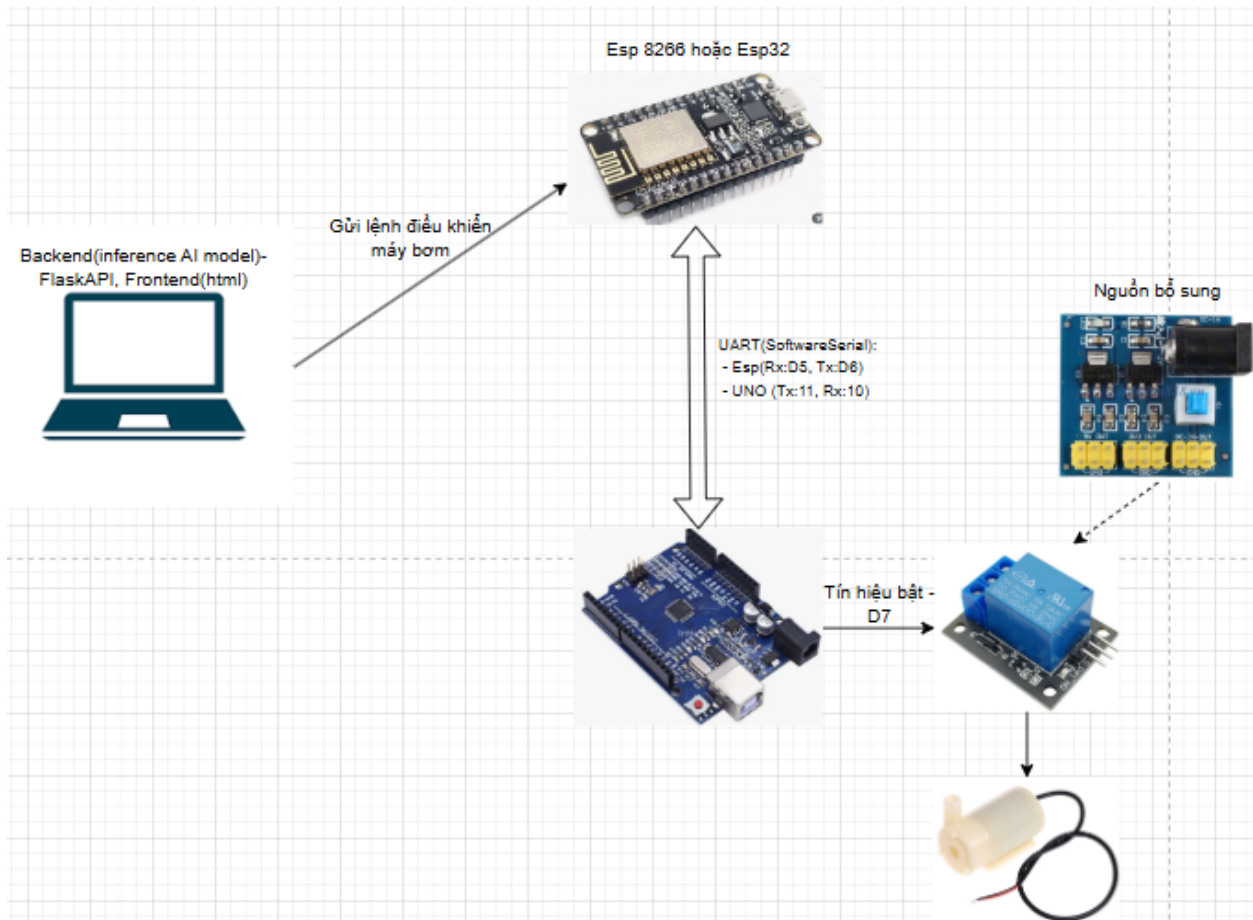
