### **🚀 VeriHarvest Backend (API & Database)**

**Role:** Backend Developer  
 **Module:** **FastAPI-based API with PostgreSQL Database**

## **📌 1. Overview**

This module provides a **REST API** to store and retrieve **food batch data**, allowing integration with **blockchain, AI, and IoT modules**.

✅ **FastAPI for high-speed backend** ✅ **PostgreSQL for food batch storage** ✅ **CRUD operations for managing food traceability** ✅ **Endpoints to interact with smart contracts & IoT data**

## **📌 2. How to Run**

### **🔹 Prerequisites**

✅ Install **Python 3.8+** ✅ Install dependencies:

pip install fastapi psycopg2 uvicorn

✅ **Set up PostgreSQL Database**

sudo apt update && sudo apt install postgresql

sudo -i -u postgres

psql

CREATE DATABASE veriharvest\_db;

CREATE USER user WITH PASSWORD 'password';

GRANT ALL PRIVILEGES ON DATABASE veriharvest\_db TO user;

**✅ Create Table**

CREATE TABLE food\_batches (

batch\_id SERIAL PRIMARY KEY,

product\_name TEXT,

supplier TEXT,

freshness\_score INT,

blockchain\_verified BOOLEAN,

status TEXT

);

### **🔹 Steps to Execute**

**1️⃣ Run the FastAPI Server**

python veriharvest\_backend.py

2️⃣ **Access API Endpoints**

* **Add a Batch**

curl -X POST "http://localhost:8000/add\_batch/" -H "Content-Type: application/json" -d '{"batch\_id":1,"product\_name":"Organic Apples","supplier":"Farm A","freshness\_score":95,"blockchain\_verified":true,"status":"Safe"}'

* **Get a Batch**

curl -X GET "http://localhost:8000/get\_batch/1"

* **Get All Batches**

curl -X GET "<http://localhost:8000/get_all_batches/>"

## **📌 3. Testing Backend API**

* Use of **Postman** to send test requests.
* Validate **data integrity in PostgreSQL**.
* Ensure **API responses match expected blockchain data**.