**Mục lục**

Mục lục

Lịch sử chỉnh sửa

1. Giới thiệu

1.1 Mục đích

1.2 Quy ước

1.3 Đối tượng và hướng dẫn đọc

1.4 Phạm vi dự án

1.5 Tài liệu tham khảo

2. Mô tả tổng quan

2.1 Góc nhìn sản phẩm

2.2 Tính năng sản phẩm

2.3 Lớp người dùng và đặc điểm

2.4 Môi trường hoạt động

2.5 Hạn chế thiết kế và triển khai

2.6 Tài liệu người dùng

2.7 Giả định và phụ thuộc

3. Tính năng hệ thống

3.1 Tính năng: Thu thập dữ liệu cảm biến thời tiết

3.2 Tính năng: Dự đoán khả năng mưa bằng mô hình hồi quy

3.3 Tính năng: Giao tiếp UART/Wi-Fi và hiển thị dữ liệu

4. Yêu cầu giao diện

4.1 Giao diện người dùng

4.2 Giao diện phần cứng

4.3 Giao diện phần mền

4.4 Giao diện truyền thông

5. Các yêu cầu phi chức năng khác

5.1 Yêu cầu hiệu năng

5.2 Yêu cầu an toàn

5.3 Yêu cầu bảo mật

5.4 Thuộc tính chất lượng phần mềm

6. Các yêu cầu khác

Phụ lục A: Thuật ngữ

Phụ luc B: Mô hình phân tích

Phụ lục C: Danh sách vấn đề

**1.Giới thiệu**

**1.1 Mục đích**

Tài liệu này mô tả các yêu cầu phần mềm cho hệ thống Dự báo mưa minh phiên bản 1.0. Mục tiêu của tài liệu là cung cấp một đặc tả đầy đủ và chi tiết về các chức năng, hiệu năng, giao diện và ràng buộc của hệ thống nhằm hỗ trợ việc thiết kế, phát triển, kiểm thử và bảo trì phần mềm.

Phạm vi của hệ thống bao gồm việc thu thập dữ liệu môi trường từ các cảm biến (nhiệt độ, độ ẩm, độ ẩm đất, v.v.), xử lý dữ liệu và sử dụng các thuật toán học máy (machine learning) để đưa ra dự báo thời tiết như khả năng có mưa, nhiệt độ trung bình, độ ẩm dự kiến,… Hệ thống được thiết kế để triển khai trên các nền tảng nhúng sử dụng MicroPython và có thể tích hợp với các thiết bị IoT hoặc giao diện hiển thị kết quả.

Tài liệu SRS này chỉ tập trung vào phần mềm lõi của hệ thống, bao gồm việc thu thập dữ liệu, xử lý và dự báo. Các thành phần phần cứng (như cảm biến, vi điều khiển) và phần giao diện người dùng sẽ được mô tả ở tài liệu riêng biệt nếu cần.

**1.2 Quy ước tài liệu**

Tài liệu này tuân theo định dạng và tiêu chuẩn trình bày của mẫu tài liệu SRS do IEEE (Institute of Electrical and Electronics Engineers) đề xuất. Các quy ước cụ thể được sử dụng trong tài liệu bao gồm:

- Phông chữ: Văn bản sử dụng phông Times New Roman, cỡ chữ 13 cho nội dung chính và cỡ chữ 14 cho tiêu đề mục.

- Tô đậm được sử dụng để nhấn mạnh tên thành phần hệ thống, thuật ngữ quan trọng hoặc các tiêu đề cấp con.

- In nghiêng được dùng để làm nổi bật các ví dụ hoặc từ khóa kỹ thuật.

- Gạch đầu dòng liệt kê các chức năng hoặc yêu cầu nhằm tăng tính rõ ràng.

- Các yêu cầu phần mềm sẽ được đánh số theo cấu trúc phân cấp (ví dụ: 3.2.1, 3.2.2...) để dễ dàng tra cứu.

- Mỗi yêu cầu đều có độ ưu tiên riêng biệt (Cao, Trung bình, Thấp) và loại yêu cầu (chức năng, phi chức năng, ràng buộc thiết kế,...).

- Các thuật ngữ chuyên ngành sẽ được định nghĩa rõ ràng trong phần “1.4 Định nghĩa, từ viết tắt và từ viết tắt chuyên ngành”.

Các yêu cầu cấp cao hơn không mặc định được kế thừa bởi các yêu cầu chi tiết. Mỗi yêu cầu sẽ được đánh giá và ghi rõ mức độ ưu tiên độc lập để đảm bảo tính chính xác và toàn diện trong quá trình phát triển.

**1.3 Đối tượng và hướng dẫn đọc**

Tài liệu này được xây dựng để phục vụ cho nhiều nhóm đối tượng khác nhau tham gia vào quá trình phát triển, triển khai và sử dụng hệ thống Dự báo mưa. Các đối tượng chính bao gồm:

- Nhà phát triển phần mềm: Sử dụng tài liệu để hiểu rõ các yêu cầu chức năng và phi chức năng, nhằm hiện thực hóa hệ thống đúng với mục tiêu đặt ra.

- Quản lý dự án: Theo dõi tiến độ và đảm bảo các yêu cầu được đáp ứng đầy đủ theo kế hoạch phát triển.

- Người kiểm thử (Tester): Xác định các trường hợp kiểm thử phù hợp dựa trên các yêu cầu cụ thể được liệt kê trong tài liệu.

- Nhân viên kỹ thuật: Hỗ trợ vận hành, bảo trì và triển khai hệ thống dựa trên các thông tin về thiết kế và yêu cầu kỹ thuật.

- Người dùng cuối (End-user): Có thể tham khảo phần mô tả tổng quan để hiểu chức năng và phạm vi hoạt động của hệ thống.

- Nhà phân tích nghiệp vụ (BA) và nhân viên viết tài liệu kỹ thuật: Sử dụng để bảo đảm tính nhất quán giữa yêu cầu kỹ thuật và tài liệu hướng dẫn sử dụng hoặc đào tạo người dùng.

Gợi ý đọc tài liệu:

Tài liệu được tổ chức theo chuẩn IEEE và bao gồm các phần sau:

- Giới thiệu: Trình bày mục tiêu, phạm vi, định nghĩa, đối tượng và cách sử dụng tài liệu.