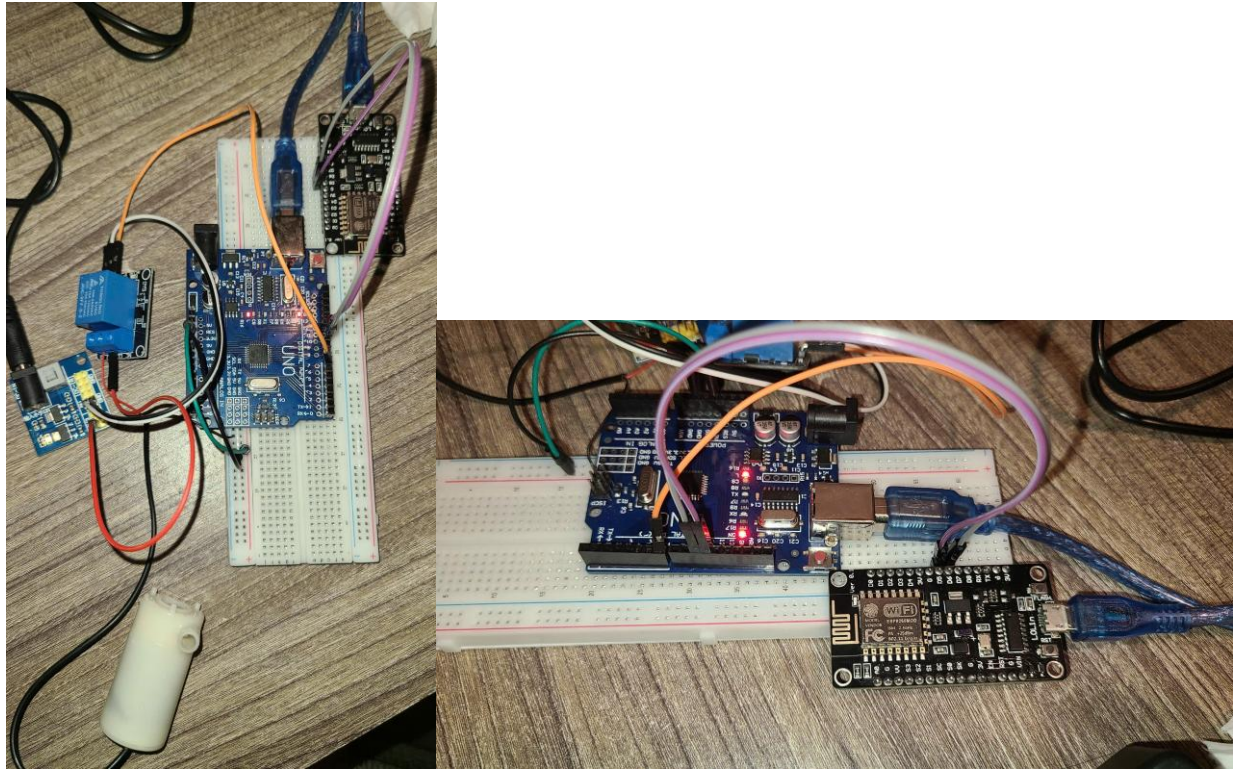


