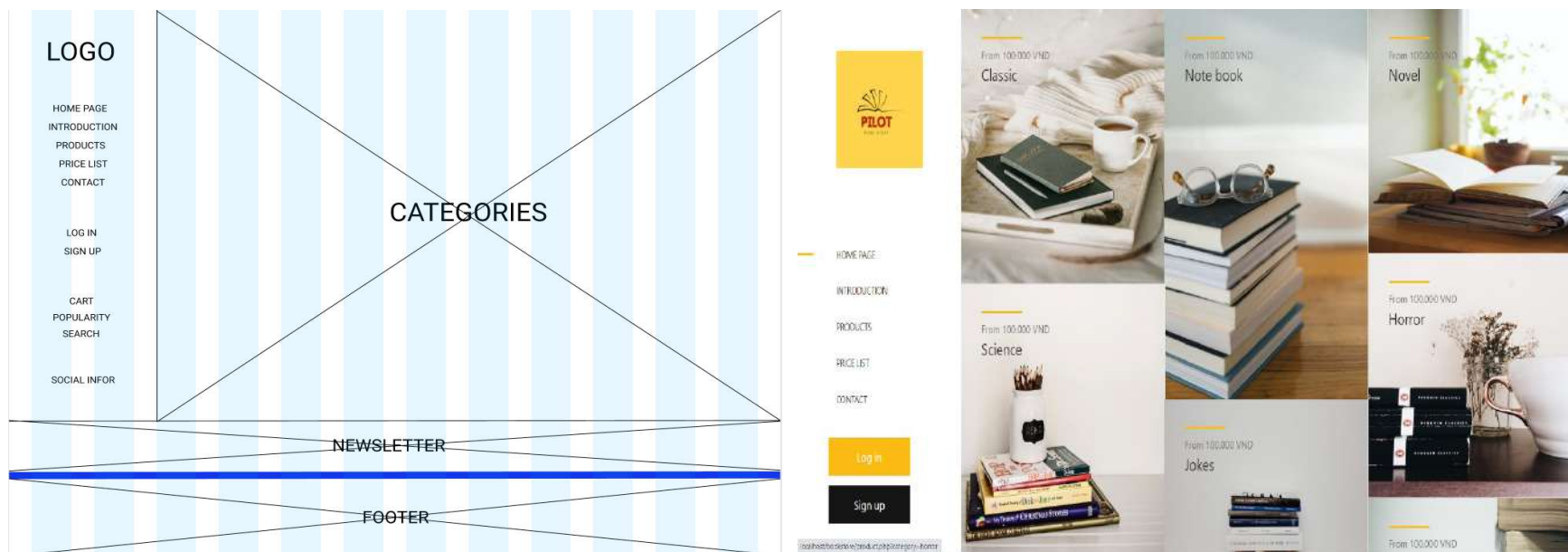


Figure 70: Wireframe and Prototype

**Images:** Make certain that your website's photos are flawless. Examine the image's resolution, breadth, and height for quality. Apart from that, determine which watermarks are left over and ensure that they are altered, with an attractive distance between them.

