

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Đồ án tổng hợp - hướng kỹ thuật dữ liệu (CO3127)

Thiết kế và triển khai website truy xuất thông tin nông sản qua QR Code

Giảng viên hướng dẫn: TS.PHAN TRỌNG NHÂN

Sinh viên thực hiện	Trần Đức Khôi Nguyên	MSSV: 2352837
	Lý Quốc Hào	MSSV: 2352294
	Lương Đức Huy	MSSV: 2352384
	Nguyễn Ngọc Minh Hiền	MSSV: 2352339

Lớp: CN01

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 12 2025



Mục lục

List of Tables	3
List of Figures	3
Danh sách sinh viên và phân công	5
1 Giới thiệu chung	6
1.1 Bối cảnh ra đời dự án	6
1.1.1 Xu hướng ứng dụng công nghệ số trong quản lý và vận hành	6
1.1.2 Hạn chế của các hệ thống truyền thống	6
1.1.3 Nhu cầu phát triển ứng dụng web tích hợp QR	7
1.2 Project Objectives	7
1.3 Scope of Work	8
1.4 Technologies Used	8
2 Thiết kế & Chuẩn bị Database	10
2.1 Bảng chi tiết	10
2.1.1 Products	10
2.1.2 Farms	13
2.1.3 Product_Certifications	15
2.1.4 Certifications	17
2.1.5 ProductImages	19
2.1.6 ApprovalRequests	21
2.1.7 Users	23
3 Xây dựng ứng dụng	26
3.1 Mô tả kiến trúc hệ thống	26
3.2 Chức năng quét QR	28
3.3 Chức năng CRUD và Quy trình phê duyệt trong hệ thống FruitTrace . . .	35
3.3.1 Quản lý Sản phẩm (Product Management)	35
3.3.2 Quản lý yêu cầu thay đổi (Request List)	41
3.3.3 Quản lý người dùng và phân quyền	47
4 Các kĩ thuật dữ liệu được áp dụng	48
4.1 Normalization	48



4.2	Query Optimization	48
4.3	Data Indexing	50
4.4	Data Backup & Recovery	51
4.5	Database Security	52
5	Đánh giá các hạn chế và nội dung chưa đạt được	53
6	Đánh giá kết quả đạt được của dự án	55

Danh sách bảng

0.1	Danh sách sinh viên và phân công	5
-----	--	---

Danh sách hình vẽ

2.1	Data mẫu bảng PRODUCTS	10
2.2	Data mẫu bảng FARM	13
2.3	Data mẫu bảng PRODUCT CERTIFICATIONS	16
2.4	Data mẫu bảng CERTIFICATIONS	18
2.5	Data mẫu bảng PRODUCT IMAGES	20
2.6	Data mẫu bảng APPROVAL REQUESTS	21
2.7	Data mẫu bảng USER	23
3.1	Giao diện quét mã QR	29
3.2	Giao diện quét QR bằng camera	30
3.3	Giao diện quét QR bằng camera	30
3.4	Giao diện tải lên ảnh chứa mã QR	31
3.5	Giao diện hiển thị sản phẩm thành công	32
3.6	Bảng hộp thoại Product Information	33
3.7	Product not found or has been deleted!	34
3.8	Giao diện Dashboard tổng quan của hệ thống FruitTrace	35
3.9	Giao diện Dashboard biểu đồ	36
3.10	Giao diện Product Management	36
3.11	Giao diện bộ lọc giúp tối ưu việc tra cứu	37
3.12	Giao diện hiển thị thông tin sản phẩm còn hàng IN STOCK	38
3.13	Giao diện hiển thị thông tin sản phẩm OUT OF STOCK	38



3.14	Giao diện hiển thị mã QR	39
3.15	Giao diện chỉnh sửa thông tin	40
3.16	Giao diện quản lý yêu cầu	41
3.17	Giao diện thêm thông tin sản phẩm	42
3.18	43
3.19	Giao diện chỉnh sửa thông tin	44
3.20	45
3.21	46
3.22	46



Danh sách sinh viên và phân công

No.	Fullname	Student ID	Task	% done
1	Trần Đức Khôi Nguyên	2352837	Backend	100%
2	Lý Quốc Hào	2352294	Backend	100%
3	Lương Đức Huy	2352384	Database	100%
4	Nguyễn Ngọc Minh Hiền	2352339	Frontend, report	100%

Table 0.1: Danh sách sinh viên và phân công

1 Giới thiệu chung

1.1 Bối cảnh ra đời dự án

1.1.1 Xu hướng ứng dụng công nghệ số trong quản lý và vận hành

Trong những năm gần đây, sự phát triển nhanh chóng của các công nghệ số đã mang lại những thay đổi căn bản trong cách thức quản lý và vận hành ở nhiều lĩnh vực khác nhau. Đặc biệt, các ngành như bán lẻ, logistics, quản lý kho, theo dõi sản phẩm và chăm sóc khách hàng đã chứng kiến những bước tiến đáng kể nhờ việc áp dụng các giải pháp công nghệ số. Sự số hóa không chỉ giúp tối ưu hóa quy trình vận hành mà còn tạo điều kiện nâng cao hiệu quả quản lý, giảm thiểu chi phí và tăng cường trải nghiệm người dùng.

Trong số các giải pháp công nghệ được áp dụng, mã QR (Quick Response Code) đã trở thành một công cụ quan trọng và phổ biến. QR code là một dạng mã hai chiều có khả năng lưu trữ thông tin phong phú và được quét nhanh chóng bằng thiết bị di động hoặc máy quét chuyên dụng. Việc sử dụng QR code mang lại nhiều lợi ích thiết thực: giảm thiểu các thao tác nhập liệu thủ công, hạn chế sai sót dữ liệu do con người gây ra, đồng thời tăng tốc độ xử lý thông tin. Các doanh nghiệp hiện nay thường ứng dụng QR code để quản lý tồn kho, theo dõi sản phẩm trong chuỗi cung ứng, kiểm tra thông tin sản phẩm và tạo trải nghiệm tương tác với khách hàng một cách tiện lợi. Ví dụ, trong ngành bán lẻ, nhân viên có thể quét QR code để kiểm tra nhanh tình trạng tồn kho hoặc cập nhật thông tin sản phẩm; trong lĩnh vực logistics, QR code giúp theo dõi hành trình vận chuyển hàng hóa từ kho đến điểm giao hàng, giảm thiểu nhầm lẫn và thất thoát.

1.1.2 Hạn chế của các hệ thống truyền thống

Tuy nhiên, bên cạnh những lợi ích trên, nhiều hệ thống quản lý truyền thống vẫn dựa vào nhập liệu thủ công, dẫn đến nhiều hạn chế đáng kể. Quá trình này thường tiêu tốn nhiều thời gian, dễ xảy ra sai sót dữ liệu, độ trễ trong xử lý thông tin cao và khó đảm bảo tính chính xác cũng như đồng bộ của hệ thống. Đồng thời, trải nghiệm người dùng trong các hệ thống truyền thống chưa thực sự tối ưu; nhân viên phải thao tác nhiều bước, dữ liệu không được cập nhật kịp thời, gây khó khăn trong việc ra quyết định và quản lý. Những vấn đề này càng trở nên rõ ràng trong bối cảnh doanh nghiệp cần xử lý lượng thông tin lớn, yêu cầu tốc độ và độ chính xác cao.

1.1.3 Nhu cầu phát triển ứng dụng web tích hợp QR

Do đó, việc xây dựng một ứng dụng web tích hợp QR code và kết nối trực tiếp với hệ quản trị cơ sở dữ liệu là nhu cầu cấp thiết trong bối cảnh hiện nay. Ứng dụng này không chỉ giúp tự động hóa quy trình quản lý, giảm thiểu sai sót dữ liệu mà còn nâng cao hiệu quả vận hành, giúp các doanh nghiệp quản lý thông tin nhanh chóng, chính xác và tiện lợi hơn. Thêm vào đó, việc ứng dụng công nghệ này còn mở ra cơ hội phát triển các giải pháp thông minh, như phân tích dữ liệu thời gian thực, theo dõi hành vi khách hàng và tối ưu hóa quy trình vận hành nội bộ. Nhờ đó, doanh nghiệp có thể cải thiện năng lực cạnh tranh, tăng cường trải nghiệm khách hàng và đẩy mạnh hiệu quả kinh doanh trong kỷ nguyên số.

Tóm lại, sự kết hợp giữa công nghệ số, QR code và quản lý dữ liệu tự động không chỉ giải quyết các hạn chế của hệ thống truyền thống mà còn mở ra hướng đi mới cho các doanh nghiệp trong việc quản lý thông tin, vận hành hiệu quả và nâng cao trải nghiệm người dùng. Việc phát triển các ứng dụng web có khả năng quét QR và tích hợp trực tiếp với cơ sở dữ liệu đang trở thành một giải pháp thiết thực, cần thiết và đầy tiềm năng trong tương lai gần.

1.2 Project Objectives

Dự án Truy Xuất Nguồn Gốc Nông Sản là một hệ thống web hỗ trợ quản lý và truy xuất nguồn gốc các sản phẩm nông nghiệp thông qua mã QR. Hệ thống hướng tới việc hiện đại hóa quy trình quản lý nông sản, tăng tính minh bạch, hỗ trợ người tiêu dùng và nâng cao hiệu quả vận hành cho doanh nghiệp. Hệ thống được thiết kế với ba nhóm người dùng chính, mỗi nhóm có quyền hạn và chức năng riêng biệt:

- Người dùng công cộng (Public User): Có thể quét mã QR gắn trên bao bì hoặc tem truy xuất để xem toàn bộ thông tin liên quan đến sản phẩm như: tên sản phẩm, ngày thu hoạch, nơi sản xuất, quy trình đóng gói, chứng nhận an toàn,... Người dùng không cần đăng nhập và chỉ có quyền xem.
- Nhân viên (Staff): Có tài khoản đăng nhập để thực hiện các tác vụ quản lý sản phẩm. Nhân viên có thể tạo mới, chỉnh sửa hoặc yêu cầu xóa sản phẩm. Tuy nhiên, tất cả các thay đổi đều phải chờ Admin phê duyệt trước khi được áp dụng lên hệ thống chính thức. Điều này giúp đảm bảo tính chính xác và minh bạch của dữ liệu.
- Quản trị viên (Admin): Có toàn quyền đối với hệ thống. Admin có thể duyệt hoặc từ chối các yêu cầu thay đổi sản phẩm từ nhân viên, quản lý tài khoản nhân sự, và

giám sát toàn bộ hoạt động của hệ thống. Admin đóng vai trò đảm bảo chất lượng dữ liệu và tính minh bạch trước khi người dùng cuối truy xuất.

Hệ thống được phát triển tuân thủ các nguyên tắc của Data Engineering như chuẩn hoá dữ liệu, tối ưu truy vấn, lập chỉ mục và đảm bảo tính toàn vẹn dữ liệu. Đồng thời, dung mã QR giúp rút ngắn quy trình truy xuất thông tin, giảm sai sót thủ công và tăng độ tin cậy trong quản lý chuỗi cung ứng.

Mục tiêu của dự án là phát triển một ứng dụng web đơn giản cho phép:

- Quét mã QR bằng camera thiết bị trực tiếp trên trình duyệt.
- Truy xuất và đồng bộ dữ liệu với cơ sở dữ liệu thông qua API.
- Hỗ trợ các thao tác quản lý dữ liệu cơ bản (CRUD) một cách nhanh chóng và chính xác.
- Minh chứng các kỹ thuật thiết kế và tối ưu hoá cơ sở dữ liệu đã được học.

1.3 Scope of Work

Phạm vi dự án bao gồm:

- Thiết kế và xây dựng cơ sở dữ liệu mẫu với tối thiểu 3 bảng.
- Xây dựng ứng dụng web cho phép quét mã QR và thực hiện các thao tác CRUD.
- Áp dụng các kỹ thuật về dữ liệu: chuẩn hoá, tối ưu truy vấn, lập chỉ mục, sao lưu và phục hồi dữ liệu, cũng như các biện pháp bảo mật cơ bản.
- Viết báo cáo tổng kết mô tả quá trình triển khai và kết quả.

Các nội dung ngoài phạm vi, chẳng hạn như tích hợp hệ thống phức tạp, triển khai thực tế quy mô lớn hoặc chức năng thương mại hoá, sẽ không được đề cập trong báo cáo.

1.4 Technologies Used

Dự án sử dụng các công nghệ và công cụ chính sau:

- **Frontend:** Giao diện người dùng được xây dựng bằng **HTML5**, **CSS3** và **JavaScript**, kết hợp với **Bootstrap 5** nhằm tối ưu hóa khả năng responsive trên nhiều kích thước màn hình. Ngoài ra, nhóm sử dụng **Axios** để thực hiện các yêu cầu HTTP bất đồng

bộ, giúp giao diện tương tác mượt mà hơn. Toàn bộ cấu trúc giao diện được thiết kế theo hướng tối giản, dễ mở rộng và phù hợp với trải nghiệm người dùng hiện đại.

- **Backend:** Phần xử lý phía máy chủ được phát triển bằng **Node.js** với framework **Express.js**. Kiến trúc RESTful API được áp dụng nhằm đảm bảo khả năng tách biệt giữa frontend và backend, đồng thời tạo điều kiện thuận lợi cho việc mở rộng hệ thống trong tương lai. Backend chịu trách nhiệm xử lý logic nghiệp vụ, xác thực dữ liệu đầu vào, giao tiếp với cơ sở dữ liệu và quản lý luồng thông tin giữa các thành phần của hệ thống.
- **Database:** Dự án sử dụng hệ quản trị cơ sở dữ liệu **MySQL**. Toàn bộ dữ liệu người dùng, thông tin sản phẩm, lịch sử thao tác và các bảng liên quan được thiết kế theo mô hình quan hệ (Relational Model). Nhóm áp dụng chuẩn hóa dữ liệu đến 3NF nhằm giảm thiểu trùng lặp dữ liệu và đảm bảo tính toàn vẹn. ORM **Sequelize** được tích hợp để giúp quá trình truy vấn, ánh xạ bảng và quản lý dữ liệu trở nên đơn giản, an toàn và dễ bảo trì hơn.
- **Công cụ hỗ trợ:** Nhóm sử dụng **Git** và nền tảng **GitHub** để quản lý phiên bản mã nguồn, hỗ trợ phân chia công việc và phát hiện xung đột mã. Công cụ **Postman** được dùng để kiểm thử API, giúp đảm bảo các endpoint hoạt động ổn định trước khi tích hợp vào frontend. Ngoài ra, **VS Code** là môi trường phát triển chính, kết hợp với các extension như Prettier và ESLint để giữ cho mã nguồn sạch, thống nhất và dễ đọc. Trong quá trình báo cáo và triển khai tài liệu, nhóm sử dụng **LaTeX** để đảm bảo tính chuyên nghiệp và nhất quán về định dạng.

2 Thiết kế & Chuẩn bị Database

2.1 Bảng chi tiết

2.1.1 Products

Bảng Products được sử dụng để quản lý thông tin chi tiết của các sản phẩm nông sản trong hệ thống, đóng vai trò trung tâm trong việc truy xuất nguồn gốc thông qua mã QR. Mỗi dòng dữ liệu tương ứng với một sản phẩm cụ thể, được liên kết với bảng Farms thông qua FarmID, cho phép xác định rõ sản phẩm được sản xuất tại trang trại nào. Bảng này cung cấp đầy đủ thông tin từ mô tả, giá bán cho đến vòng đời sản xuất và tình trạng tồn kho của sản phẩm.

