­Trường Ngoại Ngữ - Tin Học TPHCM

Khoa Công nghệ Thông Tin

Đồ án Tốt nghiệp Đại học

Bộ môn: Công nghệ phần mềm

Farm Hub

|  |  |
| --- | --- |
| Sinh viên thực hiện | Trần Hoàng Long – 15DH110018  Phùng Ngô Trọng Nghĩa – 15DH110364 |
| Người hướng dẫn | TS.Bùi Thị Thanh Tú |

­­

TPHCM, 5/2019

# Lời cam kết

Họ và tên sinh viên: Trần Hoàng Long

Điện thoại liên lạc: 0522936344. . . . . . . . . . Email: tranhoanglong51198@gmail.com . . . . .

Lớp: 15TH1501 . . . . . . . . . . . . . . . . . . . . . . Hệ đào tạo: Đại Học chính qui . . . . . . . . . . . .

Tôi – Trần Hoàng Long – cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *Thạc sĩ Bùi Thị Thanh Tú*. Các kết quả nêu trong ĐATN là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm với dù chỉ một sao chép vi phạm quy chế của nhà trường.

|  |  |
| --- | --- |
|  | *Hồ Chí Minh, ngày 11 tháng 5 năm 2019*  Tác giả ĐATN  *Họ và tên sinh viên* |

# Lời cảm ơn

Lời cảm ơn của sinh viên (SV) tới người yêu, gia đình, bạn bè, thầy cô, và chính bản thân mình vì đã chăm chỉ và quyết tâm thực hiện ĐATN để đạt kết quả tốt nhất ☺

# Tóm tắt

Sinh viên viết tóm tắt ĐATN của mình trong mục này, với 200 đến 350 từ. Theo trình tự, các nội dung tóm tắt cần có: (i) Giới thiệu vấn đề (tại sao có vấn đề đó, hiện tại được giải quyết chưa, có những hướng tiếp cận nào, các hướng này giải quyết như thế nào, hạn chế là gì), (ii) Hướng tiếp cận sinh viên lựa chọn là gì, vì sao chọn hướng đó, (iii) Tổng quan giải pháp của sinh viên theo hướng tiếp cận đã chọn, và (iv) Đóng góp chính của ĐATN là gì, kết quả đạt được sau cùng là gì. Sinh viên cần viết thành đoạn văn, không được viết ý hoặc gạch đầu dòng.

# Abstract

Mục này khuyến khích sinh viên viết lại mục “Tóm tắt” đồ án tốt nghiệp ở trang trước bằng tiếng Anh. Phần này phải có đầy đủ các nội dung như trong phần tóm tắt bằng tiếng Việt.

Sinh viên không nhất thiết phải trình bày mục này. Nhưng nếu lựa chọn trình bày, sinh viên cần đảm bảo câu từ và ngữ pháp chuẩn tắc, nếu không sẽ có tác dụng ngược, gây phản cảm.

# Mục lục

[Lời cam kết iii](#_Toc529131830)

[Lời cảm ơn iv](#_Toc529131831)

[Tóm tắt v](#_Toc529131832)

[Abstract vi](#_Toc529131833)

[Mục lục vii](#_Toc529131834)

[Danh mục hình vẽ xi](#_Toc529131835)

[Danh mục bảng xii](#_Toc529131836)

[Danh mục công thức xiii](#_Toc529131837)

[Danh mục các từ viết tắt xiv](#_Toc529131838)

[Danh mục thuật ngữ xv](#_Toc529131839)

[Chương 1 Giới thiệu đề tài 1](#_Toc529131840)

[1.1 Đặt vấn đề 1](#_Toc529131841)

[1.2 Mục tiêu và phạm vi đề tài 1](#_Toc529131842)

[1.3 Định hướng giải pháp 2](#_Toc529131843)

[1.4 Bố cục đồ án 2](#_Toc529131844)

[Chương 2 Khảo sát và phân tích yêu cầu 4](#_Toc529131845)

[2.1 Khảo sát hiện trạng 4](#_Toc529131846)

[2.2 Tổng quan chức năng 4](#_Toc529131847)

[2.2.1 Biểu đồ use case tổng quan 5](#_Toc529131848)

[2.2.2 Biểu đồ use case phân rã XYZ 5](#_Toc529131849)

[2.2.3 Quy trình nghiệp vụ 5](#_Toc529131850)

[2.3 Đặc tả chức năng 5](#_Toc529131851)

[2.3.1 Đặc tả use case A 5](#_Toc529131852)

[2.3.2 Đặc tả use case B 5](#_Toc529131853)

[2.4 Yêu cầu phi chức năng 6](#_Toc529131854)

[Chương 3 Công nghệ sử dụng 7](#_Toc529131855)

[Chương 4 Phát triển và triển khai ứng dụng 8](#_Toc529131856)

[4.1 Thiết kế kiến trúc 8](#_Toc529131857)

[4.1.1 Lựa chọn kiến trúc phần mềm 8](#_Toc529131858)

[4.1.2 Thiết kế tổng quan 8](#_Toc529131859)

[4.1.3 Thiết kế chi tiết gói 9](#_Toc529131860)

[4.2 Thiết kế chi tiết 10](#_Toc529131861)

[4.2.1 Thiết kế giao diện 10](#_Toc529131862)

[4.2.2 Thiết kế lớp 10](#_Toc529131863)

[4.2.3 Thiết kế cơ sở dữ liệu 11](#_Toc529131864)

[4.3 Xây dựng ứng dụng 11](#_Toc529131865)

[4.3.1 Thư viện và công cụ sử dụng 11](#_Toc529131866)

[4.3.2 Kết quả đạt được 11](#_Toc529131867)

[4.3.3 Minh hoạ các chức năng chính 12](#_Toc529131868)

[4.4 Kiểm thử 12](#_Toc529131869)

[4.5 Triển khai 12](#_Toc529131870)

[Chương 5 Các giải pháp và đóng góp nổi bật 13](#_Toc529131871)

[Chương 6 Kết luận và hướng phát triển 14](#_Toc529131872)

[6.1 Kết luận 14](#_Toc529131873)

[6.2 Hướng phát triển 14](#_Toc529131874)

[Tài liệu tham khảo 15](#_Toc529131875)

[Phụ lục A-1](#_Toc529131876)

[A Hướng dẫn viết đồ án tốt nghiệp A-1](#_Toc529131877)

[A.1 Quy định chung A-1](#_Toc529131878)

[A.2 Tạo đề mục A-2](#_Toc529131879)

