

**Presented by Brian Park,
Rayhan Biju,
& Sabrina Hart**

LEAFRX

AGENDA

- Project Overview
- Front End Overview
- Back End Overview
- Demo
- Challenges and Lesson's Learned
- Closing Thoughts





LEAFRX'S MAIN SERVICE IS TO PROVIDE PATIENTS, THAT ARE INTERESTED IN MEDICAL CANNABIS, A WAY TO SCHEDULE A TELEHEALTH APP WITH A PROVIDER.

LEAFRX'S GOAL IS TO ENSURE PATIENT DATA SECURITY, PROVIDE AN EASY TO USE APPLICATION FOR BOTH PATIENT AND PROVIDER, AND TO MAKE SCHEDULING APPOINTMENTS EASY.

LEAFRX MISSION

Goals

For our project, our goal is to create a HIPAA Compliant scheduling web based application.

HIPAA (Health Insurance Portability and Accountability Act) is a U.S. law that protects people's private health information from being shared without their consent. It sets rules for how doctors, clinics, and apps must securely handle and store patient data.



System Overview



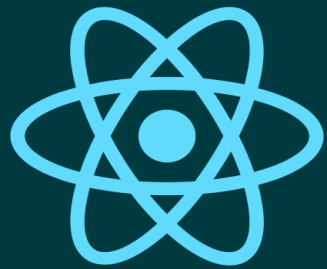
PATIENT

A PATIENT WILL ENTER THEIR MEDICAL HISTORY, THEN THEY SELECT A DATE AND TIME FOR THEIR APPOINTMENT, AND FINALLY BE PROMPTED TO PAY FOR THEIR FUTURE VISIT.



PROVIDER

A PROVIDER WILL BE ABLE TO SEE THEIR FULL SCHEDULE AND WHO HAS CHOSEN THE RESPECTIVE TIME SLOT AND THEN VIEW THE PATIENT'S MEDICAL HISTORY. ADDITIONALLY, THEY PROVIDE SELECT TIMES THAT ARE ABLE TO WORK.



