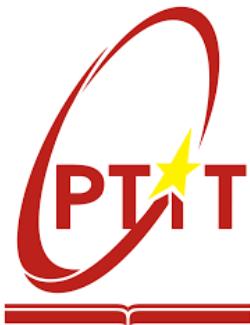


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1



BÁO CÁO GIỮA KÌ

Lớp : E22HTTT

Nhóm : 1

Chủ đề : Đèn chiếu sáng thông minh

Thành viên : Đinh Trí Đạt - B22DCVT124
Nguyễn Hoàng - B22DCVT209
Nguyễn Bá Dương - B22DCDT070
Trịnh Quang Bách - B22DCVT045

Ha Noi -2025

Contents

| | |
|---|----|
| I. Tổng quan Đề tài | 3 |
| 1. Giới thiệu Đề tài..... | 3 |
| 2. Lý do chọn Đề tài..... | 3 |
| 3. Mục tiêu Thành công | 3 |
| 4. Phạm vi Triển khai..... | 3 |
| II. Xác định Yêu cầu Chức năng (Functional Requirements) | 4 |
| III. Xác định Yêu cầu Phi Chức năng (Non-Functional Requirements) | 5 |
| IV. Phân tích Ràng buộc (Constraints Analysis) | 5 |
| V. Mô hình hóa Yêu cầu..... | 6 |
| VI. Công nghệ sử dụng | 6 |
| 1. Giao thức Truyền tin: MQTT | 6 |
| 2. Máy chủ MQTT Broker: HiveMQ Public Broker..... | 7 |
| 3. Bluetooth Low Energy (BLE) | 7 |
| 4. Nền tảng Ứng dụng: Native Mobile App (Android Studio) | 8 |
| VII. Thiết bị sử dụng..... | 9 |
| 1. Vi điều khiển (Chip): ESP32 | 9 |
| 2. Thiết bị Chấp hành: Module Relay 5V | 10 |
| 3. Cảm biến | 10 |
| VIII. Xác minh & Phê duyệt Yêu cầu (Ưu tiên Tính năng)..... | 11 |
| IX. Phân chia Công việc và Kế hoạch Triển khai..... | 11 |
| 1. Giai đoạn 1: Lõi (Core) & Thiết lập (MUST-HAVE) | 11 |
| 2. Giai đoạn 2: Trí tuệ (Intelligence) (SHOULD-HAVE) | 11 |
| 3. Giai đoạn 3: Nâng cao (NICE-TO-HAVE) & Hoàn thiện | 12 |

I. Tổng quan Đề tài

1. Giới thiệu Đề tài

- Tên Đề tài: Xây dựng Hệ thống Giám sát và Điều khiển Đèn Chiếu sáng Thông minh sử dụng kiến trúc IoT. Hệ thống giải quyết vấn đề Giám sát, Tự động hóa và Điều khiển từ xa việc sử dụng năng lượng điện thông qua chiếu sáng.
- Mục tiêu: Phát triển một giải pháp điều khiển chiếu sáng linh hoạt, cho phép người dùng giám sát trạng thái thiết bị và điều khiển từ xa thông qua nhiều kênh giao tiếp (Web/App/Chatbot), đồng thời tích hợp các tính năng bảo trì từ xa.
- Mức độ Dự án: Trung bình (Intermediate - Mức độ 2). Dự án này yêu cầu tích hợp nhiều giao thức (Wi-Fi, MQTT, BLE) và triển khai các tính năng nâng cao như OTA Firmware Update và điều khiển qua Chatbot.

2. Lý do chọn Đề tài

- Đề tài đáp ứng nhu cầu thực tiễn về tiết kiệm năng lượng và tăng tiện nghi trong môi trường sống hiện đại. Việc sử dụng giao thức nhẹ MQTT giúp tối ưu hiệu suất và băng thông, trong khi tính năng OTA đảm bảo khả năng mở rộng và bảo trì hệ thống dễ dàng sau này.