[A.3 Bảng biểu A-2](#_Toc529131880)

[A.4 Hình vẽ A-3](#_Toc529131881)

[A.5 Tài liệu tham khảo A-4](#_Toc529131882)

[A.6 Công thức toán học A-5](#_Toc529131883)

[A.7 Tham chiếu chéo A-5](#_Toc529131884)

[A.8 Cập nhật mục lục và tham chiếu chéo A-6](#_Toc529131885)

[A.9 In quyển đồ án tốt nghiệp A-6](#_Toc529131886)

[B Đặc tả use case B-6](#_Toc529131887)

[B.1 Đặc tả use case “Thống kê tình hình mượn sách” B-6](#_Toc529131888)

[B.2 Đặc tả use case “Đăng ký làm thẻ mượn” B-6](#_Toc529131889)

[C Công nghệ sử dụng C-7](#_Toc529131890)

[C.1 Công nghệ bảo mật dữ liệu C-7](#_Toc529131891)

[C.2 Công nghệ blockchain C-7](#_Toc529131892)

[D Thiết kế gói D-7](#_Toc529131893)

[D.1 Thiết kế gói cho kiến trúc tổng quan D-7](#_Toc529131894)

[D.2 Thiết kế gói cho chức năng “Trả sách” D-7](#_Toc529131895)

[E Thiết kế lớp E-7](#_Toc529131896)

# Danh mục hình vẽ

[**Hình 1** Ví dụ biểu đồ phụ thuộc gói 9](#_Toc529131897)

[**Hình 2** Ví dụ thiết kế gói 10](#_Toc529131898)

[**Hình 3** Ví dụ hình vẽ A-4](#_Toc529131899)

# Danh mục bảng

[**Bảng 1** Danh sách thư viện và công cụ sử dụng 11](#_Toc529131900)

[**Bảng 2** Ví dụ sử dụng bảng A-3](#_Toc529131901)

# Danh mục công thức

[**Công thức 1** Khai triển Newton A-5](#_Toc529131902)

Lưu ý: Nếu ĐATN có ít hơn ba công thức toán học, sinh viên có thể xóa bỏ mục này.

# Danh mục các từ viết tắt

|  |  |
| --- | --- |
| API | Application Programming Interface  Giao diện lập trình ứng dụng |
| **EUD** | End-User Development  Phát triển ứng dụng người dùng cuối |
| **GWT** | Google Web Toolkit  Công cụ lập trình Javascript bằng Java của Google |
| **HTML** | HyperText Markup Language  Ngôn ngữ đánh dấu siêu văn bản |
| **CNTT** | Công nghệ thông tin |
| **ĐATN** | Đồ án tốt nghiệp |
| **SV** | Sinh viên |

# Danh mục thuật ngữ

|  |  |
| --- | --- |
| Browser | Trình duyệt |
| **Cache memory** | Bộ nhớ đệm |
| **E-commerce** | Thương mại điện tử |
| **Bloatware** | Ứng dụng nhà sản xuất tích hợp vào thiết bị |
| **Interpreter** | Trình thông dịch |
| **Compiler** | Trình biên dịch |

# Giới thiệu đề tài

## Đặt vấn đề

Sản xuất nông lâm, là lĩnh vực sản xuất quan trọng nhất đối với sự phát triển của Việt Nam, nhiều biến đổi thăng trầm từ những năm tháng thống nhất đất nước vào năm 1975. Sự tăng trưởng mạnh được ghi nhận vào năm 1976—tăng đến 10% so với năm 1975—nhưng sản xuất lại giảm khoảng đến còn 95% trong những năm 1976, 1977 và 1978 và sự phục hồi đáng kể trong năm 1979.

Vấn đề đầu ra cho sản phẩm nông nghiệp đang là một khó khăn rất lớn cho sản xuất nông nghiệp; điều dễ nhận thấy nhất đó là hiện tượng được mùa- rớt giá, được giá-mất mùa thường xuyên xảy ra. Nguyên nhân chủ yếu đó là: chưa gắn kết được chính sách đầu tư phát triển nông nghiệp của Nhà nước với việc qui hoạch vùng chuyên canh sản xuất còn chậm và nhiều hạn chế.

Người chủ trang trại nước ta rất cần cù chịu khó, sẵn sàng tiếp thu kiến thức mới để tạo thiên nhiên, giúp ích cho hoạt động nông nghiệp của mình. Tuy nhiên, một trong nhưng nhược điểm của chủ trang trại hiện nay là làm việc manh mún. Bên cạnh đó, do tập quán làm việc theo cảm tính dẫn đến người chủ trang trại không có định hướng phát triển hoạt động nông nghiệp rõ ràng nếu như không có sự tư vấn chi tiết của các cơ quan chuyên môn, của những người có kinh nghiệm. Đã có nhiều hiện tượng người chủ trang trại không có hứng thú sản xuất trên mảnh đất của mình do diện tích đất canh tác nhỏ, năng suất lao động thấp, hoặc sự đầu tư của họ không đúng hướng, dẫn đến việc khủng hoảng dư thừa sản phẩm nông sản (điều, cà phê, mía,...) khiến không bán được hàng, gây lỗ vốn nặng cho người chủ trang trại.

Về phía doanh nghiệp, cũng chính vì không có sự tương tác với chủ trang trại nên hầu như cũng không nắm rõ được tình hình nguồn cung. Điều này đưa đến một tình cảnh đó là "bên cung không nắm được cầu, và bên cầu không nắm được cung" gây hệ quả là doanh nghiệp không có đủ thông tin để dự báo và lập kế hoạch cung ứng hiệu quả.

Vậy có ba vấn đề đặt ra là:

* Làm sao để giúp người chủ trang trại nắm bắt được giá cả thị trường cũng như khối lượng nông sản cung cấp trên toàn thị trường để tránh tình cung nhiều hơn cầu và ngược lại ?
* Làm sao để tránh tình trạng bán phá giá, doanh nghiệp ép giá chủ trang trại khiến giá nông sản vươt ngoài tầm kiểm soát ?
* Làm sao để giúp doanh nghiệp và chủ trang trại có thể tương tác tốt với nhau ?

## Mục tiêu và phạm vi đề tài

Mục tiêu:

* Xây dựng một website cung cấp cho chủ trang trại và doanh nghiệp thông tin và công cụ hỗ trợ về thị trường (nông sản, giá, khối lượng,...).
* Cung cấp phương thức để doanh nghiệp và chủ trang trại có thể nhận biết được nhu cầu và khả năng cung cấp của nhau.
* Giúp doanh nghiệp và chủ trang trại có thể tương tác với nhau không màng khoảng cách địa lý.
* Góp phần bình ổn giá cả thị trường, thúc đẩy nền kinh tế thị trường.