Hướng dẫn tổng quan hệ thống tự động tưới nước thông minh dựa trên AI

Mục đích tài liệu này là giúp sinh viên hiểu: mỗi thành phần trong hệ thống đảm nhiệm nhiệm vụ gì, dữ liệu di chuyển như thế nào, và toàn bộ luồng hoạt động ra sao—mà không đi sâu vào code(Code đã gửi kèm).

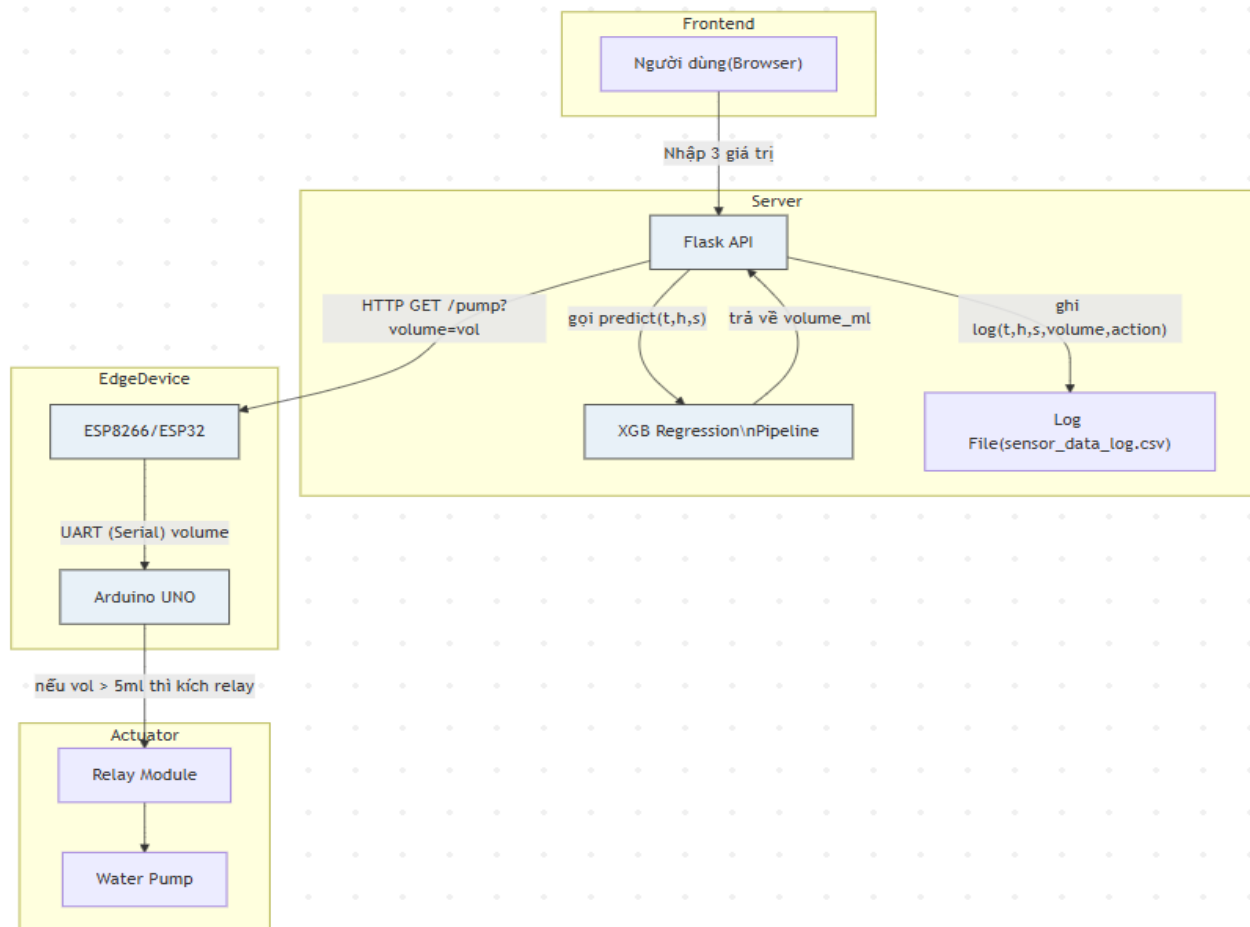
1. Tổng quan kiến trúc hệ thống





Hệ thống gồm 4 khối chính:

1. **Người dùng (Browser)**
2. **Flask Server** (mô hình ML + giao diện web)
3. **ESP8266/ESP32** (nhận lệnh HTTP, chuyển qua UART)
4. **Arduino UNO + Relay + Pump** (nhận lệnh, điều khiển relay và bơm)

