

TRƯỜNG ĐẠI HỌC KHOA HỌC HUẾ

KHOA CÔNG NGHỆ THÔNG TIN



PHÁT TRIỂN ỨNG DỤNG IOT – NHÓM 6

MÃ HỌC PHẦN: TIN4024.006

ĐỀ TÀI:

Điều khiển từ xa qua giao thức MQTT với ESP32

Giáo viên hướng dẫn :Võ Việt Dũng

Sinh viên thực hiện :Nguyễn Khắc Tuấn Long

Huế, tháng 4 năm 2025

MỤC LỤC

CHƯƠNG 1: MỞ ĐẦU	3
1. Giới thiệu	3
1.1. Bối cảnh về nhà thông minh và tự động hóa	3
1.2. Tầm quan trọng của điều khiển từ xa qua giao thức IoT	3
2. Tổng quan về dự án	3
2.1. Mục tiêu dự án	3
2.2. Thành phần chính	3
3. Mô tả hệ thống	4
CHƯƠNG 2: NỘI DUNG	5
1. Cơ sở lý thuyết	5
1.1. Giao thức MQTT (Message Queuing Telemetry Transport) là gì?	5
1.2. Giới thiệu ESP32	5
1.3. Vai trò ESP32 Broker	6
1.4. Nền tảng giám sát Blynk	7
1.5. Nền tảng Wokwi	7
1.6. Các cảm biến và thiết bị điều khiển	7
2. Thiết kế hệ thống	8
2.1. Thành phần hệ thống	8
2.2. Mô hình hoạt động	9
2.2.1. Kiến trúc tổng thể	9
2.2.2. Luồng hoạt động hệ thống	9
2.2.3. Hoạt động liên tục và tương tác hai chiều	11
2.2.4. Ưu điểm của mô hình này	12
2.3. Sơ đồ Wokwi	12
CHƯƠNG 3: KẾT LUẬN	14
1. Tóm tắt	14
2. Đánh giá	14
3. Bài học	14
4. Hướng phát triển	14
TÀI LIỆU THAM KHẢO	16
PHỤ LỤC	17

CHƯƠNG 1: MỞ ĐẦU

1. Giới thiệu

1.1. Bối cảnh về nhà thông minh và tự động hóa

- Trong kỷ nguyên công nghệ hiện đại, nhà thông minh và hệ thống tự động hóa đã trở thành biểu tượng của sự tiện nghi và hiệu quả. Nhờ vào sự phát triển mạnh mẽ của công nghệ IoT (Internet of Things), các thiết bị điện tử trong gia đình giờ đây có thể được kết nối và điều khiển một cách dễ dàng. Xu hướng này giúp đáp ứng nhu cầu của người dùng trong việc giám sát và quản lý các thiết bị từ xa.
- Đặc biệt, việc điều khiển từ xa các thiết bị như: đèn chiếu sáng, quạt, hệ thống điều hòa nhiệt độ, hoặc cảm biến môi trường.v.v.v, không chỉ mang lại sự tiện lợi mà còn góp phần tối ưu hóa chi phí năng lượng và tăng cường an toàn cho người sử dụng.

1.2. Tầm quan trọng của điều khiển từ xa qua giao thức IoT

- Hệ thống IoT với giao thức MQTT mang lại các lợi ích:
 - Tính tiện dụng: Người dùng có thể điều khiển các thiết bị từ mọi nơi chỉ bằng một ứng dụng di động.
 - Hiệu suất năng lượng cao: Điều chỉnh trạng thái thiết bị dựa trên nhu cầu thực tế, tránh lãng phí.
 - Tích hợp dữ liệu: Thu thập và quản lý dữ liệu thiết bị, từ đó hỗ trợ các quyết định sử dụng hợp lý và tiết kiệm hơn.
 - Mở rộng linh hoạt: Hệ thống dễ dàng mở rộng hoặc kết hợp với các công nghệ thông minh khác.

