Mini PROJECT REPORT

On

"Fashionova-An E-commerce Website"

Submitted by

KRISHNA SHARMA
(2115000554)
ROHIT SINGH
(2115000877)
KRISHNA KUMAR SINGH
(2115000550)
GARVIT Gupta
(2115000398)

Department of Computer Engineering & Applications

Institute of Engineering & Technology

GLA University
Mathura- 281406, INDIA

2023-2024

Department of computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

Declaration

We hereby declare that the work which is being presented in the Mini Project "**Fashionova",** in partial fullfilment of the requirements for Mini Project viva voice, is an authentic record of our own work carried by the team members under the supervision of our mentor Mrs.Ruchi Gupta.

Group Members:

1. KRISHNA SHARMA (2115000554)
2. ROHIT SINGH(2115000877)
3. KRISHNA KUMAR SINGH(2115000550)
4. GARVIT GUPTA(2115000398)

Course: B.Tech (Computer Science and Engineering)

Year: 3rd

Semester: 5th

Supervised By: MRS.RUCHI GUPTA

Department of computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,
Mathura – 281406

Certificate

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

_____

Supervisor

MRS.RUCHI GUPTA

_____ _____

Project Coordinator Program Coordinator

(Mr. Surendra Singh) (Mr. Rakesh Kumar Galav)

Motivation

Development is a method of using technology to bring people and ideas together despite of the geographical barriers. The technology has been available for years but the acceptance was quite recent. Our project "Fashionova" is basically an e-commerce website which focuses on providing it's users fashionable costumes based on their interests and it is also divided into seasonal fashion raiment with affordable prices and best qualities.

Requirements

a). Software Requirements:

- Languages/Technologies Used: HTML, CSS, JavaScript
- IDE Used: Visual Studio Code
- Web Browser: Google Chrome
- Supported System: Web
- GitHub: GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. GitHub Repository: A GitHub repository can be used to store a development project. It can contain folders and any type of files (HTML, CSS, JavaScript, Documents, Data, Images). A GitHub repository should also include a license file and a README file about the project. A GitHub repository can also be used to store ideas, or any resources that you want to share.
- Visual Studio Code: Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. Microsoft has released Visual Studio Code's source code on the VS Code repository of GitHub.com, under the permissive MIT License, while
  the compiled binaries are freeware.

b). Hardware Requirements:

- Processor Required: Intel i7
- Operating System: Windows 11
- RAM: 16GB
- Hardware Devices: Computer System
- Hard Disk: 1TB

Acknowledgement

It gives us a great sense of pleasure to present the synopsis of the B.Tech MiniProjectI (Fashinova) undertaken during B.Tech3rd Year. This project itself is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe specistyal debt of gratitude to Mr. Sachin Barpanda, (Mentor, Coding Blocks), for providing us with an encouraging platform to develop this project, who thus helped us in shaping our abilities towards a constructive goal and for her support and guidance to our work. Her sincerity, thoroughness and perseverance is been a constant source of inspiration for us. We believe that she will shower us with all her extensively experienced ideas a insightful comments at different stages of the project & also taught us about the industry-oriented technologies. We also do not like to miss the opportunity to acknowledge the contribution of faculty members of the department for their kind guidance and cooperation.

Fashionova

Abstract

In this project, we are creating an FullStack website, basically an e-commerce website which is a one-stop virtual fashion store where the customers gets what they need. Also, it provides it's customers with frequent sales and discounts on festives and end of the month sales with trending variegation in clothing industry. Easy to return policy with flexible time duration (conditions apply).We believe in quality over quantity thus we present the finest quality of clothes to our customers. Also the cancellation policy of our website is quite quick and hassle free encouraging our clients an easy way to experience an amazing shopping at our virtual store.

Contents

Acknowledgment…….....06

Abstract……..……07

1. Introduction:

Introduction to HTML.....…….....11

Introduction to CSS……...…….…14

Introduction to JavaScript……..….17

2. Screen shot of front end……...20

Chapter1

Introduction

An e-commerce website is a digital platform that facilitates online transactions of goods and services. It allows businesses to showcase their products or services, and consumers can browse, select, and purchase items using electronic payment methods. These websites have become a crucial component of the modern economy, providing a convenient and efficient way for businesses to reach a global audience and for consumers to shop from the comfort of their homes.

Here's an introduction that covers various aspects of a full e-commerce website:

Title: Unleashing the Digital Marketplace: An Introduction to Full E-commerce Websites

In the fast-paced digital era, e-commerce has emerged as a transformative force, reshaping the way businesses operate and consumers shop. A full e-commerce website is a comprehensive online platform that serves as the digital storefront for businesses, offering a dynamic and interactive space for buyers and sellers to connect seamlessly.

1. Storefront and Product Showcase: A central element of any e-commerce website is the storefront, where businesses can display their products or services. Through visually appealing design and intuitive navigation, these websites aim to create a compelling online shopping experience. High-quality images, detailed product descriptions, and user-friendly interfaces contribute to engaging potential customers.

2. Shopping Cart and Checkout Process: The shopping cart functionality is the digital counterpart to the traditional shopping basket. Users can add items to their cart as they browse the website, and the checkout process allows for a smooth and secure transaction. E-commerce websites integrate various payment gateways, providing flexibility for customers to choose from credit cards, digital wallets, and other electronic payment methods.

3. Inventory Management: Behind the scenes, e-commerce websites employ robust inventory management systems to keep track of available products, update stock levels in real-time, and notify both customers and administrators about product availability. This ensures a seamless and efficient supply chain.

4. User Accounts and Personalization: To enhance the shopping experience, e-commerce websites often incorporate user account features. Customers can create accounts to track their order history, save favorite items, and receive personalized recommendations based on their preferences and purchase history.

5. Security and Trust: Security is paramount in the world of e-commerce. Full e-commerce websites implement secure sockets layer (SSL) encryption to protect sensitive information during transactions. Additionally, trust signals such as customer reviews, secure payment badges, and transparent policies contribute to building trust among online shoppers.

6. Mobile Responsiveness: Recognizing the prevalence of mobile devices, full e-commerce websites prioritize mobile responsiveness. This ensures that users can seamlessly browse and make purchases on various devices, including smartphones and tablets.

7. Marketing and Analytics: E-commerce websites leverage digital marketing strategies such as search engine optimization (SEO), social media integration, and email campaigns to drive traffic and increase visibility. Analytics tools provide valuable insights into customer behaviour, allowing businesses to refine their strategies for better performance.

In conclusion, a full e-commerce website is not just a virtual storefront; it's a dynamic and multifaceted platform that brings together the convenience of online shopping, the efficiency of digital transactions, and the expansive reach of the internet. As businesses continue to embrace the digital landscape, e-commerce websites play a pivotal role in shaping the future of retail and commerce.

Chapter 2

Technologies Used

HTML

HTML (Hypertext Markup Language): A Comprehensive Overview

HTML, or Hypertext Markup Language, is the fundamental language of the web. Developed by Sir Tim Berners-Lee in the early 1990s, HTML serves as the backbone of web pages, enabling the structure and presentation of content on the World Wide Web. Below is a comprehensive overview of HTML, covering its syntax, elements, and key concepts.

1. Syntax:

HTML uses a markup language composed of tags and attributes to structure content. Tags are enclosed in angle brackets, and most come in pairs, with an opening tag and a closing tag. The content is placed between these tags. The basic syntax looks like this:

```
html
<tagname>Content goes here</tagname>
```

2. Document Structure:

An HTML document typically consists of several key elements:

- `<!DOCTYPE html>`: Declares the document type and version of HTML.
- `<html>`: The root element of an HTML page.
- `<head>`: Contains metadata such as title, character set, and linked stylesheets.
- `<title>`: Sets the title of the HTML document (appears in the browser tab).
- `<body>`: Contains the content of the HTML document.

3. Common HTML Elements:

- **Headings:**`<h1>` to `<h6>` for different heading levels.
- **Paragraphs:**`<p>` for defining paragraphs.
- **Lists:**`<ul>` for unordered lists, `<ol>` for ordered lists, `<li>` for list items.
- **Links:**`<a>` for creating hyperlinks.
- **Images:**`<img>` for embedding images.
- **Forms:**`<form>`, `<input>`, `<button>`, etc., for creating interactive forms.

4. Attributes:

HTML tags can have attributes that provide additional information about the element. Attributes are placed within the opening tag and typically come in name/value pairs. For example:

```
html
<ahref="https://www.example.com">Visit Example.com</a>
```

5. Semantic HTML:

Semantic HTML tags provide meaning to the content, aiding in accessibility and search engine optimization. Examples include `<header>`, `<nav>`, `<article>`, `<section>`, `<footer>`, etc.

6. HTML5:

HTML5, the latest version of HTML, introduced new elements and APIs for multimedia, form validation, local storage, and more. Some HTML5 features include `<video>`, `<audio>`, `<canvas>`, and `<localStorage>`.

7. Responsive Web Design:

HTML is crucial for building responsive websites. Media queries and flexible grid layouts help create designs that adapt to various screen sizes, ensuring a positive user experience across devices.

8. Validation:

W3C provides a Markup Validation Service that allows developers to check whether their HTML code follows the official specifications. Valid HTML ensures consistent rendering across different browsers.

9. Frameworks and Templates:

Many web development frameworks and templates leverage HTML as their core language. Popular examples include Bootstrap, Foundation, and various content management systems (CMS) like WordPress.

10. Best Practices:

- Use proper indentation and whitespace for code readability.
- Employ semantic HTML for improved accessibility and SEO.
- Separate content from presentation by using CSS for styling.
- Regularly validate HTML code to ensure compliance with standards.

HTML is the cornerstone of web development, providing the structure and foundation for creating interactive and visually appealing websites. Combined with CSS (Cascading Style Sheets) and JavaScript, HTML forms a powerful trio that empowers developers to craft engaging and dynamic web experiences.

CSS

Cascading Style Sheets (CSS): A Comprehensive Overview

CSS, or Cascading Style Sheets, is a styling language used in web development to control the presentation and layout of HTML documents. CSS allows developers to define how the elements of a web page should appear, making it a crucial technology for creating visually appealing and responsive websites. Here's a comprehensive overview of CSS, covering its syntax, selectors, properties, and key concepts.

1. Syntax:

CSS consists of rules that define the styling of HTML elements. Each rule consists of a selector and a declaration block. The basic syntax looks like this:

```
css
selector {
  property: value;
}
```

2. Selectors:

Selectors target HTML elements to apply styles. There are various types of selectors, including:

- **Element Selector:** Targets HTML elements by their name (e.g., `p` for paragraphs).
- **Class Selector:** Targets elements with a specific class (e.g., `.my-class`).
- **ID Selector:** Targets a specific element with a unique ID (e.g., `#my-id`).
- **Attribute Selector:** Targets elements based on their attributes (e.g., `input[type="text"]`).

3. Properties and Values:

CSS properties define the aspects of an element to be styled, and values determine the specific styling. Some common properties include:

- **Color:**`color: red;`
- **Font:**`font-family: Arial, sans-serif;`
- **Margin:**`margin: 10px;`
- **Padding:**`padding: 5px;`
- **Border:**`border: 1px solid #000;`
- **Background:**`background-color: #f0f0f0;`

4. Box Model:

The CSS box model describes the layout of elements on a page. It includes content, padding, border, and margin. Understanding the box model is essential for creating well-structured layouts.

5. Layout:

CSS provides mechanisms for creating responsive and flexible layouts. Techniques include:

- **Flexbox:** A one-dimensional layout model for creating flexible and efficient container layouts.
- **Grid:** A two-dimensional layout system for creating complex, grid-based designs.

6. Media Queries:

Media queries allow developers to apply different styles based on characteristics such as screen size, device orientation, or resolution. This is crucial for responsive web design.

7. Transitions and Animations:

CSS enables the creation of smooth transitions and animations, enhancing the user experience. Properties like `transition` and `animation` facilitate these effects.

8. Selectors and Specificity:

CSS follows a specificity hierarchy to determine which styles take precedence when conflicting rules are present. Understanding selector specificity is crucial for writing maintainable and predictable stylesheets.

9. Preprocessors:

CSS preprocessors like Sass and LESS extend the capabilities of CSS by introducing variables, and functions. These tools enhance code maintainability and organization.

10. CSS Frameworks:

Frameworks like Bootstrap and Foundation provide pre-designed CSS components and layouts, allowing developers to build responsive websites more efficiently.

11. Browser Compatibility:

While CSS is designed to be cross-browser compatible, differences in rendering engines may lead to inconsistencies. Testing and using vendor prefixes can help address compatibility issues.

12. CSS-in-JS:

In modern web development, CSS-in-JS solutions, like styled-components in React, allow developers to write CSS directly within JavaScript components, enhancing component encapsulation and styling modularity.

13. Best Practices:

- Use external stylesheets for better maintainability.
- Practice modular and scalable CSS architecture.
- Optimize styles for performance, considering page load times.
- Regularly test and debug for cross-browser compatibility.

CSS is a foundational technology that, when combined with HTML and JavaScript, empowers developers to create engaging and visually appealing web experiences. As web standards evolve, staying updated with the latest features and best practices is crucial for effective and efficient styling.Top of Form

Java-Script

JavaScript: A Comprehensive Overview

JavaScript is a versatile, high-level programming language that is primarily used to create dynamic and interactive content on websites. Developed by Brendan Eich in 1995, JavaScript has evolved into one of the most widely used programming languages, not only for web development but also for server-side development (Node.js), mobile app development, and more. Here's a comprehensive overview of JavaScript, covering its syntax, features, and key concepts.

1. Syntax:

JavaScript syntax is similar to other programming languages like C and Java. It uses variables, data types, operators, control structures, and functions. The basic syntax includes:

```
javascript
// Variables and data types
let x = 5; // Variable declaration
constPI = 3.14; // Constant declaration
let message = "Hello, World!"; // String variable

// Functions
functiongreet(name) {
return"Hello, " + name + "!";
}

// Control structures
if (x >0) {
console.log("Positive number");
} else {
console.log("Non-positive number");}
```

2. Data Types:

JavaScript has dynamic typing, meaning you don't need to explicitly declare a variable's type. Common data types include numbers, strings, booleans, objects, arrays, and functions.

3. Functions:

Functions in JavaScript are first-class citizens, which means they can be assigned to variables, passed as arguments, and returned from other functions. Anonymous functions (lambdas) and arrow functions are commonly used.

4. Objects:

Objects in JavaScript are key-value pairs, and they are fundamental to the language. The Document Object Model (DOM) allows interaction with HTML and XML documents, enabling dynamic manipulation of web page content.

5. Arrays:

Arrays are used to store and manipulate collections of data. They are versatile and come with a variety of built-in methods for manipulation and iteration.

6. Events:

JavaScript enables the creation of interactive web pages by responding to user actions (events). Event handlers can be used to trigger functions when events like clicks, keypresses, or form submissions occur.

7. Asynchronous Programming:

JavaScript supports asynchronous programming through mechanisms like callbacks, promises, and async/await. This is crucial for handling operations such as fetching data from servers without blocking the main thread.

8. DOM Manipulation:

The Document Object Model (DOM) represents the structure of HTML documents as a tree of objects. JavaScript allows developers to manipulate the DOM, enabling dynamic updates and interactions on web pages.

9. AJAX (Asynchronous JavaScript and XML):

AJAX allows for the asynchronous exchange of data between the browser and the server, facilitating dynamic updates without requiring a full page reload.

10. JSON (JavaScript Object Notation):

JSON is a lightweight data interchange format widely used in JavaScript for data serialization. It is easy to read and write, making it a popular choice for APIs.

11. ES6 and Beyond:

ECMAScript 6 (ES6), released in 2015, introduced significant enhancements to JavaScript, including arrow functions, template literals, destructuring assignment, and the `let` and `const` keywords. Subsequent ECMAScript versions have continued to bring new features and improvements.

12. Modules:

JavaScript now supports modules, allowing developers to organize code into reusable and maintainable pieces. This is particularly important for large-scale applications.

13. Frameworks and Libraries:

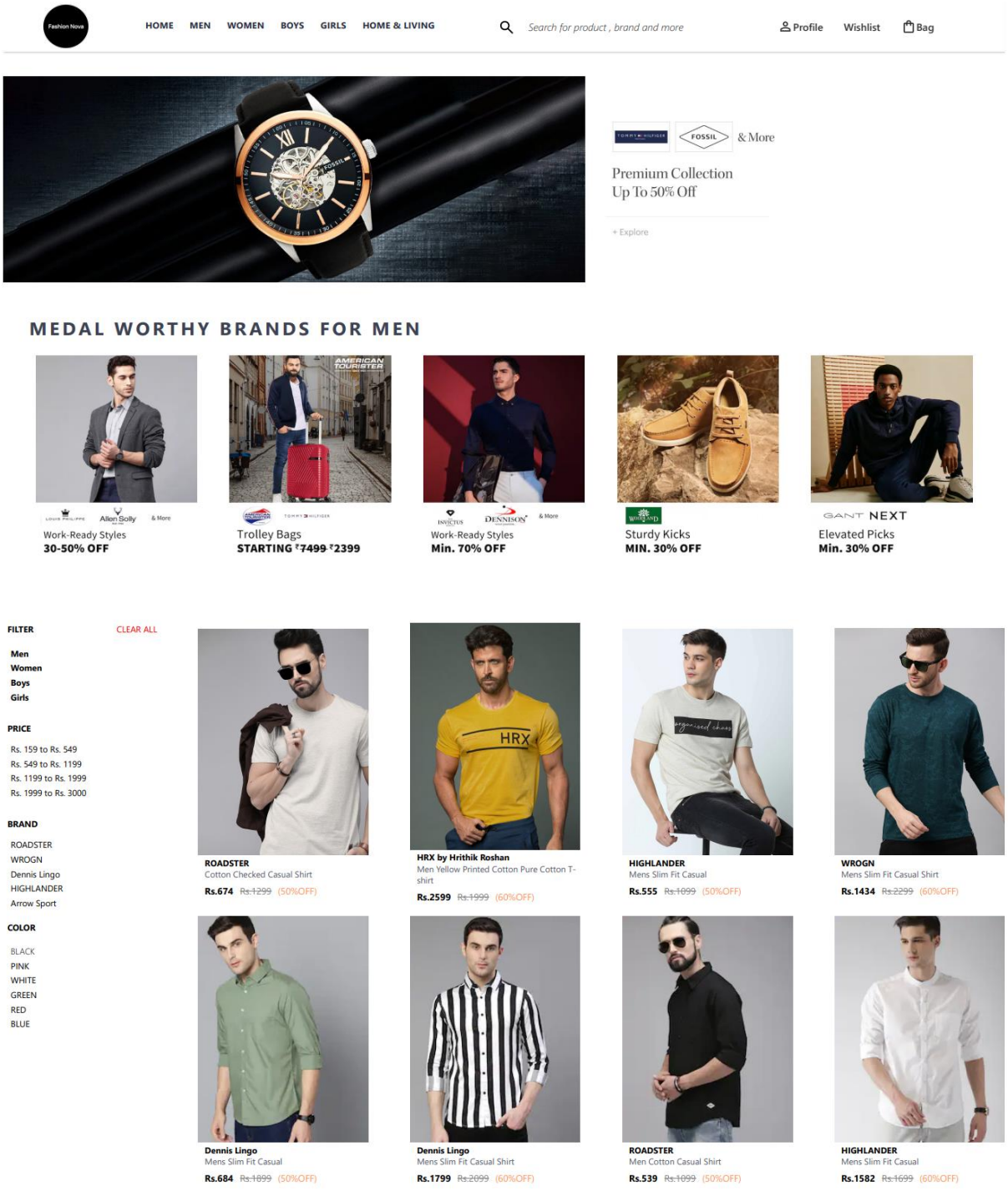
Numerous frameworks and libraries have been built on top of JavaScript to simplify and accelerate web development, such as React, Angular, Vue.js, and jQuery.

15. Best Practices:

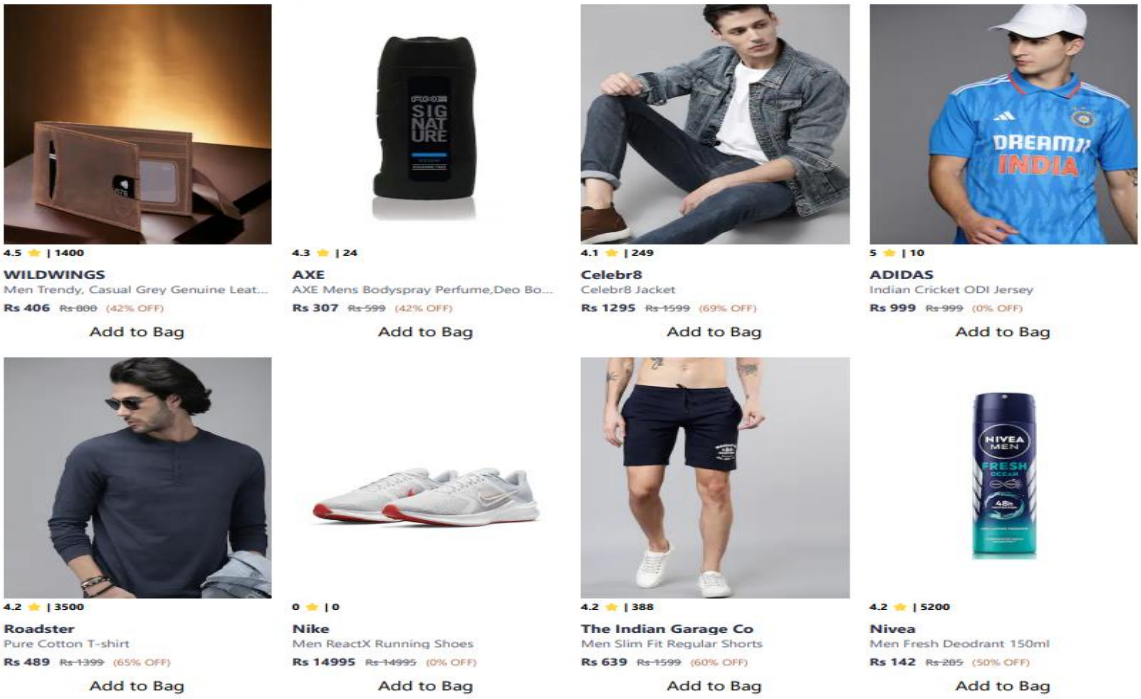
- Use meaningful variable and function names.
- Embrace modular and reusable code.
- Pay attention to scope and avoid global variables.
- Keep code DRY (Don't Repeat Yourself) to enhance maintainability.

JavaScript is a cornerstone of modern web development, enabling the creation of dynamic, interactive, and responsive web applications. Its versatility extends beyond the browser, making it a key technology for various application domains. As JavaScript continues to evolve, staying abreast of the latest language features and best practices is essential for developers.

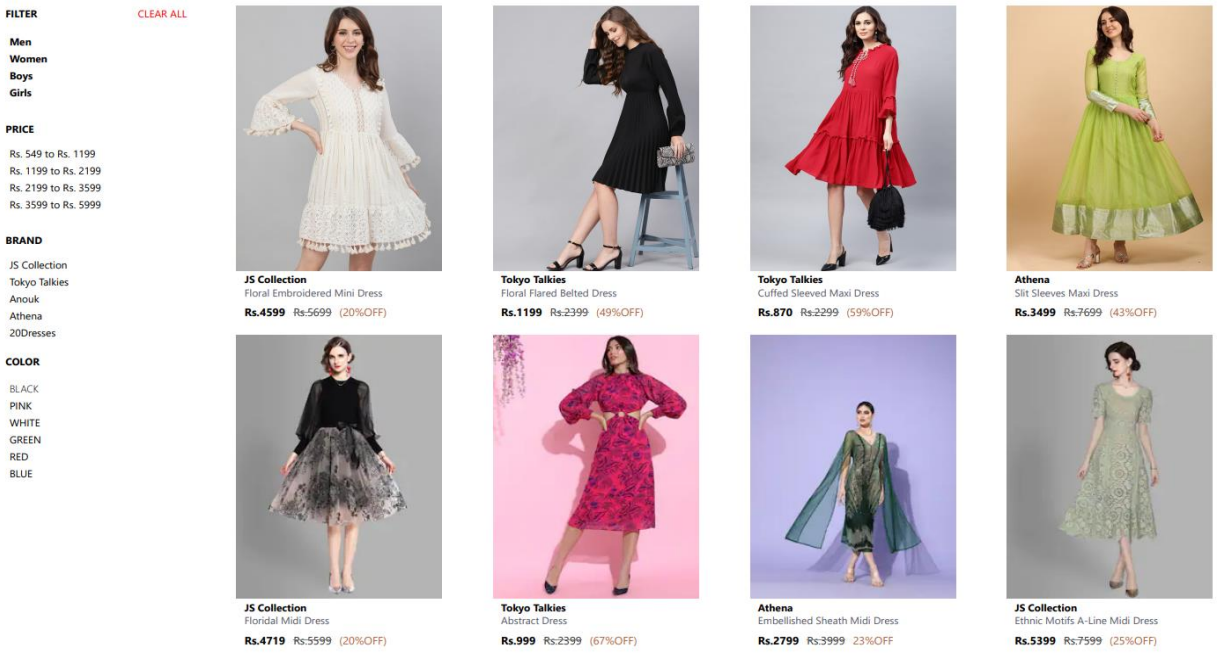
Screen-Shot of the projects



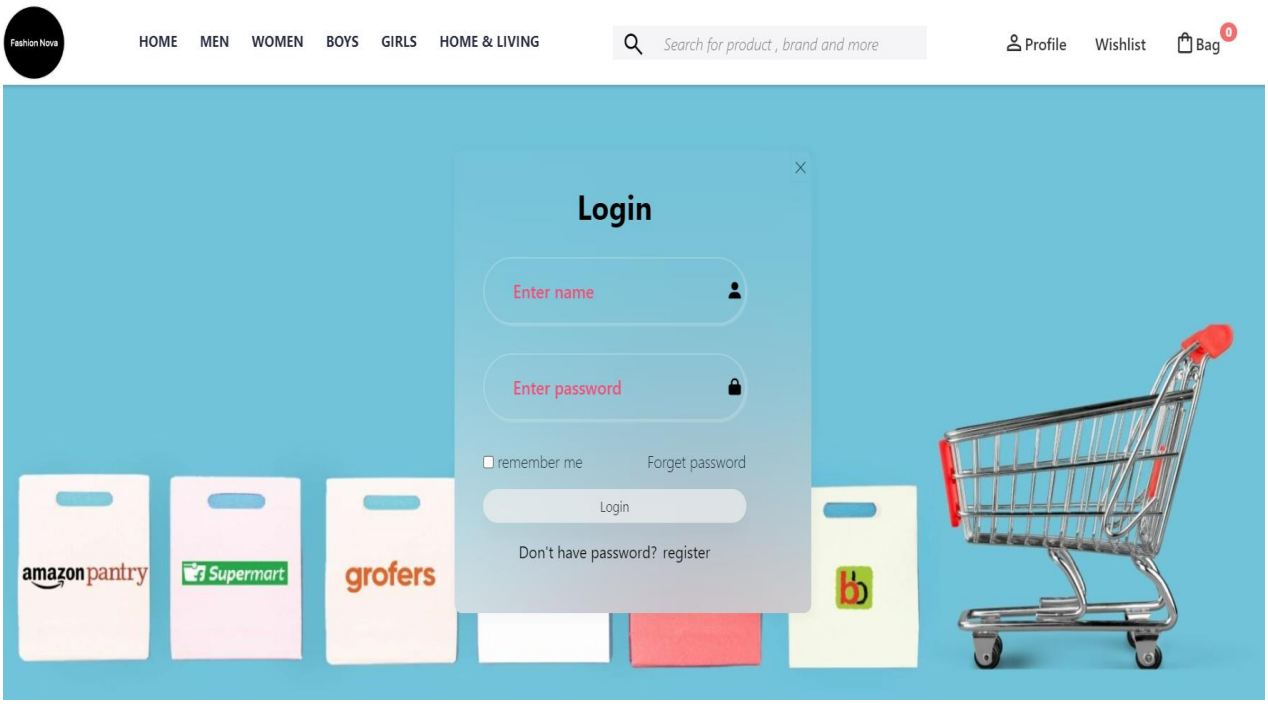
(HOME PAGE)



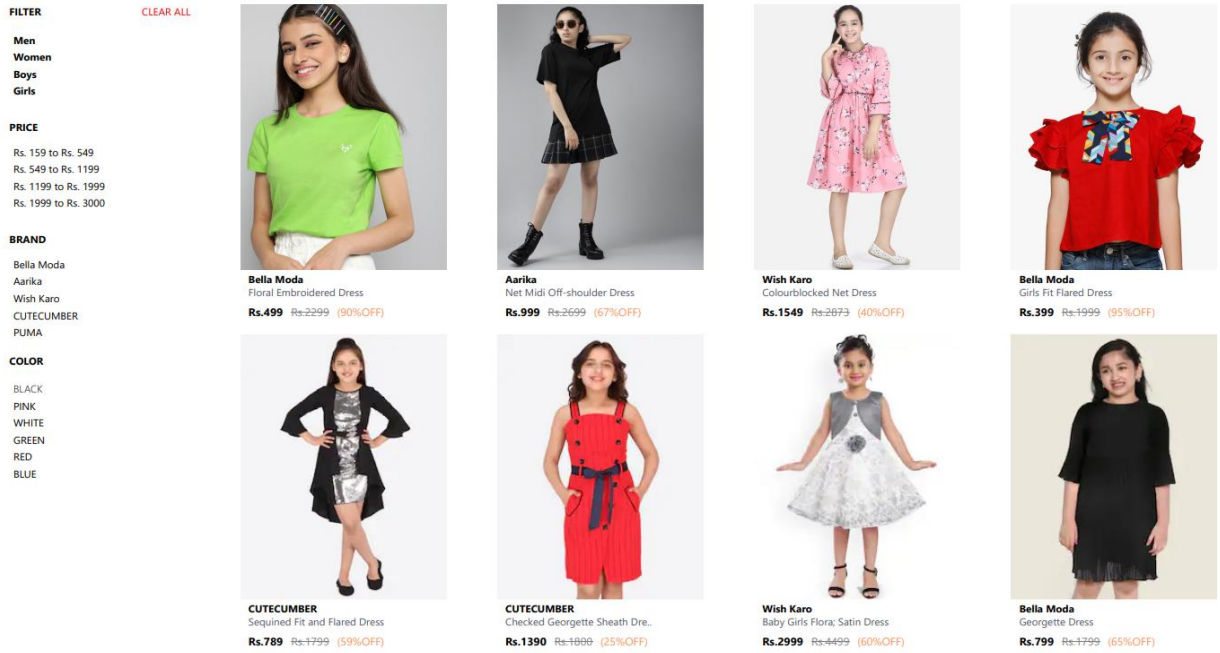
(Men Section)



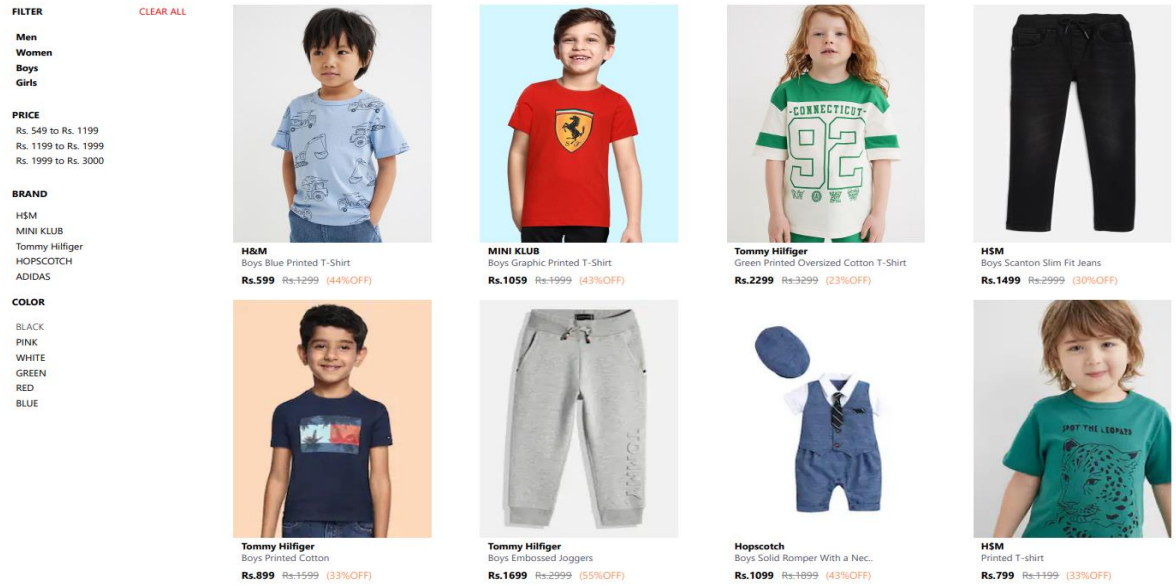
(Women Section)



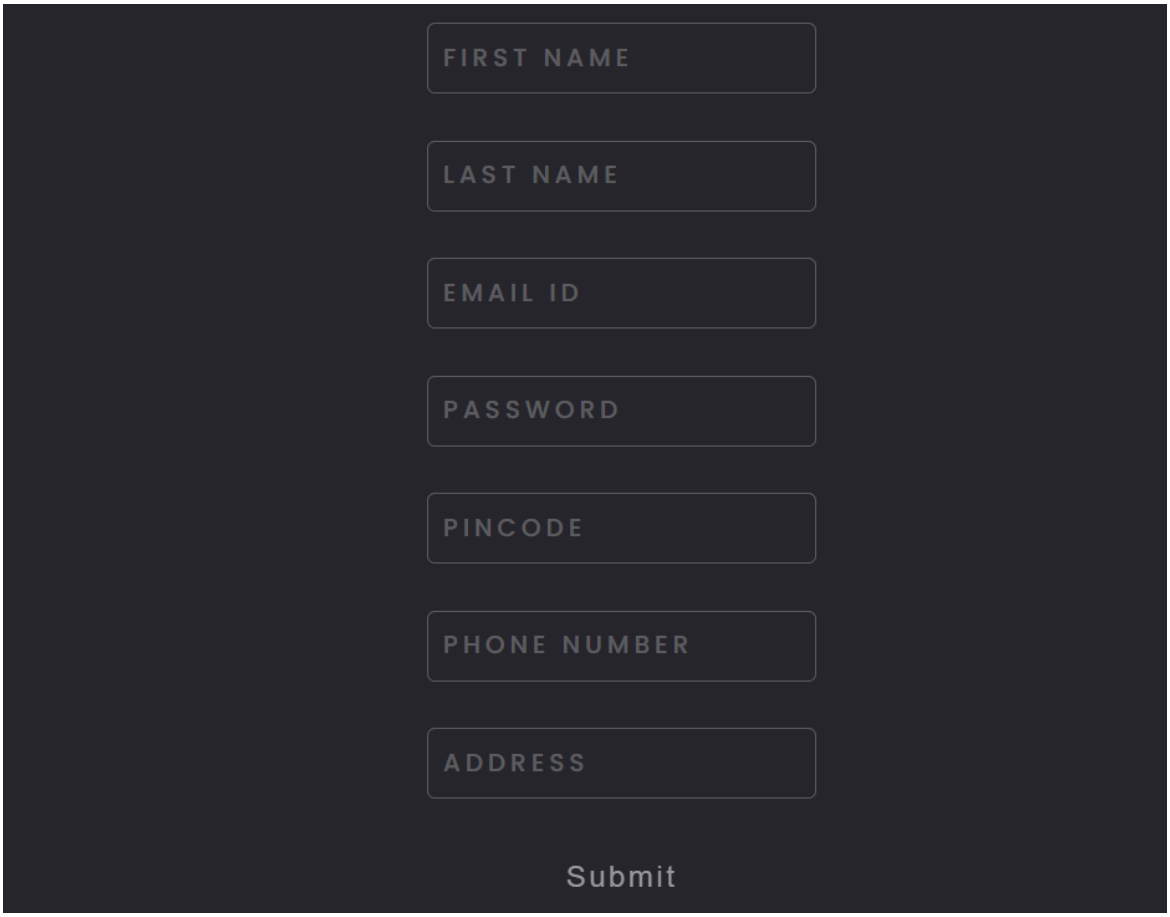
(Login Page)



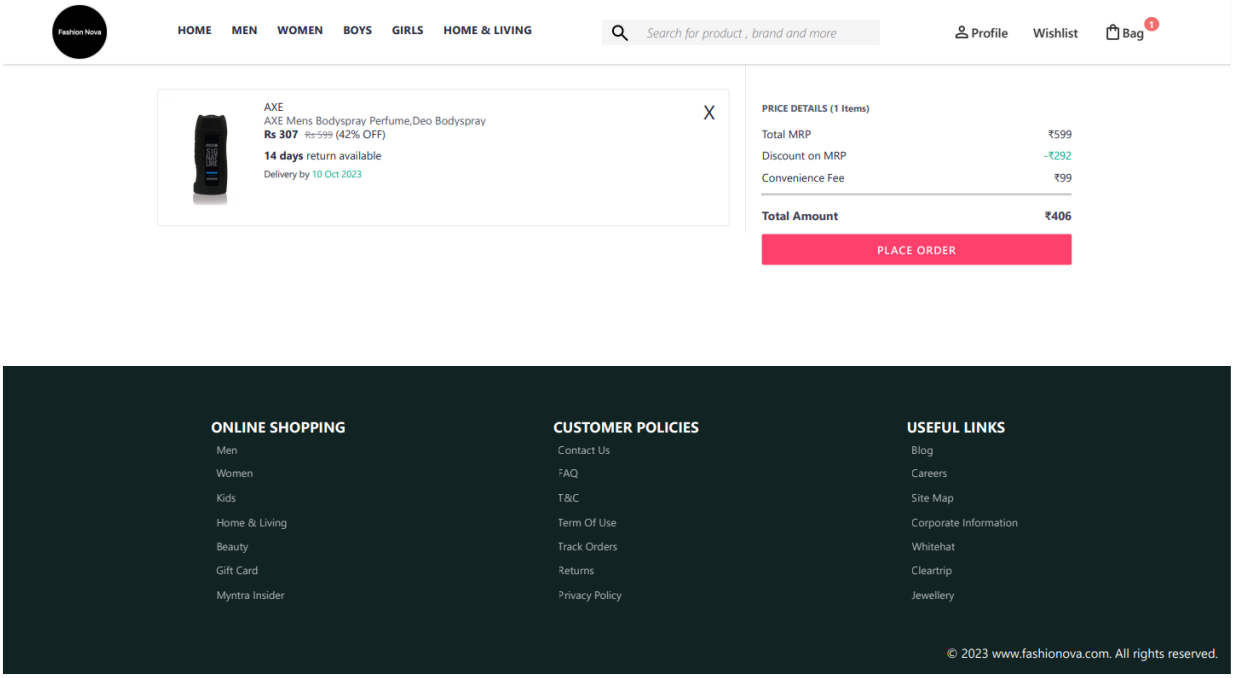
(Girls Section)



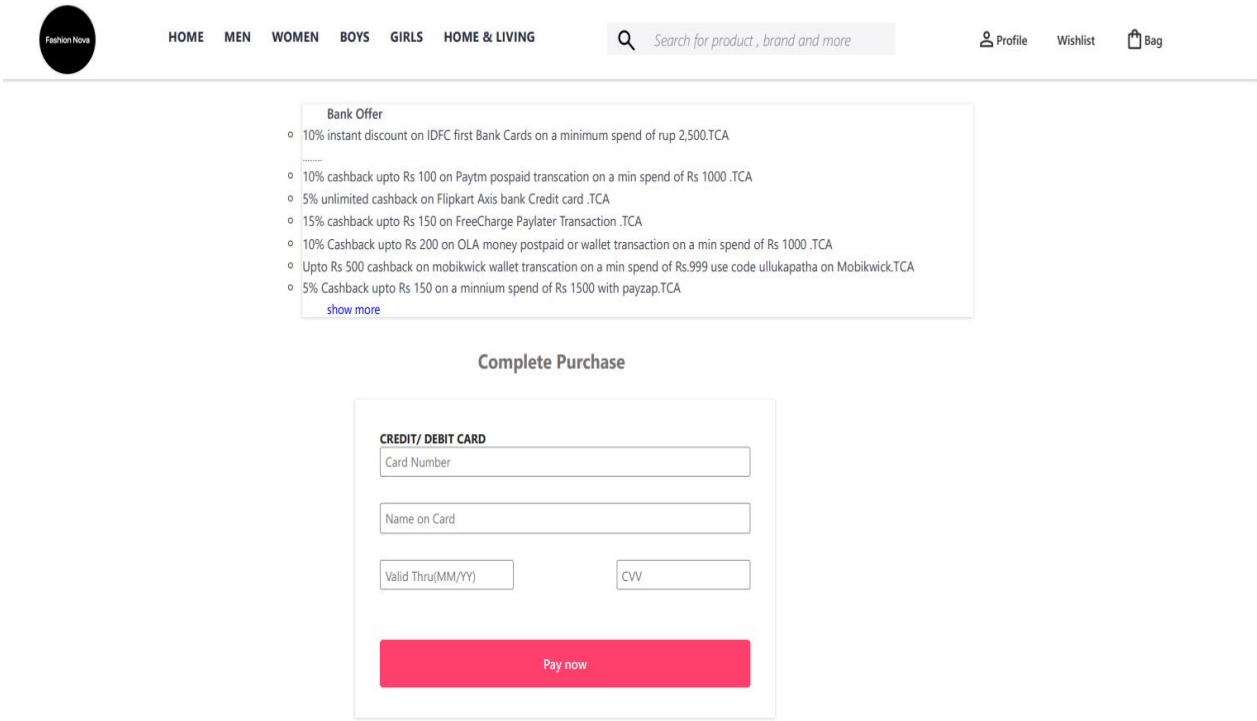
(Boys Section)



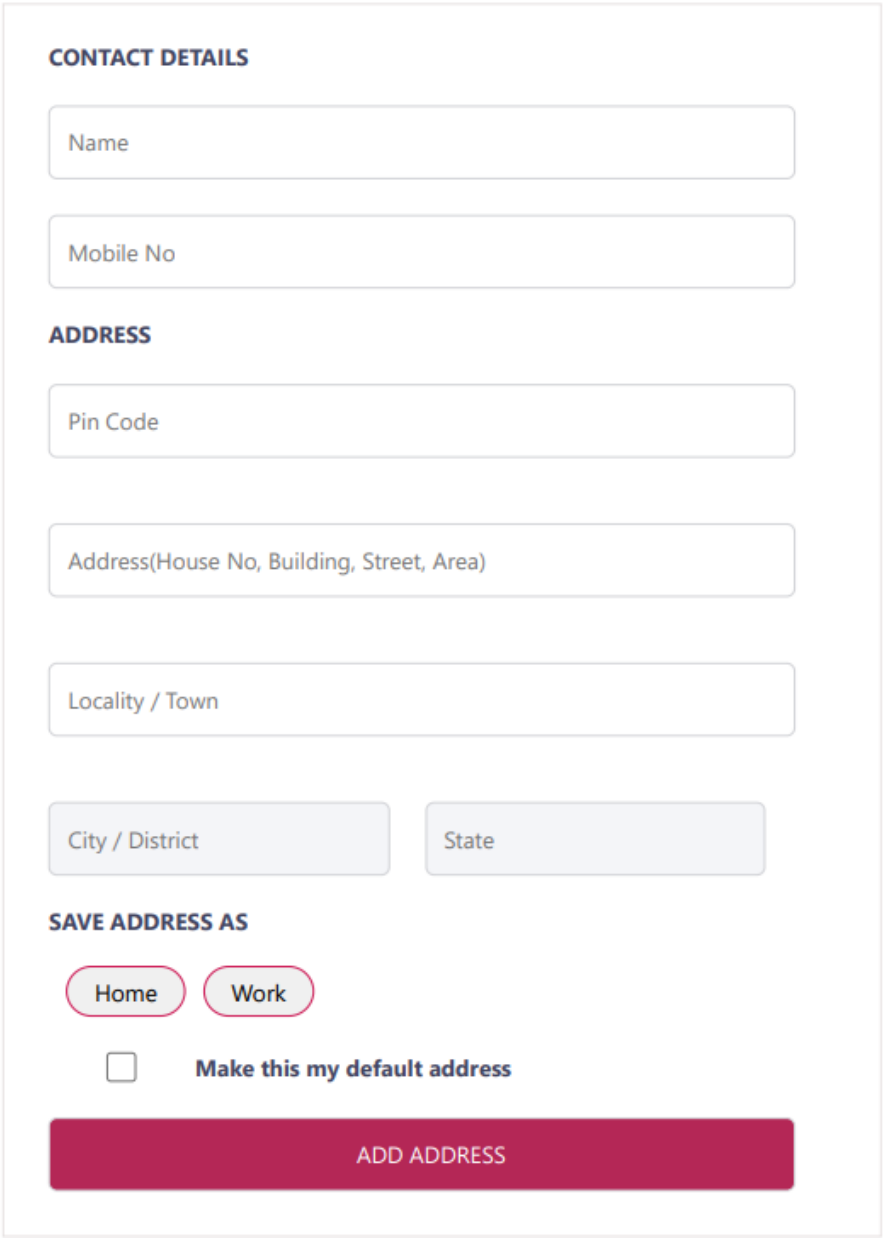
(Forgot Password Recovery)



(Add To Bag Page)



(Payment Page)



(Input Address For Delivery)