# Verdan Platform - Device & Subscription Management

Generated on: 2025-02-07 07:49:32

## 1. Project Structure

The project is designed for modularity, scalability, and ease of management.

/verdan\_platform  
│── /apps # Holds individual app modules  
│── /devices # Manages physical IoT devices  
│ ├── models.py # Device database models  
│ ├── routes.py # API for managing devices  
│ ├── services.py # Business logic for device handling  
│── /device\_groups # Organizes devices into update groups  
│ ├── models.py # Device group database models  
│ ├── routes.py # API for managing device groups  
│── /subscriptions # Manages app & device subscriptions  
│ ├── models.py # Subscription database models  
│ ├── routes.py # API for subscriptions  
│ ├── services.py # Handles Stripe integration  
│── /alerts # Handles alert triggers & notifications  
│── /models # Global models (accounts, apps, devices, etc.)  
│── /services # Business logic for global entities  
│── /config.py # Configuration settings  
│── /main.py # Main Flask app  
│── /wsgi.py # WSGI entry point  
│── .env # Stores environment variables (excluded from Git)  
│── /requirements.txt

## 2. Database Models

### Devices Table

CREATE TABLE devices (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 account\_id UUID REFERENCES accounts(id) ON DELETE CASCADE,  
 app\_id UUID REFERENCES apps(id) ON DELETE CASCADE,  
 device\_group\_id UUID REFERENCES device\_groups(id) ON DELETE SET NULL,  
 stripe\_product\_id TEXT UNIQUE NOT NULL,  
 stripe\_price\_id TEXT UNIQUE NOT NULL,  
 iccid TEXT UNIQUE NOT NULL,  
 status TEXT CHECK (status IN ('active', 'paused', 'deactivated')) DEFAULT 'active',  
 created\_at TIMESTAMP DEFAULT NOW()  
 );

### Sim Cards Table

CREATE TABLE sim\_cards (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 iccid TEXT UNIQUE NOT NULL,  
 device\_id UUID REFERENCES devices(id) ON DELETE SET NULL,  
 status TEXT CHECK (status IN ('active', 'deactivated')) DEFAULT 'active',  
 created\_at TIMESTAMP DEFAULT NOW()  
 );

### Device Groups Table

CREATE TABLE device\_groups (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 name TEXT NOT NULL,  
 firmware\_version TEXT NOT NULL,  
 created\_at TIMESTAMP DEFAULT NOW()  
 );

### Subscriptions Table

CREATE TABLE subscriptions (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 account\_id UUID REFERENCES accounts(id) ON DELETE CASCADE,  
 type TEXT CHECK (type IN ('app', 'device')) NOT NULL,  
 reference\_id UUID NOT NULL,  
 stripe\_subscription\_id TEXT UNIQUE NOT NULL,  
 status TEXT CHECK (status IN ('active', 'paused', 'canceled')) DEFAULT 'active',  
 created\_at TIMESTAMP DEFAULT NOW()  
 );

### Payment Methods Table

CREATE TABLE payment\_methods (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 account\_id UUID REFERENCES accounts(id) ON DELETE CASCADE,  
 stripe\_payment\_method\_id TEXT UNIQUE NOT NULL,  
 last4 TEXT NOT NULL,  
 brand TEXT NOT NULL,  
 is\_default BOOLEAN DEFAULT FALSE,  
 created\_at TIMESTAMP DEFAULT NOW()  
 );

## 3. Flask API Routes

### Create a New Device

POST /devices/add  
 Request Body:  
 {  
 "account\_id": "account-uuid",  
 "app\_id": "app-uuid",  
 "device\_group\_id": "device-group-uuid",  
 "iccid": "1234567890123456789",  
 "monthly\_price": 19.99,  
 "yearly\_price": 199.99  
 }

### List All Devices for an Account

GET /devices/list?account\_id=<account\_uuid>&app\_id=<app\_uuid>

### Pause a Device Subscription

POST /devices/pause  
 Request Body:  
 {  
 "device\_id": "device-uuid"  
 }

### Create a Device Group

POST /admin/device\_groups/add  
 Request Body:  
 {  
 "name": "Soil Moisture Sensors",  
 "firmware\_version": "1.0.2"  
 }

### Assign a Device to a Device Group

POST /admin/devices/update\_group  
 Request Body:  
 {  
 "device\_id": "device-uuid",  
 "device\_group\_id": "group-uuid"  
 }

### Update Device Pricing

POST /admin/devices/update\_price  
 Request Body:  
 {  
 "device\_id": "device-uuid",  
 "monthly\_price": 24.99,  
 "yearly\_price": 249.99  
 }

### Add Payment Method

POST /payment\_methods/add  
 Request Body:  
 {  
 "account\_id": "account-uuid",  
 "stripe\_payment\_method\_id": "pm\_1AbCD2...",  
 "last4": "4242",  
 "brand": "Visa"  
 }

### Set Default Payment Method

POST /payment\_methods/set\_default  
 Request Body:  
 {  
 "account\_id": "account-uuid",  
 "payment\_method\_id": "pm\_1AbCD2..."  
 }

## 4. Admin Panel UI Structure

\*\*Device Groups Page\*\*  
- Lists all device groups with:  
 - Name  
 - Firmware version  
 - Number of devices  
 - Edit/Delete options  
- Button to add a new device group  
  
\*\*Devices Page (Admin)\*\*  
- Lists all devices with:  
 - ICCID  
 - Group Assignment (Dropdown)  
 - Monthly/Yearly Pricing (Editable)  
 - Status (Active, Paused, Deactivated)

## 5. Summary

✅ Devices are linked to accounts & apps.  
✅ SIM cards are uniquely assigned to devices.  
✅ Admin can manage device groups & pricing.  
✅ Subscriptions are handled via Stripe.  
✅ Payment methods are stored locally & in Stripe.