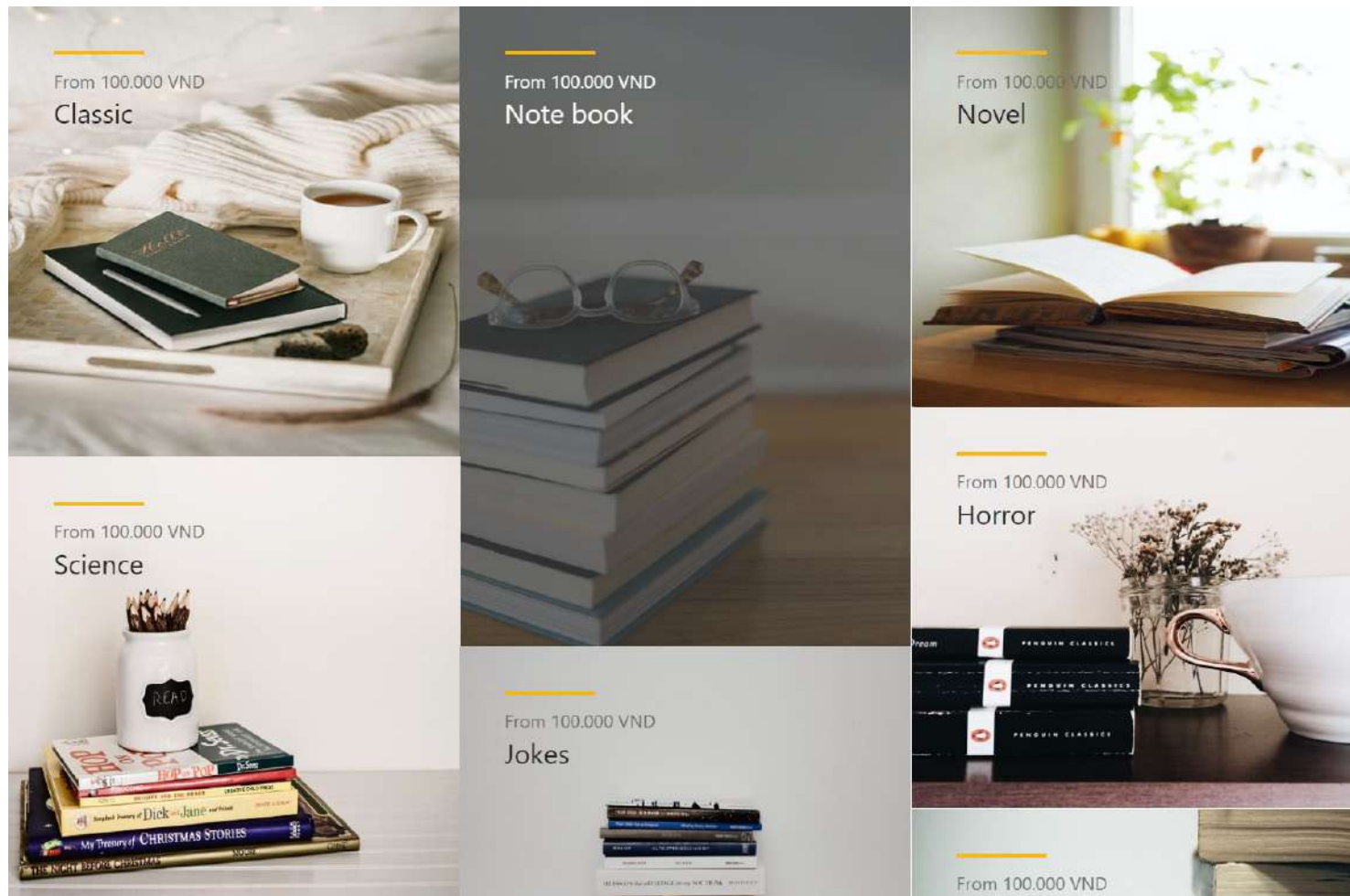


Figure 71: Images

**Content:** Make sure your content has all the pages it needs to launch, is quality, and all has proper grammar, spelling, and more.