3. Mục tiêu Thành công

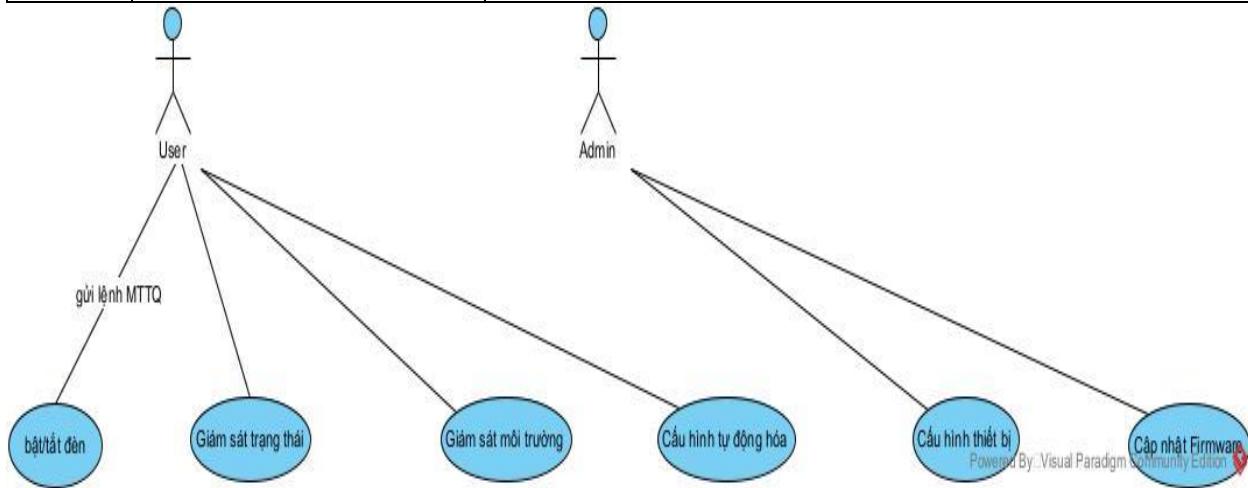
- Hiệu suất: Tốc độ phản hồi lệnh điều khiển dưới 3 giây (Độ trễ thấp).
- Độ tin cậy: Khả năng kết nối lại MQTT thành công sau khi mất mạng trong vòng 60 giây.
- Mức độ Tự động hóa: Hệ thống BẬT/TẮT đèn tự động theo LDR với độ chính xác trên 90% so với ngưỡng cài đặt.
- Thời gian triển khai: Hoàn thành các tính năng cốt lõi trong 8 tuần (Phù hợp với ràng buộc thời gian).

4. Phạm vi Triển khai

- Quy mô: Một phòng/khu vực thử nghiệm.
- Môi trường: Ánh sáng môi trường (LDR).
- Số lượng Thiết bị: Một ESP32 điều khiển một Module Relay (mô phỏng đèn).
- Phương thức Giao tiếp: MQTT và Wi-Fi.

II. Xác định Yêu cầu Chức năng (Functional Requirements)

| ID | Chức năng | Mô tả chi tiết |
|------|----------------------------------|--|
| FR01 | Điều khiển Đèn | Người dùng gửi lệnh BẬT/TẮT tới ESP32 qua Native Mobile App. |
| FR02 | Giám sát Trạng thái | ESP32 Publish trạng thái (ON/OFF) theo thời gian thực lên Broker; App hiển thị trạng thái này. |
| FR03 | Giám sát Môi trường | ESP32 thu thập giá trị LDR và Publish lên Broker; App hiển thị giá trị này để người dùng theo dõi và xác thực. |
| FR04 | Cấu hình Tự động hóa | ESP32 thực thi logic tự động BẬT/TẮT đèn khi giá trị LDR vượt quá ngưỡng cài đặt. |
| FR05 | Cấu hình Thiết bị (Provisioning) | Cho phép cấu hình Wi-Fi/Broker mà không cần nạp lại code. |
| FR06 | Cập nhật Firmware OTA | Cho phép cập nhật phần mềm mới cho ESP32 từ xa qua Wi-Fi |



III. Xác định Yêu cầu Phi Chức năng (Non-Functional Requirements)

| Chất lượng | Yêu cầu đặt ra | Công nghệ giải quyết |
|------------|--|--|
| Hiệu năng | Độ trễ giao tiếp (đặc biệt là điều khiển) phải dưới 3 giây. | Giao thức MQTT nhẹ. |
| Độ tin cậy | Thiết bị phải tự động kết nối lại Broker khi bị mất mạng và tiếp tục chức năng tự động hóa cục bộ. | HiveMQ Broker (ổn định) và logic code nhúng. |
| Mở rộng | Dễ dàng thêm thiết bị mới (thêm ESP32 mới). | Kiến trúc MQTT Pub/Sub. |
| Chi phí | Chi phí linh kiện thấp nhất có thể. | Sử dụng ESP32 và Module Relay 5V giá rẻ. |

