**1. Problem Identification**

The need for immediate, accurate medical guidance and efficient management of first-aid supplies in emergencies is crucial. Traditional first-aid solutions lack the integration of smart technology, leading to missed opportunities for timely and informed assistance.

**2. Solution Overview**

SmartMedKit is an innovative IoT-enabled first-aid kit equipped with real-time tracking and AI-based analysis features. It provides instant medical advice through a user-friendly web application, revolutionizing the way first-aid assistance is administered.

**3. Existing Alternatives**

Conventional first-aid kits are static and lack interactivity, while standalone medical apps do not offer integration with physical medical supplies. These limitations hinder effective emergency response.

**4. User Demographics**

SmartMedKit is designed for a broad audience, including households, educational institutions, and workplaces. It serves anyone seeking an efficient, smart emergency response tool.

**5. Features and User Flows – Updated With Implemented Functionality**

**Inventory Management:**

Automated tracking of supplies via weight sensors, ensuring items are always stocked.

**AI Analysis:**

AI model using a fine-tuned VGG16 neural network to accurately identify burns and cuts from images.

**User Interface:**

A simple and intuitive web app allows users to easily navigate through guidance options and supply management. Including scannable QR code to provide quick access to the AI model to identify an injury.

**Admin Dashboard:**

Enables administrators to monitor the kit's status, manage inventory, and review statistics about the kit.

**Automated Alerts:**

The Celery notification system asynchronously monitors kit stock levels and sends email alerts to administrators when thresholds are crossed. Redis is used as a message broker to efficiently queue and distribute these notification tasks to Celery workers.

**6. Dependencies**

SmartMedKit's functionality relies on a stable internet connection, compatibility with a range of mobile devices, access to medical image datasets for AI analysis, and the seamless operation of integrated sensors.

**7. Technology Stack**

Backend: Python, Flask

Frontend: JavaScript, React, Bootstrap

Database: SQLite, Redis

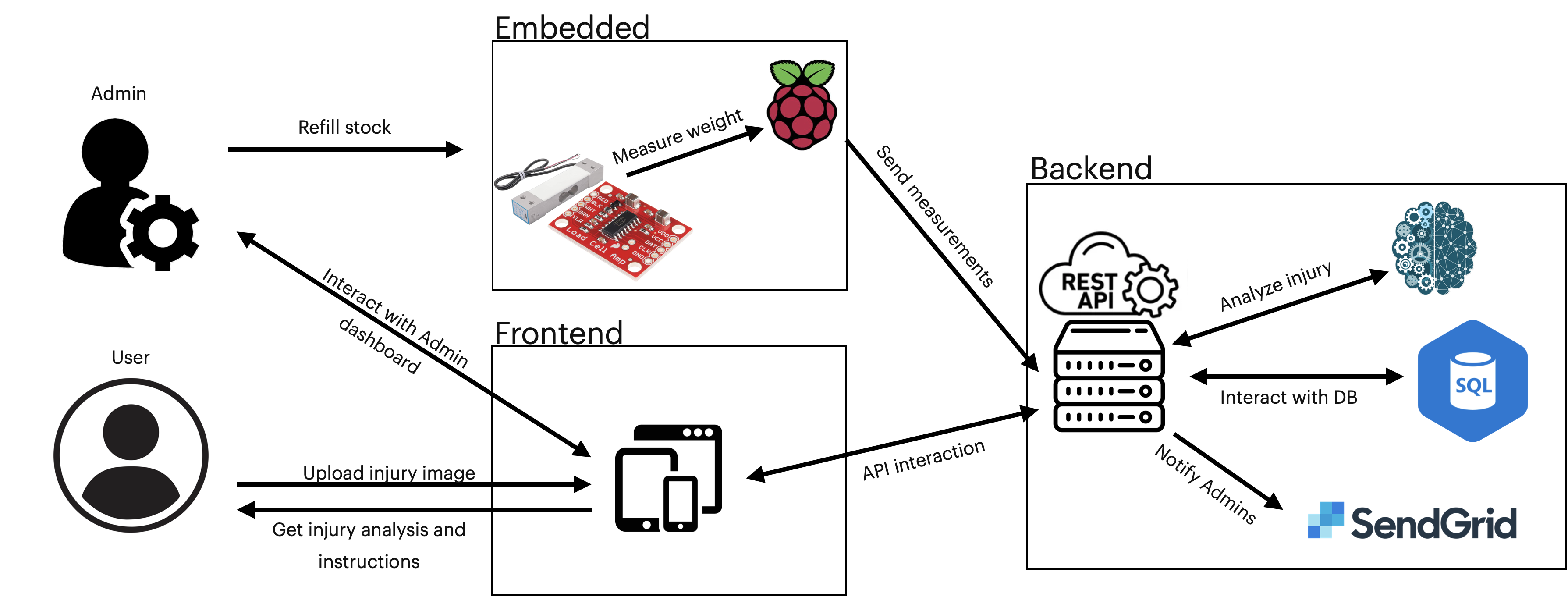
Machine Learning: TensorFlow, OpenCV, VGG16 fine-tuned model

Containerization: Docker

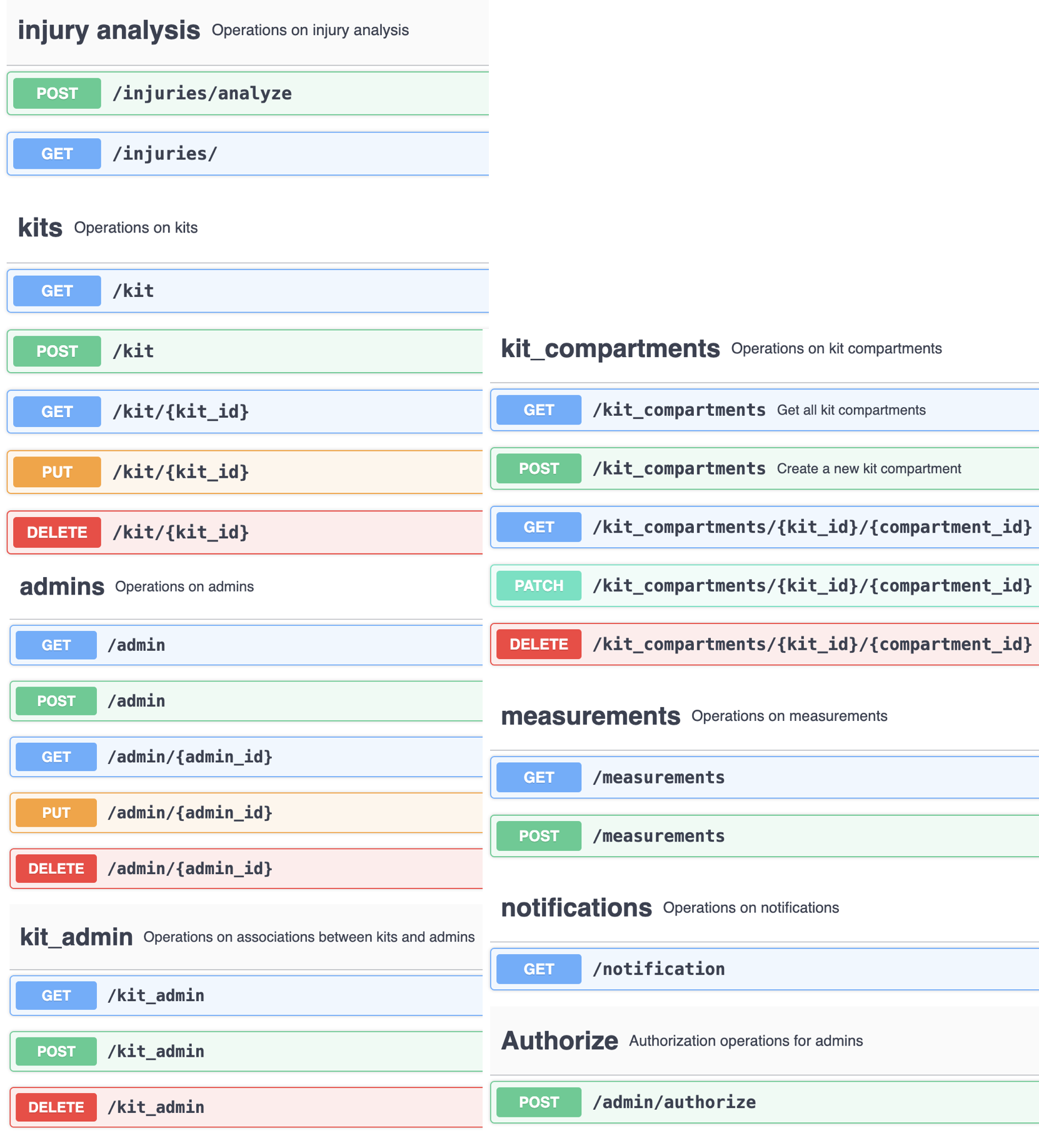
Cloud Services: Google Cloud

Hardware: Raspberry Pi OS, weight sensors with HX711 Amps

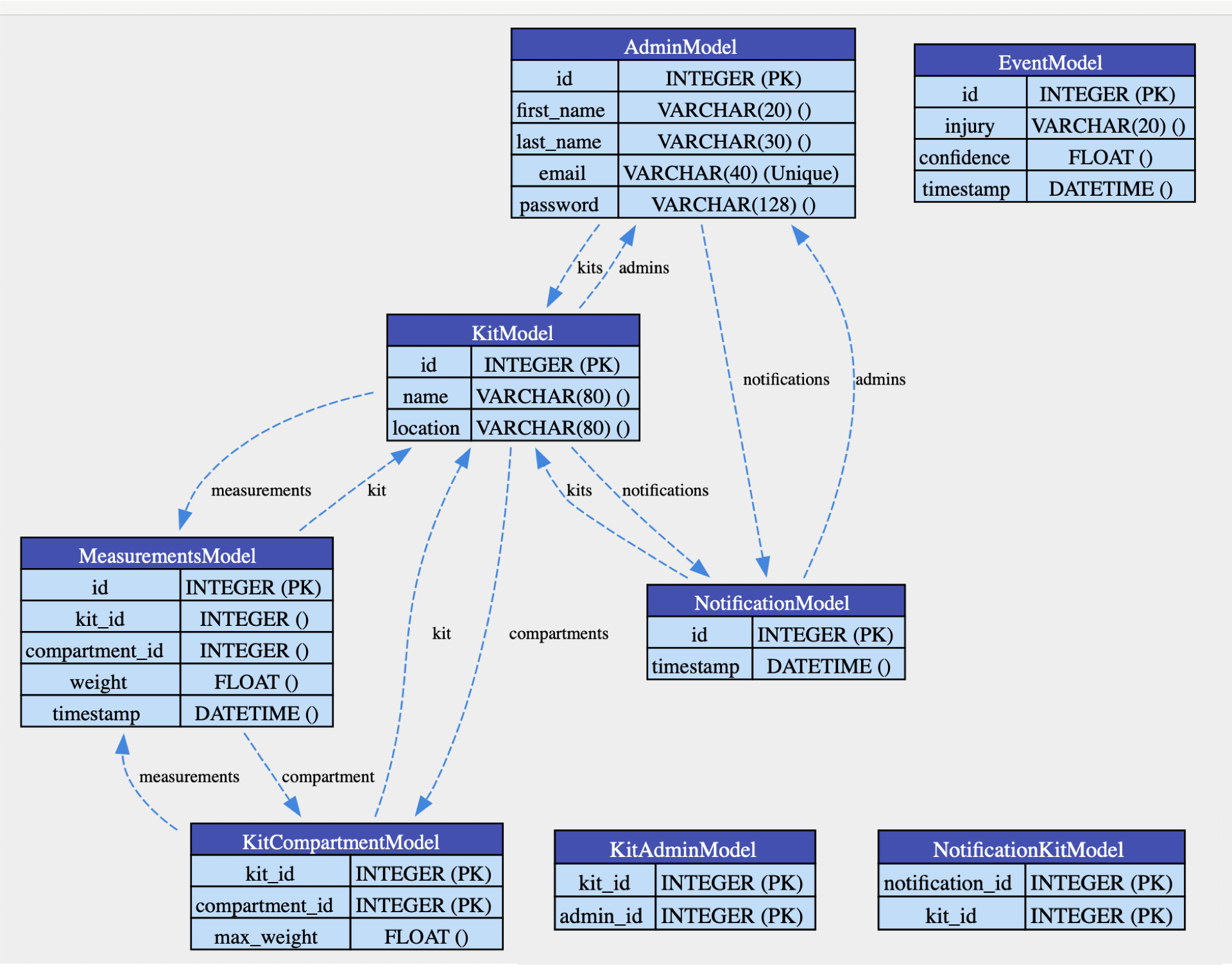
**Project Architecture**



