

# E-COMMERCE WEB APPLICATION USING MERN STACK

Submitted by:

INTI JITHENDRAN

Roll No: 22481A1266

BACHELOR OF TECHNOLOGY

Department of INFORMATION TECHNOLOGY

Seshadri Rao Gudlavalleru Engineering College

## ABSTRACT

The rapid digital transformation in recent years has revolutionized how businesses operate and how consumers purchase goods and services. Traditional shopping methods have gradually shifted towards online platforms, offering convenience, flexibility, and a wide range of product choices. This project aims to develop a **full-stack e-commerce application** using the **MERN stack (MongoDB, Express.js, React.js, and Node.js)** that enables both **customers** and **administrators** to manage their activities seamlessly.

The proposed system allows users to **register, log in securely, browse products, add items to the cart, and place orders with delivery addresses**. For security, the system incorporates **JWT (JSON Web Token)** authentication and **bcrypt-based password hashing** to protect user credentials. The system also includes an **admin dashboard** that provides administrative control to manage products, view all orders, and maintain inventory efficiently.

The MERN stack was chosen due to its popularity, scalability, and ability to provide a complete JavaScript-based development environment. MongoDB serves as a NoSQL database for storing dynamic data, while Express.js and Node.js handle server-side logic and API endpoints. React.js is used for building a responsive and dynamic user interface that delivers an excellent user experience. This project provides a robust and scalable solution for businesses looking to transition to an online platform, offering improved user engagement and operational efficiency.

## INTRODUCTION

---

### 2.1 INTRODUCTION

In today's digital era, e-commerce has become an integral part of the global economy. The term **e-commerce**, short for electronic commerce, refers to the buying and selling of goods or services over the Internet. Over the past decade, e-commerce has witnessed exponential growth due to technological advancements, increased Internet penetration, and the convenience it offers to consumers. Instead of visiting physical stores, customers now prefer purchasing products online from the comfort of their homes.

This project focuses on developing a **full-stack e-commerce web application** that provides a seamless shopping experience for users and administrative control for business owners. Unlike traditional websites, this application integrates modern web technologies to ensure **speed, security, and scalability**. The chosen technology stack for this application is the **MERN stack**, which stands for **MongoDB, Express.js, React.js, and Node.js**. Each component plays a crucial role in delivering a dynamic and robust application.

The **front-end** of the application is built using **React.js**, a popular JavaScript library for building user interfaces. React enables the development of responsive and interactive web pages with reusable components, ensuring an engaging user experience. The **back-end** uses **Node.js and Express.js**, which allow building RESTful APIs to handle user requests and manage business logic. The **database** component, MongoDB, is a NoSQL database designed for handling large amounts of unstructured data, making it ideal for applications that require flexibility and scalability.

One of the key considerations in e-commerce applications is **security**. The system handles sensitive user information such as personal details and login credentials. Therefore, this project uses **JWT (JSON Web Token)** for authentication and **bcrypt** for hashing passwords before storing them in the database. These features protect user data against common security threats.

---

### 2.2 WHY E-COMMERCE?

The popularity of e-commerce platforms like **Amazon**, **Flipkart**, and **eBay** highlights the significant demand for online shopping systems. The main advantages include:

- **Convenience:** Users can shop anytime, anywhere, without visiting physical stores.
- **Wide Product Range:** Customers can access a vast array of products from different categories.
- **Time and Cost Efficiency:** Reduces the time spent on shopping and often provides discounts.
- **Scalability:** Businesses can reach a global audience without geographical limitations.

Businesses without an online presence risk losing potential customers. Hence, developing an e-commerce solution is not just an innovation but a necessity in today's competitive market.

---

## 2.3 MERN STACK OVERVIEW

The MERN stack is a popular JavaScript-based technology stack used for building modern web applications. Its four components are:

- **MongoDB:** A NoSQL database that stores data in a flexible, JSON-like format. Ideal for handling dynamic product data and user records.
- **Express.js:** A lightweight web application framework for Node.js that simplifies building APIs.
- **React.js:** A front-end JavaScript library for building responsive and reusable UI components.
- **Node.js:** A runtime environment that executes JavaScript on the server side, ensuring fast and scalable applications.

The advantage of using MERN is that the entire stack uses JavaScript, allowing developers to work with a single language across the front end, back end, and database layers. This improves productivity and reduces complexity.