React



Flask

Front End

React, CSS, & JavaScript



Back End

Flask

Database

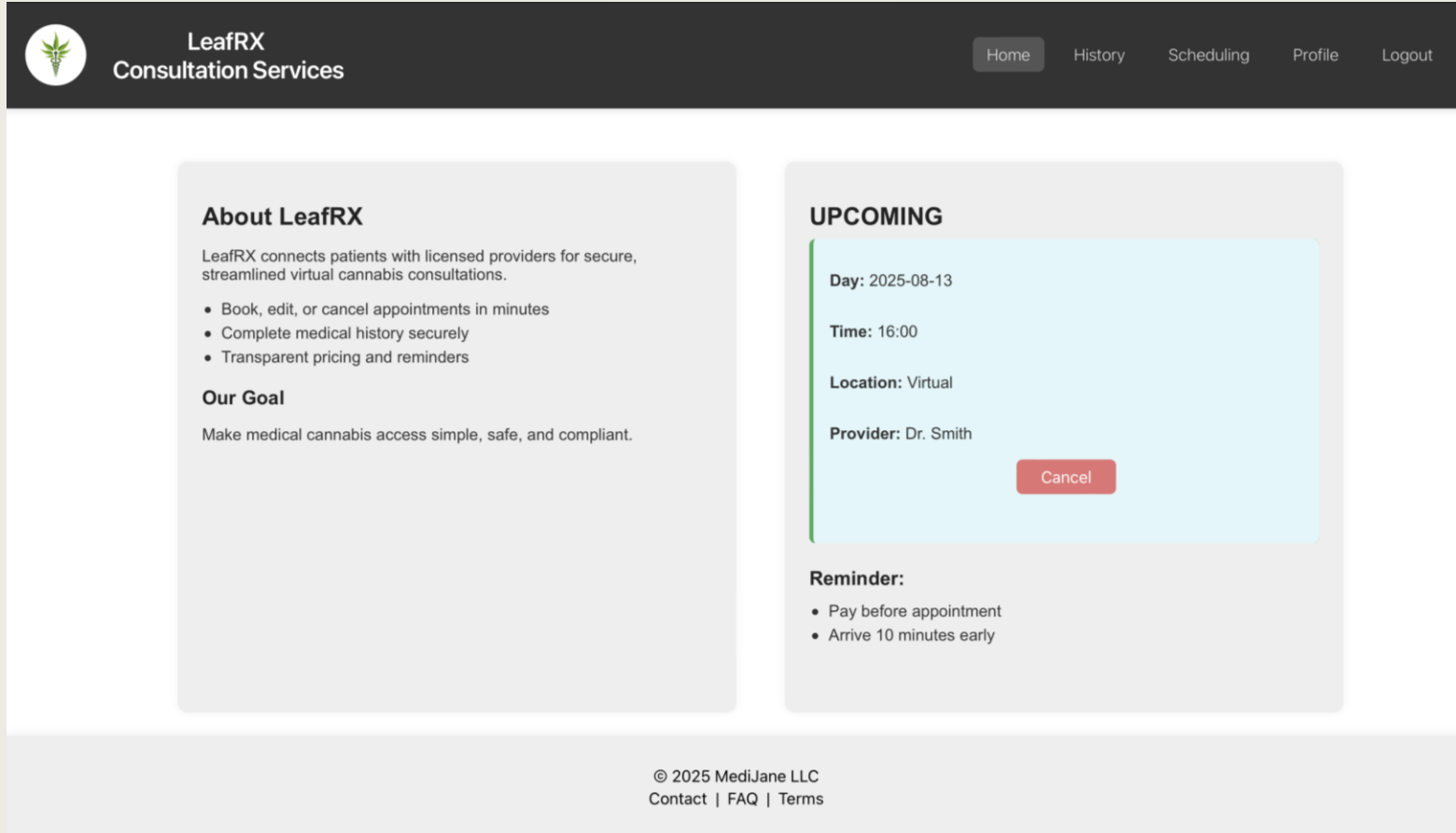
PostgreSQL

TECHNICAL
ARCHITECTURE

Payment

Stripe

FRONT END IMPLEMENTATION



- **React Component Architecture:** Built modular, reusable components.
- **State Management & Routing:** Used React Router for page navigation and component rendering, and used state managed hooks.
- **UI/UX Enhancements:** Implemented design with CSS.