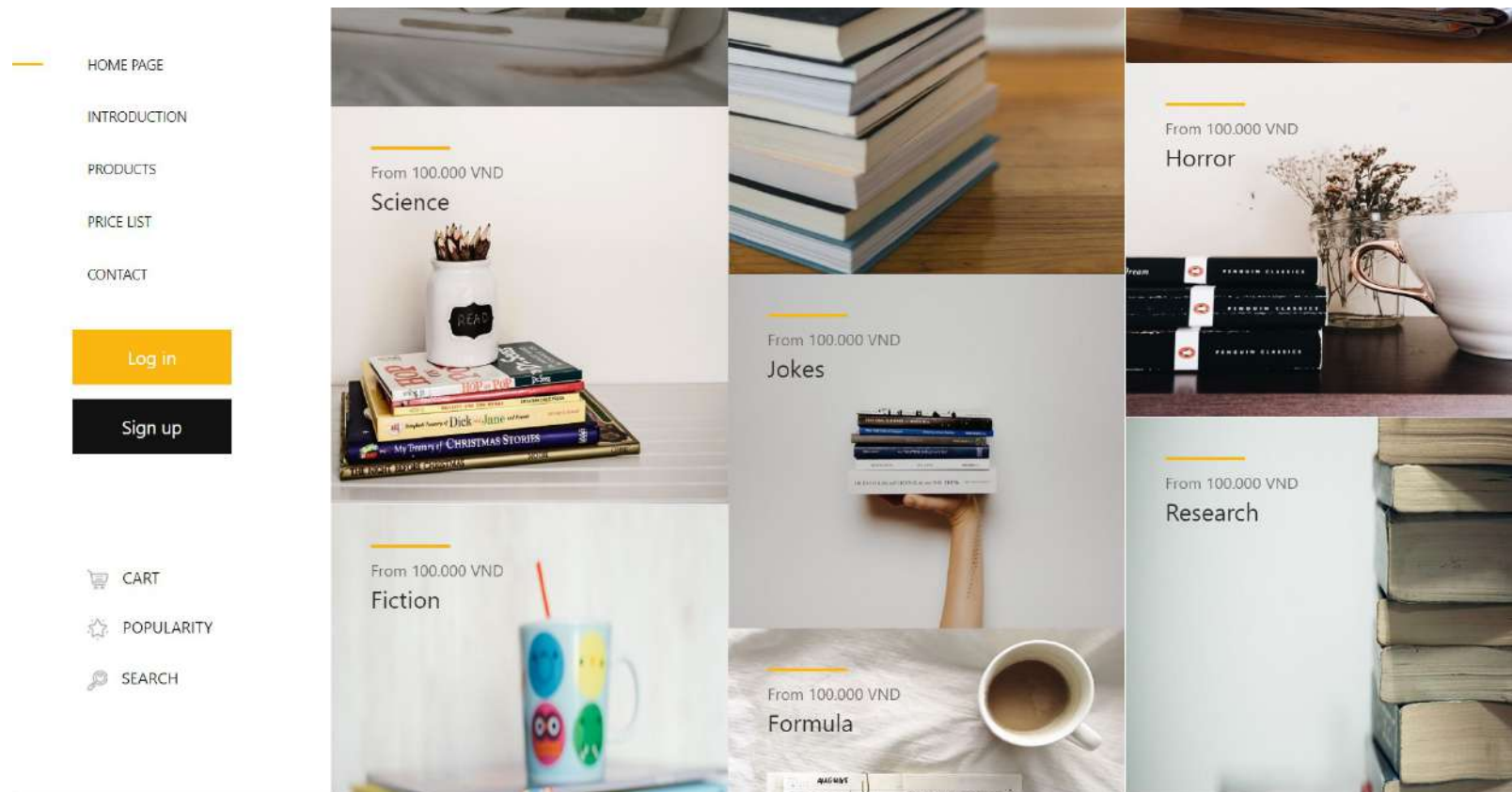
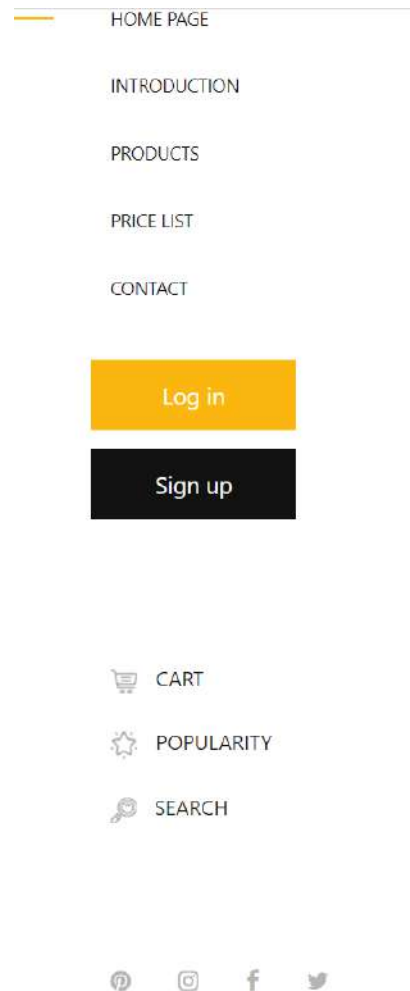