IV. Phân tích Ràng buộc (Constraints Analysis)

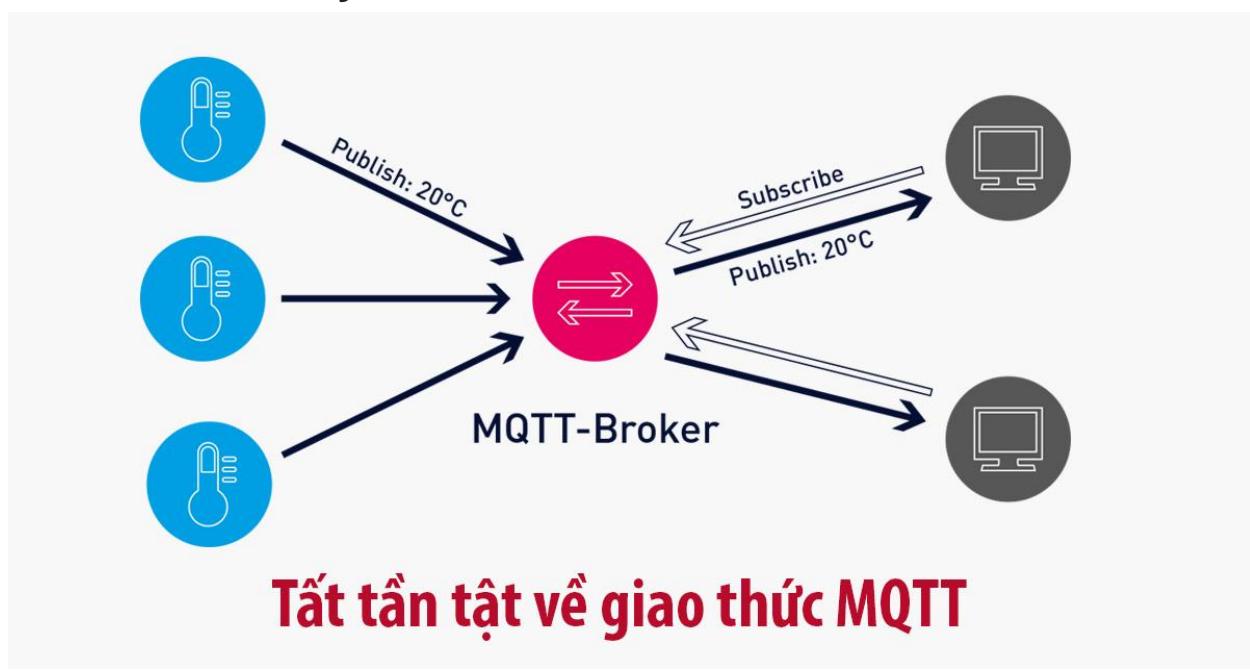
| Ràng buộc | Chi tiết | Tác động đến dự án |
|---------------|---|---|
| Môi trường | Nguồn cấp điện trong nhà. | Bắt buộc sử dụng Module Relay 5V để đơn giản hóa nguồn cấp và đảm bảo an toàn thử nghiệm. |
| Cáu hình Mạng | Ưu tiên tính đơn giản. | Cáu hình Wi-Fi và Broker sẽ được Cấu hình Bằng Tay (Hardcode) trong firmware. |
| Thời gian | Dự án phải hoàn thành và kiểm thử trong 8 tuần. | Phạm vi tính năng đã được tinh gọn, tập trung vào các chức năng cốt lõi. |

V. Mô hình hóa Yêu cầu

| Lớp | Thành phần | Mô tả giao tiếp |
|------------------------|-----------------------------|--|
| Perception (Cảm nhận) | ESP32, LDR, Relay 5V | Thu thập dữ liệu LDR và thực thi lệnh Relay (Output). |
| Network (Mạng) | Wi-Fi, HiveMQ Broker | Dữ liệu được truyền tải qua Wi-Fi/MQTT (Lệnh điều khiển, dữ liệu LDR). |
| Application (Ứng dụng) | Native Mobile App (Android) | Xử lý giao diện người dùng, gửi/nhận lệnh MQTT. |

