

# Ecommerce - HackerU final project

Welcome to my E-Commerce Hacker-U project. This is a fullstack project built from scratch with frontend and backend, fully responsive to all mobiles and PC.

## Live website:

<https://ty-ecommerce.onrender.com/>

## Install node\_modules:

```
npm i
```

## In order to run server side:

```
npm run server
```

## In order to run client side:

```
npm start
```

## Postman requests documentation of my project:

<https://documenter.getpostman.com/view/26188259/2s93JnVSZa>

## Technologies used to create the Backend:

- Node.js
- mongoose
- dotenv
- bcryptjs
- multer
- morgan
- express.js

## Technologies used to create the Frontend:

- React
- Bootstrap

- CSS
- joi-browser
- axios
- React-Bootstrap
- toastify
- JavaScript

### **Login as admin example:**

- Email: [admin@example.com](mailto:admin@example.com)
- Password: Admin@1234

### **Login as user example:**

- Email: [user@example.com](mailto:user@example.com)
- Password: User@1234

### **Paypal sandbox user for payment:**

- Email: [ty-paypal@example.com](mailto:ty-paypal@example.com)
- Password: 12345678

## **Users can login on live website.**

There are two types of users - admin and normal user.

#### **As normal user:**

A normal user can edit its own profile (name, email and password), a table with 'My orders' that includes paid status (X/V) and a status of delivered order(X/V). A user can easily add products to their cart by clicking the "Add to Cart" button next to the item they want to purchase.

#### **As admin user:**

An admin has the ability to manage by creating new products, editing existing products, managing user accounts by changing user information(name and email only), change user type (to admin), delete users, and viewing all orders placed on the site.

# **Frontend:**

## **Header - Navigation bar**

The header section includes all the links and options necessary for users to navigate the website easily, such as search bar, login and registration, contact and a cart icon that displays the total number of products currently in the user's cart, making it easy for them to keep track of their selections throughout their shopping experience.

## **Add new product**

Using the admin panel, an admin can create a new product by providing details such as the product name, image, price, brand, available stock, category, and description, ensuring that customers have all the necessary information before making a purchase.

## **Edit/delete users as admin**

Admin panel provides the ability to view a list of all registered users, including their personal information(name and email only), order history, and account status, with the option to edit or delete user accounts as needed for efficient management of the website.

## **View all orders as admin**

Admin panel allows admin to not only view all orders placed on the site, but also provides the ability to accept and process orders, enabling the efficient fulfillment of customer purchases and ensuring a positive experience for all users.

## **Register**

The registration process involves the user entering their personal details, which are then validated by joi-browser system to ensure that the information provided is accurate and meets the website's requirements for creating an account. A new user cannot be created if email is already in use. I coded regex validation for both my password and email to ensure that they met the necessary requirements and provided a secure user experience.

## **Login**

When logging into website, users are required to provide their login credentials, which are then validated by joi-browser to ensure that the information entered is accurate and matches the user's account details.

### **Store page**

The store page allows users to browse and view all available products, making it easy for them to find what they are looking for and make a purchase.

### **Search bar**

Search bar allows users to search for specific products by typing in relevant keywords, and displays a list of products that match the query along with the corresponding URL.

### **Place an order**

Once a user has provided their address information and selected their preferred payment method, such as PayPal or Visa, they can easily place an order and complete the transaction, allowing them to receive their desired products in a timely and efficient manner.

### **Reviews system**

Review and rating system is designed to provide users with helpful feedback from other customers, but can only be accessed when the user is logged into their account to ensure that the system is accurate and reliable.

### **Payment system**

Payment system based on trusted and widely-used methods such as PayPal and Visa, making it easy for customers to complete their transactions securely and efficiently.

### **Cart**

The cart feature displays a list of all products that have been added by the user, along with their corresponding prices, and calculates the total cost of the purchase, providing a comprehensive summary of the items selected before proceeding to checkout. Continue to payment page requires login.

### **Product card**

A product card displays all relevant information about a particular item, including product details, images, price, and availability, while also featuring an "Add to Cart" button for easy purchasing, as well as a review system that allows users to provide valuable feedback about their experience with the product.

## **Footer**

The footer section contains all the important information that customers may need, such as contact details, links to social media pages, terms and conditions, privacy policy, shipping and returns policy, and any other relevant information that can help customers make informed decisions before making a purchase.

## **Pagination**

Pagination system is designed to efficiently display a large number of products by dividing them into smaller, more manageable pages, allowing users to navigate through the site with ease and ensuring that load times remain fast and reliable.

## **Carousel product**

Dynamically update the top 3 rated products based on user reviews and ratings, ensuring that the most popular and highly-rated items are always prominently featured and providing valuable feedback for both customers and the website's admin.

## **Redux**

In my ecommerce project, I used Redux to efficiently manage the flow of data between the backend and frontend, allowing me to easily retrieve and update information from the backend as needed, and providing a centralized store for application state that could be accessed by components throughout the frontend, resulting in a more streamlined and responsive user experience.

---

## **Backend:**

### **Hashed password**

To ensure the security of user data in the project, hashed passwords can be used in Node.js by generating a unique salt value for each password, combining the salt and password, and then using a hashing algorithm such as bcrypt to securely store the resulting hash in a database, making it virtually impossible for unauthorized users to access sensitive information even if they manage to gain access to the database.

### **Backend architecture**

the backend organized using a model-view-controller (MVC) architecture, with controllers, routes, models, middlewares, and configuration files all serving specific roles in managing data and functionality within the application, allowing for a more modular and organized approach to development that makes it easier to maintain and scale the project over time.

### **Authorization and authentication**

In my ecommerce project, I employed robust authentication and authorization mechanisms in the backend to ensure that only authorized users had access to sensitive data and functionality, utilizing techniques such as user authentication and role-based access control to authenticate and authorize user actions and manage access to different parts of the application based on user privileges and permissions, resulting in a more secure and reliable ecommerce platform for users and admins.

### **MongoDB**

I utilized the MongoDB cloud database to store and manage all of the project's data, including user account information, product details, order history, and more.