2. Tổng quan về dự án

2.1. Mục tiêu dự án

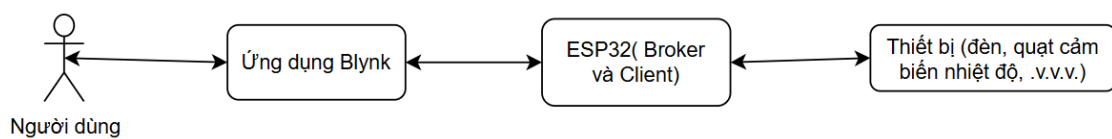
- Dự án hướng đến xây dựng một hệ thống điều khiển từ xa qua giao thức MQTT, trong đó ESP32 sẽ đảm nhận cả vai trò client và broker. Điều này giúp tối giản hệ thống mà vẫn đảm bảo hiệu quả vận hành.

2.2. Thành phần chính

- ESP32 Client: Chịu trách nhiệm nhận lệnh từ ứng dụng và thực hiện các tác vụ, như bật/tắt LED hoặc gửi dữ liệu từ cảm biến DHT22 (nhiệt độ và độ ẩm).
- ESP32 Broker: Đóng vai trò nhà môi giới, nhận và truyền tải các tin nhắn giữa các thiết bị và ứng dụng điều khiển.
- Ứng dụng Blynk: Là giao diện người dùng để gửi lệnh và theo dõi trạng thái hệ thống.

3. Mô tả hệ thống

- Sơ đồ tổng quát:



- Quá trình hoạt động:
 - o Người dùng nhập lệnh trên ứng dụng Blynk.
 - o Lệnh được gửi đến ESP32 Broker qua giao thức MQTT.
 - o ESP32 Broker truyền tiếp lệnh đến ESP32 Client.
 - o ESP32 Client thực thi lệnh, như bật LED hoặc gửi dữ liệu cảm biến.
 - o ESP32 Client cập nhật trạng thái của thiết bị.
 - o ESP32 Client gửi trạng thái tới ESP32 Broker.
 - o ESP32 Broker gửi tới Blynk.
 - o Blynk hiển thị trạng thái thiết bị cho người dùng xem.

CHƯƠNG 2: NỘI DUNG

1. Cơ sở lý thuyết

1.1. Giao thức MQTT (Message Queuing Telemetry Transport) là gì?

- MQTT là giao thức truyền thông dựa trên kiến trúc publish/subscribe, được thiết kế tối ưu cho môi trường mạng không ổn định và các thiết bị có tài nguyên giới hạn như trong hệ thống IoT. Đây là giao thức nhẹ với kích thước dữ liệu nhỏ, phù hợp để gửi các thông điệp thời gian thực giữa các thiết bị..
- Ưu điểm:
 - o Tiết kiệm băng thông: Tin nhắn MQTT nhỏ gọn, giảm thiểu lưu lượng mạng.
 - o Độ tin cậy cao: Hỗ trợ các mức độ dịch vụ (QoS 0, 1, 2) để đảm bảo dữ liệu được gửi chính xác.
 - o Thời gian thực: Cho phép giao tiếp nhanh giữa các thiết bị, phù hợp với các ứng dụng yêu cầu độ trễ thấp.
 - o Mở rộng linh hoạt: Kiến trúc publish/subscribe giúp dễ dàng thêm thiết bị mới mà không ảnh hưởng đến hệ thống hiện tại.
- Ứng dụng trong IoT:
 - o MQTT rất hữu dụng trong các ứng dụng như nhà thông minh, giám sát môi trường, tự động hóa công nghiệp và theo dõi thiết bị từ xa
- Trong thiết kế này, ESP32 sẽ đảm nhận cả hai vai trò client và broker, xử lý toàn bộ việc gửi/nhận thông điệp, mà không cần sử dụng broker bên ngoài như Mosquitto.

1.2. Giới thiệu ESP32

- ESP32 là vi điều khiển mạnh mẽ, thường được dùng trong các ứng dụng IoT nhờ khả năng kết nối Wi-Fi và Bluetooth tích hợp. Trong hệ thống này, ESP32 vừa đóng vai trò client vừa là broker MQTT, điều này giúp tối giản cấu trúc và giảm sự phụ thuộc vào các máy chủ trung gian.
- Một số tính năng nổi bật của ESP32:

- Kết nối WiFi và Bluetooth: Hỗ trợ WiFi 802.11 b/g/n và Bluetooth 4.2 (BLE), giúp giao tiếp không dây linh hoạt.
 - Hỗ trợ nhiều giao thức truyền thông: SPI, I2C, UART, ADC, DAC, PWM, giúp dễ dàng kết nối với cảm biến và các thiết bị ngoại vi.
 - Tiêu thụ điện năng thấp: Có chế độ tiết kiệm năng lượng Deep Sleep, phù hợp cho các ứng dụng chạy bằng pin.
 - Hỗ trợ MQTT: ESP32 có thể giao tiếp với broker MQTT để gửi và nhận dữ liệu điều khiển từ xa một cách hiệu quả.
- Nhờ các tính năng trên, ESP32 rất phù hợp để triển khai hệ thống bật/tắt thiết bị từ xa qua giao thức MQTT, giúp tăng cường khả năng giám sát và điều khiển thiết bị trong các ứng dụng IoT thực tế.
 - Bảng thông số kỹ thuật ESP32:

Thông số	Chi tiết
CPU	Dual-core Xtensa LX6
Tốc độ xử lý	Tối đa 240 MHz
RAM	520 KB SRAM
Flash	Từ 4 MB
GPIO	Hơn 30 chân
Kết nối	Wi-Fi 802.11 b/g/n, Bluetooth v4.2 BLE
Điện áp hoạt động	3.0 – 3.6V

1.3. Vai trò ESP32 Broker

- Trong thiết kế này, ESP32 sẽ thực hiện các nhiệm vụ của một broker MQTT:
 - Quản lý chủ đề (topics): ESP32 lưu giữ thông tin về các chủ đề được đăng ký, đảm bảo dữ liệu được truyền đúng đến client.
 - Phân phối tin nhắn: ESP32 nhận dữ liệu từ client rồi gửi đến các subscriber tương ứng.
 - Kiểm soát bảo mật: Với khả năng hỗ trợ mã hóa TLS, broker ESP32 đảm bảo sự an toàn trong truyền thông.

- Việc ESP32 làm broker mang lại lợi ích lớn về mặt tối giản hệ thống và giảm chi phí triển khai

1.4. Nền tảng giám sát Blynk

- Blynk là một công cụ hữu ích cho việc điều khiển và giám sát thiết bị IoT. Trong dự án này, Blynk sẽ được kết hợp với ESP32 thông qua giao thức MQTT, đóng vai trò giao diện người dùng.
- Lợi ích của Blynk:
 - o Giao diện trực quan: Người dùng có thể điều khiển thiết bị chỉ với vài thao tác trên ứng dụng di động.
 - o Tích hợp MQTT: Blynk hỗ trợ giao tiếp với ESP32 thông qua MQTT mà không cần các phần mềm trung gian.
 - o Theo dõi dữ liệu thời gian thực: Hiển thị trạng thái thiết bị và thông số cảm biến (nhiệt độ, độ ẩm, v.v.).

1.5. Nền tảng Wokwi

- Wokwi là nền tảng mô phỏng mạnh mẽ, cho phép người dùng kiểm tra hệ thống IoT với ESP32 mà không cần phần cứng.
- Ưu điểm của Wokwi:
 - o Hỗ trợ ESP32: Mô phỏng đầy đủ cả chức năng broker và client của ESP32.
 - o Tiết kiệm chi phí: Thay vì mua thiết bị thực, Wokwi cung cấp mô hình ảo để thử nghiệm.
 - o Giao diện thân thiện: Cung cấp môi trường mã hóa trực quan để kiểm tra hoạt động của hệ thống.

1.6. Các cảm biến và thiết bị điều khiển

- Thiết bị bật/tắt (Relay, LED,...):
 - o Các thiết bị cơ bản như relay và LED sẽ được điều khiển trực tiếp từ ESP32 thông qua GPIO, giúp thực hiện lệnh bật/tắt từ xa.
- Bảng thông số kỹ thuật mẫu:

Thiết bị	Thông số kỹ thuật chính
Relay	Điện áp hoạt động: 5V, Dòng tối đa: 10A
LED	Điện áp hoạt động: 2V, Dòng: 10-20mA
Cảm biến DHT22	Dải đo nhiệt độ: -40 đến 80°C, Sai số: $\pm 0.5^{\circ}\text{C}$

