

MAJOR PROJECT

ON

E-SHOP Website

COMPUTER ENGINEERING

(6th sem, Sec - B)

2022 – 2025

Submitted to:-
Mr. Jatin Verma

Submitted by:-
Saurabh Kumar (10622560)



**Delhi Skill and
Entrepreneurship University**
UGC Recognized

**GURU NANAK DEV INSTITUTE OF
TECHNOLOGY**

Rohini Sector 15 Delhi 110089

Acknowledgement

I express my heartfelt gratitude to everyone who contributed to the successful completion of this project, "**E-SHOP.**"

First and foremost, I would like to thank my institution and the department for providing me with the opportunity to work on this project and for offering the necessary guidance and resources.

I am deeply indebted to my mentor/guide, **Mr. Jatin Verma**, whose continuous support, expertise, and valuable feedback were pivotal in shaping this project. Your guidance throughout every phase of this work has been invaluable.

I would also like to thank my friends and peers for their encouragement and support. Their constructive criticism and feedback helped me enhance the quality of this project.

This project has been a valuable learning experience, and I am truly thankful to everyone who helped me along the way.

Name: Saurabh Kumar (10622560)

Declaration

I hereby declare that the project titled "E-shop" submitted as part of my major project work is a result of my independent effort and research. The work presented here is original and has not been submitted in any form to any other institution or organization for any purpose.

I have developed this project entirely on my own, using resources and tools available in the public domain. Any references or external materials used in this project have been properly acknowledged.

I take full responsibility for the content of this project and ensure that it meets the ethical and academic standards required for its submission.

Saurabh kumar(10622560)

Abstract:

The **E-Shop** project is a responsive and user-friendly e-commerce website developed using HTML, CSS, JavaScript, and the Bootstrap framework. It enables users to browse a variety of products, view detailed information, manage a shopping cart, and complete a checkout process with order confirmation.

Bootstrap plays a key role in ensuring a consistent and mobile-friendly layout across all pages, enhancing the overall user experience. The site includes essential features such as product listings, search functionality, and informational pages like About and Contact. This front-end prototype lays the groundwork for a complete online shopping platform and offers scope for future expansion through backend integration, secure payment gateways, and user account systems.

Table of Contents:

1. Introduction
2. Objectives
3. Tools and Technologies
4. Features
5. Code Walkthrough
 - ⌚ HTML
 - ⌚ CSS
 - ⌚ JavaScript
6. Working Flow of the Application
7. Testing and Debugging
8. Results
9. Conclusion
10. Future Scope
11. Images of Website
12. Coding
13. GitHub Deployment

1. Introduction

The **E-Shop** project is a front-end e-commerce website developed to replicate the core functions of an online shopping platform. It allows users to explore various products, view detailed information, add items to a cart, and proceed to checkout. Built using HTML, CSS, JavaScript, and Bootstrap, the site offers a clean interface and responsive design that works seamlessly across different screen sizes.

Bootstrap plays a key role in ensuring visual consistency and mobile compatibility throughout the website. Pages like Home, Products, Cart, Checkout, Contact, and Order Confirmation are designed to enhance user experience and mimic real-world e-commerce flows. While the current version focuses on front-end functionality, it lays the groundwork for integrating backend services, user accounts, and secure payment systems in future development phases.

2. Objectives

The **E-SHOP** project aims to achieve the following key objectives:

1. **Develop a responsive e-commerce website:** Using HTML, CSS, JavaScript, and Bootstrap to create a fully functional online shopping platform. The website will adapt to various screen sizes, ensuring accessibility across desktops, tablets, and mobile devices.
2. **Provide seamless user experience:** Allow users to browse through a wide range of products, view detailed information, add items to the cart, and complete purchases with an intuitive checkout process. The aim is to make the shopping experience smooth and efficient.
3. **Implement core e-commerce functionalities:** Simulate essential e-commerce features such as product listings, detailed product pages, cart management, and order confirmation. This will help users interact with the site as they would on a real online shopping platform.
4. **Create a user-friendly, adaptable layout:** Ensure the website's layout is consistent and easy to navigate, whether accessed on a small smartphone screen or a larger desktop monitor. The use of Bootstrap will ensure responsiveness across different devices.