VI. Công nghệ sử dụng

1. Giao thức Truyền tin: MQTT



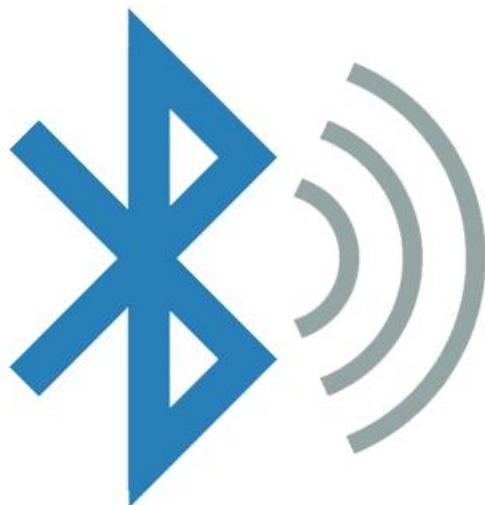
- Lý do sử dụng: Là giao thức nhắn tin nhẹ nhất hiện nay, hoạt động theo mô hình Publish/Subscribe.
- Giá trị mang lại:
 - Đảm bảo lệnh điều khiển và trạng thái được truyền tải gần như tức thời (độ trễ dưới 3 giây).
 - Tiết kiệm băng thông và tài nguyên bộ nhớ cho ESP32.

2. Máy chủ MQTT Broker: HiveMQ Public Broker



- Lý do sử dụng: Sử dụng dịch vụ Public Broker (Máy chủ công cộng miễn phí).
- Giá trị mang lại:
 - o Giúp nhóm loại bỏ hoàn toàn khâu thiết lập, cài đặt và duy trì máy chủ (server) riêng.
 - o Cho phép các thành viên bắt đầu lập trình và kiểm thử ngay lập tức mà không bị phụ thuộc vào môi trường Backend phức tạp.

3. Bluetooth Low Energy (BLE)



- Lý do sử dụng: Dùng cho tính năng Provisioning (Cấu hình mạng). BLE cung cấp kênh giao tiếp an toàn và Native API ổn định trên hệ điều hành Android.

4. Nền tảng Ứng dụng: Native Mobile App (Android Studio)



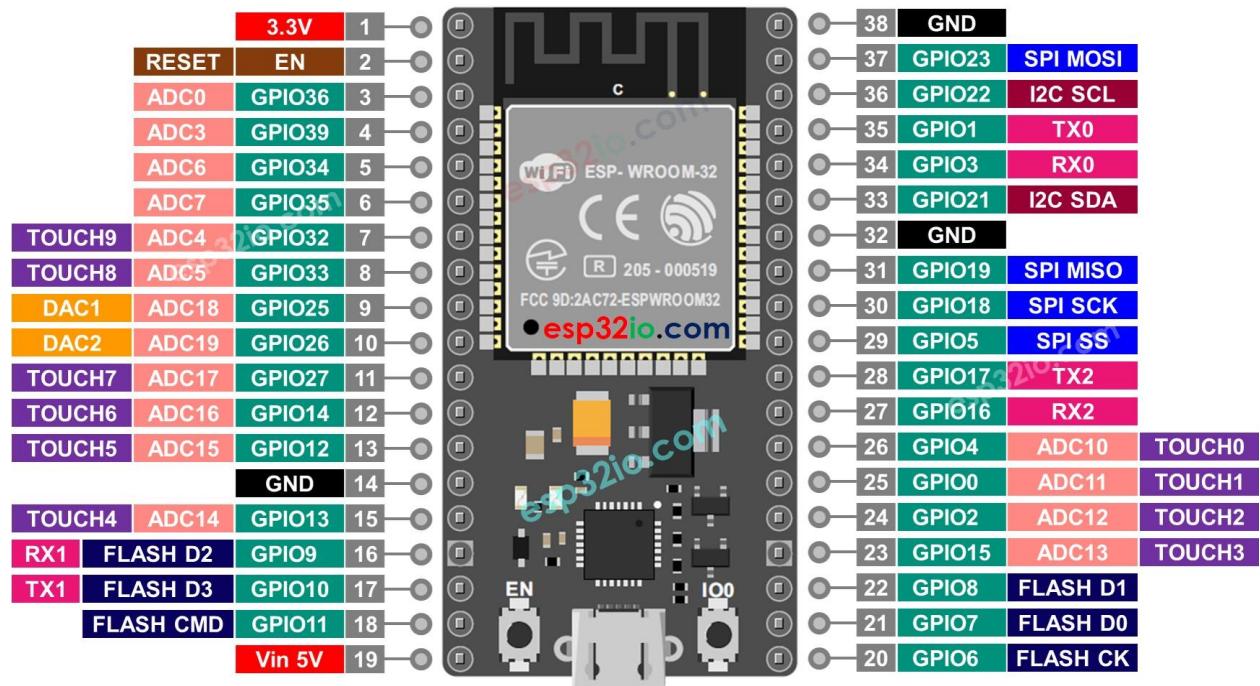
- Lý do sử dụng: Cung cấp trải nghiệm người dùng mượt mà và khả năng tận dụng phần cứng tốt.

- Giá trị mang lại:

- Tận dụng Native Bluetooth API của Android, giúp dễ dàng triển khai chức năng BLE Provisioning (nếu nhóm quyết định làm tính năng này sau).
- Là giao diện chính để điều khiển (FR01) và giám sát dữ liệu (FR02, FR03).

VII. Thiết bị sử dụng

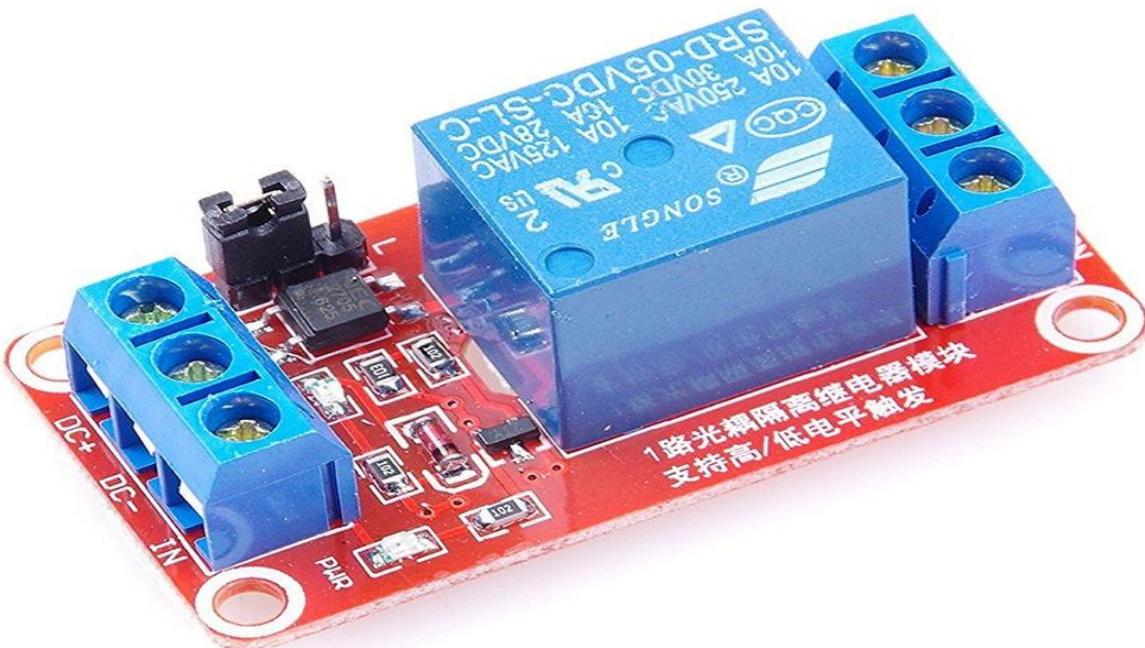
1. Vi điều khiển (Chip): ESP32



- Lý do sử dụng:

- Chi phí thấp và thông dụng trên thị trường.
- Tích hợp sẵn Wi-Fi để kết nối MQTT và Bluetooth (BLE) để sử dụng cho Provisioning (tính năng NICE-TO-HAVE).
- Hỗ trợ lập trình dễ dàng qua môi trường Arduino, phù hợp với người mới bắt đầu.

