

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



BÁO CÁO GIỮA KÌ
HỌC PHẦN IOT VÀ ỨNG DỤNG

Đề tài: Thiết kế hệ thống IOT giám sát ánh sáng và điều khiển đèn tự động trong phòng học.

Giảng viên hướng dẫn: Kim Ngọc Bách

Lớp: D22CNPM02

Nhóm: 04

Thành viên:

- Đàm Anh Đức – B22DCCN219
- Lê Phương Nam – B22DCCN555
- Nguyễn Văn Nhất – B22DCCN579
- Nguyễn Việt Quang – B22DCCN651

Hà Nội – 2025

Mục lục

I. Tài liệu yêu cầu SRS(Software Requirement Specification)	4
1. Giới thiệu.	4
1.1. Mục tiêu:.....	4
1.2. Phạm vi.	4
1.3. Tiêu chí thành công	5
1.4. Kết quả mong đợi	5
2. Mô tả tổng quan.....	6
2.1. Môi trường.....	6
2.2. Ràng buộc	6
2.3. Giả định	6
3. Yêu cầu chức năng	7
3.1. Thu thập dữ liệu.....	7
3.2. Xử lý và ra quyết định	7
3.3. Điều khiển thiết bị (Relay)	7
3.4. Truyền và lưu trữ dữ liệu.....	7
3.5. Hiển thị và điều khiển từ xa	7
3.6. Tự động – Thủ công	7
4. Yêu cầu phi chức năng	8
4.1. Hiệu năng.....	8
4.2. Bảo mật.....	8
4.3. Độ tin cậy.....	8
4.4. Mở rộng	8
4.5. Chi phí và năng lượng	8
5. Ràng buộc kỹ thuật và chức năng.....	8
5.1. Ràng buộc kỹ thuật và môi trường	8
5.2. Ràng buộc pháp lý	9

5.3. Tài nguyên và thiết bị	10
5.4. Ràng buộc chức năng	10
6. Mô hình yêu cầu	11
6.1. Use Case Diagram	11
6.2. Mô hình 3 lớp IoT (Perception – Network – Application).....	12
II. Lý thuyết và công nghệ sử dụng	14
1. Lý thuyết sử dụng và công nghệ sử dụng	14
1.1. Cơ sở về Internet of Things (IoT).....	14
1.2. Cảm biến ánh sáng.....	15
1.3. Relay điều khiển đèn	16
1.4. Vi điều khiển ESP32	17
1.5. Lý thuyết điều khiển tự động (Ngưỡng – Hysteresis)	19
1.6. Phần mềm và nền tảng IOT	19
1.7. Tiêu chuẩn chiếu sáng trong phòng học	20

I. Tài liệu yêu cầu SRS(Software Requirement Specification)

1. Giới thiệu.

Vấn đề thực tế: Chúng ta thường phải bật/tắt điện bằng tay,

Hệ thống **IoT giám sát ánh sáng và điều khiển đèn tự động trong phòng học** được xây dựng nhằm **tự động hóa việc bật/tắt đèn dựa trên cường độ ánh sáng môi trường**, giúp tiết kiệm điện năng và tạo môi trường học tập thoải mái, ổn định cho sinh viên và giảng viên.

1.1. Mục tiêu:

Mục tiêu của hệ thống là xây dựng một mô hình **IoT (Internet of Things)** có khả năng:

- **Giám sát cường độ ánh sáng** trong phòng học thông qua cảm biến ánh sáng.
- **Tự động bật/tắt đèn** dựa trên mức ánh sáng môi trường nhằm tiết kiệm điện năng.
- **Cho phép người dùng điều khiển và theo dõi trạng thái đèn từ xa** qua ứng dụng di động hoặc nền tảng web.
- **Tiết kiệm điện năng**, giảm chi phí vận hành và nâng cao tính tự động hóa của cơ sở vật chất trong nhà trường.

1.2. Phạm vi.

Phạm vi	Mô tả chi tiết
Địa điểm áp dụng	Phòng học của trường đại học hoặc trung tâm đào tạo.
Thiết bị chính	<ul style="list-style-type: none">- Cảm biến ánh sáng: LDR hoặc BH1750 dùng để đo cường độ ánh sáng môi trường.- Vi điều khiển: ESP32 làm trung tâm xử lý và truyền dữ liệu qua Wi-Fi.- Module Relay: điều khiển bật/tắt đèn 220V hoặc LED mô phỏng đèn học.- Nguồn điện: 5V–12V DC cho mạch điều khiển, 220V AC cho đèn.- Máy chủ (Server): lưu trữ dữ liệu và cung cấp giao diện điều khiển.
Môi trường thử nghiệm	Mô hình phòng học thu nhỏ với điều kiện ánh sáng tự thay đổi.

Kết nối	Sử dụng Wi-Fi nội bộ (LAN) để truyền dữ liệu giữa ESP32 và máy chủ. Giao thức truyền thông dự kiến là HTTP hoặc MQTT để đảm bảo tốc độ và độ tin cậy.
Đối tượng sử dụng	- Nhân viên kỹ thuật. - Giảng viên.

1.3. Tiêu chí thành công

Tiêu chí	Mô tả chi tiết	Đơn vị đo	Giá trị mục tiêu
1. Độ chính xác đo sáng	Mức sai số giữa dữ liệu cảm biến thu thập được và giá trị thực tế đo bằng thiết bị chuyên dụng (lux meter).	% sai số	$\leq 5\%$
2. Độ trễ phản hồi (Latency)	Khoảng thời gian từ khi cảm biến ghi nhận thay đổi ánh sáng đến khi đèn được bật/tắt hoặc phản ứng trên giao diện.	giây	≤ 2 giây
3. Độ tin cậy hệ thống (Uptime)	Tỷ lệ thời gian hệ thống hoạt động ổn định không lỗi trong tổng thời gian chạy thử.	% uptime	$\geq 99\%$
4. Hiệu quả tiết kiệm năng lượng	Mức giảm điện năng tiêu thụ so với khi vận hành thủ công trong cùng điều kiện.	% tiết kiệm	$\geq 20\%$
5. Khả năng mở rộng	Dễ dàng thêm cảm biến, thiết bị đèn, hoặc phòng học mới mà không cần thay đổi cấu trúc chính.	Định tính	Có thể mở rộng dễ dàng
6. Tính thân thiện người dùng	Mức độ dễ sử dụng của giao diện điều khiển và giám sát, đánh giá qua phản hồi người dùng thử nghiệm.	Điểm trung bình (1–5)	≥ 4
7. Tính ổn định truyền thông	Tỷ lệ gói tin cảm biến gửi đến server thành công (không bị mất gói).	%	$\geq 98\%$
8. Chi phí triển khai	Tổng chi phí thiết bị, vật tư và thiết lập phần mềm cho một phòng học.	VNĐ	$\leq 1.000.000$ VNĐ/phòng

1.4. Kết quả mong đợi

- Hệ thống tự động bật/tắt đèn dựa trên cường độ ánh sáng trong phòng.
- Có thể giám sát và điều khiển đèn thủ công qua giao diện web/app.
- Dữ liệu ánh sáng và trạng thái đèn được hiển thị theo thời gian thực.
- Hệ thống hoạt động ổn định, phản hồi nhanh và dễ sử dụng.

2. Mô tả tổng quan

2.1. Môi trường

Hệ thống giám sát ánh sáng và điều khiển đèn tự động được triển khai trong **phòng học** (hoặc mô hình phòng học thu nhỏ).

Các thành phần hoạt động trong môi trường có **kết nối Wi-Fi**, bao gồm:

- **Cảm biến ánh sáng** (LDR hoặc BH1750) để đo độ rọi môi trường.
- **Vi điều khiển ESP32** đảm nhiệm việc xử lý dữ liệu và điều khiển relay.
- **Relay module** để bật/tắt đèn 220V (hoặc LED mô phỏng).
- **Nền tảng IoT** (như Blynk hoặc ThingSpeak) để hiển thị và điều khiển thiết bị từ xa.

Hệ thống được lập trình bằng **Arduino IDE**, giao tiếp giữa các thành phần thông qua **Wi-Fi**, và hiển thị kết quả trên **ứng dụng di động** hoặc **giao diện web**.

2.2. Ràng buộc

- **Nguồn điện ổn định** (5V cho ESP32, 220V cho đèn).
- **Phụ thuộc vào mạng Wi-Fi**: nếu mất kết nối Internet, chức năng điều khiển từ xa sẽ bị gián đoạn.
- **Độ nhạy cảm biến ánh sáng** có thể bị ảnh hưởng bởi điều kiện môi trường (bóng đổ, ánh sáng mặt trời, vị trí lắp đặt).
- **Giới hạn công suất relay** (thường $\leq 10A - 250V$ AC).
- **Chuẩn chiếu sáng trong phòng học**: duy trì độ rọi trong khoảng **300–500 lux** theo TCVN 7114-1:2008.

2.3. Giả định

- Người dùng có **kết nối Wi-Fi ổn định** trong khu vực lắp đặt.
- Hệ thống được **lắp đặt đúng kỹ thuật**, đảm bảo an toàn điện.
- Người dùng có **ứng dụng Blynk** (hoặc nền tảng tương tự) để điều khiển và theo dõi.
- Các cảm biến, relay và ESP32 hoạt động bình thường trong điều kiện môi trường phòng học.
- Mô hình có thể mở rộng và triển khai thực tế ở nhiều phòng học khác nhau.

3. Yêu cầu chức năng

3.1. Thu thập dữ liệu

- ESP32 đọc liên tục giá trị ánh sáng từ **cảm biến LDR hoặc BH1750**.
- Dữ liệu thu thập là **độ rọi (lux)**, phản ánh cường độ sáng môi trường.

3.2. Xử lý và ra quyết định

- So sánh giá trị đo được với **ngưỡng chiếu sáng tiêu chuẩn** (ví dụ 300 lux theo TCVN 7114-1:2008).
- Áp dụng **thuật toán điều khiển tự động (hysteresis)** để tránh bật/tắt đèn liên tục khi ánh sáng dao động quanh ngưỡng.

Ví dụ: nếu ánh sáng < 280 lux \rightarrow bật đèn; nếu > 320 lux \rightarrow tắt đèn.

3.3. Điều khiển thiết bị (Relay)

- Gửi tín hiệu điều khiển **bật/tắt relay** tương ứng với đèn 220V.
- Đảm bảo **an toàn cách ly điện áp cao – thấp**.

3.4. Truyền và lưu trữ dữ liệu

- Gửi dữ liệu cảm biến (lux, trạng thái đèn) lên **nền tảng cloud (Blynk, Firebase, hoặc ThingSpeak)** qua Wi-Fi.
- Cho phép **lưu trữ và phân tích lịch sử chiếu sáng**.

3.5. Hiển thị và điều khiển từ xa

- Người dùng xem dữ liệu ánh sáng và trạng thái đèn trên **ứng dụng di động Blynk**.
- Có thể **điều khiển bật/tắt đèn thủ công** qua ứng dụng (chế độ override).
- Giao diện hiển thị gồm:
 - Widget hiển thị độ rọi (lux)
 - Nút điều khiển đèn (manual)
 - Biểu đồ lịch sử độ sáng (tùy chọn)

3.6. Tự động – Thủ công

- Hai chế độ hoạt động:
 - **Tự động:** điều khiển theo cảm biến.
 - **Thủ công:** người dùng điều khiển trực tiếp qua app Blynk.

4. Yêu cầu phi chức năng

4.1. Hiệu năng

- **Độ trễ (Latency):** ≤ 1 giây giữa hành động người dùng trên app và phản hồi của đèn.
- **Tần suất đọc cảm biến:** mỗi 1–2 giây/lần.
- **Tốc độ phản hồi:** bật/tắt relay tức thời (<100 ms).
- **Chịu tải:** hệ thống có thể mở rộng để giám sát **nhiều cảm biến và đèn** trong cùng một mạng.

4.2. Bảo mật

- Giao tiếp Wi-Fi và cloud phải được **mã hóa (SSL/TLS)**.
- Xác thực bằng **token Blynk** để ngăn truy cập trái phép.
- Có thể tích hợp **cơ chế phân quyền người dùng** (admin – user).

4.3. Độ tin cậy

- Hệ thống tự khởi động lại khi mất kết nối hoặc mất điện (auto reconnect).
- ESP32 có thể lưu **giá trị ngưỡng và trạng thái đèn cuối cùng** trong bộ nhớ EEPROM.
- Cảm biến và relay được **bảo vệ điện áp** chống nhiễu và quá tải.

4.4. Mở rộng

- Dễ dàng **thêm nhiều cảm biến ánh sáng hoặc relay** khác (theo phòng, khu vực).
- Có thể **tích hợp nền tảng khác** như Firebase hoặc MQTT broker.
- Cho phép **nâng cấp firmware OTA (Over-the-Air)** qua Wi-Fi.

4.5. Chi phí và năng lượng

- Tối ưu phần cứng, chỉ dùng ESP32, cảm biến và relay phổ thông.
- **Chi phí thấp (~200.000–300.000 VNĐ/bộ).**
- **Tiêu thụ năng lượng thấp**, phù hợp hoạt động 24/7.
- Dung lượng băng thông nhỏ (dữ liệu vài trăm byte/s).

5. Ràng buộc kỹ thuật và chức năng

5.1. Ràng buộc kỹ thuật và môi trường

Hệ thống IoT giám sát ánh sáng và điều khiển đèn tự động trong phòng học được thiết kế để hoạt động ổn định trong môi trường thực tế, tuy nhiên vẫn chịu ảnh hưởng bởi một số yếu tố sau:

- **Nhiệt độ:**
Các linh kiện như vi điều khiển ESP32, cảm biến ánh sáng LDR hoặc BH1750 có dải hoạt động tốt nhất trong khoảng **0°C đến 50°C**. Khi nhiệt độ vượt ngưỡng, độ chính xác đo sáng có thể giảm, và relay hoạt động không ổn định.
- **Độ ẩm:**
Môi trường có **độ ẩm cao (>80%)** có thể gây **oxy hóa chân linh kiện** hoặc **chập mạch**, đặc biệt nếu hệ thống đặt gần cửa sổ hoặc khu vực có hơi nước.
- **Nhiều sóng:**
Hệ thống sử dụng WiFi 2.4GHz nên có thể bị ảnh hưởng bởi các thiết bị phát sóng cùng tần số (modem WiFi, camera IP, thiết bị Bluetooth). Nhiều sóng mạnh có thể làm **mất kết nối tạm thời** hoặc **truyền dữ liệu chậm**.
- **Nguồn cấp:**
Vi điều khiển và cảm biến cần nguồn ổn định **5V DC (hoặc 3.3V)**. Nếu sử dụng adapter hoặc cổng USB kém chất lượng, hệ thống có thể **tự khởi động lại hoặc relay bật/tắt sai**.
Do đó, cần dùng nguồn có **dòng ra tối thiểu 2A** để đảm bảo hoạt động ổn định.

5.2. Ràng buộc pháp lý

Khi triển khai hệ thống IoT, cần tuân thủ các quy định và chuẩn kỹ thuật về truyền thông không dây và bảo mật dữ liệu:

- **Quy định tần số vô tuyến:**
Module ESP32 sử dụng sóng **WiFi 2.4GHz**, thuộc dải tần ISM được **Bộ Thông tin và Truyền thông cho phép sử dụng miễn phí**, nhưng phải đảm bảo **công suất phát sóng dưới giới hạn quy định** để không gây nhiễu cho các thiết bị khác.
- **Bảo mật dữ liệu cá nhân:**
Hệ thống có thể truyền dữ liệu trạng thái phòng học (bật/tắt đèn, độ sáng môi trường) lên server hoặc nền tảng IoT (ví dụ: Blynk, Firebase, MQTT).
Vì vậy, cần đảm bảo **bảo mật thông tin** bằng cách:
 - Sử dụng **mã hóa đơn giản hoặc token xác thực** khi gửi dữ liệu.
 - Không thu thập thông tin nhạy cảm của người dùng.
 - Giới hạn quyền truy cập chỉ cho các tài khoản được phép.

5.3. Tài nguyên và thiết bị

Do đây là hệ thống mô hình IoT nhỏ, các linh kiện sử dụng đều có **tài nguyên phần cứng hạn chế**, dẫn đến một số ràng buộc như sau:

- **Bộ nhớ và CPU:**

Vi điều khiển ESP32 có **RAM khoảng 520KB** và **Flash 4MB**, đủ khả năng chạy các chương trình điều khiển phức tạp hơn, bao gồm xử lý dữ liệu cảm biến, kết nối Wi-Fi, và truyền dữ liệu thời gian thực.

Tuy nhiên, **dung lượng bộ nhớ vẫn có giới hạn**, nên hệ thống **không phù hợp cho việc lưu trữ dữ liệu lớn hoặc xử lý học máy trực tiếp trên chip**; thay vào đó, dữ liệu nên được gửi lên **nền tảng Cloud** để lưu trữ và phân tích.

- **Dung lượng pin hoặc nguồn điện:**

Nếu hệ thống chạy bằng pin, thời gian hoạt động bị giới hạn do **module WiFi tiêu thụ điện năng cao** khi truyền dữ liệu.

Do đó, giải pháp tối ưu là **dùng nguồn adapter cắm trực tiếp** để đảm bảo hoạt động liên tục.

- **Giới hạn mở rộng thiết bị:**

Hệ thống cơ bản chỉ hỗ trợ **1 cảm biến ánh sáng và 1 relay**.

Nếu muốn mở rộng cho nhiều phòng học, cần **nâng cấp lên vi điều khiển ESP32** (nhiều chân I/O hơn, RAM lớn hơn) hoặc tích hợp **hệ thống mạng mesh WiFi**.

5.4. Ràng buộc chức năng

Hệ thống được thiết kế với các chức năng chính là giám sát ánh sáng và điều khiển đèn tự động. Tuy nhiên, phạm vi chức năng vẫn bị giới hạn trong một số khía cạnh:

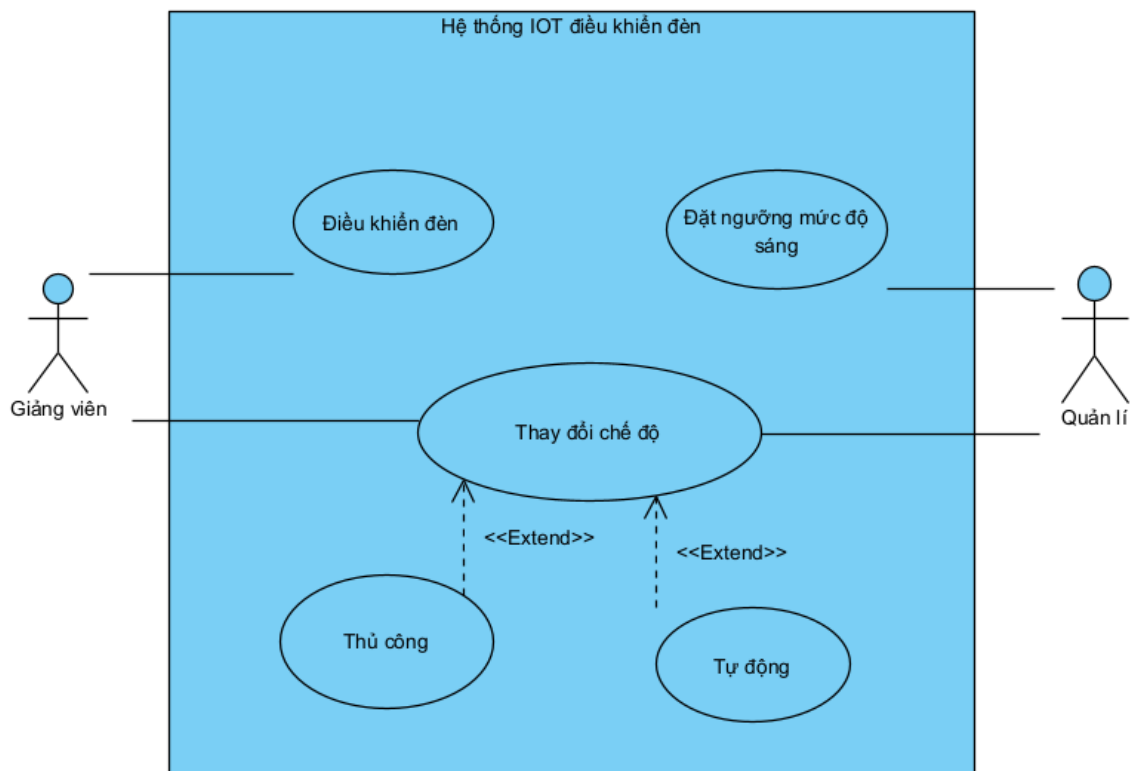
STT	Chức năng	Giới hạn / Mô tả
1	Giám sát ánh sáng	Độ sáng đo được chỉ phản ánh khu vực đặt cảm biến, không đại diện toàn phòng.
2	Điều khiển tự động	Bật/tắt đèn dựa theo ngưỡng ánh sáng cố định, chưa hỗ trợ điều chỉnh thông minh hoặc học thói quen.

STT	Chức năng	Giới hạn / Mô tả
3	Điều khiển từ xa	Chỉ hoạt động khi có kết nối WiFi và Internet ổn định. Mất mạng thì chỉ chạy chế độ tự động cục bộ.
4	Lưu trữ dữ liệu	Vi điều khiển không đủ bộ nhớ để lưu dữ liệu dài hạn, chỉ có thể gửi dữ liệu lên server IoT.
5	Thao tác thủ công	Khi người dùng bật/tắt đèn thủ công, hệ thống phải tạm ngưng chế độ tự động để tránh xung đột.
6	Phạm vi điều khiển	Mỗi bộ điều khiển chỉ quản lý được một đèn hoặc một khu vực. Muốn mở rộng cần thêm thiết bị mới.

6. Mô hình yêu cầu

6.1. Use Case Diagram

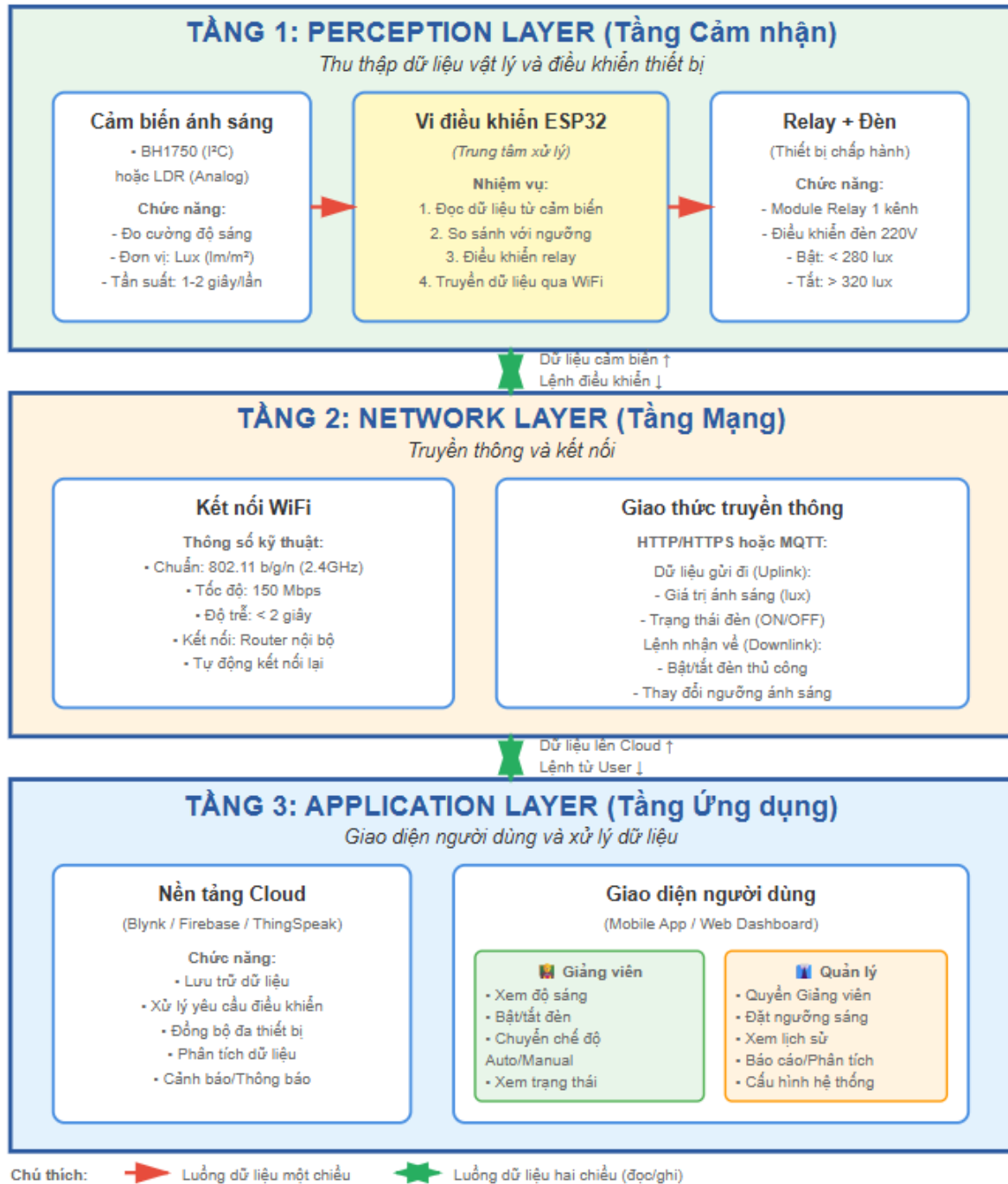
- **Actor :**
 - **Giảng viên:** điều khiển đèn(bật/tắt), thay đổi chế độ điều khiển đèn(thủ công, tự động).
 - **Quản lí:** Đặt ngưỡng của mức độ sáng, thay đổi chế độ điều khiển đèn(thủ công, tự động).
- **Use case:**
 - Điều khiển đèn.
 - Đặt ngưỡng mức độ sáng.
 - Thay đổi chế độ.
- Biểu đồ



6.2. Mô hình 3 lớp IoT (Perception – Network – Application)

Tầng	Thành phần cụ thể	Chức năng chính
Tầng cảm nhận (Perception Layer)	- Cảm biến ánh sáng (LDR hoặc BH1750) - Relay điều khiển đèn - ESP32	Thu thập dữ liệu cường độ sáng trong phòng học và điều khiển bật/tắt đèn.
Tầng mạng (Network Layer)	- Kết nối Wi-Fi của ESP32 - Giao thức truyền thông (MQTT hoặc HTTP)	Truyền dữ liệu từ cảm biến và trạng thái đèn lên nền tảng Cloud; nhận lệnh điều khiển từ người dùng.
Tầng ứng dụng (Application Layer)	- Ứng dụng Blynk / Web App - Người dùng: Giảng viên, Quản lý	Giao diện để Giảng viên điều khiển đèn và Quản lý đặt ngưỡng cường độ sáng, thay đổi chế độ điều khiển.

MÔ HÌNH 3 LỚP IoT - HỆ THỐNG GIÁM SÁT ÁNH SÁNG



II. Lý thuyết và công nghệ sử dụng

1. Lý thuyết sử dụng và công nghệ sử dụng

1.1. Cơ sở về Internet of Things (IoT)

Internet of Things (IoT) là một mô hình mạng lưới trong đó các thiết bị vật lý (things) như cảm biến, vi điều khiển, thiết bị điện... có khả năng **thu thập dữ liệu, trao đổi thông tin và điều khiển lẫn nhau thông qua Internet**.

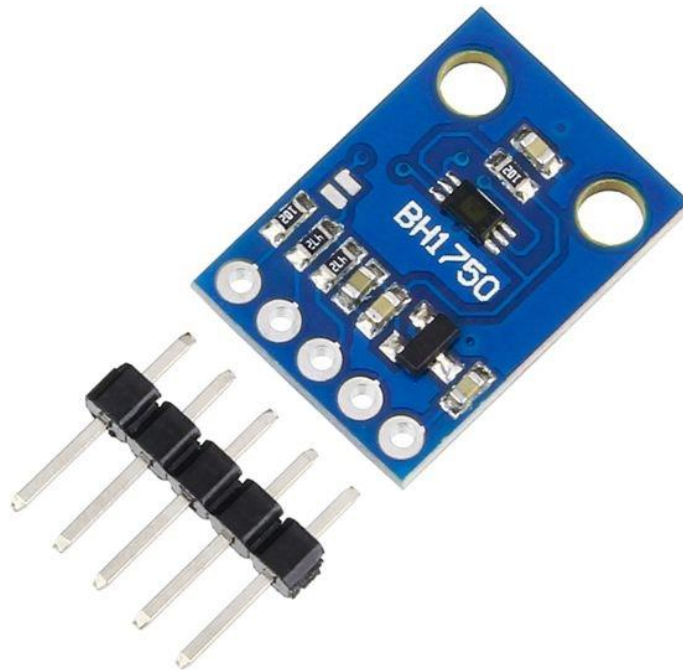
Trong hệ thống giám sát ánh sáng, IoT được ứng dụng để **kết nối vi điều khiển ESP32 với cảm biến ánh sáng, relay điều khiển đèn và ứng dụng di động**, giúp người dùng theo dõi và điều khiển thiết bị từ xa.

Một hệ thống IoT cơ bản bao gồm:

- **Cảm biến (Sensor):** thu thập dữ liệu môi trường như ánh sáng, nhiệt độ, độ ẩm,...
- **Bộ xử lý (Controller):** xử lý dữ liệu từ cảm biến và đưa ra quyết định điều khiển (ví dụ ESP32).
- **Thiết bị chấp hành (Actuator):** thực hiện hành động như bật/tắt đèn thông qua relay.
- **Mạng truyền thông:** kết nối Wi-Fi để truyền dữ liệu.
- **Nền tảng Cloud / Ứng dụng:** hiển thị thông tin, cho phép điều khiển từ xa và lưu trữ dữ liệu.

Trong đề tài này, ESP32 vừa đóng vai trò **bộ xử lý trung tâm** vừa là **thiết bị IoT kết nối Wi-Fi**, truyền dữ liệu lên nền tảng cloud như **Blynk, Firebase hoặc ThingSpeak**.

1.2. Cảm biến ánh sáng



Cảm biến ánh sáng là thiết bị có khả năng **chuyển đổi cường độ ánh sáng thành tín hiệu điện**. Hai loại cảm biến phổ biến được sử dụng trong đề tài là **LDR (Light Dependent Resistor)** và **BH1750**.

- **LDR (Light Dependent Resistor):** là điện trở quang có giá trị điện trở thay đổi theo cường độ ánh sáng chiếu vào.
 - Khi ánh sáng **mạnh**, điện trở **giảm** → điện áp đầu ra **tăng**.
 - Khi ánh sáng **yếu**, điện trở **tăng** → điện áp đầu ra **giảm**.
→ Dựa vào tín hiệu điện áp này, ESP32 có thể xác định mức sáng để bật/tắt đèn tự động.
- **BH1750:** là cảm biến ánh sáng kỹ thuật số, giao tiếp qua giao thức **I²C**, cho phép đo độ rọi trực tiếp theo đơn vị **lux (lm/m²)** với độ chính xác cao hơn LDR.

Nguyên lý hoạt động:

ESP32 liên tục đọc giá trị độ sáng từ cảm biến, so sánh với ngưỡng định trước (ví dụ 300 lux).

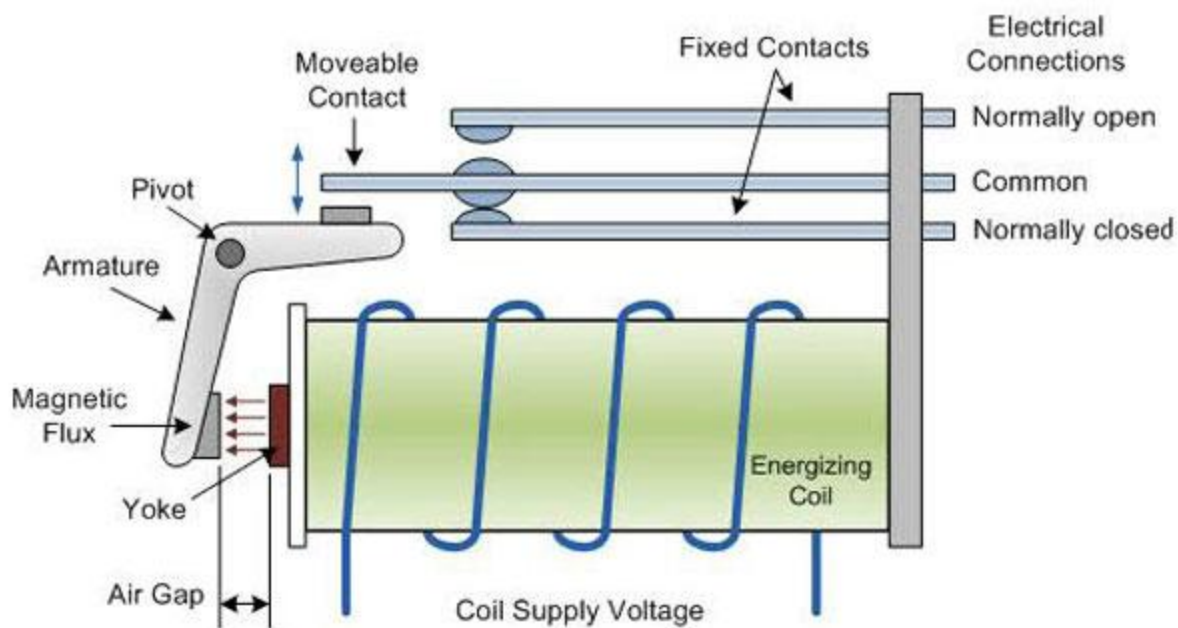
- Nếu độ sáng thấp hơn ngưỡng → bật relay để bật đèn.
- Nếu độ sáng cao hơn ngưỡng → tắt relay để tắt đèn.

Relay là **một công tắc điện tử được điều khiển bằng tín hiệu điện áp nhỏ** (thường 3.3V hoặc 5V). Khi nhận được tín hiệu điều khiển từ vi điều khiển (như ESP32), relay có thể **đóng/ngắt mạch điện công suất lớn** — ví dụ như đèn 220V AC.

Trong hệ thống này, relay đóng vai trò trung gian giữa **mạch điều khiển logic (ESP32)** và **mạch tải (đèn hoặc LED 220V)**, đảm bảo an toàn và cách ly điện áp.

1.3. Relay điều khiển đèn

Electromechanical Relay Construction



Cấu tạo cơ bản:

Một relay cơ điện (**Electromechanical Relay**) thông thường gồm các phần chính:

1. **Cuộn dây (Coil):** khi được cấp điện, tạo ra từ trường hút tiếp điểm.
2. **Tiếp điểm (Contacts):** gồm ba chân:
 - COM (Common): chân chung.
 - NO (Normally Open): bình thường hở, chỉ nối khi relay kích hoạt.
 - NC (Normally Closed): bình thường đóng, ngắt khi relay kích hoạt.
3. **Lò xo hồi (Spring):** giúp tiếp điểm trở lại trạng thái ban đầu khi ngắt điện.
4. **Mạch bảo vệ (diode, transistor):** trên module relay giúp chống nhiễu và bảo vệ vi điều khiển.

Nguyên lý hoạt động

Relay hoạt động theo cơ chế **điều khiển logic** – đóng/ngắt cơ học hoặc điện tử:

- Khi **ESP32 xuất tín hiệu HIGH (3.3V)** đến chân điều khiển của module relay:
 - Dòng điện chạy qua cuộn dây / mạch bán dẫn.
 - Tiếp điểm NO và COM đóng lại → **mạch tải khép kín** → **đèn sáng**.
- Khi **ESP32 xuất tín hiệu LOW (0V)**:
 - Cuộn dây không có dòng điện.
 - Tiếp điểm trở về trạng thái ban đầu → **mạch hở** → **đèn tắt**.

1.4. Vi điều khiển ESP32



Vi điều khiển (Microcontroller - MCU) là một máy tính nhỏ gọn được tích hợp trên một vi mạch duy nhất, chứa đựng bộ xử lý, bộ nhớ và các chân đầu vào/đầu ra (I/O). Trong dự án này, "bộ não" trung tâm của hệ thống là ESP32.

ESP32 không chỉ là một vi điều khiển thông thường mà là một hệ thống trên một vi mạch (System-on-Chip - SoC) mạnh mẽ được phát triển bởi Espressif Systems. Nó được thiết kế đặc biệt cho các ứng dụng IoT và thiết bị di động nhờ sự kết hợp giữa hiệu năng cao, chi phí thấp và khả năng kết nối không dây vượt trội.

Các đặc tính kỹ thuật chi tiết:

- Bộ xử lý (CPU): ESP32 tích hợp bộ xử lý lõi kép Tensilica Xtensa LX6 32-bit, có tốc độ xung nhịp lên đến 240 MHz. Kiến trúc lõi kép mang lại lợi thế lớn:
 - Lõi A có thể được dành riêng để quản lý các tác vụ nặng như kết nối Wi-Fi và Bluetooth.
 - Lõi B có thể tập trung hoàn toàn vào việc thực thi logic của chương trình chính, như đọc dữ liệu từ cảm biến và điều khiển cơ cấu chấp hành. Điều này đảm bảo hệ thống hoạt động mượt mà, ổn định và có khả năng phản hồi nhanh.
- Kết nối không dây: Đây là điểm mạnh lớn nhất của ESP32.
 - Wi-Fi: Hỗ trợ chuẩn 802.11 b/g/n, cho phép thiết bị kết nối trực tiếp với mạng Internet thông qua các bộ định tuyến (router) gia đình hoặc công cộng.
 - Bluetooth: Tích hợp cả Bluetooth cổ điển (Classic) và Bluetooth tiết kiệm năng lượng (Bluetooth Low Energy - BLE), rất hữu ích cho việc giao tiếp tầm gần với điện thoại thông minh (để cấu hình ban đầu) hoặc các thiết bị đeo thông minh.
- Bộ nhớ: Bao gồm bộ nhớ SRAM (dùng cho biến và thực thi chương trình) và bộ nhớ Flash (dùng để lưu trữ chương trình). Các phiên bản phổ biến thường có 520 KB SRAM và 4 MB Flash.
- Ngoại vi và chân I/O: ESP32 cung cấp một hệ thống ngoại vi cực kỳ phong phú:
 - GPIO (General Purpose Input/Output): Nhiều chân có thể được cấu hình làm đầu vào hoặc đầu ra kỹ thuật số.
 - ADC (Analog-to-Digital Converter): Các bộ chuyển đổi tín hiệu tương tự sang số, cho phép đọc giá trị từ các cảm biến analog (ví dụ: cảm biến ánh sáng, độ ẩm đất).
 - DAC (Digital-to-Analog Converter): Bộ chuyển đổi tín hiệu số sang tương tự, có thể tạo ra điện áp đầu ra tùy ý.

- Giao thức giao tiếp: Hỗ trợ đầy đủ các chuẩn phổ biến như UART, SPI, và I2C, giúp dễ dàng kết nối với hầu hết các loại cảm biến, màn hình và module khác trên thị trường.
- PWM (Pulse Width Modulation): Các chân có khả năng tạo ra tín hiệu điều chế độ rộng xung, rất cần thiết để điều khiển độ sáng của đèn LED và vị trí của động cơ servo.

Lý do lựa chọn: ESP32 được chọn vì nó là một giải pháp "tất cả trong một", giúp giảm thiểu sự phức tạp của phần cứng, tiết kiệm không gian và chi phí so với việc phải kết hợp một vi điều khiển (như Arduino) với các module Wi-Fi/Bluetooth rời.

Trong đề tài, ESP32:

- Thu thập dữ liệu từ cảm biến ánh sáng.
- Ra lệnh bật/tắt relay.
- Gửi dữ liệu lên ứng dụng cloud để hiển thị và điều khiển.

1.5. Lý thuyết điều khiển tự động (Ngưỡng – Hysteresis)

Hệ thống hoạt động dựa trên **nguyên lý điều khiển phản hồi (feedback control)**:

- Đầu vào: độ sáng môi trường (lux) từ cảm biến
- Bộ điều khiển: thuật toán trong ESP32.
- Đầu ra: trạng thái đèn(bật/tắt).
- Phản hồi: độ sáng đèn sau khi bật tắt.

1.6. Phần mềm và nền tảng IOT

- **Phần mềm lập trình: Arduino IDE**

Arduino IDE (Integrated Development Environment) là một phần mềm mã nguồn mở, đa nền tảng, giúp việc viết mã và nạp chương trình cho các bo mạch vi điều khiển trở nên cực kỳ đơn giản.

- Ngôn ngữ: Dựa trên C/C++ nhưng được đơn giản hóa với một bộ khung (framework) gọi là Wiring, cung cấp các hàm để sử dụng như `digitalWrite()`, `analogRead()`.
- Hỗ trợ ESP32: Thông qua trình quản lý Boards Manager, người dùng có thể dễ dàng cài đặt bộ công cụ (toolchain) cho ESP32, cho phép lập trình ESP32 với cú pháp và thư viện quen thuộc của Arduino.
- Hệ sinh thái thư viện: Sức mạnh lớn nhất của Arduino IDE là kho thư viện khổng lồ do cộng đồng đóng góp, hỗ trợ gần như mọi loại cảm biến,

module và giao thức. Điều này giúp đẩy nhanh quá trình phát triển sản phẩm mà không cần phải xây dựng mọi thứ từ đầu.

- **Nền tảng IoT: Blynk**

Nền tảng IoT (IoT Platform) là một dịch vụ đám mây đóng vai trò trung gian, kết nối các thiết bị vật lý với thế giới số. Nó xử lý việc giao tiếp, lưu trữ và trực quan hóa dữ liệu. Blynk là một nền tảng nổi bật nhờ sự thân thiện và tốc độ triển khai nhanh. Cấu trúc của Blynk:

1. Blynk App: Ứng dụng trên điện thoại (iOS/Android) cho phép người dùng tạo giao diện điều khiển bằng cách kéo và thả các widget (nút bấm, thanh trượt, biểu đồ, màn hình hiển thị giá trị...) mà không cần kỹ năng lập trình di động.
 2. Blynk Server: Máy chủ chịu trách nhiệm xử lý toàn bộ giao tiếp giữa phần cứng và ứng dụng di động. Người dùng có thể sử dụng máy chủ miễn phí của Blynk hoặc tự cài đặt máy chủ riêng.
 3. Blynk Libraries: Các thư viện dành cho phần cứng (như ESP32) giúp đơn giản hóa việc kết nối đến máy chủ Blynk và trao đổi dữ liệu với các widget trên ứng dụng.
- Luồng hoạt động: ESP32 sử dụng thư viện Blynk để kết nối với máy chủ Blynk qua Wi-Fi. Dữ liệu từ cảm biến được gửi lên các chân ảo (Virtual Pins). Ứng dụng Blynk trên điện thoại sẽ lắng nghe dữ liệu từ các chân ảo này và hiển thị lên các widget tương ứng. Ngược lại, khi người dùng tương tác với widget (ví dụ: nhấn nút), ứng dụng sẽ gửi lệnh đến máy chủ, và máy chủ sẽ chuyển tiếp lệnh này về cho ESP32 để thực thi.

1.7. Tiêu chuẩn chiếu sáng trong phòng học

Theo tiêu chuẩn **TCVN 7114-1:2008 (hoặc EN 12464-1)**:

- Mức độ chiếu sáng yêu cầu cho phòng học: **300 – 500 lux**.
- Độ đồng đều ≥ 0.7 , đảm bảo ánh sáng phân bố đều và không gây chói.
Hệ thống sẽ điều chỉnh bật/tắt đèn để duy trì độ rọi trong khoảng tiêu chuẩn này.