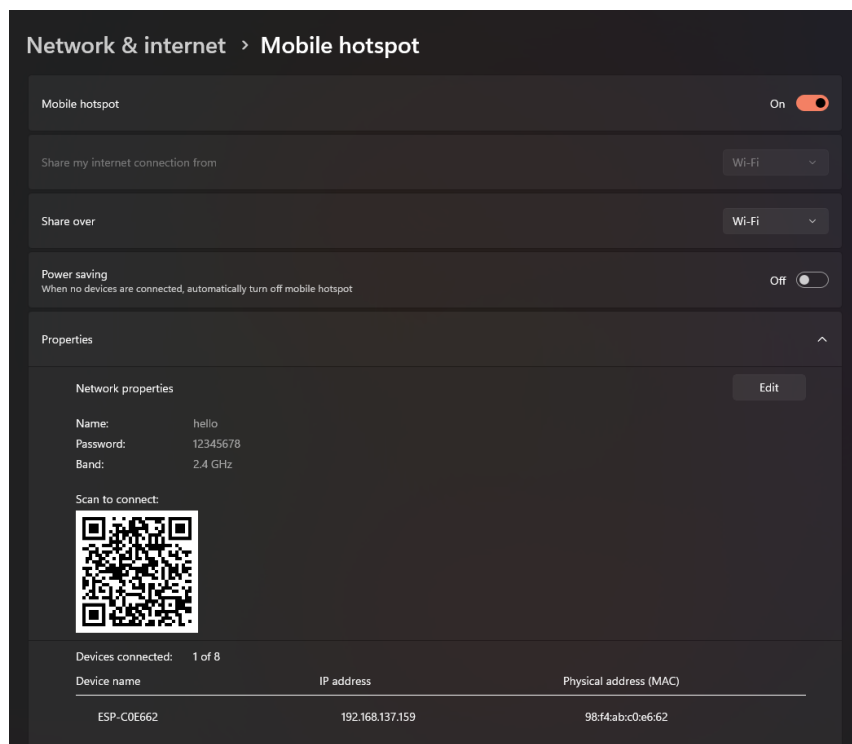


2. Nhiệm vụ từng thành phần

Thành phần	Nhiệm vụ chính
Browser (FE)	- Cung cấp form để nhập ba giá trị: nhiệt độ, độ ẩm không khí, độ ẩm đất.
	- Gửi POST về Server, nhận kết quả dự đoán và hiển thị cho người dùng.
Flask Server(app.py)	- Nhận dữ liệu POST từ Browser.
	- Gọi mô hình XGB để dự đoán volume_ml.
	- Gửi HTTP GET kèm tham số volume đến ESP8266.
	- Ghi log toàn bộ dữ liệu và kết quả vào file CSV để theo dõi.
XGB Model	- Nhận đầu vào (T,H,S), trả về giá trị số ml cần bơm.
	- Được đóng gói dưới dạng pipeline: scaler + XGBRegressor.
ESP8266/ESP32(esp8266.txt – sao chép code và dán vào)	- Kết nối Wi-Fi đến hotspot (laptop hoặc router).
	- Cung cấp endpoint /pump?volume=....

Arduino IDE cho việc nạp code)	- Khi nhận request, parse giá trị volume, gửi qua UART đến Arduino UNO.
Arduino UNO(Arduino.txt – sao chép code và dán vào Arduino IDE cho việc nạp code)	- Nhận chuỗi volume qua SoftwareSerial (UART emulation).
	- Chuyển chuỗi thành số, so sánh với ngưỡng 5 ml.
	- Nếu volume > 5 ml, kích RELAY cho phép dòng điện đến bơm trong khoảng volume * X ms.
Relay + Pump	- RELAY đóng/ngắt mạch điện điều khiển bơm.
	- Bơm di chuyển nước tương ứng với thời gian RELAY đóng.

Thiết lập phát wifi từ laptop:



3. Luồng dữ liệu chi tiết

1. Nhập và truyền dữ liệu

- Sinh viên mở trình duyệt và truy cập địa chỉ <http://127.0.0.1:5000>.
- Điền ba thông số cảm biến (T, H, S) vào form.

- Khi nhấn “Gửi”, trình duyệt thực hiện **HTTP POST** kèm JSON hoặc form data chứa ba trường.

2. Xử lý trên Flask Server(thay địa chỉ ESP_IP sử dụng-có thể xem trên serial hoặc phần hotspot trên laptop sẽ thấy IP thiết bị kết nối đến).

- Server nhận POST, trích xuất T, H, S.
- Gọi tới pipeline ML, nhận giá trị volume_ml (float).
- Luôn thực hiện **HTTP GET** tới `http://<ESP_IP>/pump?volume=<volume_ml>`.
- Ghi dòng mới vào file sensor_data_log.csv:

timestamp, temperature, humidity_env, soil_moisture, predicted_volume, action ` ` `

- Trả về trang HTML với kết quả dự đoán và trạng thái gửi lệnh.

3. ESP8266/ESP32

- ESP đang ở chế độ station, kết nối vào hotspot của laptop(laptop phát wifi ở dạng sóng 2.4Ghz).
- Khi nhận GET /pump?volume=XY.ZZ, module đọc giá trị tham số volume.
- Gửi chuỗi ASCII "XY.ZZ" kèm ký tự xuống dòng qua cổng UART (SoftwareSerial) tới Arduino UNO.

4. Arduino UNO

- Uno mở cổng SoftwareSerial (ví dụ Rx=D10, Tx=D11) để đọc dữ liệu từ ESP.
- Đọc chuỗi đến khi gặp \n, chuyển thành biến float vol.
- So sánh với THRESHOLD = 5.0 ml:
 - Nếu $vol > 5.0$, đưa tín hiệu HIGH lên chân RELAY trong thời gian $vol * K$ ms (K là hệ số tương ứng: ví dụ 1000 ms cho 1 ml).
 - Nếu $vol \leq 5.0$, không kích hoạt RELAY.
- Arduino cũng in thông tin vol và trạng thái “Pumped” hoặc “Skipped” ra Serial Monitor để debug.

5. Relay → Pump

- Khi chân RELAY được kích HIGH, mạch điện cấp nguồn (5V hoặc 12V tùy bơm) đến đầu bơm.

- Khi RELAY trở về LOW, bơm ngừng hoạt động.