Figure 72: Content

**Menus and Links:** Perform all of the menu items' tasks, making ensuring that their links all point to the right places and that no pages are lacking information. In many situations, your links will also include PDFs and other papers available for download from your website. Check to see if website links open in the same window as your website or in a separate window as you requested.



*Figure 73: Navigation bar*

**Email forms:** Fill up your own email forms, double-checking that all fields are relevant, accurate, and functional. Then double-check your email for the results to ensure you've received them.

**Browser test:** To compare the performance of your website across many browsers, test it in different browsers. In rare circumstances, compatibility issues with older or less popular browsers may be the problem. As a result, test your website on the most common browsers, including Microsoft Edge, Safari, and FireFox, as well as Google Chrome.



Figure 74: Performance by Chrome



**Privacy policy:** Is it Required for a Website to Have a Privacy Policy? Some web developers employ a canonicalized version of your site and ensure that it is present. Make sure you have a more stringent language requirement if you're gathering sensitive information.

**Test:** Check out any ecommerce components or bespoke systems you may have on your website! When using a third-party payment gateway, try completing a payment and confirming that it flows to your bank account. Keep in mind that if you make a charge on your own, you'll have to reimburse yourself afterwards.

**Website responsiveness:** Make sure your web interface works on both PCs and mobile devices. One thing to keep in mind is that having a mobile-friendly website layout will help you attract more clients.

**VII. Critically evaluation the results of your Test Plan and include a review of the overall success of your multipage website; use this evaluation to explain any areas of success and provide justified recommendations for areas that require improvement (D3)**

- 1. Evaluate the web**
  - a. Front-end**

Through the testing and the case study presented above, I give the following assessment

In terms of the front-end, the findings reveal that for the Web project, I need to start utilizing Wiframe to design ahead of time, or else I'll have problems after the Web code is ready. To avoid overwriting difficulties between the frameworks, image material must be provided, and the framework must be properly researched.

Although the previous Case Study highlighted the usage of bootstrap for layout, it does not cover every scenario that may arise on your website. As a result, I had to override and fix them using ID classes. The naming of components in SCSS and JS must adhere to BEM guidelines. The urls and queries made to other pages have been more smoothly deployed thanks to JS.

**Conclusion:** The creation of a static web page necessitates a great deal of aesthetic considerations, attention to detail, working in line with the original design, the proper method, and SEO content criteria for Web pages.

## b. Back-end

On the server, I utilize the PHP programming language. The PHP syntax is well-organized and logical. Even command functions are simple to grasp. The nicest thing about PHP is that it allows you to make modifications even after you've started the project, which saves you time. PHP has the potential to be an efficient language, depending on how I code. When developing programs, it is scalable, and it can also be utilized to create a huge number of applications.

PHP is a free web language since it is an open-source project. Purchasing pricey licenses or software does not cost anything. It can operate with a variety of databases, including MySQL, Apache, and PostgreSQL. The cost of creating a website with PHP is quite low. Because PHP supports tags, website developers may add and/or combine HTML tags to create highly dynamic content.

