

# Medicare Shopping Website - Project Documentation

---

## 1. Title & Introduction Section

---

### Slide 1: Title Slide

College/Institution: [Your College Name Here]

## Medicare Shopping Website

---

A Comprehensive E-Commerce Solution for Healthcare Products

---

### Slide 2: Presenter Information

Presented By: [Your Name]

---

### Slide 3: Abstract

**Project Summary:** "Medicare" is a robust, web-based e-commerce application designed to streamline the purchase of medicines and healthcare products. It provides a seamless platform for users to browse, search, and order medical supplies while offering administrators powerful tools to manage inventory, orders, and products.

#### Key Technologies:

- **Frontend:** React.js, Tailwind CSS
- **State Management:** React Context API
- **Data Handling:** LocalStorage (Prototype) / JSON Mock Data

**Conclusion:** The system is designed to be efficient, user-friendly, and responsive. Its modular architecture allows for easy scalability and future enhancements, such as integration with real-time payment gateways and backend databases.

---

### Slide 4: Introduction

**Overview:** This web-based application addresses the need for accessible healthcare by bringing the pharmacy to the user's fingertips. It bridges the gap between suppliers and consumers through a digital interface.

#### Technical Stack:

- **Coding Languages:** JavaScript (ES6+), HTML5, CSS3
- **Database:** LocalStorage (Client-side persistence for prototype) / Mock JSON Data

#### User Roles:

1. **Admin:** Manages the entire platform, including product catalog, order processing, and system settings.
2. **User (Customer):** Browses products, manages cart, places orders, and tracks delivery status.

## 2. Modules Section

---

### Slide 5: Admin Module

The Admin module grants privileged access to control system operations. Key features include:

- **Dashboard:** View real-time statistics (Total Users, Orders, Revenue).
- **Product Management:** Add, edit, or delete medicines and health products.
- **Order Management:** View, approve, or reject customer orders.
- **Prescription Verification:** Review and approve uploaded prescriptions.
- **Settings:** Configure system-wide preferences and profile settings.

### Slide 6: User Module

The User module focuses on a smooth shopping experience. Key capabilities include:

- **Browse & Search:** Search medicines by category, name, or health concern.
- **Product Details:** View detailed information (ingredients, dosage, usage) and reviews.
- **Cart & Wishlist:** Manage selected items and save favorites for later.
- **Checkout:** Securely place orders with address management.
- **Order History:** Track the status of current and past orders.
- **Prescription Upload:** Upload doctor's prescriptions for verified purchase.

### 3. Requirements Section

---

#### Slide 7: Hardware Configuration (Client Side)

Component	Minimum Requirement	Recommended
Processor	Intel Core i3 or equivalent	Intel Core i5 or higher
RAM	4 GB	8 GB
Hard Disk	256 GB HDD/SSD	512 GB SSD
Display	1366 x 768 Resolution	1920 x 1080 (Full HD)
Internet	2 Mbps BroadBand	10 Mbps or higher

#### Slide 8: Hardware Configuration (Server Side)

*(For Hosting/Deployment)*

Component	Minimum Requirement	Recommended
Processor	Dual Core CPU	Quad Core vCPU
RAM	2 GB	4 GB
Storage	20 GB SSD	40 GB SSD
Architecture	x86_64	x86_64

#### Slide 9: Software Requirement (Client)

Component	Requirement
Operating System	Windows 10/11, macOS, or Linux
Web Browser	Google Chrome, Firefox, Safari, or Edge (Latest Versions)

#### Slide 10: Software Requirement (Server/Development)

Component	Requirement
Operating System	Linux (Ubuntu/Debian) or Windows Server
Web Server	Nginx or Apache (for serving static build)

Component	Requirement
Runtime Environment	Node.js (for development/build tools)
Database	<i>LocalStorage (Current)</i> / MongoDB/MySQL (Future)

## 4. Diagrams Section

---

### Slide 11: Use Case Diagram (Admin)

```
usecaseDiagram
    actor Admin
    rectangle "Medicare System" {
        usecase "Login" as UC1
        usecase "Manage Products" as UC2
        usecase "Manage Orders" as UC3
        usecase "Verify Prescriptions" as UC4
        usecase "View Dashboard" as UC5
    }
    Admin --> UC1
    Admin --> UC2
    Admin --> UC3
    Admin --> UC4
    Admin --> UC5
```

### Slide 12: Use Case Diagram (User)

```
usecaseDiagram
    actor User
    rectangle "Medicare System" {
        usecase "Register/Login" as UC1
        usecase "Search Products" as UC2
        usecase "Add to Cart" as UC3
        usecase "Place Order" as UC4
        usecase "Upload Prescription" as UC5
        usecase "Write Review" as UC6
    }
    User --> UC1
    User --> UC2
    User --> UC3
    User --> UC4
    User --> UC5
    User --> UC6
```

