

Digital Egypt Pioneers Initiative

React Front-End Development

Medical Application by React

Care Plus

**Supervised by:**

Eng. Basma Abdel Halim

**Team members:**

Banoub Nagy Edwar

Asmaa Mahmoud Ebeed

Mark Moheb Mamdouh

Sabah Abdelbaset Ahmed Ali

Yousef Ayman Metry

Acknowledgement

We extend our heartfelt appreciation to Eng. Basma Abdel Halim for her invaluable guidance and unwavering support throughout our journey in the Digital Egypt Pioneers Initiative. Her mentorship has been instrumental in shaping our experience, providing us with the knowledge, insights, and encouragement needed to excel.

Her technical expertise, constructive feedback, and commitment to fostering innovation have played a crucial role in the development and success of this initiative. She has consistently provided thoughtful advice and practical solutions, helping us overcome challenges and refine our approach.

We are truly grateful for the opportunities to learn, grow, and collaborate under her insightful leadership. Her dedication and passion have left a lasting impact on our journey, and we sincerely appreciate her efforts in guiding and inspiring us every step of the way.

Abstract

The growing demand for medical and healthcare supplies has highlighted the need for efficient, secure, and user-friendly digital platforms to facilitate their accessibility. This project focuses on developing an e-commerce platform for medical and health-related products, utilizing React for the front-end to deliver a seamless and responsive user experience. The platform is designed to streamline the purchasing process, ensuring ease of use for both individual consumers and healthcare institutions. By integrating a dedicated back-end system, it enables secure transactions, effective inventory management, and a structured approach to order processing. The system is tailored to provide a reliable and scalable solution that meets the specific needs of the medical sector.

Through the integration of modern front-end technologies and a well-structured back-end architecture, this project aims to enhance the accessibility and availability of essential medical supplies. It contributes to the digital transformation of healthcare commerce, offering a practical and innovative approach to online medical product distribution.

Table of Contents

1. Project Planning & Management5

1.1. Project Proposal5

* 1. Project Plan6
  2. Task Assignment & Roles7
  3. Risk Assessment & Mitigation Plan8
  4. KPIs (Key Performance Indicators)8

1. Literature Review9
   1. Feedback & Evaluation9
   2. Suggested Improvements9
   3. Final Grading Criteria9
2. Requirements Gathering 10
   1. Stakeholder Analysis10
   2. User Stories & Use Cases10
   3. Functional Requirements11
   4. Non-functional Requirements11
3. System Analysis & Design 19
   1. Problem Statement & Objectives20
   2. Database Design & Data Modeling21
   3. Data Flow & System Behavior22
   4. UI/UX Design & Prototyping23
   5. System Deployment & Integration24
   6. Additional Deliverables25

**Project Planning & Management**

* 1. **Project Proposal**

**Overview:**

The project aims to develop a comprehensive medical e-commerce and management application using React, React-Bootstrap, MongoDB, and Context API. This application will serve both administrators and customers, offering a seamless experience for managing medical products, handling online and cash payments, and overseeing transactions efficiently**.**

**Objectives:**

- Develop a medical e-commerce platform where customers can browse and purchase medical products.

- Implement a dashboard for administrators to track orders, manage products, and monitor payments.

- Enable secure and efficient online and cash payment options, with all transactions recorded in the admin panel.

- Ensure a scalable architecture to accommodate future expansions in features and services.

**Scope:**

The system will have two main interfaces:

1. Customer Portal:

- Browse and purchase medical products.

- Choose between online and cash payment methods.

- View order history and track order status.

2. Admin Dashboard:

- Manage product inventory, prices, and availability.

- Monitor customer orders and payment transactions.

- Generate reports for sales and customer activities.

* 1. **Project Plan**

**Timeline:**

**A diagram of application development

AI-generated content may be incorrect.**

**Milestones:**

- UI/UX design and database setup.

- Core functionality for product management and payments.

- Successful testing of transactions and admin controls.

- Deployment and final documentation submission.

**Deliverables:**

- A fully functional medical e-commerce system with admin and customer interfaces.

- Secure payment system with transaction tracking.

- A comprehensive admin dashboard for order and inventory management.

- Detailed Documentation & Presentation covering technical and user aspects.