ProductID	FarmID	ProductName	ProdDescription	Price	PlantDate	HarvestDate	QRUrl	Status
5	1	Licensed Ceramic Bike	Innovative Computer featuring	23100	2025-10-02	2026-09-21	http://localhost:3000/product/	outstock
6	3	Luxurious Ceramic Mouse	Discover the eagle-like agility	17200	2025-11-24	2026-07-01	http://localhost:3000/product/	instock
7	20	Licensed Bronze Tuna	Professional-grade Keyboard	18800	2025-07-11	2026-05-14	http://localhost:3000/product/	instock
8	14	Soft Bamboo Chips	Savor the tender essence in o	10300	2025-09-13	2026-08-23	http://localhost:3000/product/	instock
9	10	Gorgeous Aluminum Bike	The olive Chicken combines B	28400	2025-02-21	2026-07-21	http://localhost:3000/product/	instock
11	9	Ergonomic Rubber Hat	Experience the azure brillianc	15600	2025-10-15	2026-11-27	http://localhost:3000/product/	outstock
12	18	Oriental Gold Soap	Stylish Soap designed to mak	29700	2024-12-07	2026-11-09	http://localhost:3000/product/	instock
13	17	Frozen Bamboo Pants	The Diverse transitional encoc	23800	2025-09-04	2026-09-04	http://localhost:3000/product/	instock
14	8	Oriental Wooden Hat	Featuring Dubnium-enhanced	18500	2025-03-13	2026-09-21	http://localhost:3000/product/	outstock
15	16	Refined Steel Mouse	Our fox-friendly Table ensur	23500	2025-11-18	2026-08-25	http://localhost:3000/product/	instock
16	1	Bespoke Silk Ball	The Yadira Tuna is the latest	18600	2025-12-04	2025-12-06	http://localhost:3000/product/	instock
17	6	Tasty Granite Sausages	Stylish Gloves designed to	30000	2025-09-24	2026-03-18	http://localhost:3000/product/	outstock
18	1	Modern Concrete Bacon	Featuring Thorium-enhanced	23200	2025-12-08	2025-12-11	http://localhost:3000/product/	instock
20	11	Intelligent Steel Fish	Our whale-friendly Car ensur	28400	2025-10-27	2026-06-02	http://localhost:3000/product/	outstock
21	20	Gorgeous Granite Mouse	The Ergonomic mission-critica	24800	2025-11-17	2026-08-28	http://localhost:3000/product/	instock

Figure 2.1: Data mẫu bảng PRODUCTS

- **ProductID**: Mã định danh duy nhất của sản phẩm trong hệ thống. Mỗi sản phẩm chỉ có một ProductID và không được trùng lặp hoặc để trống. Ví dụ: $ProductID = 5$, $ProductID = 19$. Trường này đóng vai trò là **khóa chính** của bảng Products, đồng thời được sử dụng để liên kết với các bảng liên quan như *ProductImages* và *Product_Certifications*, phục vụ việc quản lý hình ảnh và chứng nhận của từng sản phẩm.
- **FarmID**: Mã định danh của trang trại sản xuất sản phẩm, tham chiếu trực tiếp đến bảng *Farms*. Ví dụ: $FarmID = 1$, $FarmID = 20$. Trường này giúp xác định rõ **nguồn gốc sản xuất** của sản phẩm và đảm bảo mọi sản phẩm đều gắn với một trang trại hợp lệ, từ đó hỗ trợ truy xuất nguồn gốc khi người dùng quét mã QR.



- **ProductName:** Tên hiển thị của sản phẩm trong hệ thống. Ví dụ: *Licensed Ceramic Bike, Oriental Gold Soap*. Trường này giúp người dùng dễ dàng nhận diện sản phẩm khi xem thông tin chi tiết hoặc khi truy xuất dữ liệu thông qua mã QR.
- **ProdDescription:** Mô tả chi tiết về sản phẩm, dùng để giới thiệu đặc điểm, chất lượng hoặc giá trị nổi bật. Nội dung mô tả giúp tăng mức độ minh bạch thông tin và nâng cao trải nghiệm người dùng khi tìm hiểu về sản phẩm.
- **Price:** Giá bán của sản phẩm, thường được lưu dưới dạng số nguyên dương. Ví dụ: *23100, 30000*. Trường này phục vụ cho việc hiển thị giá, quản lý thương mại và thống kê dữ liệu kinh doanh trong hệ thống.
- **PlantDate:** Ngày bắt đầu gieo trồng sản phẩm. Ví dụ: *2025-10-02*. Trường này phản ánh thời điểm khởi đầu của vòng đời sản xuất, giúp người dùng và hệ thống theo dõi quá trình canh tác.
- **HarvestDate:** Ngày thu hoạch sản phẩm. Ví dụ: *2026-09-21*. Trường này cho phép xác định thời gian kết thúc chu kỳ sản xuất và hỗ trợ đánh giá độ tươi mới cũng như tính hợp lệ của sản phẩm.
- **QRUrl:** Đường dẫn đến trang thông tin chi tiết của sản phẩm trong hệ thống. Ví dụ: *http://localhost:3000/product/*. Đây là cầu nối giữa sản phẩm vật lý và dữ liệu số, được sử dụng khi người dùng quét mã QR trên bao bì sản phẩm.
- **Status:** Trạng thái tồn kho của sản phẩm trong hệ thống. Các giá trị hợp lệ bao gồm *instock* (còn hàng) và *outstock* (hết hàng). Trường này hỗ trợ quản lý cung ứng và đảm bảo hệ thống hiển thị trạng thái phù hợp cho người dùng.

Lưu ý về ràng buộc và tính toàn vẹn dữ liệu:

- **ProductID:**
 - Không được trùng lặp trong bảng Products.
 - Không được để trống; nếu bỏ trống, hệ thống không thể định danh sản phẩm.
 - Dùng làm khóa chính và làm khóa ngoại liên kết với các bảng khác.
- **FarmID:**



- Phải tồn tại trong bảng Farms; nếu FarmID không tồn tại, hệ thống sẽ phát sinh lỗi khóa ngoại.
- Không được NULL; nếu bỏ trống sẽ làm sản phẩm “mất nguồn gốc”.
- **ProductName và ProdDescription:**
 - Không được để trống; nếu NULL sẽ giảm khả năng nhận diện sản phẩm.
 - Nội dung phải phù hợp; tránh nhập dữ liệu rác, ký tự đặc biệt gây lỗi hiển thị.
- **Price:**
 - Phải là số nguyên dương; giá âm hoặc bằng 0 bị xem là không hợp lệ.
 - Dùng cho tính toán thương mại và báo cáo, nếu sai sẽ làm lệch doanh thu hoặc tồn kho.
- **PlantDate và HarvestDate:**
 - HarvestDate không được nhỏ hơn PlantDate; nếu vi phạm gây sai lệch vòng đời sản xuất.
 - Phải đúng định dạng ngày; lỗi định dạng sẽ gây lỗi xử lý dữ liệu.
- **QRUrl:**
 - Phải là URL hợp lệ; nếu thiếu hoặc sai định dạng, chức năng truy xuất sản phẩm qua QR sẽ không hoạt động.
 - URL cần trỏ tới trang chi tiết sản phẩm; không được trống.
- **Status:**
 - Chỉ nhận giá trị *instock* hoặc *outstock*.
 - Các giá trị khác đều không hợp lệ, sẽ gây lỗi hiển thị tồn kho hoặc đặt hàng.
- **Toàn vẹn dữ liệu tổng thể:**
 - Nếu bất kỳ khóa ngoại nào (FarmID, ProductID) không tồn tại, hệ thống báo lỗi.
 - Dữ liệu trống hoặc sai định dạng ở các trường quan trọng sẽ ảnh hưởng đến các chức năng truy xuất nguồn gốc, hiển thị sản phẩm, thống kê tồn kho và phân tích thương mại.



- Hệ thống cần kiểm tra các ràng buộc khi thêm, sửa, xóa sản phẩm để tránh mất tính toàn vẹn dữ liệu.

Về ý nghĩa tổng thể, bảng Products giúp hệ thống không chỉ quản lý sản phẩm ở mức dữ liệu, mà còn phản ánh đầy đủ nguồn gốc, quá trình sản xuất, giá trị thương mại và trạng thái cung ứng. Đây là bảng cốt lõi để kết nối trang trại, chứng nhận, hình ảnh và mã QR, từ đó xây dựng một hệ thống truy xuất nguồn gốc nông sản minh bạch, đáng tin cậy và có khả năng mở rộng trong tương lai.

2.1.2 Farms

Bảng Farms được sử dụng để quản lý thông tin về các trang trại/nơi sản xuất trong hệ thống, là cơ sở để xác định nguồn gốc nông sản và liên kết trực tiếp với bảng Products thông qua FarmID. Mỗi dòng dữ liệu tương ứng với một trang trại cụ thể, cho phép một trang trại có thể cung cấp nhiều sản phẩm khác nhau, từ đó hỗ trợ truy xuất nguồn gốc rõ ràng khi người dùng quét mã QR.

FarmID int	FarmName varchar(255)	Address text	Owner varchar(255)	Contact text	CreatedAt timestamp
1	Green Farm Demo	2116 Harrison Burgs	Kristi Turner	SĐT: (482) 343-3044 x704	2025-12-04 10:54:22
2	Trang trại cash Loystad	6655 Evangeline Cove	Mr. Arturo Waters	SĐT: 1-415-886-8278	2025-12-04 10:54:22
3	Trang trại tribe Lake Daishafur	998 Lakin Lane	Ellen Reinger	SĐT: 544-912-6162	2025-12-04 10:54:22
4	Trang trại assist Port Gussie	16409 Leone Fields	Ray Berge	SĐT: 432.567.1969 x220	2025-12-04 10:54:22
5	Trang trại handful Manhattan	17534 Jameson Causeway	Dr. Leslie Shields	SĐT: 327.992.1854	2025-12-04 10:54:22
6	Trang trại taxicab East Wendy	974 Hilpert Mountains	Lisa Runte	SĐT: 870.304.9978	2025-12-04 10:54:22
7	Trang trại mousse Keller	611 Prohaska Views	Saul Marvin	SĐT: (436) 748-8776 x0941	2025-12-04 10:54:22
8	Trang trại commodity North Ar	6763 Joyce Parkways	Wendy Welch	SĐT: (484) 769-6295	2025-12-04 10:54:22
9	Trang trại duffel Fall River	5209 Jena Courts	Alfredo Blanda	SĐT: 488-559-0182 x6202	2025-12-04 10:54:22
10	Trang trại honesty Fort Martin	94105 N Bridge Street	Joy Lemke	SĐT: 914-911-8492 x35868	2025-12-04 10:54:23
11	Trang trại siege South Arch	78798 Nyah Club	Alicia Schamberger	SĐT: 758-441-3634 x8472	2025-12-04 10:54:23
12	Trang trại reconsideration Noe	29551 S 2nd Street	Mamie McGlynn	SĐT: 1-364-554-1573 x3553	2025-12-04 10:54:23
13	Trang trại vibration West Junil	578 Reichel Extension	Kara Hane	SĐT: 596.575.9597 x882	2025-12-04 10:54:23
14	Trang trại circumference New	9615 Mina Crossing	Sonia Wisozk	SĐT: 934-962-0460 x504	2025-12-04 10:54:23
15	Trang trại circumference Mish	219 West Road	Reginald Dare	SĐT: (953) 811-8154 x2428	2025-12-04 10:54:23

Figure 2.2: Data mẫu bảng FARM

- FarmID

- Mã định danh duy nhất của trang trại
- Ví dụ: FarmID = 1, FarmID = 10



- Được sử dụng để liên kết với các sản phẩm trong bảng *Products*

- **FarmName**

- Tên trang trại
- Có thể là tên chính thức (Green Farm Demo) hoặc tên mô tả (Trang trại honesty Fort Martin)
- Giúp người dùng dễ nhận diện nguồn sản xuất

- **Address**

- Địa chỉ cụ thể của trang trại (số nhà, đường, khu vực)
- Phục vụ xác minh nguồn gốc
- Có thể mở rộng để tích hợp bản đồ trong tương lai

- **Owner**

- Tên chủ sở hữu hoặc người đại diện trang trại
- Ví dụ: Kristi Turner, Joy Lemke
- Thể hiện đơn vị chịu trách nhiệm chính về sản xuất

- **Contact**

- Thông tin liên hệ của trang trại (chủ yếu là số điện thoại)
- Có thể có nhiều định dạng khác nhau (có mã vùng, máy nhánh)

- **CreatedAt**

- Thời điểm trang trại được tạo trong hệ thống
- Ví dụ: 2025-12-04 10:54:22
- Do hệ thống tự sinh, phục vụ quản lý và theo dõi dữ liệu

Về mặt giá trị dữ liệu, FarmID không được trùng lặp hoặc để trống; FarmName, Address và Owner không nên để NULL vì sẽ làm giảm tính minh bạch nguồn gốc; Contact phải là chuỗi hợp lệ để đảm bảo liên hệ được; CreatedAt do hệ thống tự sinh và không chỉnh sửa thủ công. Nếu FarmID không tồn tại nhưng vẫn được dùng trong bảng Products, hoặc thông tin liên hệ bị bỏ trống hoàn toàn, hệ thống sẽ phát sinh lỗi về tính toàn vẹn dữ liệu và ảnh hưởng đến chức năng truy xuất nguồn gốc.

Về ý nghĩa tổng thể, bảng Farms giúp hệ thống không chỉ biết sản phẩm là gì mà còn biết sản phẩm đến từ đâu và ai chịu trách nhiệm. Việc quản lý rõ ràng thông tin trang trại góp phần nâng cao tính minh bạch, độ tin cậy của nông sản, đồng thời tạo nền tảng để mở rộng các chức năng như thống kê theo trang trại, đánh giá chất lượng vùng sản xuất và hiển thị thông tin chi tiết cho người tiêu dùng.

2.1.3 Product_Certifications

Bảng Product_Certifications được sử dụng để quản lý thông tin chứng nhận gắn liền với từng sản phẩm, thể hiện mức độ đạt chuẩn về chất lượng, an toàn hoặc nguồn gốc. Mỗi dòng dữ liệu tương ứng với một chứng nhận của một sản phẩm, được liên kết thông qua ProductID và CertID, cho phép một sản phẩm có thể sở hữu nhiều chứng nhận khác nhau, và một loại chứng nhận cũng có thể áp dụng cho nhiều sản phẩm.

- **ProductID**

- Mã định danh của sản phẩm, đã tồn tại trong bảng *Products*
- Ví dụ: ProductID = 5, ProductID = 12
- Dùng để xác định chứng nhận này thuộc về sản phẩm nào
- Không được trùng lặp trong cùng một cặp ProductID – CertID
- Không được NULL
- Nếu ProductID không tồn tại trong bảng Products, hệ thống sẽ phát sinh lỗi khóa ngoại (foreign key constraint)

- **CertID**

- Mã định danh của loại chứng nhận
- Giúp phân biệt các chuẩn chất lượng, an toàn và nguồn gốc khác nhau
- Một sản phẩm có thể có nhiều CertID khác nhau
- Không được NULL
- Nếu CertID không tồn tại trong bảng chứng nhận, dữ liệu sẽ không hợp lệ

- **IssueDate**

- Ngày chứng nhận được cấp
- Ví dụ: 2025-02-27



 Produ int	 Cer int	IssueDate date	ExpireDate date
5	14	2025-02-27	2026-03-13
5	18	(NULL)	(NULL)
6	7	2025-10-01	2026-08-02
7	10	2025-10-27	2026-05-15
8	12	2025-08-10	2026-11-24
9	14	2025-11-08	2026-02-02
11	13	2025-05-20	2026-12-02
12	9	2025-03-19	2026-06-11
13	4	2025-10-03	2026-02-25
14	15	2025-10-17	2026-05-29
15	8	2025-04-10	2026-02-13
16	17	2025-05-29	2026-06-29
17	16	2025-10-18	2026-11-21
18	2	2024-12-10	2026-08-20
20	17	2025-10-02	2026-05-04

Figure 2.3: Data mẫu bảng PRODUCT CERTIFICATIONS

- Thể hiện thời điểm sản phẩm bắt đầu đạt chuẩn
- Có thể mang giá trị NULL trong trường hợp chứng nhận đang xét duyệt hoặc chưa được cấp chính thức
- Không được lớn hơn ExpireDate (nếu ExpireDate tồn tại)

- **ExpireDate**

- Ngày hết hạn của chứng nhận
- Ví dụ: 2026-03-13

- Dùng để theo dõi hiệu lực và nhắc nhở gia hạn chứng nhận
- Có thể mang giá trị NULL đối với chứng nhận không có thời hạn
- Nếu *ExpireDate* nhỏ hơn *IssueDate*, dữ liệu được xem là không hợp lệ

Về mặt ý nghĩa, bảng *Product_Certifications* giúp hệ thống theo dõi tính hợp lệ của các chứng nhận theo thời gian, đảm bảo chỉ những sản phẩm còn chứng nhận hợp lệ mới được hiển thị hoặc giới thiệu đến người dùng. Đồng thời, việc quản lý rõ ràng ngày cấp và ngày hết hạn giúp nâng cao tính minh bạch, hạn chế rủi ro về pháp lý và củng cố niềm tin của người tiêu dùng đối với sản phẩm.

