

Texas College of Management and IT

Internet fundamental Application

Explore, Learn and Excel

Assignment 2

Submitted By: Timothy shakya

Submitted To:

Name: Timothy shakya

Department Of IT

LCID: LC00017002848

Program: BIT

Sections: A

Date: 2024/AUG/15



Abstract

This project involves the development of a dynamic and responsive website using HTML, CSS, and JavaScript. The primary goal was to create a visually engaging and user-friendly website that adheres to modern web development practices. HTML was employed to structure the content, CSS was used to style the website with a cohesive color scheme and typography, and JavaScript was integrated to enhance interactivity and functionality. Key features of the project include a responsive layout that adapts to various screen sizes, interactive elements such as sliders and dropdown menus, and dynamic content updates driven by JavaScript. The project also emphasizes cross-browser compatibility and performance optimization to ensure a seamless user experience across different devices and browsers. This report documents the design and implementation process, including the integration of HTML, CSS, and JavaScript, and evaluates the effectiveness of the final website in meeting its design and functional objectives..

Table of Contents

Table of Contents

Assignment 2.....	1
Department Of IT.....	1
Program:BIT	1
 Abstract.....	 1
1. Introduction	3
1.1 Background	3
1.2 Objectives	3
2. Website Design	3
2.1 Layout and Structure.....	3
2.2 Components.....	4
1. Navigation Bar:.....	4
2. Loader (2):	5
3. Swiper Slider Component:.....	5
4. Product Card:	6
5. Footer:	6
3. CSS components.....	8
3.1 Navigation Bar.....	8
3.2 Loader	10
3.3 Swiper	11
3.4 Product Card	12
3.5 Footer	14
3. Inter-VLAN Routing	15
3.4 Layer 3 switching using SVIs [1].....	15
3.5 Subnetting.....	16
4. Java Scrit.....	16
4.4 Script for loader	16
4.5 Script for slider.....	16
5. Responsiveness	17
Responsive Layout CSS.....	17
2. Responsive Header CSS	17
4. Responsive Loader CSS.....	18
5. Responsive Swiper CSS.....	18
6. Responsive Product Card CSS.....	18
7. Responsive Section CSS.....	19
8. Responsive Footer CSS.....	19
6. CSS code with Bootstrap for better performance	20
6.4 Simulation of overall	20
• Overall Objectives Met:.....	22
7. Conclusion.....	22
7.4 Summary.....	22
7.5 Lessons Learned	22
8. Future Work	22
8.4 Potential Improvements	22
9. References	23

1. Introduction

1.1 Background

In today's digital world, an effective online presence is crucial for businesses and individuals alike. With the proliferation of devices and screen sizes, it is imperative for web developers to design websites that are not only visually appealing but also functional and accessible across various platforms. This project focuses on building a website that leverages the core technologies of HTML, CSS, and JavaScript to create a comprehensive and interactive web experience.

HTML provides the fundamental structure of the website, allowing for the organization and presentation of content. CSS enhances this structure by applying styles that improve visual aesthetics and ensure a cohesive look and feel. JavaScript adds dynamic behavior and interactivity, enabling features such as animations, form validations, and interactive elements that respond to user actions.

The website developed in this project is designed to address modern web design challenges, including responsiveness to different screen sizes, cross-browser compatibility, and user engagement through interactive elements. By integrating these technologies, the project aims to demonstrate best practices in web development and deliver a robust, user-friendly online experience. This background sets the stage for understanding the design principles and implementation strategies used throughout the project.

1.2 Objectives

The main goals of this project are to:

1. **Create an Attractive Website:** Build a website that looks modern and appealing using HTML and CSS. Use advanced design techniques to make it visually engaging.
2. **Add Interactive Elements:** Use JavaScript to include dynamic features such as animations, form validations, and interactive components to make the website more engaging for users.
3. **Ensure the Website Works Well on All Devices:** Design the site so that it looks and works great on different screen sizes and devices, including phones, tablets, and computers.
4. **Follow Best Web Development Practices:** Use good practices in coding with HTML, CSS, and JavaScript to make sure the website is well-organized and easy to maintain.
5. **Improve User Experience:** Make sure the website is easy to navigate and enjoyable to use, with clear and simple interactions.
6. **Document the Process:** Keep a detailed record of how the website was designed, built, and tested, including any problems encountered and how they were solved.

