Contents

1.Introduction

Overview

Features

Tech Stack

2.Project Structure

Frontend (React + Vite)

Backend (Node.js + Express)

Database

3.Getting Started

Prerequisites (what i frotend and what is backend, js/node, html css?)

Installation

Running the App (Dev Mode)

4.Frontend

Folder Structure

Routing

State Management

API Communication

Components Overview

Styling

5.Backend

Folder Structure

API Endpoints

Middleware

Authentication / Authorization

Database Schema / Models

Configuration & Environment Variables

6.Core Features

User Authentication (Register/Login)

Product Listing

Product Details

Shopping Cart

Checkout Process

Admin Panel (Product Management, Orders, etc.)

7.Troubleshooting & FAQs

Common Issues

Debugging Tips

Acknowledgments

1.Introduction

OverView:

The Project consists of an online eshop (front,backend and database implemented) with which users can buy products. Also the admins can manage the the orders and products .

Features:

Users have to create an account via registering and so their credentials are saved in a database. Logged in users are able to add or remove items from a cart and checkout whenever they want thus creating an order.

Admins have the power to Edit or Delete an existing product or Add a new one. Last they can delete/remove an order placed whenever they want.

Tech Stack:

The collection of tools,softwares,frameworks e.t.c. used to create the eshop are the following :

-Node.js

-React.js

-MongoDb

-Mongoose

-dotnev

-bcrypt

-express

-express-validator

-axios

If not explained later will explain here

2.Project Structure

What is Frontend?

Frontend refers to the presentation layer that users interact with. For the developing of our eshop we have chosen the React.js + Vite framework. (Wiki)

What is Backend?

Backend involves the data management and processing behind the scenes.

For our backend we are using node.js + express.js framework.(Wiki)

Database:

A Database is the most important thing when developing an app as it is a structured collection of data, essentially being used as a storage for our anytime tampering with said data. In our case we will be using MongoDb which differs from the more known databse Sql in the way of it no storing the data in tables but as collections, thus making it an non relational database.

\*Describe and show relational db and non relational db and tell why we choose mongo for sites\* (scalability and such)