

IoT Platform

Phiên bản <0.1>

<13.01.2025>

I>	GIỚI THIỆU	4
1.1.	TỔNG QUAN	4
1.2	MỤC TIÊU TÀI LIỆU	4
1.3	PHẠM VI TÀI LIỆU.....	4
II>	KIẾN TRÚC.....	5
2.1.	KIẾN TRÚC TỔNG QUAN	5
2.2.	CÁC THÀNH PHẦN CHÍNH	5
1.	IoT Things (Thiết bị IoT)	5
2.	IoT Gateway (Cổng kết nối IoT)	5
3.	IoT Platform (Nền tảng IoT)	6
4.	Application (Ứng dụng).....	6
2.3.	LUỒNG DỮ LIỆU	6
1.	Dữ liệu được thu thập bởi các thiết bị IoT:	7
2.	IoT Gateway nhận dữ liệu và xử lý sơ bộ:	7
3.	Gateway gửi dữ liệu IoT Platform thông qua các giao thức như MQTT, AMQP, HTTP:	7
4.	IoT Platform xử lý, lưu trữ và phân tích dữ liệu:	8
5.	Các ứng dụng kết nối với IoT Platform thông qua REST API hoặc WebSocket để truy cập dữ liệu và cung cấp dịch vụ cho người dùng:.....	8
III>	CÁC CHỨC NĂNG CHÍNH	9
3.1.	QUẢN LÝ THIẾT BỊ IOT (DEVICE MANAGEMENT).....	9
3.2.	KẾT NỐI VÀ GIAO TIẾP THIẾT BỊ (DEVICE CONNECTIVITY)	9
3.3.	THU THẬP VÀ XỬ LÝ DỮ LIỆU (DATA INGESTION & PROCESSING) ..	9
3.4.	QUẢN LÝ SỰ KIỆN VÀ HÀNH ĐỘNG (EVENT & ACTION MANAGEMENT)	10
3.5.	BẢO MẬT VÀ QUẢN TRỊ HỆ THỐNG (SECURITY & GOVERNANCE)..	10

3.6. PHÂN TÍCH VÀ TRỰC QUAN HÓA DỮ LIỆU (DATA ANALYTICS & VISUALIZATION).....	11
IV> KẾ HOẠCH TRIỂN KHAI IoT PLATFORM GIAI ĐOẠN 1	11
4.1. MỤC TIÊU GIAI ĐOẠN 1	11
4.2. CÁC CÔNG VIỆC CHI TIẾT TRONG GIAI ĐOẠN 1	11
1. Thiết lập môi trường phát triển.....	11
2. Phát triển IoT Gateway.....	11
3. Phát triển IoT Platform (Core)	12
4. Phát triển Application (Cơ bản):	13
5. Kiểm thử:.....	13
6. Tài liệu:.....	13
7. Các điểm cần chú ý trong giai đoạn 1:	13
4.3. CƠ CẤU TỔ CHỨC ĐỘI NGŨ	14
1. Đội ngũ	14
2. Vai trò và trách nhiệm của từng vị trí:	14
3. Tổ chức làm việc và giao tiếp:.....	15
4. Phân bổ thời gian:.....	15
5. Chi phí, ngân sách	15

I> GIỚI THIỆU

1.1. TỔNG QUAN

IoT Platform là một nền tảng công nghệ được thiết kế để tự động hóa việc thu thập, xử lý, quản lý dữ liệu từ nhiều nguồn khác nhau nhằm hỗ trợ các ứng dụng kết nối với thiết bị IoT. Dữ liệu thu thập được từ IoT Platform được sử dụng để đưa ra quyết định, tạo lợi thế cạnh tranh với các đối thủ.

IoT Platform cung cấp một hạ tầng linh hoạt để làm việc với nhiều loại dữ liệu, từ dữ liệu thô (raw data) đến dữ liệu đã xử lý (processed data), và giúp tối ưu hóa các hoạt động kinh doanh, vận hành và phân tích.