---

## 2.4 OBJECTIVES OF THE PROJECT

The main objectives of this project are:

### 1. User Functionality:

- Secure registration and login using JWT authentication.
- Browsing and searching products.
- Adding products to the cart and placing orders.

### 2. Admin Functionality:

- Adding, updating, and deleting products.
- Viewing all orders placed by customers.

### 3. Security Measures:

- Password hashing with bcrypt.
- Token-based authentication using JWT.

### 4. Scalable Architecture:

- Use of MongoDB for flexible data storage.
- Modular code structure for easy maintenance.

This project demonstrates the use of **modern web development practices**, combining an interactive front-end interface with a secure and efficient back-end infrastructure.

## SYSTEM REQUIREMENTS & TECHNOLOGY USED

---

### 3.1 SYSTEM REQUIREMENTS

Before building an e-commerce application using the MERN stack, it is essential to identify the hardware and software requirements. These requirements

ensure that the development and deployment environments are suitable for running the application efficiently.

---

### 3.1.1 Hardware Requirements

Component	Specification
Processor	Intel Core i3 or higher (Minimum Dual-Core)
RAM	4 GB minimum (8 GB recommended)
Hard Disk	100 GB free space
Monitor	1024 x 768 resolution or higher
Keyboard & Mouse	Standard input devices

---

### 3.1.2 Software Requirements

Software	Version
Operating System	Windows 10 / Linux / macOS
Node.js	16.x or later
MongoDB	5.x or later
React.js	18.x or later
Browser	Google Chrome / Firefox
IDE	VS Code / WebStorm

---

## 3.2 TECHNOLOGIES USED

This project utilizes the **MERN stack**, which consists of four major technologies:

---

### 3.2.1 MongoDB

- **Description:** MongoDB is a NoSQL database that stores data in a document-oriented format using JSON-like structures.
  - **Why MongoDB?**
    - Flexible schema design.
    - High performance and scalability.
    - Ideal for e-commerce because of varying product attributes.
  - **Role in Project:**

Stores user data, product details, cart items, and order history.
- 

### 3.2.2 Express.js

- **Description:** A minimal and flexible Node.js web application framework.
  - **Features:**
    - Middleware support.
    - Routing system.
    - Handles HTTP requests and responses.
  - **Role in Project:**

Acts as the back-end framework for building RESTful APIs to manage authentication, products, cart, and orders.
- 

### 3.2.3 React.js

- **Description:** A JavaScript library for building user interfaces.
- **Features:**
  - Component-based architecture.
  - Virtual DOM for high performance.
  - Hooks for state and lifecycle management.

- **Role in Project:**  
Powers the front-end user interface for both customers and administrators. Responsible for rendering product listings, cart, and admin dashboard.
- 

### 3.2.4 Node.js

- **Description:** A JavaScript runtime environment for executing JavaScript code outside the browser.
  - **Features:**
    - Non-blocking I/O operations.
    - High concurrency support.
  - **Role in Project:**  
Executes the server-side code, handles API requests, connects with MongoDB.
- 

## 3.3 ADDITIONAL TOOLS

- **Axios:** For making HTTP requests between React and Express API.
- **bcrypt.js:** For hashing passwords before storing them in the database.
- **JWT (JSON Web Token):** For implementing secure authentication.
- **Multer:** For handling file uploads (e.g., product images).
- **CSS:** For styling and creating a responsive design.

## SYSTEM DESIGN

The **system design** phase focuses on the **overall structure of the e-commerce application**, detailing how different components interact, the data flow, and the relationships between entities. It provides **blueprints** for developers to build the application efficiently.

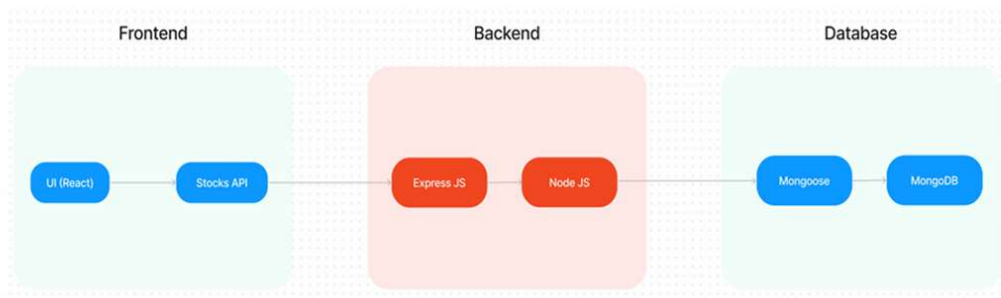
---

## 4.1 ARCHITECTURE OF THE SYSTEM

The project follows a **3-tier MERN stack architecture**, which includes:

- **Presentation Layer (Frontend):**
    - Built with **React.js** for UI rendering.
    - Handles routing, user interactions, and state management.
  - **Application Layer (Backend):**
    - Implemented using **Express.js** and **Node.js**.
    - Provides RESTful APIs for client-server communication.
  - **Database Layer:**
    - Uses **MongoDB** for storing structured and unstructured data.
- 

### 4.1.1 Architecture Diagram



- **React.js:** Displays products, cart, and order details.
  - **Express.js:** Handles authentication, CRUD operations for products, and order management.
  - **MongoDB:** Stores user details, product catalog, cart, and order history.
- 

## 4.2 USE CASE DIAGRAM

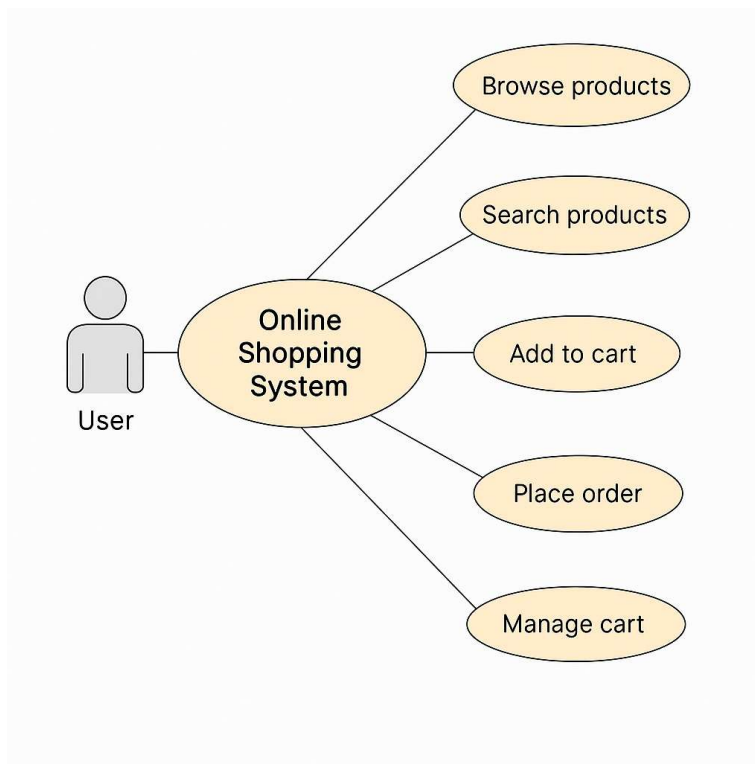
### Actors



- **User:**
  - Register/Login
  - Browse products
  - Add to cart
  - Place orders
- **Admin:**
  - Manage products
  - View all orders

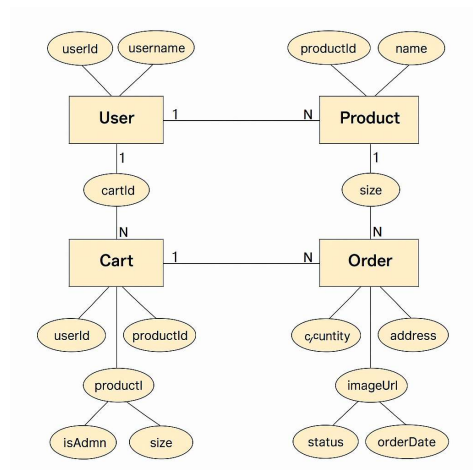
---

### Use Case Diagram (Textual Representation)



### 4.3 ENTITY-RELATIONSHIP (ER) DIAGRAM

+-----+	+-----+	+-----+	+-----+
User	Product	Cart	Order
+-----+	+-----+	+-----+	+-----+
userId	productId	cartId	orderId
username	name	userId	userId
email	price	productId	productId
password	size[]	quantity	quantity
isAdmin	imageUrl	size	size
+-----+	+-----+	+-----+	address
			imageUrl
			status
			orderDate
			+-----+



#### Relationships

- User → Cart → Product
- User → Order → Product

---

### 4.4 FLOWCHART (Order Placement Process)

Start → User Login → View Products → Add to Cart → Enter Address → Confirm Order → Save in Database → Show Order Summary → End