2.1.4 Certifications

Bảng *Certifications* được sử dụng để quản lý toàn bộ thông tin về các chứng nhận mà hệ thống hỗ trợ, bao gồm tên chứng nhận, mô tả chi tiết và tổ chức cấp. Mỗi bản ghi trong bảng tương ứng với một chứng nhận duy nhất, phản ánh các tiêu chuẩn, quy định hoặc xác nhận chất lượng mà sản phẩm có thể đạt được. Thông tin này đóng vai trò quan trọng trong việc xác thực và minh bạch hóa dữ liệu sản phẩm, đồng thời giúp người dùng, staff và admin dễ dàng kiểm tra các chứng nhận hợp lệ trước khi áp dụng vào sản phẩm.

Thông qua **CertID**, bảng *Certifications* được liên kết trực tiếp với bảng *Product_Certifications*, cho phép xác định sản phẩm nào sở hữu chứng nhận nào, cũng như quản lý mối quan hệ nhiều-nhiều giữa sản phẩm và chứng nhận. Điều này rất quan trọng trong việc xây dựng một hệ thống truy xuất nguồn gốc đầy đủ và đáng tin cậy, nơi mà mỗi sản phẩm có thể có nhiều chứng nhận khác nhau, đồng thời một chứng nhận cũng có thể áp dụng cho nhiều sản phẩm.

Ngoài ra, bảng còn hỗ trợ hệ thống trong việc kiểm soát và đánh giá tính hợp lệ của các chứng nhận theo thời gian. Các thông tin về tên chứng nhận (*CertName*) và tổ chức cấp (*Issuer*) giúp xác định uy tín, nguồn gốc và mức độ đáng tin cậy của chứng nhận, từ đó củng cố niềm tin của người tiêu dùng và nâng cao chất lượng dữ liệu quản lý. Trường *CertDescription* cung cấp thông tin chi tiết về tiêu chuẩn, phạm vi áp dụng hoặc nội dung chứng nhận, giúp người dùng và nhân viên nghiệp vụ hiểu rõ giá trị và mục đích của chứng nhận trước khi quyết định áp dụng vào sản phẩm.

- **CertID**

- Mã định danh duy nhất của chứng nhận trong hệ thống.

 CertID int	* CertName varchar(100)	CertDescription text	Issuer varchar(255)
1	Chứng nhận middle_0	Tandem aspicio articulus admi	Streich LLC
2	Chứng nhận lyre_1	Suffragium dens id tergum su	Metz - Powlowski
3	Chứng nhận bowling_2	Bellicus vomica thermae cubic	Weimann, Kirlin and D'Amore
4	Chứng nhận cheese_3	Accusantium tumultus coaegr	Ebert, Bayer and Grant
5	Chứng nhận utilization_4	Denuo curia conturbo volup pl	O'Conner, Goyette and Jenkin
6	Chứng nhận hutch_5	Contego sublime turbo.	Marks, Wintheiser and Frami
7	Chứng nhận contrail_6	Vilitas aegrus tergo deleo cim	Watsica and Sons
8	Chứng nhận hope_7	Appello suffragium approbo c	Greenfelder LLC
9	Chứng nhận shipper_8	Suggero optio ago sit totam c	VonRueden LLC
10	Chứng nhận swordfish_9	Caput color modi aspicio cont	Kovacek - Mills
11	Chứng nhận doubter_10	Tamquam tubineus asper volu	Marks - Streich
12	Chứng nhận mythology_11	Unde caritas claro virtus.	Cruickshank - Waters
13	Chứng nhận reservation_12	Surgo talis abundans audeo ve	Dooley LLC
14	Chứng nhận alb_13	Totidem adeptio quisquam aut	Russel - Adams
15	Chứng nhận hunger_14	Solus sapiente super.	Bashirian Group

Figure 2.4: Data mẫu bảng CERTIFICATIONS

- Ví dụ: CertID = 1, CertID = 15.
- Trường này đóng vai trò là **khóa chính** và được sử dụng để liên kết với bảng *Product_{certifications}*.
- *Không được trùng lặp và không được NULL.*

• CertName

- Tên hiển thị của chứng nhận.
- Ví dụ: Chứng nhận middle₀, *Chngnhnswordfish₉*.
- Giúp người dùng nhận diện chứng nhận và liên kết dễ dàng với sản phẩm.
- Không được NULL; tên nên ngắn gọn, rõ ràng và dễ đọc.

• CertDescription

- Mô tả chi tiết về chứng nhận, bao gồm tiêu chuẩn, phạm vi áp dụng hoặc thông tin quan trọng liên quan.

- Giúp người dùng và staff hiểu rõ ý nghĩa của chứng nhận và mức độ chuẩn hóa của sản phẩm.
- Có thể NULL nếu chứng nhận chưa có mô tả chi tiết.

- **Issuer**

- Tổ chức hoặc công ty cấp chứng nhận.
- Ví dụ: Streich LLC, Kovacek - Mills.
- Giúp xác thực nguồn gốc và uy tín của chứng nhận.
- Không được NULL; tên tổ chức nên chính xác để tránh nhầm lẫn.

- **Lưu ý về ràng buộc và tính toàn vẹn dữ liệu:**

- CertID phải duy nhất và tồn tại để liên kết với bảng *Product_Certifications*.
- CertName và Issuer không được NULL để đảm bảo thông tin chứng nhận đầy đủ và rõ ràng.
- CertDescription có thể NULL nhưng nếu bỏ trống sẽ giảm mức độ minh bạch và chi tiết của chứng nhận.
- Hệ thống cần kiểm tra các ràng buộc khi thêm, sửa, xóa chứng nhận để đảm bảo toàn vẹn dữ liệu.

Về ý nghĩa tổng thể, bảng *Certifications* không chỉ lưu trữ dữ liệu mà còn là nền tảng cho việc quản lý, kiểm soát và tra cứu chứng nhận sản phẩm. Bảng này đóng vai trò quan trọng trong việc đảm bảo rằng chỉ những chứng nhận hợp lệ, đầy đủ thông tin mới được gán cho sản phẩm, nâng cao tính minh bạch, chính xác và chuyên nghiệp của hệ thống quản lý sản phẩm. Đồng thời, việc lưu trữ và tổ chức thông tin chứng nhận hợp lý còn tạo điều kiện thuận lợi cho việc mở rộng hệ thống, thêm các loại chứng nhận mới hoặc tích hợp với các quy trình nghiệp vụ khác trong tương lai.

2.1.5 ProductImages

Bảng ProductImages được thiết kế để quản lý dữ liệu đa phương tiện gắn liền với từng sản phẩm trong hệ thống, đóng vai trò trực quan hóa thông tin khi người dùng truy xuất sản phẩm thông qua website hoặc mã QR. Mỗi bản ghi đại diện cho một hình ảnh của một sản phẩm cụ thể, được liên kết đảm bảo tính toàn vẹn và nhất quán dữ liệu. Mỗi bản ghi tương ứng với một hình ảnh của một sản phẩm, được liên kết thông qua ProductID,

nhờ đó một sản phẩm có thể có nhiều hình ảnh khác nhau như ảnh tổng thể, ảnh cận cảnh hay ảnh bao bì.

12	Image int	* Product int	* ImageUrl text	MediaType enum('image')	ImageData text
4		5	https://picsum.photos/seed/Dl	image	(NULL)
5		6	https://picsum.photos/seed/Rl	image	(NULL)
6		7	https://picsum.photos/seed/Bl	image	(NULL)
7		8	https://picsum.photos/seed/Jd	image	(NULL)
8		9	https://picsum.photos/seed/8\	image	(NULL)
10		11	https://picsum.photos/seed/rn	image	(NULL)
11		12	https://picsum.photos/seed/Oi	image	(NULL)
12		13	https://picsum.photos/seed/9i	image	(NULL)
13		14	https://picsum.photos/seed/gc	image	(NULL)
14		15	https://picsum.photos/seed/Gt	image	(NULL)
15		16	https://picsum.photos/seed/Aé	image	(NULL)
16		17	https://picsum.photos/seed/d\	image	(NULL)
17		18	https://picsum.photos/seed/N\	image	(NULL)
19		20	https://picsum.photos/seed/7\	image	(NULL)
20		21	https://picsum.photos/seed/bc	image	(NULL)

Figure 2.5: Data mẫu bảng PRODUCT IMAGES

Cụ thể, ImageID là mã định danh duy nhất cho từng hình ảnh. ProductID mang giá trị là mã của sản phẩm đã tồn tại trong bảng Products, ví dụ như ProductID = 5 hoặc ProductID = 12, giúp xác định hình ảnh này thuộc về sản phẩm nào. Trường ImageUrl chứa đường dẫn đến hình ảnh, chẳng hạn <https://picsum.photos/seed/Dl>, cho phép hệ thống tải và hiển thị ảnh khi người dùng quét mã QR hoặc xem trên website. MediaType hiện có giá trị là image, dùng để phân biệt loại nội dung, đồng thời mở đường cho việc mở rộng trong tương lai như video hoặc document. Trường ImageData có thể để trống (NULL) hoặc dùng để lưu dữ liệu ảnh ở dạng mã hóa khi cần, nhưng trong hệ thống hiện tại được bỏ trống để tối ưu dung lượng lưu trữ.

Về ý nghĩa, bảng ProductImages giúp tăng tính trực quan và độ tin cậy của thông tin sản phẩm. Người dùng không chỉ đọc mô tả mà còn có thể nhìn thấy hình ảnh thực tế, từ đó dễ dàng đánh giá và nhận diện sản phẩm. Đồng thời, cách tổ chức dữ liệu theo

đường dẫn hình ảnh cũng giúp hệ thống vận hành nhẹ, dễ bảo trì và thuận tiện cho việc mở rộng về sau.

2.1.6 ApprovalRequests

Bảng Request được thiết kế để ghi nhận tất cả các yêu cầu thay đổi dữ liệu sản phẩm trong hệ thống. Nó bao gồm các thao tác chính như tạo mới sản phẩm (create), cập nhật thông tin sản phẩm (update) hoặc xóa sản phẩm (delete). Mỗi dòng dữ liệu trong bảng tương ứng với một yêu cầu riêng lẻ, giúp hệ thống và đội ngũ quản lý theo dõi lịch sử thao tác, kiểm soát quá trình phê duyệt, đồng thời giảm thiểu sai sót dữ liệu.

Ngoài ra, bảng còn lưu trữ các thông tin bổ sung để giám sát quy trình phê duyệt: ai là người gửi yêu cầu, ai là người đánh giá, trạng thái cuối cùng của yêu cầu, và các ghi chú liên quan. Điều này không chỉ giúp đảm bảo tính minh bạch và trách nhiệm trong quản lý dữ liệu, mà còn hỗ trợ trong việc truy xuất nguồn gốc quyết định khi có sự cố xảy ra hoặc khi cần rà soát các thay đổi trong hệ thống.

RequestId int	ProductId int	RequestType enum('create','update')	RequestedBy int	ReviewedBy int	Status enum('pending','approved','rejected')	Notes text	CreatedAt timestamp	ReviewedAt timestamp
1	20	create	2	1	approved	Adicio cura adipisci.	2025-12-04 10:56:59	2025-12-05 04:51:36
2	39	update	2	1	rejected	Cupressus ager cauda vesica	2025-12-04 10:56:59	2025-12-04 10:56:59
3	12	create	2	1	rejected	Tepesco benigne ars tabernus	2025-12-04 10:56:59	2025-12-04 10:56:59
4	29	delete	1	1	rejected	Pauci virtus cruciamentum am	2025-12-04 10:56:59	2025-12-04 10:56:59
5	20	create	1	1	rejected	Acidus aqua celer spero crude	2025-12-04 10:56:59	2025-12-04 10:56:59
6	21	delete	3	1	approved	Sub thorax aequus arto atton	2025-12-04 10:56:59	2025-12-04 10:56:59
7	(NULL)	create	2	1	rejected	Malores barba dolore tener.	2025-12-04 10:57:00	2025-12-04 10:57:00
8	31	create	1	1	rejected	Aperiam cruciamentum condic	2025-12-04 10:57:00	2025-12-04 10:57:00
9	28	create	3	1	rejected	Pecus sunt vorax provident illi	2025-12-04 10:57:00	2025-12-04 10:57:00
10	31	delete	2	1	approved	Ex delinquo vito tunc bibo.	2025-12-04 10:57:00	2025-12-04 10:57:00
11	(NULL)	create	3	1	rejected	Commodi porro comprehendo	2025-12-04 10:57:00	2025-12-04 10:57:00
12	(NULL)	update	2	1	rejected	Cotidie vulariter natus dedico	2025-12-04 10:57:00	2025-12-04 10:57:00
13	15	create	2	1	approved	Cito certus cibo somniculosus	2025-12-04 10:57:00	2025-12-04 10:57:00
14	30	update	1	1	approved	Depopulo nulla adamo.	2025-12-04 10:57:00	2025-12-04 10:57:00
15	(NULL)	delete	2	1	approved	Umerus somnus cruentus spe	2025-12-04 10:57:00	2025-12-04 10:57:00

Figure 2.6: Data mẫu bảng APPROVAL REQUESTS

- **RequestId:** Mã định danh duy nhất của yêu cầu.
 - Ví dụ: 1, 10, 15.
 - Đây là khóa chính, bắt buộc và không được trùng lặp.
 - Dùng để truy xuất nhanh từng yêu cầu trong báo cáo, log hoặc khi cần tra cứu chi tiết xử lý.
- **ProductId:** Mã sản phẩm liên quan đến yêu cầu.

- Ví dụ: 12, 20, 31; giá trị có thể NULL nếu yêu cầu tạo mới sản phẩm.
- Nếu ProductId không tồn tại trong bảng Products (ví dụ sản phẩm đã bị xóa), yêu cầu sẽ bị từ chối hoặc cảnh báo lỗi.
- Trường này giúp hệ thống biết yêu cầu áp dụng cho sản phẩm nào, để khi phê duyệt, dữ liệu sản phẩm được cập nhật chính xác.
- **RequestType:** Loại thao tác được yêu cầu.
 - Các giá trị phổ biến: **create** (tạo mới), **update** (cập nhật), **delete** (xóa).
 - Ví dụ: một yêu cầu **create** cho phép thêm sản phẩm mới vào hệ thống, **update** thay đổi thông tin như giá hoặc mô tả, **delete** gỡ sản phẩm không còn bán.
 - Chỉ nhận các giá trị hợp lệ; nhập sai loại thao tác sẽ gây lỗi phê duyệt.
- **RequestedBy:** Mã nhân viên hoặc hệ thống gửi yêu cầu.
 - Ví dụ: 1, 2, 3.
 - Hỗ trợ theo dõi ai là người chịu trách nhiệm tạo yêu cầu, phục vụ báo cáo trách nhiệm và tra cứu lịch sử thao tác.
- **ReviewedBy:** Mã nhân viên phê duyệt hoặc từ chối yêu cầu.
 - Ví dụ: 1, 2, 3.
 - Giúp xác định ai chịu trách nhiệm đánh giá và đảm bảo mọi thay đổi đều được kiểm soát.
- **Status:** Kết quả phê duyệt của yêu cầu.
 - Các giá trị: **approved** (được chấp thuận), **rejected** (bị từ chối).
 - Ví dụ: nếu yêu cầu **create** được **approved**, sản phẩm mới sẽ được thêm vào bảng Products; nếu **rejected**, không có thay đổi nào xảy ra.
- **Notes:** Ghi chú hoặc lý do từ người gửi hoặc người đánh giá.
 - Thường dùng để giải thích lý do từ chối, hướng dẫn chỉnh sửa hoặc các lưu ý nghiệp vụ.
 - Có thể để trống nếu không cần bổ sung chi tiết.
- **CreatedAt:** Thời điểm yêu cầu được tạo trong hệ thống.



- Ví dụ: 2025-12-04 10:56:59.
- Quan trọng để theo dõi thời gian xử lý, đánh giá tốc độ phản hồi của bộ phận phê duyệt.
- **ReviewedAt:** Thời điểm yêu cầu được đánh giá/phê duyệt.
 - Ví dụ: 2025-12-05 04:51:36.
 - Dùng để tính thời gian xử lý trung bình và giám sát quy trình nội bộ.

Bảng Request không chỉ là nơi lưu trữ dữ liệu, mà còn đóng vai trò như một bản ghi nhật ký hoạt động quan trọng, phục vụ cho quản trị nội bộ, bảo mật dữ liệu và đánh giá hiệu quả quy trình vận hành của hệ thống quản lý sản phẩm.

2.1.7 Users

Bảng *Users* được sử dụng để quản lý thông tin người dùng trong hệ thống, phục vụ cơ chế xác thực và phân quyền truy cập. Mỗi dòng dữ liệu tương ứng với một người dùng duy nhất, được định danh bằng **UserID**, đóng vai trò **khóa chính** của bảng. Bảng này giúp hệ thống phân biệt quyền hạn giữa các nhóm người dùng, đồng thời ghi nhận các thông tin liên hệ và đăng nhập để đảm bảo an toàn và minh bạch trong quản lý dữ liệu.

UserID	* Username	* Password	* Email	* Role
int	varchar(255)	varchar(255)	varchar(255)	enum('admin','staff')
1	HuyLD	\$2b\$10\$Tg3Jry5YVOD4v5cPl	huy2005luong@gmail.com	admin
2	NguyenTDK	\$2b\$10\$VQyr9FGoMjZZqXbB	nguyentdk@gmail.com	admin
3	HaoLy	\$2b\$10\$T16bx.3E.lvgA2ELUv	daitengu0613@gmail.com	staff

Figure 2.7: Data mẫu bảng USER

- **UserID**
 - Mã định danh duy nhất của người dùng.
 - Ví dụ: 1, 2, 3.
 - Trường này không được trùng lặp hoặc để trống; đồng thời được sử dụng làm khóa chính để liên kết với các bảng khác nếu cần.
- **Username**



- Tên đăng nhập của người dùng, dùng để xác thực khi đăng nhập vào hệ thống.
- Ví dụ: *HuyLD*, *NguyenTDK*, *HaoLy*.
- Phải duy nhất trong hệ thống, không được để trống, giúp ngăn trùng lặp tài khoản và đảm bảo an toàn khi đăng nhập.

- **Password**

- Mật khẩu đăng nhập, được lưu trữ dưới dạng đã mã hóa (hashed) để bảo mật.
- Ví dụ: *\$2b\$10\$Tg3Jry5YVOD4v5cPl*.
- Trường này không được để trống; hệ thống sử dụng giá trị mã hóa để so sánh khi người dùng đăng nhập, đảm bảo rằng mật khẩu thực tế không bị lộ.

- **Email**

- Địa chỉ email liên hệ của người dùng, phục vụ thông báo, hỗ trợ xác thực và quản lý tài khoản.
- Ví dụ: *huy2005luong@gmail.com*, *nguyentdk@gmail.com*, *daitengu0613@gmail.com*.
- Phải là địa chỉ hợp lệ và không được để trống, giúp hệ thống gửi thông báo hoặc khôi phục mật khẩu khi cần thiết.

- **Role**

- Vai trò của người dùng trong hệ thống, quyết định quyền hạn và khả năng thực hiện các thao tác.
- Các giá trị hợp lệ bao gồm *admin* và *staff*.
- Admin có quyền quản lý toàn bộ hệ thống, bao gồm quản lý người dùng, dữ liệu sản phẩm, mã QR và phân quyền. Staff chỉ được thực hiện các thao tác nghiệp vụ giới hạn như thêm, cập nhật thông tin sản phẩm, tạo và quản lý mã QR, nhưng không có quyền quản trị.

Lưu ý về ràng buộc và tính toàn vẹn dữ liệu:

- **UserID**: Không được trùng lặp, không để trống, dùng làm khóa chính.
- **Username**: Phải duy nhất và không để trống, đảm bảo xác thực chính xác.
- **Password**: Phải được mã hóa, không để trống.



- **Email:** Phải hợp lệ, không để trống.
- **Role:** Chỉ nhận giá trị *admin* hoặc *staff*; các giá trị khác đều không hợp lệ.

Về ý nghĩa tổng thể, bảng *Users* đảm bảo hệ thống vận hành an toàn, kiểm soát quyền hạn và phân quyền hợp lý. Việc quản lý đầy đủ thông tin người dùng không chỉ hỗ trợ xác thực, phân quyền mà còn ghi nhận lịch sử thao tác, giúp hệ thống minh bạch và có khả năng mở rộng các chức năng bảo mật và quản lý người dùng trong tương lai.

3 Xây dựng ứng dụng

3.1 Mô tả kiến trúc hệ thống

Hệ thống xây dựng theo cấu trúc đơn giản gồm phần mã nguồn của backend và frontend. Trong đó backend chịu trách nhiệm chính về các chức năng của hệ thống. Dưới đây là hệ thống thư mục và chức năng của backend

```
backend/
|-- endpoints/ ..... API route definitions
|   |-- admin.js ..... Admin endpoints
|   |-- approval.js ..... Approval workflow APIs
|   |-- crud.js ..... Generic CRUD operations
|   |-- farmnCert.js ..... Farm & certification APIs
|   |-- login.js ..... Authentication endpoints
|   |-- manager.js ..... Manager role APIs
|   |-- password.js ..... Password management APIs
|   |-- public.js ..... Public-access APIs
|   |-- qrmanager.js ..... QR code management APIs
|   |-- supporter.js ..... Supporter role APIs
|
|-- services/ ..... Core backend services
|   |-- authMiddleware.js ..... Authentication & middleware
|   |-- backupCloudinary.js ..... Cloudinary backup service
|   |-- db.js ..... Database connection handler
|   |-- debugServer.js ..... Server debugging utilities
|   |-- Indexes.js ..... Database indexes definition
|   |-- mailer.js ..... Email sending service
|
|-- uploads/ ..... Uploaded files storage
|
|-- .env ..... Environment configuration
|-- ca.pem ..... SSL certificate
|-- package.json ..... Project metadata & scripts
|-- package-lock.json ..... Dependency lock file
|-- server.js ..... Express server entry point
```



Frontend được xây dựng bằng ReactJS, chịu trách nhiệm hiển thị giao diện và tương tác trực tiếp với người dùng. Các chức năng chính bao gồm đăng nhập, quản lý sản phẩm, quét mã QR và hiển thị thông tin truy xuất nguồn gốc. Cấu trúc frontend được tổ chức theo các component, trong đó: Các file trong thư mục **src/** đại diện cho các màn hình và chức năng chính của hệ thống. File **api.js** đóng vai trò trung gian giao tiếp với backend thông qua các API REST. Các file CSS được tách riêng nhằm đảm bảo giao diện nhất quán và dễ chỉnh sửa.

```
frontend/  
|-- src/  
|   |-- api.js  
|   |-- App.js  
|   |-- Approvals.js  
|   |-- Dashboard.js / Dashboard.css  
|   |-- ForgotPassword.js / ForgotPassword.css  
|   |-- Login.js / Login.css  
|   |-- ProductForm.js  
|   |-- ProductManagement.js  
|   |-- ProductView.js / ProductView.css  
|   |-- RequestList.js / RequestList.css  
|   |-- ResetPassword.js / ResetPassword.css  
|   |-- Scanner.js / Scanner.css  
|   |-- Sidebar.js  
|   |-- UserManagement.js  
|   |-- Service.js  
|   |-- index.js  
|  
|-- public/
```

3.2 Chức năng quét QR

- **Mục tiêu của chức năng**

Người tiêu dùng ngày nay không chỉ quan tâm đến giá cả hay hình thức sản phẩm, mà còn có nhu cầu kiểm chứng nguồn gốc sản xuất, quy trình canh tác và các chứng nhận chất lượng đi kèm. Chức năng Quét mã QR được xây dựng như cổng truy xuất thông tin công khai của hệ thống, cho phép người tiêu dùng, nhân viên (staff) và các bên liên quan nhanh chóng tiếp cận dữ liệu nguồn gốc nông sản thông qua mã QR được in trên bao bì sản phẩm. Mục tiêu cốt lõi của chức năng này là đảm bảo tính minh bạch – chính xác – tức thời trong việc cung cấp thông tin sản phẩm, đồng thời giảm thiểu sự phụ thuộc vào các kênh tra cứu thủ công.

Về mặt kỹ thuật, đây là một chức năng truy vấn dữ liệu chỉ đọc (read-only), được tối ưu để phục vụ số lượng lớn người dùng đồng thời, với thời gian phản hồi thấp và khả năng xử lý các trường hợp dữ liệu không hợp lệ.

- **Luồng hoạt động người dùng**

Chức năng quét mã QR được thiết kế theo hướng tối giản thao tác nhưng vẫn đảm bảo cung cấp đầy đủ thông tin cần thiết cho người dùng. Toàn bộ quá trình tương tác được xây dựng xoay quanh nguyên tắc: người dùng không cần hiểu về hệ thống bên trong nhưng vẫn có thể truy xuất thông tin một cách nhanh chóng, chính xác và đáng tin cậy.

Từ góc độ kỹ thuật, luồng hoạt động này đảm bảo dữ liệu chỉ được truy vấn trong phạm vi cho phép, ở chế độ chỉ đọc (read-only), đồng thời không làm lộ các chức năng quản trị nội bộ của hệ thống.

Luồng hoạt động tổng quát của chức năng quét mã QR bao gồm các bước sau:

1. Người dùng truy cập vào trang quét mã QR của hệ thống.
2. Người dùng lựa chọn một trong hai phương thức quét:
 - Sử dụng camera của thiết bị để quét trực tiếp mã QR.
 - Tải lên hình ảnh có chứa mã QR từ bộ nhớ thiết bị.
3. Hệ thống tiến hành giải mã nội dung mã QR và gửi yêu cầu truy vấn dữ liệu đến backend thông qua API công khai.
4. Backend thực hiện xác thực tính hợp lệ của mã QR và kiểm tra trạng thái của sản phẩm tương ứng trong cơ sở dữ liệu.

5. Frontend hiển thị kết quả truy xuất cho người dùng:

- Hiển thị hộp thoại thông tin sản phẩm nếu quá trình truy xuất thành công.
- Hiển thị thông báo lỗi trong trường hợp mã QR không hợp lệ hoặc sản phẩm không tồn tại trong hệ thống.

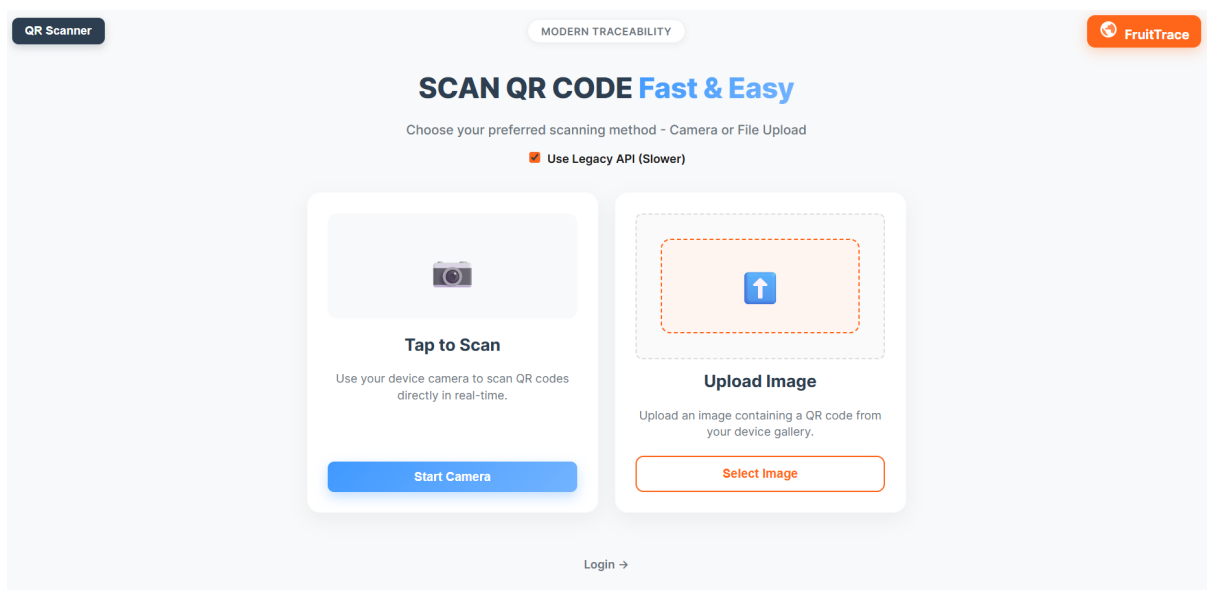


Figure 3.1: Giao diện quét mã QR

Giao diện quét mã QR được thiết kế theo hướng thân thiện và dễ tiếp cận, hướng đến nhiều nhóm người dùng khác nhau như người tiêu dùng cuối, nhân viên bán hàng, hoặc các bên kiểm tra, giám sát chất lượng. Ngay khi truy cập vào trang quét mã, người dùng được dẫn dắt trực tiếp đến khu vực thao tác chính với thông điệp rõ ràng, nhấn mạnh tính nhanh chóng và tiện lợi của việc truy xuất thông tin.

Phương thức quét bằng Camera

Đối với kịch bản sử dụng trong đời sống thực, chẳng hạn tại cửa hàng, siêu thị hoặc kho phân phối, người dùng thường cầm trực tiếp sản phẩm trên tay. Khi nhấn vào nút “Tap to Scan”, hệ thống yêu cầu quyền truy cập camera của thiết bị:

Sau khi được cấp quyền, camera được kích hoạt và hiển thị khung quét trực quan, giúp người dùng dễ dàng căn chỉnh mã QR:

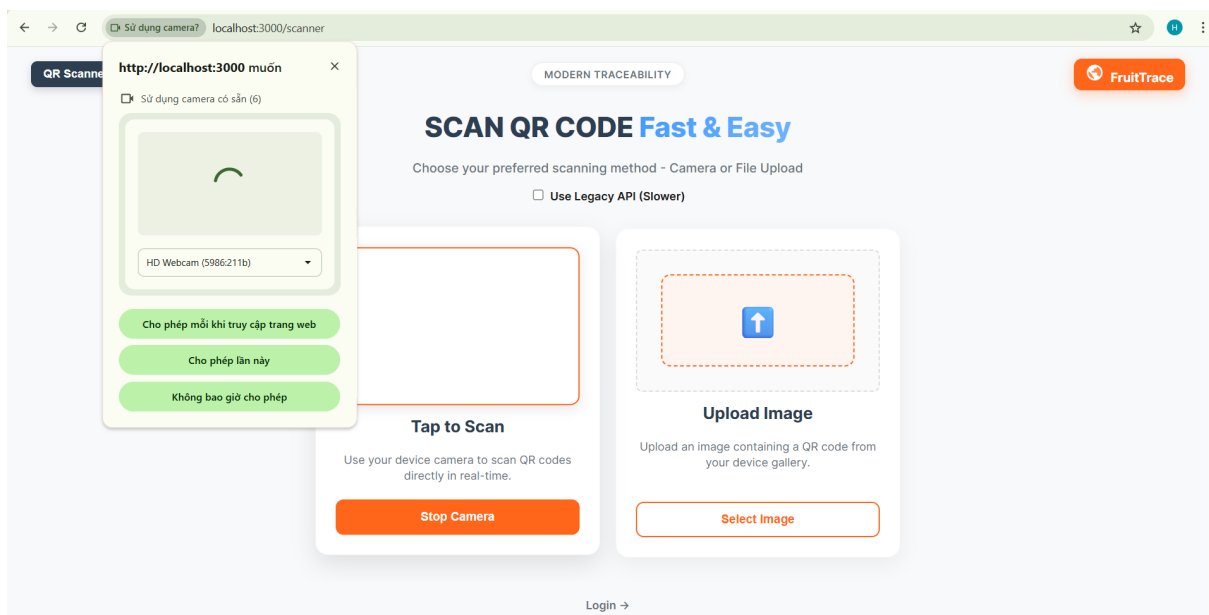


Figure 3.2: Giao diện quét QR bằng camera

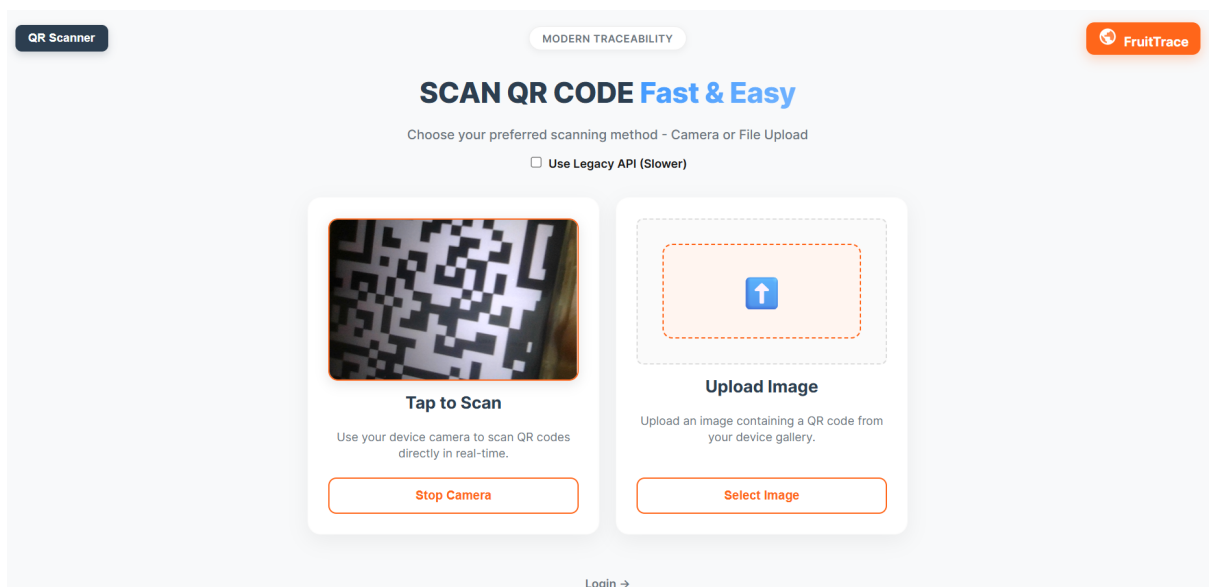


Figure 3.3: Giao diện quét QR bằng camera

Ngay khi mã QR nằm trong vùng nhận diện, hệ thống tự động giải mã và xử lý mà không yêu cầu thêm thao tác xác nhận. Điều này giúp giảm tối đa thời gian chờ và tránh gây gián đoạn trải nghiệm. Về mặt ý nghĩa sử dụng, cơ chế quét trực tiếp này giúp người dùng nhanh chóng kiểm tra nguồn gốc sản phẩm ngay tại thời điểm mua hàng, từ đó nâng cao mức độ tin tưởng và giảm rủi ro sử dụng sản phẩm không rõ

xuất xứ.

Phương thức quét bằng cách tải ảnh (Upload Image)

Trong nhiều tình huống khác, người dùng không trực tiếp cầm sản phẩm, ví dụ như kiểm tra thông tin từ ảnh chụp, tài liệu điện tử hoặc mã QR được gửi qua các kênh trực tuyến. Khi đó, người dùng có thể lựa chọn nút “Upload Image” để tải lên một hình ảnh chứa mã QR.

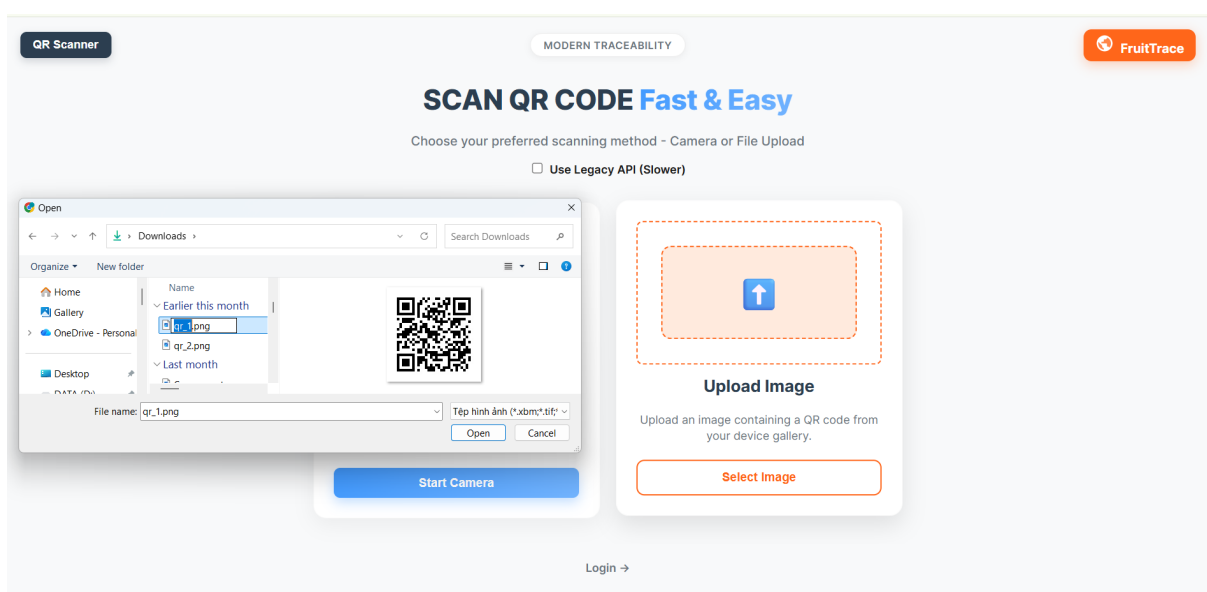


Figure 3.4: Giao diện tải lên ảnh chứa mã QR

Sau khi ảnh được tải lên, hệ thống tự động xử lý hình ảnh, trích xuất mã QR và thực hiện truy vấn dữ liệu tương tự như phương thức quét bằng camera. Việc hỗ trợ hình thức tải ảnh không chỉ giúp mở rộng phạm vi sử dụng của hệ thống mà còn phản ánh tính linh hoạt trong thiết kế, đáp ứng các kịch bản làm việc từ xa, lưu trữ hồ sơ hoặc kiểm tra hậu kỳ.

Sau khi dữ liệu mã QR được giải mã, hệ thống backend tiến hành kiểm tra tính hợp lệ của mã và truy vấn dữ liệu tương ứng trong cơ sở dữ liệu. Kết quả xử lý được phân thành hai trường hợp chính như sau:

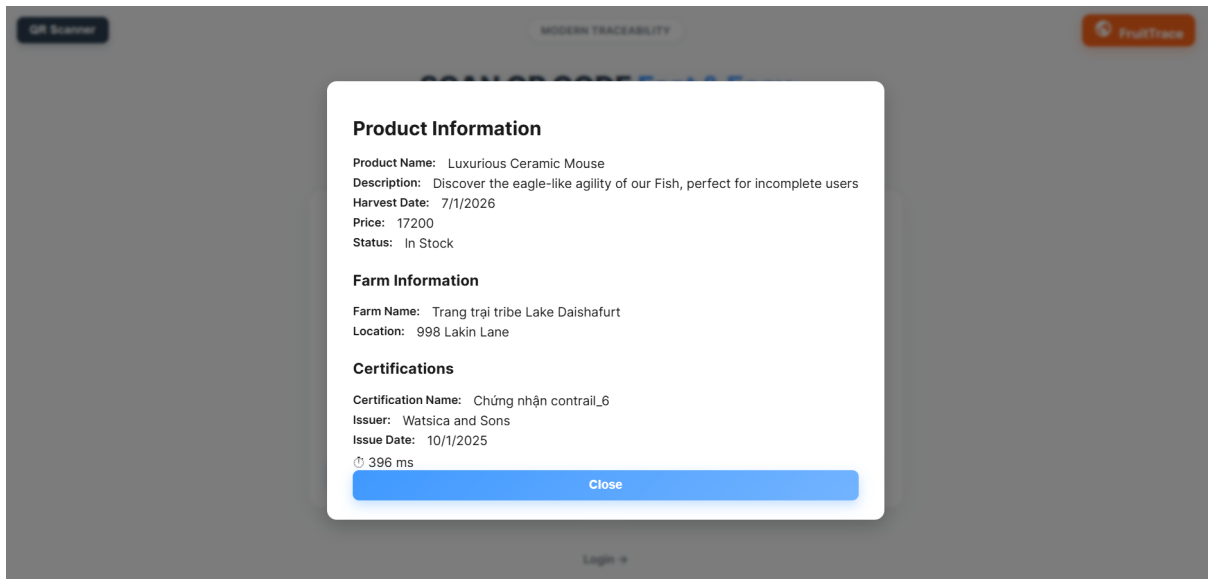


Figure 3.5: Giao diện hiển thị sản phẩm thành công

Trường hợp truy xuất thành công

Khi mã QR hợp lệ và sản phẩm tồn tại trong hệ thống, frontend hiển thị hộp thoại *Product Information*, trong đó bao gồm các nhóm thông tin chi tiết sau:

Nhóm thông tin sản phẩm

- **Product Name:** Tên sản phẩm (ví dụ: *Luxurious Ceramic Mouse*), được sử dụng để nhận diện sản phẩm trong hệ thống.
- **Description:** Mô tả ngắn gọn về đặc điểm hoặc công dụng của sản phẩm, giúp người dùng hiểu rõ hơn về mặt hàng đang truy xuất.
- **Harvest Date:** Ngày thu hoạch, thể hiện mốc thời gian quan trọng trong chuỗi cung ứng nông sản.
- **Price:** Giá bán tham khảo của sản phẩm.
- **Status:** Trạng thái hiện tại của sản phẩm (ví dụ: *In Stock*), phản ánh tính hợp lệ và khả năng lưu hành trên thị trường.

Nhóm thông tin nông trại

- **Farm Name:** Tên nông trại sản xuất sản phẩm.



Product Information

Product Name: Luxurious Ceramic Mouse

Description: Discover the eagle-like agility of our Fish, perfect for incomplete users

Harvest Date: 7/1/2026

Price: 17200

Status: In Stock

Farm Information

Farm Name: Trang trại tribe Lake Daishafurt

Location: 998 Lakin Lane

Certifications

Certification Name: Chứng nhận contrail_6

Issuer: Watsica and Sons

Issue Date: 10/1/2025

🕒 396 ms

Close

Figure 3.6: Bảng hộp thoại Product Information

- **Location:** Địa chỉ nông trại.

Nhóm thông tin này đóng vai trò xác thực nguồn gốc sản xuất, là thành phần cốt lõi trong việc đảm bảo khả năng truy xuất nguồn gốc (*traceability*) của sản phẩm.

Nhóm thông tin chứng nhận

- **Certification Name:** Tên chứng nhận chất lượng.
- **Issuer:** Đơn vị cấp chứng nhận.
- **Issue Date:** Ngày cấp chứng nhận.

Các thông tin chứng nhận cho thấy sản phẩm đã được kiểm định và đáp ứng các tiêu chuẩn chất lượng nhất định, từ đó nâng cao mức độ tin cậy đối với người tiêu dùng.

Trường hợp truy xuất thất bại

Trong trường hợp mã QR không tồn tại, sản phẩm đã bị xóa hoặc không còn ở trạng thái hợp lệ, hệ thống hiển thị thông báo lỗi với nội dung: Product not found or has been deleted!

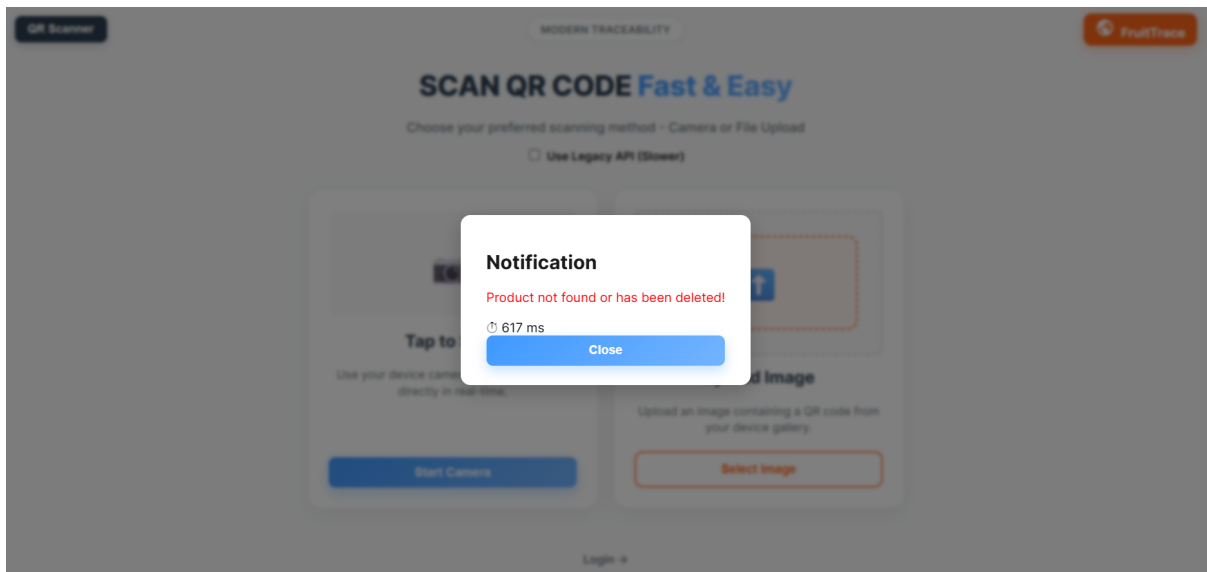


Figure 3.7: Product not found or has been deleted!

Thông báo này phản ánh cơ chế kiểm tra dữ liệu phía backend, đảm bảo chỉ những sản phẩm hợp lệ mới được phép truy xuất công khai, đồng thời tránh việc hiển thị các dữ liệu sai lệch hoặc không còn giá trị sử dụng.

3.3 Chức năng CRUD và Quy trình phê duyệt trong hệ thống FruitTrace

3.3.1 Quản lý Sản phẩm (Product Management)

Chức năng CRUD (Create – Read – Update – Delete) trong hệ thống FruitTrace không được thực hiện trực tiếp một cách tự do, mà được thiết kế gắn liền với quy trình kiểm soát và phê duyệt, nhằm đảm bảo tính minh bạch, an toàn dữ liệu và phân quyền rõ ràng giữa các vai trò người dùng. Cơ chế này đặc biệt phù hợp với các hệ thống quản lý sản phẩm và truy xuất nguồn gốc, nơi mọi thay đổi dữ liệu đều cần được theo dõi và xác thực.

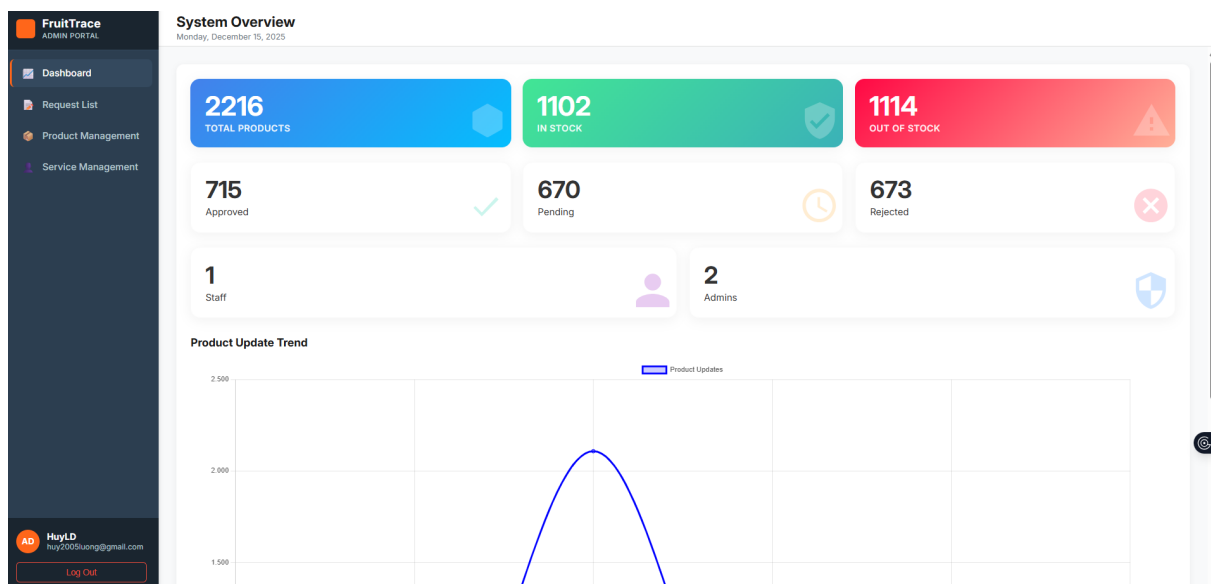


Figure 3.8: Giao diện Dashboard tổng quan của hệ thống FruitTrace

Giao diện này cung cấp cái nhìn tổng thể về tình trạng vận hành hệ thống thông qua các chỉ số thống kê chính như tổng số sản phẩm, số lượng sản phẩm còn hàng và hết hàng, cũng như trạng thái xử lý các yêu cầu thay đổi dữ liệu.

Bên cạnh đó, Dashboard còn hiển thị thông tin về số lượng yêu cầu đã được phê duyệt, đang chờ xử lý và bị từ chối, giúp quản trị viên nhanh chóng nắm bắt tiến độ xử lý nghiệp vụ. Thông tin về số lượng người dùng theo vai trò (Staff và Admin) cũng được tổng hợp nhằm hỗ trợ công tác quản lý và phân quyền.

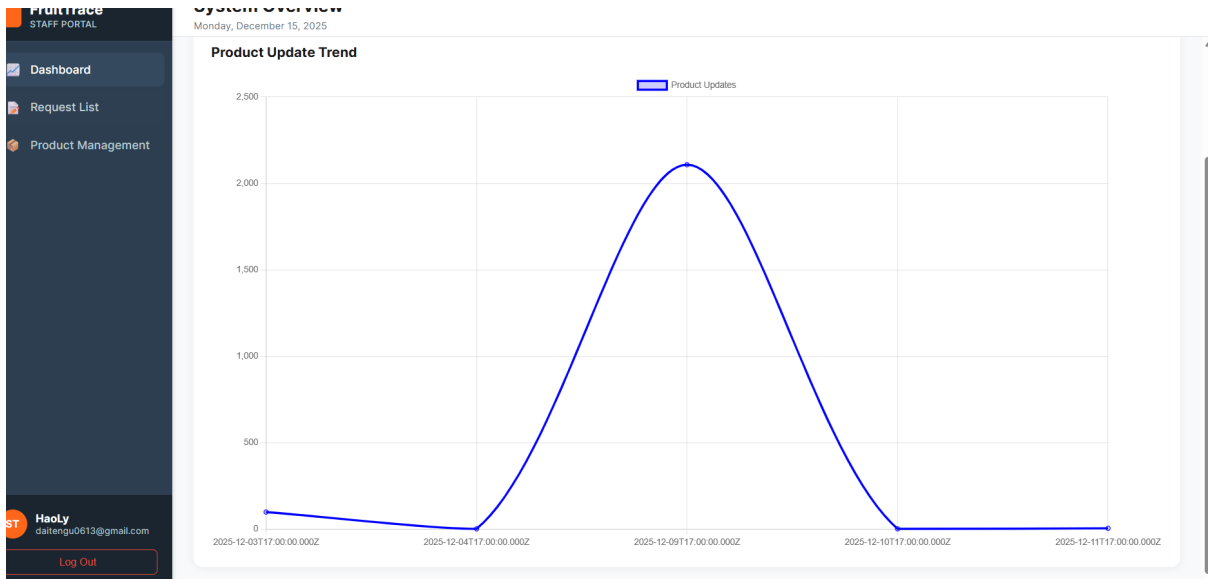
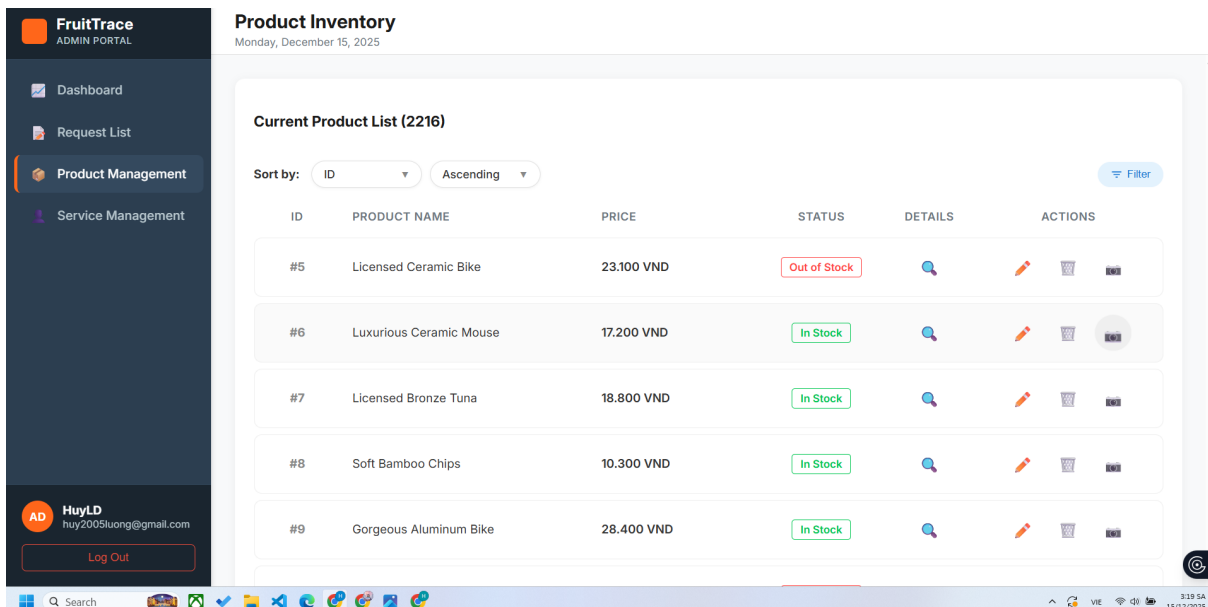


Figure 3.9: Giao diện Dashboard biểu đồ

Ngoài các chỉ số dạng thẻ (summary cards), giao diện còn tích hợp biểu đồ xu hướng cập nhật sản phẩm theo thời gian, qua đó phản ánh tần suất và mức độ thay đổi dữ liệu trong hệ thống.



The screenshot shows the 'FruitTrace ADMIN PORTAL' interface. The main section is titled 'Product Inventory' with the date 'Monday, December 15, 2025'. Below this is a 'Current Product List (2216)' table. The table has columns for ID, PRODUCT NAME, PRICE, STATUS, DETAILS, and ACTIONS. The table is sorted by ID in ascending order. The first five rows are visible, showing products with their respective prices and stock statuses.

ID	PRODUCT NAME	PRICE	STATUS	DETAILS	ACTIONS
#5	Licensed Ceramic Bike	23.100 VND	Out of Stock		
#6	Luxurious Ceramic Mouse	17.200 VND	In Stock		
#7	Licensed Bronze Tuna	18.800 VND	In Stock		
#8	Soft Bamboo Chips	10.300 VND	In Stock		
#9	Gorgeous Aluminum Bike	28.400 VND	In Stock		

Figure 3.10: Giao diện Product Management

Chức năng quản lý sản phẩm là thành phần cốt lõi của cổng *Admin Portal* trong hệ thống FruitTrace, cho phép người dùng thuộc các vai trò *Staff* và *Admin* theo

dối, quản lý và kiểm soát dữ liệu sản phẩm phục vụ cho mục tiêu truy xuất nguồn gốc. Toàn bộ các thao tác quản lý sản phẩm được thiết kế xoay quanh nguyên tắc kiểm soát thay đổi, đảm bảo dữ liệu luôn nhất quán và có khả năng truy vết.

Hệ thống hỗ trợ chức năng **Read** thông qua giao diện *Product Inventory*, nơi hiển thị danh sách toàn bộ sản phẩm hiện có trong hệ thống (tại thời điểm khảo sát là 2216 sản phẩm). Giao diện này cung cấp cái nhìn tổng quan về dữ liệu sản phẩm thông qua các trường thông tin cơ bản như mã sản phẩm, tên sản phẩm, giá bán và trạng thái tồn kho. Nhằm hỗ trợ truy vấn dữ liệu hiệu quả, hệ thống cho phép người dùng sắp xếp danh sách theo mã sản phẩm ở cả hai chiều tăng dần và giảm dần, đồng thời cung cấp chức năng lọc theo trạng thái tồn kho hoặc theo khoảng giá.

Filter Products



Status

☒ All Status ☐ In Stock ☐ Out Stock

Price Range (VND)

Min Price (e.g. 10000)

—

Max Price (e.g. 500000)

Reset

Apply Filter

Figure 3.11: Giao diện bộ lọc giúp tối ưu việc tra cứu

Bên cạnh việc xem danh sách tổng hợp, người dùng có thể truy cập vào thông tin chi tiết của từng sản phẩm thông qua chức năng xem chi tiết. Giao diện chi tiết sản phẩm cung cấp đầy đủ các dữ liệu liên quan đến truy xuất nguồn gốc, bao gồm tên



sản phẩm, giá bán, mô tả, thông tin trang trại, ngày thu hoạch, trạng thái hiện tại và các chứng nhận liên quan.

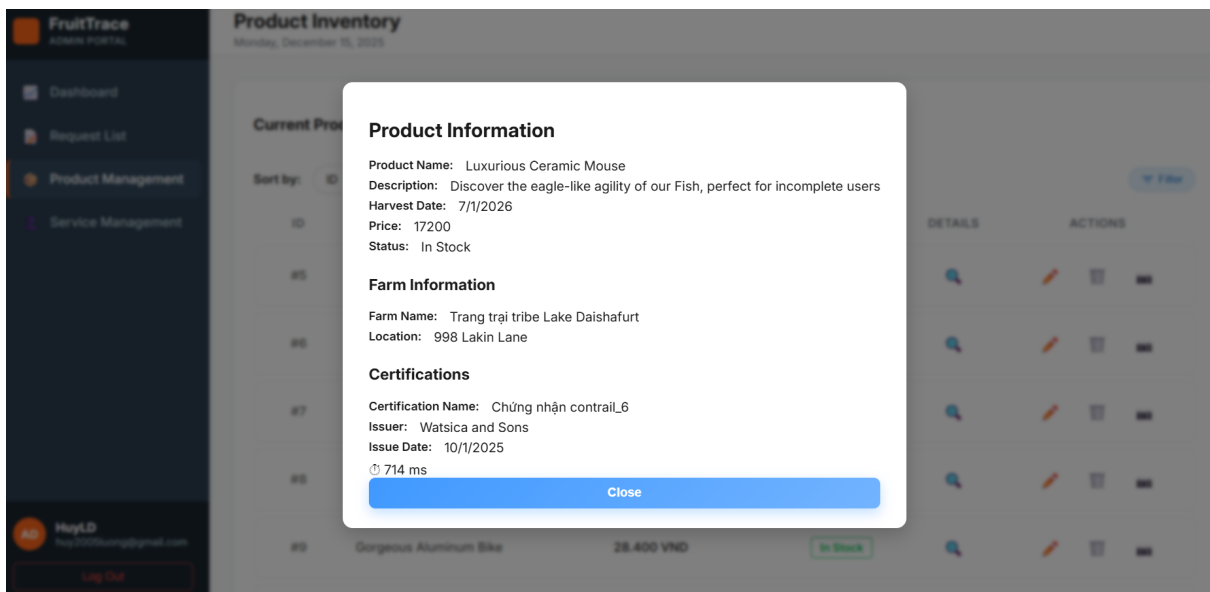


Figure 3.12: Giao diện hiển thị thông tin sản phẩm còn hàng IN STOCK

Notification

Product not found or has been deleted!

🕒 927 ms

Close

Figure 3.13: Giao diện hiển thị thông tin sản phẩm OUT OF STOCK

Ngoài ra, hệ thống còn hỗ trợ hiển thị và tải về mã QR tương ứng với từng sản phẩm, phục vụ cho việc tra cứu thông tin sản phẩm từ phía người tiêu dùng hoặc các bên liên quan.

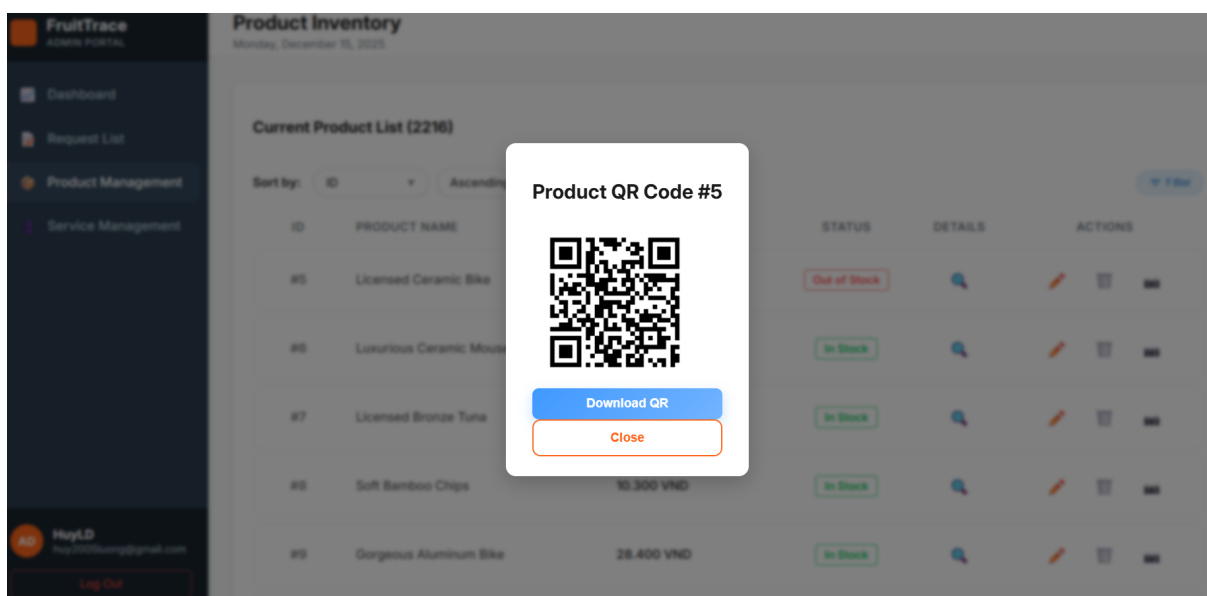


Figure 3.14: Giao diện hiển thị mã QR

Đối với các thao tác **Update** và **Delete**, hệ thống FruitTrace không cho phép chỉnh sửa hoặc xóa dữ liệu sản phẩm trực tiếp trên danh sách quản lý. Thay vào đó, hệ thống áp dụng một quy trình xét duyệt thông qua cơ chế tạo yêu cầu thay đổi. Khi người dùng thực hiện thao tác chỉnh sửa hoặc xóa sản phẩm, hệ thống sẽ ghi nhận hành động này dưới dạng một yêu cầu và chuyển yêu cầu đó vào danh sách *Request List* để chờ phê duyệt. Trong trường hợp cập nhật thông tin, giao diện chỉnh sửa sản phẩm cho phép người dùng nhập các dữ liệu mới như giá bán, trạng thái tồn kho, ngày trồng, ngày thu hoạch, thông tin trang trại và chứng nhận; các thay đổi này chỉ được áp dụng khi yêu cầu được quản trị viên chấp thuận.



Edit Product



Product Name *

Licensed Ceramic Bike

Price *

23100

Farm *

Green Farm Demo



Status *

Out of Stock



Planting Date *

01/10/2025



Harvest Date *

20/09/2026



Certification *

-- Select Certification --



Description *

Figure 3.15: Giao diện chỉnh sửa thông tin

Hệ thống cũng được trang bị cơ chế xử lý lỗi nhằm đảm bảo tính ổn định và nhất quán của dữ liệu. Ví dụ, khi người dùng cố gắng truy cập vào một sản phẩm không tồn tại hoặc đã bị xóa, hệ thống sẽ hiển thị thông báo lỗi tương ứng để ngăn chặn các thao tác không hợp lệ.



Cách tiếp cận này giúp hạn chế rủi ro sai sót trong quá trình quản lý dữ liệu, đồng thời đảm bảo mọi thay đổi đều được ghi nhận, kiểm soát và phê duyệt rõ ràng. Qua đó, hệ thống thiết lập một luồng làm việc minh bạch giữa *Staff* – người tạo yêu cầu thay đổi – và *Admin* – người có quyền phê duyệt và áp dụng các thay đổi lên dữ liệu chính thức.

3.3.2 Quản lý yêu cầu thay đổi (Request List)

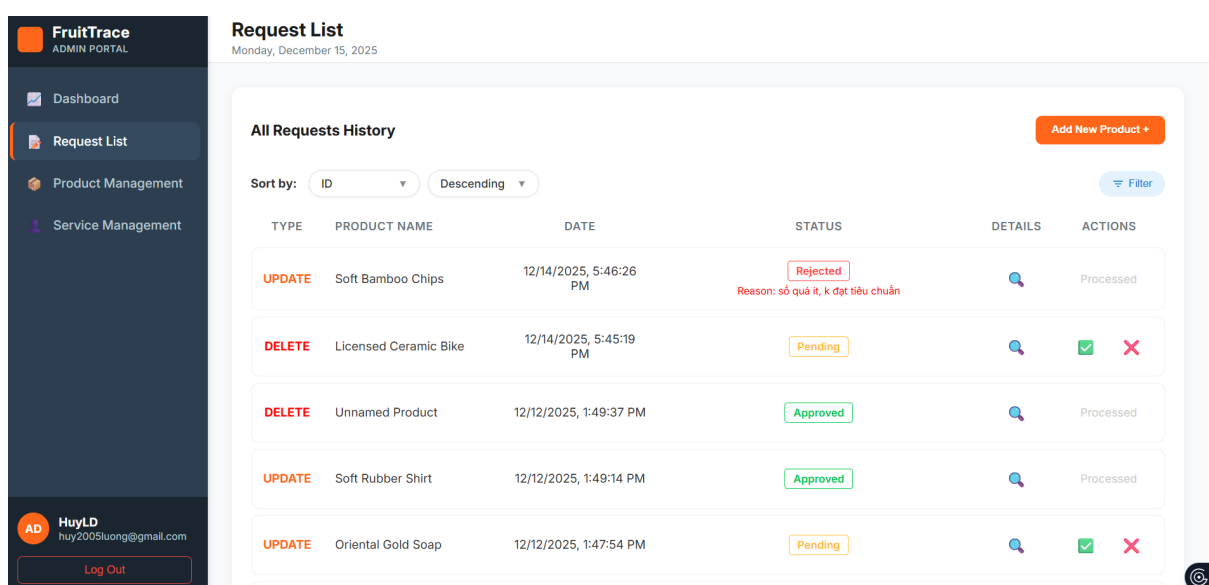


Figure 3.16: Giao diện quản lý yêu cầu

Chức năng Request List đóng vai trò trung tâm trong cơ chế kiểm soát dữ liệu của hệ thống FruitTrace. Thay vì cho phép thay đổi dữ liệu trực tiếp, hệ thống yêu cầu mọi thao tác liên quan đến việc tạo mới, chỉnh sửa hoặc xóa sản phẩm đều phải được thực hiện thông qua các yêu cầu xét duyệt. Cách tiếp cận này giúp đảm bảo dữ liệu truy xuất nguồn gốc luôn được kiểm soát chặt chẽ và có khả năng truy vết rõ ràng.



Add Product



Product Name *

Price *

Farm *

Status *

Planting Date *



Harvest Date *



Certification *

Description *

Figure 3.17: Giao diện thêm thông tin sản phẩm

Giao diện All Requests History hiển thị toàn bộ lịch sử các yêu cầu đã được tạo trong hệ thống. Cụ thể, hệ thống cho phép lọc yêu cầu theo loại thao tác, bao gồm tạo mới (Create), cập nhật (Update) và xóa (Delete), hoặc hiển thị toàn bộ yêu cầu khi cần theo dõi tổng quan. Bên cạnh đó, người dùng có thể lọc theo trạng thái xử



lý của yêu cầu, bao gồm Pending, Approved và Rejected. Việc kết hợp hai tiêu chí lọc này giúp Admin dễ dàng xác định các yêu cầu đang chờ phê duyệt hoặc rà soát lại các yêu cầu đã được xử lý trước đó.

Filter Requests



Request Type

☒ All ☐ Create ☐ Update ☐ Delete

Status

☒ All ☐ Pending ☐ Approved
☐ Rejected

Reset

Apply Filter

Figure 3.18

Các yêu cầu trong Request List được phân loại theo ba trạng thái chính. Trạng thái Pending thể hiện các yêu cầu đang chờ quản trị viên xem xét và chưa được áp dụng vào dữ liệu chính thức. Với các yêu cầu ở trạng thái này, Admin có quyền đưa ra quyết định phê duyệt hoặc từ chối thông qua các thao tác trực tiếp trên giao diện. Trạng thái Approved cho biết yêu cầu đã được chấp thuận và dữ liệu tương ứng đã được cập nhật vào hệ thống. Trong khi đó, trạng thái Rejected thể hiện các yêu cầu bị từ chối do không đáp ứng tiêu chuẩn hoặc yêu cầu nghiệp vụ, thường đi kèm với



lý do từ chối cụ thể.

Edit Product



Product Name *

Licensed Ceramic Bike

Price *

23100

Farm *

Green Farm Demo



Status *

Out of Stock



Planting Date *

01/10/2025



Harvest Date *

20/09/2026



Certification *

-- Select Certification --



Description *

Figure 3.19: Giao diện chỉnh sửa thông tin

Bên cạnh việc theo dõi trạng thái, hệ thống còn hỗ trợ các chức năng sắp xếp và lọc yêu cầu theo loại thao tác hoặc trạng thái xử lý. Điều này giúp quản trị viên dễ dàng quản lý số lượng lớn yêu cầu, đặc biệt trong các giai đoạn hệ thống có nhiều



thay đổi dữ liệu.

Khi truy cập vào chi tiết của một yêu cầu, quản trị viên có thể xem đầy đủ thông tin liên quan như thời gian tạo yêu cầu, nội dung thay đổi được đề xuất và trạng thái hiện tại. Đối với các yêu cầu bị từ chối, hệ thống hiển thị rõ lý do từ chối, ví dụ như dữ liệu không đạt tiêu chuẩn hoặc không phù hợp với quy định quản lý. Cơ chế này không chỉ hỗ trợ kiểm soát chất lượng dữ liệu mà còn giúp người tạo yêu cầu hiểu rõ nguyên nhân và điều chỉnh trong các lần gửi yêu cầu tiếp theo.

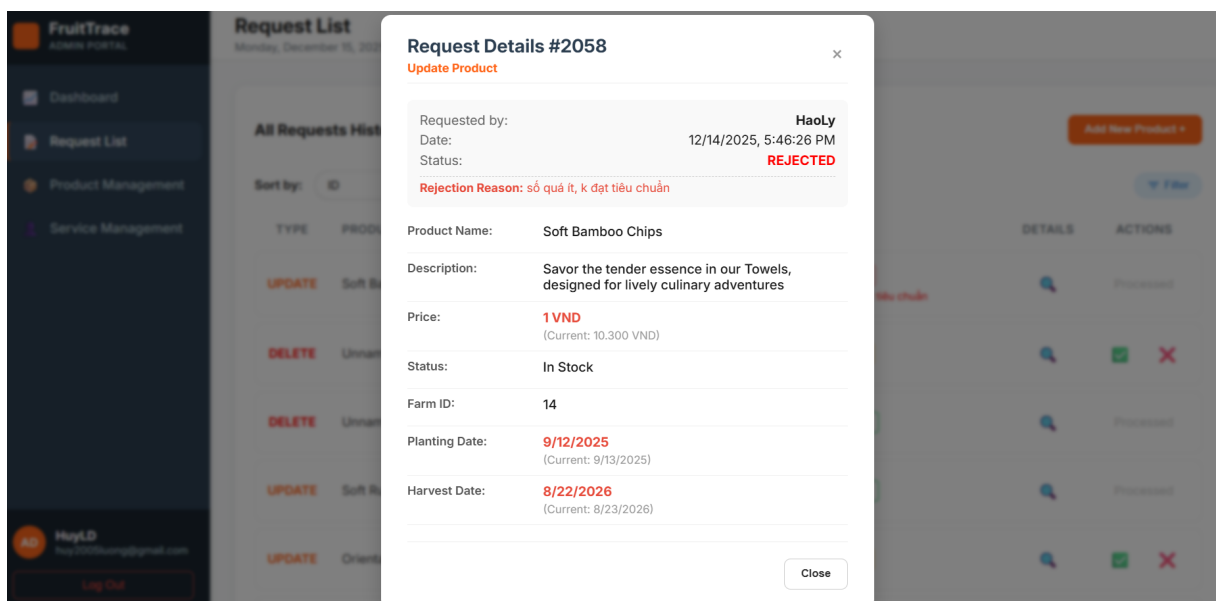


Figure 3.20

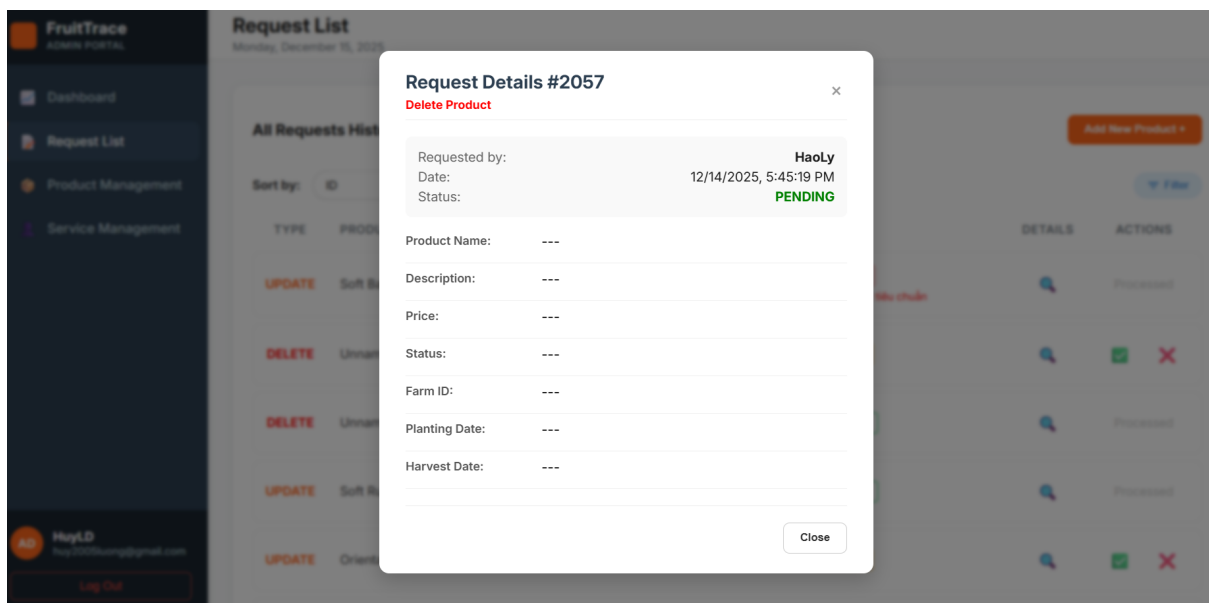


Figure 3.21

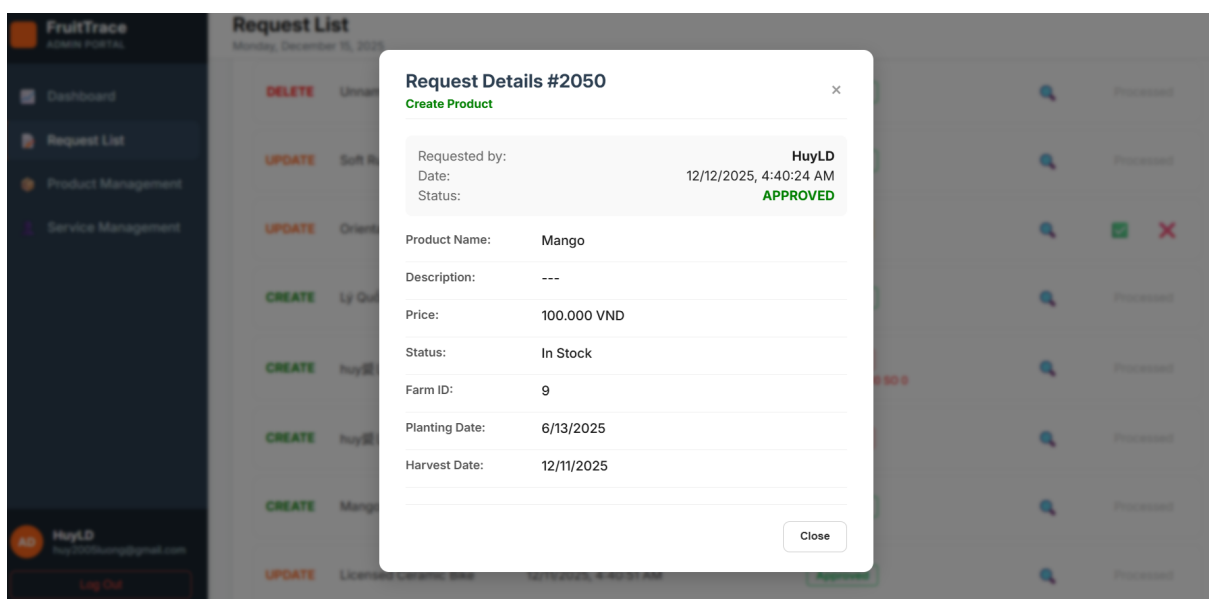


Figure 3.22

Thông qua Request List, hệ thống FruitTrace thiết lập một luồng làm việc minh bạch giữa người tạo yêu cầu (Staff) và người xét duyệt (Admin). Mọi thay đổi dữ liệu đều được ghi nhận, kiểm soát và phê duyệt trước khi áp dụng, góp phần nâng cao độ tin cậy và tính nhất quán của dữ liệu truy xuất nguồn gốc trong hệ thống.

3.3.3 Quản lý người dùng và phân quyền

Hệ thống FruitTrace cho thấy hệ thống đã được thiết kế với cơ chế phân quyền rõ ràng giữa các vai trò người dùng, bao gồm Staff và Admin. Việc phân tách vai trò này đóng vai trò quan trọng trong việc kiểm soát quyền truy cập cũng như đảm bảo an toàn cho dữ liệu trong toàn bộ hệ thống.

Hệ thống hỗ trợ các chức năng quản lý tài khoản cơ bản (CRUD), bao gồm việc tạo mới tài khoản, xem và tra cứu thông tin người dùng, cập nhật quyền hạn, cũng như vô hiệu hóa tài khoản khi không còn nhu cầu sử dụng. Các chức năng này cho phép quản trị viên kiểm soát chặt chẽ vòng đời của từng tài khoản, đồng thời đảm bảo rằng chỉ những người dùng được cấp quyền hợp lệ mới có thể truy cập và thao tác trên hệ thống.

Trong mô hình hoạt động của FruitTrace, Cả Admin và Staff đảm nhiệm vai trò nhập liệu và đề xuất thay đổi thông tin liên quan đến sản phẩm. Các thay đổi này không được áp dụng trực tiếp mà phải được gửi dưới dạng yêu cầu chờ phê duyệt. Admin là người chịu trách nhiệm xem xét, phê duyệt hoặc từ chối các yêu cầu đó, đồng thời là vai trò duy nhất có quyền tác động trực tiếp lên dữ liệu chính thức của hệ thống. Cơ chế này giúp đảm bảo rằng mọi thay đổi đều được kiểm tra trước khi ghi nhận, hạn chế tối đa sai sót trong quá trình quản lý dữ liệu.

Nhờ việc áp dụng cơ chế phân quyền kết hợp với quy trình phê duyệt, hệ thống FruitTrace có thể đảm bảo tính toàn vẹn, nhất quán và độ tin cậy của dữ liệu truy xuất nguồn gốc sản phẩm. Đồng thời, cách tiếp cận này cũng giúp giảm thiểu rủi ro phát sinh từ các thao tác không được xác thực hoặc thiếu kiểm soát, góp phần nâng cao chất lượng vận hành và mức độ an toàn của toàn bộ hệ thống.

4 Các kĩ thuật dữ liệu được áp dụng

4.1 Normalization

Hệ thống cơ sở dữ liệu được thiết kế theo hướng chuẩn hoá đến Third Normal Form (3NF) nhằm giảm thiểu dư thừa dữ liệu và đảm bảo tính toàn vẹn.

Các thực thể chính như **Users**, **Farms**, **Products**, **Certifications** được tách thành các bảng riêng biệt, trong đó mỗi bảng chỉ lưu trữ các thuộc tính phụ thuộc trực tiếp vào khóa chính. Các mối quan hệ nhiều-nhiều, điển hình là giữa sản phẩm và chứng nhận, được biểu diễn thông qua bảng trung gian **Product_Certifications**.

Việc chuẩn hoá giúp tránh các bất thường khi cập nhật (update anomaly), thêm (insert anomaly) hoặc xoá (delete anomaly), đồng thời giúp hệ thống dễ bảo trì và mở rộng trong tương lai.

4.2 Query Optimization

Hệ thống áp dụng các kỹ thuật tối ưu truy vấn nhằm cải thiện hiệu năng truy xuất dữ liệu, đặc biệt đối với các API công khai phục vụ người dùng cuối. Một ví dụ tiêu biểu là API lấy chi tiết sản phẩm công khai, trong đó chỉ hiển thị các sản phẩm đang ở trạng thái **instock**.

Listing 1: Truy vấn tối ưu lấy chi tiết sản phẩm công khai

```
router.get('/products/public/:id', (req, res) => {  
  const sql = '  
    SELECT  
      p.ProductID, p.ProductName, p.ProdDescription AS Description,  
      p.Price, p.Status, p.HarvestDate,  
      f.FarmName, f.Address AS Location,  
      c.CertName, c.Issuer,  
      pc.IssueDate  
    FROM Products p  
    LEFT JOIN Farms f ON p.FarmID = f.FarmID  
    LEFT JOIN Product_Certifications pc ON p.ProductID = pc.  
      ProductID  
    LEFT JOIN Certifications c ON pc.CertID = c.CertID  
    WHERE p.ProductID = ? AND p.Status = ?;  
';
```



```
db.query(sql, [req.params.id, 'instock'], (err, results) => {  
  if (err || !results.length) {  
    return res.status(404).json({ msg: 'San pham khong ton tai  
    hoac da het hang' });  
  }  
  res.json(results[0]);  
});  
});
```

Truy vấn tối ưu được thiết kế để:

- Chỉ truy xuất các cột cần thiết thay vì sử dụng `SELECT *`
- Áp dụng điều kiện lọc trực tiếp trong câu lệnh SQL
- Giảm khối lượng dữ liệu trả về từ cơ sở dữ liệu

Cụ thể, truy vấn tối ưu chỉ lấy thông tin liên quan từ các bảng `Products`, `Farms`, `Product_Certifications` và `Certifications`, đồng thời áp dụng điều kiện lọc theo `ProductID` và `Status` ngay tại tầng cơ sở dữ liệu.

Listing 2: Truy vấn không tối ưu: lọc dữ liệu ở tầng ứng dụng

```
router.get('/products/public-slow/:id', (req, res) => {  
  const sql = '  
    SELECT  
      p.*, f.*, u.*, c.*, pc.*  
    FROM Products p  
    LEFT JOIN Farms f ON p.FarmID = f.FarmID  
    LEFT JOIN Users u ON u.UserID = p.CreatedBy  
    LEFT JOIN Product_Certifications pc ON p.ProductID = pc.  
      ProductID  
    LEFT JOIN Certifications c ON pc.CertID = c.CertID;  
  ';  
  
  db.query(sql, (err, results) => {  
    const product = results.find(  
      (r) => r.ProductID == req.params.id && r.Status === 'instock'  
    );  
    if (!product) {  
      return res.status(404).json({ msg: 'Out of stock' });  
    }  
    res.json(product);  
  });  
});
```

```
});  
});
```

Ngược lại, phiên bản truy vấn không tối ưu lấy toàn bộ dữ liệu từ nhiều bảng bằng `SELECT *` và thực hiện lọc dữ liệu ở tầng ứng dụng (JavaScript). Cách tiếp cận này làm tăng lượng dữ liệu truyền tải và thời gian xử lý không cần thiết.

Trong thử nghiệm giả lập với khoảng 2.000 sản phẩm, truy vấn tối ưu đạt thời gian phản hồi trung bình khoảng 50 ms, trong khi truy vấn không tối ưu có thời gian phản hồi lên đến khoảng 180 ms. Kết quả cho thấy việc áp dụng các kỹ thuật tối ưu truy vấn giúp cải thiện hiệu năng đáng kể và giảm tải cho hệ thống backend.

4.3 Data Indexing

Để tăng tốc độ truy vấn, hệ thống sử dụng cơ chế đánh chỉ mục (Indexing) trên các cột thường xuyên được sử dụng trong điều kiện tìm kiếm và lọc dữ liệu. Các chỉ mục được tạo cho các trường như tên người dùng, trạng thái phê duyệt, tên sản phẩm và người yêu cầu phê duyệt.

Listing 3: Script tạo chỉ mục dữ liệu

```
const indexes = [  
  'CREATE INDEX idx_user_username ON Users(Username)',  
  'CREATE INDEX idx_product_name ON Products(ProductName)',  
  'CREATE INDEX idx_approval_status ON ApprovalRequests(Status)',  
  'CREATE INDEX idx_approval_requestedby ON ApprovalRequests(  
    RequestedBy)',  
];  
  
async function runIndexing() {  
  for (const query of indexes) {  
    db.query(query);  
  }  
}
```

Quá trình tạo chỉ mục được thực hiện thông qua script riêng, đảm bảo có thể chạy lại nhiều lần mà không gây lỗi khi chỉ mục đã tồn tại. Việc đánh chỉ mục giúp cải thiện đáng kể hiệu năng truy vấn, đặc biệt đối với các chức năng tìm kiếm, lọc danh sách và quản lý yêu cầu phê duyệt.

4.4 Data Backup & Recovery

Hệ thống triển khai cơ chế sao lưu và phục hồi dữ liệu tự động nhằm đảm bảo an toàn dữ liệu. Dữ liệu MySQL được sao lưu bằng công cụ `mysqldump`, tạo ra các file sao lưu định dạng `.sql` theo thời gian thực.

Listing 4: Cơ chế sao lưu dữ liệu MySQL lên Cloudinary

```
const backupToCloudinary = async () => {
  const timestamp = new Date().toISOString().replace(/[:.]/g, '-');
  const fileName = `backup-${timestamp}.sql`;
  const filePath = path.join(__dirname, `../uploads/${fileName}`);

  const dumpCommand = `mysqldump -h ${host} -P ${port} -u ${user} \
    -p"${password}" --column-statistics=0 \
    --set-gtid-purged=OFF --no-tablespaces ${database} > "${
    filePath
  }";

  exec(dumpCommand, async () => {
    const result = await cloudinary.uploader.upload(filePath, {
      resource_type: 'raw',
      folder: 'backup_db_doan'
    });
    fs.unlinkSync(filePath);
  });
};
```

Listing 5: Khôi phục dữ liệu từ bản backup mới nhất

```
const recoverFromCloudinary = async () => {
  const result = await cloudinary.search
    .expression('folder:backup_db_doan AND resource_type:raw')
    .sort_by('created_at', 'desc')
    .max_results(1)
    .execute();

  const fileUrl = result.resources[0].secure_url;
  const restoreCmd = `mysql -h ${host} -P ${port} -u ${user} \
    -p"${password}" ${database} < restore_temp.sql`;

  exec(restoreCmd);
};
```

Sau khi sao lưu, các file backup được tải lên dịch vụ lưu trữ đám mây Cloudinary, giúp dữ liệu được lưu trữ ở vị trí độc lập với máy chủ chính. Cơ chế phục hồi cho phép tải về bản sao lưu mới nhất từ Cloudinary và khôi phục lại dữ liệu vào hệ thống khi xảy ra sự cố.

Giải pháp này giúp giảm thiểu rủi ro mất dữ liệu, đồng thời đảm bảo khả năng khôi phục nhanh chóng trong các tình huống lỗi hệ thống hoặc hỏng hóc dữ liệu.

4.5 Database Security

Bảo mật cơ sở dữ liệu được đảm bảo thông qua cơ chế phân quyền theo vai trò (Role-Based Access Control). Hệ thống định nghĩa hai vai trò chính là **admin** và **staff**, trong đó mỗi vai trò có quyền hạn khác nhau đối với các thao tác dữ liệu.

Cơ chế xác thực người dùng được triển khai bằng JSON Web Token (JWT). Sau khi đăng nhập thành công, hệ thống cấp token cho người dùng và token này được sử dụng để xác thực các yêu cầu tiếp theo.

Đối với các thao tác quan trọng như tạo, cập nhật hoặc xóa dữ liệu, các yêu cầu từ vai trò **staff** không được áp dụng trực tiếp lên cơ sở dữ liệu. Thay vào đó, các thao tác này được ghi nhận vào bảng **ApprovalRequests** và chỉ được thực thi sau khi được vai trò **admin** phê duyệt. Cách tiếp cận này giúp tăng cường kiểm soát dữ liệu, giảm thiểu rủi ro thao tác sai và đảm bảo tính an toàn cho hệ thống.

5 Đánh giá các hạn chế và nội dung chưa đạt được

Mặc dù hệ thống FruitTrace đã hoàn thành các chức năng cốt lõi và cơ bản đáp ứng được mục tiêu quản lý, truy xuất và kiểm soát dữ liệu sản phẩm, quá trình xây dựng và triển khai hệ thống vẫn còn tồn tại một số hạn chế và khó khăn nhất định.

Trước hết, do giới hạn về thời gian thực hiện và phạm vi của đề án, hệ thống hiện nay mới tập trung chủ yếu vào các chức năng quản trị và kiểm soát dữ liệu ở phía backend và giao diện quản trị (Admin Portal). Các chức năng mở rộng dành cho người dùng cuối vẫn chưa được triển khai đầy đủ, cụ thể:

- Chưa xây dựng giao diện truy xuất thông tin sản phẩm dành cho khách hàng.
- Chưa phát triển ứng dụng di động hoặc giao diện tối ưu cho thiết bị di động.

Điều này khiến khả năng tiếp cận hệ thống cũng như trải nghiệm của người dùng cuối chưa được đánh giá một cách toàn diện.

Bên cạnh đó, cơ chế xét duyệt yêu cầu (Request List) hiện tại vẫn chủ yếu dựa trên thao tác thủ công và quyết định trực tiếp của quản trị viên. Hệ thống chưa tích hợp các cơ chế hỗ trợ tự động, bao gồm:

- Các quy tắc kiểm tra và sàng lọc ban đầu đối với các yêu cầu không hợp lệ.
- Các tiêu chí đánh giá thông minh nhằm hỗ trợ quá trình phê duyệt.

Do đó, trong trường hợp số lượng yêu cầu tăng cao, khối lượng công việc của Admin có thể trở nên lớn, ảnh hưởng đến hiệu suất xử lý và thời gian phản hồi.

Ngoài ra, việc kiểm tra và xác thực dữ liệu đầu vào hiện mới dừng ở mức cơ bản, chưa bao phủ đầy đủ các tình huống phát sinh trong thực tế. Một số nội dung vẫn còn hạn chế như:

- Các tiêu chí đánh giá chất lượng sản phẩm chưa được chuẩn hóa hoàn toàn.
- Điều kiện về chứng nhận và quy trình sản xuất chủ yếu được kiểm soát thông qua bước xét duyệt thủ công.
- Chưa áp dụng các ràng buộc nghiệp vụ chặt chẽ ngay tại thời điểm nhập dữ liệu.

Về mặt kỹ thuật, hệ thống chưa được triển khai và kiểm thử trong môi trường thực tế với quy mô lớn. Do đó, các vấn đề liên quan đến:

- Hiệu năng hệ thống khi xử lý khối lượng dữ liệu lớn,
- Bảo mật dữ liệu và cơ chế phân quyền người dùng,
- Khả năng mở rộng và tích hợp với các hệ thống bên thứ ba

mới chỉ được xem xét ở mức độ lý thuyết và chưa được đánh giá đầy đủ trong quá trình vận hành thực tế.

Nhìn chung, những hạn chế nêu trên chủ yếu xuất phát từ giới hạn về thời gian, nguồn lực và phạm vi nghiên cứu của đề án. Tuy nhiên, đây cũng là cơ sở quan trọng để nhóm xác định các hướng phát triển tiếp theo, nhằm tiếp tục hoàn thiện hệ thống FruitTrace và nâng cao tính ứng dụng trong thực tiễn.

6 Đánh giá kết quả đạt được của dự án

Dự án FruitTrace đã hoàn thành đầy đủ các mục tiêu đề ra ban đầu, bao phủ toàn bộ các nội dung chính từ thiết kế hệ thống đến triển khai và đánh giá giải pháp. Cụ thể, các kết quả nổi bật mà dự án đạt được bao gồm:

- Thiết kế và xây dựng cơ sở dữ liệu phục vụ cho việc quản lý và truy xuất nguồn gốc sản phẩm.
- Phát triển ứng dụng web hỗ trợ quét mã QR trực tiếp trên trình duyệt, giúp tự động hóa quá trình truy xuất thông tin sản phẩm.
- Triển khai đầy đủ các thao tác CRUD và áp dụng cơ chế kiểm soát thay đổi dữ liệu thông qua quy trình xét duyệt.
- Ứng dụng các kỹ thuật Data Engineering trong quá trình xử lý, lưu trữ và quản lý dữ liệu.

Thông qua quá trình thực hiện, nhóm đã xây dựng được một hệ thống mẫu hoạt động ổn định, thể hiện khả năng kết nối liền mạch giữa frontend, backend và cơ sở dữ liệu. Việc tích hợp chức năng quét mã QR trực tiếp trên trình duyệt đã góp phần giảm thiểu các thao tác thủ công, hạn chế sai sót và cải thiện tốc độ xử lý dữ liệu trong quá trình vận hành hệ thống.

Bên cạnh phần triển khai kỹ thuật, nhóm cũng đã tiến hành thử nghiệm và đánh giá hiệu quả của một số giải pháp hỗ trợ vận hành hệ thống, bao gồm:

- Chuẩn hóa dữ liệu nhằm đảm bảo tính nhất quán giữa các bảng và thực thể.
- Tối ưu truy vấn và lập chỉ mục để cải thiện hiệu năng truy xuất dữ liệu.
- Xây dựng cơ chế sao lưu và phục hồi dữ liệu ở mức cơ bản.
- Áp dụng một số biện pháp bảo mật nhằm bảo vệ dữ liệu và hệ thống.

Kết luận

Dự án FruitTrace, mặc dù được triển khai ở quy mô thử nghiệm, đã giúp nhóm hiểu rõ hơn quy trình xây dựng một hệ thống quản lý dữ liệu hoàn chỉnh, từ thiết kế cơ sở dữ liệu, phát triển API, xây dựng giao diện người dùng đến tích hợp và vận hành hệ thống. Đây là nền tảng quan trọng để nhóm tiếp tục mở rộng hệ thống trong tương lai, chẳng hạn như bổ sung cơ chế phân quyền chi tiết, triển khai trên môi trường thực tế hoặc mở rộng khả năng phân tích dữ liệu theo các hướng nâng cao như Big Data và Machine Learning.

Thông qua quá trình thực hiện đồ án, nhóm không chỉ củng cố được kiến thức lý thuyết đã học mà còn tích lũy được nhiều kinh nghiệm thực tiễn trong việc xây dựng và triển khai hệ thống. Mặc dù vẫn còn tồn tại một số hạn chế nhất định, kết quả đạt được cho thấy tính khả thi của mô hình FruitTrace trong việc hỗ trợ quản lý và truy xuất dữ liệu sản phẩm, đồng thời đáp ứng tốt các yêu cầu đặt ra của môn học.