5. **Enhance front-end development skills:** Provide an opportunity to apply web development techniques and tools in a real-world context, focusing on user interface design, interactivity, and functionality.
6. **Establish a scalable prototype:** Develop a basic, scalable prototype that can later be extended with backend functionalities, such as user authentication, real-time inventory management, and secure payment gateways.
7. **Lay the groundwork for future enhancements:** This project serves as a foundational model for a full-fledged e-commerce platform. The front-end can be integrated with backend services, enabling more complex features like personalized user experiences and real-time product tracking.

3. Tools and Technologies

HTML (Hypertext Markup Language): : HTML serves as the foundational structure for the E-Shop website, organizing the content into various elements such as headings, paragraphs, images, and links. It enables the creation of semantic elements like product descriptions and navigation, which help with both accessibility and search engine optimization.

CSS (Cascading Style Sheets): CSS is responsible for defining the look and feel of the website. It controls the layout, colors, typography, and positioning of all elements, creating a visually engaging user experience. The CSS in this project ensures that the website is both aesthetically pleasing and functional, allowing it to adapt seamlessly to different screen sizes and devices, providing a consistent experience for all users.

JavaScript: JavaScript is used to add interactivity and dynamic elements to the website, such as product filtering, adding items to the shopping cart, and handling user input during the checkout process. It enables real-time updates to the cart and other sections of the website without requiring a page reload. JavaScript enhances the user experience by making the website more responsive and interactive, providing instant feedback and updates to users.

Bootstrap Framework: Bootstrap is the primary framework used in your project to ensure the website is responsive and mobile-friendly. It helps with layout design, grid systems, buttons, navigation bars, and other UI elements, offering a consistent and modern look across all devices.

Visual Studio Code (VS Code): VS Code is a lightweight and efficient code editor used for writing and managing the E-Shop project's code. It

offers advanced features such as syntax highlighting, IntelliSense, and debugging tools, making it easier to spot errors and improve code quality. With built-in Git support and an integrated terminal, VS Code is a comprehensive tool for front-end development, allowing for efficient workflow management.

Git and GitHub: Git is used for version control, allowing you to track changes and collaborate with others. GitHub serves as the cloud-based repository to store your code and track your project's progress.

4. Features

- 1. Product Browsing:** The E-Shop website offers a user-friendly product browsing experience, where users can explore a wide range of products categorized by type, price, or popularity. Each product listing includes essential information such as product name, price, description, and image. Users can filter and sort products based on various criteria, making it easy to find specific items.
- 2. Product Detail Pages:** Every product has a dedicated product detail page that provides in-depth information, including larger images, detailed descriptions, specifications, and customer reviews. This helps users make informed decisions before adding products to their shopping cart. The product page also includes related items or recommendations to encourage further exploration.
- 3. Shopping Cart:** The website features an interactive shopping cart that allows users to add, remove, or modify the quantity of items. The cart dynamically updates as users interact with it, showing a summary of the selected products, their prices, and the total cost. This feature ensures a smooth shopping experience and helps users keep track of their purchases.
- 4. Checkout Process:** The checkout process is designed to be straightforward and user-friendly, guiding users through a series of simple steps to complete their purchase. Users can review their cart, enter shipping and payment details, and confirm their order. The process minimizes friction and ensures users can easily finalize their purchase without confusion.
- 5. Order Confirmation:** After completing the checkout process, users are presented with an order confirmation page that displays the details of their purchase, including the order number, items purchased, total cost, and estimated delivery date. This page provides reassurance to users and confirms that their transaction was successfully processed.

6. **Search Functionality:** The E-Shop website includes a search bar that allows users to quickly find products by entering keywords or product names. The search results are displayed in real time, showing matching products, and allowing users to narrow down their choices with additional filters. This feature enhances navigation and makes it easier to find specific products.
7. **Responsive Design:** The website is fully responsive, meaning it automatically adjusts its layout and design to fit different screen sizes and devices. Whether users are browsing on a desktop, tablet, or smartphone, they will have a consistent and optimized experience. This is made possible through the use of Bootstrap's responsive grid system and mobile-first design principles.
8. **About and Contact Pages:** The E-Shop website includes informational pages such as "About Us" and "Contact Us." The "About Us" page provides background information on the business, its mission, and values, while the "Contact Us" page includes a contact form and other ways for customers to reach out for support, inquiries, or feedback. These pages help build trust and engagement with users.

