**Tên đề tài:**

**Hệ thống Phân tích và Gợi ý Nhiệt độ, Độ ẩm Cá nhân hóa bằng Trí tuệ Nhân tạo**

*(AI-Powered Personalized Climate Analysis and Recommendation System)*

**1. Tóm tắt đề tài**

Dự án này tập trung vào việc xây dựng một hệ thống IoT thông minh có khảd năng theo dõi điều kiện nhiệt độ và độ ẩm của môi trường sống theo thời gian thực, đồng thời thu thập phản hồi từ người dùng để **huấn luyện một mô hình học máy**. Mục tiêu cuối cùng là cung cấp những gợi ý được cá nhân hóa nhằm tối ưu hóa sự thoải mái cho từng người dùng cụ thể, giúp họ điều chỉnh điều hòa, máy tạo ẩm hoặc các thiết bị khác một cách hiệu quả và tiết kiệm năng lượng.

**2. Đặt vấn đề**

Sự thoải mái về nhiệt độ và độ ẩm là một yếu tố chủ quan, khác biệt đáng kể giữa mỗi người. Các hệ thống điều hòa không khí truyền thống thường hoạt động dựa trên một ngưỡng nhiệt độ cài đặt sẵn, không có khả năng thích ứng với cảm nhận riêng của từng cá nhân hay sự thay đổi của các yếu tố môi trường khác. Điều này dẫn đến tình trạng lãng phí năng lượng khi môi trường quá lạnh hoặc không đủ mát, cũng như ảnh hưởng tiêu cực đến sức khỏe và năng suất làm việc.

Việc ứng dụng **Trí tuệ nhân tạo (AI)** và **Học máy (Machine Learning)** để phân tích dữ liệu môi trường và phản hồi của người dùng sẽ tạo ra một giải pháp đột phá, cho phép hệ thống "học" và hiểu được sở thích cá nhân, từ đó đưa ra các khuyến nghị chính xác và thông minh hơn.

**3. Mục tiêu của đề tài**

* **Xây dựng Module phần cứng:** Thiết kế và lắp đặt thành công một thiết bị IoT sử dụng vi điều khiển **ESP32** và cảm biến **DHT22** để đo lường chính xác nhiệt độ () và độ ẩm (%).
* **Thu thập và Lưu trữ Dữ liệu:** Gửi dữ liệu cảm biến (nhiệt độ, độ ẩm, thời gian) và dữ liệu phản hồi của người dùng (quá nóng, quá lạnh, thoải mái) đến một máy chủ hoặc nền tảng đám mây (Cloud) để lưu trữ.
* **Xây dựng và Huấn luyện Mô hình AI:**
  + Sử dụng dữ liệu đã thu thập để huấn luyện một mô hình học máy.
  + Mô hình này có khả năng dự đoán mức độ thoải mái của người dùng dựa trên các điều kiện môi trường hiện tại.
* **Phát triển Hệ thống Gợi ý:** Dựa trên dự đoán của mô hình AI, hệ thống sẽ đưa ra các gợi ý cụ thể cho người dùng, ví dụ: "Tăng nhiệt độ điều hòa lên 26°C" hoặc "Bật máy tạo ẩm".
* **Tạo Giao diện Người dùng (Tùy chọn mở rộng):** Xây dựng một ứng dụng di động hoặc trang web đơn giản để người dùng có thể cung cấp phản hồi và nhận các gợi ý từ hệ thống.

**4. Sơ đồ hệ thống & Kiến trúc**

Hệ thống sẽ được xây dựng theo kiến trúc 3 lớp chính:

**a. Lớp Phần cứng (Hardware Layer)**

* **Vi điều khiển trung tâm:** **ESP32**. Đây là lựa chọn lý tưởng vì có tích hợp sẵn Wi-Fi, Bluetooth, hiệu năng mạnh mẽ và giá thành hợp lý.
* **Cảm biến:** **DHT22**. Cung cấp các phép đo nhiệt độ và độ ẩm với độ chính xác khá tốt cho các ứng dụng trong nhà.
* **Nguồn:** Cung cấp nguồn 5V qua cổng USB hoặc pin.