Phạm vi đề tài: Tất cả doanh nghiệp và chủ trang trại trên đất nước.

## Định hướng giải pháp

Từ việc xác định rõ nhiệm vụ cần giải quyết ở phần 1.2, sinh viên đề xuất định hướng giải pháp của mình theo trình tự sau: (i) Sinh viên trước tiên trình bày sẽ giải quyết vấn đề theo định hướng, phương pháp, thuật toán, kỹ thuật, hay công nghệ nào; Tiếp theo, (ii) sinh viên mô tả ngắn gọn giải pháp của mình là gì (khi đi theo định hướng/phương pháp nêu trên); và sau cùng, (iii) sinh viên trình bày đóng góp chính của đồ án là gì, kết quả đạt được là gì.

Sinh viên lưu ý không giải thích hoặc phân tích chi tiết công nghệ/thuật toán trong phần này. Sinh viên chỉ cần nêu tên định hướng công nghệ/thuật toán, mô tả ngắn gọn trong một đến hai câu và giải thích nhanh lý do lựa chọn.

Mạng Internet phổ biến rất rộng rãi, vì vậy phương pháp tối ưu nhất chính là xây dựng một website online để cả chủ trang trại và doanh nghiệp có thể tiếp cận nhau dễ dàng hơn.

Website bao gổm hai phần.

* Phần thứ nhất là dành cho chủ trang trại, dủng để đăng ký thông tin các nông trại thuộc quyền sở hữu và thông tin nông sản (giống, mùa vụ,..) sẽ trồng ở nông trại đó. Chủ trang trại có thể đăng bán sản phẩm của mình, hệ thống sẽ tự động lọc ra những nhu cầu mua phù hợp, chủ trang trại cũng sẽ được cập nhật tình hình giá mua bán trung bình trên toàn thị trường và sản lượng nông sản cung cấp dự tính để tránh bị dội hàng.
* Phần thứ hai của website dành cho doanh nghiệp, dùng để đăng tin nhu cầu mua của mình, hệ thống sẽ tự động lọc những trang trại có trồng loại nông sản phù hợp với yêu cầu. Ngoài ra doanh nghiệp cũng được hỗ trợ cập nhật giá trung bình toàn thị trường, tổng khối lượng các loại nông sản dự tính.

## Bố cục đồ án

Phần còn lại của báo cáo đồ án tốt nghiệp này được tổ chức như sau.

Chương 2 trình bày về v.v.

Trong Chương 3, em/tôi giới thiệu về v.v.

**Chú ý:** Sinh viên cần viết mô tả thành đoạn văn đầy đủ về nội dung chương. Tuyệt đối không viết ý hay gạch đầu dòng. Chương 1 không cần mô tả trong phần này.

Ví dụ tham khảo mô tả chương trong phần bố cục đồ án tốt nghiệp: Chương \*\*\* trình bày đóng góp chính của đồ án, đó là một nền tảng ABC cho phép khai phá và tích hợp nhiều nguồn dữ liệu, trong đó mỗi nguồn dữ liệu lại có định dạng đặc thù riêng. Nền tảng ABC được phát triển dựa trên khái niệm DEF, là các module ngữ nghĩa trợ giúp người dùng tìm kiếm, tích hợp và hiển thị trực quan dữ liệu theo mô hình cộng tác và mô hình phân tán.

**Chú ý:** Trong phần nội dung chính, mỗi chương của đồ án nên có phần *Tổng quan* và *Kết chương*. Hai phần này đều có định dạng văn bản “Normal”, sinh viên không cần tạo định dạng riêng, ví dụ như không in đậm/in nghiêng, không đóng khung, v.v.

Trong phần *Tổng quan* của chương N, sinh viên nên có sự liên kết với chương N-1 rồi trình bày sơ qua lý do có mặt của chương N và sự cần thiết của chương này trong đồ án. Sau đó giới thiệu những vấn đề sẽ trình bày trong chương này là gì, trong các đề mục lớn nào.

Ví dụ về phần *Tổng quan*: Chương 3 đã thảo luận về nguồn gốc ra đời, cơ sở lý thuyết và các nhiệm vụ chính của bài toán tích hợp dữ liệu. Chương 4 này sẽ trình bày chi tiết các công cụ tích hợp dữ liệu theo hướng tiếp cận “mashup”. Với mục đích và phạm vi của đề tài, sáu nhóm công cụ tích hợp dữ liệu chính được trình bày bao gồm: (i) nhóm công cụ ABC trong phần 4.1, (ii) nhóm công cụ DEF trong phần 4.2, nhóm công cụ GHK trong phần 4.3, v.v.

Trong phần *Kết chương*, sinh viên đưa ra một số kết luận quan trọng của chương. Những vấn đề mở ra trong *Tổng quan* cần được tóm tắt lại nội dung và cách giải quyết/thực hiện như thế nào. Sinh viên lưu ý không viết *Kết chương* giống hệt *Tổng quan*. Sau khi đọc phần *Kết chương*, người đọc sẽ nắm được sơ bộ nội dung và giải pháp cho các vấn đề đã trình bày trong chương. Trong *Kết chương*, Sinh viên nên có thêm câu liên kết tới chương tiếp theo.

Ví dụ về phần *Kết chương*: Chương này đã phân tích chi tiết sáu nhóm công cụ tích hợp dữ liệu. Nhóm công cụ ABC và DEF thích hợp với những bài toán tích hợp dữ liệu phạm vi nhỏ. Trong khi đó, nhóm công cụ GHK lại chứng tỏ thế mạnh của mình với những bài toán cần độ chính xác cao, v.v. Từ kết quả nghiên cứu và phân tích về sáu nhóm công cụ tích hợp dữ liệu này, tôi đã thực hiện phát triển phần mềm tự động bóc tách và tích hợp dữ liệu sử dụng nhóm công cụ GHK. Phần này được trình bày trong chương tiếp theo – Chương 5.

# ­­Khảo sát và phân tích yêu cầu

Chương này có độ dài từ 9 đến 11 trang. Với đồ án nghiên cứu, sinh viên đổi tên chương thành “Tình hình nghiên cứu hiện nay” (Related works – State of the art) và phân tích rõ ngữ cảnh bài toán cũng như các kết quả nghiên cứu tương tự. Nếu là đồ án nghiên cứu, các đề mục cần thay đổi cho phù hợp. Sinh viên cần trao đổi kỹ với GV hướng dẫn để đưa ra được đề mục phù hợp nhất.