## Slide 13: Class Diagram

```
classDiagram
class User {
    +int id
    +string username
    +string email
    +string password
    +string role
    +login()
    +register()
}
class Product {
    +int id
    +string name
    +float price
    +string category
    +string description
    +int stock
    +updateStock()
}
class Order {
    +string id
    +int userId
    +date date
    +float total
    +string status
    +updateStatus()
}
class Cart {
    +int userId
    +List~Product~ items
    +addItem()
    +removeItem()
    +checkout()
}

User "1" --> "*" Order : places
User "1" --> "1" Cart : has
Order "1" --> "*" Product : contains
Cart "1" --> "*" Product : contains
```

## Slide 14: ER Diagram

```
erDiagram
    USER ||--o{ ORDER : places
```

```
USER ||--o{ REVIEW : writes
PRODUCT ||--o{ ORDER_ITEM : "included in"
ORDER ||--|{ ORDER_ITEM : contains
PRODUCT ||--o{ REVIEW : has
```

```
USER {
    int id PK
    string email
    string password
    string role
}
PRODUCT {
    int id PK
    string name
    float price
    int stock
    string category
}
ORDER {
    string id PK
    int user_id FK
    float total
    string status
    date order_date
}
ORDER_ITEM {
    string order_id FK
    int product_id FK
    int quantity
    float price
}
```

## Slide 15: Data Flow Diagram (DFD) Analysis

**Definition:** A Data Flow Diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows to show data inputs, outputs, storage points and the routes between each destination.

## Slide 16: Zero Level DFD (Context Diagram)

```
graph LR
    User[User] -- Orders/Requests --> System(Medicare System)
    Admin[Admin] -- Management/Updates --> System
    System -- Order Status/Products --> User
    System -- Reports/Data --> Admin
```

## Slide 17: First Level DFD (Admin)

```
graph TD
    Admin[Admin] -->|Login| P1(Process: Auth)
    P1 -->|Session| D1[(Dashboard)]
    Admin -->|Product Details| P2(Process: Manage Products)
    P2 -->|Update| DB1[(Product DB)]
    Admin -->|Status Update| P3(Process: Manage Orders)
    P3 -->|Update| DB2[(Order DB)]
```

## Slide 18: First Level DFD (User)

```
graph TD
    User[User] -->|Search| P1(Process: Browse)
    P1 -->|Query| DB1[(Product DB)]
    User -->|Select Item| P2(Process: Add to Cart)
    P2 -->|Cart Data| D1[(Cart Store)]
    User -->|Checkout| P3(Process: Place Order)
    P3 -->|Order Details| DB2[(Order DB)]
```

## Slide 19: Second Level DFD (Order Processing)

```
graph TD
    User[User] -->|1. Select Items| Cart[Cart Management]
    Cart -->|2. Review Cart| Checkout[Checkout Process]
    Checkout -->|3. Enter Address| Validation{Validate Details}
    Validation -->|Invalid| Checkout
    Validation -->|Valid| Order[Order Generation]
    Order -->|4. Save Order| DB1[(Order DB)]
    Order -->|5. Update Stock| ProductDB[(Product DB)]
    Order -->|6. Notify Admin| Admin[Admin Dashboard]
```