**Conclusion:** Creating and optimizing the application was a breeze for me. I don't have to develop any new codes or command functions since I can make adjustments to the current ones and use them. When using PHP, I don't have to worry about writing codes in the correct location because they are typed within tags. As a result, functions and scripts need not be written in any particular order as long as they are included within the tags. PHP has a large, active, and supportive PHP community. Furthermore, this programming language provides a wealth of resources, such as instructions, functions, and codes, that may be simply rebuilt and reused at no cost. PHP is one of the most popular server-side programming languages due to its simplicity of use, integration, cost effectiveness, and accessibility.

## 2. Recommendations for improvements

### Front-end

- Set up CSS with JS on the Notify Web page.
- To make the most of CSS command lines on Web pages, use SCSS.
- Improving the site's sophisticated SEO.
- Improve the aesthetics of Web sites with pictures, making them smoother and faster to load.
- Enhance your ability to utilize BEM standard naming and write basic CSS commands without having to use the framework.



## **Back-end**

- Create API functions for payments.
- After identifying consumers with Cookies and Session, proceed to load fresh data more quickly.
- Using the HTTPS protocol, improve the security of Web pages.
- The admin page's improved user interface makes it easier to manage personnel and customers. Add more options for printing reports.

## **3. Advantages, disadvantages of my web project**

### **Advantages**

- Attractive UI, clear graphics, clear presentation characteristics, and a footer with a printed Dev image.
- To establish Web homogeneity, the pages are formatted according to the base.htm standard.
- User-friendly sitemap; test consumers are happy with the remaining 99 percent - 1 percent without purchasing.
- The products are well labeled.
- In JS, notifications assist users in understanding what they have done (Message for logging in and signing up).
- Use Lazy Loading to load Web pages rapidly.

### **Disadvantages**

- The product system has been streamlined.
- Payment via PayPal API has not yet been confirmed.
- There is no way to save favourites.
- In the following frames, there is no good responsive.
- Make several requests for images (Src = "Hyper Link Image").
- Using Back-end Code is incompatible with a wide range of device hardware.

## REFERENCES

CENTRE FOR ACCESSIBILITY, n.d. *What is the WCAG Standard? - Centre For Accessibility Australia*. [online] Centre For Accessibility Australia. Available at: <https://www.accessibility.org.au/guides/what-is-the-wcag-standard/#:~:text=The%20Web%20Content%20Accessibility%20Guidelines,accessible%20to%20people%20with%20disabilities> [Accessed 26 December 2021].

Visual Paradigm, n.d. *What is Use Case Diagram?*. [online] Visual-paradigm.com. Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/> [Accessed 26 December 2021].

Affairs, A., n.d. *Wireframing*. [online] Usability.gov. Available at: <https://www.usability.gov/how-to-and-tools/methods/wireframing.html> [Accessed 26 December 2021].

Tom Hallissey, 2017. *What is a Sitemap, and what is its Purpose in Web Design?*. [online] Fatguymedia.com. Available at: <https://www.fatguymedia.com/web-design/what-is-a-sitemap/> [Accessed 26 December 2021].

Lucid Content Team, n.d. *How to Create Software Design Documents*. [online] Lucidchart. Available at: <https://www.lucidchart.com/blog/how-to-create-software-design-documents#:~:text=What%20are%20software%20design%20documents,%2C%20goals%2C%20etc> [Accessed 26 December 2021].

Technology, T., 2018. *Top 5 Challenges in Web Application Development - TP&P Technology*. [online] TP&P Technology. Available at: <https://www.tpptechnology.com/difficulties-in-web-application-development> [Accessed 26 December 2021].

Alexander S. Gillisv, n.d. *What is Quality Assurance?*. [online] SearchSoftwareQuality. Available at: <https://searchsoftwarequality.techtarget.com/definition/quality-assurance> [Accessed 26 December 2021].

Shreya Bose, 2020. *How to set up QA processes from scratch*. [online] BrowserStack. Available at: <https://www.browserstack.com/guide/setup-qa-process> [Accessed 26 December 2021].

