

Code Documentation

Penguin Healthcare Solutions — Project Documentation

This document provides complete setup, installation, and run instructions for the project across macOS, Windows, and Linux. It also documents how the frontend connects to the backend server.

Clone from GitHub

You can clone the repository using HTTPS, SSH, or GitHub CLI.

HTTPS

```
```bash
git clone https://github.com/ddulmai1/penguin-healthcare-solutions.git
cd penguin-healthcare-solutions
````
```

GitHub CLI

```
```bash
gh repo clone ddulmai1/penguin-healthcare-solutions
cd penguin-healthcare-solutions
````
```

Code Contents

- Full source code available in this repository under:
 - `src/` (frontend and backend source)
 - `bin/` (utility scripts)
- Setup script: `bin/setup.sh` (macOS/Linux)

Architecture Overview

- Backend: Spring Boot REST API (Java)
- Frontend: React

Directory Structure (Key Files)

```
penguin-healthcare-solutions/
├── build.gradle
├── settings.gradle
├── gradlew / gradlew.bat
├── package.json
├── bin/
│   └── setup.sh
├── docs/
│   └── Documentation.md
├── public/
│   └── index.html
└── src/
    ├── main/
    │   ├── java/com/SWE_Team_Penguin/Penguin_Healthcare_Solutions/
    │   │   ├── PenguinHealthcareSolutionsApplication.java
    │   │   ├── controller/
    │   │   ├── model/
    │   │   ├── repositories/
    │   │   └── services/
    │   └── resources/
    │       └── application.properties
    └── test/
        └── java/com/SWE_Team_Penguin/Penguin_Healthcare_Solutions/
            ├── components/
            │   ├── AppointmentScheduler.jsx
            │   ├── AdmissionsNotifications.jsx
            │   ├── PrescriptionForm.jsx
            │   ├── PatientRecordView.jsx
            │   ├── UpdatePatientRecord.jsx
            └── dashboard.jsx
```

```
|   |   └── LoginForm.jsx
|   └── services/
|       ├── appointmentsService.js
|       ├── prescriptionsService.js
|       ├── admissionsService.js
|       └── authService.js
|   └── data/
|       └── mockPatientData.js
└── App.jsx
└── index.js
└── gradle/wrapper/gradle-wrapper.properties
...
``
```

Prerequisites

- Java JDK 17+ (OpenJDK recommended)
- Gradle Wrapper (included: `gradlew` / `gradlew.bat`)
- Node.js 18+ and npm

macOS (zsh)

- Optional installers via Homebrew:

```
```bash
brew install openjdk@17
brew install node
...
``
```

- One-time setup:

```
```bash
chmod +x bin/setup.sh
./bin/setup.sh
...
``
```

- Run backend:

```
```bash
./gradlew bootRun
```

```

- Run frontend:

```
```bash
npm install # or npm ci
npm run start
```

```

- URLs:

- Backend: `http://localhost:8080`
- Frontend: `http://localhost:3000`

Windows (PowerShell)

- Install JDK 17+ (Temurin/Adoptium or Oracle) and add `JAVA_HOME`/PATH.
- Install Node.js 18+ from nodejs.org.
- Verify tools:

```
```powershell
java -version
node -v
npm -v
```

```

- Install frontend deps and run:

```
```powershell
npm ci # or npm install
npm run start
```

```

- Run backend:

```
```powershell
./gradlew.bat bootRun
```

```

- URLs:

- Backend: `http://localhost:8080`
- Frontend: `http://localhost:3000`

Linux (bash)

- Install JDK 17+ and Node.js via your distro:

```
```bash
```

```
sudo apt update && sudo apt install -y openjdk-17-jdk nodejs npm # Debian/Ubuntu
```

or

```
sudo dnf install -y java-17-openjdk nodejs npm # Fedora
```

...

- Verify tools:

```
```bash
```

```
java -version
```

```
node -v
```

```
npm -v
```

...

- Install deps and run:

```
```bash
```

```
npm ci # or npm install
```

```
npm run start
```

...

- Run backend:

```
```bash
```

```
./gradlew bootRun
```

...

- URLs:

- Backend: `http://localhost:8080`
- Frontend: `http://localhost:3000`

Connecting Frontend to Backend

- Frontend services call the backend at `http://localhost:8080` (see `src/services/*.js`).
- Start the backend before using routes that require authentication or server data.

Demo Mode (No Backend Required)

- Demo routes allow testing without the backend:
 - `http://localhost:3000/patient-record-demo`
 - `http://localhost:3000/admissions-demo`
- Demo mode uses mock data and bypasses authentication.

Available Routes

- `/' — Login (backend required)
- `/dashboard` — App dashboard
- `/patient-record` — Patient record viewer (backend required)
- `/patient-record-demo` — Patient record viewer (demo)
- `/admissions` — ADT submission (backend required)
- `/admissions-demo` — ADT submission (demo)
- `/update-patient` — Update medical information (backend required)
- `/update-patient-demo/P001` (also P002, P003) — Update medical information (demo)
- `/appointments` — Schedule/manage appointments (backend required)
- `/appointments-demo` — Appointments (demo)
- `/prescriptions` — Create e-prescriptions (backend required)
- `/prescriptions-demo` — Prescriptions (demo)

Test Credentials

- Admin: `admin` / `12345`
- Clinician: `Rita` / `pass`

Demo :

<https://docs.google.com/videos/d/1pA4R4Pp4tKcPJvseUCcZls805RhUSwbjCT2D2fRVjP0/edit?usp=sharing>