- Mô tả tổng quan: Cung cấp cái nhìn toàn diện về hệ thống, bao gồm các chức năng chính, ràng buộc, giả định và phụ thuộc.

- Yêu cầu cụ thể: Trình bày chi tiết các yêu cầu chức năng, phi chức năng, giao diện, và các ràng buộc hệ thống.

- Phụ lục: Cung cấp các tài liệu tham khảo, định nghĩa thuật ngữ, hoặc biểu đồ hỗ trợ.

Trình tự đọc khuyến nghị:

- Tất cả độc giả nên bắt đầu với phần Giới thiệu để nắm tổng quan.

- Nhà phát triển và tester nên tập trung vàoYêu cầu cụ thể để hiểu rõ yêu cầu kỹ thuật.

- Quản lý dự án nên đọc cả Giới thiệu,Mô tả tổng quan và các phần chính trong Yêu cầu cụ thể để giám sát tiến độ và đánh giá mức độ ưu tiên của các yêu cầu.

- Người dùng cuối chỉ cần đọc Giới thiệu và phần mô tả chức năng chính trong Mô tả tổng quan để nắm vai trò và lợi ích hệ thống mang lại.

**1.4 Phạm vi dự án**

Hệ thống Dự báo mưa là một ứng dụng phần mềm tích hợp các cảm biến môi trường (nhiệt độ, độ ẩm, độ ẩm đất, v.v.), thuật toán phân tích dữ liệu và mô hình học máy (Machine Learning) để thu thập, xử lý và dự đoán các hiện tượng thời tiết như mưa, độ ẩm không khí, và nhiệt độ trong tương lai gần.

Mục tiêu của hệ thống là cung cấp các cảnh báo và dự báo thời tiết chính xác theo thời gian thực nhằm hỗ trợ người dùng trong các hoạt động liên quan đến môi trường, nông nghiệp, và sinh hoạt hằng ngày. Hệ thống được thiết kế để hoạt động trên thiết bị nhúng (MicroPython), có khả năng truyền dữ liệu qua UART hoặc giao tiếp không dây, và tích hợp với một hệ thống hiển thị hoặc lưu trữ trung tâm.

Lợi ích chính của hệ thống:

- Nâng cao hiệu quả quản lý nông nghiệp thông minh.

- Giảm thiểu thiệt hại do thời tiết bất thường bằng cảnh báo sớm.

- Tăng tính tự động hóa trong việc thu thập và xử lý dữ liệu môi trường.

- Hỗ trợ ra quyết định dựa trên dữ liệu thời gian thực và mô hình học máy.

**1.5 Tài liệu tham khảo**

Tài liệu SRS này tham chiếu đến các tài liệu và nguồn tham khảo sau nhằm đảm bảo tính thống nhất về tiêu chuẩn, công nghệ và định hướng phát triển phần mềm:

- Sách, tài liệu về IoT: Các nguyên lý về cảm biến và giao tiếp không dây.

- Hướng dẫn sử dụng cảm biến nhiệt độ & độ ẩm:

BME280 đo nhiệt độ, độ ẩm và áp suất không khi

- Tài liệu về lập trình ứng dụng di động (Android/iOS).

- Hướng dẫn bảo mật dữ liệu IoT để đảm bảo an toàn cho người dùng.

**2. Mô tả chung**

**2.1 Góc nhìn sản phẩm**

Hệ thống Dự báo mưa là một sản phẩm mới, độc lập, được phát triển nhằm mục tiêu thu thập, phân tích và dự đoán điều kiện thời tiết dựa trên dữ liệu thực tế từ môi trường. Sản phẩm là sự kết hợp của công nghệ IoT (Internet of Things) với trí tuệ nhân tạo (AI), sử dụng các cảm biến Bme28 cùng với mô hình hồi quy tuyến tính nhằm đưa ra các dự đoán thời tiết chính xác hơn cho các ứng dụng nông nghiệp hoặc môi trường học đường.

Hệ thống bao gồm hai thành phần chính:

- Phần cứng nhúng: bao gồm các vi điều khiển (chạy MicroPython), các cảm biến môi trường và mô-đun truyền thông (UART/Wi-Fi).

-Phần mềm xử lý và phân tích dữ liệu: bao gồm mô hình học máy chạy trên nền tảng máy chủ hoặc máy tính cục bộ, nơi xử lý dữ liệu thu thập được từ thiết bị và hiển thị kết quả cho người dùng.

**2.2 Tính năng sản phẩm**

hệ thống dự báo mưa cung cấp một tập hợp các chức năng chính nhằm thu thập dữ liệu môi trường, xử lý và dự đoán thời tiết, từ đó hỗ trợ người dùng đưa ra quyết định phù hợp. Các tính năng chính bao gồm:

* Thu thập dữ liệu môi trường theo thời gian thực

- Ghi nhận các thông số như nhiệt độ, độ ẩm, áp xuất không khí Bme280.

- Dữ liệu được truyền về máy chủ thông qua UART hoặc Wi-Fi.

* Lưu trữ và quản lý dữ liệu

- Lưu trữ dữ liệu cảm biến dưới dạng tập tin hoặc cơ sở dữ liệu để phục vụ huấn luyện và kiểm thử mô hình.

- Cho phép truy xuất dữ liệu theo ngày, giờ hoặc khu vực.

* Phân tích và dự đoán thời tiết bằng mô hình học máy

- Sử dụng thuật toán hồi quy tuyến tính để dự đoán các chỉ số như khả năng mưa, độ ẩm trong những giờ sắp tới.

- Cập nhật mô hình định kỳ dựa trên dữ liệu mới để cải thiện độ chính xác.

* Hiển thị thông tin qua giao diện người dùng

Cung cấp giao diện web đơn giản cho người dùng để xem:

- Dữ liệu hiện tại (real-time)

- Kết quả dự đoán (forecast)

- Biểu đồ thời gian (nhiệt độ, độ ẩm, v.v.)

* Gửi cảnh báo và khuyến nghị

- Gửi thông báo (email, pop-up, hoặc tin nhắn) khi phát hiện điều kiện thời tiết bất thường như nhiệt độ quá cao, nguy cơ hạn hán, hoặc khả năng mưa cao.