## 5. Testing & Showcase Section

---

### Slide 20: Implementation and System Testing

**Testing Strategy:** We employed a rigorous testing approach to ensure system reliability:

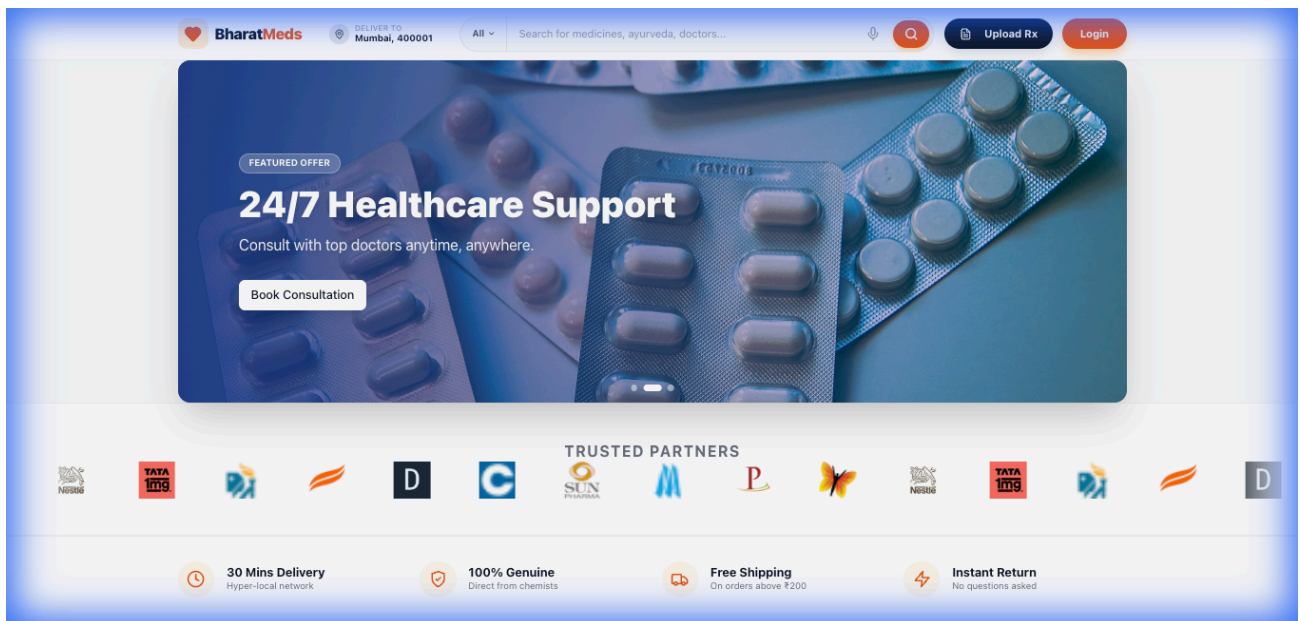
1. **Unit Testing:** Individual components (e.g., `ProductCard`, `CartItem`) were tested in isolation to verify their logic and rendering.

2. **Integration Testing:** Modules were tested together (e.g., "Add to Cart" flow updating the Context state) to ensure seamless data flow.
3. **System Testing:** End-to-end scenarios (Login -> Browse -> Order -> Admin Approval) were executed to validate the complete workflow.