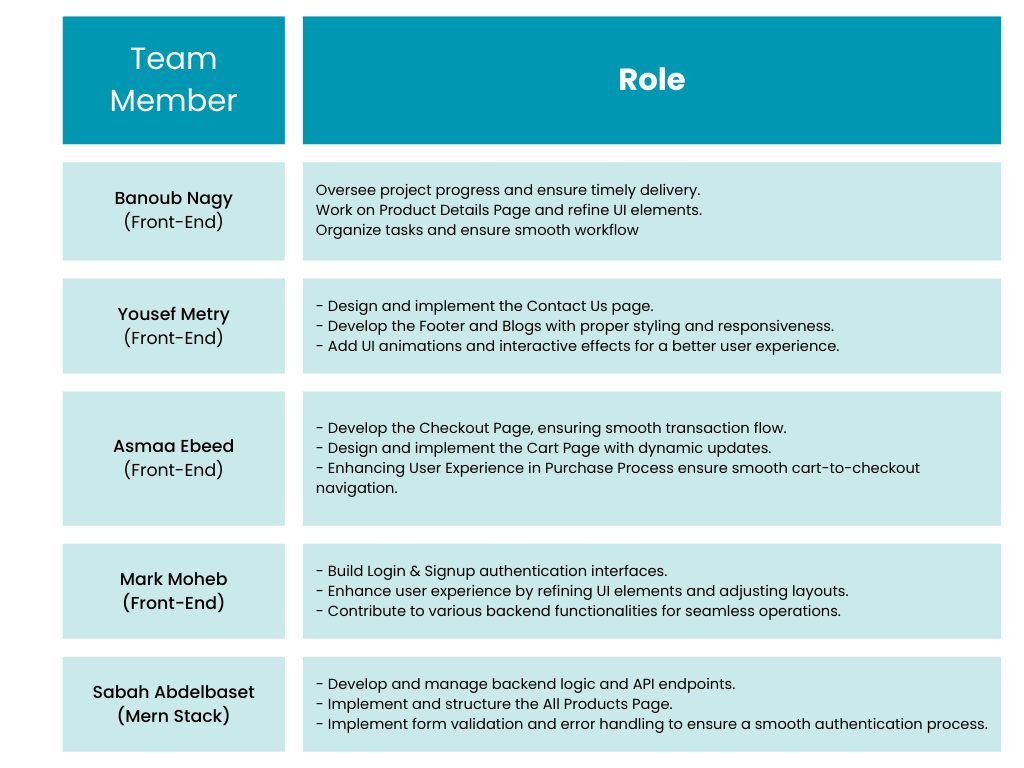
**Resource Allocation:**

- Development Team: Frontend Devs, Backend Devs, UI/UX .

- Tech Stack: React, React-Bootstrap, MongoDB, Node.js, Context API, Payment Gateway.

* 1. **Task Assignment & Roles**

**Responsibilities for team members:**

****

* 1. **Risk Assessment & Mitigation Plan**

**Delays in Development:**

Define clear milestones and conduct regular progress reviews.

Allocate buffer time in the project timeline.

* 1. **KPIs (Key Performance Indicators):**

1. System Uptime (%) – Ensures system availability and reliability. Target: 99.9% uptime.

2. Response Time (ms) – Measures how quickly the system responds to user requests. Target: < 500ms.

3. Order Processing Time (minutes) – Tracks the average time taken from order placement to confirmation.

4. User Adoption Rate (%) – Percentage of new users actively engaging with the platform within the first month.

5. Customer Satisfaction Score (1-5) – Based on feedback and reviews from users.

6. Security Incidents (count) – Number of security breaches or vulnerabilities detected and resolved.

7. Scalability & Load Handling – Performance of the system under peak usage conditions.

**Literature Review**

* 1. **Feedback & Evaluation**
  2. **Suggested Improvements**

**Enhanced User Review System**

Implement a structured review format with rating categories (e.g., delivery speed, product quality, user experience) and Allow users to submit detailed feedback with images to improve credibility.

**Real-Time Feedback Collection & Automated Resolution**

Add quick survey prompts after order completion for immediate insights.

Develop an automated response system to handle complaints efficiently and improve service.

**Advanced Payment & Order Management**

Implement multi-currency support to make the platform scalable for global users.

Introduce a refund & cancellation system to enhance customer trust and flexibility.

**Admin Dashboard & Analytics Enhancements**

Add visual analytics (charts & reports) for tracking sales, user activity, and order status.

Enable pharmacists & doctors to validate medication accuracy, ensuring better service quality.

* 1. **Final Grading Criteria**

**Requirements Gathering**

**3.1 Stakeholder Analysis:**

**Customers (End-Users):** Need an easy-to-use platform to browse, purchase, and manage medical products with a seamless checkout experience.

**Admins & Pharmacists:** Require a dashboard to manage products, process orders, track payments, and validate prescriptions if needed.