Với phương pháp phân tích thiết kế hướng đối tượng, sinh viên sử dụng biểu đồ use case theo hướng dẫn của template này. Với các phương pháp khác, sinh viên trao đổi với giáo viên hướng dẫn để đổi tên và sắp xếp lại đề mục cho phù hợp. Ví dụ, thay vì sử dụng biểu đồ use case, sinh viên đi theo hướng tiếp cận Agile có thể dùng User Story.

## Khảo sát hiện trạng

Thông thường, khảo sát chi tiết về hiện trạng và yêu cầu của phần mềm sẽ được lấy từ ba nguồn chính, đó là (i) người dùng/khách hàng, (ii) các hệ thống đã có, (iii) và các ứng dụng tương tự.

Sinh viên cần tiến hành phân tích, so sánh, đánh giá chi tiết ưu nhược điểm của các sản phẩm/nghiên cứu hiện có. Sinh viên có thể lập bảng so sánh nếu cần thiết. Kết hợp với khảo sát người dùng/khách hàng (nếu có), sinh viên nêu và mô tả sơ lược các tính năng phần mềm quan trọng cần phát triển.

Nông dân cần phải biết rõ sản phẩm của mình sẽ bán ở đâu, phương thức thế nào, giá cả, khối lượng bao nhiêu, yêu cầu chất lượng ra sao ? Đây là vấn đề rất quan trọng, thiết thực để nông dân chủ động hoạch định và tổ chức sản xuất.

Người nông dân cần phải nắm bắt được thông tin thị trường, nhu cầu thị trường cần gì, từ đó có thể chủ động chọn cây trồng, chọn giống và vụ mùa phù hợp, liên kết được với nhau trong sản xuất để khắc phục tình trạng bất cập: sản xuất nhỏ lẻ, phân tán, chuyển đổi chậm, không đồng bộ

## Tổng quan chức năng

Phần 2.2 này có nhiệm vụ tóm tắt các chức năng của phần mềm. Trong phần này, sinh viên lưu ý chỉ mô tả chức năng mức cao (tổng quan) mà không đặc tả chi tiết cho từng chức năng. Đặc tả chi tiết được trình bày trong phần 2.3.

Website gồm các chức năng chính như:

* Đăng ký thông tin người dùng
* Đăng ký thông tin và quản lý nông trại
* Đăng ký thông tin và quản lý nông sản sẽ trồng ở nông trại xác định
* Đăng nhu cầu bán nông sản
* Đăng nhu cầu mua nông sản
* Xem lịch sử giao dịch
* Xem thống kê giá giao dịch trung bình, sản lượng nông sản cung cấp, vv.. trên toàn thị trường

### Biểu đồ use case tổng quan

Sinh viên vẽ biểu đồ use case tổng quan và giải thích các tác nhân tham gia là gì, nêu vai trò của từng tác nhân, và mô tả ngắn gọn các use case chính.

### Biểu đồ use case phân rã XYZ

Với mỗi use case mức cao trong biểu đồ use case tổng quan, sinh viên tạo một mục riêng như mục 2.2.2 và tiến hành phân rã use case đó. Lưu ý tên use case cần phân rã trong biểu đồ use case tổng quan phải khớp với tên đề mục.

Trong mỗi mục như vậy, sinh viên vẽ và giải thích ngắn gọn các use case phân rã.

### Quy trình nghiệp vụ

Nếu sản phẩm/hệ thống cần xây dựng có quy trình nghiệp vụ quan trọng/đáng chú ý, sinh viên cần mô tả và vẽ biểu đồ hoạt động minh họa quy trình nghiệp vụ đó. Sinh viên lưu ý đây không phải là luồng sự kiện của *từng use case*, mà là luồng hoạt động kết hợp nhiều use case để thực hiện một nghiệp vụ nào đó.

Ví dụ, một hệ thống quản lý thư viện có quy trình nghiệp vụ mượn trả với mô tả sơ bộ như sau: Sinh viên làm thẻ mượn, sau đó sinh viên đăng ký mượn sách, thủ thư cho mượn, và cuối cùng sinh viên trả lại sách cho thư viện. Một hệ thống có thể có một vài quy trình nghiệp vụ quan trọng như vậy.

## Đặc tả chức năng

Sinh viên lựa chọn từ 4 đến 7 use case quan trọng nhất của đồ án để đặc tả chi tiết. Mỗi đặc tả bao gồm ít nhất các thông tin sau: (i) Tên use case, (ii) Luồng sự kiện (chính và phát sinh), (iii) Tiền điều kiện, và (iv) Hậu điều kiện. Sinh viên chỉ vẽ bổ sung biểu đồ hoạt động khi đặc tả use case phức tạp.

### Đặc tả use case A

### Đặc tả use case B

## Yêu cầu phi chức năng

Trong phần này, sinh viên đưa ra các yêu cầu khác nếu có, bao gồm các yêu cầu phi chức năng như hiệu năng, độ tin cậy, tính dễ dùng, tính dễ bảo trì, hoặc các yêu cầu về mặt kỹ thuật như về CSDL, công nghệ sử dụng, v.v.

Tính dễ dùng:

* Giao diện đơn giản dễ nhìn, không rối mắt
* Màu sắc tươi tắn hài hòa, bắt mắt
* Các ô nhập liệu luôn luôn có gợi ý để người dùng xác định được là sẽ nhập thông tin gì
* Trong trường hợp người dùng nhập sai, nhập thiếu dữ liệu, thông điệp cảnh báo sẽ hiện lên. Ngoài ra nội dung cảnh báo cũng sẽ mô tả chi tiết về lỗi và cách sửa lỗi giúp người dùng dễ xác định

Tính dễ bảo trì:

* Website dùng chung một layout chuẩn
* Mỗi chức năng (thêm, xóa, sửa, vv...) dùng những form chung cố định. Có ba form chung chính: form thêm mới, form chi tiết, bảng danh sách dữ liệu
* Code được phân vùng rõ ràng theo mô hình MVC. Có ba vùng lớn chính: Code dùng cho chủ trang trại (Areas/ Farmer), code dùng cho Admin ((Areas/ Admin) và code dùng cho doanh nghiệp

# Công nghệ sử dụng

Chương này có độ dài từ 10 đến 13 trang. Nếu cần trình bày dài hơn, sinh viên đưa vào phần phụ lục. Chú ý đây là kiến thức đã có sẵn; SV sau khi tìm hiểu được thì phân tích và tóm tắt lại. Sinh viên không trình bày dài dòng, chi tiết.

