Doctor Portal React App - Project Documentation

# Overview

This project is a Doctor Portal built using React.js, styled-components, and React Router. It features a structured, clean frontend with multiple reusable components and routing across 12 different pages.

# Tech Stack

- Frontend: React.js  
- Styling: styled-components  
- Routing: React Router DOM  
- State Management: useState, useEffect (React Hooks)

# Project Structure

Doctor-Portal-main/  
├── public/  
├── src/  
│ ├── assets/ # Static images or icons  
│ ├── components/ # Reusable UI components  
│ ├── pages/ # Individual page components (12 total)  
│ ├── App.js # Main App component  
│ ├── index.js # ReactDOM render  
│ ├── routes.js # Page routing definitions  
│ └── theme.js # styled-components theme config  
├── .gitignore  
├── package.json  
├── README.md  
└── yarn.lock / package-lock.json

# Pages Implemented (12 Pages)

* - Dashboard - Default landing page after login
* - Appointments - Shows upcoming and past appointments
* - Patients - List of patients
* - Messages - Chat/message functionality
* - Profile - Doctor's profile details
* - Settings - Portal and personal settings
* - Login - Auth gateway
* - Register - New account creation
* - Forgot Password - Recover credentials
* - Notifications - Alerts or updates for doctor
* - Help - FAQ / support page
* - Logout - Handles user sign out

# Key Features

* - Responsive UI
* - Client-side routing
* - Custom styled-components for consistency
* - Component-based structure for reusability
* - Modern, minimal design

# Setup Instructions

1. Clone the Repo

git clone https://github.com/[your-username]/Doctor-Portal-main.git

2. Install Dependencies

cd Doctor-Portal-main  
npm install

3. Run Locally

npm start

The app will run on: http://localhost:3000

# Hosting

If hosted using Vercel, the hosted link can be found here:  
🔗 [Live App](https://your-vercel-link.vercel.app)

# Submission Guidelines

Please share your GitHub link, YT video, and any hosted version link. Include relevant documentation inside your repo or in a separate file.