## **Estrutura de pastas e arquivos do projeto Agendei – WEB**

AgendeiJS-WEB/

├── public/

│ ├── favicon.ico

│ ├── index.html

│ ├── logo.png

│ └── manifest.json

├── src/

│ ├── assets/

│ │ ├── images/

│ │ │ ├── logo.png

│ │ │ ├── banner.jpg

│ │ │ └── avatar.png

│ │ ├── styles/

│ │ │ ├── global.css

│ │ │ ├── buttons.css

│ │ │ └── forms.css

│ │ └── icons/

│ │ ├── calendar.svg

│ │ ├── user.svg

│ │ └── doctor.svg

│ ├── components/

│ │ ├── Button.jsx

│ │ ├── InputField.jsx

│ │ ├── Modal.jsx

│ │ ├── Navbar.jsx

│ │ ├── Sidebar.jsx

│ │ └── Table.jsx

│ ├── pages/

│ │ ├── appointment-add/

│ │ │ ├── AppointmentAdd.jsx

│ │ │ └── styles.css

│ │ ├── appointments/

│ │ │ ├── AppointmentsList.jsx

│ │ │ ├── AppointmentDetails.jsx

│ │ │ └── styles.css

│ │ ├── doctors-add/

│ │ │ ├── DoctorAdd.jsx

│ │ │ └── styles.css

│ │ ├── doctors/

│ │ │ ├── DoctorsList.jsx

│ │ │ ├── DoctorProfile.jsx

│ │ │ └── styles.css

│ │ ├── login/

│ │ │ ├── Login.jsx

│ │ │ ├── LoginForm.jsx

│ │ │ └── styles.css

│ │ ├── register/

│ │ │ ├── Register.jsx

│ │ │ ├── RegisterForm.jsx

│ │ │ └── styles.css

│ │ ├── users-add/

│ │ │ ├── UserAdd.jsx

│ │ │ └── styles.css

│ │ ├── users/

│ │ │ ├── UsersList.jsx

│ │ │ ├── UserProfile.jsx

│ │ │ └── styles.css

│ ├── services/

│ │ ├── api.js

│ │ ├── authService.js

│ │ ├── appointmentService.js

│ │ ├── doctorService.js

│ │ └── userService.js

│ ├── utils/

│ │ ├── helpers.js

│ │ ├── validators.js

│ │ └── constants.js

│ ├── App.jsx

│ ├── main.jsx

│ └── routes.jsx

├── .gitignore

├── package.json

├── package-lock.json

├── README.md

├── vite.config.js

└── jsconfig.json

**📌 Estrutura das Páginas e Conexões**

**1️⃣ Autenticação: login/ e register/**

Essas páginas são a porta de entrada do sistema.

* **login/**
  + **Login.jsx**: Página principal do login.
  + **LoginForm.jsx**: Componente que contém o formulário de autenticação.
  + **Conexões:**
    - Ao fazer login, o usuário é redirecionado para a página principal (appointments/).
    - Se não tem conta, pode navegar para register/.
* **register/**
  + **Register.jsx**: Tela principal do registro.
  + **RegisterForm.jsx**: Contém o formulário de cadastro.
  + **Conexões:**
    - Após o cadastro, o usuário pode ser redirecionado automaticamente para login/ ou já ser autenticado e ir para appointments/.

**2️⃣ Agendamentos: appointments/ e appointment-add/**

Essas páginas permitem visualizar, criar e gerenciar agendamentos.

* **appointments/**
  + **AppointmentsList.jsx**: Lista de todos os agendamentos.
  + **AppointmentDetails.jsx**: Exibe os detalhes de um agendamento específico.
  + **Conexões:**
    - Um clique em um item da lista leva para AppointmentDetails.jsx.
    - O botão "Novo Agendamento" leva para appointment-add/.
* **appointment-add/**
  + **AppointmentAdd.jsx**: Página onde o usuário preenche um formulário para adicionar um novo agendamento.
  + **Conexões:**
    - Após a criação, o usuário pode ser redirecionado de volta para appointments/.

**3️⃣ Médicos: doctors/ e doctors-add/**

Gerencia os médicos cadastrados no sistema.

* **doctors/**
  + **DoctorsList.jsx**: Exibe todos os médicos registrados.
  + **DoctorProfile.jsx**: Mostra informações detalhadas de um médico.
  + **Conexões:**
    - Clicar em um médico na lista leva para DoctorProfile.jsx.
    - O botão "Adicionar Médico" leva para doctors-add/.
* **doctors-add/**
  + **DoctorAdd.jsx**: Formulário para adicionar um novo médico ao sistema.
  + **Conexões:**
    - Após adicionar, o usuário pode ser redirecionado para doctors/.

**4️⃣ Usuários: users/ e users-add/**

Controla os usuários do sistema (possivelmente pacientes ou administradores).

* **users/**
  + **UsersList.jsx**: Lista os usuários cadastrados no sistema.
  + **UserProfile.jsx**: Exibe detalhes de um usuário específico.
  + **Conexões:**
    - Clicar em um usuário na lista leva para UserProfile.jsx.
    - O botão "Adicionar Usuário" leva para users-add/.
* **users-add/**
  + **UserAdd.jsx**: Formulário para cadastrar um novo usuário.
  + **Conexões:**
    - Após cadastrar, o usuário pode ser redirecionado para users/.

**🔗 Fluxo Geral da Aplicação**

1. O usuário acessa **login/** para autenticação.
2. Se necessário, pode se cadastrar em **register/**.
3. Após login, ele é redirecionado para **appointments/** (tela inicial).
4. A partir daí, pode:
   * Criar um novo agendamento em **appointment-add/**.
   * Acessar detalhes de um agendamento em **AppointmentDetails.jsx**.
   * Gerenciar médicos em **doctors/** e **doctors-add/**.
   * Gerenciar usuários em **users/** e **users-add/**.
5. Caso precise sair, pode se deslogar e voltar para **login/**.