5. Code Walkthrough

The E-Shop project is built using a combination of HTML, CSS, JavaScript, and Bootstrap to provide a seamless, responsive, and interactive shopping experience. Below is a breakdown of the key sections of the code:

1. HTML Structure

The HTML files (e.g., `index.html`, `products.html`, `product-view.html`, etc.) provide the core structure of the website. The layout is divided into various sections such as:

- **Header:** Contains the navigation bar with links to different pages (Home, Products, Cart, Contact, etc.). It also includes a search bar and a cart icon that displays the number of items in the cart.
- **Main Content:** Each page has a main content area where product listings, product details, and other information are displayed. The content is organized into containers, rows, and columns to make the layout responsive and adaptable to different screen sizes.

2. CSS Styling

CSS is used to style the website, defining the visual appearance of elements. The key CSS files (e.g., `styles.css`) control:

- **Global Styles:** These are general settings applied across the website, such as font choices, background colors, margins, and padding.
- **Layout and Grid:** Custom CSS overrides the Bootstrap grid system to adjust the layout and ensure it fits the design requirements. It controls the positioning of products, cart items, and other elements.

- **Responsive Design:** Using media queries, the CSS ensures that the website adapts to various screen sizes, from large desktop monitors to smaller mobile devices.

3. JavaScript Functionality

JavaScript is used to add interactivity to the website. Key functionalities include:

- **Product Filtering:** On the `products.html` page, JavaScript allows users to filter products based on categories, price range, and other attributes. It updates the product listing dynamically without requiring a page reload.
- **Cart Management:** The shopping cart functionality is powered by JavaScript. Users can add or remove products, update quantities, and see the total price. JavaScript handles updating the cart dynamically, ensuring a smooth shopping experience.
- **Checkout Process:** JavaScript validates the checkout form by ensuring that all required fields (such as address and payment information) are completed before submission. It also displays an order confirmation page upon successful purchase.
- **Event Listeners:** JavaScript is used to manage events like button clicks, form submissions, and input changes. It ensures that the website responds immediately to user actions, such as adding a product to the cart or filtering products.

4. Bootstrap Framework

Bootstrap is used to quickly build the layout and ensure responsiveness. It provides a set of pre-designed components, such as:

- **Navigation Bar:** Bootstrap's navbar component is used to create the website's header, making it easy to create a responsive and mobile-friendly navigation menu.
- **Grid System:** The Bootstrap grid system is used to create the flexible layout of the website, allowing the product listings to adjust based on screen size. The layout is designed to show products in a multi-column format on large screens and a single column on smaller devices.

- **Buttons and Forms:** Bootstrap is also used for styling buttons (e.g., "Add to Cart" and "Checkout") and form elements, providing a consistent and polished appearance.

5. GitHub Repository and Version Control

The project is version-controlled using **Git** and hosted on **GitHub**. The GitHub repository contains all the source code, including HTML, CSS, JavaScript files, and assets (images, icons, etc.). GitHub allows for version tracking, collaboration, and easy deployment of the project.

- **Branches:** The project is organized into branches for different features and tasks. For example, a branch might be created for the cart functionality, while another branch is dedicated to implementing the checkout process. This ensures that each feature is developed independently and can be tested before merging it into the main branch.
- **Commits:** Each commit contains a message describing the changes made to the project, helping to track progress and revert to earlier versions if needed.

6. Working Flow of the Application

1. **Homepage:** Users land on the homepage, where they can view featured products and navigate to other pages like Products, Cart, and Contact.
2. **Product Browsing:** On the Products page, users browse and filter products, or search for specific items.
3. **Product Details:** Clicking on a product opens its detailed page, where users can view more info and add it to their cart.
4. **Adding to Cart:** Users add products to the cart, which updates the cart icon and the total price dynamically.
5. **Cart Review:** Users review the cart, modify quantities, or remove items before proceeding to checkout.
6. **Checkout:** Users enter their shipping and payment details on the checkout page.
7. **Order Confirmation:** After placing the order, users are shown a confirmation page with order details and a unique order ID.
8. **Post-Purchase:** Users can browse more products or exit the site, with future enhancements planned for user accounts and backend integration.