## DETAILED DESIGN & IMPLEMENTATION

This section will cover:

---

### 5.1 FRONTEND (React.js)

The frontend is built using React.js to provide a dynamic and interactive user interface. The application is divided into multiple components such as:

- **Navbar Component**  
Displays navigation links like Home, Products, Cart, Orders, Login, Register, and Logout.  
It also shows the logged-in username and provides Logout functionality.
- **Home Page**  
A visually appealing landing page with a background image and a Shop Now button redirecting to the Products page.
- **Login & Register Pages**  
Designed using custom CSS for a clean and attractive UI with background images and modern input fields.
  - **Login:** Authenticates users and stores JWT token and user info in localStorage.
  - **Register:** Creates a new user account.
- **Products Page**  
Displays all products in card layout with image, name, price, and Add to Cart button.
- **Cart Page**  
Shows items added to the cart with quantity, size, and remove option.
- **Orders Page**  
Allows users to place orders by entering a delivery address and shows the order history with details like product name, size, date, and status.
- **Admin Dashboard**  
Provides full control to admin:

- Add new products with an image upload feature.
  - View all products with the ability to delete.
  - View all orders placed by users, including user details and address.
- 

## 5.2 BACKEND (Node.js + Express.js)

The backend is implemented using Express.js, which provides APIs for:

- Authentication: Register and Login routes with JWT for secure login.
- Product Management: Add, Get, Delete products with image upload.
- Cart Management: Add, View, Delete items in the cart.
- Order Management: Place orders, View user orders, View all orders (Admin).

### Middleware

- authMiddleware.js ensures only authenticated users can access protected routes.
- 

## 5.3 DATABASE (MongoDB)

We use MongoDB for storing application data.

### Collections:

- Users: Stores username, email, password (hashed using bcrypt), and role (admin/user).
- Products: Stores product name, description, price, size array, and image URL.
- Cart: Stores userId, product details, quantity, size.
- Orders: Stores userId, product details, address, order date, status.

## **6. IMPLEMENTATION & SCREENSHOTS**

The implementation of this E-Commerce MERN Application is divided into two major parts:

---

### **6.1 Frontend (React.js)**

- **Framework: React.js**
- **State Management: Local Storage & React Hooks**
- **Routing: React Router DOM**
- **API Communication: Axios**
- **Styling: CSS and custom components**

**Key Features Implemented in Frontend:**

- **User Registration & Login Forms**
  - **Dynamic Navbar (Shows Login/Logout based on state)**
  - **Product Listing Page with Add-to-Cart**
  - **Cart Page with Update/Delete Items**
  - **Orders Page (Place Order & View Orders)**
  - **Admin Dashboard for Product Management & Viewing Orders**
- 

### **6.2 Backend (Node.js + Express)**

- **Authentication: JWT-based token system**
- **Password Security: bcrypt.js for hashing**
- **Database: MongoDB with Mongoose models**
- **File Uploads: Multer for product images**
- **APIs Implemented:**
  - **/auth → Register, Login**

