TỔ CHỨC THỰC HIỆN

# Yêu cầu của dự án

|  |  |  |
| --- | --- | --- |
| Yêu cầu | Chức năng, Thông số,.. | Mức độ ưu tiên |
| Thiết bị Device (vitrual hoặc physic): MQTT,  CoAP (cảm biến ánh sáng, nhiệt độ cần low power) |  |  |
| Gateway (edge) (sử dụng HTTP để kết nối tới cloud) |  |  |
| Cloud (server) (có thể sử dụng cloud có sẵn ) |  |  |
| App (mobile) (tính năng nâng cao, có thể làm trên điện thoại, hoặc đơn giản chỉ cần dùng trình duyệt để điều khiển) |  |  |
| Sử dụng truyền thông Wifi kết nối Devices với Gateway |  |  |
| Điều khiển local và điều khiển qua Internet (khi không có Internet thiết bị trong mạng vẫn có thể điều khiển local được) |  |  |

# Giới thiệu thành viên của dự án

|  |  |
| --- | --- |
| (ảnh) | Họ và tên: Nguyễn Trọng Phong  MSSV  Phụ trách công việc |
| (ảnh) | Họ và tên: Đào Văn Quân  MSSV  Phụ trách công việc |
| (ảnh) | Họ và tên: Thái Huy Hưng  MSSV  Phụ trách công việc |
|  |  |
|  |  |

# Kế hoạch thực hiện chung của dự án

|  |  |  |  |
| --- | --- | --- | --- |
| Nội dung | Kết quả cần đạt | Thời gian (tuần) | Ghi chú |
| Tìm hiểu về các bài toán liên quan | Báo cáo tổng quan, đánh giá | T1-T5 |  |
| Lên phương án sơ bộ | Sơ đồ khối chức năng, yêu cầu,… | T5-T |  |
| .... |  |  |  |

# Kế hoạch và nội dung thực hiện của từng thành viên

Nguyễn Văn A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện (theo tuần) | Ghi chú |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nguyễn Văn B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện (theo tuần) | Ghi chú |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nguyễn Văn C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện (theo tuần) | Ghi chú |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Tự đánh giá tỷ lệ đóng góp của từng thành viên trong dự án theo kế hoạch (trước khi thực hiện, thực hiện trong khi lên kế hoạch thực hiện dự án)

|  |  |  |
| --- | --- | --- |
| Người thực hiện | Tỷ lệ | Giải quyết được những vấn đề gì của dự án (cần ghi rõ để có cơ sở đánh giá tỷ lệ) |
| Nguyễn Văn A | Xx% |  |
| Nguyễn Văn B | Yy% |  |
| Nguyễn Văn C | Zz% |  |

NỘI DUNG THỰC HIỆN

# Nhật ký

*(ghi ngắn gọn, mỗi tuần ghi không quá 1 trang A4 đánh máy)*

## Tuần 1

### Tổ chức

* Địa điểm: OFFLINE
* Thời gian: 05/11/2022
* Tham dự: Đầy đủ
* Không tham dự: …….. Lý do:…..

### Nội dung cuộc họp:

* Nội dung 1: Tìm hiểu về các đặc trưng của IoT
* Nội dung 2: Tìm hiểu về mô hình IoT, mô hình kết nối IoT và Cloud
* Nội dung 3: Các thành viên bắt đầu tìm hiểu về dòng chip ESP32, được ứng dụng phổ biến trong các sản phẩm IoT.
* Nội dung 4: Tìm hiểu về môi trường lập trình (IDE) bằng Extension Platform IO trên VS Code
* Nội dung 5: Tìm hiểu về các giao thức MQTT, giao thức mạng HTTP

### Kết luận của buổi họp:….

## Tuần 2

### Tổ chức

* Địa điểm: OFFLINE
* Thời gian: 12/11/2022
* Tham dự: Đầy đủ
* Không tham dự: …….. Lý do:…..