2. Website Design

2.1 Layout and Structure

The website is designed with a clear and organized layout to ensure ease of navigation and visual appeal. It includes the following main sections:

- **Header:** Features a navigation bar with links to different parts of the site and a branding area.



Figure 1:header

Main Content: Comprises several sections such as Home, Product, and Contact. Each section is distinct and contributes to the overall content presentation.

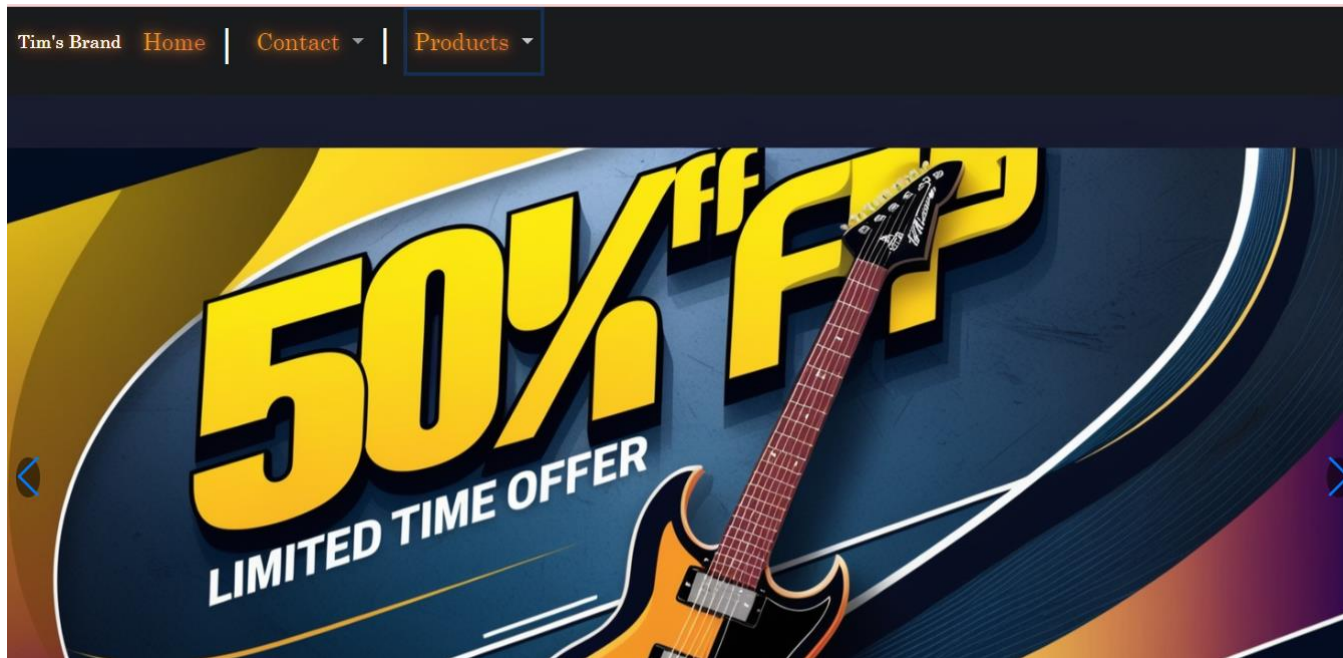


Figure 1:main content

Footer: Contains essential information like copyrights social media links.

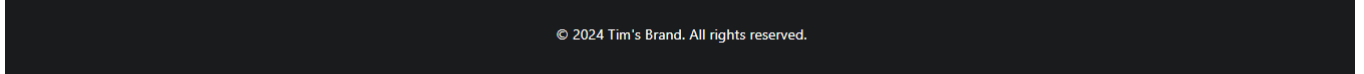


Figure 1:footer content

2.2 Components

The website includes several key components that enhance functionality and user interaction. Each component is described along with a screenshot of its HTML/CSS code and its visual representation:

1. Navigation Bar:

```
<!-- Header Component -->
<nav class="navbar navbar-expand-lg navbar-dark">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Tim's Brand</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">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Home</a>
        </li>
```

```

    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownContact" role="button" data-bs-
toggle="dropdown" aria-expanded="false">
        Contact
      </a>
      <ul class="dropdown-menu" aria-labelledby="navbarDropdownContact">
        <li><a class="dropdown-item" href="#"><i class="fab fa-facebook"></i> Facebook</a></li>
        <li><a class="dropdown-item" href="#"><i class="fab fa-instagram"></i> Instagram</a></li>
        <li><a class="dropdown-item" href="#"><i class="fas fa-phone"></i> Phone Number</a></li>
      </ul>
    </li>
    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-
toggle="dropdown" aria-expanded="false">
        Products
      </a>
      <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
        <li><a class="dropdown-item" href="#stratocaster">Stratocaster</a></li>
        <li><a class="dropdown-item" href="#lespaul">Les Paul</a></li>
        <li><a class="dropdown-item" href="#telecaster">Telecaster</a></li>
        <li><a class="dropdown-item" href="#jazzmaster">Jazzmaster</a></li>
        <li><a class="dropdown-item" href="#jaguar">Jaguar</a></li>
        <li><a class="dropdown-item" href="#sg">SG</a></li>
        <li><a class="dropdown-item" href="#flyingv">Flying V</a></li>
        <li><a class="dropdown-item" href="#firebird">Firebird</a></li>
      </ul>
    </li>
  </ul>
</div>
</div>
</nav>

```

2. Loader (2):

```

<!-- Loader Component -->
<div id="loader">
  <h1>HI, THANK YOU FOR REACHING US</h1>
  <h1>ENJOY PURCHASING</h1>
  <h1>HAPPY SHOPPING</h1>
</div>

```

3. Swiper Slider Component:

```

<!-- Swiper Slider Component -->
<div class="swiper">
  <div class="swiper-wrapper">
    <div class="swiper-slide">
      
    </div>
    <div class="swiper-slide">
      
    </div>
  <div class="swipe<!-- Swiper Slider Component -->
<div class="swiper">

```

```

<div class="swiper-wrapper">
  <div class="swiper-slide">
    
  </div>
  <div class="swiper-slide">
    
  </div>
  <div class="swiper-slide">
    
  </div>
</div>
<div class="swiper-pagination"></div>
<div class="swiper-button-prev"></div>
<div class="swiper-button-next"></div>
<div class="swiper-scrollbar"></div>
</div>r-slide">
  
</div>
</div>
<div class="swiper-pagination"></div>
<div class="swiper-button-prev"></div>
<div class="swiper-button-next"></div>
<div class="swiper-scrollbar"></div>
</div>

```

4. Product Card:

```

<!-- Product Card Component -->
<div class="col-md-3 mb-4">
  <div class="product-card">
    
    <div class="card-body">
      <h5>Product Name</h5>
      <p>$Price</p>
      <button class="btn-buy">Buy Now</button>
    </div>
  </div>
</div>
</div>

```

5. Footer:

```

<!-- Footer Component -->
<footer class="text-center py-4">
  <p>&copy; 2024 Tim's Brand. All rights reserved.</p>
</footer>

```


3. CSS components

3.1 Navigation Bar

```
.navbar {  
  background: #1a1b1c;  
  position: fixed;  
  top: 0;  
  width: 100%;  
  z-index: 9999;  
  border: none;  
  box-shadow: none;  
  padding: 0;  
  height: 120px;  
  animation: slideIn 1s ease-out;  
  transition: transform 0.3s ease, box-shadow 0.3s ease;  
}  
  
.navbar .container-fluid {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  height: 100%;  
}  
  
.navbar-brand {  
  font-family: 'Century Schoolbook', serif;  
  font-size: 2.5em;  
  background: linear-gradient(to right, #ffa31d, #ef5454);  
  -webkit-background-clip: text;  
  -webkit-text-fill-color: white;  
  text-shadow: 0 0 10px rgba(255, 255, 255, 0.5), 0 0 20px rgba(255, 255, 255, 0.5), 0 0 30px  
  rgba(255, 255, 255, 0.5), 0 0 40px rgba(255, 255, 255, 0.5);  
  white-space: nowrap;  
  position: relative;  
  animation: glow-animation 1.5s infinite alternate;  
}  
  
.navbar-nav {  
  display: flex;  
  gap: 40px;  
}  
  
.navbar-nav .nav-item {  
  position: relative;  
  transition: transform 0.3s ease;  
}  
  
.navbar-nav .nav-item:not(:last-child)::after {  
  content: '|';
```

```

    color: white;
    position: absolute;
    right: -20px;
    top: 50%;
    transform: translateY(-50%);
    font-size: 2.5em;
}

.navbar-nav .nav-item .nav-link {
    background: linear-gradient(to right, #ffa31d, #ef5454);
    -webkit-background-clip: text;
    -webkit-text-fill-color: transparent;
    font-family: 'Century Schoolbook', serif;
    font-size: 1.6em;
    padding: 15px;
    white-space: nowrap;
    text-decoration: none;
    animation: glow-animation 1.5s infinite alternate;
    transition: color 0.3s ease, transform 0.3s ease;
}

@keyframes glow-animation {
    0% {
        text-shadow: 0 0 5px rgba(255, 163, 29, 0.4), 0 0 10px rgba(255, 163, 29, 0.3), 0 0 15px
        rgba(239, 84, 84, 0.2), 0 0 20px rgba(239, 84, 84, 0.1);
    }
    100% {
        text-shadow: 0 0 10px rgba(255, 163, 29, 0.7), 0 0 20px rgba(255, 163, 29, 0.6), 0 0 30px
        rgba(239, 84, 84, 0.5), 0 0 40px rgba(239, 84, 84, 0.4);
    }
}

.navbar-nav .nav-item .nav-link.active {
    background: linear-gradient(to right, #ffa31d, #ef5454);
    -webkit-background-clip: text;
    -webkit-text-fill-color: transparent;
}

.navbar-nav .nav-item .nav-link:hover {
    color: #ffffff;
    transform: scale(0.95);
}

.navbar:hover {
    transform: translateY(10px);
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.3);
}

.nav-link:hover {
    color: #ffa31d;
    text-decoration: underline;}