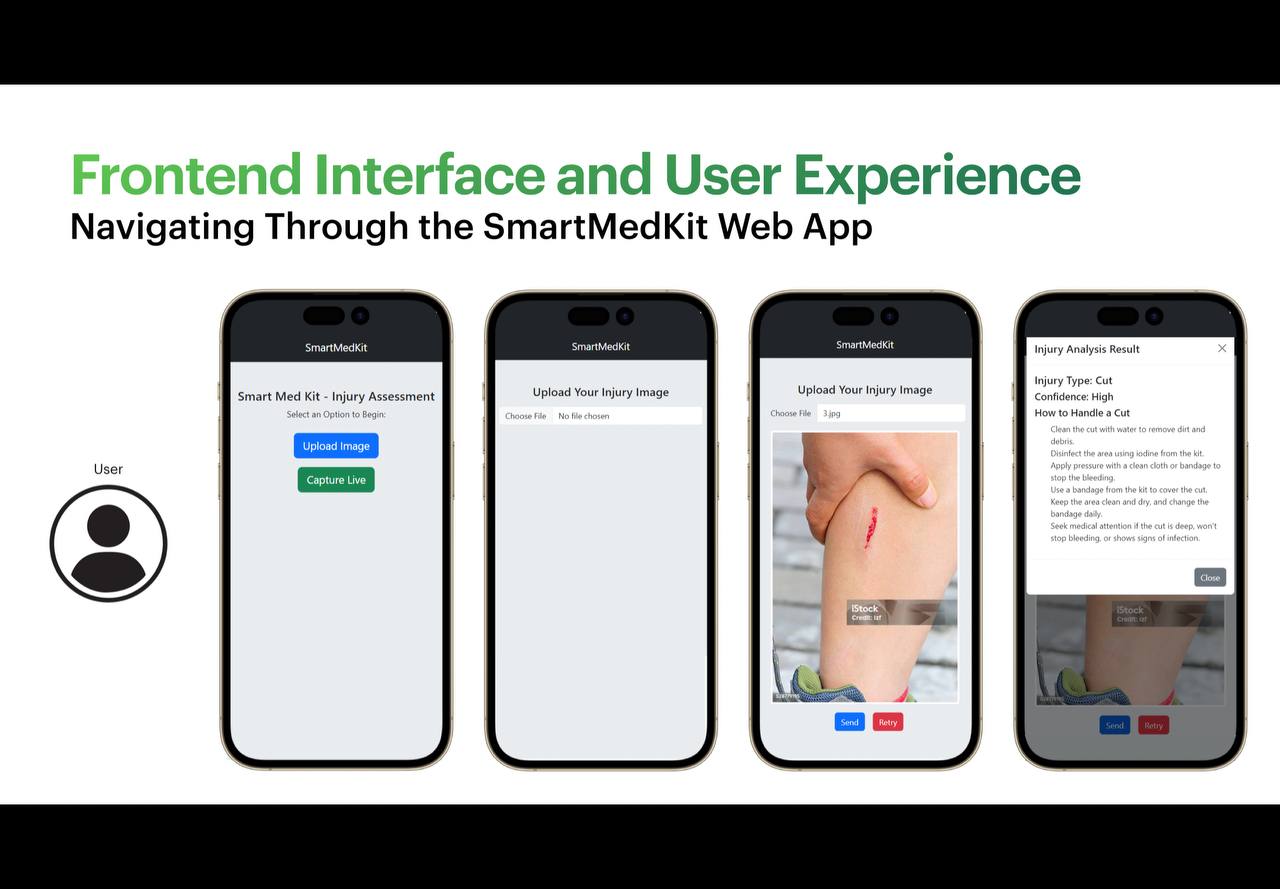
**API Endpoints Overview – Created using Flask & Python**

