



INSTITUTE: BANO QABIL ALKHIDMAT

CAMPUS: IDARA NOOR E HAQ

CONSTRUCTOR: SIR USAMA USMAN

BATCH: FEBURARY TO AUGUST (2023)

SUBMITTED BY:

HOORIYA HANIF	CODE SKETCHERS	2 TO 4
RIDA AZAM	CODE SKETCHERS	2 TO 4
EMAAN SALAHUDDIN	CODE SKETCHERS	2 TO 4

E-Commerce Website Documentation

Setup Instructions:

Node.js and npm:

1. Install Node.js and npm if not already installed.
2. You can download them from the official website: <https://nodejs.org/>

Clone the Repository:

1. Clone the eCommerce website repository from the provided source (e.g., GitHub).
e.g. `git clone [repository_url]`

Installation:

1. Open a terminal and navigate to the project directory.
2. Run “npm install” to install project dependencies.
3. Install additional dependencies:
 - Tailwind CSS: “npm install tailwindcss”
 - React Bootstrap: “npm install react-bootstrap”
 - React Router: “npm install react-router-dom”

Running the Application:

1. In the terminal, run “npm run dev” to start the development server.
2. The application should now be accessible locally in a web browser.

Opening the Application:

1. Open a web browser.
2. Navigate to `http://localhost:3000` (or the port specified in your configuration).

API Integration:

1. Explain how the eCommerce website integrates with APIs (e.g., payment gateways, product inventory).
2. Provide instructions for setting up API keys or credentials.

Folder Structure:

1. Describe the organization of your project's folders and files.
2. Highlight key directories (e.g., components, pages, assets) and their purposes.

React Routers:

1. Explain how React Router is used for navigation within the application.
2. Provide examples of defining routes and navigating between them.

Styling (Tailwind CSS):

1. Explain how to use Tailwind CSS utility classes for styling components.
2. Provide examples of common styling patterns.

Deployment:

1. Offer instructions for deploying the eCommerce website to a hosting platform (e.g., Vercel, Netlify, Heroku).
2. Include any necessary configuration steps.

Customization:

1. Describe how developers can customize the website's appearance, features, and functionality.
2. Provide guidelines for making changes without breaking the core functionality.

Troubleshooting:

1. Compile a list of common issues users or developers might encounter.
2. Offer solutions or workarounds for each issue.