4. Hướng dẫn lắp đặt phần cứng

1. **Nguồn chung:** Nối chung GND của laptop/ESP/UNO/Relay/Pump.

2. **Kết nối UART:**

- ESP Tx (GPIO D5) → UNO Rx (D10).
- ESP Rx (GPIO D6) ← UNO Tx (D11) (nếu không gửi phản hồi thì có thể không cần).

3. **Relay:**

- UNO D7 → IN trên Relay.
- Relay VCC → 5V (UNO hoặc nguồn riêng).
- Relay GND → GND chung.

4. **Bơm:**

- Relay COM → cực dương bơm.
- Relay NO → nguồn cấp (+5V hoặc +12V).
- Cực âm bơm → GND chung.

Thiết bị A	Chân A	Thiết bị B	Chân B	Chức năng
ESP8266/ESP32	D5 (GPIO14)	UNO	D10 (SoftwareSerial RX)	TX từ ESP → UNO (gửi volume)
ESP8266/ESP32	D6 (GPIO12)	UNO	D11 (SoftwareSerial TX)	RX vào ESP ← UNO (nếu cần)
UNO	D7	Relay module	IN	Điều khiển bật/tắt relay
UNO	5 V	Relay module	VCC	Cung cấp nguồn 5 V cho module relay

UNO	GND	Relay module	GND	Mass chung (5 V)
UNO	GND	ESP8266/ESP32	GND	Mass chung giữa ESP và UNO
Mạch chia	3V3	Relay module	COM	Relay kích hoạt sẽ thông mạch COM với NO
Mạch chia	NO	Máy bơm	Dây đỏ	Cấp điện cho máy bơm từ chân NO

Mạch chia chỉ có nhiệm vụ cấp điện, nếu không có các bạn có thể cắm vào chân 5V hoặc 3V tương ứng trên ArduinoUNO R3 hoặc Esp nhưng hệ thống sẽ hoạt động không ổn định.

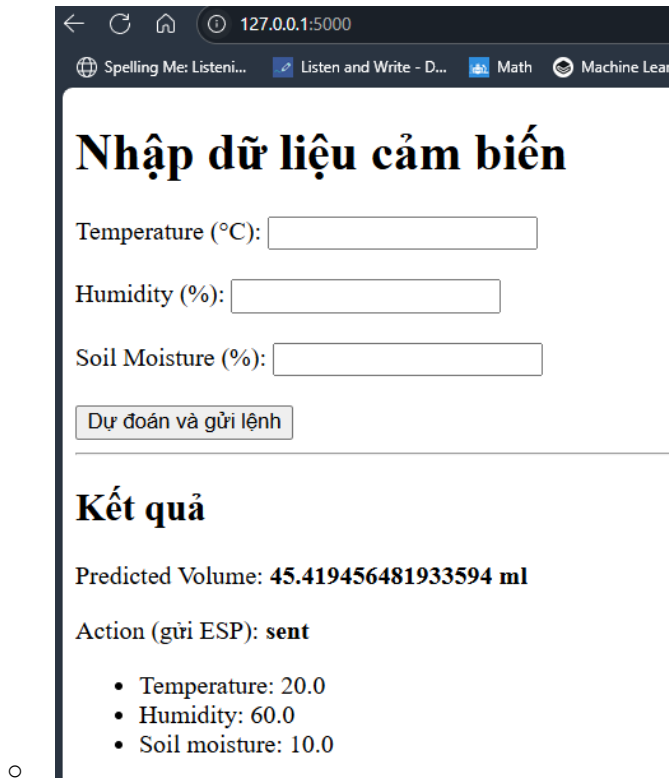
5. Quy trình kiểm thử

1. Kiểm thử mô hình:

- Chạy train_pipeline.py, đảm bảo xgb_reg_pipeline.joblib được sinh ra.

2. Kiểm thử Server-FE:

- python app.py → truy cập form, submit giá trị test, xem kết quả hiện trên web.



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000'. The browser tabs include 'Spelling Me: Listen...', 'Listen and Write - D...', 'Math', and 'Machine Learn...'. The main content area has a title 'Nhập dữ liệu cảm biến' (Enter sensor data) in a large, bold, black font. Below the title are three input fields: 'Temperature (°C):', 'Humidity (%)', and 'Soil Moisture (%)'. A button labeled 'Dự đoán và gửi lệnh' (Predict and send command) is positioned below the input fields. The results section, titled 'Kết quả' (Results), shows 'Predicted Volume: 45.419456481933594 ml' and 'Action (gửi ESP): sent'. A bulleted list of sensor values is displayed: Temperature: 20.0, Humidity: 60.0, and Soil moisture: 10.0.