### Nội dung cuộc họp:

* Nội dung 1: Thảo luận trao đổi về nội dung các thành viên đã tìm hiểu được trong tuần vừa rồi
* Nội dung 2: Lựa chọn thư viện lập trình (SDK – Software Development Kit) để lập trình ESP32

Có 2 gói thư viện chính đó là Arduino và ESP-IDF. Vậy ưu nhược điểm của các gói này ra sao.

* Arduino: Được viết dựa trên ESP – IDF, cũng có thể nói Arduino lib là thư viện con của ESP – IDF

Điểm mạnh: quen thuộc với user, code nhanh có nhiều nguồn hỗ trợ

Điểm yếu: không tối ưu được code và thiếu một số chức năng

* ESP – IDF: là gói thư viện do hãng phát hành, có nguyên 1 bản hướng dẫn sử dụng API, phù hợp với các bạn đọc được tiếng anh.

<https://docs.espressif.com/projects/esp-idf/en/stable/esp32/index.html>

Điểm mạnh: Tối ưu, được các nhà phát triển sử dụng trong các sản phẩm thương mại, tài liệu đầy đủ chính xác

Điểm yếu: Ít được cộng đồng hỗ trợ hơn

* Nội dung 3: Thảo luận chung về đề tài dự án và phân tích các yêu cầu

1. Yêu cầu Thiết bị Device (vitrual hoặc physic): MQTT,  CoAP (cảm biến ánh sáng, nhiệt độ cần low power)

**Lựa chọn MCU cho thiết bị Device: ESP32**

ESP32 là chip sinh ra cho các sản phẩm IOT (Internet of Things), chính vì vậy học cách kết nối ESP32 với Internet mới là quan trọng nhất đối với dòng chip này. Ngoài ra Low - Power mode (chế độ tiết kiệm năng lượng) cho các ứng dụng dùng pin và FOTA ( Firmware over The Air) nạp firmware từ xa cũng cần thiết không kém.

Ultra Low Power giải quyết vấn đề năng lượng cho ESP bởi vì sử dụng Wi-Fi sẽ rất ngốn điện đặc biệt khi chúng ta sử dụng pin phải tính toán rất kĩ.

**Giao thức truyền thông: MQTT**

**Lựa chọn cảm biến nhiệt độ DS18B20 với các thông số datasheet:**

Dải đo nhiệt độ -55oC ÷ +125oC

Độ chính xác: 0.5 oC

Độ phân giải hiển thị: 0.05 oC

Ngoài các thông số phía trên, chúng ta để ý tới một số các thông số khác của cảm biến như:

* Cảm biến có 2 chế độ
  + Active Mode kết hợp Low Power
  + Standby Mode
* Thời gian chuyển đổi nhiệt độ (Temperature Conversion Time)

với điều kiện độ phân giải 12 bit

* Cảm biến kỹ thuật số sử dụng giao tiếp One – Wire, tích hợp các chức năng cảnh báo nhiệt độ khi vượt ngưỡng và không bị suy hao tín hiệu đường dây dài.

### Kết luận của buổi họp:….

…….

## Tuần 3

### Tổ chức

* Địa điểm: OFFLINE
* Thời gian: 19/11/2022
* Tham dự: Đầy đủ
* Không tham dự: …….. Lý do:…..

### Nội dung cuộc họp:

* Nội dung 1: Thảo luận chung về đề tài dự án và phân tích các yêu cầu

1. Gateway (edge) (sử dụng HTTP để kết nối tới cloud)
2. Cloud (server) (có thể sử dụng cloud có sẵn)

**Lựa chọn MCU cho Gateway (edge): ESP32**

**Cloud: Thingspeak**

**Giao thức mạng sử dụng HTTP để kết nối tới Cloud**

ThingSpeak hỗ trợ ghi dữ liệu thông qua cả 2 giao thức HTTP GET và HTTP POST. Trong mã này mình sẽ sử dụng cả 2 ví dụ để làm mẫu. Bạn có thể tìm hiều thêm về [ESP32 HTTP GET](https://vidieukhien.xyz/2020/02/18/lap-trinh-esp32-voi-arduino-esp32-httpclient-get/) và [ESP32 HTTP POST](https://vidieukhien.xyz/2020/02/19/lap-trinh-esp32-voi-arduino-esp32-httpclient-post/)