Với đồ án ứng dụng, sinh viên để tên chương là “Công nghệ sử dụng”. Trong chương này, sinh viên giới thiệu về các công nghệ, nền tảng sử dụng trong đồ án. Sinh viên cũng có thể trình bày thêm nền tảng lý thuyết nào đó nếu cần dùng tới.

Với đồ án nghiên cứu, sinh viên đổi tên chương thành “Cơ sở lý thuyết”. Khi đó, nội dung cần trình bày bao gồm: Kiến thức nền tảng, cơ sở lý thuyết, các thuật toán, phương pháp nghiên cứu, v.v.

Với từng công nghệ/nền tảng/lý thuyết được trình bày, sinh viên phải phân tích rõ công nghệ/nền tảng/lý thuyết đó dùng để để giải quyết vấn đề/yêu cầu cụ thể nào ở Chương 2. Hơn nữa, với từng vấn đề/yêu cầu, sinh viên phải liệt kê danh sách các công nghệ/hướng tiếp cận tương tự có thể dùng làm lựa chọn thay thế, rồi giải thích rõ sự lựa chọn của mình.

Lưu ý: Nội dung ĐATN phải có tính chất liên kết, liền mạch, và nhất quán. Vì vậy, các công nghệ/thuật toán trình bày trong chương này phải khớp với nội dung giới thiệu của sinh viên ở phần 1.3.

Trong chương này, để tăng tính khoa học và độ tin cậy, sinh viên nên chỉ rõ nguồn kiến thức mình thu thập được ở tài liệu nào, đồng thời đưa tài liệu đó vào trong danh sách tài liệu tham khảo rồi tạo các tham chiếu chéo (xem hướng dẫn ở phụ lục A.7).

Website chia làm ba phần chính: front-end, back-end và database.

## Front End

Những thứ như giao diện, tính năng, cấu trúc, font chữ, đồ họa, link liên kết, các nút chức năng, ảnh động, vv... mà người dùng nhìn thấy được đó chính là front-end. Front-end đóng vai trò quan trọng trong việc thiết kế và xây dựng website.

### Các công nghệ được sử dụng

Năm 1990, ông Tim Berners-Lee tạo ra ba công nghệ nền tảng, vẫn tiếp tục được sử dụng đến ngày nay gồm:

* Ngôn ngữ đánh dấu siêu văn bản HyperText Markup Language (HTML)
* Định danh tài nguyên đồng nhất Uniform Resource Indentifier (URI hay phổ biến hơn là URL)
* Phương thức giao tiếp siêu văn bản HyperTest Transfer Protocal (HTTP).

Nếu HTML là ngôn ngữ của web, URL có vai trò như địa chỉ truy cập tài nguyên, thì HTTP là phương thức quyết định tài nguyên được truy xuất như thế nào.

Ngoài ra, còn có những công nghệ khác như JavaScripts, jQuerry và Bootstrap:

* JavaScripts là một ngôn ngữ lập trình được sử dụng để tạo ra những trang web tương tác. Nó được tích hợp và nhúng trong HTML. JavaScripts cho phép kiểm soát các hành vi của trang web tốt hơn so với khi chỉ có một mình HTML. Với JavaScripts, một website sẽ trở nên sinh động hơn với các trải nghiệm động, tăng tính tương tác, tăng thêm tính năng cho website
* jQuery là một thư viện JavaScript nhanh, nhỏ, đa nền tảng và giàu tính năng. Nó được thiết kế để đơn giản hóa client-side scripting của HTML. Nó thực hiện những nhiệm vụ như HTML document traversal và manipulation, animation, event handling, AJAX một cách rất đơn giản bằng một API dễ sử dụng và có khả năng hoạt động trên nhiều loại trình duyệt khác nhau. Mục đích chính của jQuery là làm việc sử dụng JavaScript trên trang web trở nên dễ dàng hơn, giúp website có tính tương tác và hấp dẫn hơn. Nó cũng được sử dụng để thêm vào các animation.
* Bootstrap là một framework cho phép thiết kế website reponsive nhanh hơn và dễ dàng hơn. Bootstrap bao gồm các HTML templates, CSS templates và Javascript tạo ra những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm các plugin JavaScript trong nó. Giúp cho việc thiết kế reponsive dễ dàng hơn và nhanh chóng hơn.

### Ứng dụng vào website

Khi thiết kế layout cho website, Bootstrap trở nên cực kì hữu dụng vì nó cung cấp khả năng responsive, giúp không bị bể giao diện khi thu phóng to nhỏ màn hình. Ngoài ra Bootstrap cũng giúp rất nhiều về việc tăng cao tính thẩm mỹ cho website khi được sử dụng để thiết kế nút, slideshow, vv...

Để dựng nên một biểu đồ dựa trên data-base thì không thể nào thiếu ChartJS, đó là một thư viện được viết bằng JavaScripts. JavaScripts hỗ trợ rất nhiều tính năng khác như upload ảnh, vv... JavaScripts là thứ không thể thiếu đối với hầu hết các trang web hiện nay.

Ngoài những công nghệ nêu trên, còn rất nhiều cách để dựng nên một giao diện đẹp như PHP, AngularJS, vv...

## Back End

Nếu Front End Developer có quyền lực kiến tạo nên vẻ đẹp của các trang web, thì Back End developer là người xử lý mọi logic nghiệp vụ phức tạp ở ẩn ở phía sau, giúp cho hệ thống hoạt động trơn tru. Dữ liệu của người dùng, thuật toán phân tích … đều nằm ở back-end.