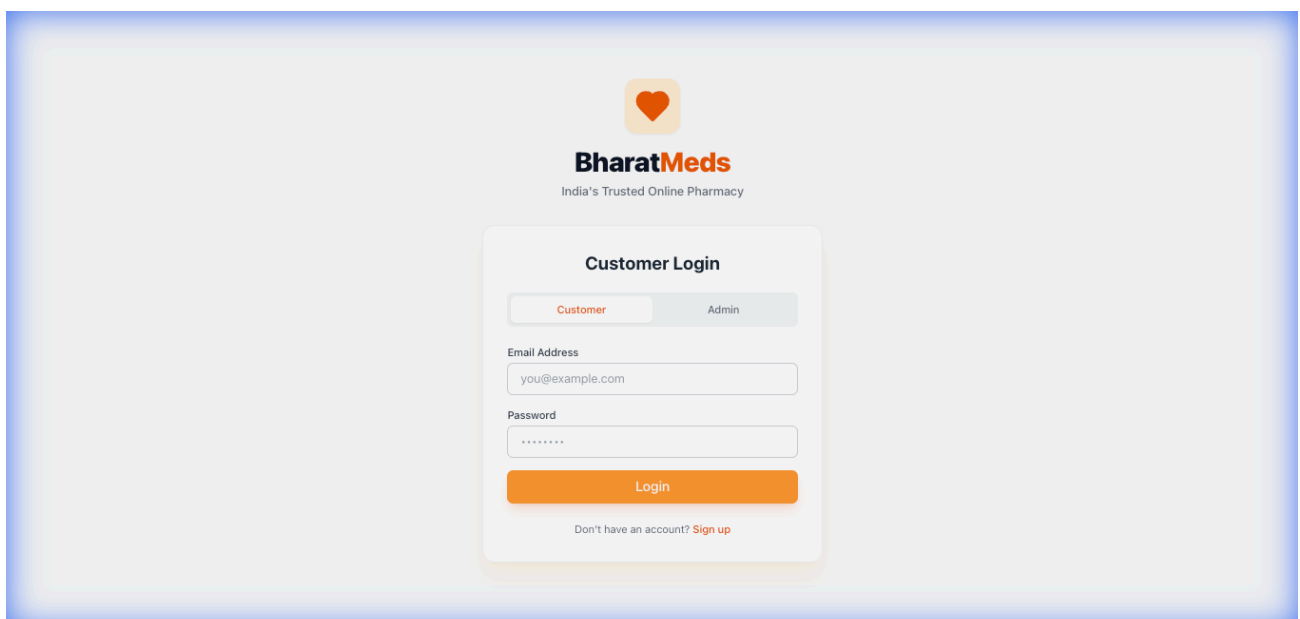
## Slides 21-34: Project Screens

*(Screenshots will be inserted below)*

### Homepage

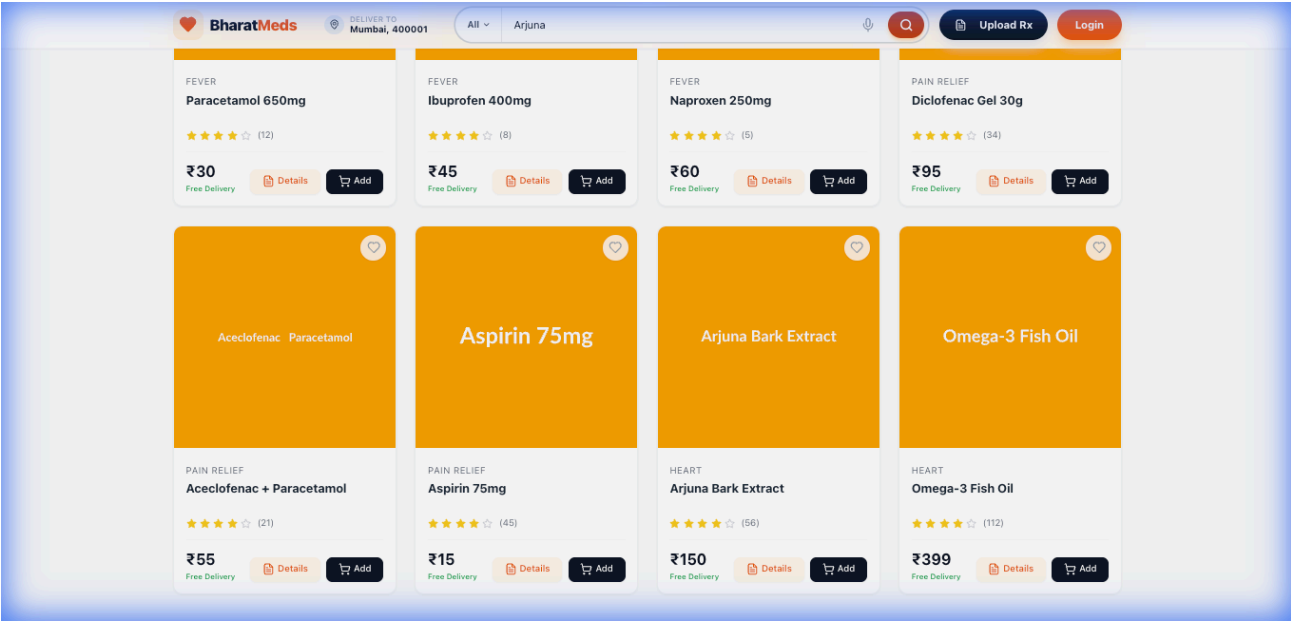


### Login Screen

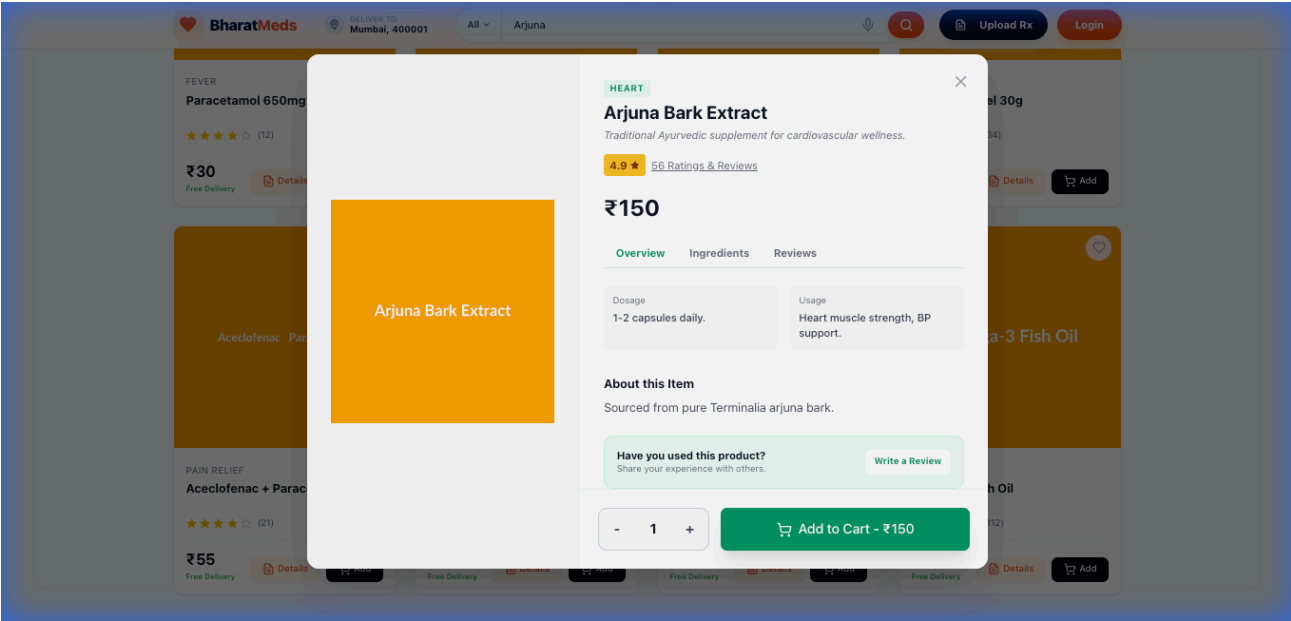




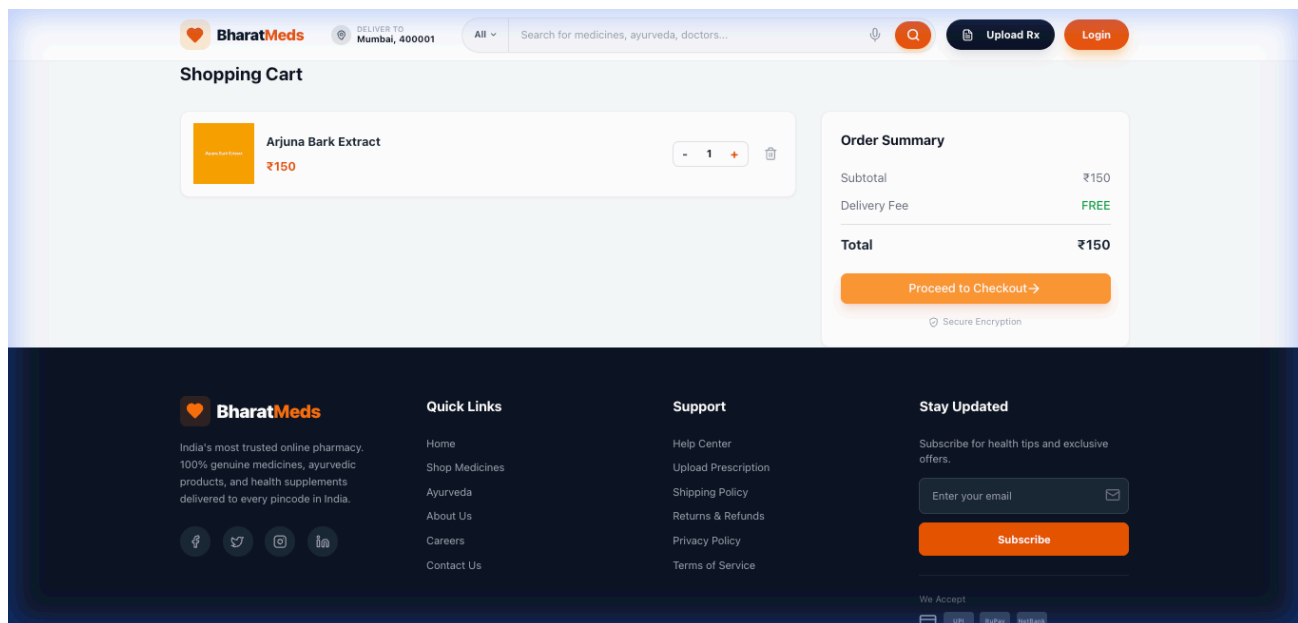
Product Search Results



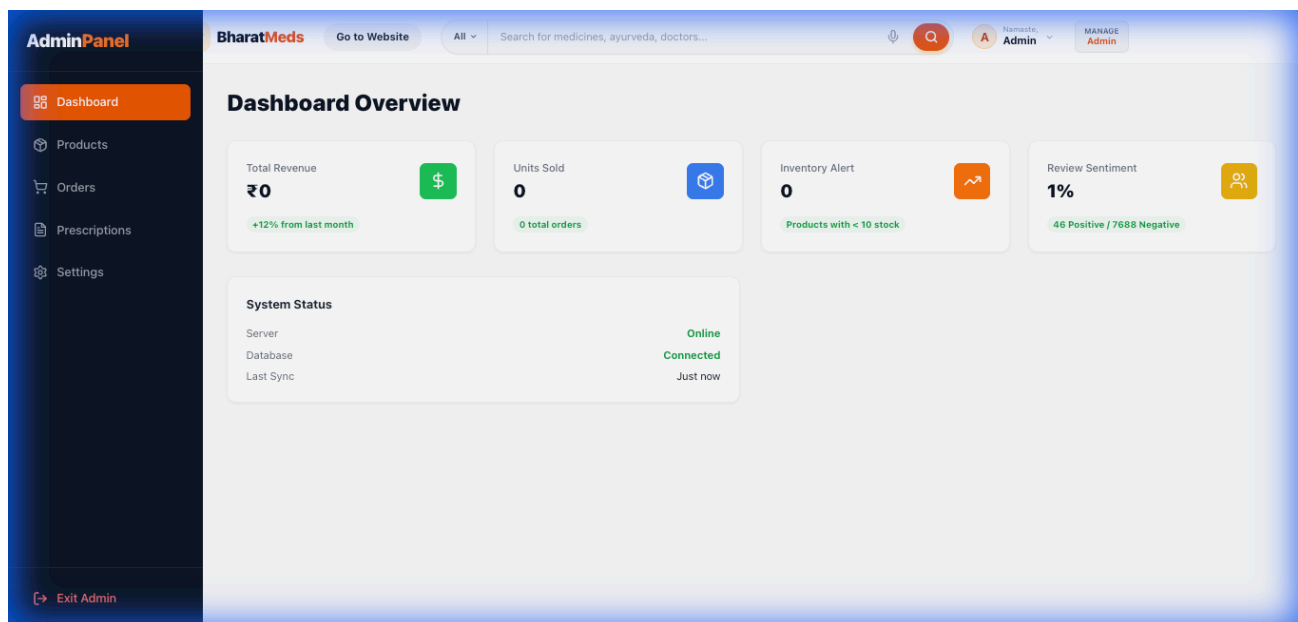