7. Testing and Debugging

1. **Functional Testing:** Ensured product browsing, cart management, and checkout processes work as expected. Verified dynamic updates and form validation. Testing also included checking edge cases, like adding out-of-stock products to the cart.
2. **Usability Testing:** Tested the website for ease of navigation and ensured it adapts to different devices through responsive design. Feedback from users helped identify any usability issues and improve the flow.
3. **Cross-Browser Testing:** Checked compatibility on browsers like Chrome, Firefox, Safari, and Edge to ensure consistent performance and appearance. Addressed minor issues like font rendering and button alignment differences.
4. **Debugging Tools:** Used **Chrome Developer Tools** for debugging JavaScript errors, inspecting CSS, and identifying performance issues. This helped track down bugs related to product filtering and cart updates.
5. **Performance Testing:** Ensured fast page load times and optimized images and code for better performance. Tested with various network conditions to confirm the site remains functional under slow internet speeds.
6. **Error Handling:** Added form validation and custom error pages to handle issues like missing fields or broken links. Error messages were designed to be user-friendly and guide users toward fixing their input.
7. **User Testing:** Gathered feedback from beta testers to identify bugs and improve usability. Their insights also led to improvements in accessibility features, like clearer font sizes and color contrasts.

8. Results

The E-Shop project successfully demonstrates the core functionality of an e-commerce website, offering users an interactive and responsive platform to browse, select, and purchase products. By using **HTML**, **CSS**, **JavaScript**, and the **Bootstrap** framework, the project provides an intuitive, mobile-friendly interface that adapts seamlessly across different devices and screen sizes.

Key outcomes include:

- **Fully Functional E-Commerce Flow:** Users can browse products, view detailed information, add products to the cart, and complete the checkout process with an order confirmation.
- **Responsive Design:** The site works effectively on desktops, tablets, and mobile devices, ensuring a consistent experience for all users.
- **Smooth User Experience:** Features like dynamic cart updates, form validation, and product search improve usability, making the shopping experience more efficient and enjoyable.
- **Testing & Debugging Success:** Extensive testing across multiple browsers and devices, as well as user feedback, ensured the site is bug-free and functional.

Overall, the E-Shop project provides a solid prototype for a full-scale online shopping platform, with room for future enhancements and feature expansions, such as real-time inventory tracking, payment integration, and personalized user accounts.

9. Conclusion

The E-Shop project successfully demonstrates the fundamental components of an e-commerce website, utilizing **HTML**, **CSS**, **JavaScript**, and **Bootstrap** to create a responsive and user-friendly interface. The platform allows users to browse products, manage their cart, and complete the checkout process efficiently, offering a seamless shopping experience. Rigorous testing ensured the website functions smoothly across various devices and browsers.

This project serves as a foundational prototype for future developments, such as backend integration, secure payment systems, and user account management. While the current version focuses on front-end features, it paves the way for a fully functional e-commerce platform with the potential for scalability and future enhancements.