2. Thiết bị Chấp hành: Module Relay 5V



- Lý do sử dụng:

- An toàn và Đơn giản: Relay 5V hoạt động với nguồn 5V, có thể dùng chung nguồn cấp từ cổng USB máy tính hoặc củ sạc điện thoại với ESP32.
- Chức năng: Thực hiện việc BẬT/TẮT nguồn điện cho đèn mô phỏng theo lệnh từ ESP32.

3. Cảm biến

- Để tiết kiệm thời gian, nhóm sẽ mô phỏng dữ liệu ánh sáng môi trường (thay cho LDR thật) và lưu trữ trong Database.

VIII. Xác minh & Phê duyệt Yêu cầu (Ưu tiên Tính năng)

| Tính năng | Phân loại | Lý do |
|--------------------------------|--------------|---|
| Điều khiển Đèn (FR01) | MUST-HAVE | Chức năng cốt lõi của hệ thống điều khiển. |
| Giám sát Trạng thái (FR02) | MUST-HAVE | Đảm bảo phản hồi và xác nhận lệnh. |
| Cấu hình Tự động hóa (FR04) | SHOULD-HAVE | Tính năng thông minh hóa (Giá trị cốt lõi của IoT). |
| Giám sát Môi trường LDR (FR03) | SHOULD-HAVE | Cần thiết để xác thực tính năng tự động hóa và tăng tính minh bạch. |
| Cấu hình Thiết bị (FR05) | NICE-TO-HAVE | Phức tạp, chỉ nên xem xét sau khi dự án hoàn thành. |
| Cập nhật Firmware OTA (FR06) | NICE-TO-HAVE | Phức tạp, chỉ nên xem xét sau khi dự án hoàn thành. |

IX. Phân chia Công việc và Kế hoạch Triển khai

1. Giai đoạn 1: Lõi (Core) & Thiết lập (MUST-HAVE)

- Mục tiêu: Thiết lập môi trường, kết nối HiveMQ và hoàn thành giao tiếp hai chiều cơ bản.
- Nhiệm vụ:
 - T1 (Firmware): Lập trình kết nối Wi-Fi, MQTT và điều khiển Relay.
 - T2 (Backend/DB): Cài đặt Database (cho dữ liệu mô phỏng) và chuẩn bị cấu trúc bảng dữ liệu.
 - T3 (Mobile App): Xây dựng giao diện App cơ bản và tích hợp thư viện MQTT Client.
 - T4 (Kiểm thử): Viết các Test Case cơ bản cho chức năng Bật/Tắt (FR01) và Giám sát (FR02).

2. Giai đoạn 2: Trí tuệ (Intelligence) (SHOULD-HAVE)

- Mục tiêu: Tích hợp dữ liệu mô phỏng và hoàn thành logic thông minh.
- Nhiệm vụ:
 - T1 (Firmware): Nhận lệnh/đọc dữ liệu Tự động hóa từ Database (hoặc Backend) và triển khai Logic Tự động hóa (FR04).

- T2 (Backend/DB): Hoàn thành việc điền Dữ liệu Mô phỏng vào Database và xây dựng API/Logic Backend để hỗ trợ T1 và T3.
- T3 (Mobile App): Lập trình để App đọc Dữ liệu Mô phỏng (FR03) và hiển thị biểu đồ.
- T4 (Kiểm thử): Thực hiện kiểm thử tích hợp cho FR03 và FR04.

3. Giai đoạn 3: Nâng cao (NICE-TO-HAVE) & Hoàn thiện

- Mục tiêu: Triển khai các tính năng mở rộng nếu còn thời gian và chuẩn bị báo cáo.

- Nhiệm vụ:

- T1 & T4 (Provisioning): Phối hợp triển khai Provisioning qua BLE (NICE1) và kiểm thử độ ổn định.
- T1 (Firmware): Triển khai OTA Firmware Update (NICE2).
- T3 (Mobile App): Hoàn thiện UX/UI và tích hợp các tính năng NICE1, NICE2.
- T2, T3, T4: Tổng hợp Báo cáo và tài liệu hướng dẫn.