* Khả năng mở rộng và tích hợp

- Cho phép tích hợp với hệ thống điều khiển tự động (tưới tiêu, cảnh báo đèn báo).

- Cấu trúc phần mềm và giao tiếp được thiết kế theo mô-đun giúp dễ dàng mở rộng tính năng hoặc cập nhật phần cứng.

**2.3 Lớp người dùng và đặc điểm**

Hệ thống Dự báo mưa được thiết kế để phục vụ nhiều lớp người dùng khác nhau. Mỗi lớp có nhu cầu sử dụng, kỹ năng và quyền truy cập riêng biệt. Việc phân loại này giúp đảm bảo các chức năng và giao diện được tùy chỉnh phù hợp với từng nhóm.

1. Người dùng phổ thông (nông dân, người dân địa phương)

Tần suất sử dụng: Hằng ngày hoặc định kỳ theo mùa vụ.

Chức năng truy cập:

- Xem thông tin thời tiết hiện tại.

- Xem dự báo và khuyến nghị (ví dụ: khả năng mưa, độ ẩm,..).

- Nhận cảnh báo thời tiết bất thường.

- Trình độ kỹ thuật: Cơ bản, không cần kiến thức chuyên sâu.

- Đặc điểm: Giao diện phải đơn giản, trực quan, hỗ trợ ngôn ngữ địa phương và dễ sử dụng trên thiết bị di động.

- Ưu tiên: Cao – Đây là lớp người dùng chính, ảnh hưởng trực tiếp đến hiệu quả ứng dụng hệ thống.

1. Kỹ thuật viên/Quản trị viên kỹ thuật

Tần suất sử dụng: Hằng ngày hoặc mỗi khi cần bảo trì hệ thống.

Chức năng truy cập:

- Cấu hình cảm biến và hệ thống phần cứng.

- Kiểm tra tình trạng truyền dữ liệu.

- Quản lý lưu trữ dữ liệu và sao lưu.

- Trình độ kỹ thuật: Trung bình đến cao, có kiến thức về phần cứng và phần mềm nhúng

- Đặc điểm: Cần giao diện quản trị riêng, yêu cầu xác thực và phân quyền nâng cao.

- Ưu tiên: Cao – Đảm bảo hệ thống hoạt động ổn định.

1. Nhà phát triển hệ thống

Tần suất sử dụng: Không thường xuyên, chủ yếu trong giai đoạn triển khai, cập nhật hoặc mở rộng.

Chức năng truy cập:

- Xem và chỉnh sửa mã nguồn

- Cập nhật mô hình dự đoán.

- Phân tích dữ liệu và cải tiến thuật toán.

- Trình độ kỹ thuật: Cao – có kiến thức lập trình, AI và xử lý dữ liệu.

- Đặc điểm: Cần truy cập toàn bộ hệ thống, bao gồm backend và mã mô hình.

- Ưu tiên: Trung bình – Cần thiết trong giai đoạn phát triển và bảo trì.

1. Cơ quan quản lý hoặc tài trợ (nếu có)

Tần suất sử dụng: Không thường xuyên.

Chức năng truy cập:

- Xem báo cáo tổng hợp.

- Đánh giá hiệu quả sử dụng của hệ thống.

- Trình độ kỹ thuật: Trung bình – không cần thao tác kỹ thuật.

- Đặc điểm: Cần các báo cáo dạng biểu đồ hoặc bảng tổng hợp.

- Ưu tiên: Thấp đến trung bình – phụ thuộc vào mục tiêu triển khai dự án.

**2.4 Môi trường hoạt động**

hệ thống dự báo mưa được triển khai trong một môi trường tích hợp giữa phần cứng nhúng, truyền dữ liệu và nền tảng phần mềm phân tích. Các thành phần chính của môi trường hoạt động bao gồm:

1. Phần cứng

- Bo mạch điều khiển: ESP32 (hoặc tương đương hỗ trợ Wi-Fi/UART).

- Cảm biến môi trường:

- Bme280) – đo nhiệt độ,độ ẩm và áp xuất không khí.

- Nguồn điện: Sử dụng pin hoặc nguồn DC 5V/3.3V.

1. Hệ điều hành và môi trường nhúng

- Firmware: MicroPython (phiên bản từ 1.19 trở lên).

- Giao tiếp dữ liệu: UART, Wi-Fi

1. Hệ thống phần mềm phân tích

Máy chủ nhận dữ liệu và xử lý:

- Hệ điều hành: Ubuntu 20.04 (hoặc Windows 10+ cho môi trường thử nghiệm).

- Ngôn ngữ lập trình: Python 3.8+

- Thư viện: NumPy, Pandas, scikit-learn, matplotlib (cho phân tích và mô hình dự báo).

Giao diện người dùng:

- Web-based (sử dụng Flask/Django hoặc Node.js).

- Tương thích trình duyệt phổ biến như Chrome, Firefox.

- Hỗ trợ thiết bị di động (responsive design).

1. Cơ sở dữ liệu (tùy chọn)

- Hệ quản trị: SQLite hoặc MySQL.

- Mục đích: Lưu trữ dữ liệu cảm biến lịch sử và kết quả dự báo để phân tích hoặc truy xuất theo yêu cầu.

1. Yêu cầu tương thích và tích hợp

- Phần mềm phải hoạt động trơn tru trên cả máy tính cá nhân và thiết bị nhúng.

- Tương thích với giao thức truyền thông UART/Wi-Fi.

- Dữ liệu được truyền phải ở định dạng JSON hoặc dạng khối dữ liệu có cấu trúc rõ ràng.

**2.5 Hạn chế về thiết kế và triển khai**

Trong quá trình phát triển hệ thống Dự báo mưa, các nhà phát triển sẽ phải tuân thủ một số hạn chế và yêu cầu cụ thể. Những yếu tố này có thể ảnh hưởng đến các quyết định thiết kế và triển khai của hệ thống:

1. Hạn chế về phần cứng

- Bộ nhớ hạn chế: Do việc sử dụng phần cứng nhúng (ESP32), bộ nhớ RAM và bộ nhớ trong của thiết bị rất hạn chế. Điều này yêu cầu tối ưu hóa mã nguồn, giảm thiểu bộ nhớ cần thiết và sử dụng các thuật toán xử lý dữ liệu nhẹ.