Product Details Modal



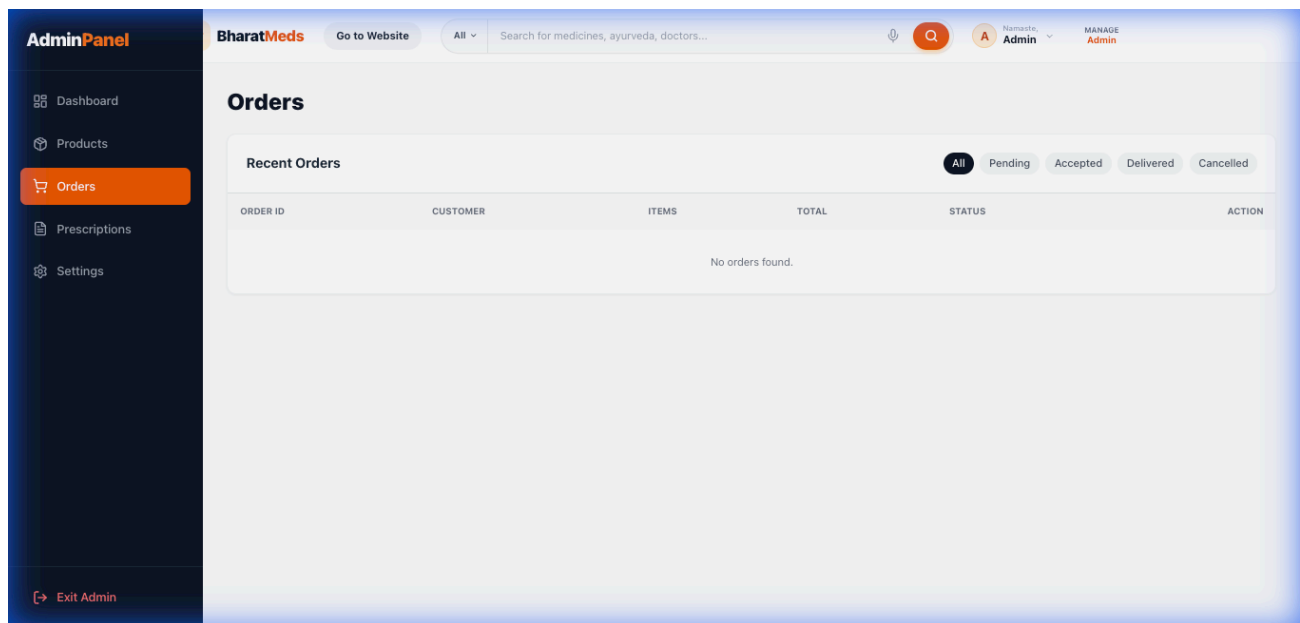
Cart Page



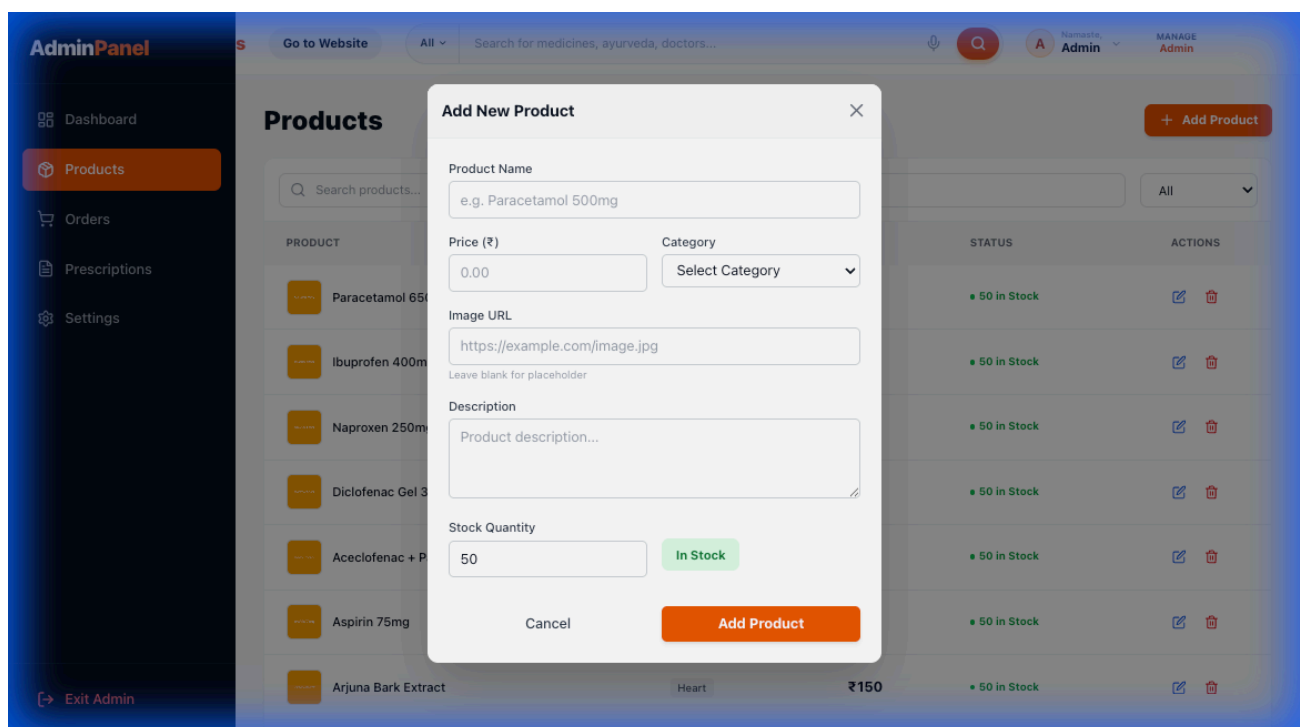
## Admin Dashboard



## Admin Order Management



## Admin Product Management



## Checkout Page

BharatMeds

DELIVER TO

Mumbai, 400001

All

Search for medicines, ayurveda, doctors...

Upload Rx

R

Namaste, Rudra

1

←

Checkout & Payment

Delivery Address

Enter your full street address...

Detect Location

Payment Method

Cash on Delivery

Pay fast with cash/UPI upon delivery

Pay Online (UPI/Cards)

Secure payment via Razorpay

Order Summary

Subtotal

₹30

Delivery Fee

FREE

Items

1

Total

₹30

Confirm Order

Secure Encryption

BharatMeds

India's most trusted online pharmacy.  
100% genuine medicines, ayurvedic  
products, and health supplements

Quick Links

Home

Shop Medicines

Ayurveda

Support

Help Center

Upload Prescription

Shipping Policy

Stay Updated

Subscribe for health tips and exclusive  
offers.