```

3.2 Loader



HI, THANK YOU FOR REACHING US

```
/* Loader Component */
#loader {
  position: fixed;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  background: rgba(255, 255, 255, 0.8);
  z-index: 9999;
}
#loader h1 {
  font-size: 2rem;
  text-align: center;
} /* Loader Component */
#loader {
  position: fixed;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  background: rgba(255, 255, 255, 0.8);
  z-index: 9999;
}
#loader h1 {
  font-size: 2rem;
  text-align: center;
```

3.3 Swiper



/* Swiper Styles */

```
.swiper {  
  width: 100%;  
  height: 100vh;  
  position: relative;  
  background-color: #1a1b1c;  
}  
  
.swiper-slide {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  background-color: #1a1b1c;  
}  
  
.swiper-slide img {  
  width: 100%;  
  height: auto;  
  object-fit: cover;  
  border: none;  
}  
  
.swiper-button-prev,  
.swiper-button-next {  
  color: #fff;  
  width: 40px;  
  height: 40px;  
  background-color: rgba(0, 0, 0, 0.5);  
  border-radius: 50%;  
}  
  
.swiper-button-prev::after,  
.swiper-button-next::after {  
  font-size: 20px;
```

```

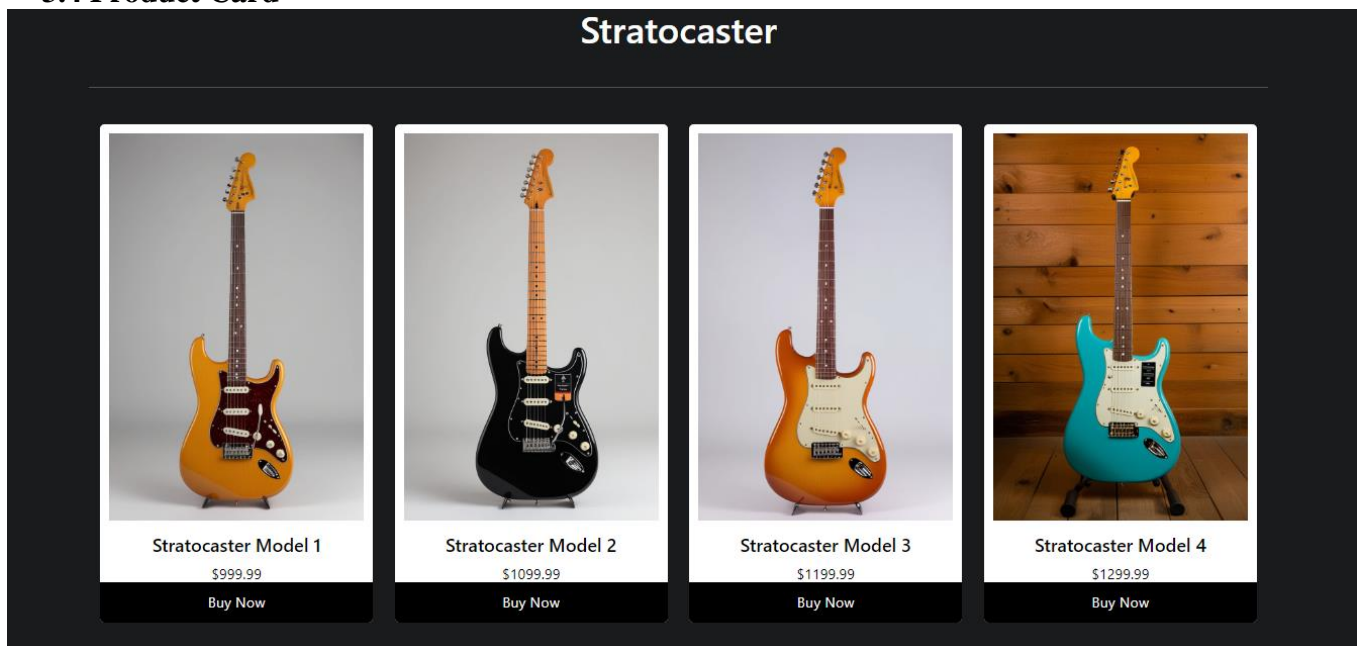
}

.swiper-pagination-bullet {
  background-color: #fff;
}

.swiper-scrollbar {
  background-color: rgba(0, 0, 0, 0.5);
}