2. Thiết kế hệ thống

2.1. Thành phần hệ thống

- Hệ thống được xây dựng dựa trên các thành phần chính sau:
- Phần cứng:
 - ESP32: Đảm nhận cả hai vai trò:
 - Broker MQTT: Quản lý truyền nhận dữ liệu giữa các thiết bị IoT.
 - Client MQTT: Thực hiện các lệnh bật/tắt thiết bị và gửi dữ liệu từ cảm biến.
 - Relay: hoạt động như một công tắc điện tử, cho phép bật/tắt thiết bị điện áp cao từ xa, ví dụ: bóng đèn hoặc quạt.
 - LED: Hiển thị trạng thái hệ thống (ví dụ: bật/tắt thành công).
 - Cảm biến DHT22: Thu thập dữ liệu nhiệt độ và độ ẩm, gửi về ESP32 để xử lý.
 - Nguồn cung cấp điện: Đảm bảo cung cấp năng lượng cho ESP32 và các thiết bị ngoại vi.
- Phần mềm:
 - MQTT Protocol: ESP32 sử dụng giao thức này để giao tiếp giữa broker và client.
 - Blynk Application: Là nền tảng giám sát và điều khiển từ xa, cung cấp giao diện cho người dùng để bật/tắt thiết bị hoặc xem thông số cảm biến.
- Wokwi Platform: Dùng để mô phỏng hệ thống trước khi triển khai thực tế.
- Kết nối mạng:

- Hệ thống sử dụng kết nối Wi-Fi để ESP32 có thể hoạt động với vai trò là broker MQTT và giao tiếp với ứng dụng Blynk.

2.2. Mô hình hoạt động

2.2.1. Kiến trúc tổng thể

- Hệ thống được xây dựng dựa trên mô hình ESP32 làm broker và client MQTT để quản lý các thiết bị điều khiển và cảm biến. Đây là một thiết kế tối giản nhưng hiệu quả, phù hợp với hệ thống IoT nhỏ gọn. Tất cả các thành phần sẽ tương tác qua giao thức MQTT, và ứng dụng Blynk đóng vai trò giao diện điều khiển và giám sát.

2.2.2. Luồng hoạt động hệ thống

- Hệ thống vận hành theo một chuỗi các bước cụ thể dưới đây:

Bước 1: Kết nối ban đầu

- ESP32 khởi động và kết nối Wi-Fi:
 - Khi được cấp nguồn, ESP32 sẽ kết nối với mạng Wi-Fi được định cấu hình sẵn. Đây là bước quan trọng để đảm bảo ESP32 có thể giao tiếp với các thành phần khác qua MQTT và kết nối với ứng dụng Blynk.
 - Lúc này, ESP32 kích hoạt cả hai vai trò:
 - Làm broker MQTT: Đảm nhiệm chức năng quản lý giao tiếp dữ liệu giữa các client.
 - Làm client MQTT: Chuẩn bị nhận lệnh điều khiển hoặc thu thập dữ liệu từ các cảm biến.
- Kết nối ứng dụng Blynk với ESP32:
 - Ứng dụng Blynk được cấu hình để giao tiếp với ESP32 thông qua địa chỉ IP của broker (ESP32) và cổng MQTT (mặc định là 1883).
 - Giao diện Blynk hiển thị các nút điều khiển (bật/tắt relay, LED) và biểu đồ thời gian thực để theo dõi dữ liệu cảm biến (nhiệt độ, độ ẩm).

Bước 2: Người dùng gửi lệnh từ ứng dụng Blynk

- Tương tác trên ứng dụng:
 - Người dùng nhấn nút bật/tắt trên giao diện ứng dụng Blynk để điều khiển thiết bị (ví dụ: relay, LED).
 - Lệnh được định nghĩa dưới dạng thông điệp MQTT bao gồm:
 - Chủ đề (Topic): Xác định thiết bị cần điều khiển, ví dụ: "home/relay1".
 - Payload (Dữ liệu): Nội dung của lệnh, ví dụ: "ON" để bật relay hoặc "OFF" để tắt relay.
- Gửi lệnh đến broker (ESP32):
 - Ứng dụng Blynk truyền thông điệp MQTT tới broker ESP32 qua mạng Wi-Fi.
 - Lệnh được broker tiếp nhận và lưu vào hàng đợi xử lý.