Ngoài ra việc tích hợp IoT Platform vào vận hành sản xuất giúp doanh nghiệp bắt kịp với xu thế AI Machine Learning đang dần trở nên phổ biến toàn cầu. Dữ liệu được thu thập phân tích và xử lý thời gian thực với AI giúp tối giản hoá chu trình làm việc, nâng cao hiệu suất công việc từ đó tạo sức cạnh tranh ngày càng cao cho doanh nghiệp.

1.2 MỤC TIÊU TÀI LIỆU

Mục tiêu của tài liệu này là cung cấp một cái nhìn toàn diện về nền tảng IoT Platform, bao gồm các mục tiêu chính, phạm vi áp dụng và các yếu tố quan trọng trong việc thiết kế, triển khai và vận hành hệ thống.

Cung cấp góc nhìn kỹ thuật cụ thể về hệ thống, các khâu kiện cũng như cách thức hoạt động của nền tảng. Cùng với đó là cách thức triển khai hệ thống lên thực tế sản xuất của doanh nghiệp.

Tài liệu giúp quản lý, các nhóm phát triển và người dùng cuối hiểu rõ cách thức hoạt động của một hệ thống IoT, các tiêu chuẩn kỹ thuật và các quy trình cần thiết để đảm bảo dữ liệu được thu thập, lưu trữ, xử lý và phân tích một cách hiệu quả.

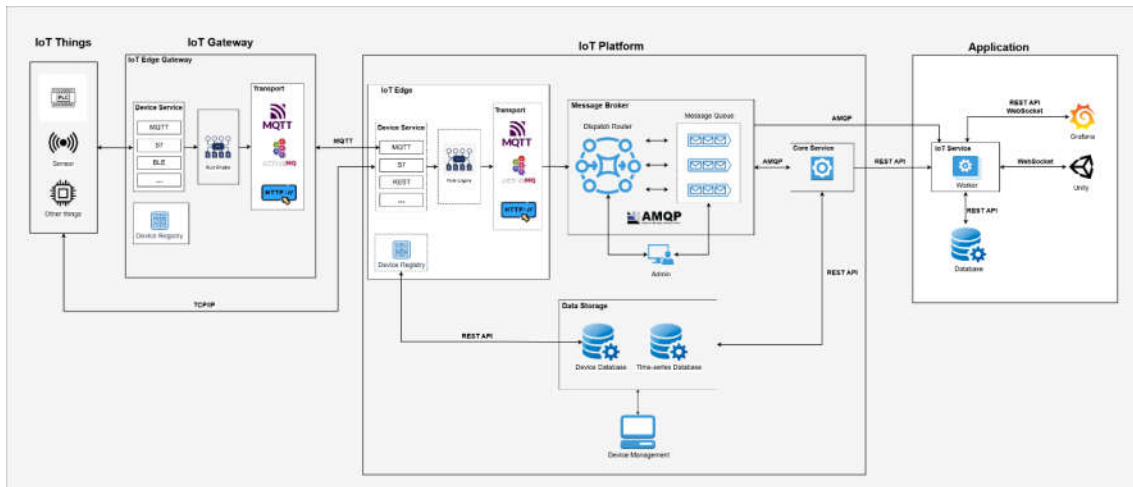
1.3 PHẠM VI TÀI LIỆU

Phạm vi của tài liệu bao gồm:

- Các tính năng kỹ thuật, vận hành của IoT platform. Tổng quan kiến trúc hệ thống.
- Kế hoạch triển khai hệ thống IoT với cụ thể chi tiết từng bước.
- Cách thức tổ chức nhân lực và nguồn lực để triển khai hệ thống.
- Tính năng kỹ thuật của nền tảng IoT.

II> KIẾN TRÚC

2.1. KIẾN TRÚC TỔNG QUAN



Hình 1: Kiến trúc tổng quan IoT Platform.