```

3.4 Product Card



```

.section {
  padding: 40px 0;
  background-color: #1a1b1c;
  color: #fff;
}

.section h2 {
  text-align: center;
}

```

```

    margin-bottom: 20px;
    font-size: 2.5em;
}

.section .row {
    margin: 0;
}

.section img {
    width: 100%;
    height: auto;
    object-fit: cover;
}

.section hr {
    border: none;
    border-top: 2px solid #fff;
    margin: 40px 0;
}

.product-card {
    position: relative;
    overflow: hidden;
    border-radius: 5px;
    background-color: #fff;
    color: #000;
    margin-bottom: 20px;
    padding: 10px;
    transition: transform 0.3s ease, box-shadow 0.3s ease;
}

.product-card:hover {
    transform: scale(1.05);
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
}

.product-card img {
    width: 100%;
    height: auto;
    object-fit: cover;
}

.product-card .card-body {
    padding: 15px;
    text-align: center;
    padding-bottom: 60px;
}

.product-card .btn-buy {
    background-color: #000;
    color: #fff;

```

```
border: none;
border-radius: 0;
padding: 10px 0;
cursor: pointer;
position: absolute;
bottom: 0;
left: 0;
right: 0;
}
```

3.5 Footer

It doesn't need any cs html is enough

3. Inter-VLAN Routing

3.4 Layer 3 switching using SVIs [1]

L3_SW1

enable

configure terminal

interface vlan 10

ip address 10.10.10.1 255.255.255.0

no shutdown

exit

interface vlan 20

ip address 20.20.20.1 255.255.255.0

no shutdown

exit

interface vlan 30

ip address 30.30.30.1 255.255.255.0

no shutdown

exit

interface vlan 40

ip address 40.40.40.1 255.255.255.0

no shutdown

exit

3.5 Subnetting

Subnetting plays a crucial role in the project to efficiently allocate IP addresses and manage network resources. The base network address of 192.168.0.0/22 is subnetted to accommodate different departments. For example, VLAN 10 might use the subnet 192.168.10.0/24, while VLAN 20 could use 192.168.20.0/24. Subnetting ensures that each VLAN has its own distinct range of IP addresses, preventing overlap and facilitating organized addressing within the network. This approach enhances security, simplifies network management, and supports future scalability by providing a structured allocation of IP resources to individual VLANs.

4. Java Scrit

4.4 Script for loader

```
document.addEventListener('DOMContentLoaded', () => {
  const loader = document.getElementById('loader');

  // Ensure loader is visible when the page loads
  loader.classList.remove('hidden');

  // Simulate a loading period (adjust time as needed)
  setTimeout(() => {
    // Hide the loader after the specified time
    loader.classList.add('hidden');
  }, 3000); // Adjust this time to fit your needs
});
```

4.5 Script for slider

```
document.addEventListener('DOMContentLoaded', function() {
  // Initialize Swiper
  var swiper = new Swiper('.swiper-container', {
    loop: true, // Loop slides
    autoplay: {
      delay: 5000, // Change slide every 5 seconds
      disableOnInteraction: false, // Continue autoplay after interactions
    },
    pagination: {
      el: '.swiper-pagination',
      clickable: true, // Pagination clickable
    },
    navigation: {
      nextEl: '.swiper-button-next', // Next button
      prevEl: '.swiper-button-prev', // Previous button
    },
  });
});
```

5. Responsiveness

Responsive Layout CSS

Description: This file contains media queries that adjust the layout of the main container and columns for different screen sizes. It ensures that the website layout remains fluid and visually appealing across various devices.

- **@media (max-width: 1200px):** The container width is set to 95% to provide more space on smaller screens.
- **@media (max-width: 992px):** Each column takes up 50% width, making two columns per row on medium-sized screens.
- **@media (max-width: 768px):** Each column takes up 100% width, stacking them vertically for small screens.

File: styles/responsive-layout.css

```
css
/* Responsive Layout */
@media (max-width: 1200px) {
  .container {
    width: 95%;
  }
}

@media (max-width: 992px) {
  .col-md-3 {
    width: 50%;
  }
}

@media (max-width: 768px) {
  .col-md-3 {
    width: 100%;
  }
}
```

2. Responsive Header CSS

Description: This file adjusts the font sizes of the navigation bar elements to ensure readability and usability on smaller screens.

- **@media (max-width: 768px):** The font size of the navbar brand and navigation links are reduced to fit better on mobile screens

File: styles/responsive-header.css

```
css

/* Responsive Header */
@media (max-width: 768px) {
  .navbar-brand {
```

```

        font-size: 1.2rem;
    }
    .nav-link {
        font-size: 0.9rem;
    }
}

```

4. Responsive Loader CSS

Description: This file modifies the size of the text in the loader component to make sure it remains readable on smaller screens.

- **@media (max-width: 768px):** The font size of the loader text is decreased for better fit and readability on mobile devices.

File: styles/responsive-loader.css

```

css
/* Responsive Loader */
@media (max-width: 768px) {
    #loader h1 {
        font-size: 1.5rem;
    }
}

```

5. Responsive Swiper CSS

Description: This file ensures that the images in the swiper slider component resize appropriately for different screen sizes.

- **@media (max-width: 768px):** The images within the swiper slides are set to 100% width and automatic height, making them scale correctly on smaller screens.

File: styles/responsive-swiper.css

```

css
/* Responsive Swiper */
@media (max-width: 768px) {
    .swiper-slide img {
        width: 100%;
        height: auto;
    }
}

```

6. Responsive Product Card CSS

Description: This file contains media queries to adjust the appearance of product cards, including margins, padding, and button sizes for smaller screens.

- **@media (max-width: 992px):** Adjusts the margin bottom, padding of the card body, and the button padding for medium-sized screens.
- **@media (max-width: 768px):** Ensures the product images and card body padding scale appropriately, and the button size is reduced for small screens.

File: styles/responsive-product-card.css

```
css
/* Responsive Product Card */
@media (max-width: 992px) {
  .product-card {
    margin-bottom: 30px;
  }
  .card-body {
    padding: 10px;
  }
  .btn-buy {
    padding: 8px 16px;
  }
}

@media (max-width: 768px) {
  .product-card img {
    width: 100%;
    height: auto;
  }
  .card-body {
    padding: 5px;
  }
  .btn-buy {
    padding: 6px 12px;
  }
}
```

7. Responsive Section CSS

Description: This file modifies the headings and horizontal rules within sections to maintain a consistent look and feel on smaller screens.

- **@media (max-width: 768px):** The font size of section headings is reduced, and the width of the horizontal rule is decreased for better visual hierarchy on mobile devices.

File: styles/responsive-section.css

```
css
/* Responsive Section */
@media (max-width: 768px) {
  .section h2 {
    font-size: 1.5rem;
  }
  .section hr {
    width: 30px;
  }
}
```

8. Responsive Footer CSS

Description: This file ensures that the footer text scales down appropriately for smaller screens to maintain readability and aesthetic appeal.

- **@media (max-width: 768px):** The font size of the footer text is decreased to fit better on mobile screens.

File: styles/responsive-footer.css

```
css
/* Responsive Footer */
@media (max-width: 768px) {
  footer p {
    font-size: 0.8rem;
  }
}
```

6. CSS code with Bootstrap for better performance

6.4 Simulation of overall

At first it run the loader

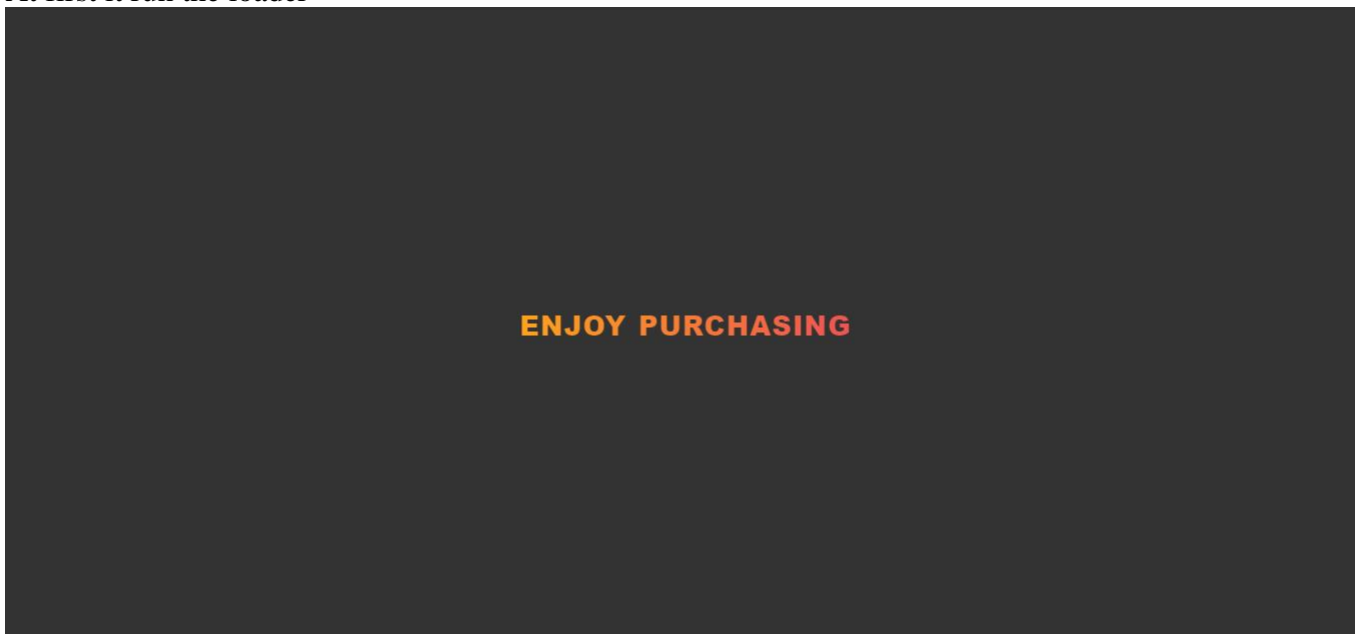
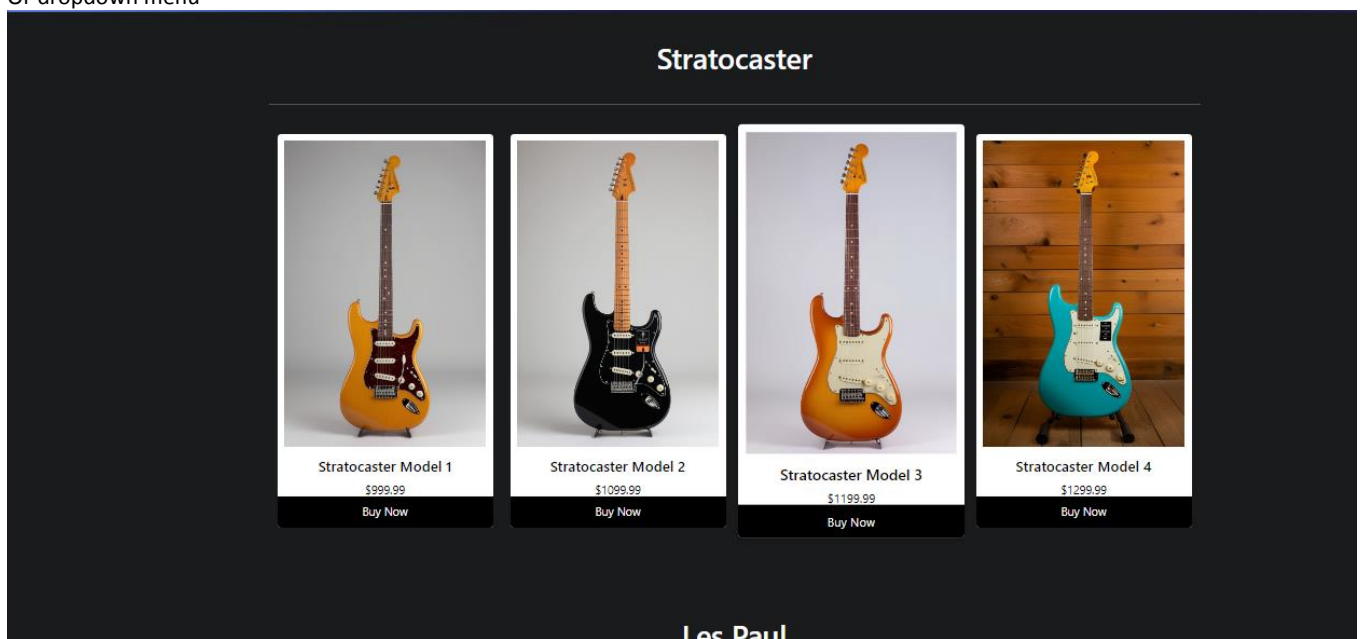