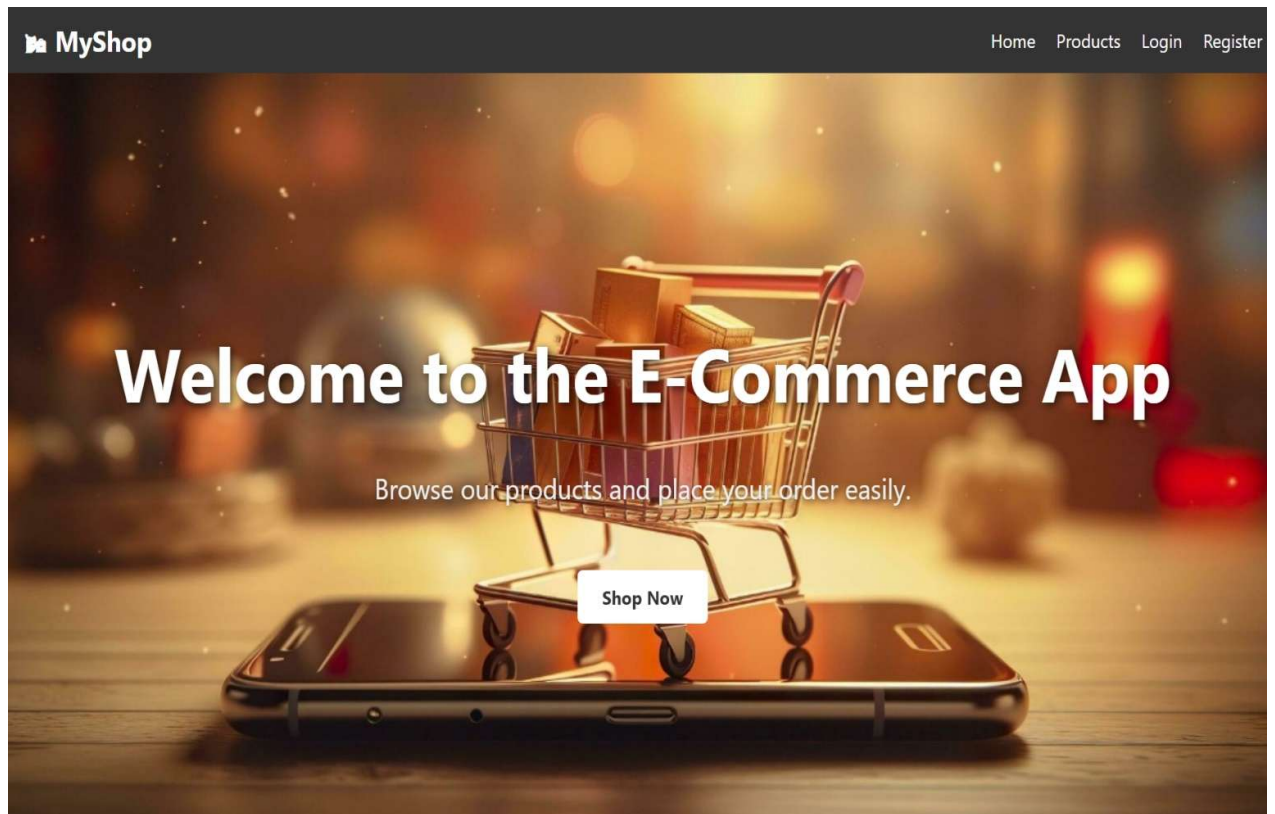
- /products → Add, Fetch, Delete products
- /cart → Add to cart, View cart, Delete from cart
- /orders → Place order, Fetch user orders, Fetch all orders (Admin only)

---

## 6.3 Screenshots

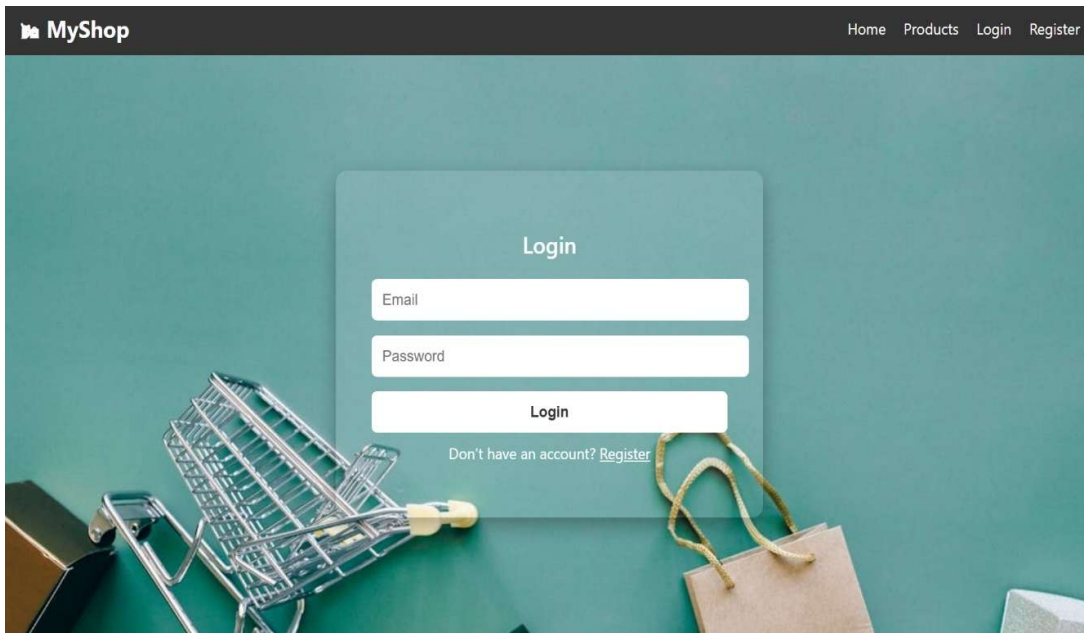
---

### 6.3.1 Home Page



---


### 6.3.2 Login Page



---

### 6.3.3 Products Page

### Available Products




**HP LAPTOP**

HP 15, AMD Ryzen 5 7530U Laptop (16GB DDR4,512GB SSD) Anti-Glare, Micro-Edge,15.6"/39.6cm, FHD, Win11, M365 Basic(1yr),Office Home24, Silver,1.59kg, Radeon Graphics, FHD Camera w/Shutter, fc0389AU

**₹49999**

[Add to Cart](#)




**WATCH**

Fastrack Limitless Glide X 1.83" Smart Watch with Ultra UV HD Display, SpO2, Heart Rate & Sleep Tracking, Bluetooth Calling, 100+ Sports Modes, 5-Day Battery, Smartwatch for Men & Women (Black)

**₹1399**

[Add to Cart](#)




**ONE PLUS MOBILE**

OnePlus Nord 4 5G (Oasis Green, 8GB RAM, 256GB Storage) | Lifetime Display Warranty | Qualcomm® Snapdragon™ 7 Plus Gen 3 | ANTUTU Score 1.5Mn AI

**₹29599**

[Add to Cart](#)




**iphone**

Apple iPhone 13 (128GB) - Blue

**₹43999**

[Add to Cart](#)




**BOAT Airdopes**

boAt Airdopes 121 Pro Plus, 100HRS Battery, 4Mics ENx, 50ms Low Latency, Fast Charge, LED Display, IPX5, v5.3 Bluetooth in Ear Earbuds, TWS Ear Buds Wireless Earphones with mic (Black)

**₹1299**

[Add to Cart](#)




**TV**

VW 80 cm (32 inches) Frameless Series HD Ready Android Smart LED TV VW32S (Black)

**₹7999**

[Add to Cart](#)




**BAG**

American Tourister Valex 28 Ltrs Large 5 Compartment Laptop Backpack with Bottle Pocket and Front Organizer-Black

**₹1299**

[Add to Cart](#)



**WIFI**


TP-Link WiFi 6 AX1500 Mbps Archer AX10,Smart WiFi,Triple-Core CPU, Gigabit, Dual-Band OFDMA, MU-MIMO, Compatible with Alexa, Wireless Router,Black

**₹2999**

[Add to Cart](#)

### 6.3.4 Cart Page

### Your Cart




**iphone**

Quantity: 1

Size: 128

[Remove](#)



**BAG**

Quantity: 1

Size: ---

[Remove](#)



### 6.3.5 Orders Page

MyShop


HomeProductsCartOrdersRajuLogout

Place Order

Enter delivery address

Place Order

Your Orders



TV

Qty: 1

Size: 32inch

Address: AP,west godavari dist, mogalthur, 534281

Date: 20/6/2025, 2:54:29 pm

Status: Pending

### 6.3.6 Admin Dashboard

MyShop

HomeProductsAdminJithendraLogout

Admin Dashboard

Add New Product

Product Name

Description

Price

Size


Choose FileNo file chosen

Add Product

All Products


HP LAPTOP — ₹49999

HP 15, AMD Ryzen 5 7530U Laptop (16GB DDR4,512GB SSD) Anti-Glare, Micro-Edge,15.6"/39.6cm, FHD, Win11, M365 Basic(1yr),Office Home24, Silver,1.59kg, Radeon Graphics, FHD Camera w/Shutter, fc0389AU




WATCH — ₹1399

Fastrack Limitless Glide X 1.83" Smart Watch with Ultra UV HD Display, SpO2, Heart Rate & Sleep Tracking, Bluetooth Calling, 100+ Sports Modes, 5-Day Battery, Smartwatch for Men & Women (Black)



ONE PLUS MOBILE — ₹29599

OnePlus Nord 4 5G (Oasis Green, 8GB RAM, 256GB Storage) | Lifetime Display Warranty | Qualcomm® Snapdragon™ 7 Plus Gen 3 | ANTUTU



All Orders


ONE PLUS MOBILE — Qty: 1 | Size: 8GB+256GB

User: Prakash (prakash@gmail.com)

Address: .....

Date: 20/6/2025, 3:15:06 pm

Status: Pending



iphone — Qty: 1 | Size: 128

User: Prakash (prakash@gmail.com)

Address: A.P, krishna district, gudlavalleru

Date: 20/6/2025, 3:12:05 pm

Status: Pending