### Các công nghệ được sử dụng

**ASP.NET** là một nền tảng ứng dụng web (*web application framework*) được phát triển và cung cấp bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft), cho phép những người lập trình tạo ra những trang web động, những [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) và những [dịch vụ web](https://vi.wikipedia.org/wiki/D%E1%BB%8Bch_v%E1%BB%A5_web). Lần đầu tiên được đưa ra thị trường vào tháng 1 năm 2002 cùng với phiên bản 1.0 của.NET framework, là công nghệ nối tiếp của Microsoft's Active Server Pages(ASP). ASP.NET được biên dịch dưới dạng [Common Language Runtime](https://vi.wikipedia.org/w/index.php?title=Common_Language_Runtime&action=edit&redlink=1) (CLR), cho phép những người lập trình viết mã ASP.NET với bất kỳ ngôn ngữ nào được hỗ trợ bởi.NET language. Đây là một framework rất đáng được sử dụng vì:

* **SoC:** Một trong những lợi ích chính được đưa ra bởi ASP.NET MVC đó là Separation of Concern (phân tách mối bận tâm). Framework này cung cấp sự phân tách rất rõ gàng giữa model, logic nghiệp vụ, giao diện người dùng và dữ liệu.
* **Control:** Khi sử dụng framework ASP.NET MVC, người dùng được cung cấp một bộ control rất mạnh mẽ trên JavaScript, HTML và CSS so với các control được cung cấp bởi một số hình thức truyền thống trên web.
* **Tích hợp:** Khi tích hợp MVC với jQuery, người dùng có thể viết code của mình chạy trong các trình duyệt web. Điều này sẽ giúp giảm tải cho các web server.

**C#** là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) [hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng) được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft), là phần khởi đầu cho kế hoạch [.NET](https://vi.wikipedia.org/w/index.php?title=.NET&action=edit&redlink=1)của họ. C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic), [Delphi](https://vi.wikipedia.org/wiki/Delphi_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)).

* Được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.
* C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.
* C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

C# có rất nhiều ưu điểm nổi bật:

* C# có cấu trúc khá gần gũi với các ngôn ngữ lập trình truyền thống, nên cũng khá dể dàng tiếp cận và học nhanh với C#.
* C# có thể biên dịch trên nhiều nền tảng máy tính khác nhau.
* C# được xây dựng trên nền tảng của C++ và Java nên nó được thừa hưởng những ưu điểm của ngôn ngữ đó.
* C# là một phần của .NET Framework nên được sự chống lưng khá lớn đến từ bộ phận này.
* C# có IDE Visual Studio cùng nhiều plug-in vô cùng mạnh mẽ.

### Ứng dụng vào website

Được sử dụng ở cả ba lớp trong mô hình MVC, C# chính là thành phần cốt lõi khi xây dựng back-end cho website

Ở hầu hết mọi website, trang chủ luôn phức tạp nhất vì phải hiển thị nhiều thông tin từ nhiều bảng khác nhau (danh sách chào mua, danh sách chào bán, top nông sản hot nhất, vv...), và biểu đồ. Vấn đề này có thể giải quyết bằng cách tạo một Model chứa tất cả những thông tin cần hiển thị, sau đó nhận dữ liệu từ Controller rồi hiển thị lên View

Phần xử lý được chia ra làm hai luồng. Luồng thứ nhất là xử lý về dữ liệu gồm các thao tác như truy xuất, tạo mới, chỉnh sửa, xóa dữ liệu, được thực hiện ở tầng DAO (Data Access Layer). Ví dụ như truy xuất danh sách nhu cầu bán của chủ trang trại (logined), rồi dựa vào danh sách đó truy xuất được những nhu cầu mua phù hợp từ phía doanh nghiệp, vv... Luồng thứ hai là xử lý luồng nghiệp vụ, các thao tác của người dùng, được thực hiện ở Controller

Phần View có định dạng là .cshtml, nhờ vậy có thể sử dụng C# thông qua Razor. Ở tầng View, C# dùng vòng lặp foreach để hiển thị các danh sách dữ liệu truy xuất được từ tầng DAO hiển thị lên View

## Data Base

Database hay còn gọi là dữ liệu nền tảng hay cơ sở dữ liệu. Một cơ sở dữ liệu là một tập hợp tổ chức các dữ liệu. Đây là bộ sưu tập các lược đồ (Schema), bảng (Tables), truy vấn (Query), báo cáo (Report), và các đối tượng khác.

**Cơ sở dữ liệu** hiện diện ở tất cả các ứng dụng trên nền tảng công nghệ thông tin. Có đến 99.99% các ứng dụng cần tới cơ sở dữ liệu Từ cơ sở dữ liệu dạng đơn giản như cơ sở dữ liệu dạng file text, dạng file đơn giản như microsoft access, sql lite cho đến các cơ sở dữ liệu loại siêu lớn như SQLServer, Oracle, các database dạng No-SQL....

### Các công nghệ được sử dụng

SQL Server hay còn gọi là Microsoft SQL Server, viết tắt là **MS SQL Server**. Đây là một phần mềm được phát triển bởi Microsoft dùng để lưu trữ dữ liệu dựa trên chuẩn RDBMS, và nó cũng là một hệ quản trị cơ sở dữ liệu quan hệ đối tượng (ORDBMS).

SQL Server cung cấp đầy đủ công cụ để quản lý, từ giao diện GUI cho đến việc sử dụng ngôn ngữ truy vấn SQL. Ngoài ra điểm mạnh của nó là Microsoft có khá nhiền nền tảng kết hợp hoàn hảo với SQL Server như ASP.NET, C# xây dựng Winform, bởi vì nó hoạt động hoàn toàn độc lập. SQL Server có những tính năng giúp việc sử dụng hiệu quả hơn như:

* Cho phép tạo nhiều cơ sở dử liệu
* Duy trì lưu trữ bền vững
* Bảo mật cao
* Phân tích dữ liệu bằng SSAS - SQL Server Analysis Services
* Tạo được báo cáo bằng SSRS - SQL Server Reporting Services
* Thực hiện quá trình ETL (Extract-Transform-Load) bằng SSIS - SQL Server Integration Services

### Ứng dụng vào website

SQL Server thực sự là một trợ thủ đắc lực khi cung cấp rất nhiều tính năng tiện lợi khác ngoài việc lưu trữ dữ liệu.

Khi truy xuất nhiều bảng liên thông với nhau, SQL Server hỗ trợ tạo một Database Diagrams gồm tất cả hay một vài bảng chỉ định, giúp lập trình viên có thể thấy được một cách trực quan các mối quan hệ giữa các bảng, các khóa chính, khóa phụ, các trường dữ liệu, vv...

Ngoài ra, Store Procedure, Trigger, Job được sử dụng để hỗ trợ việc đảm bảo tính đúng đắn của dữ liệu, thực hiện tác vụ, tự động xử lý một công việc tại thời điểm ấn định, vv...