2.2. CÁC THÀNH PHẦN CHÍNH

Kiến trúc IoT Platform gồm 4 thành phần chính:

1. IoT Things (Thiết bị IoT)

Đây là lớp thấp nhất, bao gồm các thiết bị vật lý thu thập dữ liệu từ môi trường. Ví dụ: cảm biến (sensor), bộ điều khiển logic (PLC), và các thiết bị khác.

- Các thiết bị này sử dụng các giao thức công nghiệp như modbus, TCP/IP, Bacnet... để giao tiếp với IoT Gateway.

2. IoT Gateway (Cổng kết nối IoT)

Đóng vai trò là cầu nối giữa các thiết bị IoT và IoT Platform.

- **Device Service:** Quản lý kết nối và giao tiếp với các thiết bị. Chuyển đổi dữ liệu từ các giao thức khác nhau sang định dạng chuẩn.
- **Transport:** Sử dụng các giao thức như MQTT, AMQP, HTTP để truyền dữ liệu lên nền tảng IoT.
- **Rule Engine:** Thực hiện xử lý dữ liệu tại biên, giảm độ trễ và băng thông. Ví dụ: lọc dữ liệu, cảnh báo cục bộ.

- **Data Storage:** Lưu trữ tạm thời dữ liệu khi kết nối với nền tảng bị gián đoạn.
- **Device Registry:** Lưu trữ thông tin về các thiết bị kết nối với biên.
- **Security:** Thực hiện các biện pháp bảo mật tại biên, như xác thực thiết bị, mã hóa dữ liệu.
- **Management:** Cung cấp khả năng quản lý và giám sát các thiết bị và ứng dụng tại biên.

3. *IoT Platform (Nền tảng IoT)*

Đây là trung tâm xử lý, lưu trữ và phân tích dữ liệu từ các thiết bị IoT.

- **IoT Edge** (cũng có thể được triển khai trên Platform): Cho phép quản lý và xử lý dữ liệu từ nhiều gateway.
- **Transport:** Nhận dữ liệu từ Gateway thông qua MQTT, AMQP, HTTP.
- **Message Broker:** Quản lý và phân phối dữ liệu giữa các thành phần trong Platform.
- **Message Queue:** Lưu trữ tạm thời dữ liệu để đảm bảo việc xử lý không bị mất dữ liệu.
- **Dispatch Router:** Định tuyến dữ liệu đến các dịch vụ xử lý phù hợp.
- **Rule Engine:** Xử lý dữ liệu dựa trên các quy tắc được định nghĩa.
- **Core Services:** Cung cấp các chức năng chính của Platform, ví dụ như quản lý thiết bị, xử lý dữ liệu, phân tích.
- **REST API/WebSocket:** Cung cấp giao diện cho các ứng dụng bên ngoài truy cập dữ liệu và chức năng của Platform.
- **Data Storage:**
 - **Device Database:** Lưu trữ thông tin về thiết bị.
 - **Time-series Database:** Lưu trữ dữ liệu theo thời gian, thường được sử dụng cho dữ liệu cảm biến.
- **Device Management:** Cung cấp các công cụ để quản lý và giám sát các thiết bị IoT.

4. *Application (Ứng dụng)*

Sử dụng dữ liệu từ IoT Platform để cung cấp các dịch vụ cho người dùng.

- **IoT Services:** Cung cấp các dịch vụ cụ thể dựa trên dữ liệu IoT.
- **REST API/WebSocket:** Kết nối với IoT Platform để lấy dữ liệu.
- **Database:** Lưu trữ dữ liệu liên quan đến ứng dụng.

2.3. LUỒNG DỮ LIỆU

Luồng dữ liệu trong hệ thống IoT:

1. Dữ liệu được thu thập bởi các thiết bị IoT:

- Đây là bước khởi đầu của toàn bộ quy trình. Các thiết bị IoT, được trang bị cảm biến hoặc bộ phận chấp hành, tương tác với môi trường vật lý và thu thập dữ liệu.
- Dữ liệu được thu thập có thể ở nhiều định dạng khác nhau: analog, digital, hình ảnh, văn bản, v.v.
- Tần suất thu thập dữ liệu phụ thuộc vào ứng dụng và yêu cầu cụ thể. Ví dụ, cảm biến nhiệt độ có thể đo mỗi phút một lần, trong khi camera giám sát có thể ghi hình liên tục.

