# **MedConnect Installation and Testing Instructions**

(You can find the same in readme.md and installation.md, this is just a doc version)

Prerequisites:

- Node.js (version 14 or higher)

- npm (comes with Node.js)

- MongoDB (for the backend)

Installation Steps:

1. Clone the Repository:

git clone https://github.com/Gous0786/medconnect.git

cd medconnect

Backend Setup:

1. Navigate to the Backend Directory:

cd medconnect-backend

2. Install Backend Dependencies:

npm install

3. Environment Variables:

Create a .env file in the medconnect-backend directory and add your environment variables:

PORT=5000

MONGODB\_URI=your\_mongodb\_connection\_string

GEMINI\_API\_KEY=your\_gemini\_api\_key

4. Run the Backend:

npm run dev

The server should start on port 5000 (or the port specified in your .env file).

Frontend Setup:

1. Navigate to the Frontend Directory:

cd medconnect-frontend

2. Install Frontend Dependencies:

npm install

3. Run the Frontend:

npm start

The application should open on port 3000 (or the port specified in your .env file).

Verifying the Installation:

- The frontend should be accessible at: http://localhost:3000

- The backend API should be accessible at: http://localhost:5000/api

Testing Instructions:

Backend Testing:

1. Navigate to the Backend Directory:

cd medconnect-backend

2. Install Testing Dependencies:

npm install --save-dev jest supertest

3. Set Up Environment Variables:

Before running the integration tests, make sure to add your Gemini API key to the .env file:

GOOGLE\_API\_KEY=your\_gemini\_api\_key

4. Run Tests:

npm test

Frontend Testing:

1. Navigate to the Frontend Directory:

cd medconnect-frontend

2. Install Testing Dependencies:

npm install --save-dev @testing-library/react @testing-library/jest-dom

3. Run Tests:

npm test

Additional Setup Options:

- To seed the database with sample data, run:

npm run seed