Bước 3: Xử lý lệnh tại ESP32 (Broker)

- Broker tiếp nhận lệnh:
 - ESP32 (broker) nhận thông điệp MQTT từ ứng dụng Blynk. Broker kiểm tra chủ đề (topic) của lệnh để xác định thiết bị client nào cần thực hiện lệnh.
 - Ví dụ: Nếu lệnh có topic "home/relay1", broker sẽ biết rằng đây là lệnh dành cho relay 1.
- Phân phối lệnh đến client (ESP32):
 - Broker gửi lại thông điệp MQTT đến ESP32 (client) thông qua mô hình publish/subscribe. Vì ESP32 trong hệ thống này cũng là client, nó sẽ trực tiếp nhận lệnh từ chính broker.

Bước 4: Thực thi lệnh tại ESP32 (Client)

- ESP32 client xử lý lệnh:
 - Sau khi nhận thông điệp từ broker, ESP32 (client) phân tích nội dung payload:

- Nếu payload là "ON", client sẽ kích hoạt relay để bật thiết bị điện (ví dụ: đèn, quạt).
 - Nếu payload là "OFF", client sẽ ngắt relay để tắt thiết bị.
- Đồng thời, trạng thái relay được cập nhật trên GPIO của ESP32 để giữ tính đồng bộ.
- Phản hồi trạng thái:
 - ESP32 gửi lại trạng thái hiện tại (ON/OFF) đến broker để cập nhật lên ứng dụng Blynk. Điều này đảm bảo người dùng luôn có thông tin chính xác về trạng thái thiết bị.

Bước 5: Đọc dữ liệu cảm biến

- Thu thập dữ liệu từ DHT22:
 - ESP32 định kỳ đọc dữ liệu từ cảm biến DHT22, bao gồm:
 - Nhiệt độ (°C).
 - Độ ẩm (%).
 - Dữ liệu được mã hóa thành thông điệp MQTT để gửi đến broker ESP32.
- Cập nhật lên ứng dụng Blynk:
 - Broker ESP32 gửi dữ liệu từ cảm biến đến ứng dụng Blynk thông qua topic, ví dụ: "home/sensor/temp" (nhiệt độ) hoặc "home/sensor/humidity" (độ ẩm).
 - Ứng dụng hiển thị dữ liệu này theo thời gian thực trên giao diện để người dùng dễ dàng theo dõi.

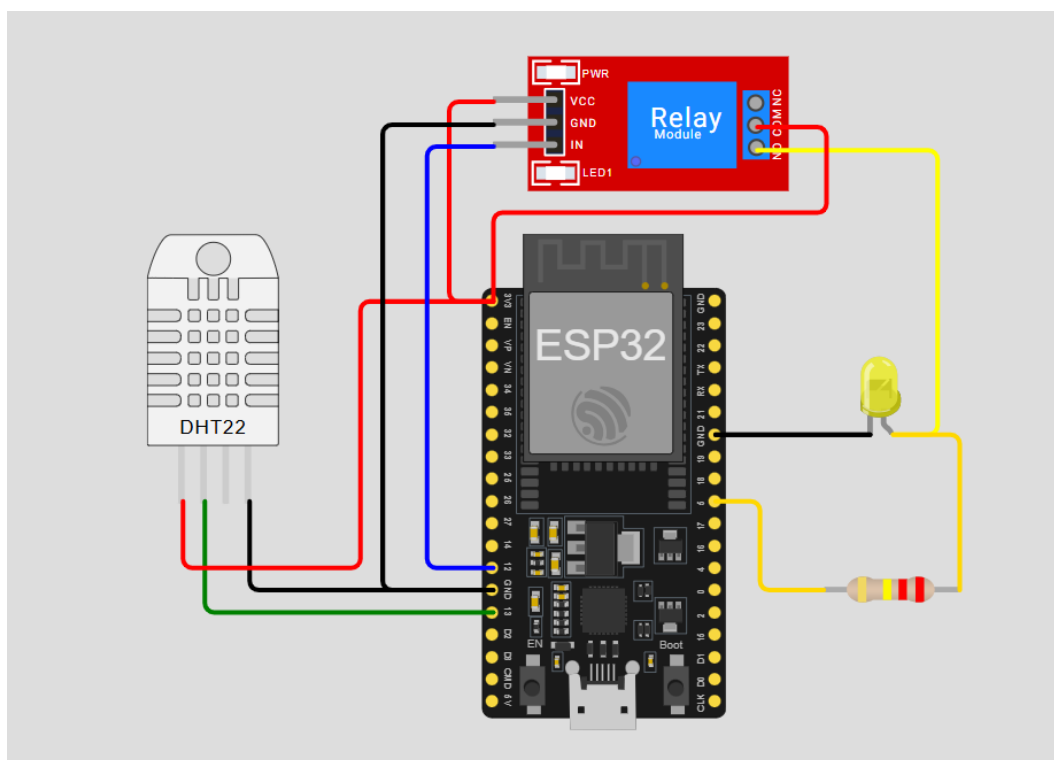
2.2.3. Hoạt động liên tục và tương tác hai chiều