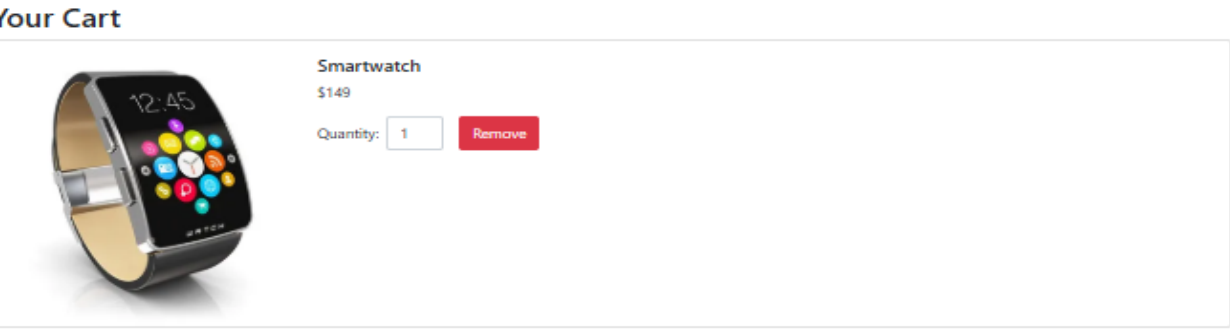
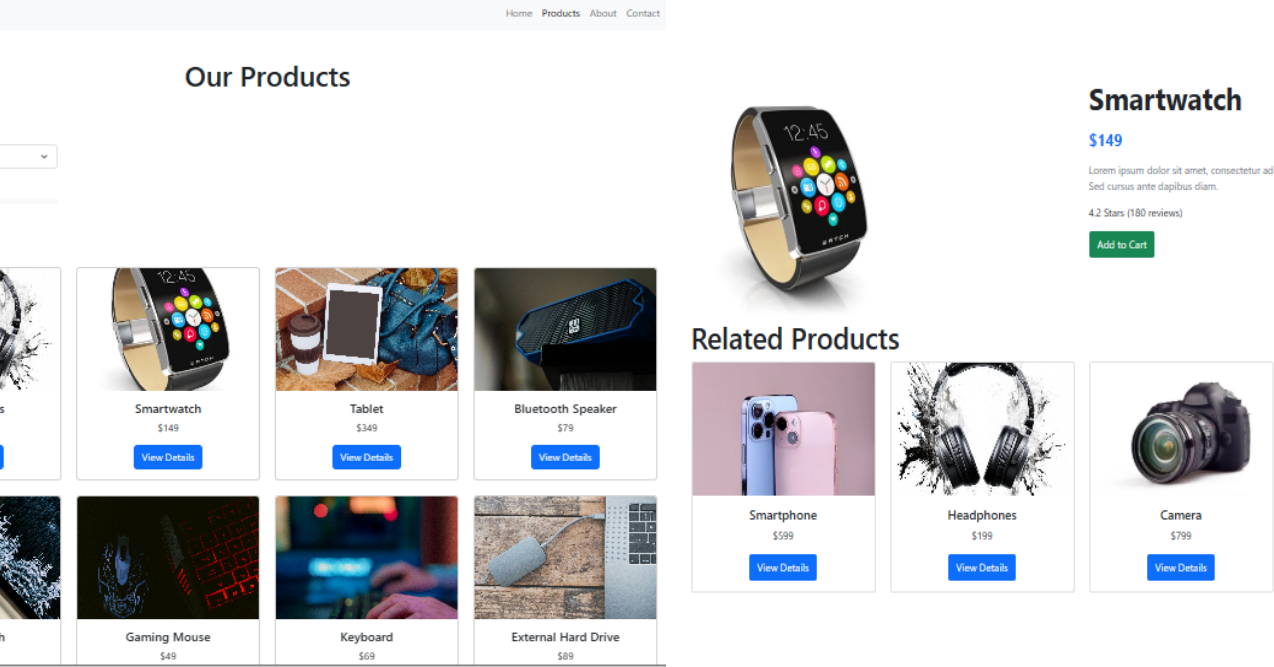
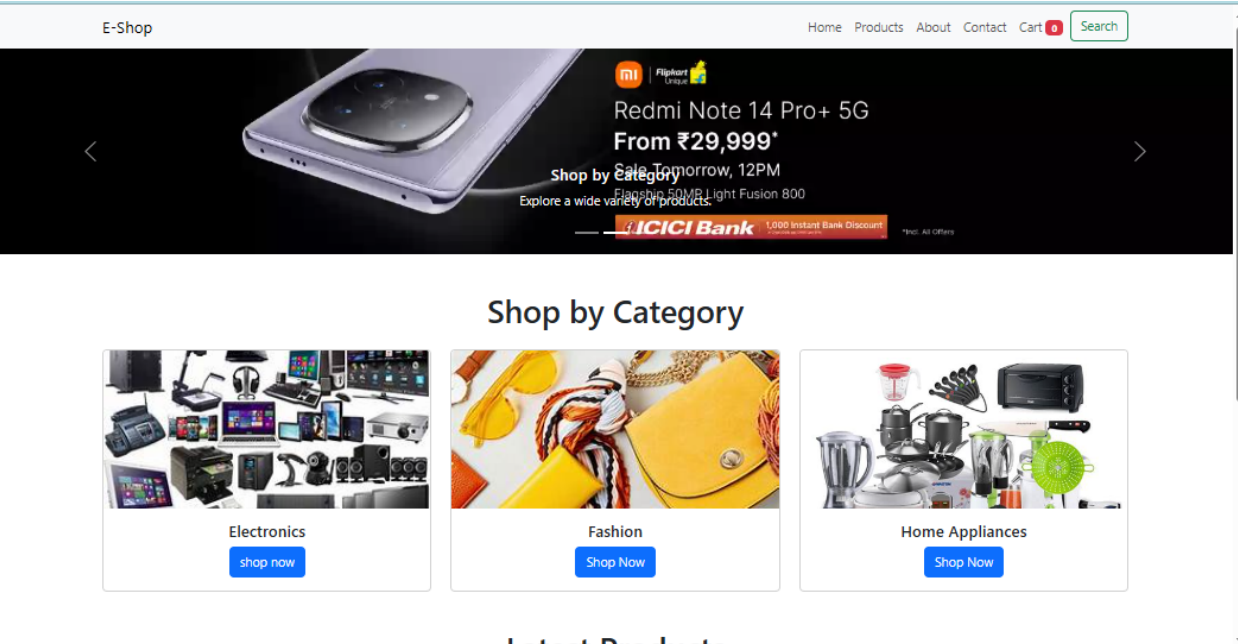
10. Future Scope