# Phát triển và triển khai ứng dụng

## Thiết kế kiến trúc

**Kiến trúc phần mềm** của một [chương trình máy tính](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m) hay một hệ thống tính toán là cấu trúc của các thành phần trong hệ thống đó. Kiến trúc phần mềm bao gồm các phần tử [phần mềm](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m), các thuộc tính và mối quan hệ giữa chúng. Ngoài ra, thuật ngữ "kiến trúc phần mềm" cũng đề cập đến các tài liệu kiến trúc phần mềm của một hệ thống, thuận tiện cho việc trao đổi thông tin giữa các thành phần trong một dự án. Kiến trúc phần mềm giúp việc quyết định ở mức cao trong thiết kế phần mềm dễ dàng hơn và cho phép tái sử dụng các thành phần và [mẫu thiết kế](https://vi.wikipedia.org/wiki/M%E1%BA%ABu_thi%E1%BA%BFt_k%E1%BA%BF) của các dự án.

### Lựa chọn kiến trúc phần mềm

Mục này có độ dài từ một đến ba trang. Sinh viên cần lựa chọn kiến trúc phần mềm cho ứng dụng của mình như: kiến trúc ba lớp MVC, MVP, SOA, Microservice, v.v. rồi giải thích sơ bộ về kiến trúc đó (không giải thích chi tiết/dài dòng).

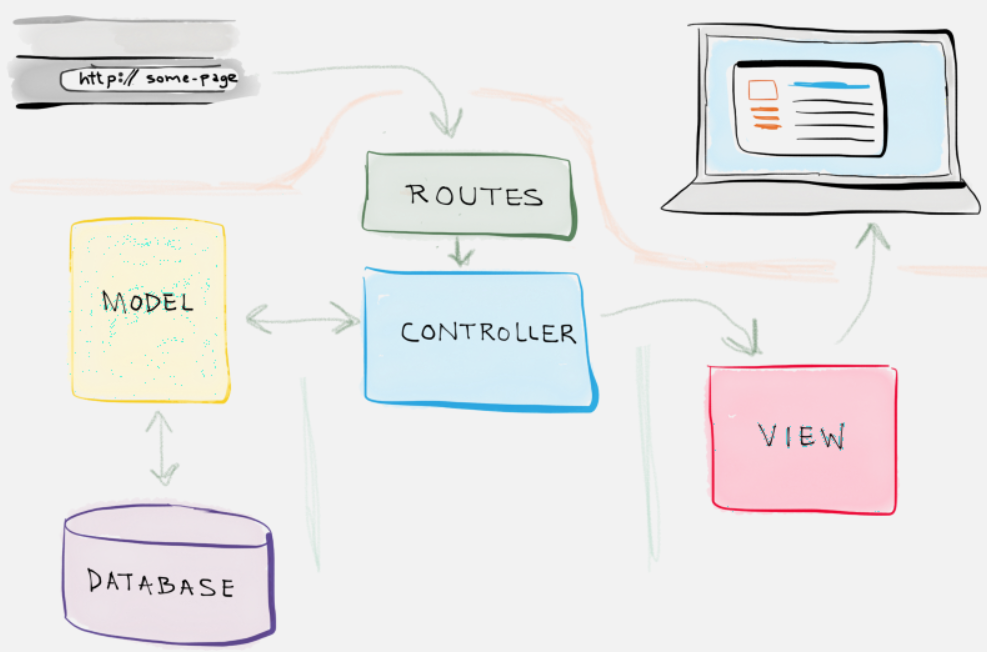
Sử dụng kiến trúc phần mềm đã chọn ở trên, sinh viên mô tả kiến trúc cụ thể cho ứng dụng của mình. Gợi ý: sinh viên áp dụng lý thuyết chung vào hệ thống/sản phẩm của mình như thế nào, có thay đổi, bổ sung hoặc cải tiến gì không. Ví dụ, thành phần M trong kiến trúc lý thuyết MVC sẽ là *những thành phần cụ thể nào* (ví dụ: là interface I + class C1 + class C2, v.v.) trong kiến trúc phần mềm của sinh viên.

**MVC** (*viết tắt của* **M**odel-**V**iew-**C**ontroller) là một mẫu kiến trúc phần mềm  để tạo lập giao diện người dùng trên máy tính. MVC chia một ứng dụng thành ba phần tương tác được với nhau để tách biệt giữa cách thức mà thông tin được xử lý nội hàm và phần thông tin được trình bày và tiếp nhận từ phía người dùng.

Khi sử dụng đúng cách, mẫu *MVC* giúp cho người phát triển phần mềm cô lập các nguyên tắc nghiệp vụ và giao diện người dùng một cách rõ ràng hơn. Phần mềm phát triển theo mẫu *MVC* tạo nhiều thuận lợi cho việc bảo trì vì các nguyên tắc nghề nghiệp và giao diện ít liên quan với nhau.

Trong mẫu *Model-View-Controller*:

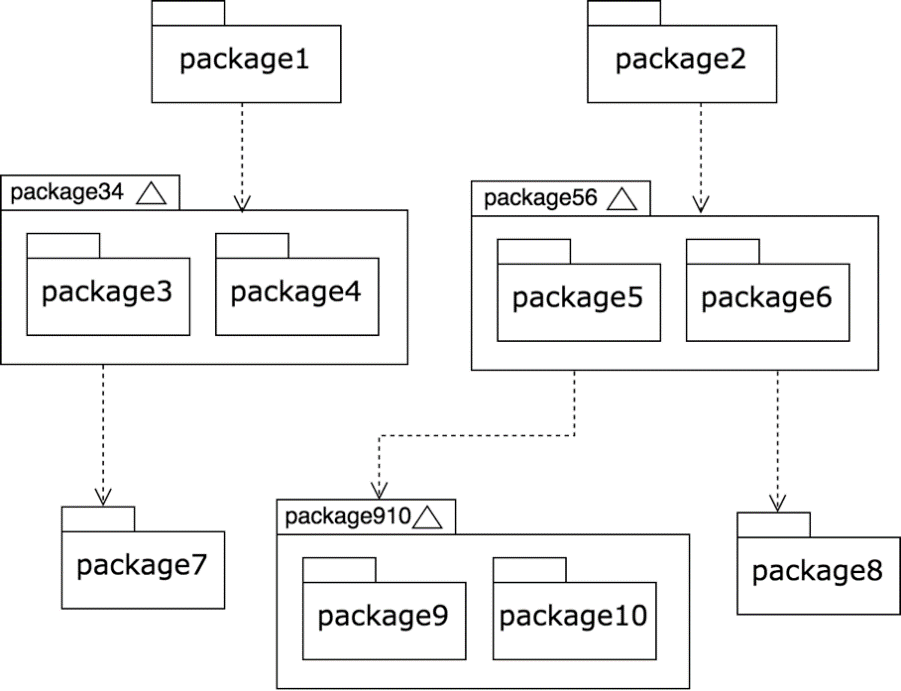
* Mô hình (*model*) tượng trưng cho [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) của chương trình phần mềm.
* Tầm nhìn hay khung nhìn (*view*) bao gồm các thành phần của giao diện người dùng.
* Bộ kiểm tra hay bộ điều chỉnh (*controller*) quản lý sự trao đổi giữa dữ liệu và các nguyên tắc nghề nghiệp trong các thao tác liên quan đến mô hình.