Figure 1: React Example

BACK END DESIGN

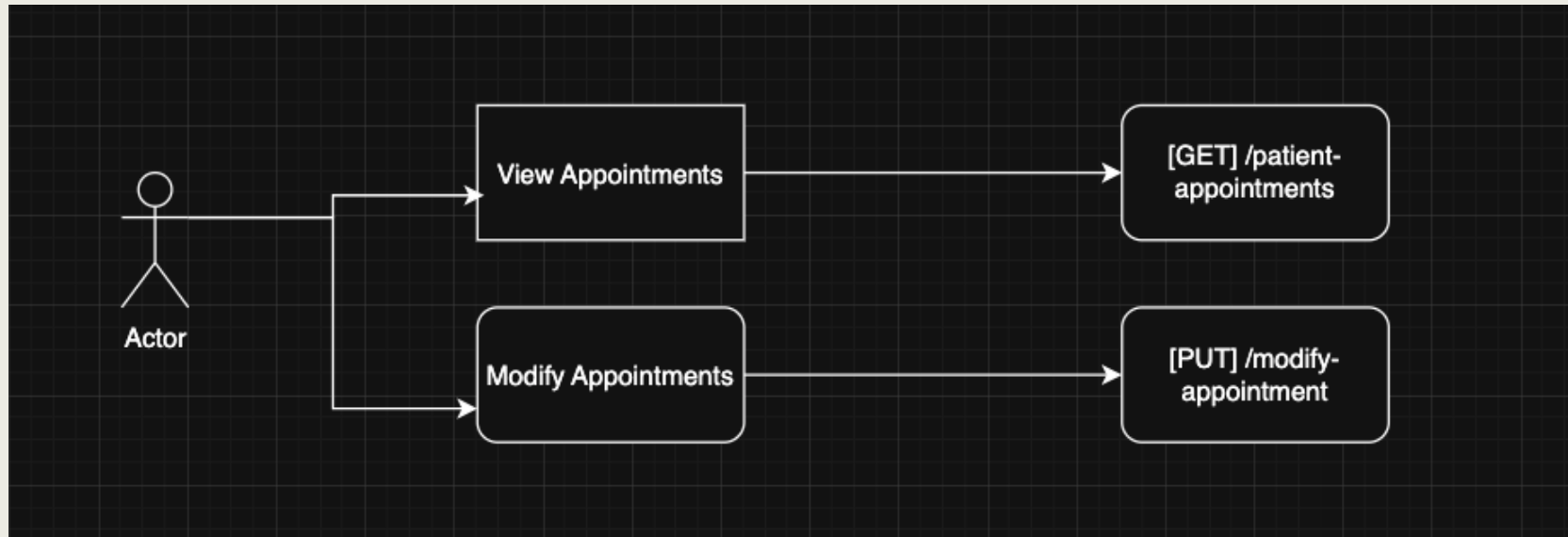


Figure 2: Example API Endpoint Diagram

HIPPA IMPLEMENTATION

- **HIPAA-Compliant Data Security:** Implemented encryption for Protected Health Information (PHI) at rest using the cryptography library's Fernet class (AES-128 in CBC mode with HMAC), securing emails, phone numbers, medical history, and appointment details.
- **Hashed Indexing:** Applied hashing to email addresses to allow efficient database lookups without storing raw values, further protecting sensitive identifiers.
- **Secure Data Flow:** Combined encryption with authentication mechanisms to ensure confidentiality and integrity of PHI at the database layer and during storage, aligning with HIPAA's Security Rule.

BACKEND IMPLEMENTATION

- **Flask Application Architecture:** Structured with Flask to handle RESTful API endpoints for authentication, patient management, appointment scheduling, and payment integration, using decorators for role-based access control.
- **PostgreSQL Integration via SQLAlchemy:** Configured SQLAlchemy ORM with PostgreSQL to define and manage models
- **Authentication:** Implemented authentication flow and state management via JWT tokens.
- **Stripe:** Integrated into the backend to securely process patient appointment payments through a third-party payment API after booking.

DEMO!

DEVELOPMENT PROCESS

- **Agile Methodology:** Followed an iterative sprint-based workflow (Sprints 3–8), with defined goals, backlog tracking, and regular sprint reviews to adapt features based on feedback.
- **Collaborative Development:** Coordinated frontend, backend, and database work through GitHub repositories, using pull requests and code reviews to maintain code quality.
- **Incremental Feature Delivery:** Prioritized high-impact features each sprint, beginning with core scheduling and registration flows, then progressively integrating payments, encryption, and UI/UX enhancements.





CHALLENGES

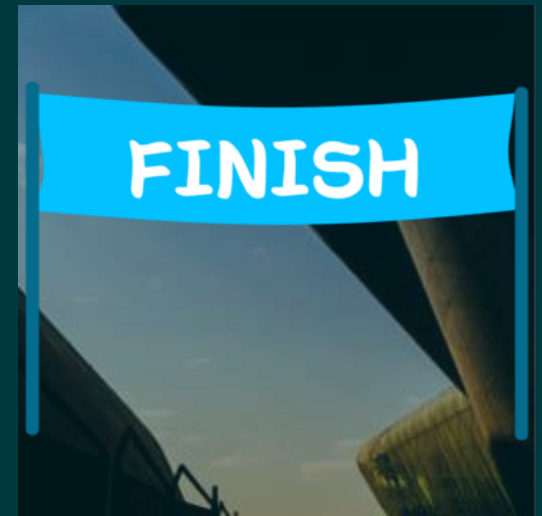
- Prioritizing time being spent on the old sponsor project for the first couple of weeks
- All of us work full time, so it was hard to find time
- Some of us are inexperienced in full stack, so there was a slight learning curve

LESSONS LEARNED

- GATHER REQUIREMENTS AS EARLY AS POSSIBLE
- WORK ON THE MAIN COMPONENTS AND FUNCTIONALITY FIRST
- GIVE AT LEAST TWO SPRINTS FOR BUG FIXES AND CLEANING UP
- COMMUNICATION WITH EACH OTHER IS KEY
- BE FLEXIBLE AND WILLING TO LEARN NEW THINGS

IF WE HAD MORE TIME...

- Rescheduling appointments
- Mass scheduling for provider
- Messaging system between client and admins
- More admin functionality
- Implementing a virtual consultation system



Q&A!
Thank you!

UI: https://github.com/ssabrinahart/medicine_ui
Backend: <https://github.com/rayb26/MediJane-Wix-Site>