**Quá trình gửi dữ liệu từ Gateway lên Cloud:** Sử dụng HTTP GET bạn có thể gửi dữ liệu lên ThingSpeak thông qua Request với URL

B1: Khởi tạo 1 URL chứa đường dẫn và API key

B2: Tiếp đó, sau khi đọc nhiệt độ cần convert chúng từ float sang String bằng hàm sprintf(…)

B3: Cộng 2 String lại để được URL hoàn chỉnh

B4: Sử dụng hàm http request để truyền kết quả lên

* Nội dung 2: Thảo luận chung về đề tài dự án và phân tích các yêu cầu

1. Sử dụng truyền thông Wifi kết nối Devices với Gateway

**Devices giao tiếp với Gateway thông qua MQTT và công nghệ truyền thông WiFi**

Các thành viên thảo luận trao đổi về giao thức MQTT, kết hợp tìm hiểu các ngoại vi của ESP32 và giao thức truyền nhận thông tin như WiFi, học cách kết nối ESP32 với Internet

* Nội dung 3: Tìm hiểu Lowpower mode trên ESP32 (chế độ tiết kiệm năng lượng) cho các ứng dụng dùng pin
* Nội dung 4: Thảo luận chung về đề tài dự án và phân tích các yêu cầu

1. Điều khiển local và điều khiển qua Internet (khi không có Internet thiết bị trong mạng vẫn có thể điều khiển local được)

Khi không có Internet thiết bị trong mạng vẫn có thể điều khiển local được

ESP32 cung cấp khả năng tạo kết nối Wifi Access Point thế nhưng nó không kết nối với mạng dây để truy cập Internet, vậy nên gọi là Soft – AP.

Soft AP thường được sử dụng khi chúng ta cần lấy thông tin của mạng Wifi khác cho thiết bị IOT khi chúng bắt đầu hoạt động

Khi chúng ta dùng ESP32 ở chế độ Điểm truy cập (AP), nó sẽ tạo ra một mạng WiFi. Do đó, cần đặt SSID, Password, local Address IP, subnet IP và gateway IP. Khác với ví dụ trên, ở chế độ Station ESP sẽ được Router cấp cho địa chỉ IP. Ở chế độ Access-Point chúng ta phải tự đặt địa chỉ IP cho ESP. Trong trường hợp này là 192.168.1.1 (Mặc định là 192.168.4.1).

1. App (mobile) (tính năng nâng cao, có thể làm trên điện thoại, hoặc đơn giản chỉ cần dùng trình duyệt để điều khiển)

Nhóm đang thảo luận sử dụng Nodered Broker để làm App chỉ cần dùng trình duyệt để điều khiển.

### Kết luận của buổi họp:….

## Tuần 4

### Tổ chức

* Địa điểm: ONLINE (27/11/2022)
* Thời gian:
* Tham dự: Đầy đủ
* Không tham dự: …….. Lý do:…..