2. IoT Gateway nhận dữ liệu và xử lý sơ bộ:

- IoT Gateway đóng vai trò là cầu nối giữa các thiết bị IoT và IoT Platform. Nó nhận dữ liệu từ các thiết bị IoT thông qua nhiều giao thức khác nhau.
- **Xử lý sơ bộ:** Gateway có thể thực hiện một số xử lý dữ liệu ngay tại biên (edge computing) để:
 - **Lọc dữ liệu:** Loại bỏ dữ liệu nhiễu hoặc không liên quan.
 - **Tổng hợp dữ liệu:** Tính trung bình, tổng, giá trị lớn nhất/nhỏ nhất của dữ liệu.
 - **Chuyển đổi định dạng dữ liệu:** Chuyển đổi dữ liệu sang định dạng chuẩn để dễ dàng xử lý trên Platform.
 - **Mã hóa dữ liệu:** Tăng cường bảo mật bằng cách mã hóa dữ liệu trước khi gửi lên Platform.
- Việc xử lý dữ liệu sơ bộ tại Gateway giúp giảm tải cho Platform, tiết kiệm băng thông và giảm độ trễ.

3. Gateway gửi dữ liệu IoT Platform thông qua các giao thức như MQTT, AMQP, HTTP:

- Sau khi xử lý, Gateway sẽ gửi dữ liệu lên IoT Platform.
- Các giao thức truyền dữ liệu phổ biến:
 - **MQTT (Message Queuing Telemetry Transport):** Giao thức nhẹ, tiết kiệm băng thông, phù hợp cho các kết nối không ổn định. Thường được sử dụng cho các ứng dụng IoT yêu cầu độ trễ thấp.
 - **AMQP (Advanced Message Queuing Protocol):** Giao thức mạnh mẽ hơn MQTT, hỗ trợ nhiều tính năng nâng cao như đảm bảo tin nhắn được gửi tới đích, quản lý hàng đợi tin nhắn. Thường được sử dụng cho các ứng dụng doanh nghiệp yêu cầu độ tin cậy cao.

- **HTTP/HTTPS:** Giao thức web phổ biến, dễ triển khai nhưng tốn nhiều băng thông hơn so với MQTT và AMQP. Thường được sử dụng cho các ứng dụng đơn giản hoặc khi cần tích hợp với các hệ thống web hiện có.
- Việc lựa chọn giao thức phụ thuộc vào yêu cầu về hiệu suất, độ tin cậy và bảo mật của ứng dụng.

4. *IoT Platform xử lý, lưu trữ và phân tích dữ liệu:*

- IoT Platform là trung tâm xử lý, lưu trữ và phân tích dữ liệu từ các thiết bị IoT.
- **Xử lý dữ liệu:** Platform có thể thực hiện các xử lý phức tạp hơn so với Gateway, ví dụ: phân tích thời gian thực, phát hiện bất thường, dự đoán xu hướng.
- **Lưu trữ dữ liệu:** Dữ liệu được lưu trữ trong cơ sở dữ liệu, thường là cơ sở dữ liệu chuỗi thời gian (time-series database) để tối ưu cho việc truy vấn dữ liệu theo thời gian.
- **Phân tích dữ liệu:** Platform cung cấp các công cụ phân tích dữ liệu để người dùng có thể hiểu được dữ liệu và đưa ra quyết định. Ví dụ: tạo biểu đồ, báo cáo, thống kê.