The E-Shop project serves as a solid foundation for developing a fully functional and feature-rich e-commerce platform. In the future, several enhancements can be implemented to expand its capabilities and improve the user experience:

1. **Backend Integration:** Implementing a backend system using technologies like **Node.js**, **PHP**, or **Python (Django/Flask)** will allow for dynamic content, user authentication, and real-time updates for product inventory and user accounts.
2. **Payment Gateway Integration:** Adding secure payment gateways (e.g., **Stripe**, **PayPal**, **Razorpay**) will enable users to make transactions, providing a complete online shopping experience.
3. **User Accounts and Authentication:** Developing a user account system with features like order history, wishlists, and personalized recommendations will improve customer engagement and retention.
4. **Real-Time Inventory Management:** Integrating inventory management systems to track stock levels and automate restocking notifications can streamline operations for both customers and store administrators.
5. **Advanced Search and Filtering:** Enhancing the product search and filtering options using machine learning algorithms will allow users to find products more efficiently based on preferences and past browsing behavior.
6. **Mobile App Development:** Developing a mobile application for iOS and Android would extend the platform's reach and provide users with a dedicated, optimized shopping experience.
- Performance Optimization:** Further optimizing page load speeds, image compression, and server-side caching will ensure faster browsing and a smoother experience, especially during high traffic periods.

With these enhancements, the E-Shop project has the potential to evolve into a fully-fledged, scalable e-commerce solution, ready to handle larger traffic volumes, offer personalized experiences, and integrate with various external systems.

11. Images of Website



12. Coding

→ index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>E-commerce Website</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/
bootstrap.min.css" rel="stylesheet">
  <link rel="stylesheet" href="css/styles.css">
  <!-- Font Awesome CDN -->
  <link href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.0.0-beta3/
css/all.min.css" rel="stylesheet">

</head>
<body>
  <!-- Navbar -->
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container">
      <a class="navbar-brand" href="#">E-Shop</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav ms-auto">
          <li class="nav-item">
            <a class="nav-link" href="index.html">Home</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="products.html">Products</a>
          </li>

          <li class="nav-item">
```

```

        <a class="nav-link" href="about.html">About</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="contact.html">Contact</a>
    </li>
    <li class="nav-item">

        <a class="nav-link" href="cart.html">
            Cart
            <span id="cart-count" class="badge bg-danger text-
white">0</span>
        </a>

    </li>

    <li class="nav-item">
        <a href="search.html" class="btn btn-outline-success">Search</a>
    </li>

</ul>
</div>
</div>
</nav>

<!-- Carousel -->
<div id="homeCarousel" class="carousel slide" data-bs-ride="carousel">
    <div class="carousel-indicators">
        <button type="button" data-bs-target="#homeCarousel" data-bs-slide-to="0"
class="active" aria-current="true" aria-label="Slide 1"></button>
        <button type="button" data-bs-target="#homeCarousel" data-bs-slide-to="1"
aria-label="Slide 2"></button>
        <button type="button" data-bs-target="#homeCarousel" data-bs-slide-to="2"
aria-label="Slide 3"></button>
    </div>
    <div class="carousel-inner">
        <div class="carousel-item active">
            
            <div class="carousel-caption d-none d-md-block">
                <h5>Welcome to E-Shop</h5>
            </div>
        </div>
    </div>

```