Nhập dữ liệu cảm biến

Temperature (°C):

Humidity (%):

Soil Moisture (%):

Kết quả

Predicted Volume: **45.419456481933594 ml**

Action (gửi ESP): **sent**

- Temperature: 20.0
- Humidity: 60.0
- Soil moisture: 10.0

3. Kiểm thử ESP:

- Cấu hình đúng SSID/password(sử dụng laptop phát wifi, cài đặt sóng 2.4Ghz).

4. Kiểm thử UNO + Relay:

- Nạp code UNO.

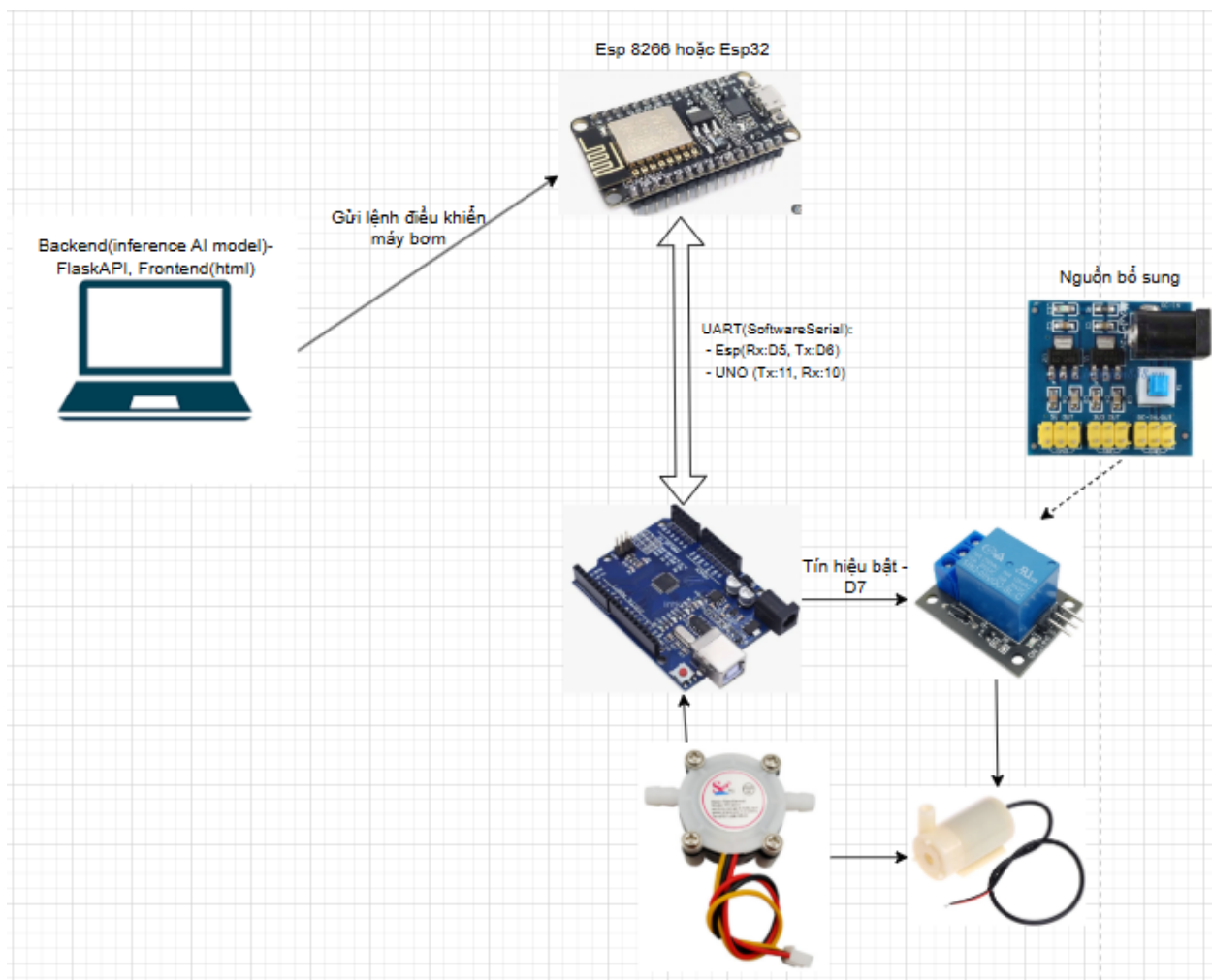
5. Kiểm thử End-to-end:

- Nhập form Flask giá trị sao cho predicted > 5 ml(lấy mẫu trong tập test.csv), submit.
- Quan sát ESP nhận GET, UNO bơm nước.
- Kiểm tra log file sensor_data_log.csv ghi đúng record.