### Nội dung cuộc họp:

* Nội dung 1: Phân tích, tổng hợp lại các yêu cầu của dự án

|  |  |
| --- | --- |
| Thiết bị Device | Các physical devices là các sensor nodes để đo các thông số môi trường như nhiệt độ, độ ẩm, ánh sáng, khí gas để gửi dữ liệu về Gateway để đánh giá.  Các cơ chấp chấp hành như relay để điều khiển bật tắt đèn được điều khiển thông qua internet hoặc điều khiển local hoặc công tắc vật lý khi không có internet  Giao thức giao tiếp giữa Device và Gateway là MQTT Protocol và mạng WiFi, với cấu trúc mạng được sử dụng là hình sao.  Tích hợp Sleep Mode trên các thiết bị Devices để đảm bảo tiết kiệm năng lượng do nguồn cung cấp là pin. |
| Gateway (edge) | Sử dụng ESP32 kết nối tới Cloud thông qua giao thức HTTP  Gateway đóng vai trò là Broker để các Device publish các topic đến và Gateway sẽ đẩy các topic đó lên Cloud |
| Cloud (server) | Sử dụng server miễn phí được cung cấp bởi Thingspeak hoặc Thingsboard  Ngoài ra, có thể tìm hiểu về Node – red cũng là một nền tảng IoT trực quan, phù hợp với dự án này |
| App (mobile) | Nhóm đang thảo luận sử dụng Nodered Broker để làm App chỉ cần dùng trình duyệt để điều khiển. |
| Điều khiển local | Điều khiển local sử dụng remote hoặc nút nhấn vật lý |
| Xây dựng các tính năng của hệ thống smarthome | Đo nhiệt độ, độ ẩm, ánh sáng của phòng.  Cảm biến khí gas và có cảnh báo  Điều khiển bật tắt các thiết bị 220V |
| Đánh giá hiệu năng hệ thống | Số bản tin trao đổi  Thời gian đo một mẫu  Quản lý tối thiểu mấy thiết bị và phương án mở rộng hệ thống. |

* Nội dung 2: Thảo luận trao đổi về nội dung các thành viên đã tìm hiểu được trong tuần vừa rồi
* Nội dung 3: Trao đổi về những khó khăn khi lập trình ESP32

Khó khăn 1: Cài đặt Extension Platform IO và thư viện lập trình

Khó khăn 2: Nạp code và debug bằng Serial Monitor

Khó khăn 3: Cài đặt thư viện cảm biến và code trên Platform IO

### Kết luận của buổi họp: Các thành viên đã cùng nhau làm rõ các yêu cầu của dự án, lên phương án thực hiện và giải quyết một số lỗi, khó khăn trong quá trình tìm hiểu lập trình ESP32.

# Phân tích các yêu cầu của dự án

|  |  |
| --- | --- |
| Yêu cầu 1 | Thiết bị Device (vitrual hoặc physic): MQTT,  CoAP (cảm biến ánh sáng, nhiệt độ cần low power) |

**Lựa chọn MCU cho thiết bị Device: ESP32**

ESP32 là chip sinh ra cho các sản phẩm IOT (Internet of Things), chính vì vậy học cách kết nối ESP32 với Internet mới là quan trọng nhất đối với dòng chip này. Ngoài ra Low - Power mode (chế độ tiết kiệm năng lượng) cho các ứng dụng dùng pin và FOTA ( Firmware over The Air) nạp firmware từ xa cũng cần thiết không kém.

Ultra Low Power giải quyết vấn đề năng lượng cho ESP bởi vì sử dụng Wi-Fi sẽ rất ngốn điện đặc biệt khi chúng ta sử dụng pin phải tính toán rất kĩ.

**Giao thức truyền thông: MQTT**

**Lựa chọn cảm biến nhiệt độ DS18B20 với các thông số datasheet:**

1. Dải đo nhiệt độ -55oC ÷ +125oC
2. Độ chính xác: 0.5 oC
3. Độ phân giải hiển thị: 0.05 oC

Ngoài các thông số phía trên, chúng ta để ý tới một số các thông số khác của cảm biến như:

1. Cảm biến có 2 chế độ

* Active Mode kết hợp Low Power
* Standby Mode

1. Thời gian chuyển đổi nhiệt độ (Temperature Conversion Time)

với điều kiện độ phân giải 12 bit

1. Cảm biến kỹ thuật số sử dụng giao tiếp One – Wire, tích hợp các chức năng cảnh báo nhiệt độ khi vượt ngưỡng và không bị suy hao tín hiệu đường dây dài.

