SMART SWIPE

Pages

User-Side:

- Homepage
- Sign In and Sign up
- Product Page + Ask Question
- Profile Page
- Cart
- Purchase Page
- Privacy Policy
- Contact Us
- About Us
- Terms and Conditions

Admin side:

- Add Products Page
- Customize Layout (Include Side bars, showing or hiding particular part)
- Add or remove order managers

Order Manager:

- Order page (show pending and completed orders)
- Answer Queries of Customers.

Reference Sites:

- **https://priceoye.pk**
- **https://cubeonline.pk**
- **https://eezepc.com**
- **https://phonebolee.com**

Languages Required:

Front-end

- **HTML/CSS**: For structuring and styling the website.
- **JavaScript**: To add interactivity and dynamic features to your site.
- **React, Angular, or Vue.js**: Modern JavaScript frameworks for building interactive user interfaces.

Back-end Development:

Server-side language: You can choose from several options, including:

- **Node.js (with Express):** JavaScript runtime for server-side programming.
- **Python (with Django or Flask):** A powerful and versatile language with frameworks for web development.
- **Ruby (with Ruby on Rails):** Known for its simplicity and convention over configuration approach.
- **PHP (with Laravel or Symfony):** A popular server-side scripting language for web development.

Database:

- **MySQL:** MySQL is a robust relational database management system that is commonly used for e-commerce websites.

Backend Server:

- **Express (Node.js), Django/Flask (Python), Ruby on Rails (Ruby)**, or
- **Laravel/Symfony (PHP):** These are server-side frameworks that help you handle HTTP requests, manage routes, and interact with the database.

Selected Languages:

1. HTML
2. CSS
3. JavaScript
4. PHP
5. MYSQL

Overview of Uses of languages:

Front-end (HTML/CSS/JavaScript):

- User interacts with the website through the browser.
- JavaScript handles client-side validation and dynamic content updates.
- Sends requests to the server for data and updates.

Back-end (PHP):

- Receives requests from the front-end.
- Processes data, interacts with the MySQL database to fetch or update information.
- Generates dynamic HTML pages to be sent back to the client.

Database (MySQL):

- Stores product information, user data, and order details.
- PHP interacts with MySQL using SQL queries to manage and retrieve data.