5. *Các ứng dụng kết nối với IoT Platform thông qua REST API hoặc WebSocket để truy cập dữ liệu và cung cấp dịch vụ cho người dùng:*

- Các ứng dụng (web, di động, desktop) kết nối với IoT Platform thông qua API để truy cập dữ liệu và sử dụng các chức năng của Platform.
- **REST API:** Giao diện dựa trên HTTP, phổ biến cho các ứng dụng web và di động.
- **WebSocket:** Giao thức cho phép giao tiếp hai chiều theo thời gian thực giữa ứng dụng và Platform. Thường được sử dụng cho các ứng dụng yêu cầu cập nhật dữ liệu liên tục.
- Ứng dụng có thể sử dụng dữ liệu để:
 - Hiển thị thông tin cho người dùng, ví dụ: biểu độ nhiệt độ, bản đồ vị trí...
 - Điều khiển thiết bị từ xa, ví dụ: bật/tắt đèn, điều chỉnh nhiệt độ...
 - Tạo cảnh báo, ví dụ: gửi thông báo khi nhiệt độ vượt quá ngưỡng cho phép.
 - Phân tích dữ liệu để đưa ra quyết định, ví dụ: dự đoán nhu cầu bảo trì máy móc.

III> CÁC CHỨC NĂNG CHÍNH

3.1. QUẢN LÝ THIẾT BỊ IOT (DEVICE MANAGEMENT)

Quản lý thiết bị IoT (Device Management) là quá trình giám sát, điều khiển và duy trì các thiết bị IoT trong một hệ sinh thái IoT. Chức năng này đảm bảo rằng các thiết bị luôn hoạt động hiệu quả và an toàn. Bao gồm:

- **Đăng ký và cấu hình thiết bị:** IoT Platform cho phép đăng ký các thiết bị mới với các thông số kỹ thuật ban đầu, chẳng hạn như định danh duy nhất, giao thức kết nối và thông tin địa lý.
- **Theo dõi trạng thái thiết bị:** Cung cấp khả năng giám sát theo thời gian thực, hiển thị trạng thái hoạt động của thiết bị, chẳng hạn như đang hoạt động, không hoạt động hoặc gặp lỗi.
- **Điều khiển từ xa:** Người dùng có thể điều chỉnh các thông số hoạt động của thiết bị, khởi động lại thiết bị hoặc thực hiện các lệnh khác từ xa.

3.2. KẾT NỐI VÀ GIAO TIẾP THIẾT BỊ (DEVICE CONNECTIVITY)

Kết nối và giao tiếp thiết bị là nền tảng giúp thiết bị IoT gửi và nhận dữ liệu qua các giao thức truyền thông khác nhau.

- **Hỗ trợ nhiều giao thức, bao gồm:**
 - **MQTT:** Nhẹ, hiệu quả với băng thông thấp, lý tưởng cho thiết bị IoT tài nguyên hạn chế.
 - **HTTP/HTTPS:** Giao thức phổ biến cho truyền tải dữ liệu qua Internet.
 - **LoRaWAN, Zigbee, BLE:** Giao thức không dây tiết kiệm năng lượng cho các cảm biến hoặc thiết bị IoT nhỏ.
- **Kết nối hai chiều:** Thiết bị có thể gửi dữ liệu lên nền tảng (upstream) hoặc nhận lệnh từ nền tảng xuống (downstream).
- **Quản lý ổn định kết nối:**
 - Hỗ trợ cơ chế tự động đồng bộ dữ liệu khi kết nối bị gián đoạn.
 - Mã hóa dữ liệu khi truyền tải để đảm bảo an toàn.

3.3. THU THẬP VÀ XỬ LÝ DỮ LIỆU (DATA INGESTION & PROCESSING)

Thu thập và xử lý dữ liệu là quy trình tiếp nhận, chuyển đổi và phân tích dữ liệu được gửi từ các thiết bị IoT. Quá trình diễn ra:

- **Thu thập dữ liệu:** IoT Platform thu thập dữ liệu từ các cảm biến, thiết bị và nguồn dữ liệu khác qua hai hình thức:
 - **Thu thập hàng loạt (Batch):** Dữ liệu được nhóm thành từng lô (batch) và gửi về hệ thống theo chu kỳ. Phương pháp này hiệu quả cho các tình

huống không yêu cầu dữ liệu theo thời gian thực, chẳng hạn như báo cáo hàng ngày.