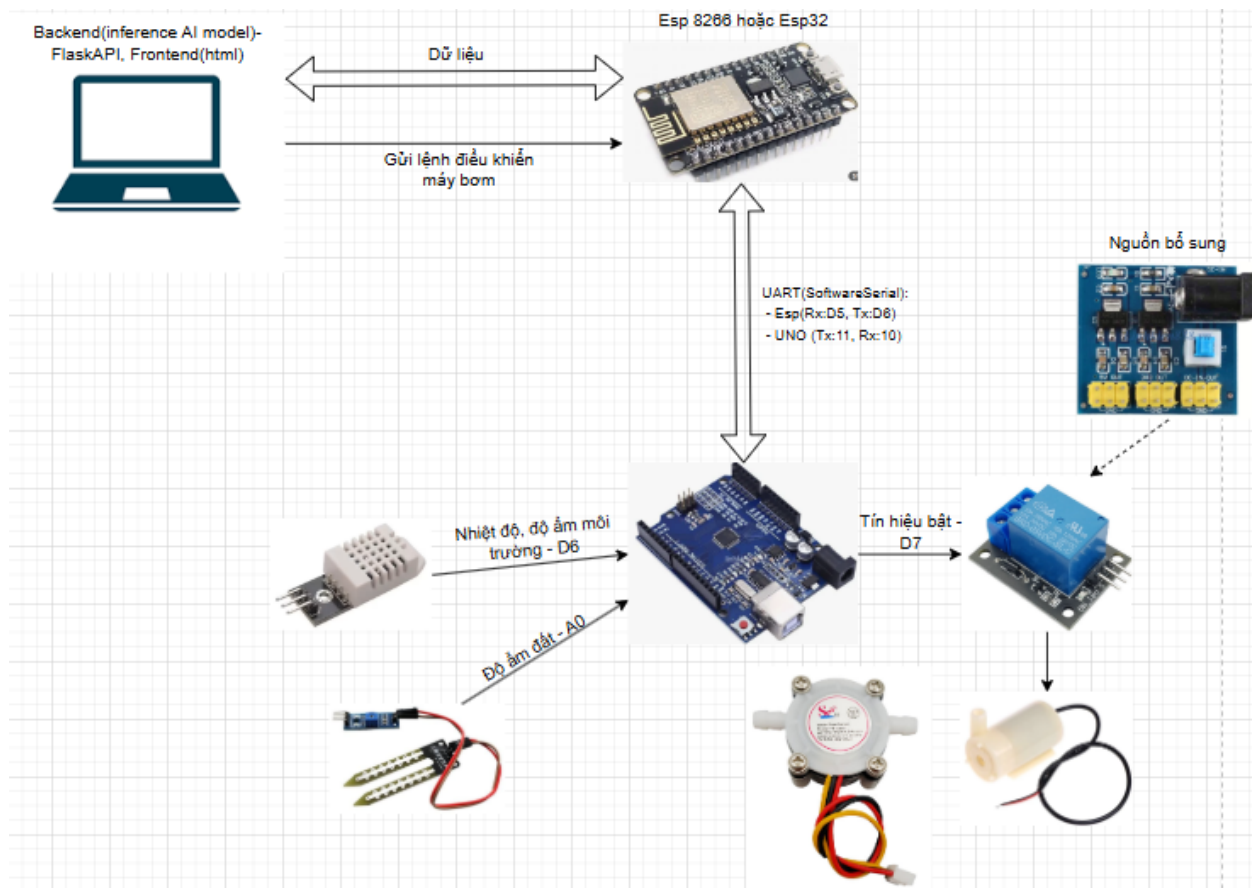
6. Yêu cầu phát triển

Từ mô hình 1, thực hiện phát triển mô hình: 2; 3; 4 được mô tả phía dưới.