|  |  |
| --- | --- |
| Yêu cầu 2 | Gateway (edge) (sử dụng HTTP để kết nối tới cloud) |

**Lựa chọn MCU cho Gateway (edge): ESP32**

**Cloud: Thingspeak**

**Giao thức mạng sử dụng HTTP để kết nối tới Cloud**

ThingSpeak hỗ trợ ghi dữ liệu thông qua cả 2 giao thức HTTP GET và HTTP POST. Trong mã này mình sẽ sử dụng cả 2 ví dụ để làm mẫu. Bạn có thể tìm hiều thêm về [ESP32 HTTP GET](https://vidieukhien.xyz/2020/02/18/lap-trinh-esp32-voi-arduino-esp32-httpclient-get/) và [ESP32 HTTP POST](https://vidieukhien.xyz/2020/02/19/lap-trinh-esp32-voi-arduino-esp32-httpclient-post/)

**Quá trình gửi dữ liệu từ Gateway lên Cloud:** Sử dụng HTTP GET bạn có thể gửi dữ liệu lên ThingSpeak thông qua Request với URL

1. Khởi tạo 1 URL chứa đường dẫn và API key
2. Tiếp đó, sau khi đọc nhiệt độ cần convert chúng từ float sang String bằng hàm sprintf(…)
3. Cộng 2 String lại để được URL hoàn chỉnh
4. Sử dụng hàm http request để truyền kết quả lên

|  |  |
| --- | --- |
| Yêu cầu 3 | Sử dụng truyền thông Wifi kết nối Devices với Gateway |

**Devices giao tiếp với Gateway thông qua MQTT và công nghệ truyền thông WiFi**

**Khái quát về MQTT:**

source1: <https://bom.so/k9gM1n>

source2:<https://vidieukhien.xyz/2018/03/14/mqtt-tutorial-co-ban-den-nang-cao-khai-niem/>

MQTT dựa trên giao thức TCP để truyền dữ liệu. Một biến thể, MQTT-SN, được sử dụng trên các phương tiện vận chuyển khác như UDP hoặc Bluetooth.

**Ưu điểm:** (Message Queuing Telemetry Transport) là một giao thức truyền nhận dữ liệu được sử dụng phổ biến trong các dự án IoT nhờ các đặc điểm sử dụng băng thông thấp để truyền dữ liệu thích hợp trong mạng truyền bị giới hạn, gói tin nhẹ, độ tin cậy cao và khả năng hoạt động được trong điều kiện đường truyền không ổn định đồng thời code đơn giản

**Bản chất**: MQTT dựa trên giao thức tin nhắn Publish – Subscribe, sử dụng TCP/IP để truyền dữ liệu.

Một hệ thống hoạt động theo giao thức MQTT gồm 2 phần chính là **Broker** (MQTT server) và **Client** (các thiết bị kết nối tới broker). Các client kết nối tới broker bằng cách đăng ký các kênh (gọi là **topic**). Quá trình đăng ký này gọi là **subscrib**e. Client sẽ nhận được dữ liệu khi topic đã đăng ký có dữ liệu mới. Dữ liệu trong MQTT được gọi là **message**. Quá trình client gửi message tới topic được gọi là **publish**.

* Về cơ bản, các thiết bị IoT (devices) sẽ đóng vai trò client và đăng ký với Broker dữ liệu nó muốn gửi lên hoặc dữ liệu nó muốn nhận được. Nhiệm vụ của Broker là thiết bị trung gian nhận dữ liệu truyền lên từ các thiết bị client và gửi dữ liệu đến Gateway.

Các khái niệm cơ bản về MQTT:

* **Client**: là device IoT muốn gửi/nhận dữ liệu trong network

VD: MQTT fx có thể dùng làm Client để test Broker

* **Broker**: là server cài MQTT server thu thập dữ liệu và giao tiếp với các client

VD: HiveMQ Broker, Mosquitto Broker, Nodered, …

* **Thiết bị trung gian**
* **Topic**: là chủ đề mà Broker tạo ra để Device gửi dữ liệu vào. Nó như 1 folder chứa dữ liệu nên có thể ở dạng đường dẫn.