****

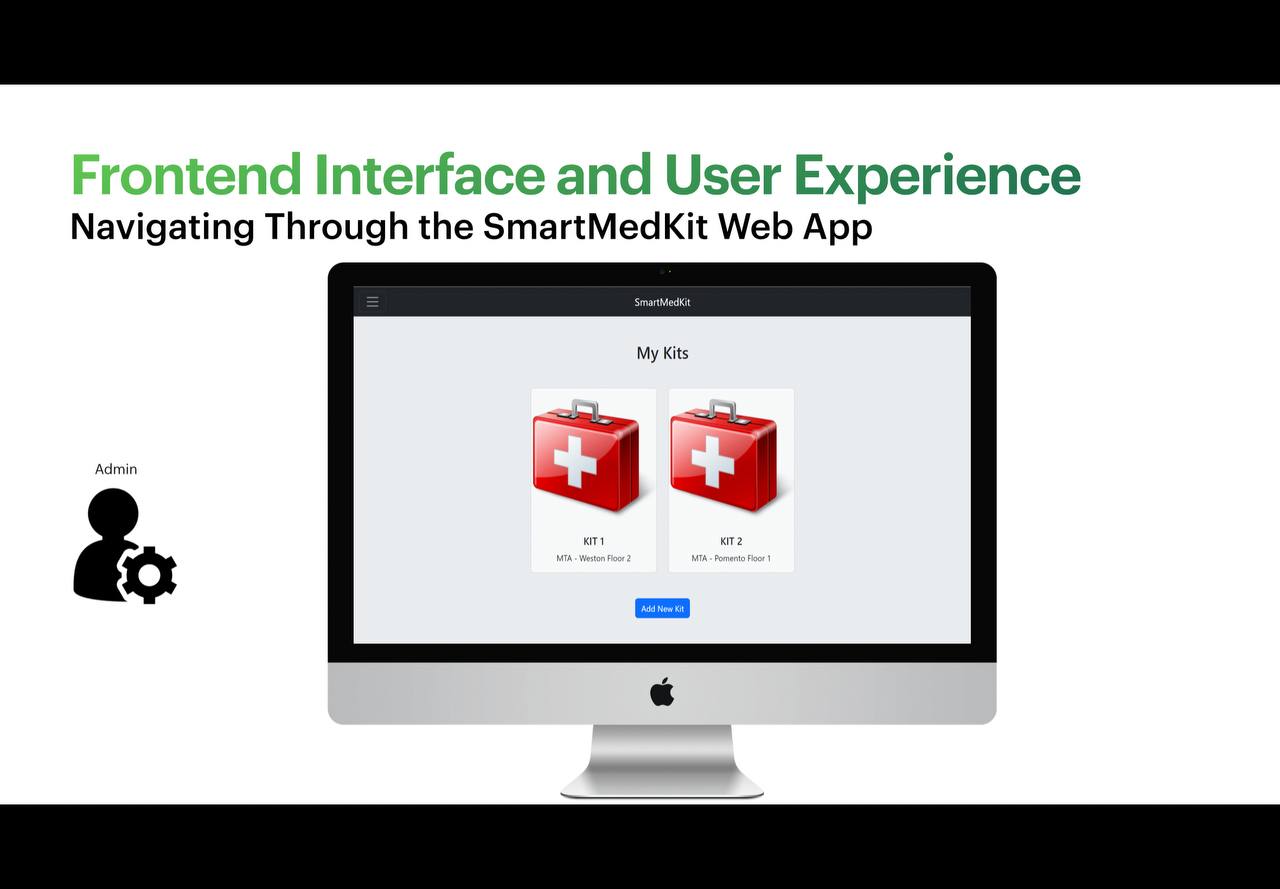
**Database Schema With Relationships**



**Frontend Interfaces:**



A computer screen with a graph on it

Description automatically generated

A circuit board with wires

Description automatically generated**Embedded Electrical Circuit**