**b. Lớp Giao tiếp & Dữ liệu (Communication & Data Layer)**

* **Giao thức:** Dữ liệu từ ESP32 sẽ được gửi lên server thông qua Wi-Fi bằng giao thức **MQTT** hoặc **HTTP POST**. MQTT được ưu tiên vì tính nhẹ, hiệu quả và phù hợp cho các ứng dụng IoT.
* **Nền tảng Cloud/Server:**
  + **Lựa chọn 1 (Dễ triển khai):** Sử dụng các nền tảng IoT như **Blynk IoT, Adafruit IO, hoặc Thingspeak**. Các nền tảng này cung cấp sẵn cơ sở dữ liệu, API và giao diện trực quan hóa.
  + **Lựa chọn 2 (Linh hoạt hơn):** Tự dựng một backend server (sử dụng Node.js, Python Flask/Django) và cơ sở dữ liệu (MySQL, PostgreSQL, MongoDB) trên một dịch vụ đám mây như AWS, Google Cloud hoặc Heroku.
* **Cơ sở dữ liệu:** Sẽ lưu trữ các bản ghi bao gồm: timestamp, temperature, humidity, user\_feedback.

**c. Lớp Ứng dụng & Trí tuệ Nhân tạo (Application & AI Layer)**

* **Mô hình Học máy:**
  + **Loại bài toán:** Đây là một bài toán **phân loại (Classification)**. Dữ liệu đầu vào là (nhiệt độ, độ ẩm, thời gian) và đầu ra là một trong ba nhãn: (Quá lạnh, Thoải mái, Quá nóng).
  + **Thuật toán đề xuất:**
    - **Support Vector Machine (SVM):** Hoạt động tốt với các bộ dữ liệu không quá lớn và có ranh giới quyết định rõ ràng.
    - **Random Forest:** Rất hiệu quả, dễ triển khai và có khả năng chống nhiễu tốt.
    - **Mạng Neural Network (NN) đơn giản:** Có thể học được các mối quan hệ phức tạp hơn nếu bộ dữ liệu đủ lớn.
* **Quy trình huấn luyện:**
  1. **Tiền xử lý dữ liệu:** Làm sạch dữ liệu, xử lý các giá trị thiếu (nếu có).
  2. **Phân chia dữ liệu:** Chia bộ dữ liệu thành tập huấn luyện (training set) và tập kiểm thử (testing set) theo tỷ lệ 80/20.
  3. **Huấn luyện (Training):** Sử dụng tập huấn luyện để "dạy" cho mô hình.
  4. **Đánh giá (Evaluation):** Dùng tập kiểm thử để đánh giá độ chính xác của mô hình.
* **API Gợi ý:** Tạo một API endpoint để hệ thống có thể gửi dữ liệu nhiệt độ, độ ẩm hiện tại và nhận lại dự đoán/gợi ý từ mô hình AI đã được huấn luyện.

**5. Lộ trình thực hiện chi tiết**

**Giai đoạn 1: Xây dựng Phần cứng và Thu thập Dữ liệu (2 tuần)**

1. **Kết nối mạch:** Nối cảm biến DHT22 với ESP32.
2. **Lập trình ESP32:**
   * Viết code Arduino hoặc MicroPython để đọc dữ liệu từ DHT22.
   * Viết code kết nối Wi-Fi và gửi dữ liệu nhiệt độ, độ ẩm lên nền tảng Cloud đã chọn qua MQTT/HTTP theo chu kỳ (ví dụ: 5 phút một lần).
3. **Thiết lập cơ sở thu thập phản hồi:** Tạo 3 nút nhấn vật lý kết nối với ESP32 (tương ứng với "Lạnh", "OK", "Nóng") hoặc tạo giao diện đơn giản trên Blynk/Adafruit IO để người dùng nhập phản hồi. Mỗi khi người dùng nhấn nút, dữ liệu phản hồi sẽ được gửi lên server cùng với dữ liệu cảm biến tại thời điểm đó.
4. **Bắt đầu thu thập:** Để hệ thống chạy liên tục và khuyến khích người dùng cung cấp phản hồi thường xuyên trong các điều kiện môi trường khác nhau. **Cần thu thập ít nhất vài trăm điểm dữ liệu để mô hình có ý nghĩa.**

