

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



**MÔN HỌC
IoT VÀ ỨNG DỤNG**

Lớp	D22CNPM02
Tên đề tài	Cửa siêu thị tự động.
Danh sách	<ol style="list-style-type: none">1. Trần Văn Duy – B22DCCN1592. Nguyễn Đăng Hải - B22DCCN2673. Vũ Hoàng Nam - B22DCCN5674. Ngô Thanh Vân - B22DCCN891

Hà Nội – 2025

LỜI MỞ ĐẦU

Trong thời đại công nghiệp 4.0, việc ứng dụng Internet of Things (IoT) vào đời sống đang trở nên phổ biến và mang lại nhiều tiện ích vượt trội. Các hệ thống tự động không chỉ giúp tiết kiệm sức lao động mà còn nâng cao hiệu quả quản lý và trải nghiệm người dùng. Trong đó, mô hình “Cửa siêu thị thông minh tự động mở khi có người đến và hiển thị số lượng người bên trong” là một ứng dụng thực tế và mang tính thiết thực cao.

Đè tài hướng đến việc áp dụng các công nghệ cảm biến, vi điều khiển và lập trình IoT để xây dựng một hệ thống giám sát, điều khiển cửa tự động vừa tiện lợi, vừa an toàn, giúp nâng cao tính hiện đại cho các siêu thị, trung tâm thương mại hoặc cửa hàng lớn.

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1. Lý do chọn đề tài

Trong các siêu thị hiện nay, việc kiểm soát ra vào và đảm bảo không gian hoạt động an toàn, thông thoáng là rất quan trọng. Tuy nhiên, nhiều nơi vẫn sử dụng cửa thủ công hoặc hệ thống bán tự động, chưa tích hợp khả năng đếm số lượng người ra vào để hỗ trợ quản lý hiệu quả.

Bên cạnh đó, sau đại dịch COVID-19, nhu cầu về hạn chế tiếp xúc trực tiếp ngày càng cao, đặc biệt trong môi trường đông người như siêu thị. Do đó, việc nghiên cứu và thiết kế cửa tự động thông minh có khả năng nhận diện người và hiển thị số lượng người hiện có bên trong là cần thiết, vừa nâng cao tính tự động hóa, vừa giúp quản lý an toàn, chính xác.

Ngoài ra, đây còn là một ứng dụng trực quan của IoT, giúp sinh viên vận dụng kiến thức đã học vào thực tiễn, bao gồm cảm biến, vi điều khiển, xử lý dữ liệu và hiển thị thông tin theo thời gian thực.

1.2. Mục tiêu và ý nghĩa ứng dụng

a. Mục tiêu

- Thiết kế và xây dựng mô hình cửa siêu thị thông minh có khả năng:
 - Tự động mở khi có người đến gần, đóng lại khi không có người.
 - Đếm và hiển thị số lượng người đang có trong siêu thị thông qua màn hình LED hoặc giao diện web.
- Ứng dụng các công nghệ IoT để thu thập, xử lý và truyền dữ liệu
- Giúp sinh viên hiểu rõ quy trình thiết kế – phân tích – triển khai hệ thống IoT thực tế.

b. Ý nghĩa ứng dụng

- Thực tiễn:** Nâng cao tính tự động hóa cho siêu thị, giúp giảm nhân lực giám sát cửa và hạn chế tiếp xúc trực tiếp.
- Kỹ thuật:** Giúp người học rèn luyện kỹ năng làm việc với cảm biến (cảm biến siêu âm), module Wi-Fi (ESP8266), và điều khiển động cơ servo/motor.
- Giáo dục:** Là một ví dụ sinh động cho việc áp dụng kiến thức IoT vào đời sống thực tế.
- Xã hội:** Hướng đến mô hình siêu thị thông minh, hiện đại, góp phần phát triển đô thị thông minh (Smart City).

CHƯƠNG 2. PHÂN TÍCH & THIẾT KẾ HỆ THỐNG IoT

2.1. Mục tiêu hệ thống

Hệ thống được thiết kế để:

- Phát hiện người đến gần cửa:** Dùng cảm biến khoảng cách hoặc cảm biến chuyển động.
- Tự động điều khiển cửa:** Động cơ servo mở cửa khi phát hiện người, đóng cửa khi không còn người.
- Đếm số người ra/vào:** Sử dụng hai cảm biến hồng ngoại hoặc siêu âm để phân biệt chiều di chuyển.
- Hiển thị dữ liệu:** Số lượng người trong siêu thị hiển thị trên màn hình LED hoặc trên web/app qua kết nối Wi-Fi.

2.2. Phạm vi triển khai

Phạm vi	Mô tả chi tiết
Địa điểm áp dụng	Siêu thị, cửa hàng tiện lợi, trung tâm thương mại, khu vực công cộng
Thiết bị chính	Cảm biến siêu âm, vi điều khiển ESP8266, động cơ servo, LED, module Wi-Fi
Phạm vi đếm	Tối đa 10 người trong mô hình thử nghiệm (có thể mở rộng)
Môi trường thử nghiệm	Mô hình thu nhỏ hoặc cửa giả lập
Kết nối	Nội bộ Wi-Fi hoặc Internet
Giao diện hiển thị	LED hoặc web dashboard hiển thị số người hiện tại

2.3. Tiêu chí thành công (KPIs)

STT	Tiêu chí	Mục tiêu cụ thể	Cách đo lường
1	Phát hiện người chính xác	$\geq 95\%$	Kiểm tra với 10 lượt người ra/vào

2	Cửa tự động hoạt động đúng	100%	Cửa mở/dóng đúng tín hiệu cảm biến
3	Đếm người chính xác	$\geq 90\%$	So sánh số đếm với thực tế
4	Hiển thị dữ liệu ổn định	Không lỗi trong 1 giờ hoạt động	Theo dõi hoạt động liên tục
5	Giao diện dễ quan sát	Dữ liệu hiển thị rõ ràng, không trễ $> 2s$	Đo thời gian cập nhật

2.4. Kết quả mong đợi

- Hoàn thiện mô hình cửa siêu thị thông minh có khả năng hoạt động ổn định.
- Cửa tự động mở khi có người đến gần và đóng khi không có người.
- Số lượng người được hiển thị chính xác, cập nhật theo thời gian thực.
- Hệ thống có thể mở rộng để kết nối với ứng dụng web hoặc hệ thống quản lý giám sát từ xa.
- Nâng cao nhận thức và kỹ năng thiết kế hệ thống IoT tích hợp phần cứng và phần mềm.

2.5 Yêu cầu chức năng, phi chức năng

2.5.1 Yêu cầu chức năng

Số thứ tự	Tên chức năng	Mô tả chi tiết
1	Phát hiện người đến gần	Cảm biến siêu âm phát hiện khi có người tiến lại gần cửa siêu thị
2	Mở cửa tự động	Khi cảm biến phát hiện người, vi điều khiển ESP8266 gửi tín hiệu điều khiển động cơ servo để mở cửa.
3	Đếm số người ra/vào	Hệ thống sử dụng cặp cảm biến đếm người để ghi nhận mỗi lượt ra và vào, từ đó tính tổng số người hiện có trong siêu thị.
4	Hiển thị số người hiện tại trên LED	ESP8266 gửi dữ liệu số người đến màn hình LED để hiển thị cho nhân viên và khách hàng.

2.5.2 Yêu cầu phi chức năng.

Số thứ tự	Yêu cầu phi chức năng	Mô tả chi tiết cho hệ thống
1	Hiệu năng (Performance)	Hệ thống phải phản hồi nhanh khi phát hiện người đến gần. Quá trình đếm và hiển thị số

		người trên LED diễm ra gần như tức thời, đảm bảo hoạt động real-time.
2	Bảo mật (Security)	Hệ thống cần đảm bảo chỉ có người quản lý mới được phép reset bộ đếm hoặc thay đổi giới hạn số người tối đa. Nếu có kết nối IoT trong tương lai, cần bổ sung xác thực người dùng và mã hóa dữ liệu truyền.
3	Độ tin cậy (Reliability)	Hệ thống phải hoạt động ổn định liên tục 24/7 mà không bị lỗi cảm biến hoặc treo vì điều khiển.
4	Khả năng mở rộng (Scalability)	Thiết kế phần mềm và phần cứng cho phép mở rộng: có thể thêm cảm biến, kết nối Wi-Fi/MQTT, hoặc hiển thị thông tin lên website mà không cần thay đổi kiến trúc lõi.
5	Chi phí và năng lượng (Cost & Energy)	Hệ thống sử dụng linh kiện giá rẻ, tiết kiệm điện năng (cảm biến siêu âm, LED, servo nhỏ).

2.6 Hạn chế của hệ thống.

- Độ chính xác của cảm biến:** Có thể bị ảnh hưởng bởi ánh sáng môi trường, vật cản hoặc tốc độ di chuyển của người, dẫn đến việc đếm sai.
- Giới hạn xử lý của vi điều khiển:** Nếu có nhiều người ra/vào cùng lúc, hệ thống có thể không xử lý kịp thời.
- Không nhận diện được đối tượng:** Hệ thống chỉ dựa vào cảm biến chuyển động, không phân biệt người thật hay vật thể di chuyển.
- Phụ thuộc nguồn điện:** Nếu mất điện, hệ thống ngừng hoạt động hoàn toàn.
- Hiển thị đơn giản:** LED chỉ thể hiện số lượng, chưa có giao diện hiển thị chi tiết hoặc lưu lịch sử ra/vào.
- Không kết nối mạng:** Không thể giám sát hoặc điều khiển từ xa qua internet.

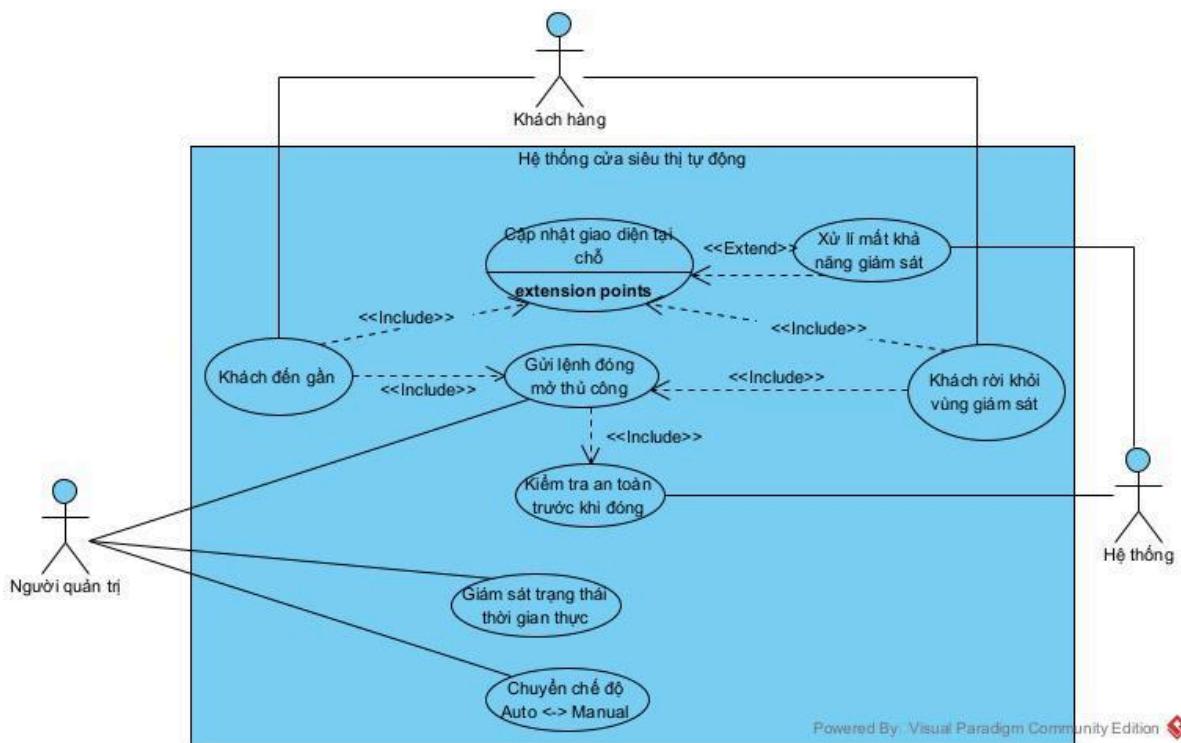
2.7 Bảng mô tả use case chính.

2.7.1 Use case Diagram (Mô tả bằng bảng)

Actor	Mục tiêu	Mô tả ngắn gọn	Điều kiện trước	Điều kiện sau
Người sử dụng	Tiến đến gần cửa để vào phòng	Cảm biến phát hiện người và gửi tín hiệu mở cửa.	Hệ thống hoạt động, cảm biến sẵn sàng.	Cửa tự động mở trong vài giây.
Người sử dụng	Đi qua cửa	Hệ thống xác định hướng di chuyển, cập nhật bộ đếm người.	Cửa đang mở, cảm biến đếm hoạt động.	Số người trong phòng được cập nhật.

Hệ thống (tự động)	Giám sát giới hạn người trong phòng	So sánh bộ đếm với giới hạn, bật còi/LED nếu vượt quá.	Bộ đếm hoạt động bình thường.	Cảnh báo được kích hoạt khi vượt ngưỡng.
Người quản ly	Reset hệ thống	Đặt lại giá trị đếm về 0 trong trường hợp cần thiết.	Hệ thống đang hoạt động.	Bộ đếm trở lại trạng thái ban đầu (0 người).

2.7.2 Use case Diagram(Bảng hình)



2.7.3 Mô hình 4 lớp IoT.

Tầng	Thành phần thực tế trong đồ án	Nhiệm vụ logic chính
Device Layer	Cảm biến siêu âm HC-SR04 + Servo SG90 + ESP8266	- Thu thập dữ liệu khoảng cách (cm) - Thực thi lệnh mở/dóng cửa
Gateway/Edge Layer	ESP8266 (NodeMCU)	- Đo khoảng cách liên tục - Xử lý logic tự động (5–10 cm → mở cửa)

		<ul style="list-style-type: none"> - Auto-close sau 5s - Ưu tiên lệnh thủ công - Đóng gói và publish dữ liệu
Cloud Layer	HiveMQ Cloud Broker	<ul style="list-style-type: none"> - Nhận, định tuyến tất cả message (QoS 0) - Đảm bảo kết nối ổn định - Lưu lịch sử (nếu tích hợp DB sau)

2.7.4 Software Requirement Specification (SRS)

a. Mục tiêu hệ thống:

Tự động phát hiện và mở cửa khi có người đến gần, đếm số người trong phòng nhằm đảm bảo an toàn, tiết kiệm và tiện lợi.

b. Môi trường hoạt động:

- Trong nhà, nguồn điện 5V – 12V.
- Vị trí điều khiển ESP8266.
- Cảm biến siêu âm, LED, servo hoặc motor cửa.

c. Yêu cầu phần cứng:

- 01 Vị trí điều khiển chính (ESP8266 Node MCU)
- 01 Cảm biến phát hiện người đến gần
- 01 LED hiển thị
- 01 Servo/motor cửa

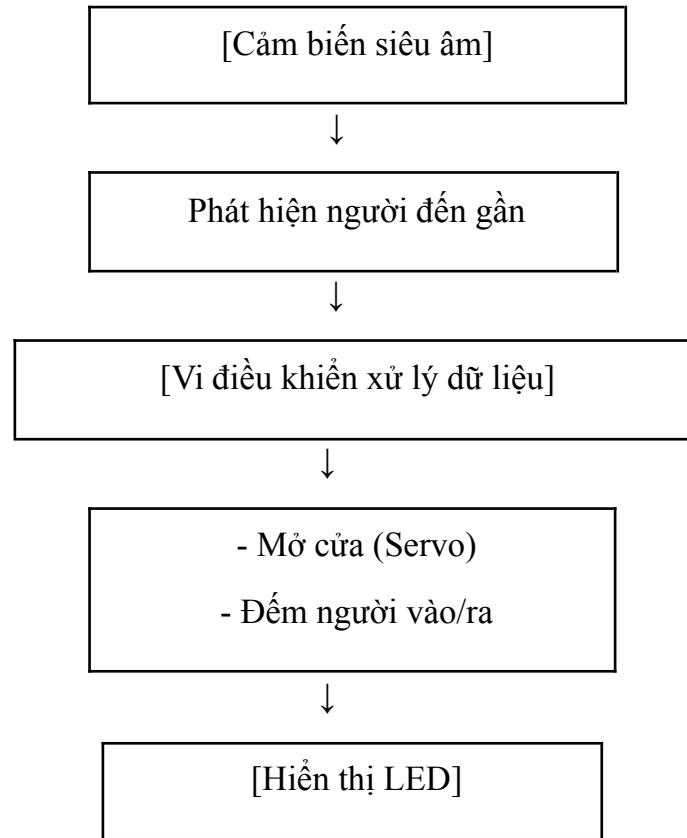
d. Yêu cầu phần mềm:

- Ngôn ngữ: C/C++ (Arduino IDE)
- Thư viện: Servo.h, LiquidCrystal.h,...
- Hoạt động liên tục, phản hồi thời gian thực.

e. Các ràng buộc:

- Giới hạn trong mô hình: số lượng áp dụng trong 1 mô hình nhỏ.
- Khoảng cách phát hiện người: $\leq 10\text{cm}$.
- Cửa tự động đóng sau 5 giây nếu không có người mới đến gần.

2.8 Sơ đồ khối



CHƯƠNG 3: CÁC CÔNG NGHỆ VÀ LÝ THUYẾT ÁP DỤNG

Chương này trình bày sâu về nền tảng công nghệ và các nguyên lý kỹ thuật được áp dụng trong đề tài. Việc hiểu rõ các thành phần này là cốt lõi để phân tích, thiết kế và triển khai thành công một hệ thống IoT hoàn chỉnh.

3.1 Vi điều khiển ESP8266 Node

Vi điều khiển (Microcontroller – MCU) là một máy tính thu nhỏ được tích hợp trên một vi mạch duy nhất, bao gồm bộ xử lý, bộ nhớ và các chân vào/ra (I/O). Trong dự án này, “bộ não” điều khiển của hệ thống là ESP8266.

ESP8266 không chỉ là một vi điều khiển thông thường mà là một System-on-Chip (SoC) mạnh mẽ do Espressif Systems phát triển. Đây là dòng chip cực kỳ phổ biến trong các dự án IoT nhờ mức giá rẻ, dễ lập trình và khả năng kết nối Wi-Fi tích hợp.

Các đặc tính kỹ thuật chi tiết:

- Bộ xử lý (CPU): ESP8266 sử dụng bộ xử lý Tensilica L106 32-bit, hoạt động ở tần số lên đến 80 MHz (có thể ép lên 160 MHz). Dù chỉ có 1 lõi, nhưng hiệu năng của ESP8266 vẫn đủ mạnh để:
 - Xử lý các tác vụ kết nối Wi-Fi.
 - Đóng thời thực thi các logic chương trình như đọc dữ liệu cảm biến, điều khiển động cơ, LED,...
- Kết nối không dây: Đây cũng là thế mạnh lớn nhất của ESP8266.
 - Wi-Fi: Hỗ trợ chuẩn 802.11 b/g/n, kết nối được với mọi loại router gia đình hoặc điểm phát Wi-Fi công cộng.
 - Bảo mật: Hỗ trợ WPA/WPA2, đảm bảo an toàn khi truyền dữ liệu.
- 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 50 KB SRAM và 4 MB Flash.
- Ngoại vi và chân I/O: ESP8266 cung cấp một hệ thống ngoại vi:
 - GPIO: Khoảng 9–11 chân tùy phiên bản, có thể dùng làm input/output.
 - ADC: 1 kênh ADC 10-bit để đọc cảm biến analog (ánh sáng, độ ẩm,...).
 - PWM: Hỗ trợ PWM để điều khiển LED hoặc động cơ servo.
 - Giao tiếp: Hỗ trợ đầy đủ UART, SPI, I2C

Lý do lựa chọn: ESP8266 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 rời.

3.2 Cảm biến siêu âm

Cảm biến siêu âm (Ultrasonic Sensor) là một loại cảm biến rất phổ biến trong các dự án IoT, robot, và tự động hóa — dùng để đo khoảng cách hoặc phát hiện vật cản bằng sóng âm tần số cao.

Nguyên lý hoạt động:

- Cảm biến siêu âm hoạt động dựa trên nguyên tắc phản xạ âm thanh (giống như dơi hay tàu ngầm sonar).
- Nó phát ra sóng âm tần số cao (khoảng 40 kHz), khi sóng này gặp vật cản, nó sẽ phản xạ lại.
- Bộ phận thu của cảm biến sẽ đo thời gian từ lúc phát đến lúc thu về (Δt).
- Từ đó tính ra khoảng cách (D) theo công thức:

$$D = (\text{Tốc độ âm} * \Delta t) / 2$$

3.3 Động cơ Servo

Động cơ Servo là một cơ cấu chấp hành vòng kín (closed-loop) có khả năng quay đến một vị trí góc cụ thể và giữ vững vị trí đó. Đây là điểm khác biệt cơ bản so với động cơ DC thông thường (chỉ quay liên tục khi được cấp điện).

Cấu tạo và cơ chế điều khiển:

Bên trong một servo bao gồm:

- Động cơ DC: Tạo ra chuyển động quay.
- Hộp số giảm tốc: Giảm tốc độ quay nhưng tăng mô-men xoắn (lực quay).
- Biến trớ (Potentiometer): Được nối với trực chính của servo, hoạt động như một cảm biến vị trí, liên tục đo góc quay hiện tại của trực.
- Mạch điều khiển: Mạch điện tử so sánh vị trí góc mong muốn (từ tín hiệu điều khiển) với vị trí góc thực tế (từ biến trớ). Nếu có sự sai lệch, mạch sẽ điều khiển động cơ DC quay cho đến khi sai số bằng 0.

Điều khiển bằng tín hiệu PWM:

Servo không được điều khiển bằng cách thay đổi điện áp. Thay vào đó, nó được điều khiển bằng một chuỗi xung tín hiệu PWM (Pulse Width Modulation). Vị trí của servo được quyết định bởi độ rộng của xung dương trong mỗi chu kỳ (thường là 20ms).

- Xung rộng ~1ms: Servo sẽ quay đến vị trí 0 độ.
- Xung rộng ~1.5ms: Servo sẽ quay đến vị trí 90 độ (trung tâm).
- Xung rộng ~2ms: Servo sẽ quay đến vị trí 180 độ.

Bằng cách tạo ra các xung có độ rộng trong khoảng 1ms đến 2ms, vi điều khiển ESP8266 có thể ra lệnh cho servo di chuyển chính xác đến bất kỳ góc nào trong phạm vi hoạt động của nó.

3.5 LED

Màn hình là phương tiện giao tiếp trực quan, cho phép hệ thống hiển thị thông tin trạng thái, dữ liệu cảm biến hoặc các thông báo cho người dùng.

- Màn hình LED (Light Emitting Diode):
 - Nguyên lý: Công nghệ LED sử dụng các điốt phát quang để hiển thị thông tin. Mỗi đèn LED sẽ phát sáng khi có dòng điện chạy qua, tạo thành các điểm sáng hiển thị chữ, số hoặc hình ảnh tùy theo cấu trúc của màn hình (ma trận LED hoặc 7 đoạn).
 - Ưu điểm:
 1. Độ sáng cao: Có thể nhìn rõ ngay cả trong môi trường có ánh sáng mạnh.
 2. Tuổi thọ dài: LED có thể hoạt động hàng chục nghìn giờ.
 3. Tiết kiệm năng lượng: Hiệu suất phát sáng cao, tiêu thụ ít điện năng.
 4. Phản hồi nhanh: Thời gian đáp ứng gần như tức thì, phù hợp cho các ứng dụng thời gian thực.
 5. Độ bền cơ học tốt: Ít bị hỏng khi va đập nhẹ.
 - Giao tiếp: Màn hình LED (loại ma trận hoặc 7 đoạn) thường giao tiếp với vi điều khiển như ESP8266 thông qua chuẩn SPI hoặc GPIO trực tiếp.

3.6 Phần mềm & 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 dễ sử dụng như digitalWrite(), analogRead().
- Hỗ trợ ESP8266: Thông qua trình quản lý Boards Manager, người dùng có thể dễ dàng cài đặt bộ công cụ (toolchain) cho ESP8266, cho phép lập trình ESP8266 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 ESP8266) 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: ESP8266 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 ESP8266 để thực thi.

Chương 4. Kế hoạch triển khai dự án

4.1 Phân chia công việc trong nhóm

Nhóm em được chia thành 4 thành viên, mỗi người đảm nhận 1 phần công việc cụ thể, đảm bảo tiến độ và chất lượng sản phẩm.

Các nhiệm vụ được phân bổ dựa theo thế mạnh kỹ thuật và khả năng của mỗi thành viên.

Bảng phân công nhiệm vụ cho báo cáo giữa kỳ:

ST T	Họ và tên	Nội dung phụ trách	Kết quả mong đợi
1	Ngô Thanh Vân	Lời mở đầu, giới thiệu đề tài, mục tiêu và phạm vi hệ thống, KPIs và kết quả mong đợi	Hoàn thiện phần mở đầu và bối cảnh báo cáo.
2	Nguyễn Đăng Hải	Phân tích yêu cầu, đặc tả phần mềm (SRS), sơ đồ Use Case và DFD	Phân tích logic hoạt động và sơ đồ minh họa hoàn chỉnh.
3	Vũ Hoàng Nam	Các tính năng dự kiến, công nghệ và lý thuyết áp dụng	Mô tả chi tiết phần cứng, phần mềm, công nghệ IoT sử dụng
4	Trần Văn Duy	Triển khai kế hoạch,	Hoàn thiện tiến độ nhóm, rà soát và tổng hợp báo cáo, đề xuất hướng mở rộng

4.2 Lịch trình thực hiện

Tiến độ dự án chia thành các khoảng thời gian để đảm bảo nhóm hoàn thành mô hình, kiểm thử và viết báo cáo đúng hạn.

STT	Thời gian dự kiến	Công việc chính	Người phụ trách
1	22/09/2025 - 25/09/2025	Xác định đề tài.	Cả nhóm
2	26/09/2025 - 01/10/2025	Phân tích và phân chia nội dung cho cả nhóm.	Duy
3	02/10/2025 - 13/10/2025	Tìm hiểu nội dung và viết nội dung thành 1 bản riêng.	Cả nhóm

4	12/10/2025 - 15/10/2025	Rà soát lại nội dung, tổng hợp và viết lại thành 1 bản báo cáo.	Duy
---	-------------------------	---	-----

CHƯƠNG 5: KẾT LUẬN

Sau thời gian tìm hiểu và phân tích, nhóm đã đưa ra đề tài “Cửa siêu thị thông minh”, một ứng dụng tiêu biểu của công nghệ Internet of Things (IoT) trong lĩnh vực tự động hóa.

Hệ thống được xây dựng dựa trên vi điều khiển ESP8266, có khả năng kết nối và xử lý tín hiệu từ cảm biến siêu âm để:

- Phát hiện người hoặc vật cản di chuyển trước cửa,
- Tự động điều khiển servo motor mở hoặc đóng cửa,
- Đếm số người ra/vào thông qua cảm biến siêu âm đặt ở lối đi,
- Hiển thị số lượng người hiện có trong siêu thị trên màn hình LED.