Mô hình 2: Thực hành lập trình với cảm biến lưu lượng nước để kiểm soát lượng nước được bơm



Tiếp tục giữ cấu trúc của mô hình 2, phát triển mô hình 3:

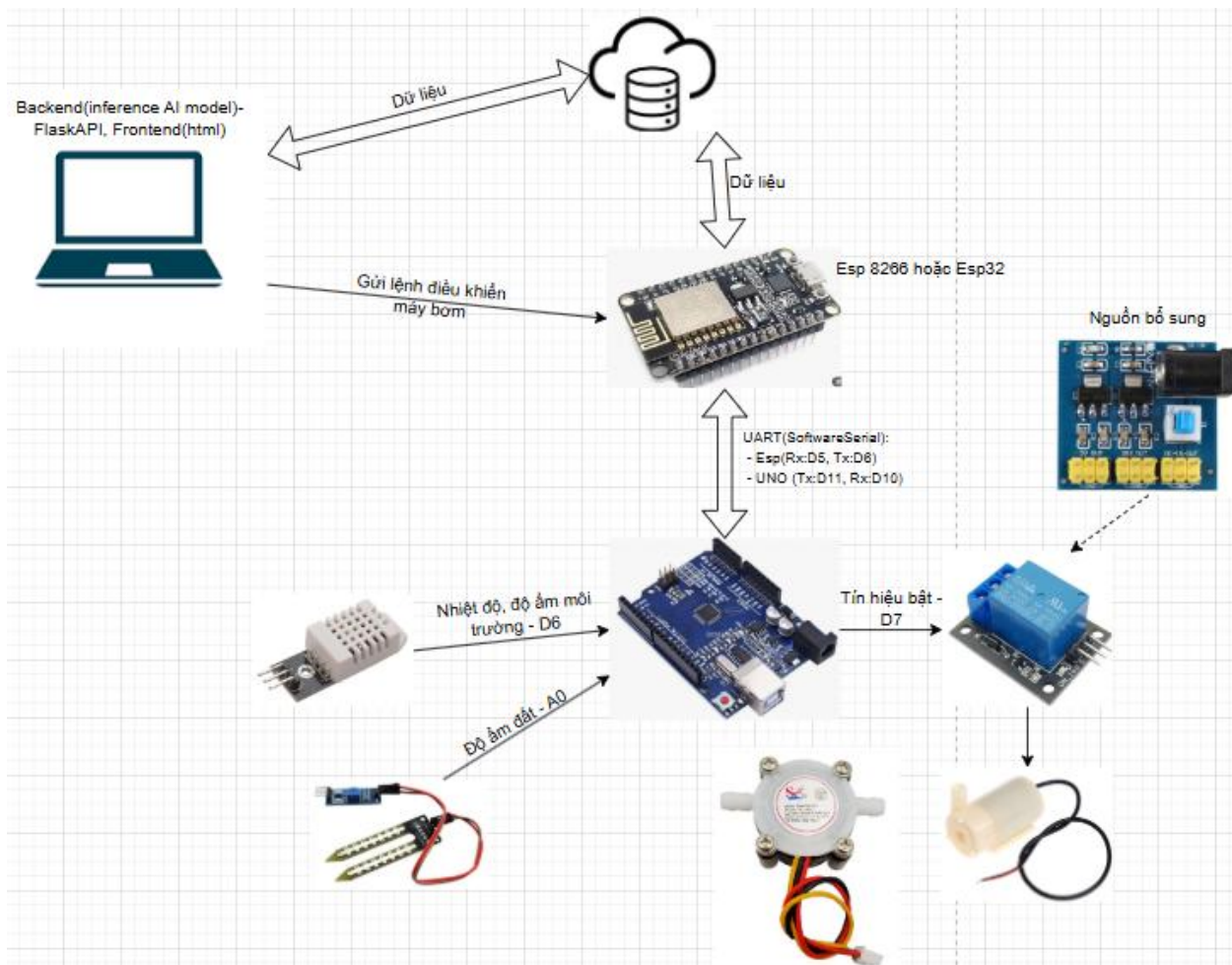


Giữ luồng điều khiển bơm như mô hình 2, thực hiện thêm:

- Kết nối cảm biến dht11 và cảm biến đo độ ẩm đất với UNO
- UNO thực hiện gửi dữ liệu thu được tới ESP
- ESP thực hiện gửi dữ liệu lên và lưu tại Server

=> Lúc này đưa dữ liệu thành đầu vào của mô hình AI, không cần tới giao diện nhập chỉ số

Tiếp tục phát triển lên Mô hình 4:



Giữ luồng hoạt động như Mô hình 3, thực hiện thêm:

- Dữ liệu từ cảm biến sẽ được Esp gửi và lưu trên cloud
- Server cần truy cập dữ liệu sẽ thực hiện lấy từ cloud