Get Newsletter

## Registration Page

BharatMeds

India's Trusted Online Pharmacy

Create Account

Full Name

John Doe

Email Address

you@example.com

Password

\*\*\*\*\*

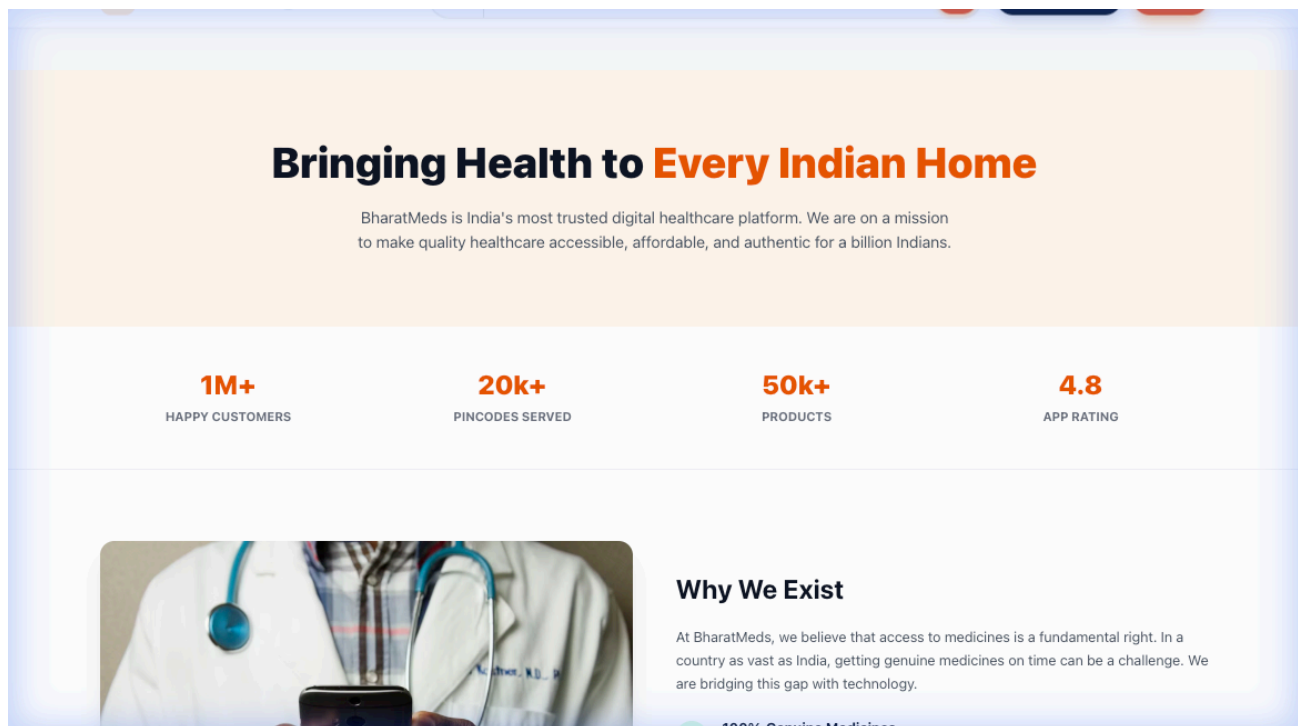
Confirm Password

\*\*\*\*\*

Sign Up

Already have an account? Login

## About Us Page



## Contact Us Page

The image displays the "Contact Us" page of the BharatMeds website. The top navigation bar includes the BharatMeds logo, a "DELIVER TO" dropdown set to "Mumbai, 400001", a search bar with the placeholder "Search for medicines, ayurveda, doctors...", and buttons for "Upload Rx" and "Login". The main content area has a heading "Get in Touch" with the subtext "We are here to help you 24/7". On the left, a light orange box titled "Contact Information" contains three items: "CALL US 1800-108-1008" (Toll-free, 24/7), "EMAIL US care@bharatmeds.in", and "VISIT US BharatMeds HQ" (Bandra Kurla Complex, Mumbai, Maharashtra - 400051). On the right, a white box titled "Send us a Message" contains input fields for "First Name" and "Last Name", an "Email" field, and a "Message" text area. A large orange "Send Message" button is at the bottom of this form.

## Privacy Policy Page

## Terms & Policies

### Privacy Policy

Last updated: 2/15/2026

At BharatMeds, we take your privacy seriously. This policy describes how we collect, use, and handle your personal information.

- We collect information you provide directly to us (name, address, prescription).
- We use this information to process orders and improve our services.
- We do not sell your personal data to third parties.
- Your payment data is processed securely via PCI-DSS compliant partners.

### Terms of Service

By using BharatMeds, you agree to the following terms:

- You must be at least 18 years old to use this service.
- Prescription medicines require a valid doctor's prescription.
- Prices are subject to change without notice.
- We reserve the right to refuse service to anyone.

### Shipping & Returns

Shipping

## 6. Closing Section

---

### Slide 35: Conclusion

#### Summary of Benefits:

- **Efficiency:** Automates the manual process of order taking and inventory tracking.
- **User Experience:** Provides a clean, modern interface for effortless navigation.
- **Accessibility:** Makes healthcare products available 24/7.
- **Scalability:** Built on a modular React architecture, ready for backend integration and mobile app expansion.

### Slide 36: Bibliography & References

1. React Documentation: [react.dev](https://react.dev)
2. Tailwind CSS: [tailwindcss.com](https://tailwindcss.com)
3. Vite Build Tool: [vitejs.dev](https://vitejs.dev)
4. MDN Web Docs: [developer.mozilla.org](https://developer.mozilla.org)

### Slide 37: Thank You

# Thank You!

---

Questions?