**Business Owners:** Expect real-time analytics, sales tracking, and a secure payment system to ensure smooth operations.

**Delivery Personnel:** Need a system that provides clear order details, delivery addresses, and status updates

**3.2 User Stories & Use Cases:**

As a customer, I want to browse medical products, add them to my cart, and securely complete my purchase.

As a customer, I want to track my order status and receive real-time updates.

As an admin, I want to manage inventory, approve or reject orders, and monitor payment transactions.

As an admin, I want to access real-time sales reports and user activity analytics.

As a pharmacist, I want to validate prescription-based orders before approval.

**3.3 Functional Requirements :**

**User Authentication:** Login & registration system with JWT authentication.

**Product Management:** Ability to add, update, and remove medical products.

**Cart & Checkout System:** Customers can add products to the cart, apply discounts, and complete payments.

**Payment Integration:** Secure online payment (Stripe, PayPal) and cash-on-delivery options.

**Order Management:** Admins can view, approve, and track orders.

**Review & Feedback System:** Users can rate and review products and services.

**Analytics Dashboard:** Real-time reports on sales, order status, and user activity.

**3.4 Non-functional Requirements:**

**Performance:** API response time should be under 500ms for a smooth user experience.

**Security:** Data encryption, secure payment processing, and role-based access control (RBAC).

**Usability:** Intuitive UI/UX with mobile-friendly design and accessibility support.

**Reliability:** 99.9% system uptime with cloud deployment and backup strategies.

**System Analysis & Design**

**4.1 Problem Statement & Objectives:**

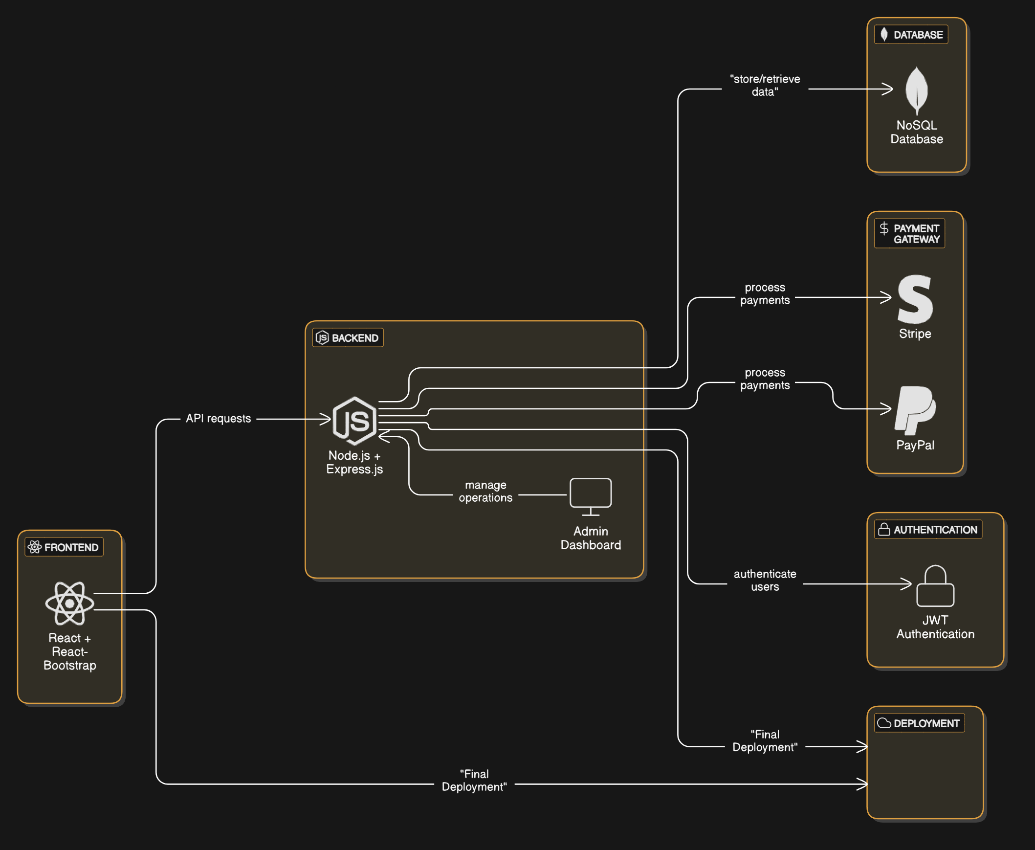
**Problem Statement:**

Managing online medical product sales and healthcare transactions can be challenging due to inefficient order processing, lack of real-time tracking, and limited digital payment options. This project aims to build a scalable, secure, and user-friendly medical e-commerce platform that streamlines online purchasing, payment, and admin management.

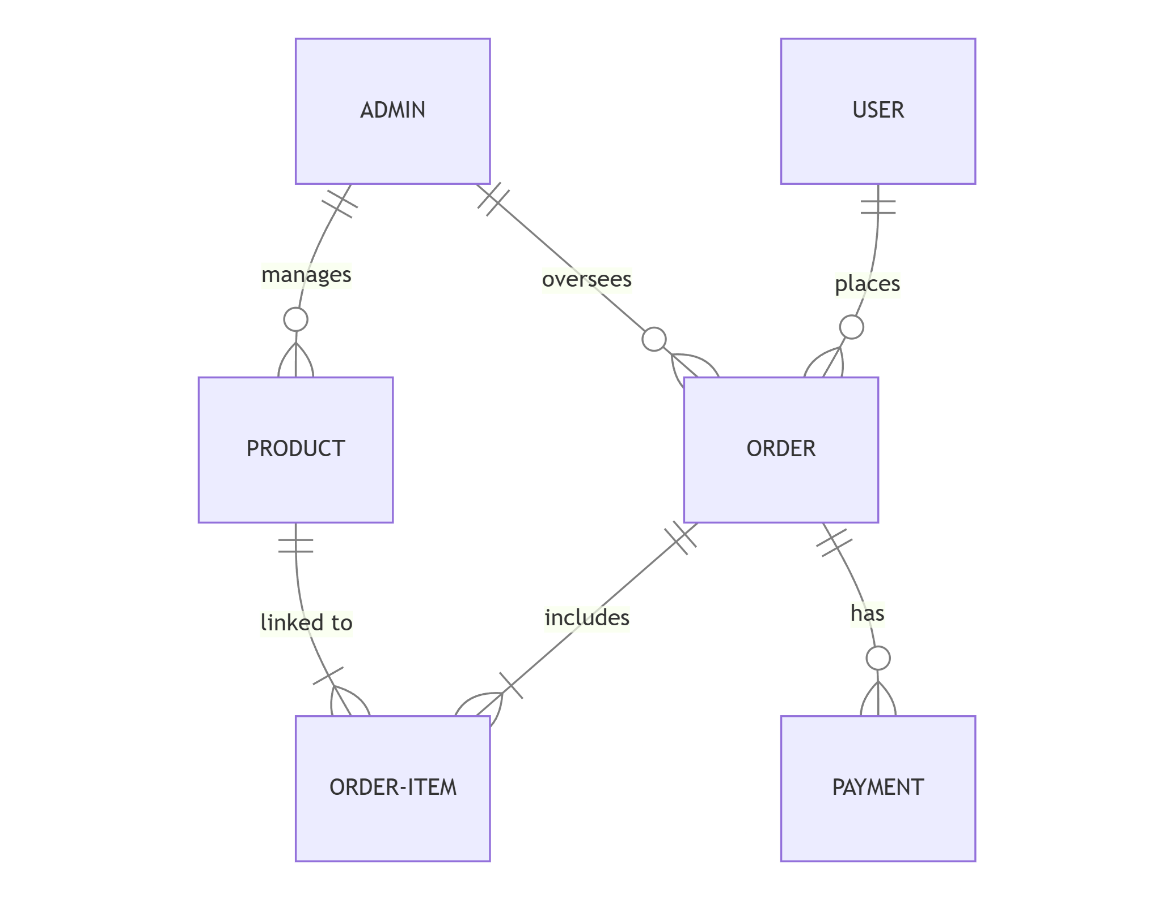
**A diagram of a diagram

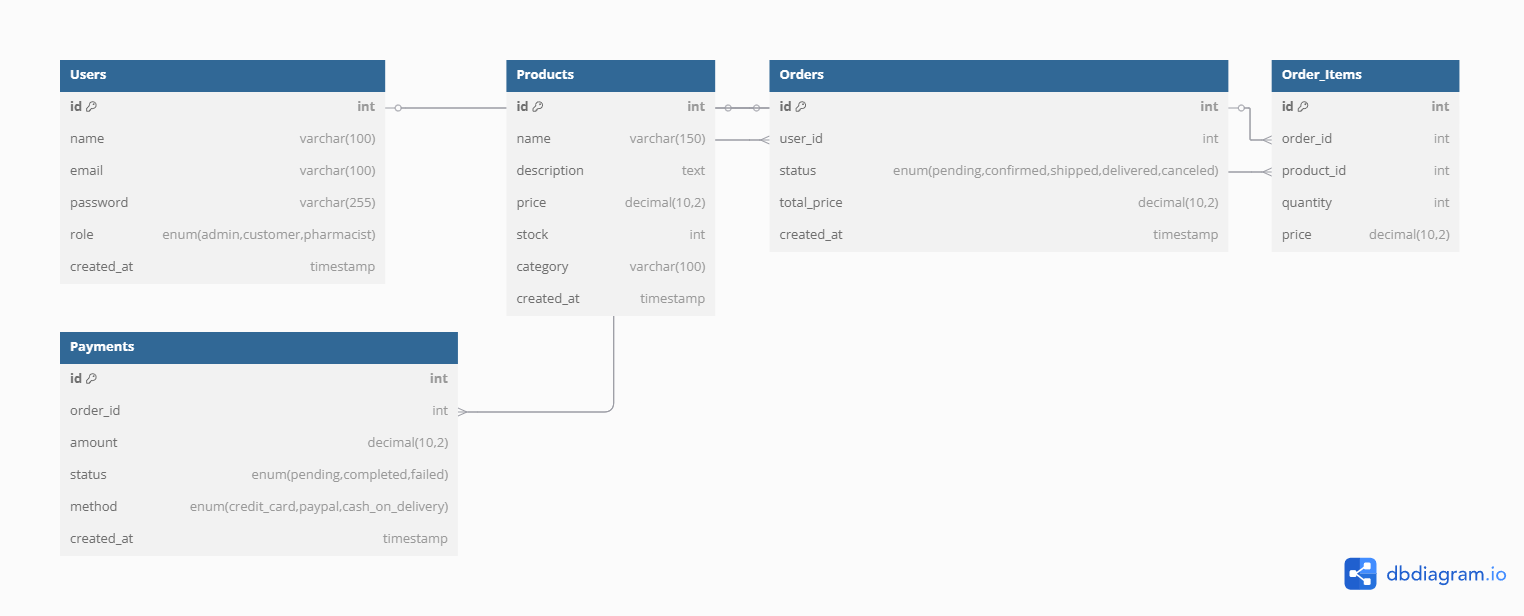
AI-generated content may be incorrect.**

**Software Architecture:**

****

**4.2 Database Design & Data Modeling:**

****

****

**4.3 Data Flow & System Behavior**

**Data Flow Diagram**

**A diagram of a product

AI-generated content may be incorrect.**

**Sequence Diagram**

**A diagram of a payment method

AI-generated content may be incorrect.**

**Activity Diagram:**

**A diagram of a diagram

AI-generated content may be incorrect.**

**State Diagram:**

**A diagram of a process

AI-generated content may be incorrect.**

**Class Diagram:**

**A diagram of a computer

AI-generated content may be incorrect.**

**4.4 UI/UX Design & Prototyping:**

**App Wireframes :** [**Care+**](https://drive.google.com/drive/folders/1bBrc7dthgeKVt1XYBlaqzEy5CJ-Lw20R?usp=sharing)

**Our Idea Based On** [**Here**](https://vitasana.qodeinteractive.com/)

**4.5** **System Deployment & Integration**

**Technology Stack:**

Frontend: React.js + React-Bootstrap

Backend: Node.js + Express.js

Database: MongoDB (NoSQL)

Payment: Stripe, Cash

Deployment: Vercel

**Deployment Diagram:**

The Client Device hosts the React App, which runs in the user's browser and interacts with the backend through API calls.

The Cloud Server houses the Node.js API and Express.js framework, handling business logic, authentication, and communication with the database.

MongoDB serves as the primary database, storing user data, orders, products, and transactions.

Payment Gateways (Stripe & PayPal) are integrated with the backend to handle secure payment processing.

The Admin Dashboard Server provides an interface for administrators to manage users, orders, and product inventory

A diagram of a server

AI-generated content may be incorrect.

**Component Diagram:**

**Frontend (React.js + React-Bootstrap):** The user-facing interface where customers can browse products, place orders, and track deliveries. It communicates with the backend via RESTful API requests.

**Backend (Node.js + Express.js):** The core logic of the system, responsible for handling user authentication, processing orders, managing products, and handling payment transactions.

**Database (MongoDB):** Stores all essential data, including user details, product catalogs, orders, and payment records. It ensures data consistency and efficient retrieval.

**Payment (Stripe/Cash):** Handles secure transactions and payment processing for orders.

**Admin Panel:** A dedicated interface for administrators to manage users, products, and monitor system activities.

A diagram of a medical application

AI-generated content may be incorrect.