

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH
KHOA ĐIỆN ĐIỆN TỬ
BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

ĐỒ ÁN MÔN HỌC

**THIẾT KẾ HỆ THỐNG CÂN THÔNG MINH
SỬ DỤNG TRONG QUẢN LÝ KHO
CHO DOANH NGHIỆP SẢN XUẤT**

NGÀNH HỆ THỐNG NHÚNG VÀ IOT

Sinh viên: **TRẦN DUY VƯƠNG**
MSSV: 22139078
LƯU NGỌC THIÊN
MSSV: 22139065

Hướng dẫn: **TS. TRƯƠNG QUANG PHÚC**

TP. HỒ CHÍ MINH – 06/2025

MỤC LỤC

DANH MỤC HÌNH	X
DANH MỤC BẢNG	XI
CÁC TỪ VIẾT TẮT	XII
CHƯƠNG 1 TỔNG QUAN.....	1
1.1 ĐẶT VẤN ĐỀ.....	1
1.2 TỔNG QUAN TÌNH HÌNH NGHIÊN CỨU	2
1.3 MỤC TIÊU ĐỀ TÀI.....	3
1.4 Ý NGHĨA ĐỀ TÀI.....	3
1.5 PHẠM VI ĐỀ TÀI	3
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	5
2.1 CÁC CÔNG NGHỆ TRONG SOFTWARE	5
2.1.1 Django backend trong python.....	5
2.1.2 HTMX.....	6
2.1.3 PostgreSQL.....	7
2.1.4 DaisyUI.....	8
2.2 CÁC CHUẨN TRUYỀN THÔNG	8
2.2.1 Giao thức UART.....	8
2.2.2 Giao thức I ² C	9
2.2.3 Giao thức HTTP.....	9
2.3 CÁC LINH KIỆN ĐƯỢC LỰA CHỌN	10
2.3.1 Vi điều khiển ESP32-CAM	10
2.3.2 Vi điều khiển STM32F103C8T6	11
2.3.3 Mạch chuyển đổi ADC 24bit Loadcell HX711	12
2.3.4 Màn hình LCD, nút nhấn và nguồn	13
CHƯƠNG 3 THIẾT KẾ HỆ THỐNG	15
3.1 TỔNG QUAN.....	15
3.1.1 Yêu cầu hệ thống	16
3.1.2 Sơ đồ khối hệ thống	18

3.1.3	Sơ đồ nguyên lý	20
3.2	THIẾT KẾ PHẦN CỨNG	21
3.2.1	Lưu đồ hoạt động của phần cứng.....	21
3.3	THIẾT KẾ PHẦN MỀM	22
3.3.1	Thiết kế cơ sở dữ liệu	22
3.3.2	Lưu đồ quản lí đơn vị.....	24
3.3.3	Lưu đồ quản lí nguyên vật liệu	24
3.3.4	Lưu đồ quản lí hệ số quy đổi đơn vị cho từng loại nguyên vật liệu .	25
3.3.5	Lưu đồ nhập xuất tồn kho	26
3.3.6	Lưu đồ tìm kiếm lịch sử nhập xuất	28
3.4	KẾT QUẢ ĐẠT ĐƯỢC	29
3.4.1	Kết quả thi công phần cứng	29
3.4.2	Kết quả thi công phần mềm	31
CHƯƠNG 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....		35
4.1	KẾT LUẬN.....	35
4.2	HƯỚNG PHÁT TRIỂN	35
PHỤ LỤC		36
TÀI LIỆU THAM KHẢO		37

LỜI CẢM ƠN

Để hoàn thiện được đề tài này, chúng tôi xin chân thành cảm ơn CÔNG TY TNHH DỊCH VỤ XUẤT NHẬP KHẨU ĐÔNG NAM Á đã tạo điều kiện và môi trường thuận lợi để nhóm có cơ hội tiếp cận thực tế với quy trình sản xuất, vận hành và quản lý kho trong doanh nghiệp. Những kiến thức thực tiễn thu nhận được từ doanh nghiệp là nền tảng quý giá để nhóm triển khai và phát triển hệ thống một cách sát với nhu cầu thực tế.

Chúng tôi cũng xin gửi lời cảm ơn sâu sắc đến TS. TRƯƠNG QUANG PHÚC, giảng viên hướng dẫn, người đã luôn tận tình hỗ trợ, định hướng và động viên nhóm trong suốt quá trình thực hiện đề tài. Kiến thức chuyên môn, kinh nghiệm thực tế và những góp ý quý báu của Thầy chính là hành trang quan trọng giúp chúng tôi hoàn thiện đồ án và phát triển bản thân trong tương lai.

Cuối cùng, mặc dù nhóm đã nỗ lực để hoàn thành tốt mục tiêu của đề tài, song do hạn chế về thời gian và kinh nghiệm, chắc chắn không thể tránh khỏi những thiếu sót. Chúng tôi kính mong quý Thầy/Cô và các bạn sinh viên thông cảm, đồng thời rất mong nhận được những góp ý chân thành để nhóm có thể tiếp tục hoàn thiện và nâng cao chất lượng đề tài hơn nữa.

Chúng tôi xin chân thành cảm ơn!

TÓM TẮT

Trong đề án này, hệ thống cân thông minh đã được thiết kế và triển khai nhằm hỗ trợ hoạt động quản lý kho cho doanh nghiệp sản xuất, đặc biệt trong việc xử lý các vật tư nhỏ và không đếm được bằng tay. Hệ thống được xây dựng gồm khối phần cứng và khối phần mềm, trong đó phần cứng bao gồm ESP32-CAM để quét mã QR, STM32 để đọc giá trị từ cân Loadcell, và LCD hiển thị thông tin. Dữ liệu sẽ được chuyển tới backend Django qua HTTP request để xử lý, ghi nhận lịch sử giao dịch, và tự động quy đổi khối lượng sang số lượng dựa theo định mức, sau đó ghi nhận vào cơ sở dữ liệu PostgreSQL. Kết quả đã cho thấy hệ thống hoạt động ổn định, giao diện dễ sử dụng, đáp ứng được nhu cầu quản lý kho nhanh chóng và chính xác.

DANH MỤC HÌNH

Hình 2.1 Mô hình hoạt động chính của Django	5
Hình 2.2 Module ESP32 CAM	10
Hình 2.3 Module STM32F103C8T6.....	12
Hình 2.4 Mạch chuyển đổi ADC 24bit Loadcell HX711.....	12
Hình 2.5. Màn hình LCD 2004	13
Hình 2.6. Nút nhấn 6x6mm.....	14
Hình 2.7 Mạch nguồn DC 3V3/5V	14
Hình 3.1 Sơ đồ khối hệ thống.....	18
Hình 3.2 Sơ đồ nguyên lí của khối phần cứng trong hệ thống.....	20
Hình 3.3 Lưu đồ hoạt động của phần cứng	21
Hình 3.4 Thiết kế cơ sở dữ liệu	22
Hình 3.5 Lưu đồ quản lí đơn vị.....	24
Hình 3.6 Lưu đồ quản lí nguyên vật liệu.....	25
Hình 3.7 Lưu đồ quản lí hệ số quy đổi đơn vị cho từng loại nguyên vật liệu	26
Hình 3.8 Lưu đồ nhập xuất tồn kho	27
Hình 3.9 Lưu đồ tìm kiếm lịch sử nhập xuất.....	28
Hình 3.10 Mặt sau PCB.....	29
Hình 3.11 Trạng thái cân trước khi nhấn nút nhập/xuất	29
Hình 3.12 Trạng thái cân sau khi nhấn nút nhập/xuất.....	30
Hình 3.13 Chức năng quản lí nguyên vật liệu trong hệ thống	31
Hình 3.14 Chức năng quản lí đơn vị của hệ thống.....	32
Hình 3.15 Chức năng quy đổi đơn vị cho từng loại nguyên vật liệu	32
Hình 3.16 Chức năng tìm kiếm lịch sử nhập xuất trong hệ thống	33
Hình 3.17 Chức năng nhập kho trong hệ thống	34
Hình 3.18 Chức năng xuất kho trong hệ thống	34
Hình 3.19 Chức năng tồn kho trong hệ thống	34

DANH MỤC BẢNG

Bảng 1 Một số tính năng của ESP32.....	11
Bảng 2 Một số tính năng của STM32F103C8T6	11
Bảng 3. Thông số của màn hình LCD 2004.....	13
Bảng 4. Giá trị điện áp của các linh kiện	14

CÁC TỪ VIẾT TẮT

Từ viết tắt	Tiếng Anh đầy đủ	Giải nghĩa tiếng Việt
IoT	Internet of Things	Mạng vạn vật kết nối
LCD	Liquid Crystal Display	Màn hình tinh thể lỏng
API	Application Programming Interface	Giao diện lập trình ứng dụng
HTTP	Hypertext Transfer Protocol	Giao thức truyền siêu văn bản
UART	Universal Asynchronous Receiver Transmitter	Giao tiếp không đồng bộ

CHƯƠNG 1

TỔNG QUAN

1.1 ĐẶT VẤN ĐỀ

Quản lý kho hàng (Warehouse Management) là tập hợp các hoạt động liên quan đến việc tổ chức, giám sát, lưu trữ, và luân chuyển hàng hóa trong kho, từ lúc nhập hàng cho đến khi xuất kho và kiểm kê. Đây là một mắt xích quan trọng trong chuỗi cung ứng của doanh nghiệp, đòi hỏi sự chính xác, liên tục và tối ưu trong toàn bộ quá trình. Việc quản lý kho hiệu quả không chỉ giúp tiết kiệm chi phí lưu trữ và hạn chế thất thoát, mà còn góp phần nâng cao năng suất và khả năng cạnh tranh của doanh nghiệp trong dài hạn [1].

Trong thời đại chuyển đổi số, nhiều doanh nghiệp đang từng bước ứng dụng các công nghệ thông minh vào quản lý kho, như mã QR, RFID, IoT, và hệ thống ERP. Tuy nhiên, đối với các doanh nghiệp sản xuất, đặc biệt là những ngành nghề đòi hỏi sử dụng số lượng lớn vật tư nhỏ như ốc vít, bu lông, tán rút, đệm... việc kiểm đếm và cập nhật tồn kho lại là một thách thức lớn. Những vật tư này thường không thể đếm thủ công chính xác do số lượng lớn và kích thước nhỏ, do đó cần được quy đổi sang đơn vị khối lượng (gram hoặc kilogram) dựa trên định mức kỹ thuật đã được xác lập trước.

Việc nhập liệu sai sót, nhập thiếu hoặc nhập dư vật tư có thể dẫn đến hậu quả nghiêm trọng: gây chậm trễ sản xuất, tăng chi phí vận hành, và làm mất tính chính xác trong kiểm kê. Trước thực trạng đó, nhu cầu đặt ra là phát triển một hệ thống cân thông minh tích hợp với phần mềm quản lý kho. Hệ thống này cần có khả năng:

- Nhận dạng vật tư bằng mã QR.

- Tự động quy đổi khối lượng đo được sang số lượng vật tư tương ứng theo bảng định mức.
- Cập nhật dữ liệu nhập – xuất vật tư chỉ bằng thao tác bấm nút, thay vì nhập liệu thủ công.

Việc kết hợp giữa cân điện tử, thiết bị đọc mã QR và phần mềm quản lý kho không chỉ giúp tự động hóa quy trình nhập – xuất – tồn, mà còn nâng cao độ chính xác, giảm thiểu sai sót, tiết kiệm thời gian và chi phí nhân sự. Đây là một bước tiến quan trọng trong việc hiện đại hóa công tác quản lý kho vật tư tại các doanh nghiệp sản xuất vừa và nhỏ.

1.2 TỔNG QUAN TÌNH HÌNH NGHIÊN CỨU

Trong bối cảnh cách mạng công nghiệp 4.0 đang lan rộng, việc ứng dụng công nghệ tự động hóa và các hệ thống thông minh vào quản lý kho ngày càng trở thành xu hướng tất yếu. Nhiều nghiên cứu và giải pháp đã được phát triển nhằm nâng cao độ chính xác, tốc độ xử lý và hiệu quả vận hành trong hoạt động lưu kho và luân chuyển vật tư.

Các hệ thống quản lý kho truyền thống chủ yếu dựa vào phần mềm ERP/ hoặc Excel kết hợp với thao tác thủ công của nhân viên. Tuy nhiên, mô hình này thường tiềm ẩn rủi ro sai sót do quá trình nhập liệu, đặc biệt trong môi trường có khối lượng vật tư lớn hoặc đa dạng về chủng loại. Để khắc phục, nhiều giải pháp công nghệ đã được nghiên cứu và triển khai:

- Ứng dụng Barcode/ QR Code giúp định danh nhanh vật tư, rút ngắn thời gian tìm kiếm và kiểm kê.

- Hệ thống RFID (Radio Frequency Identification) mang lại khả năng nhận dạng không tiếp xúc và tự động hóa kiểm kê. Tuy nhiên, giá thành của thẻ RFID và hệ thống đọc cao hơn so với mã vạch và QR code/Bar code khiến RFID chưa phù hợp với các doanh nghiệp vừa và nhỏ tại Việt Nam.

- Cân điện tử kết hợp vi điều khiển và cảm biến đang được ứng dụng trong các dây chuyền sản xuất để kiểm tra trọng lượng sản phẩm. Một số hệ thống cân công nghiệp hiện nay cho phép truyền dữ liệu qua cổng RS232 hoặc kết nối Wi-

Fi, tuy nhiên phần lớn chỉ dừng lại ở việc hiển thị số liệu, chưa tích hợp trực tiếp với phần mềm quản lý kho.

1.3 MỤC TIÊU ĐỀ TÀI

Đề tài hướng đến xây dựng một hệ thống cân thông minh tích hợp với phần mềm quản lý kho, với các mục tiêu cụ thể như sau:

Về phần cứng:

- Quét mã QR để nhận dạng sản phẩm.
- Cung cấp hai nút chức năng: "Nhập kho" và "Xuất kho".
- Tự động quy đổi khối lượng vật tư sang số lượng đơn vị theo bảng định mức.
- Hiển thị thông tin cân lên màn hình LCD.

Về phần mềm:

- Quản lý danh sách vật tư và các đơn vị đo tương ứng.
- Hỗ trợ khai báo định mức cho từng loại nguyên vật liệu.
- Ghi nhận chính xác lịch sử nhập – xuất – tồn kho theo loại, tên và ngày.
- Tự động hóa quá trình cập nhật kho mà không cần nhập liệu thủ công.

1.4 Ý NGHĨA ĐỀ TÀI

Đề tài mang lại ý nghĩa thiết thực trong việc ứng dụng công nghệ vào hoạt động sản xuất và quản lý kho. Giải pháp giúp:

- Giảm thiểu tối đa sai sót do thao tác thủ công.
- Tăng độ chính xác và tốc độ trong việc nhập – xuất vật tư.
- Tối ưu hoá quy trình quản lý kho, đặc biệt với các vật tư nhỏ không thể đếm từng cái.
- Góp phần nâng cao năng lực sản xuất và hiệu quả vận hành của doanh nghiệp.

Ngoài ra, hệ thống có thể được mở rộng và tùy biến cho nhiều mô hình sản xuất khác nhau, từ doanh nghiệp nhỏ đến lớn.

1.5 PHẠM VI ĐỀ TÀI

Đề tài tập trung vào việc xây dựng hệ thống cân thông minh hỗ trợ nhập – xuất kho cho các vật tư dạng nhỏ (óc vít, bu lông...) dựa trên khối lượng. Phạm vi cụ thể gồm:

- Ứng dụng trong quản lý kho cho doanh nghiệp sản xuất.
- Thiết kế phần cứng bao gồm cảm biến load cell, nút nhấn nhập xuất, stm32, esp32cam, màn hình LCD
- Xây dựng phần mềm quản lý kho, và định mức quy đổi đơn vị vật tư.
- Tích hợp phần cứng và phần mềm thành hệ thống hoàn chỉnh.

CHƯƠNG 2

CƠ SỞ LÝ THUYẾT

2.1 CÁC CÔNG NGHỆ TRONG SOFTWARE

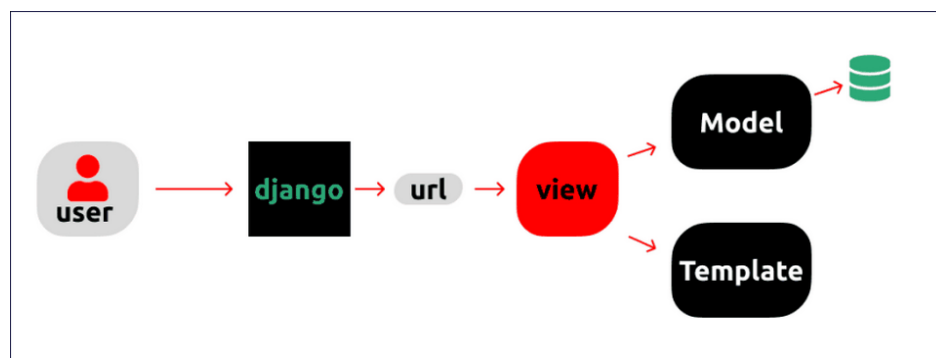
2.1.1 Django backend trong python

Django là Framework phát triển ứng dụng web được xây dựng trên ngôn ngữ lập trình Python. Được phát triển bởi Django Software Foundation và phát hành dưới giấy phép mã nguồn mở, Django cung cấp cơ sở tiếp cận hiệu quả để xây dựng các ứng dụng web có hiệu suất cao, dễ bảo trì [2].

Django là một web framework mã nguồn mở nổi bật trong hệ sinh thái Python, được thiết kế nhằm giúp lập trình viên phát triển ứng dụng web một cách nhanh chóng, an toàn và có khả năng mở rộng tốt. Django hoạt động theo mô hình MTV (Model - Template - View), tương đương với mô hình MVC truyền thống:

- Model: định nghĩa cấu trúc cơ sở dữ liệu bằng Python, Django tự động sinh các truy vấn SQL.
- Template: tạo giao diện HTML động, có thể kết hợp với HTMX hoặc JavaScript để tăng tương tác.
- View: xử lý logic nghiệp vụ và tương tác với Model để trả về nội dung cho trình duyệt.

Django được sử dụng rộng rãi trên toàn thế giới. Một số dự án tiêu biểu áp dụng nền tảng này chính là: Instagram, Spotify, Dropbox,... [2].



Hình 2.1 Mô hình hoạt động chính của Django

Ứng dụng của Django trong hệ thống cân thông minh:

- Trong hệ thống này, Django đóng vai trò là trung tâm xử lý và quản lý dữ liệu. Các chức năng cụ thể mà Django đảm nhiệm gồm:

- Tiếp nhận dữ liệu từ thiết bị ngoại vi (ESP32-CAM): Django cung cấp các endpoint Django REST framework API để thiết bị gửi dữ liệu mã QR, khối lượng, trạng thái nhập/xuất và lưu vào CSDL.

- Xử lý nghiệp vụ kho: xác định loại vật tư từ mã QR, quy đổi khối lượng sang số lượng dựa vào định mức.

- Quản lý cơ sở dữ liệu vật tư: sử dụng ORM (Object Relational Mapping) của Django để quản lý bảng định mức, bảng sản phẩm, bảng lịch sử nhập – xuất.

- Hiển thị giao diện người dùng: sử dụng Django Template và HTMX để hiển thị dữ liệu động như bảng vật tư, biểu đồ nhập – xuất, thông tin sản phẩm theo thời gian thực.

2.1.2 HTMX

HTMX là một thư viện JavaScript mã nguồn mở được phát triển bởi Carson Gross, người sáng lập dự án Big Sky Software. HTMX được ra mắt lần đầu vào năm 2020, là phần kế thừa và mở rộng từ một dự án trước đó có tên là Intercooler.js. Mục tiêu của HTMX là "đưa lại sức mạnh cho HTML" bằng cách cho phép các phần tử HTML gốc (như `<button>`, `<a>`, `<form>`) thực hiện các yêu cầu HTTP hiện đại (GET, POST, PUT, DELETE) mà không cần viết JavaScript tùy chỉnh bằng cách thêm các thuộc tính như `hx-get`, `hx-post`, `hx-target`, `hx-trigger` trực tiếp vào HTML, giúp đơn giản hóa logic giao diện phía client.

HTMX là một phần trong triết lý "hypermedia as the engine of application state" (HATEOAS) của REST, nhấn mạnh rằng HTML có thể trở thành giao diện động nếu được mở rộng đúng cách. Nó phản ánh sự quay lại với phương pháp server-side-render – nơi mà máy chủ sinh ra giao diện động và trình duyệt chỉ đóng vai trò hiển thị và tương tác đơn giản [3].

Việc chọn HTMX thay vì dùng AJAX thuần hoặc framework frontend (như React) có các lý do sau:

- Đơn giản hóa lập trình: Theo Dennis Ivy (2024) [4], HTMX giúp "đưa lại sức mạnh cho template engine" bằng cách cho phép cập nhật DOM trực tiếp từ server mà không cần viết nhiều logic JavaScript phức tạp.

- Kết hợp tự nhiên với Django Template: HTMX hoạt động rất mượt với Django vì cả hai đều thiên về hướng server-rendered HTML. Điều này giúp tận dụng tốt hệ thống template có sẵn mà không cần dùng frontend framework riêng.

- Tăng hiệu suất và UX: Chỉ tải phần cần thiết (partial HTML) giúp giao diện nhanh, mượt, không cần reload lại toàn bộ trang web hoặc viết JS phức tạp.

2.1.3 PostgreSQL

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở (Object-Relational Database Management System - ORDBMS) mạnh mẽ, được phát triển lần đầu tiên vào năm 1986 tại Đại học California, Berkeley, dưới tên gọi POSTGRES, và sau này chính thức đổi tên thành PostgreSQL vào năm 1996 [5].

PostgreSQL nổi bật với việc tuân thủ chặt chẽ các chuẩn SQL, hỗ trợ mạnh mẽ cho cả dữ liệu quan hệ (structured) và dữ liệu phi quan hệ (semi-structured) như JSON, XML. Đây là hệ quản trị dữ liệu được sử dụng phổ biến trong các ứng dụng quy mô vừa và lớn, được các công ty lớn như Apple, Instagram, Red Hat và Spotify sử dụng.

Lí do chọn PostgreSQL thay vì SQLite mặc định trong Django:

Mặc dù SQLite là một cơ sở dữ liệu nhúng gọn nhẹ, dễ triển khai và phù hợp với ứng dụng đơn giản, nhưng nó không hỗ trợ tốt các yêu cầu mở rộng và xử lý dữ liệu phức tạp. Cụ thể, đề tài yêu cầu khả năng:

- Tìm kiếm văn bản tiếng Việt có dấu (ví dụ: “óc vít”, “tán rút”) theo kiểu không phân biệt dấu hoặc phân biệt có tùy chọn. SQLite không hỗ trợ extension chuẩn nào để xử lý chuẩn Unicode với tiếng Việt, trong khi PostgreSQL cho phép sử dụng extension như unaccent, hoặc cấu hình collate vi_VN để tìm kiếm văn bản gần đúng và có hỗ trợ loại bỏ dấu tiếng Việt.

- Tối ưu hóa hiệu suất và mở rộng: PostgreSQL quản lý tốt dữ liệu lớn, giúp tăng tốc độ tìm kiếm trong các bảng lớn - điều cần thiết khi hệ thống lưu nhiều bản ghi lịch sử nhập - xuất.

Do đó, lựa chọn PostgreSQL không chỉ giúp đảm bảo hiệu suất mà còn đáp ứng tốt yêu cầu đặc thù về tìm kiếm tiếng Việt có dấu, hỗ trợ mở rộng và tích hợp chặt chẽ với Django thông qua hệ ORM.

2.1.4 DaisyUI

DaisyUI là một plugin của Tailwind CSS, cung cấp sẵn hàng trăm thành phần giao diện được thiết kế theo phong cách hiện đại, dễ dùng, và có khả năng tùy biến cao. Được phát triển nhằm đơn giản hóa việc xây dựng giao diện với Tailwind, DaisyUI kết hợp sự tiện lợi của các component UI có sẵn (như nút, thẻ, modal, navbar, v.v.) với tính linh hoạt mạnh mẽ của Tailwind.

Đặc điểm nổi bật của DaisyUI

- Component có sẵn: Gồm hàng trăm thành phần phổ biến như btn, card, modal, alert, giúp lập trình viên không cần xây dựng từ đầu.
- Tùy biến dễ dàng: Mỗi component đều có thể tùy chỉnh màu sắc, kích thước, chủ đề (theme), và hành vi mà không cần viết thêm CSS.
- Hỗ trợ theme động: DaisyUI hỗ trợ đổi nhiều giao diện (dark/light/theme brand) chỉ bằng vài class, rất tiện cho frontend hiện đại.
- Thân thiện với developer Tailwind: Người đã quen với Tailwind CSS sẽ thấy DaisyUI cực kỳ linh hoạt, không bị giới hạn như Bootstrap.

2.2 CÁC CHUẨN TRUYỀN THÔNG

2.2.1 Giao thức UART

UART (viết tắt của Universal Asynchronous Receiver Transmitter, tạm dịch: Bộ thu phát không đồng bộ đa năng) là một chuẩn giao tiếp nối tiếp được sử dụng rộng rãi để truyền dữ liệu giữa các thiết bị điện tử. Nó được sử dụng phổ biến trong các hệ thống nhúng, máy tính và các thiết bị ngoại vi.

Đặc điểm của UART:

- Giao tiếp không đồng bộ: Dữ liệu được truyền vào theo từng bit, nhưng không cần tín hiệu xung nhịp để đồng bộ hóa. Thay vào đó, mỗi bit dữ liệu được đánh dấu bằng một bit bắt đầu và một bit dừng.

- Chế độ song công bán phần: Dữ liệu chỉ có thể được truyền một chiều tại một thời điểm, nhưng có thể thay đổi hướng truyền theo yêu cầu.
- Sử dụng ít chân tín hiệu: UART chỉ sử dụng hai chân tín hiệu: TX (Transmit) và RX (Receive).
- Dễ dàng triển khai: UART có cấu trúc đơn giản và dễ dàng thực hiện phần mềm điều khiển.

Cấu trúc chân tín hiệu UART:

- TX (Transmit): Chân mà thiết bị sử dụng để gửi dữ liệu đến thiết bị khác.
- RX (Receive): Chân mà thiết bị sử dụng để nhận dữ liệu từ thiết bị khác.

2.2.2 Giao thức I²C

- Chỉ sử dụng 2 đường bus: SDA (viết tắt của “Serial Data Line”, tạm dịch là đường truyền dữ liệu nối tiếp) và SCL (viết tắt của “Serial Clock Line, tạm dịch là đường xung nhịp nối tiếp [6].

Cấu trúc chân tín hiệu của I²C :

- Chân SDA (Serial Data Line): Dây dữ liệu, truyền và nhận thông tin hai chiều giữa master và slave
- Chân SCL (Serial Clock Line): Dây xung nhịp do master tạo ra để đồng bộ việc truyền dữ liệu.

2.2.3 Giao thức HTTP

Giao thức HTTP(Tiếng anh là: HyperText Transfer Protocol – Giao thức truyền tải siêu văn bản) là một giao thức lớp ứng dụng nằm trong bộ giao thức dành cho hệ thống thông tin siêu phương tiện phân tán, cộng tác. Nó chính là nền tảng dùng để trao đổi và liên lạc dữ liệu với World Wide Web, nơi các tập tin tài liệu siêu văn bản có thể chứa các siêu liên kết dẫn đến các tài nguyên số khác mà người dùng ngón tay chạm vào màn hình cảm ứng lúc duyệt web. Nhờ đó, HTTP cho phép người sử dụng truy cập và tải về các tài nguyên như văn bản HTML, text, video, ảnh,... của các trang web và hiển thị chúng lên trình duyệt.

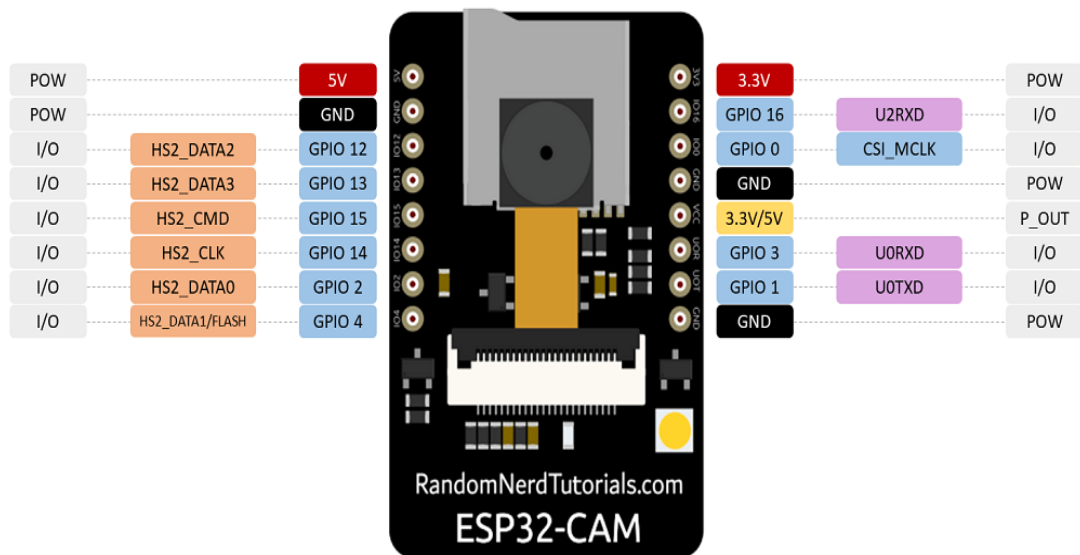
Tổng quan về mặt kỹ thuật của HTTP: HTTP hoạt động dựa trên giao thức yêu cầu-phản hồi nằm trong mô hình khách - chủ. *Khách* có thể là một trình duyệt được

một người truy cập thực hiện yêu cầu, còn *chủ* có thể là một máy chủ web trên đám mây hoặc lưu trữ cục bộ chạy trên một chiếc máy tính quản lý một hay nhiều trang web. Khách sẽ gửi một yêu cầu HTTP lên cho máy chủ. Còn máy chủ (nơi cung cấp các *tài nguyên* chẳng hạn như tập tin HTML và những nội dung, hoặc thực hiện những chức năng khác thay mặt cho khách) sẽ *phản hồi* lại thông điệp cho khách. Thông điệp được phản hồi sẽ chứa thông tin hoàn thành yêu cầu như thế nào, và đôi khi cũng sẽ chứa thông tin được yêu cầu trong phần nội dung của nó.

2.3 CÁC LINH KIỆN ĐƯỢC LỰA CHỌN

2.3.1 Vi điều khiển ESP32-CAM

ESP32-CAM là một module có tích hợp camera là một nhánh của dòng vi điều khiển ESP32. Với camera có thể nhận diện và chụp được hình ảnh, cùng với giá thành hợp lý, đây sẽ là module hợp lý cho các dự án Iot vừa và nhỏ.



Hình 2.2 Module ESP32 CAM

Một số tính năng của ESP32 được thể hiện qua Bảng 1

Bảng 1 Một số tính năng của ESP32

Vi xử lý	<ul style="list-style-type: none"> - CPU Xtensa ® 32-bit LX6 Dual-core (2 nhân), tiết kiệm điện năng. - Tốc độ xung nhịp: lên đến 160 MHz - 520 KB SRAM nội.
Kết nối	<ul style="list-style-type: none"> - Wifi 802.11 b/g/n 2.4 GHz, hỗ trợ Bluetooth BLE 4.2, - Hỗ trợ giao thức IEEE 802.11n (HT20), tốc độ dữ liệu lên đến 150 Mbps.
Ngoại vi	<ul style="list-style-type: none"> - Có 9 chân GPIO (tùy theo cấu hình). - Hỗ trợ nhiều chuẩn giao tiếp như UART, SPI, I2C, PWM, ADC,... - Camera OV2640.
Nguồn điện	<ul style="list-style-type: none"> - Điện áp hoạt động: 3.3- 5V. - Dòng điện tiêu thụ: <5µA (ở chế độ ngủ)

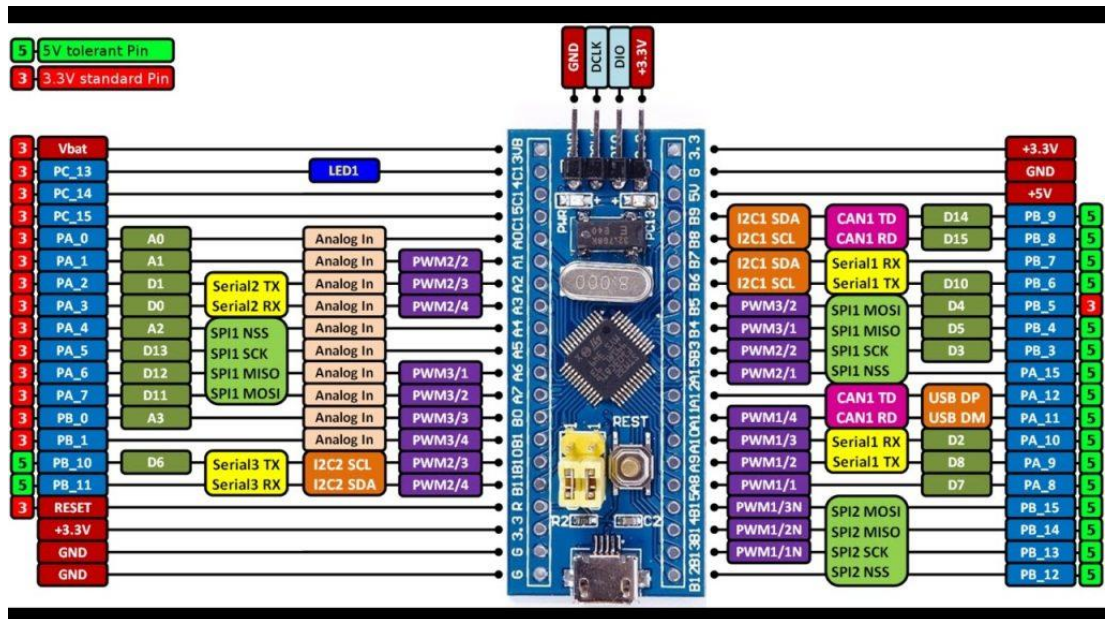
2.3.2 Vi điều khiển STM32F103C8T6

STM32F103C8T6 là một vi điều khiển 32-bit dựa trên kiến trúc AMR Cortex-M3 với FPU (Floating Point Unit - Đơn vị xử lý dấu chấm động), giúp cải thiện hiệu suất trong các ứng dụng tính toán phức tạp.

Một số tính năng của STM32 được thể hiện qua Bảng 2

Bảng 2 Một số tính năng của STM32F103C8T6

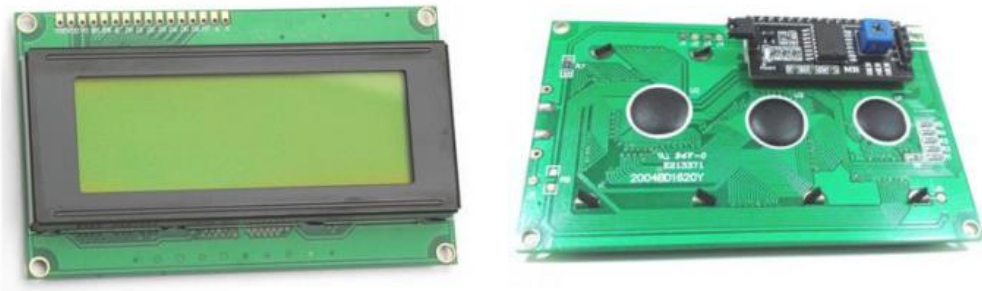
Vi xử lý	<ul style="list-style-type: none"> - Bộ xử lý ARM Cortex-M3 - Tốc độ xung nhịp: lên đến 72 MHz - Từ 512 KB đến 1MB bộ nhớ Flash. - 20 KB bộ nhớ SRAM
Ngoại vi	<ul style="list-style-type: none"> - Có 37 chân GPIO (tùy theo cấu hình). - Hỗ trợ nhiều chuẩn giao tiếp như UART, SPI, I2C, PWM, ADC,...
Nguồn điện	<ul style="list-style-type: none"> - Điện áp hoạt động: 3.3 - 5V. - Dòng điện tiêu thụ: <5µA (ở chế độ ngủ)



- Các chân giao tiếp: DT, SCK, VCC, GND
- Dòng điện tiêu thụ: <1.5 mA
- Kích thước: 38 x 21 x 10 mm

2.3.4 Màn hình LCD, nút nhấn và nguồn

Màn hình LCD I²C 20x04 là loại hiển thị ký tự gồm 20 cột và 4 dòng, sử dụng giao tiếp I²C giúp đơn giản hóa kết nối và tiết kiệm chân trên vi điều khiển như STM32, ESP32



Hình 2.5. Màn hình LCD 2004

Một số đặc điểm của màn hình LCD được thể hiện qua bảng Bảng 3. Thông số của màn hình LCD 2004

Bảng 3. Thông số của màn hình LCD 2004

Thuộc tính	Thông tin
Kích thước hiển thị	20 ký tự, 4 dòng
Giao tiếp	I ² C
Số chân kết nối	4 chân (GND, VCC, SDA, SCL)
Điện áp hoạt động	5V ADC
Dòng tiêu thụ	~ 1.5 mA
IC điều khiển	HD44780
Độ tương phản	Có chiết áp (biến trở) để điều chỉnh
Đèn nền	Có (LED backlight, có thể tắt/bật)

Nút nhấn sử dụng ở đây là nút 6x6mm, là một linh kiện điện tử cơ bản dạng công tắc nhấn (push button), cho phép người dùng tương tác với mạch điện tử. Có giá thành rẻ, phù hợp với những dự án IoT.



Hình 2.6. Nút nhấn 6x6mm

Do tính chất cố định và không phải di chuyển nhiều của hệ thống nên ta không cần chú trọng nhiều về việc cung cấp nguồn cho ESP32 và STM32. Ở đây, ta có bảng giá trị mức điện áp của các linh kiện bên dưới, dựa vào đó ta cho thấy ta có thể sử dụng addapter 5V 2A để sạc một cách nhanh chóng và an toàn. Vì vậy nguồn điện sẽ được cung cấp đủ cho các linh kiện và không bị gián đoạn kết nối. Ngoài ra, còn sử dụng IC AMS1117 để ổn định và giảm áp xuống 3.3V để cấp nguồn cho các linh kiện ngoại vi.



Hình 2.7 Mạch nguồn DC 3V3/5V

Bảng 4. Giá trị điện áp của các linh kiện

Linh kiện	Mức điện áp	Dòng điện tiêu thụ
ESP32 CAM	3.3V - 5V	160mA - 310mA
STM32F103C8T6	3.3V - 5V	20mA - 150mA
Loadcell HX711	2.7V – 5V	< 1.5mA
Màn hình LCD	3.3V – 5V	~ 1.5mA

CHƯƠNG 3

THIẾT KẾ HỆ THỐNG

3.1 TỔNG QUAN

Giải pháp “Hệ thống cân thông minh sử dụng trong quản lý kho cho doanh nghiệp sản xuất” được nhóm đề xuất như một giải pháp toàn diện hai trong một, kết hợp chặt chẽ giữa thiết bị cân điện tử thông minh và hệ thống phần mềm quản lý kho chuyên dụng. Giải pháp này ra đời nhằm khắc phục những hạn chế của quy trình quản lý kho truyền thống, đặc biệt là trong các doanh nghiệp sản xuất thường xuyên làm việc với số lượng lớn vật tư nhỏ lẻ, khó kiểm đếm.

Hệ thống được thiết kế để khắc phục các nhược điểm của phương pháp quản lý kho truyền thống như nhập liệu thủ công tốn thời gian, dễ sai sót và những phức tạp trong việc quy đổi đơn vị tính. Với giải pháp cân thông minh, quy trình được tự động hóa: người dùng chỉ cần quét mã QR của vật tư, đặt lên cân, và nhấn nút “Nhập” hoặc “Xuất”. Hệ thống sẽ tự động ghi nhận khối lượng, quy đổi về đơn vị cơ bản và cập nhật tồn kho, giúp quản lý vật tư hiệu quả và giảm thiểu sai sót.

Để hiện thực hóa giải pháp này, nhóm báo cáo đã phát triển hai khối thành phần chính:

- **Khối phần cứng:** Đây là thiết bị đầu cuối thực hiện việc thu thập dữ liệu tại kho.
 - + Chức năng: Bao gồm quét mã QR để nhận dạng vật tư (sử dụng ESP32 CAM), đo lường khối lượng chính xác (qua load cell và vi điều khiển STM32), hiển thị thông tin trực quan cho người dùng qua màn hình LCD.
 - + Tương tác: Thông qua các nút nhấn, người dùng thực hiện lệnh nhập/xuất. Dữ liệu giao dịch sau đó được gửi qua Wi-Fi đến API của khối phần mềm để xử lý.
- **Khối phần mềm:** Đây là trung tâm xử lý, lưu trữ dữ liệu và cung cấp giao diện quản lý.

+ Chức năng: Bao gồm việc quản lý chi tiết danh mục vật tư, khai báo các loại đơn vị tính và thiết lập hệ số quy đổi linh hoạt cho từng vật tư. Hệ thống tự động cập nhật tồn kho theo thời gian thực và ghi lại toàn bộ lịch sử giao dịch.

+ Giao diện và API: Cung cấp giao diện web trực quan cho người quản lý kho truy cập, theo dõi và thực hiện các tác vụ quản lý. Đồng thời, hệ thống xây dựng các API để khối phần cứng có thể giao tiếp, gửi dữ liệu và nhận phản hồi.

3.1.1 Yêu cầu hệ thống

Để đáp ứng yêu cầu hoạt động hiệu quả và chính xác trong điều kiện vận hành liên tục tại các doanh nghiệp sản xuất, hệ thống “Cân thông minh” phải đảm bảo các yêu cầu thiết kế sau:

- Nhận dạng vật tư nhanh chóng và tự động: Thông qua việc quét mã QR (sử dụng ESP32 CAM), thiết bị phải có khả năng nhận diện nhanh chóng ID và trả về tên vật tư, đồng thời hiện lên LCD.

- Gửi dữ liệu nhập xuất đến khối phần mềm: Sau mỗi thao tác cân và xác nhận (nhấn nút “Nhập kho” hoặc “Xuất kho”), thiết bị cân phải gửi dữ liệu giao dịch (bao gồm ID vật tư, khối lượng, ID đơn vị, loại nhập/ xuất) lên hệ thống phần mềm quản lý kho thông qua kết nối Wi-Fi. Điều này đảm bảo dữ liệu tồn kho luôn được cập nhật mới nhất.

- Giao diện hiển thị: Màn hình LCD tích hợp trên cân phải hiển thị các thông tin quan trọng cho người vận hành như tên vật tư (sau khi quét QR), khối lượng đang cân. Và sau khi nhấn nút nhập/ xuất, hiện thông tin JSON trả về lên màn hình bao gồm thông tin

“Nhập/Xuất ‘tên_nguyên_vật_liệu’”

“khối_lượng g - số_lượng đơn_vị_cơ_bản”

“thời_gian_ngày_nhập_xuất”

“Tồn_kho: số_lượng đơn_vị_cơ_bản”

- Nguồn cung cấp ổn định: Thiết bị cân cần được thiết kế với nguồn điện ổn định, có thể hoạt động liên tục trong môi trường nhà xưởng.

Bên cạnh đó, ở khối phần mềm (bao gồm cơ sở dữ liệu, xử lý backend và giao diện người dùng) cũng phải đáp ứng một số yêu cầu:

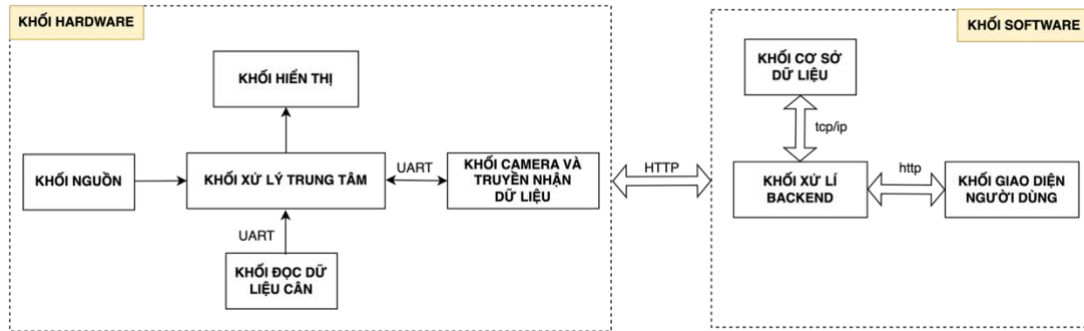
- Quản lý đơn vị: Module này cho phép người quản trị định nghĩa và duy trì một danh mục chuẩn hóa các đơn vị tính (ví dụ: kg, gram, cái, hộp, lít) sẽ được sử dụng thống nhất trong toàn bộ quá trình quản lý vật tư. Người dùng có thể dễ dàng thêm mới các đơn vị này hoặc xem lại danh sách các đơn vị đang được áp dụng.

- Quản lý Nguyên vật liệu cho phép quản lý thông tin chi tiết của từng loại vật tư, trong đó mỗi vật tư sẽ được gán một đơn vị cơ bản dùng làm chuẩn để theo dõi tồn kho. Điểm nổi bật của module này là khả năng thiết lập linh hoạt các đơn vị quy đổi khác nhau cho mỗi nguyên vật liệu, ví dụ như quy định rằng 1 cái ốc vít sẽ tương đương với 0.05g. Hơn nữa, khi một vật tư mới được tạo, hệ thống sẽ tự động tạo một bản ghi quy đổi cho chính đơn vị cơ bản đó với hệ số bằng 1, đảm bảo tính toàn vẹn và nhất quán của dữ liệu quy đổi.

- Quản lý Nhập/Xuất/Tồn kho: là thành phần trung tâm, theo dõi mọi biến động về số lượng vật tư. Hệ thống được thiết kế để tiếp nhận dữ liệu giao dịch nhập hoặc xuất trực tiếp từ thiết bị cân thông minh thông qua API, đồng thời cũng hỗ trợ người dùng thực hiện các giao dịch này một cách thủ công qua giao diện web. Khi một giao dịch được ghi nhận, phần mềm sẽ tự động áp dụng các hệ số quy đổi đã được thiết lập để tính toán ra số lượng tương ứng theo đơn vị cơ bản của vật tư (`base_quantity`). Dựa trên giá trị này, số lượng tồn kho thực tế (`inventory_level`) của vật tư liên quan sẽ được cập nhật ngay lập tức, tăng lên khi nhập và giảm đi khi xuất. Thông tin tồn kho được hiển thị một cách trực quan và có thể được xem dưới nhiều đơn vị quy đổi khác nhau, tùy theo nhu cầu của người dùng.

- Lịch sử Nhập Xuất :với khả năng tìm kiếm và lọc dữ liệu mạnh mẽ. Tất cả các giao dịch, sau khi được xử lý, đều được lưu trữ chi tiết và đầy đủ vào cơ sở dữ liệu, bao gồm các thông tin như loại vật tư, số lượng, đơn vị sử dụng trong giao dịch, loại giao dịch (nhập hay xuất), và thời gian chính xác thực hiện. Giao diện người dùng cho phép xem lại toàn bộ lịch sử giao dịch này và cung cấp các công cụ lọc tiện lợi, giúp người dùng dễ dàng truy vấn các giao dịch theo tên hoặc mã vật tư, theo loại giao dịch, hoặc theo một khoảng thời gian cụ thể.

3.1.2 Sơ đồ khối hệ thống



Hình 3.1 Sơ đồ khối hệ thống

Chức năng của khối Hardware:

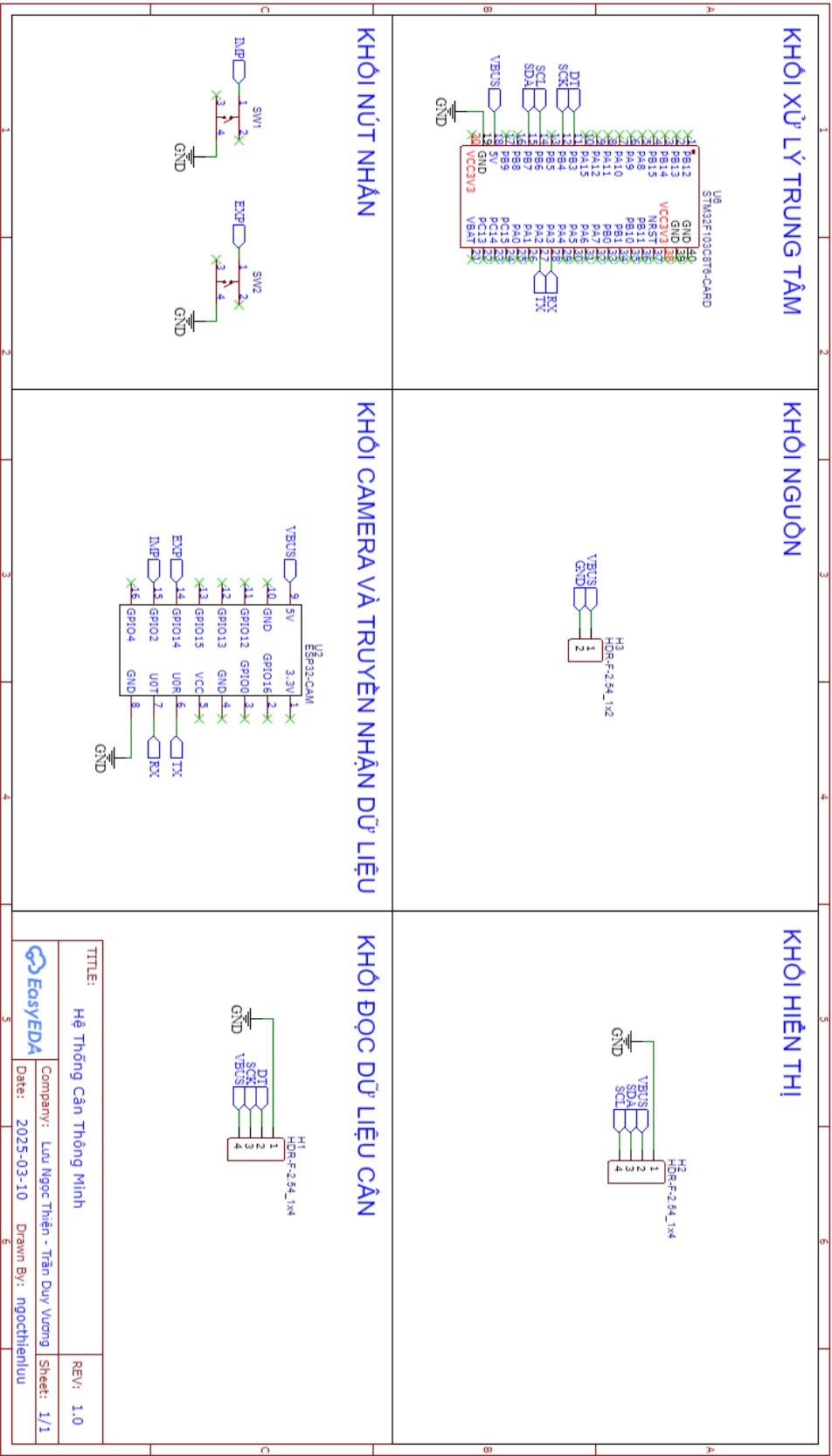
- Khối xử lý trung tâm: Nhận dữ liệu từ cân và hiện lên LCD, hiện thông tin vật tư lên LCD sau khi quét QR, đồng thời truyền số liệu cân nặng tới khối camera và truyền nhận dữ liệu. Nặng json trả về khối truyền nhận dữ liệu để hiện thông tin lên LCD.
- Khối đọc dữ liệu cân: Sử dụng module cân HX711 để chuyển đổi tín hiệu tương tự sang tín hiệu số 24bit.
- Khối hiển thị: Hiện thị khối lượng cân nặng hiện tại, thông tin vật tư, thông báo lịch sử nhập xuất của các lần cân.
- Khối camera và truyền nhận dữ liệu: Đọc QR Code, và giao tiếp với khối Software.
- Khối nguồn: Cung cấp nguồn điện 3.3 - 5V cho khối xử lý trung tâm.

Chức năng của khối Software:

- Khối xử lý backend: Sử dụng Django để tạo API xử lý dữ liệu nhận được từ khối Hardware, kết nối với cơ sở dữ liệu SQL bằng cơ chế ORM (Object-Relational Mapping). Đồng thời giúp server-side rendering ra khối giao diện người dùng
- Khối cơ sở dữ liệu: Dùng cơ sở dữ liệu quan hệ, để quản lý danh sách nguyên vật liệu, danh sách đơn vị, bảng quy đổi đơn vị, và lưu trữ dữ liệu nhập xuất nguyên vật liệu từ khối giao diện người dùng hoặc khối phần cứng.
- Khối giao diện người dùng: Giao diện web để người dùng quản lý khối cơ sở dữ liệu. Cho phép thao tác với bảng danh sách nguyên vật liệu, bảng

danh sách đơn vị, bảng quy đổi đơn vị, bảng lịch sử nhập xuất. Đồng thời tìm kiếm các nguyên vật liệu dựa trên 3 tiêu chí loại nhập xuất, tên nguyên vật liệu, khoảng thời gian nhập xuất.

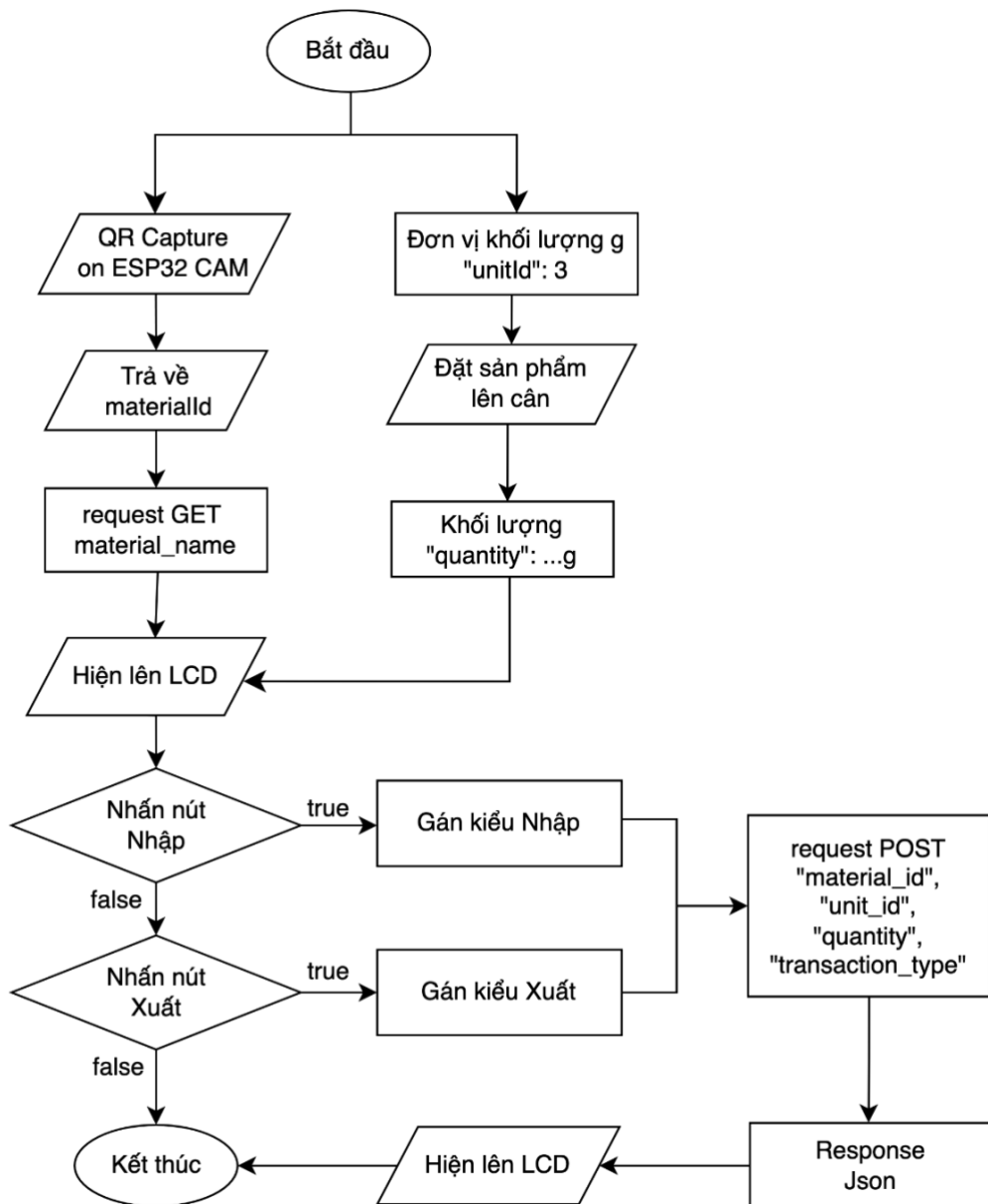
3.1.3 Sơ đồ nguyên lý



Hình 3.2 Sơ đồ nguyên lí của khối phần cứng trong hệ thống

3.2 THIẾT KẾ PHẦN CỨNG

3.2.1 Lưu đồ hoạt động của phần cứng



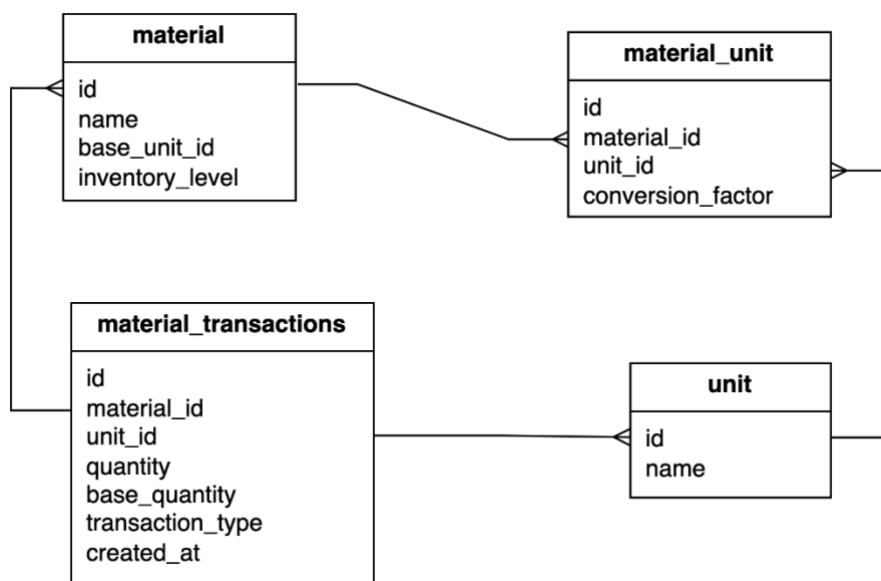
Hình 3.3 Lưu đồ hoạt động của phần cứng

Sau khi quét mã QR chứa ID sản phẩm, ESP32-CAM sẽ trích xuất ID và gửi yêu cầu HTTP-GET đến server để lấy thông tin tên sản phẩm, sau đó hiển thị tên lên màn hình LCD. Khi người dùng nhấn nút "Nhập" hoặc "Xuất", thiết bị sẽ gửi yêu cầu HTTP-POST đến server, kèm theo các thông tin: ID sản phẩm, ID đơn vị, khối lượng và loại giao dịch (nhập/xuất). Server sẽ phản hồi bằng dữ liệu JSON và nội dung này sẽ được hiển thị lên LCD.

3.3 THIẾT KẾ PHẦN MỀM

3.3.1 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu của hệ thống "Đồ án cân thông minh" được thiết kế để lưu trữ thông tin về nguyên vật liệu, đơn vị tính, các quy đổi đơn vị và lịch sử giao dịch nhập/xuất kho. Hệ thống sử dụng Django ORM (Object-Relational Mapper) để tương tác với cơ sở dữ liệu PostgreSQL, giúp ánh xạ các đối tượng Python (models) thành các bảng trong cơ sở dữ liệu một cách hiệu quả.



Hình 3.4 Thiết kế cơ sở dữ liệu

Bảng Unit (Đơn vị tính)

Lưu trữ thông tin về các đơn vị tính được sử dụng trong hệ thống (ví dụ: kg, gram, cái, hộp, thùng).

- id (Primary Key): ID tự động tăng theo số tự nhiên.
- name (VARCHAR(50)): Tên của đơn vị tính (ví dụ: “kg”, “cái”).

Bảng Material (Nguyên vật liệu)

Lưu trữ thông tin chi tiết về các loại nguyên vật liệu được quản lý trong kho.

- id (Primary Key): ID tự động tăng theo số tự nhiên.
- name (VARCHAR(255)): Tên của nguyên vật liệu
- base_unit_id (Foreign Key): Tham chiếu đến id của bảng

- inventory_level (FloatField): Số lượng tồn kho hiện tại của nguyên vật liệu, được tính theo base_unitId. Mặc định là 0.0.
- units (ManyToManyField): Tham chiếu đến các đơn vị tính khác mà nguyên vật liệu này có thể được quy đổi, thông qua bảng trung gian MaterialUnit.

Bảng MaterialUnit (Đơn vị quy đổi của Nguyên vật liệu)

Bảng trung gian này thiết lập mối quan hệ many-to-many giữa Material và Unit, đồng thời lưu trữ hệ số quy đổi giữa đơn vị cơ bản và các đơn vị khác của một nguyên vật liệu cụ thể. Ví dụ: 1 cái (đơn vị cơ bản) ốc vít = 0.5 gam (đơn vị quy đổi).

- id (Primary Key): ID tự động tăng theo số tự nhiên.
- Material_id (Foreign Key): Tham chiếu đến id của bảng Material.
- unitId (Foreign Key): Tham chiếu đến id của bảng Unit.
- conversion_factor (FloatField): Hệ số quy đổi cho Material và Unit.

Bảng MaterialTransactions (Giao dịch Nguyên vật liệu)

Ghi lại lịch sử các giao dịch nhập và xuất kho của từng nguyên vật liệu.

- id (Primary Key): ID tự động tăng theo số tự nhiên.
- material_id (Foreign Key): Tham chiếu đến id của bảng Material.
- unit_id (Foreign Key): Tham chiếu đến id của bảng Unit, xác định đơn vị tính được sử dụng trong giao dịch này.
- quantity (FloatField): Số lượng nguyên vật liệu trong giao dịch, tính theo unit_id.
- base_quantity (FloatField): Số lượng nguyên vật liệu trong giao dịch, đã được quy đổi về base_unit_id của Material. Trường này được tự động tính toán khi giao dịch được lưu.

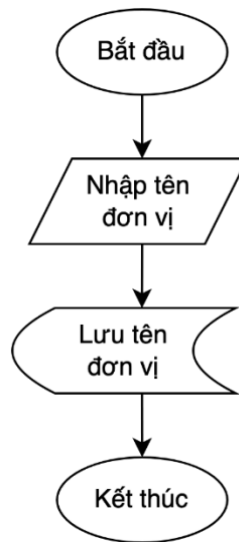
(Khi lưu: $\text{base_quantity} = \text{quantity} / \text{material_unit.conversion_factor}$)

(Khi đọc: $\text{quantity} = \text{base_quantity} * \text{material_unit.conversion_factor}$)

- transaction_type (CharField(10)): 'import' hoặc 'export'
- created_at (DateTimeField): Thời gian giao dịch được tạo, tự động ghi nhận thời điểm hiện tại khi bản ghi được thêm mới.

3.3.2 Lưu đồ quản lí đơn vị

Chức năng quản lý đơn vị cho phép người dùng định nghĩa các đơn vị tính (ví dụ: kg, cái, hộp) sẽ được sử dụng trong toàn bộ hệ thống để quản lý nguyên vật liệu và giao dịch. Lưu đồ dưới đây mô tả quy trình thêm một đơn vị mới vào hệ thống.



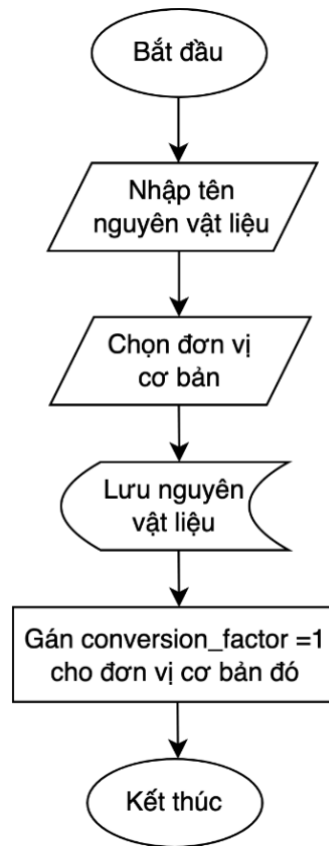
Hình 3.5 Lưu đồ quản lí đơn vị

Người dùng chọn chức năng “Khai báo đơn vị” từ menu “Quản lý” trên thanh điều hướng của ứng dụng.

Hệ thống điều hướng đến trang "Thêm đơn vị". Trang này được render bởi view `add_unit` trong `material_app/views.py` và sử dụng template `material_app/templates/units/add_unit.html`.

3.3.3 Lưu đồ quản lí nguyên vật liệu

Chức năng quản lý nguyên vật liệu là một thành phần cốt lõi của hệ thống “cân thông minh”, cho phép người dùng định nghĩa, theo dõi và quản lý danh mục các loại vật tư sử dụng trong kho. Các hoạt động chính bao gồm thêm mới nguyên vật liệu, xem danh sách, và thiết lập các đơn vị quy đổi cho từng loại nguyên vật liệu.



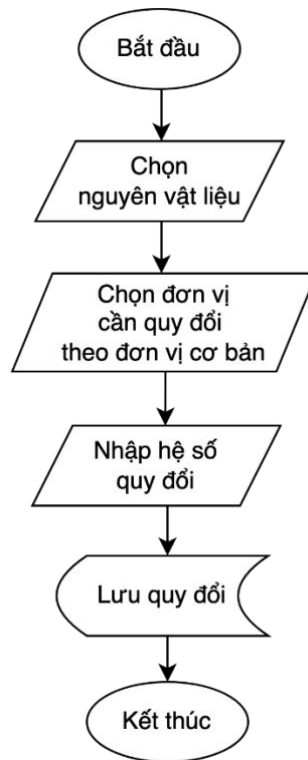
Hình 3.6 Lưu đồ quản lý nguyên vật liệu

Người dùng truy cập trang "Danh sách nguyên vật liệu". Trang này được render bởi view `material_list` và sử dụng template `material_app/templates/materials/material_list.html`.

Người dùng nhấn vào nút "Thêm nguyên vật liệu". Thao tác này sẽ kích hoạt một modal (cửa sổ nổi) để nhập thông tin. Modal này được định nghĩa trong `material_app/templates/materials/add_material.html` và được nhúng vào `material_list.html`.

3.3.4 Lưu đồ quản lý hệ số quy đổi đơn vị cho từng loại nguyên vật liệu

Sau khi một nguyên vật liệu đã được khai báo với một đơn vị cơ bản, người dùng có thể định nghĩa thêm các đơn vị quy đổi khác cho nguyên vật liệu đó. Ví dụ, nếu “Ốc vít” có đơn vị cơ bản là “cái”, người dùng có thể quy đổi sang “gam” hoặc “hộp”. Quy trình này rất quan trọng để hệ thống có thể tự động tính toán số lượng tồn kho và xử lý giao dịch một cách chính xác qua nhiều đơn vị khác nhau.



Hình 3.7 Lưu đồ quản lý hệ số quy đổi đơn vị cho từng loại nguyên vật liệu

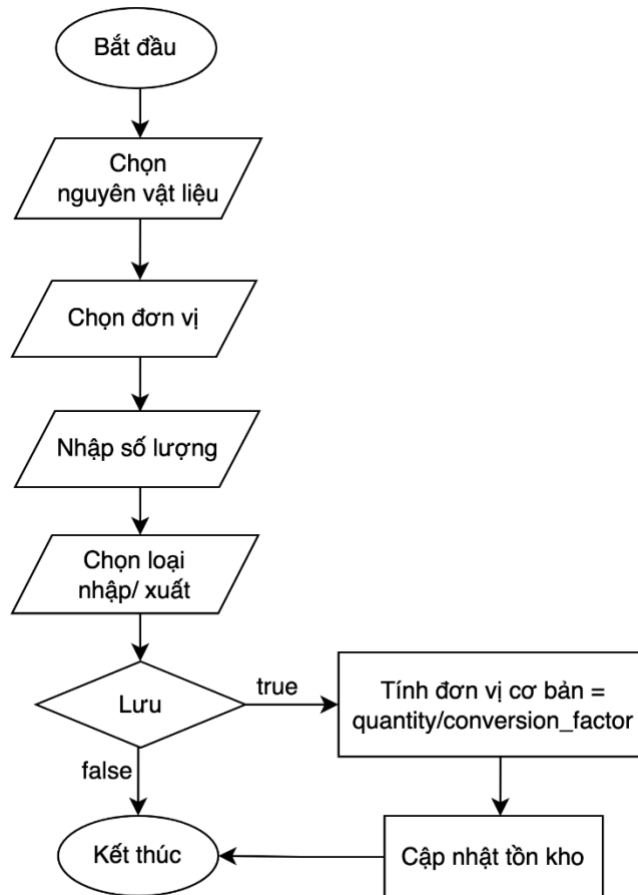
Từ trang "Danh sách nguyên vật liệu" (`material_app/templates/materials/material_list.html`), người dùng chọn một nguyên vật liệu cụ thể và nhấn vào nút "Quy đổi đơn vị" tương ứng với nguyên vật liệu đó.

Trang này được quản lý bởi view `manage_units(request, material_id)` trong `material_app/views.py` và sử dụng template `material_app/templates/materials/manage_units.html`. `material_id` của nguyên vật liệu được chọn sẽ được truyền qua URL.

3.3.5 Lưu đồ nhập xuất tồn kho

Chức năng nhập và xuất kho cho phép người dùng ghi nhận các giao dịch thay đổi số lượng tồn kho của nguyên vật liệu một cách thủ công qua giao diện web. Điều này rất quan trọng để duy trì tính chính xác của dữ liệu tồn kho, bên cạnh các giao dịch tự động từ cân thông minh.

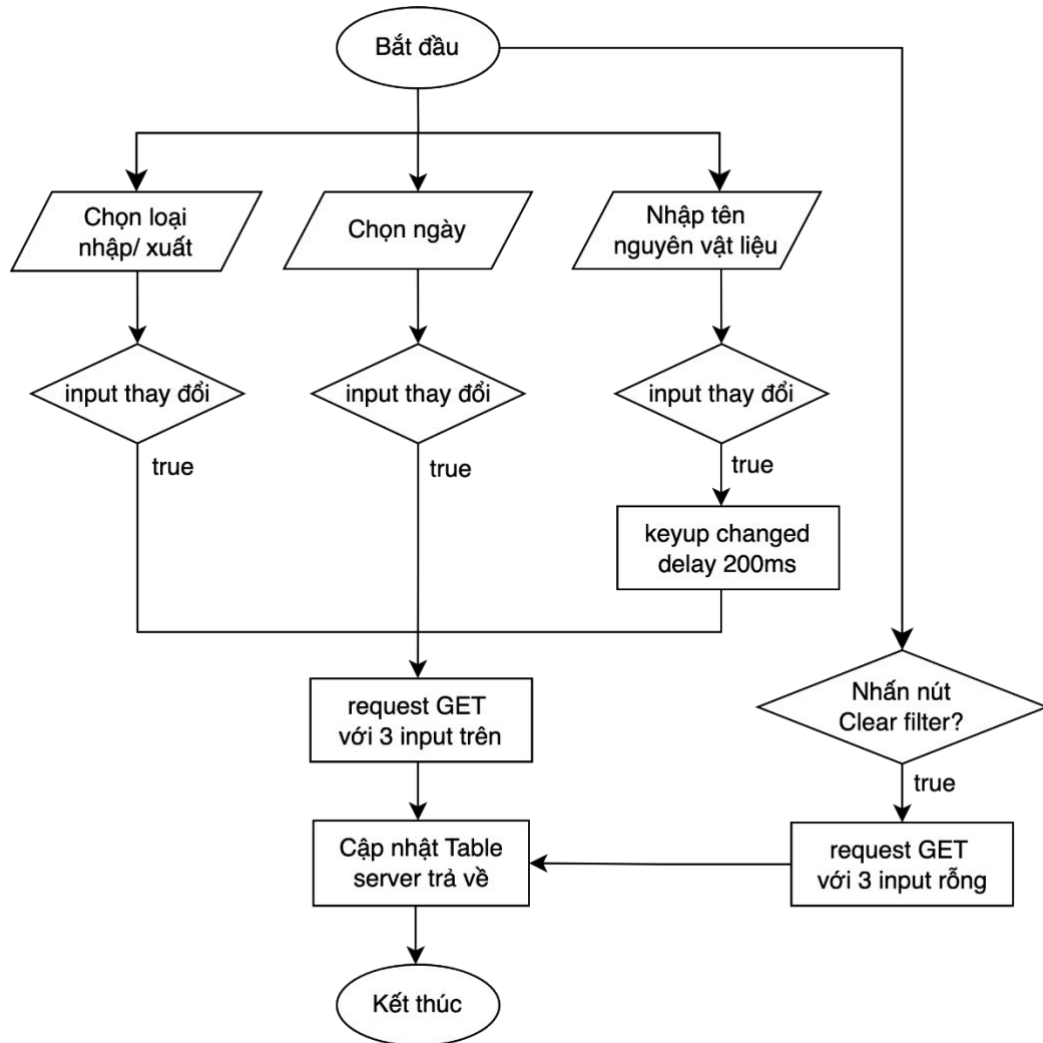
Điểm lưu ý sau khi lưu nhập xuất tồn thì django sẽ tự động tính base_quantity dựa trên conversion_factor. Và cập nhật tồn kho của material_id đó trong bảng Material.



Hình 3.8 Lưu đồ nhập xuất tồn kho

3.3.6 Lưu đồ tìm kiếm lịch sử nhập xuất

Chức năng tìm kiếm lịch sử nhập xuất cho phép người dùng lọc và xem lại các giao dịch đã thực hiện dựa trên nhiều tiêu chí khác nhau như loại nhập/xuất, khoảng thời gian, hoặc tên nguyên vật liệu.



Hình 3.9 Lưu đồ tìm kiếm lịch sử nhập xuất

View `search_transaction_table` render template `inventory/partial/transaction_table.html` với danh sách các lịch sử nhập xuất đã được lọc.

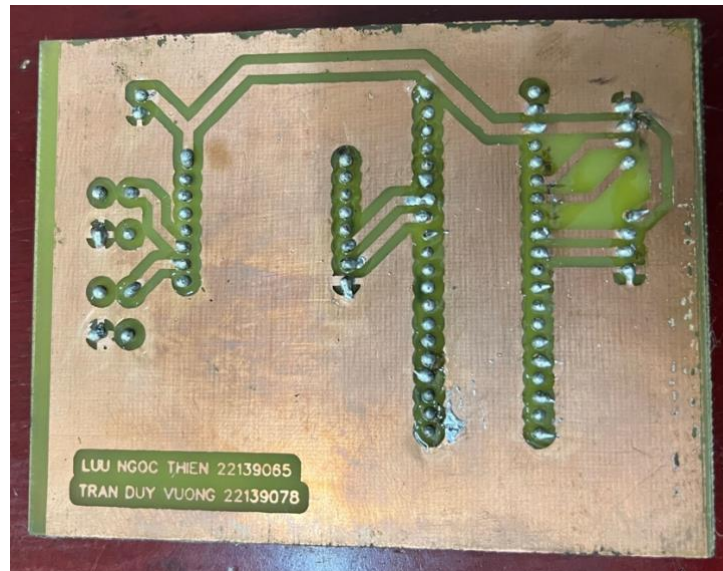
Nội dung HTML được render này sẽ được HTMX sử dụng để cập nhật bảng trả về lên giao diện.

Nếu nhấn nút Clear Filter thì yêu cầu GET đến `search_transaction_table` nhưng không kèm theo các giá trị lọc, dẫn đến hiển thị lại toàn bộ lịch sử nhập xuất.

3.4 KẾT QUẢ ĐẠT ĐƯỢC

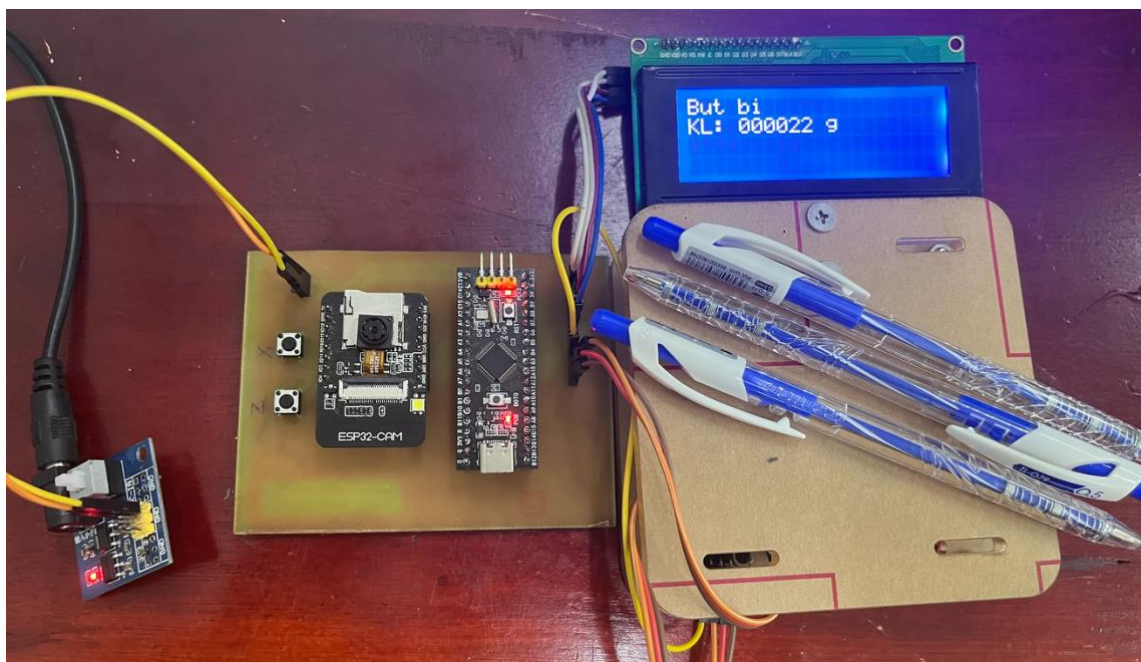
3.4.1 Kết quả thi công phần cứng

Hình 3.10 là hình ảnh thi công mặt sau PCB, gồm 2 nút nhấn và các hàng rào cái để gắn các linh kiện như esp32-cam, stm32, và nguồn.



Hình 3.10 Mặt sau PCB

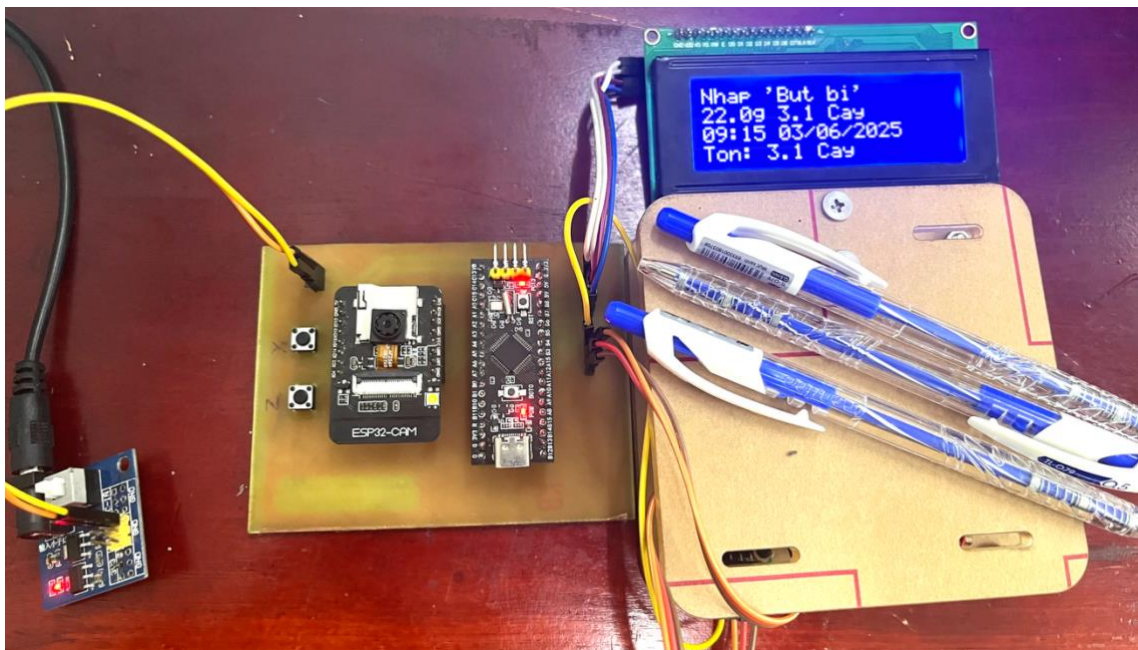
Hình 3.11 là trạng thái cân trước khi nhấn nút nhập/xuất. Ta bỏ đồ vật lên cân, sẽ hiển thị khối lượng, quét QR sẽ hiện tên sản phẩm. Nếu quét QR khác sẽ cập nhật lại tên sản phẩm.



Hình 3.11 Trạng thái cân trước khi nhấn nút nhập/xuất

Sau khi quét QR và cân, ta nhấn nút nhập/xuất. Hiện lên màn hình LCD sẽ là thông tin trả về như Hình 3.12. Bao gồm thông tin Nhập hay Xuất tên nguyên vật liệu gì. Khối lượng nhập xuất theo đơn vị gam. Kèm theo số liệu theo đơn vị cơ bản. Như ở đây định mức 1 cây bút là 7g. Nên 22g sẽ ứng với xấp xỉ 3 cây bút. Ở đây trả về 3.1 cây bởi vì sai số của load cell. Hiện lên LCD là thông tin thời gian, ngày nhập/ xuất và số lượng tồn kho trước khi nhập/xuất.

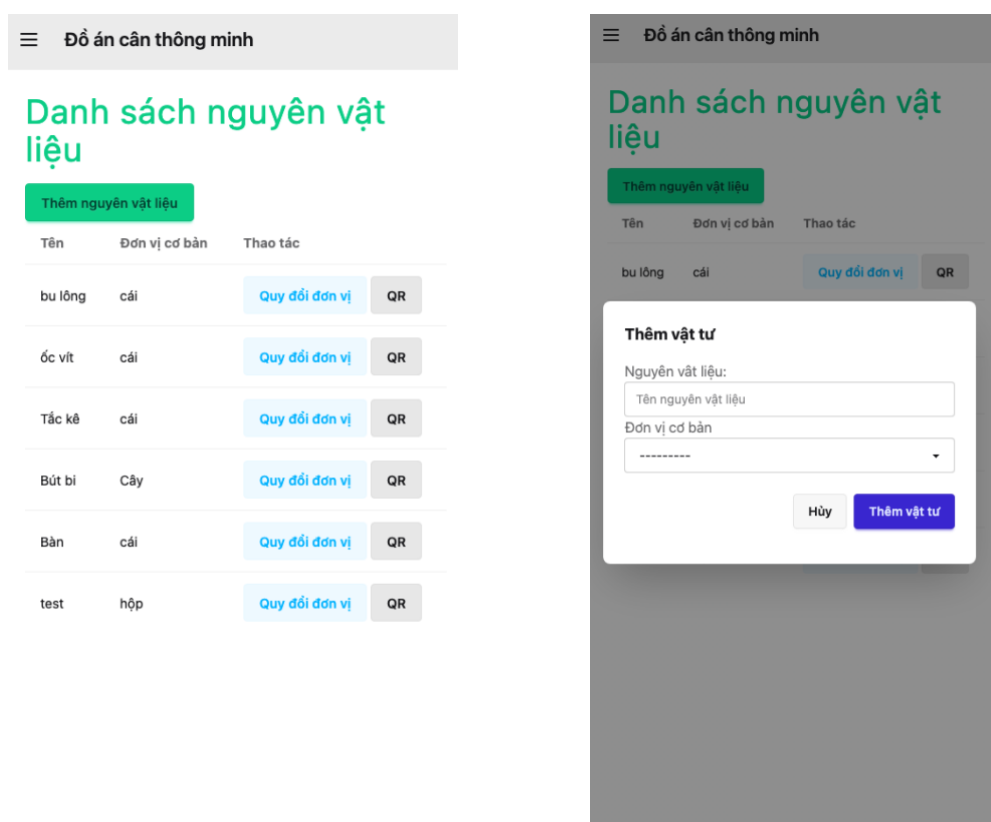
Thông tin trả về có thể tùy chỉnh theo ý muốn ở khối phần mềm. LCD ở đây mỗi hàng có 20 kí tự, nên khối phần mềm cũng trả về $20 \times 4 = 80$ kí tự, mỗi hàng tự fill dấu cách cho đủ 20 kí tự.



Hình 3.12 Trạng thái cân sau khi nhấn nút nhập/xuất

3.4.2 Kết quả thi công phần mềm

Hình 3.13 là giao diện để quản lý danh sách vật tư. Màn hình chính sẽ hiện các danh sách sản phẩm kho đang quản lý, đồng thời hiện đơn vị cơ bản. Nếu ta nhấn vào nút quy đổi đơn vị thì sẽ đến màn hình quy đổi đơn vị cho loại sản phẩm đó. Nếu nhấn vào nút thêm nguyên vật liệu. Ta sẽ thêm vật tư bằng cách nhập tên nguyên vật liệu, đồng thời chọn đơn vị cơ bản cho vật tư đó (từ danh sách đơn vị đã khai báo). Nếu ta nhấn vào nút QR thì sẽ hiện QR Code cho sản phẩm đó.



Hình 3.13 Chức năng quản lý nguyên vật liệu trong hệ thống

Hình 3.14 bên dưới là giao diện để quản lý danh sách đơn vị đang được dùng trong kho. Ta nhập vào tên đơn vị và nhấn nút thêm sẽ có danh sách đơn vị. Vì lý do bảo mật nên nếu muốn sửa xóa thì vào giao diện Django Admin riêng biệt

Đồ án cân thông minh

Thêm đơn vị

Name:

Tên đơn vị

Thêm

Danh sách đơn vị

kg

cái

g

hộp

lon

thùng

lốc

Test

Hình 3.14 Chức năng quản lí đơn vị của hệ thống

Hình 3.15 bên dưới là chức năng quy đổi đơn vị cho từng loại nguyên vật liệu trong hệ thống. Từ màn hình quản lí nguyên vật liệu, ta bấm vào nút quy đổi đơn vị bên trong hàng nguyên vật liệu sẽ dẫn đến màn hình này. Phần mềm sẽ tự khai báo đơn vị cơ bản với hệ số là 1. Các đơn vị khác được tính theo đơn vị cơ bản, chẳng hạn 1 cây bút nặng 20g thì ta nhập vào màn hình đơn vị g, và hệ số quy đổi là 20. Như vậy khi khối lượng cân nặng 20g, ta quy đổi về đơn vị cơ bản bằng cách $20/20 = 1$ cây.

Đồ án cân thông minh

Nhập xuất tồn

Lịch sử nhập xuất

Quản lý

< Back

Quy đổi đơn vị cho ốc vít

Danh sách đơn vị quy đổi

0.00005 kg

0.05 g

1.0 cái

Thêm đơn vị quy đổi cho ốc vít

Đơn vị:

Hệ số quy đổi:

Hệ số quy đổi

Thêm đơn vị

Hình 3.15 Chức năng quy đổi đơn vị cho từng loại nguyên vật liệu

Hình 3.16 thể hiện màn hình lịch sử nhập xuất vật tư. Khi form nhập xuất được submit hoặc nhấn nút nhập/ xuất từ khối phần cứng, thì thông tin nhập xuất sẽ được ghi nhận vào CSDL nhập xuất. Như hình bên dưới là toàn bộ lịch sử nhập xuất (có thể kéo xuống dưới để xem toàn bộ). Ở ô Loại giao dịch, ta có thể chọn loại Nhập/ Xuất/ Tất cả, thì sẽ lọc ra bảng với thông số tương ứng. Ở ô chọn ngày, ta có thể

chọn 1 ngày bất kì, hoặc một khoảng thời gian từ ngày nào tới ngày nào. Ở ô Nhập tên sản phẩm, thì ta nhập tên, có thể gõ vài kí tự tiếng Việt không dấu (với sự hỗ trợ extension unaccent trong PostgreSQL). Khi 1 trong 3 input thay đổi thì bảng sẽ tự động cập nhật lại 3 input bằng server-side rendering (htmx). Khi nhấn nút Clear Filter sẽ hiển thị lại toàn bộ lịch sử nhập xuất.

Đồ án cân thông minh			Nhập xuất tồn	Lịch sử nhập xuất	Quản lý ▾
Lịch sử nhập xuất					
Loại Giao Dịch:	Chọn ngày	Tìm kiếm sản phẩm			
-- Tất cả -- ▾	26/05/25 to 27/05/25	Nhập tên sp		Clear Filter	
Nguyên vật liệu	Số lượng	Đơn vị	Loại	Thời gian	
bu lông	321.0	g	export	May 27, 2025, 9:41 p.m.	
bu lông	172.0	g	export	May 27, 2025, 9:40 p.m.	
bu lông	66.0	g	export	May 27, 2025, 9:40 p.m.	
ốc vít	171.0	g	import	May 27, 2025, 9:40 p.m.	
ốc vít	183.0	g	import	May 27, 2025, 9:40 p.m.	
ốc vít	240.0	g	import	May 27, 2025, 9:39 p.m.	
ốc vít	240.0	g	import	May 27, 2025, 9:39 p.m.	
ốc vít	106.0	g	import	May 27, 2025, 9:39 p.m.	
ốc vít	17.0	g	import	May 27, 2025, 9:39 p.m.	
ốc vít	231.0	g	import	May 27, 2025, 9:39 p.m.	
Tắc kê	3900.0	cái	import	May 27, 2025, 5:05 p.m.	

Hình 3.16 Chức năng tìm kiếm lịch sử nhập xuất trong hệ thống

Hình 3.17 và Hình 3.18 là 2 form giao diện để nhập xuất thủ công qua giao diện web. Ta chọn tên nguyên vật liệu. Chọn loại đơn vị theo nguyên vật liệu đó. Nhập số lượng và nhấn nút lưu, hệ thống sẽ ghi nhận lại lịch sử nhập xuất này.

Hình 3.19 là chức năng hiển thị tồn kho trong hệ thống. Mặc định số liệu sẽ hiển thị theo đơn vị cơ bản. ở từng loại nguyên vật liệu, ta có thể chọn loại đơn vị khác thì giao diện sẽ hiển thị số liệu tồn kho theo đơn vị đó dựa theo đơn vị quy đổi. Khi nhấn nút nhập xuất trên cân thông minh, thì số lượng tồn kho theo đơn vị cơ bản sẽ được trả về màn hình LCD giống như trên web tồn kho.

Nhập

Xuất

Tồn

Nguyên vật liệu:

Đơn vị:

Số lượng:

Số lượng

Loại giao dịch:

Nhập

Lưu

Nhập

Xuất

Tồn

Nguyên vật liệu:

Đơn vị:

Số lượng:

Số lượng

Loại giao dịch:

Xuất

Lưu

Hình 3.17 Chức năng nhập kho

trong hệ thống

Hình 3.18 Chức năng xuất kho

trong hệ thống

Đồ án cân thông minh

Nhập

Xuất

Tồn

Tên	Tồn kho	Đơn vị
ốc vít	39719.0	cái
Tắc kê	965.0	cái
Bàn	0.0	cái
test	0.0	hộp
bu lông	87455.7	cái

Hình 3.19 Chức năng tồn kho trong hệ thống

CHƯƠNG 4

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1 KẾT LUẬN

Trong khuôn khổ đề tài đã đặt ra, **hệ thống cân thông minh tích hợp quản lý kho** đã được thiết kế và triển khai thành công. Nhờ đó, quy trình nhập kho, xuất kho và kiểm tra tồn kho đã được đơn giản hóa, tự động hóa, giúp tăng hiệu quả và giảm sai sót trong vận hành.

Hệ thống đã đáp ứng được các chức năng cơ bản của một mô hình quản lý kho ứng dụng cân thông minh, bao gồm: nhận diện vật tư qua mã QR, đo khối lượng vật tư, quy đổi ra số lượng theo định mức và ghi nhận giao dịch vào cơ sở dữ liệu.

Tuy nhiên, hệ thống vẫn còn tồn tại một số hạn chế:

- Sai số trong quá trình đo khối lượng do giới hạn kỹ thuật của load cell.
- Chưa tích hợp cơ chế xác thực và phân quyền người dùng.

4.2 HƯỚNG PHÁT TRIỂN

Trong thời gian tới, hệ thống có thể được mở rộng và nâng cấp theo các hướng sau nhằm đáp ứng tốt hơn nhu cầu ứng dụng thực tế tại doanh nghiệp:

- **Triển khai thực tế trên cân điện tử thật** tại các doanh nghiệp sản xuất, giúp đánh giá hiệu năng hệ thống trong điều kiện vận hành thực tế và tối ưu hoá quy trình nhập – xuất vật tư.
- **Cải tiến phần mềm quản lý kho**, bổ sung các tính năng nâng cao như:
 - Đăng nhập và phân quyền người dùng theo vai trò (quản trị viên, thủ kho, nhân viên...).
 - Tạo và quản lý BOM (Bill of Materials) để phục vụ sản xuất.
 - Quản lý nhà cung cấp và lịch sử giao dịch với đối tác.
- **Triển khai hệ thống phần mềm lên nền tảng đám mây AWS**, đảm bảo tính sẵn sàng, khả năng mở rộng và dễ dàng truy cập từ nhiều thiết bị, nhiều địa điểm khác nhau.

PHỤ LỤC

Slide:

[WMS_CanThongMinh_DoAn1.pptx](#)

Source code Hardware + Software:

https://github.com/vuongeris4/WMS_CanThongMinh

Video_demo:

https://www.youtube.com/watch?v=WtR_3SLUhm0

TÀI LIỆU THAM KHẢO

- [1] Huy Quang, "Quản lý kho là gì? 5 vai trò của quản lý kho trong nhà máy sản xuất," 17 9 2024. [Online]. Available: <https://facenet.vn/detail-product/quan-ly-kho-va-vai-tro-cua-quan-ly-kho/>. [Accessed 25 5 2025].
- [2] Nhựt Liên, "Django là gì? Giới thiệu tính ứng dụng phổ biến của Django với nguyên tắc hoạt động cơ bản," 25 5 2024. [Online]. Available: <https://fptshop.com.vn/tin-tuc/danh-gia/django-la-gi-175507>. [Accessed 15 5 2025].
- [3] Wikipedia, "htmx," [Online]. Available: <https://en.wikipedia.org/wiki/Htmx>. [Accessed 25 5 2025].
- [4] Dennis Ivy, "HTMX Saved Template Engines," 27 12 2024. [Online]. Available: <https://dennisivy.com/htmx-and-template-engines>. [Accessed 20 5 2025].
- [5] PostgreSQL, "About PostgreSQL," [Online]. Available: <https://www.postgresql.org/about/>. [Accessed 24 5 2025].
- [6] Wikipedia, "I2C," [Online]. Available: <https://vi.wikipedia.org/wiki/I%C2%B2C>. [Accessed 2 6 2025].
- [7] Wikipedia, "HTTP," [Online]. Available: https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol. [Accessed 3 6 2025].