- Hệ thống hoạt động liên tục để nhận lệnh mới từ ứng dụng hoặc cập nhật dữ liệu từ cảm biến:
 - Người dùng gửi lệnh: Có thể bật/tắt thiết bị bất kỳ lúc nào từ ứng dụng Blynk.
 - ESP32 gửi dữ liệu cảm biến: Đảm bảo cập nhật nhiệt độ, độ ẩm theo chu kỳ, giúp người dùng nắm bắt môi trường hiện tại.

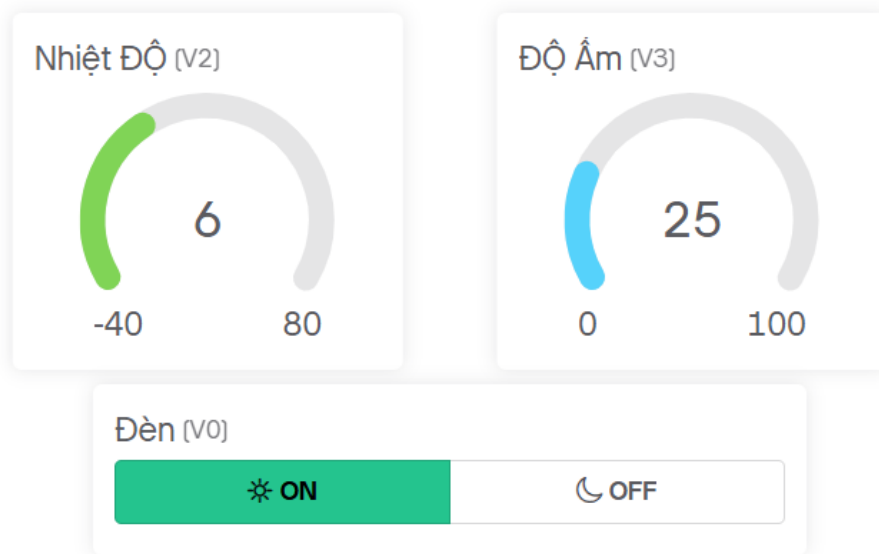
2.2.4. Ưu điểm của mô hình này

- Tối giản phần cứng:
 - o ESP32 vừa làm broker vừa làm client, giúp loại bỏ nhu cầu sử dụng thêm một broker bên ngoài như Mosquitto.
- Thời gian thực:
 - o Hệ thống phản hồi nhanh, phù hợp với các ứng dụng yêu cầu độ trễ thấp.
- Dễ dàng mở rộng:
 - o Broker ESP32 có thể dễ dàng quản lý thêm nhiều client khác như các cảm biến bổ sung (CO2, ánh sáng) hoặc thiết bị điều khiển khác (quạt, máy bơm).

2.3. Sơ đồ Wokwi



Hình 1: Sơ đồ Wokwi



Hình 2: Sơ đồ Blynk

CHƯƠNG 3: KẾT LUẬN

1. Tóm tắt

- Hệ thống điều khiển thiết bị từ xa thông qua giao thức MQTT với ESP32 đã được triển khai thành công và hoạt động tốt trên nền tảng mô phỏng Wokwi. Sử dụng ESP32 làm broker và client MQTT, hệ thống cung cấp giải pháp điều khiển và giám sát thiết bị hiệu quả, tối ưu hóa về cả chi phí và cấu trúc kỹ thuật. Ứng dụng Blynk được tích hợp làm giao diện, giúp người dùng dễ dàng điều khiển thiết bị từ xa cũng như theo dõi các thông số môi trường thời gian thực.