**Giai đoạn 2: Huấn luyện và Triển khai Mô hình AI (2 tuần)**

1. **Xuất dữ liệu:** Tải dữ liệu đã thu thập từ Cloud về dưới dạng file CSV.
2. **Xây dựng mô hình:**
   * Sử dụng các thư viện Python như **Scikit-learn**, **Pandas**, và **NumPy**.
   * Viết script để đọc dữ liệu, tiền xử lý, phân chia và huấn luyện bằng thuật toán đã chọn (ví dụ: Random Forest).
3. **Đánh giá mô hình:** Kiểm tra độ chính xác, ma trận nhầm lẫn (confusion matrix) để xem mô hình dự đoán tốt như thế nào.
4. **Lưu mô hình:** Lưu lại mô hình đã huấn luyện (ví dụ: file .pkl hoặc .h5).
5. **Triển khai mô hình:**
   * Tạo một API đơn giản (sử dụng Flask) để nhận yêu cầu chứa nhiệt độ và độ ẩm.
   * API này sẽ tải mô hình đã lưu, thực hiện dự đoán và trả về kết quả ("Thoải mái", "Quá nóng", v.v.).

**Giai đoạn 3: Tích hợp và Hoàn thiện Hệ thống (1-2 tuần)**

1. **Cập nhật code ESP32:** Sửa đổi code trên ESP32 để ngoài việc gửi dữ liệu lên, nó còn có thể gọi đến API dự đoán.
2. **Tạo logic gợi ý:** Dựa trên kết quả trả về từ API, lập trình cho ESP32 (hoặc ứng dụng di động) để hiển thị các thông báo gợi ý phù hợp.
   * Nếu API trả về "Quá nóng" -> Hiển thị "Giảm nhiệt độ AC" hoặc "Bật quạt".
   * Nếu API trả về "Quá lạnh" -> Hiển thị "Tăng nhiệt độ AC".
3. **Kiểm thử toàn diện:** Chạy thử toàn bộ hệ thống để đảm bảo các thành phần hoạt động trơn tru với nhau.
4. **Viết báo cáo và tài liệu:** Mô tả lại toàn bộ quá trình thiết kế, thực hiện, các vấn đề gặp phải và kết quả đạt được.

**6. Công nghệ và Công cụ sử dụng**

* **Phần cứng:** ESP32, DHT22, nút nhấn, breadboard, dây cắm.
* **Ngôn ngữ lập trình:** C++ (Arduino), Python.
* **Môi trường phát triển:** Arduino IDE (với ESP32 core), Visual Studio Code, Jupyter Notebook/Google Colab.
* **Thư viện Python:** Scikit-learn, Pandas, NumPy, Flask.
* **Nền tảng Cloud/IoT:** Blynk, Adafruit IO, Thingspeak, hoặc AWS/Google Cloud.
* **Giao thức:** MQTT, HTTP.

**7. Hướng phát triển và Mở rộng**

* **Thu thập thêm dữ liệu:** Tích hợp thêm các cảm biến khác như cảm biến ánh sáng (LDR), cảm biến chất lượng không khí (PM2.5) để mô hình có thêm nhiều thông tin đầu vào và trở nên chính xác hơn.
* **Tự động hóa:** Thay vì chỉ gợi ý, hệ thống có thể tích hợp với các thiết bị thông minh (ổ cắm thông minh, bộ điều khiển hồng ngoại) để **tự động điều khiển** điều hòa, quạt.
* **Hỗ trợ nhiều người dùng:** Mở rộng hệ thống để có thể học và đưa ra gợi ý cho nhiều người dùng khác nhau trong cùng một không gian.
* **Phân tích theo thời gian:** Sử dụng các mô hình chuỗi thời gian (Time-series models) như LSTM để dự đoán xu hướng nhiệt độ và đưa ra gợi ý phủ đầu.

Chúc bạn thực hiện thành công đề tài thú vị và rất thực tiễn này!