Figure 2: ICMP PDU check

Then it shows up nav



The below the slider there are different products which can be used by clicking on specific product hamburger bar Or dropdown menu



Then it come to footer

- **Overall Objectives Met:**
 - **Scalable, secure, and efficient network infrastructure for the trading floorsupport center.**

7. Conclusion

7.4 Summary

This project involves developing a dynamic and visually appealing web interface using HTML and CSS. The design is structured into various reusable components, including a responsive header with navigation, a loading screen, a Swiper slider for showcasing images, product cards for displaying items, and a footer. Each component is styled to ensure consistency and functionality across different sections of the website. The use of CSS features such as Flexbox and Grid layout ensures a modern and flexible design, while custom styles enhance the visual appeal and user experience. Additionally, responsive design principles are applied to ensure that the interface adapts seamlessly to various screen sizes, providing an optimal viewing experience on both desktop and mobile devices. This project highlights the importance of clean, modular code and effective design strategies in creating a professional and user-friendly web application.

7.5 Lessons Learned

Through this project, I gained valuable insights into web development and design. I learned how to break down complex HTML structures into manageable, reusable components, which streamlined the development process and improved code maintainability. Working with CSS taught me the importance of organized styling and how to apply responsive design principles to ensure a consistent user experience across devices. Additionally, integrating JavaScript for interactive elements like sliders enhanced my understanding of dynamic web functionalities. Overall, the project reinforced the importance of attention to detail in design and the need for a user-centric approach in web development.

8. Future Work

8.4 Potential Improvements

- Compress images to speed up loading times
- Use advanced responsive design for better mobile experience
- Add accessibility features to support all users
- Ensure compatibility across different browsers
- Include interactive elements to make the site more engaging
- Improve the navigation for a smoother user experience
- Optimize for search engines to increase visibility
- Enhance performance with techniques like code minification
- Offer options for users to personalize their experience

- **Ongoing Training and Skill Development**
 - **Regular Security Audits**
-
- **Energy Efficiency Measures**

9. References

- ❏ Bootstrap, 2024. *Bootstrap Documentation*. Available at: <https://getbootstrap.com/docs/5.3/getting-started/introduction/> [Accessed 15 August 2024].
- ❏ Swiper, 2024. *Swiper API Documentation*. Available at: <https://swiperjs.com/api/> [Accessed 15 August 2024].
- ❏ MDN Web Docs, 2024. *CSS Reference*. Available at: <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference> [Accessed 15 August 2024].
- ❏ W3Schools, 2024. *HTML Tutorial*. Available at: <https://www.w3schools.com/html/> [Accessed 15 August 2024].
- ❏ Font Awesome, 2024. *Font Awesome Documentation*. Available at: <https://fontawesome.com/docs> [Accessed 15 August 2024].

10. Appendices

Abbreviations:

- **Header Component**
- **Loader Component**
- **Swiper Slider Component**