2. Đánh giá

- Dự án đã đạt được các mục tiêu đề ra, cụ thể:
 - o Khả năng điều khiển từ xa: Người dùng có thể bật/tắt các thiết bị như relay, LED từ bất kỳ nơi đâu thông qua ứng dụng Blynk.
 - o Giám sát môi trường: Dữ liệu nhiệt độ và độ ẩm từ cảm biến DHT22 được thu thập và hiển thị theo thời gian thực trên ứng dụng.
 - o Tối giản hệ thống: Với việc ESP32 đảm nhiệm cả vai trò broker và client, hệ thống đã loại bỏ sự cần thiết của một broker bên ngoài như Mosquitto, giảm bớt chi phí và độ phức tạp.

3. Bài học

- Dự án cung cấp nhiều kiến thức và kinh nghiệm thực tiễn, bao gồm:
 - o Hiểu biết sâu hơn về giao thức MQTT: Cách thức hoạt động của mô hình publish/subscribe, cùng với vai trò của broker và client.
 - o Kỹ năng thiết kế hệ thống IoT: Việc tích hợp các cảm biến, thiết bị điều khiển và ứng dụng giám sát đã mang lại cái nhìn toàn diện về mô hình IoT.
 - o Quản lý tài nguyên ESP32: Tìm hiểu cách tối ưu hóa hiệu suất của ESP32 khi đảm nhận hai vai trò (broker và client).

4. Hướng phát triển

- Dự án hiện tại mới dừng ở mức mô phỏng trên Wokwi. Những hướng phát triển tiếp theo có thể bao gồm:

- Triển khai thực tế: Xây dựng hệ thống với các thiết bị phần cứng thật để kiểm tra và đánh giá hoạt động trong môi trường thực tế.
- Nâng cao bảo mật: Tích hợp các phương pháp mã hóa (ví dụ: TLS) vào giao thức MQTT để bảo vệ dữ liệu khi truyền trên mạng.
- Mở rộng tính năng: Thêm các cảm biến (như cảm biến chuyển động, ánh sáng) hoặc tích hợp AI để phân tích dữ liệu và dự đoán hành vi người dùng.

TÀI LIỆU THAM KHẢO

1. **MQTT.org:** Tài liệu chính thức về giao thức MQTT, cung cấp thông tin chi tiết về cách thức hoạt động và ứng dụng của MQTT.
 - Website: <https://mqtt.org>
2. **Wikipedia ESP32:** Tài liệu kỹ thuật chi tiết về ESP32, bao gồm đặc điểm, thông số kỹ thuật và cách sử dụng.
 - Website: <https://vi.wikipedia.org/wiki/ESP32>
3. **Wokwi Simulator:** Nền tảng mô phỏng các dự án IoT với ESP32, hỗ trợ thử nghiệm mã nguồn và cấu hình phần cứng.
 - Website: <https://wokwi.com>
4. **Blynk.io:** Nền tảng giám sát và điều khiển IoT, cung cấp giao diện trực quan để quản lý thiết bị.
 - Website: <https://blynk.io>

PHỤ LỤC

1. Bảng nối chân:

Thiết bị	Chân kết nối	Nối tới	Màu dây
ESP32	TX	Serial Monitor RX	N/A
ESP32	RX	Serial Monitor TX	N/A
DHT22	GND	ESP32 GND	Đen
DHT22	SDA	ESP32 chân 13	Xanh lá
DHT22	VCC	ESP32 chân 3V3	Đỏ
Relay Module	NO	LED (chân A)	Vàng
Relay Module	COM	ESP32 chân 3V3	Đỏ
Relay Module	VCC	ESP32 chân 3V3	Đỏ
Relay Module	GND	ESP32 GND	Đen
Relay Module	IN	ESP32 chân 12	Xanh lam
LED (vàng)	A	Relay Module NO	Vàng
LED (vàng)	C	ESP32 GND	Đen
LED	A	Điện trở (chân 1)	Vàng
Điện trở	Chân 2	ESP32 chân 5	Vàng