-Yêu cầu về nguồn điện: Hệ thống yêu cầu nguồn cấp điện ổn định (DC 5V hoặc 3.3V), và cần có khả năng tiết kiệm năng lượng trong trường hợp hoạt động dài hạn.

- Kết nối mạng: Hệ thống sử dụng Wi-Fi cho việc truyền tải dữ liệu. Tuy nhiên, độ ổn định của mạng có thể là yếu tố ảnh hưởng đến khả năng thu thập và truyền tải dữ liệu theo thời gian thực.

1. Giao thức truyền thông

- Giao thức UART: Mặc dù UART là lựa chọn phổ biến trong các hệ thống nhúng, tốc độ truyền tải và khoảng cách truyền có thể bị hạn chế nếu không có các biện pháp tăng cường (ví dụ: sử dụng module chuyển đổi tín hiệu).

- Giao thức Wi-Fi: Hệ thống sẽ truyền tải dữ liệu lên máy chủ qua Wi-Fi, yêu cầu kết nối ổn định và bảo mật, đồng thời phải có biện pháp xử lý khi mất kết nối.

1. Yêu cầu về ngôn ngữ và công cụ phát triển

- Ngôn ngữ lập trình: MicroPython được sử dụng cho lập trình nhúng trên ESP32, trong khi Python 3.8+ sẽ được dùng cho máy chủ phân tích và xử lý dữ liệu. Việc duy trì sự tương thích giữa hai môi trường này là một yêu cầu quan trọng.

- Công cụ phát triển: Các công cụ như Thonny IDE (cho MicroPython) và PyCharm/VS Code (cho Python) sẽ được sử dụng để phát triển, debug và kiểm tra mã nguồn.

1. Hạn chế về thời gian

- Thời gian thực thi dự báo: Hệ thống cần phải thực hiện các dự báo trong thời gian thực hoặc gần thời gian thực, do đó việc tối ưu hóa tốc độ xử lý dữ liệu từ các cảm biến và tính toán dự báo sẽ là yếu tố quan trọng.

- Cập nhật dữ liệu liên tục: Dữ liệu từ các cảm biến sẽ được thu thập và truyền tải theo chu kỳ, yêu cầu hệ thống có khả năng xử lý và cập nhật liên tục mà không gây gián đoạn.

1. Bảo mật và quyền truy cập

- Mã hóa dữ liệu: Dữ liệu cảm biến và dự báo cần phải được mã hóa khi truyền qua mạng để đảm bảo an toàn và bảo mật.

- Quyền truy cập hệ thống: Các giao diện người dùng sẽ có các cấp quyền khác nhau (quản trị viên, người dùng phổ thông) để hạn chế quyền truy cập vào các dữ liệu và chức năng quan trọng.

1. Tiêu chuẩn và quy ước thiết kế

- Tiêu chuẩn lập trình: Phải tuân thủ các tiêu chuẩn lập trình như PEP 8 (Python) và các quy ước đặt tên biến, hàm rõ ràng để dễ dàng bảo trì và nâng cấp trong tương lai.

- Quy ước thiết kế giao diện: Giao diện người dùng sẽ tuân thủ các quy ước thiết kế đơn giản và trực quan, dễ sử dụng cho người dùng không chuyên môn.

1. Tính tương thích với hệ thống khác

- Khả năng mở rộng: Hệ thống cần có khả năng mở rộng để dễ dàng tích hợp thêm các cảm biến mới hoặc thêm các chức năng như cảnh báo dự báo thời tiết hoặc dự báo chính xác hơn trong tương lai.

**2.6 Tài liệu hướng dẫn sử dụng**

Hệ thống Dự báo mưa sẽ đi kèm với các tài liệu hướng dẫn sử dụng giúp người dùng và các nhà phát triển dễ dàng sử dụng và duy trì phần mềm. Các thành phần tài liệu hướng dẫn sử dụng bao gồm:

1. Hướng dẫn sử dụng hệ thống

Mục đích: Cung cấp cho người dùng cuối một hướng dẫn chi tiết về cách sử dụng hệ thống, bao gồm cài đặt phần mềm, kết nối thiết bị cảm biến, và xem kết quả dự báo thời tiết.

Nội dung: Hướng dẫn này sẽ bao gồm các phần sau:

- Cài đặt và cấu hình phần mềm.

- Cách kết nối và sử dụng cảm biến (Bme280).

- Hướng dẫn sử dụng giao diện người dùng để xem dữ liệu và dự báo.

- Cách bảo trì và kiểm tra hệ thống.

1. Trợ giúp trực tuyến

Mục đích: Cung cấp thông tin nhanh chóng và dễ dàng cho người dùng về các vấn đề gặp phải trong quá trình sử dụng.

Nội dung: Tài liệu trợ giúp trực tuyến sẽ bao gồm các bài viết thường gặp, giải thích về lỗi, hướng dẫn khắc phục sự cố, và các vấn đề liên quan đến cài đặt hoặc sử dụng hệ thống.

1. Tài liệu hướng dẫn cài đặt phần cứng

Mục đích: Hướng dẫn người dùng hoặc kỹ thuật viên về cách lắp ráp và cài đặt các thành phần phần cứng của hệ thống.

Nội dung: Tài liệu này sẽ giải thích chi tiết cách kết nối các cảm biến, thiết bị và phần cứng nhúng (ESP32), bao gồm sơ đồ kết nối và các bước kiểm tra sau khi cài đặt.

1. Tài liệu đào tạo cho người dùng cuối

Mục đích: Cung cấp các tài liệu đào tạo để người dùng cuối hiểu rõ hơn về hệ thống và cách sử dụng nó trong các tình huống thực tế.

Nội dung: Tài liệu này sẽ bao gồm các chủ đề như:

- Giới thiệu về hệ thống và các chức năng chính.

- Cách sử dụng dữ liệu dự báo để đưa ra quyết định trong việc trồng trọt hoặc dự báo mưa.

- Các ví dụ tình huống và ứng dụng thực tế.

**2.7 Giả định và sự phụ thuộc**

Trong quá trình xây dựng và vận hành hệ thống dự báo mưa, một số giả định và sự phụ thuộc sau đây được đưa ra. Những yếu tố này có thể ảnh hưởng đến tính khả thi, hiệu quả hoặc độ chính xác của hệ thống nếu có thay đổi hoặc sai lệch so với thực tế.

Giả định:

- Kết nối mạng ổn định:

Hệ thống được giả định sẽ vận hành trong môi trường có kết nối Wi-Fi ổn định để có thể truyền dữ liệu từ các cảm biến đến máy chủ hoặc đám mây phục vụ cho việc xử lý và hiển thị.

- Cảm biến hoạt động chính xác và liên tục:

Các cảm biến Bme280 hoạt động trong điều kiện tiêu chuẩn và không bị hư hỏng trong quá trình sử dụng.

- Dữ liệu thu thập có độ tin cậy:

Các giá trị cảm biến được giả định là đúng, không nhiễu, không lỗi và không cần xử lý sai số lớn.

- Người dùng có hiểu biết cơ bản về thiết bị IoT:

Người dùng được giả định có khả năng kết nối thiết bị vào nguồn điện, mạng và biết cách truy cập giao diện người dùng để xem dữ liệu dự báo.

- Nền tảng phần cứng ESP32 có đủ tài nguyên:

Vi điều khiển ESP32 có đủ RAM, bộ nhớ và tốc độ xử lý để thu thập, xử lý dữ liệu và truyền thông qua UART hoặc Wi-Fi.

Sự phụ thuộc:

- Thư viện phần mềm bên thứ ba:

Dự án phụ thuộc vào một số thư viện MicroPython như:

- dht (đọc cảm biến Bme280),

- machine, time (điều khiển phần cứng),

- urequests hoặc giao thức UART/Wi-Fi để truyền dữ liệu.

Nếu có thay đổi về thư viện hoặc không tương thích với phiên bản firmware, hệ thống có thể cần điều chỉnh mã nguồn.

- Hệ điều hành hoặc môi trường IDE:

Dự án được phát triển và triển khai trên nền MicroPython, sử dụng công cụ như Thonny IDE hoặc uPyCraft. Việc thay đổi môi trường có thể yêu cầu cấu hình lại hệ thống.

- Sự phụ thuộc vào mô hình học máy được huấn luyện:

Mô hình hồi quy tuyến tính dùng để dự đoán mưa được giả định là đã được huấn luyện chính xác từ dữ liệu lịch sử. Nếu dữ liệu huấn luyện không đủ hoặc không phù hợp, kết quả dự đoán có thể thiếu chính xác.

- Phụ thuộc vào nguồn điện ổn định:

Các cảm biến và vi điều khiển hoạt động ổn định chỉ khi được cấp nguồn đúng mức (thường 3.3V hoặc 5V). Việc mất điện hoặc dao động nguồn có thể ảnh hưởng đến hoạt động hệ thống.

**3. Tính năng hệ thống**

**3.1 Tính năng: Thu thập dữ liệu cảm biến thời tiết**

* Mô tả và ưu tiên

Hệ thống phải thu thập dữ liệu từ các cảm biến môi trường như nhiệt độ, độ ẩm không khí (Bme280). Dữ liệu sẽ được cập nhật định kỳ để phục vụ việc dự đoán.

Mức độ ưu tiên: Cao

* Chuỗi kích thích/phản ứng

Người dùng cấp nguồn cho hệ thống → Hệ thống tự động khởi tạo các cảm biến.

Mỗi chu kỳ thời gian định sẵn (ví dụ: 5 giây) → ESP32 truy vấn dữ liệu từ cảm biến → Dữ liệu được lưu tạm và gửi đến hệ thống dự đoán.

* Yêu cầu chức năng

- Hệ thống phải lấy dữ liệu nhiệt độ và độ ẩm từ cảm biến Bme280 mỗi 5 giây.

- Hệ thống phải lấy giá trị độ ẩm đất từ cảm biến analog và chuyển đổi sang định dạng số.

- Hệ thống phải đọc trạng thái ON/OFF từ cảm biến chạm và lưu lại.

- Trong trường hợp cảm biến không phản hồi, hệ thống phải ghi log lỗi và tiếp tục vòng lặp lấy dữ liệu trong chu kỳ tiếp theo.

**3.2 Tính năng: Dự đoán khả năng mưa bằng mô hình hồi quy**

* Mô tả và ưu tiên

Hệ thống sử dụng mô hình hồi quy tuyến tính để dự đoán khả năng mưa dựa trên các thông số đo được từ cảm biến.  
Mức độ ưu tiên: Cao

* Chuỗi kích thích/phản ứng

Sau mỗi lần thu thập dữ liệu cảm biến đầy đủ → Hệ thống gọi mô hình hồi quy với đầu vào là nhiệt độ, độ ẩm không khí và độ ẩm đất → Kết quả trả về là xác suất có mưa hoặc nhãn "Rain"/"No Rain" → Hiển thị kết quả.

* Yêu cầu chức năng

- Hệ thống phải truyền dữ liệu đầu vào đến mô hình hồi quy và nhận kết quả dự đoán.

- Mô hình dự đoán phải được huấn luyện trước và lưu trữ trên hệ thống hoặc máy chủ.

- Nếu mô hình không trả về kết quả trong 3 giây, hệ thống phải hiển thị lỗi “Mất kết nối mô hình” và ghi log.

**3.3 Tính năng: Giao tiếp UART/Wi-Fi và hiển thị dữ liệu**

* Mô tả và ưu tiên

Hệ thống có thể truyền dữ liệu ra thiết bị khác qua UART hoặc giao tiếp Wi-Fi

(nếu khả dụng). Đồng thời, hiển thị dữ liệu đo và kết quả dự báo qua màn hình hoặc gửi đến máy chủ.  
Mức độ ưu tiên: Trung bình

* Chuỗi kích thích/phản ứng

- Người dùng bật nguồn → Hệ thống khởi tạo giao tiếp UART/Wi-Fi.

- Mỗi lần thu thập và dự đoán hoàn tất → Dữ liệu được gửi đến thiết bị nhận và hiển thị trên màn hình hoặc ứng dụng web.

* Yêu cầu chức năng

- Hệ thống phải định dạng dữ liệu theo JSON và truyền qua UART đến thiết bị giám sát.

- Hệ thống phải hỗ trợ kết nối Wi-Fi và gửi dữ liệu lên máy chủ qua HTTP nếu có cấu hình sẵn.

- Trong trường hợp mất kết nối, dữ liệu cần được lưu tạm và gửi lại khi có mạng.

**4 - Yêu cầu về giao diện bên ngoài**

**4.1 Giao diện người dùng**

hệ thống dự báo mưa sẽ có hai loại giao diện người dùng chính:

1. Giao diện trên thiết bị phần cứng (MicroPython ESP32):

- Hiển thị cơ bản trên màn hình LCD nhỏ gắn kèm (nếu có).

Thông tin hiển thị:

- Nhiệt độ hiện tại (°C)

- Độ ẩm không khí (%)

- Áp xuất(hPa)

- Trạng thái cảm biến chạm (bật/tắt)

- Kết quả dự báo mưa: “RAIN” hoặc “NO RAIN”

- Cập nhật định kỳ sau mỗi chu kỳ đo (ví dụ: 5 giây).

- Các biểu tượng đơn giản (mặt trời/mưa) có thể được hiển thị kèm theo để trực quan hóa.

1. Giao diện web (nếu mở rộng kết nối Wi-Fi):

- Thiết kế giao diện dạng Dashboard đơn giản.

Các thành phần chính:

- Biểu đồ nhiệt độ và độ ẩm theo thời gian

- Bản dữ liệu dự báo gần nhất.

- Trạng thái thiết bị (kết nối, nguồn, trạng thái cảm biến).

- Các nút điều khiển: “Cập nhật thủ công”, “Tải lại dữ liệu”, “Xuất CSV”.

**4.2 Giao diện phần cứng**

Phần mềm của hệ thống dự báo mưa tương tác trực tiếp với nhiều cảm biến vật lý và thiết bị thông qua vi điều khiển ESP32 chạy MicroPython. Các giao diện phần cứng chính bao gồm:

1. ESP32 (Bộ điều khiển trung tâm):

Vai trò: Xử lý tín hiệu cảm biến, tính toán dự báo và xuất dữ liệu.

Giao tiếp nội bộ:

- Giao tiếp GPIO với cảm biến chạm.

- Giao tiếp I2C/SPI với màn hình LCD (nếu có)

- Giao tiếp UART để gửi dữ liệu sang máy tính hoặc thiết bị giám sát.

- Giao tiếp Wi-Fi (qua TCP/IP) nếu có kết nối mạng.

1. Bme280 (Cảm biến nhiệt độ, độ ẩm và áp xuất không khí):

- Kết nối: Một chân tín hiệu kết nối đến chân GPIO bất kỳ trên ESP32.

- Giao thức: 1-wire (giao tiếp số).

- Tần suất đọc: 1–2 giây/lần.

- Dữ liệu truyền: Nhiệt độ (°C), độ ẩm (%) và áp xuất(hPa)

1. Màn hình LCD (nếu có):

- Kết nối: Giao tiếp I2C (SCL, SDA).

- Dữ liệu xuất: Thông tin môi trường, trạng thái dự báo.

- Tần suất cập nhật: Theo chu kỳ đo (ví dụ: mỗi 5 giây).

1. UART Interface (Kết nối ngoài):

- Mục đích: Gửi dữ liệu thời gian thực tới máy tính hoặc gateway khác.

- Tốc độ baud: 9600 hoặc 115200 bps.

- Định dạng dữ liệu: JSON hoặc CSV (có thể cấu hình).

**4.3 Giao diện phần mềm**

hệ thống dự báo mưa tích hợp với nhiều thành phần phần mềm khác để thực hiện các chức năng thu thập, xử lý, lưu trữ và truyền dữ liệu. Các giao diện phần mềm

chính bao gồm:

1. Hệ điều hành và nền tảng

- Tên: MicroPython

- Vai trò: Nền tảng chính chạy trên vi điều khiển ESP32. Cung cấp thư viện lập trình và API để tương tác với cảm biến, UART, và Wi-Fi.

- Giao diện: Dựa trên script Python (\*.py) với cấu trúc module rõ ràng.

1. Thư viện phần mềm tích hợp

| Tên thư viện | Mục đích sử dụng | Phiên bản / Nguồn |
| --- | --- | --- |
| machine | Giao tiếp với GPIO, ADC, UART | MicroPython core |
| bme280.py | Đọc dữ liệu từ cảm biến Bme280 | MicroPython core |
| time | Delay và thời gian hệ thống | MicroPython core |
| network | Kết nối Wi-Fi nếu cần (dự phòng truyền dữ liệu) | MicroPython core |

1. Giao diện người dùng phần mềm (tùy chọn mở rộng)

Công cụ: Web App có thể được tích hợp sau.

API sử dụng: Có thể tích hợp REST API hoặc MQTT nếu muốn mở rộng hệ thống.

Thông tin chia sẻ: Các chỉ số môi trường theo thời gian thực và kết quả dự báo.

**4.4 Giao diện truyền thông**

1. Giao thức UART (Giao tiếp nối tiếp)

Mục đích: Truyền dữ liệu từ vi điều khiển ESP32 đến gateway hoặc hệ thống máy chủ giám sát.

Thông số kỹ thuật:

- Tốc độ truyền: 9600 hoặc 115200 baud (tùy cấu hình).

- Cấu hình khung truyền: 8-bit data, 1 stop bit, no parity.

- Dữ liệu truyền: Chuỗi JSON hoặc định dạng văn bản CSV.

- Đồng bộ hóa: Thực hiện theo luồng tuần tự, không đa luồng. Đồng bộ bằng cách gửi ký tự bắt đầu và kết thúc nếu cần.

1. Wi-Fi (Giao tiếp không dây - tùy chọn)

Mục đích: Truyền dữ liệu đến server hoặc đám mây (Firebase hoặc API HTTP).

Chuẩn giao tiếp: IEEE 802.11 b/g/n.

Giao thức truyền: HTTP POST hoặc MQTT (nếu cấu hình mở rộng).

Định dạng dữ liệu: JSON.

Bảo mật:

- Hỗ trợ mã hóa WPA2 cho kết nối Wi-Fi.

- HTTPS hoặc MQTT qua TLS sẽ được áp dụng nếu sử dụng các API công khai.

Tần suất gửi dữ liệu: Mỗi 5 phút hoặc theo sự kiện thay đổi dữ liệu đáng kể (event-driven)

1. Máy chủ Web (tùy chọn - cho cấu hình mở rộng)

Mô hình: ESP32 có thể lưu trữ một Web Server mini hiển thị dữ liệu cảm biến trong mạng nội bộ.

Giao thức: HTTP 1.1.

Đường dẫn: /status, /data.json.

Định dạng: HTML cho giao diện, JSON cho truy vấn API.

Bảo mật: Mật khẩu truy cập hoặc IP whitelist nội bộ (nếu yêu cầu).

**5 - Các yêu cầu không chức năng khác**

**5.1 Yêu cầu về hiệu suất**

Hệ thống dự báo mưa cần đảm bảo hiệu suất hoạt động ổn định trong các điều kiện khác nhau, bao gồm cả môi trường có tín hiệu không ổn định và hạn chế về năng lượng. Các yêu cầu cụ thể như sau:

1. Tần suất thu thập dữ liệu cảm biến

Yêu cầu: Hệ thống phải thu thập dữ liệu từ các cảm biến (nhiệt độ, độ ẩm không khí,áp xuất không khí) tối thiểu mỗi 60 giây một lần.

Lý do: Đảm bảo dữ liệu đủ thường xuyên để phát hiện sớm các điều kiện bất thường và phục vụ dự đoán chính xác hơn.

1. Thời gian phản hồi từ khi nhận dữ liệu đến khi truyền đi

Yêu cầu: Thời gian xử lý dữ liệu cảm biến và gửi kết quả ra giao diện UART hoặc lên mạng Wi-Fi không được vượt quá 2 giây.

Lý do: Đảm bảo hệ thống phản hồi gần thời gian thực và duy trì khả năng cập nhật liên tục

1. Dung lượng bộ nhớ và CPU

Yêu cầu:

Bộ nhớ RAM sử dụng không vượt quá 70% tổng bộ nhớ khả dụng của ESP32 để tránh tràn stack hoặc crash.

Thời gian CPU sử dụng cho việc dự đoán không quá 1 giây/lượt.

Lý do: Giữ cho thiết bị hoạt động ổn định và tiết kiệm năng lượng, nhất là khi chạy bằng pin hoặc năng lượng mặt trời

1. Dung lượng dữ liệu truyền tải

Yêu cầu: Dung lượng gói tin gửi qua Wi-Fi hoặc UART không được vượt quá 500 byte/lần truyền.

Lý do: Giảm tải băng thông và tránh nghẽn khi có nhiều thiết bị trong cùng hệ thống truyền thông.

1. Tính sẵn sàng của hệ thống

Yêu cầu: Hệ thống phải hoạt động liên tục với độ tin cậy tối thiểu 95% trong 24 giờ hoạt động.

Lý do: Đảm bảo giám sát liên tục môi trường phục vụ phân tích thời tiết.

1. Khả năng phục hồi

Yêu cầu: Khi xảy ra mất điện hoặc khởi động lại hệ thống, thiết bị phải khôi phục trạng thái hoạt động bình thường trong vòng 10 giây.

Lý do: Đảm bảo tính ổn định và sẵn sàng cao trong môi trường hoạt động không liên tục.

**5.2 Yêu cầu về an toàn**

Hệ thống dự báo mưa là một hệ thống thu thập và phân tích dữ liệu môi trường thực tế, vì vậy các yêu cầu an toàn liên quan đến phần cứng, dữ liệu và người dùng cần được xác định rõ ràng để tránh thiệt hại vật lý, sai lệch dữ liệu và nguy cơ bảo mật.

1. Bảo vệ phần cứng

Yêu cầu: Các cảm biến và vi điều khiển phải được đặt trong vỏ chống thấm nước và chống bụi (chuẩn tối thiểu IP54) để bảo vệ khỏi thời tiết khắc nghiệt.

Biện pháp: Sử dụng hộp bảo vệ ngoài trời có khả năng tản nhiệt tốt, tránh ẩm và chống rò điện.

1. Ngăn chặn quá nhiệt và chập mạch

Yêu cầu: Hệ thống phải có cơ chế tự ngắt nguồn nếu nhiệt độ vi điều khiển hoặc dòng điện vượt ngưỡng cho phép.

Lý do: Tránh cháy nổ hoặc hư hỏng linh kiện trong trường hợp môi trường quá nóng hoặc ngắn mạch.

1. Tính toàn vẹn dữ liệu

Yêu cầu: Phải có kiểm tra CRC (Cyclic Redundancy Check) hoặc checksum để xác minh tính toàn vẹn của dữ liệu trước khi sử dụng trong các thuật toán dự báo.

Lý do: Tránh sai lệch trong dự đoán thời tiết do dữ liệu lỗi hoặc bị thay đổi.

1. An toàn khi truyền thông

Yêu cầu: Nếu dữ liệu được truyền qua mạng (Wi-Fi), phải sử dụng giao thức mã hóa đơn giản như AES-128 hoặc bảo vệ bằng mật khẩu mạng WPA2

Lý do: Ngăn chặn truy cập trái phép hoặc thay đổi dữ liệu trong quá trình truyền.

1. Đáp ứng các tiêu chuẩn và quy định

Yêu cầu: Hệ thống phải tuân thủ các tiêu chuẩn an toàn điện áp thấp và thiết bị điện tử tiêu dùng (ví dụ: IEC 61010 hoặc RoHS nếu sản xuất thương mại).

Lý do: Đảm bảo người dùng cuối không bị ảnh hưởng bởi thiết bị lỗi hoặc điện áp nguy hiểm.

1. Giới hạn người dùng tác động

Yêu cầu: Giao diện người dùng (nếu có) phải giới hạn truy cập chỉnh sửa cấu hình kỹ thuật cho người dùng không chuyên, trừ khi có xác thực.

Biện pháp: Áp dụng phân quyền hoặc yêu cầu xác thực mật khẩu khi thay đổi cấu hình hệ thống.

**5.3 Yêu cầu bảo mật**

hệ thống dự báo mưa có thể thu thập và xử lý dữ liệu môi trường từ nhiều địa điểm, do đó các biện pháp bảo mật là bắt buộc để đảm bảo tính riêng tư, toàn vẹn và an toàn của dữ liệu.

1. Xác thực người dùng

Người dùng truy cập vào giao diện quản trị (nếu có) phải thực hiện đăng nhập với tên người dùng và mật khẩu.

Mật khẩu phải được lưu trữ dưới dạng mã hóa (hash) bằng thuật toán bảo mật như SHA-256.

1. Bảo vệ dữ liệu truyền tải

Dữ liệu từ cảm biến gửi đến máy chủ/ứng dụng phải được bảo vệ bằng giao thức có mã hóa (ví dụ: HTTPS, MQTT over TLS).

Thông tin nhạy cảm như vị trí hoặc dữ liệu dự báo chính xác theo khu vực không được gửi qua mạng công khai nếu không được mã hóa.

1. Quyền riêng tư dữ liệu

Không thu thập dữ liệu cá nhân người dùng.

Nếu cần lưu nhật ký truy cập hoặc hành vi hệ thống, phải có lựa chọn bật/tắt và xóa theo yêu cầu.

1. Bảo mật thiết bị đầu cuối

Firmware của vi điều khiển phải được khóa (read-protected) để ngăn can thiệp trái phép.

Các cổng giao tiếp vật lý (UART, SPI, I2C) không được để mở trong bản phát hành cuối.

1. Yêu cầu tuân thủ

Hệ thống phải tuân thủ các quy định bảo mật cơ bản theo chuẩn OWASP IoT Top 10 và GDPR (nếu áp dụng ở môi trường thực tế với dữ liệu người dùng liên quan).

**5.4 Thuộc tính chất lượng phần mềm**

1. Tính chính xác

- Dự đoán thời tiết phải đạt độ chính xác tối thiểu 85% so với dữ liệu thực tế trong vòng 24 giờ.

- Hệ thống cần sử dụng mô hình hồi quy học máy được huấn luyện định kỳ để cập nhật độ chính xác.

1. Tính khả dụng

- Giao diện (nếu có) phải có thời gian phản hồi dưới 1 giây cho mọi thao tác người dùng.

- Hệ thống nhúng phải có thời gian khởi động dưới 5 giây từ lúc cấp nguồn.

1. Tính bảo trì

- Mã nguồn phải được tổ chức theo mô-đun, có tài liệu kỹ thuật đi kèm.

- Cập nhật firmware thông qua cổng UART hoặc OTA (nếu triển khai Wi-Fi)

1. Tính di động

- Hệ thống có thể được triển khai trên nhiều nền tảng phần cứng khác nhau như ESP32, Raspberry Pi, STM32.

- Mã MicroPython có thể chạy trên trình thông dịch chuẩn và môi trường mô phỏng.

1. Tính kiểm tra được

- Mỗi chức năng chính đều có thể kiểm tra độc lập bằng tập lệnh kiểm thử đơn vị (unit test).

- Các đầu ra (như dự báo) có thể kiểm tra được bằng cách đối chiếu với dữ liệu khí tượng từ nguồn chuẩn.

1. Tính dễ sử dụng

- Người dùng không chuyên có thể thiết lập hệ thống qua giao diện đơn giản (nút bấm, đèn báo, hoặc giao diện web tùy chọn).

**6. Các yêu cầu khác**

1. Yêu cầu cơ sở dữ liệu

- Dữ liệu thu thập từ cảm biến (nhiệt độ, độ ẩm, độ ẩm đất, trạng thái mưa) phải được lưu trữ định kỳ vào cơ sở dữ liệu dưới dạng bản ghi thời gian thực.

- Cơ sở dữ liệu đề xuất sử dụng: SQLite cho hệ thống nhúng hoặc MySQL/PostgreSQL cho hệ thống có máy chủ backend.

Mỗi bản ghi phải bao gồm:

- ID cảm biến

- Thời gian ghi nhận (timestamp)

- Giá trị đo (nhiệt độ, độ ẩm, mưa, v.v.)

- Tình trạng pin (nếu có)

1. Mục tiêu tái sử dụng

Mô-đun thu thập dữ liệu và mô hình dự báo phải được thiết kế để tái sử dụng cho các mục đích:

- Cảnh báo sớm thiên tai.

- Quản lý tưới tiêu tự động.

- Dự báo thời tiết vi mô trong nông nghiệp chính xác (precision agriculture).

**Phụ lục A: Thuật ngữ**

| Thuật ngữ | Mô tả |
| --- | --- |
| Bme280 | Cảm biến nhiệt độ, độ ẩm và áp xuất kỹ thuật số |
| UART | Giao tiếp nối tiếp không đồng bộ giữa thiết bị và vi điều khiển |
| MicroPython | Phiên bản nhẹ của Python dùng cho vi điều khiển |
| TBD | To Be Determined - Chưa được xác định rõ |
| HTTPS | Giao thức truyền siêu văn bản có mã hóa (bảo mật) |
| OTA | Over The Air - Cập nhật firmware qua mạng không dây |

**Phụ lục B: Mô hình phân tích**

1. Sơ đồ luồng dữ liệu (DFD - Data Flow Diagram)

Hiển thị dòng chảy từ cảm biến → bộ xử lý dữ liệu → hệ thống dự đoán → lưu trữ → giao diện người dùng (nếu có).  
(Đính kèm sơ đồ minh họa nếu có)

2. Sơ đồ lớp (Class Diagram)

Các lớp bao gồm: SensorData, WeatherPredictor, DatabaseHandler, UserInterface.  
(Đính kèm sơ đồ UML nếu có)

3. Sơ đồ trạng thái (State Diagram)

Trạng thái chính của thiết bị: Ngủ → Thu thập dữ liệu → Dự đoán → Gửi dữ liệu → Ngủ.

**6.3 Phụ lục C: Danh sách vấn đề**

| ID | Mô tả vấn đề | Trạng thái |
| --- | --- | --- |
| TBD-01 | Lựa chọn chính thức giữa SQLite và PostgreSQL cho cơ sở dữ liệu | Đang xem xét |
| TBD-02 | Cơ chế cập nhật mô hình học máy trong môi trường nhúng hạn chế tài nguyên | Đang phân tích |
| TBD-03 | Có cần hỗ trợ MQTT cho truyền dữ liệu cảm biến không? | Đang thảo luận |
| TBD-04 | Hình thức triển khai OTA cho ESP32 cần xác nhận khả thi | Chưa xác định |