AIQ Test Challenge One Step-by-Step Guide for EHAB ABU SAADEH Candidate

Background

I have completed the assignment with the following actions:

1. Imported the necessary data from an Excel file and created a PostgreSQL Docker image for the sheets in the Excel.
2. Created a database dump and initialized a Docker image using this database dump.
3. Chose a suitable template for Express.js API development.
4. Built the business logic according to requirements in the `PlantsServices.js` file.
5. Implemented an API that communicates with the service layer.
6. Used Sequelize ORM for database operations.
7. Implemented Swager UI for interacting with the project

Template Structure

```
+-- index.js                // Application entry point
+-- models                  // Sequelize ORM model layer
+-- services                // Business logic layer
+-- routes                  // Routing and controlling layer
+-- db                      // Database image
|   +-- Dockerfile          // Database Dockerfile
|   +-- init
|       +-- 00-database.sql // Database dump
+-- database.js             // Sequelize database connection configuration
+-- Dockerfile              // Express Dockerfile
+-- .env                    // Environment variables file
+-- ...
```

Quick Setup

Docker Compose

Run the following command in the project directory:

```
docker-compose up -d
```

This command starts the API and database on your local machine. Access them at localhost:8080 and localhost:5432 respectively.

Usage

Swagger URL: http://localhost:8080/api-docs/

Notes

This template follows the classic, bulletproof Node.js 3-layer architecture, emphasizing the **principle of separation of concerns**.

```
...
+-- models                      // Sequelize ORM model layer
+-- services                    // Business logic layer
+-- routes                      // Routing and controlling layer
...
```

Remarks

Reasons for using Sequelize ORM over direct database drivers or query builders:

Sequelize ORM was chosen to support multiple SQL databases easily. However, consider different levels of abstraction as discussed in this article.