VD: nodeWiFi32/dht11/temperature

* **Publish**: là bước gửi dữ liệu từ 1 thiết bị client đến Broker Server.

1. Trong bước này, thiết bị IoT sẽ xác định topic (là loại dữ liệu muốn gửi) và giá trị của topic đó

* **Gói tin Device gửi lên Broker Server**

1. Gói tin publish có thể là gói dữ liệu gửi vào topic hoặc gói cài đặt Retain Message hay LWT Message do cờ Retain, LWT quy định

* **Subscribe**: là bước đăng ký nhận dữ liệu từ Broker Server của 1 thiết bị Client.

Trong bước này, thiết bị IoT sẽ xác định loại dữ liệu mà nó muốn nhận. Khi Broker nhận được loại dữ liệu này từ 1 thiết bị client khác, nó sẽ gửi dữ liệu này tới thiết bị client đã đăng ký nhận

* **Unsubscribe**: là bước thông báo với Broker Server là Device Client không muốn nhận dữ liệu nữa.
* **QoS – Quality of Service:** Chất lượng đường truyền

|  |  |
| --- | --- |
| Yêu cầu 4 | Điều khiển local và điều khiển qua Internet (khi không có Internet thiết bị trong mạng vẫn có thể điều khiển local được) |

**Khi không có Internet thiết bị trong mạng vẫn có thể điều khiển local được**

ESP32 cung cấp khả năng tạo kết nối Wifi Access Point thế nhưng nó không kết nối với mạng dây để truy cập Internet, vậy nên gọi là Soft – AP.

Soft AP thường được sử dụng khi chúng ta cần lấy thông tin của mạng Wifi khác cho thiết bị IOT khi chúng bắt đầu hoạt động

Khi chúng ta dùng ESP32 ở chế độ Điểm truy cập (AP), nó sẽ tạo ra một mạng WiFi. Do đó, cần đặt SSID, Password, local Address IP, subnet IP và gateway IP. Khác với ví dụ trên, ở chế độ Station ESP sẽ được Router cấp cho địa chỉ IP. Ở chế độ Access-Point chúng ta phải tự đặt địa chỉ IP cho ESP. Trong trường hợp này là 192.168.1.1 (Mặc định là 192.168.4.1).

|  |  |
| --- | --- |
| Yêu cầu 5 | App (mobile) (tính năng nâng cao, có thể làm trên điện thoại, hoặc đơn giản chỉ cần dùng trình duyệt để điều khiển) |

Nhóm đang thảo luận sử dụng Nodered Broker để làm App chỉ cần dùng trình duyệt để điều khiển.

# Tìm hiểu các nghiên cứu, dự án liên quan

# Lựa chọn giải pháp và lên phương án thiết kế

# Thử nghiệm và Đánh giá

# Hoàn thiện sản phẩm

# Kết luận

(Đối chiếu lại kế hoạch dự kiến ở mục Tổ chức thực hiện, Nếu có thay đổi thì cần ghi rõ những phần thay đổi so với Kế hoạch, nêu lý do)

Nguyễn Văn A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện thực tế (theo tuần) | Trạng thái (hoàn thành, chư hoàn thành, không thực hiện) và kết quả đã đạt được so với dự kiến |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nguyễn Văn B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện thực tế (theo tuần) | Trạng thái (hoàn thành, chư hoàn thành, không thực hiện) và kết quả đã đạt được so với dự kiến |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nguyễn Văn C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Mô tả (tính năng, thông số,…) | Kết quả cần đạt | Thời gian thực hiện thực tế (theo tuần) | Trạng thái (hoàn thành, chư hoàn thành, không thực hiện) và kết quả đã đạt được so với dự kiến |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |