

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



**MÔN HỌC**  
**IoT VÀ ỨNG DỤNG**

Lớp  
Tên đề tài  
Danh sách

D22CNPM02  
Cửa siêu thị tự động.  
1. Trần Văn Duy – B22DCCN159  
2. Nguyễn Đăng Hải - B22DCCN267  
3. Vũ Hoàng Nam - B22DCCN567  
4. 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.

Đề 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 (PIR, Ultrasonic, IR), module Wi-Fi (ESP8266, ESP32), 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 chuyển động IR / siêu âm, vi điều khiển ESP32 hoặc Arduino, động cơ servo, LED, module Wi-Fi
Phạm vi đo 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/đóng 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

Stt	Tên chức năng	Mô tả chi tiết
1	Phát hiện người đến gần	Cảm biến hồng ngoại hoặc siêu âm (IR/Ultrasonic) 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 ESP32 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	ESP32 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.

Stt	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 diễn 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. Trong trường hợp mất điện, dữ liệu nên được lưu tạm (EEPROM hoặc bộ nhớ flash).
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 IR, 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.

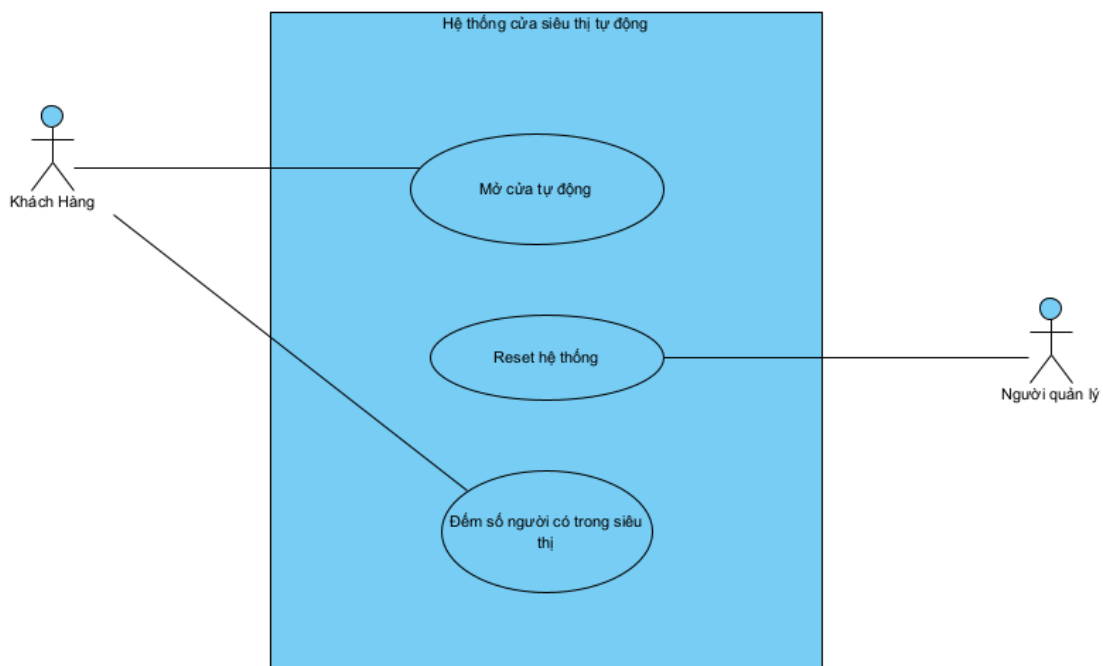
## 2.7 Bảng mô tả use case chính.

### 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	So sánh bộ đếm với giới hạn, bật	Bộ đếm hoạt động bình thường.	Cảnh báo được kích hoạt khi vượt ngưỡng.

	người trong phòng	còi/LED nếu vượt quá.		
Người quản lý	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 3 lớp IoT.

Lớp	Thành phần chính	Chức năng
Lớp cảm nhận (Perception Layer)	Cảm biến IR hoặc cảm biến siêu âm	Thu thập thông tin về chuyển động và số lượng người.
Lớp mạng (Network Layer)	Vi điều khiển (Arduino/ESP32), module Wi-Fi (nếu có)	Truyền dữ liệu giữa các thành phần hoặc đến hệ thống giám sát.
Lớp ứng dụng (Application Layer)	Màn hình LED	Hiển thị thông tin số lượng người có trong siêu thị, điều khiển mở cửa.



#### 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.
- Vi điều khiển Arduino Uno hoặc ESP32.
- Cảm biến PIR/siêu âm, LED, servo hoặc motor cửa.

##### c. Yêu cầu phần cứng:

- 01 Vi điều khiển chính (Arduino Uno/ESP32)
- 02 Cảm biến đếm người
- 01 Cảm biến phát hiện người đến gần
- 01 LED hiển thị
- 01 Servo/motor cửa

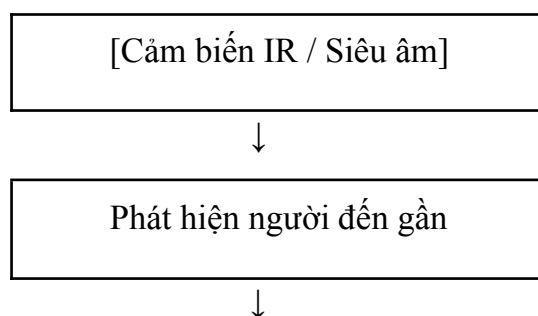
##### d. Yêu cầu phần mềm:

- 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:  $\leq 15\text{cm}$ .
- Cửa tự động đóng sau 5 giây nếu không có người mới đến gần.

#### 2.8 Sơ đồ khối



[Vi điều khiển xử lý dữ liệu]



- Mở cửa (Servo)  
- Đếm người vào/ra



[Hiển thị LED]

## CHƯƠNG 3: CÁC CÔNG NGHỆ VÀ LÝ THUYẾT ỨNG 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 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.

### 3.2 Cảm biến IR (Hồng ngoại)

Cảm biến hồng ngoại (Infrared - IR) Sensor là một thiết bị điện tử phát hiện bức xạ hồng ngoại. Trong các ứng dụng phổ thông, nó thường được sử dụng dưới dạng module phát hiện vật cản.

Nguyên lý hoạt động chi tiết:

Module cảm biến IR gồm 2 thành phần chính đặt cạnh nhau:

1. Đèn phát (IR Emitter): Là một đèn LED phát ra tia sáng trong dải hồng ngoại (có bước sóng dài hơn ánh sáng khả kiến, mắt người không nhìn thấy được).
2. Bộ thu (IR Receiver): Thường là một Photodiode hoặc Phototransistor, là linh kiện bán dẫn chỉ nhạy với ánh sáng hồng ngoại.

Cơ chế hoạt động dựa trên nguyên tắc phản xạ ánh sáng:

- Khi không có vật cản: Đèn phát liên tục phát ra tia hồng ngoại. Tia sáng này đi thẳng vào không gian và không bị phản xạ lại về phía cảm biến. Do đó, bộ thu không nhận được tín hiệu và chân tín hiệu (OUT) của module sẽ ở mức logic cao (HIGH).
- Khi có vật cản: Tia hồng ngoại phát ra sẽ đập vào bề mặt của vật cản và bị phản xạ ngược trở lại. Bộ thu nhận được tia phản xạ này, khiến điện trở của nó thay đổi và kích hoạt mạch xử lý trên module. Lúc này, chân tín hiệu (OUT) sẽ chuyển sang mức logic thấp (LOW).

Hầu hết các module IR đều có một biến trở nhỏ để người dùng có thể điều chỉnh độ nhạy (khoảng cách phát hiện) của cảm biến.

### 3.3 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 ( $\Delta t$ ).
- Từ đó tính ra khoảng cách (D) theo công thức:

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

### 3.4 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 trở (Potentiometer): Được nối với trục chính của servo, hoạt động như một cảm biến vị trí, liên tục đo góc quay hiện tại của trục.
- 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 trở). 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 ESP32 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 đi-ốt 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 ESP32 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 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.

## 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 ESP32, có khả năng kết nối và xử lý tín hiệu từ cảm biến hồng ngoại (IR Sensor) hoặc 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 2 cảm biến IR đặt ở lối đi,
- Hiển thị số lượng người hiện có trong siêu thị trên màn hình LED.