- **Thu thập theo luồng (Streaming):** Dữ liệu được gửi trực tiếp đến hệ thống ngay khi nó được tạo ra. Điều này phù hợp với các ứng dụng yêu cầu phản hồi thời gian thực, chẳng hạn như cảnh báo trong nhà máy.

- **Xử lý dữ liệu:**

- Làm sạch và chuẩn hóa dữ liệu từ nhiều nguồn khác nhau.
- Tích hợp với các công cụ như Apache Kafka hoặc Apache Spark để xử lý dữ liệu thời gian thực hoặc xử lý hàng loạt.
- Phân loại dữ liệu thô (raw data) hoặc dữ liệu đã xử lý (processed data) để lưu trữ hoặc phân tích thêm.

3.4. QUẢN LÝ SỰ KIỆN VÀ HÀNH ĐỘNG (EVENT & ACTION MANAGEMENT)

IoT Platform cung cấp khả năng thiết lập các quy tắc để tự động hóa phản ứng với các sự kiện xảy ra trong hệ thống. Cụ thể:

- **Quản lý sự kiện:**

- Phát hiện các sự kiện bất thường, chẳng hạn như cảm biến gửi dữ liệu vượt ngưỡng quy định.
- Gửi thông báo cảnh báo qua email, SMS hoặc ứng dụng nhắn tin (telegram).

- **Tích hợp hành động tự động:**

- Tự động điều khiển thiết bị, ví dụ: bật quạt khi nhiệt độ vượt ngưỡng.
- Kích hoạt các quy định hoặc API bên ngoài.

- **Kịch bản thông minh:**

- Tích hợp với AI để dự đoán sự kiện, chẳng hạn như bảo trì thiết bị dựa trên phân tích dữ liệu lịch sử.

3.5. BẢO MẬT VÀ QUẢN TRỊ HỆ THỐNG (SECURITY & GOVERNANCE)

Bảo mật và quản trị hệ thống là nền tảng đảm bảo rằng hệ thống IoT hoạt động an toàn và tuân thủ các quy định bảo mật quốc tế. Bao gồm các chức năng:

- **Xác thực:** Sử dụng giao thức OAuth 2.0 để xác thực người dùng và thiết bị.
- **Mã hóa:** Dữ liệu được mã hóa khi lưu trữ và truyền tải bằng tiêu chuẩn như TLS/SSL.
- **Giám sát an ninh:** Tích hợp hệ thống phát hiện xâm nhập (IDS) để theo dõi các hoạt động bất thường trong hệ thống.

- **Tuân thủ quy định:** Đảm bảo hệ thống đáp ứng theo tiêu chuẩn GDPR, ISO 27001 để bảo mật dữ liệu người dùng.

3.6. PHÂN TÍCH VÀ TRỰC QUAN HÓA DỮ LIỆU (DATA ANALYTICS & VISUALIZATION)

IoT Platform cung cấp các công cụ để phân tích dữ liệu từ các thiết bị IoT và hiển thị trực quan các thông tin quan trọng. Bao gồm:

- **Phân tích dữ liệu:**
 - Phân tích dữ liệu lịch sử để tìm kiếm xu hướng.
 - Phát hiện bất thường trong dữ liệu thời gian thực.
- **Trực quan hóa dữ liệu:**
 - Tạo các dashboard và biểu đồ tùy chỉnh theo yêu cầu người dùng.
 - Hiển thị các chỉ số hoạt động quan trọng một cách dễ hiểu.
- Tích hợp các mô hình máy học để dự đoán hoặc tối ưu hóa hoạt động, ví dụ: dự đoán nhu cầu năng lượng hoặc phát hiện lỗi thiết bị.

IV> KẾ HOẠCH TRIỂN KHAI IoT PLATFORM GIAI ĐOẠN 1

4.1. MỤC TIÊU GIAI ĐOẠN 1

- Xây dựng một luồng dữ liệu cơ bản từ thiết bị IoT đến Platform và ngược lại.
- Hỗ trợ kết nối thiết bị sử dụng giao thức Modbus TCP và MQTT.
- Triển khai các chức năng quản lý thiết bị cơ bản.
- Lưu trữ và hiển thị dữ liệu thu thập được.
- Đảm bảo tính bảo mật tối thiểu cho hệ thống.
- Xây dựng một hệ thống có khả năng mở rộng và dễ dàng bảo trì.

4.2. CÁC CÔNG VIỆC CHI TIẾT TRONG GIAI ĐOẠN 1

1. Thiết lập môi trường phát triển

- Lựa chọn và cài đặt các công cụ phát triển (IDE, Docker, Git...)
- Thiết lập môi trường ảo hóa.
- Cấu hình môi trường database cho lưu trữ và truy vấn dữ liệu.

2. Phát triển IoT Gateway

- **Device Service:**
 - Phát triển module kết nối và giao tiếp với các thiết bị Modbus TCP.
 - Phát triển module kết nối và giao tiếp với các thiết bị MQTT.
 - Chuyển đổi dữ liệu từ Modbus và MQTT sang định dạng JSON tiêu chuẩn.
- **Transport:**

- Triển khai module gửi dữ liệu lên IoT Platform qua MQTT (chú trọng tin cậy và hiệu suất).
- (Optional) Hỗ trợ giao thức HTTP nếu cần thiết cho việc gỡ lỗi hoặc các ứng dụng cụ thể.
- **Rule Engine (Cơ bản):**
 - Triển khai logic lọc dữ liệu cơ bản tại gateway, ví dụ: loại bỏ dữ liệu nhiễu hoặc chỉ gửi dữ liệu khi có thay đổi.
- **Data Storage:**
 - Sử dụng cơ chế lưu trữ tạm thời dữ liệu (file log hoặc queue trong bộ nhớ) khi mất kết nối đến platform.
- **Device Registry (Cơ bản):**
 - Lưu trữ thông tin thiết bị (tên, loại, địa chỉ kết nối...) trong bộ nhớ hoặc file cấu hình.
- **Security (Cơ bản):**
 - Triển khai xác thực cơ bản cho thiết bị kết nối vào gateway (username/password)
- **Management (Cơ bản):**
 - Cung cấp khả năng xem trạng thái kết nối của thiết bị.

3. *Phát triển IoT Platform (Core)*

- **Transport:**
 - Triển khai module nhận dữ liệu từ Gateway qua MQTT.
- **Message Broker/Queue:**
 - **Message core service** do nhóm phát triển IoT Platform nghiên cứu và phát triển với khả năng tích hợp và mở rộng cao hơn các message broker trên thị trường.
- **Dispatch Router:**
 - Triển khai module định tuyến dữ liệu từ broker đến các dịch vụ xử lý phù hợp.
- **Rule Engine (Cơ bản):**
 - Triển khai logic xử lý dữ liệu cơ bản, ví dụ: chuyển đổi đơn vị, kiểm tra ngưỡng.
- **Core Services:**
 - **Device Management:**
 - Triển khai chức năng đăng ký thiết bị mới, lưu trữ thông tin vào Device Database.

- Triển khai chức năng hiển thị danh sách thiết bị, trạng thái kết nối.
- **Data Storage:**
 - Thiết lập Time-series Database, ví dụ: InfluxDB hoặc TimescaleDB.
 - Triển khai chức năng lưu trữ dữ liệu từ broker vào time-series database.

4. *Phát triển Application (Cơ bản):*

- **IoT Services:**
 - Xây dựng một dashboard hiển thị dữ liệu từ thiết bị, ví dụ: đồ thị, bảng số liệu.
 - Cung cấp khả năng lọc dữ liệu theo thiết bị, khoảng thời gian.
- **REST API:**
 - Kết nối với IoT Platform thông qua REST API để lấy dữ liệu và hiển thị trên dashboard.

5. *Kiểm thử:*

- Kiểm thử luồng dữ liệu từ thiết bị Modbus/MQTT đến gateway, platform, và hiển thị trên dashboard.
- Kiểm thử kết nối và độ ổn định của gateway.
- Kiểm thử các chức năng cơ bản của platform.
- Kiểm thử các chức năng của application.

6. *Tài liệu:*

- Cập nhật tài liệu mô tả kiến trúc và luồng dữ liệu.
- Tạo tài liệu hướng dẫn sử dụng cho người dùng cuối.
- Ghi lại các quyết định thiết kế và triển khai.

7. *Các điểm cần chú ý trong giai đoạn 1:*

- Ưu tiên tính đơn giản: Tập trung vào việc xây dựng một luồng dữ liệu cơ bản và các chức năng cốt lõi.
- Bảo mật: Đảm bảo các biện pháp bảo mật tối thiểu (xác thực, mã hóa).
- Khả năng mở rộng: Thiết kế hệ thống có khả năng mở rộng để dễ dàng thêm các tính năng mới trong tương lai.
- Dễ bảo trì: Sử dụng code rõ ràng, có tài liệu hướng dẫn và có thể debug dễ dàng.

4.3. CƠ CẤU TỔ CHỨC ĐỘI NGŨ

1. *Đội ngũ*

Tổ chức đội ngũ sẽ đi theo mô hình chức năng, với các nhóm chuyên trách các mảng công việc khác nhau, nhưng vẫn đảm bảo sự phối hợp chặt chẽ:

- **Quản lý dự án (Project Manager – PM):** 1 người.
- **Nhóm phát triển IoT Gateway (IoT Gateway Development Team):** 1 người.
- **Nhóm phát triển IoT Platform (IoT Platform Development Team):** 3 người.
- **Nhóm kiểm thử (Testing Team):** 1 người (có thể là part-time).

2. *Vai trò và trách nhiệm của từng vị trí:*

- **Quản lý dự án (Project Manager):**
 - Lập kế hoạch chi tiết cho dự án, bao gồm lịch trình, ngân sách, và phân công công việc.
 - Điều phối các nhóm làm việc, đảm bảo các công việc được thực hiện đúng tiến độ và chất lượng.
 - Quản lý rủi ro, giải quyết các vấn đề phát sinh trong quá trình triển khai.
 - Báo cáo tiến độ dự án cho các bên liên quan.
 - Là cầu nối giao tiếp chính giữa các nhóm với quản lý cấp cao.
- **Nhóm phát triển IoT Gateway (IoT Gateway Development Team):**
 - Phát triển các module kết nối và giao tiếp với thiết bị Modbus TCP và MQTT.
 - Xử lý và chuyển đổi dữ liệu, triển khai logic lọc dữ liệu tại gateway.
 - Quản lý các kết nối, xử lý các lỗi và các vấn đề liên quan đến gateway.
 - Đảm bảo tính bảo mật và hiệu suất của gateway.
- **Nhóm phát triển IoT Platform (IoT Platform Development Team):**
 - Phát triển các module nhận dữ liệu, quản lý message broker/queue.
 - Triển khai rule engine, các dịch vụ cốt lõi của platform.
 - Thiết lập và quản lý database, REST API.
 - Phát triển giao diện người dùng (dashboard) để hiển thị dữ liệu.
 - Triển khai platform lên nền tảng cloud.
 - Đảm bảo platform hoạt động ổn định, bảo mật và có khả năng mở rộng.
- **Nhóm kiểm thử (Testing Team):**
 - Xây dựng test case và thực hiện kiểm thử hệ thống.
 - Phát hiện và báo cáo các lỗi, bug.
 - Đảm bảo hệ thống hoạt động ổn định và đáp ứng yêu cầu.