```

        <p>Find the best deals here!</p>
    </div>
</div>
<div class="carousel-item">
    
    <div class="carousel-caption d-none d-md-block">
        <h5>Shop by Category</h5>
        <p>Explore a wide variety of products.</p>
    </div>
</div>
<div class="carousel-item">
    
    <div class="carousel-caption d-none d-md-block">
        <h5>Latest Products</h5>
        <p>Check out the newest arrivals!</p>
    </div>
</div>
</div>
<button class="carousel-control-prev" type="button" data-bs-
target="#homeCarousel" data-bs-slide="prev">
    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Previous</span>
</button>
<button class="carousel-control-next" type="button" data-bs-
target="#homeCarousel" data-bs-slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Next</span>
</button>
</div>

```

→ styles.css


```

#scrollToTopBtn {
  position: fixed;
  bottom: 20px;
  right: 20px;
  z-index: 1000;
}
#scrollToTopBtn:hover {
  background-color: #495057; /* Slightly lighter on hover */
}
/* Custom button styles */
#scrollToTopBtn
{
  background-color: hsla(350, 100%, 57%, 0.875); /* Example color */
  border: 2px solid black;
}

/* Ensure links are clickable */
a {
  pointer-events: auto !important;
}

/* Make images responsive */
img {
  max-width: 100%;
  height: auto;
}

/* Adjust font sizes for small screens */
h1, h2, h3, h4 {
  font-size: calc(1.5rem + 1vw);
}
/* Add subtle hover effect for products */
.card:hover {
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  transform: translateY(-5px);
  transition: transform 0.3s ease-in-out, box-shadow 0.3s ease-in-out;
}

```

Script.js

```

let debounceTimer;
document.getElementById('searchQuery').addEventListener('input', function() {

```

```

clearTimeout(debounceTimer);
debounceTimer = setTimeout(function() {
    performSearch(); // Call the search function after a delay
}, 300); // 300ms delay
});

// Get the category filter and price range slider
const categoryFilter = document.getElementById('categoryFilter');
const priceRange = document.getElementById('priceRange');
const priceValue = document.getElementById('priceValue');

// Update the priceValue span when the range slider changes
priceRange.addEventListener('input', function () {
    priceValue.textContent = priceRange.value;
    filterProducts(); // Call filter function on price range change
});

// Get all product cards
const productCards = document.querySelectorAll('.card');
// Update products based on category and price filter
categoryFilter.addEventListener('change', function () {
    filterProducts(); // Call filter function when category is changed
});
// Function to filter products
function filterProducts() {
    const selectedCategory = categoryFilter.value;
    const selectedPrice = parseInt(priceRange.value);

    productCards.forEach(card => {
        const cardCategory = card.getAttribute('data-category');
        const cardPrice = parseInt(card.getAttribute('data-price'));

        // Check if the product matches the category and price criteria
        if ((selectedCategory === 'All Categories' || selectedCategory ===
cardCategory) &&
            cardPrice <= selectedPrice) {
            card.style.display = 'block'; // Show card
        } else {
            card.style.display = 'none'; // Hide card
        }
    });
}

```

```
// Set initial price value on page load
priceValue.textContent = priceRange.value;

// Initial call to filter products when page loads
filterProducts();
```

→ cart.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Your Cart</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
  <!-- Navbar -->
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="index.html">E-Com</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav ms-auto">
          <li class="nav-item">
            <a class="nav-link" href="index.html">Home</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="products.html">Products</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" href="cart.html">Cart</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
```

```

</nav>
<!-- Cart Section -->
<div class="container mt-5">
  <h2>Your Cart</h2>
  <div id="cart-items" class="row">
    <!-- Cart items will be dynamically inserted here -->
  </div>
  <div id="cart-summary" class="mt-3">
    <h3>Total: $<span id="total-price">0.00</span></h3>
    <button id="checkout-btn" class="btn btn-primary">Proceed to
Checkout</button>
  </div>
</div>

```

checkout.html

```

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Checkout - E-Shop</title>

  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet">

</head>

<body>

  <!-- Navbar -->

  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <div class="container-fluid">

      <a class="navbar-brand" href="index.html">E-Shop</a>

      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">

```

```

        <span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

    <ul class="navbar-nav ms-auto">

        <li class="nav-item">

            <a class="nav-link" href="index.html">Home</a>

        </li>

        <li class="nav-item">

            <a class="nav-link" href="products.html">Products</a>

        </li>

        <li class="nav-item">

            <a class="nav-link" href="cart.html">Cart</a>

        </li>

        <li class="nav-item">

            <a class="nav-link active" href="checkout.html">Checkout</a>

        </li>

    </ul>

</div>

</div>

</nav>

<!-- Checkout Section -->

<div class="container mt-5">

    <h1 class="text-center mb-4">Checkout</h1>

```

contact.html

```

<!-- Navbar -->

<nav class="navbar navbar-expand-lg navbar-light bg-light">

```

```
<div class="container">

    <a class="navbar-brand" href="#">E-Shop</a>

    <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">

        <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse" id="navbarNav">

        <ul class="navbar-nav ms-auto">

            <li class="nav-item">

                <a class="nav-link" href="index.html">Home</a>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="products.html">Products</a>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="cart.html">Cart</a>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="about.html">About</a>

            </li>

            <li class="nav-item">

                <a class="nav-link active" href="contact.html">Contact</a>

            </li>

        </ul>

    </div>

</div>
```

```
</nav>

<!-- Contact Us Section -->

<div class="container mt-5">

  <h1 class="text-center mb-4">Contact Us</h1>

  <!-- Contact Info -->

  <div class="row">

    <div class="col-md-6">

      <h4>Our Contact Information</h4>

      <p><strong>Address:</strong> 123 E-Shop Street, New Delhi, India</p>

      <p><strong>Phone:</strong> +915053622585</p>

      <p><strong>Email:</strong> sksksk8053@gmail.com</p>

    </div>


```

scroll.html

```
<!-- Scroll to Top Button -->

<button id="scrollToTopBtn" class="btn btn-primary position-fixed bottom-3
end-3 d-none" title="Back to Top">

  &#8679;

</button>

<!-- JavaScript -->

<script defer>

  const scrollToTopBtn = document.getElementById('scrollToTopBtn');

  const productList = document.getElementById('productList');

  // Show button when scrolling down

  window.addEventListener('scroll', () => {

    if (window.scrollY > 200) {


```

```

        scrollTopBtn.classList.remove('d-none');
    } else {
        scrollTopBtn.classList.add('d-none');
    }
});

// Smooth scroll to top
scrollTopBtn.addEventListener('click', () => {
    window.scrollTo({
        top: 0,
        behavior: 'smooth',
    });
});

document.addEventListener('DOMContentLoaded', () => {
    // Perform search when the search button is clicked

    const searchButton =
document.querySelector("button[onclick='performSearch()']");

    if (searchButton) {
        searchButton.addEventListener('click', performSearch);
    }

    function performSearch() {
        const query = document.getElementById('searchQuery').value.toLowerCase();

        const products = document.querySelectorAll('.card'); // Make sure we're
targeting the correct elements

        let resultsFound = false;

        console.log('Searching for:', query); // Log the search query

        products.forEach(product => {

            const productName = product.querySelector('.card-
title').textContent.toLowerCase();

```



```

        const productCategory = product.getAttribute('data-
category').toLowerCase();

        // Log each product's name and category being checked

        console.log('Checking product:', productName, 'Category:',
productCategory);

        // Check if the query matches the product name or category

        if (productName.includes(query) || productCategory.includes(query)) {

            product.style.display = 'block'; // Show matching product

            resultsFound = true;

            console.log('Match found:', productName);

        } else {

            product.style.display = 'none'; // Hide non-matching product

        }

    });

    // Show the correct results header

    const resultsHeader = document.getElementById('resultsHeader');

    resultsHeader.textContent = resultsFound ? 'Search Results' : 'No Results
Found';

    }

});

</script>

<!-- Bootstrap JS -->

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bund
le.min.js"></script> <script src="js/scripts.js"></script>

```

13. GitHub Deployment

The E-SHOP project has been deployed on GitHub Pages to make it accessible to a broader audience. GitHub Pages is a convenient way to host static websites and projects directly from a repository. By deploying the project, users can access the application using a publicly available link without needing to download or run the code locally.

The deployed link can be shared with others for testing or demonstration purposes. Additionally, this deployment ensures the project's version control is maintained on GitHub, allowing for future updates and improvements to be reflected instantly.

Here is the deployment link: <https://saurabhhkumarr.github.io/E-SHOP/>