Mô hình MVC khi áp dụng vào khóa luận thì có chút thay đổi. Thứ nhất, việc xử lý truy xuất dữ liệu và các thao tác khác trên database sẽ được chuyển toàn bộ sang một lớp mới tên là DAO (Data Access Layer) nhằm giúp chia nhỏ luồng xử lý của chương trình, thuận tiện cho việc bảo trì cũng như debug. Thứ hai, sẽ có 3 vùng (area) chính: Admin, Farmer, Trader, mỗi vùng này đều hoạt động dựa trên cấu trúc MVC

### Thiết kế tổng quan

Sinh viên vẽ biểu đồ gói UML (UML package diagram), nêu rõ sự phụ thuộc giữa các gói (package). SV cần vẽ các gói sao cho chúng được phân theo các tầng rõ ràng, không được sắp đặt package lộn xộn trong hình vẽ. Sinh viên chú ý các quy tắc thiết kế (Các gói không phụ thuộc lẫn nhau, gói tầng dưới không phụ thuộc gói tầng trên, không phụ thuộc bỏ qua tầng, v.v.) và cần giải thích sơ lược về mục đích/nhiệm vụ của từng package. SV tham khảo ví dụ minh họa trong Hình 1.



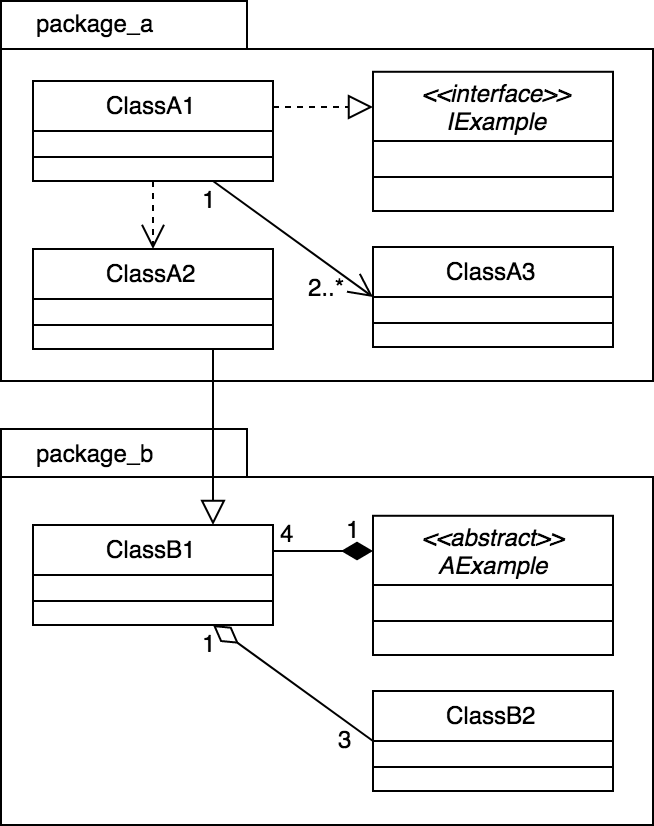
**Hình 1** Ví dụ biểu đồ phụ thuộc gói

### Thiết kế chi tiết gói

Sinh viên thiết kế và lần lượt vẽ biểu đồ thiết kế cho từng package, hoặc một nhóm các package liên quan để giải quyết một vấn đề gì đó. Khi vẽ thiết kế gói, sinh viên chỉ cần đưa tên lớp, không cần chỉ ra các thành viên phương thức và thuộc tính. SV tham khảo ví dụ minh họa trong Hình 2.

Sinh viên cần vẽ rõ ràng quan hệ giữa các lớp trong biểu đồ. Các quan hệ bao gồm: phụ thuộc (dependency), kết hợp (association), kết tập (aggregation), hợp thành (composition), kế thừa (inheritance), và thực thi (implementation). Các quan hệ này đều đã được minh họa trong Hình 2.

Sau khi vẽ hình minh họa, sinh viên cần giải thích ngắn gọn về thiết kế của mình.



**Hình 2** Ví dụ thiết kế gói

## Thiết kế chi tiết

### Thiết kế giao diện

Phần này có độ dài từ hai đến ba trang. Sinh viên đặc tả thông tin về màn hình mà ứng dụng của mình hướng tới, bao gồm độ phân giải màn hình, kích thước màn hình, số lượng màu sắc hỗ trợ, v.v. Tiếp đến, sinh viên đưa ra các thống nhất/chuẩn hóa của mình khi thiết kế giao diện như thiết kế nút, điều khiển, vị trí hiển thị thông điệp phản hồi, phối màu, v.v. Sau cùng sinh viên đưa ra một số hình ảnh minh họa thiết kế giao diện cho các chức năng quan trọng nhất. Lưu ý, sinh viên không nhầm lẫn giao diện thiết kế với giao diện của sản phẩm sau cùng.

Website có độ phân giải HD, kích cỡ màn hình 1920 x 1080. Hỗ trợ các màu có sẵn trong các trình duyệt web phổ biến: Chrome, IE, FireFox, vv....

Các chuẩn hóa về thiết kế:

* Thiết kế nút: thường các nút chức năng như thêm mới, cập nhật sẽ căn giữa màn hình, kế đến là nút hủy hoặc quay về. Viền vàng, nền trắng, chữ đen
* Thông điệp phản hồi cảnh báo lỗi có phông chữ màu đỏ. Thông điệp báo lỗi chính nằm ở phía trên form nhập liệu. Các cảnh báo lỗi khác (nhập sai định dạng tháng/ngày/năm, nhập quá phạm vi dữ liệu, nhập chữ trong trường thông tin là số, để trống những thông tin bắt buộc, vv...) có phông chữ màu đỏ, hiển thị phía dưới ô nhập liệu mà người dùng nhập sai
* Website hướng tới sự đơn giản, tránh mỏi mắt cho người dùng vì vậy sẽ chọn sử dụng những màu cơ bản, nền trắng đen

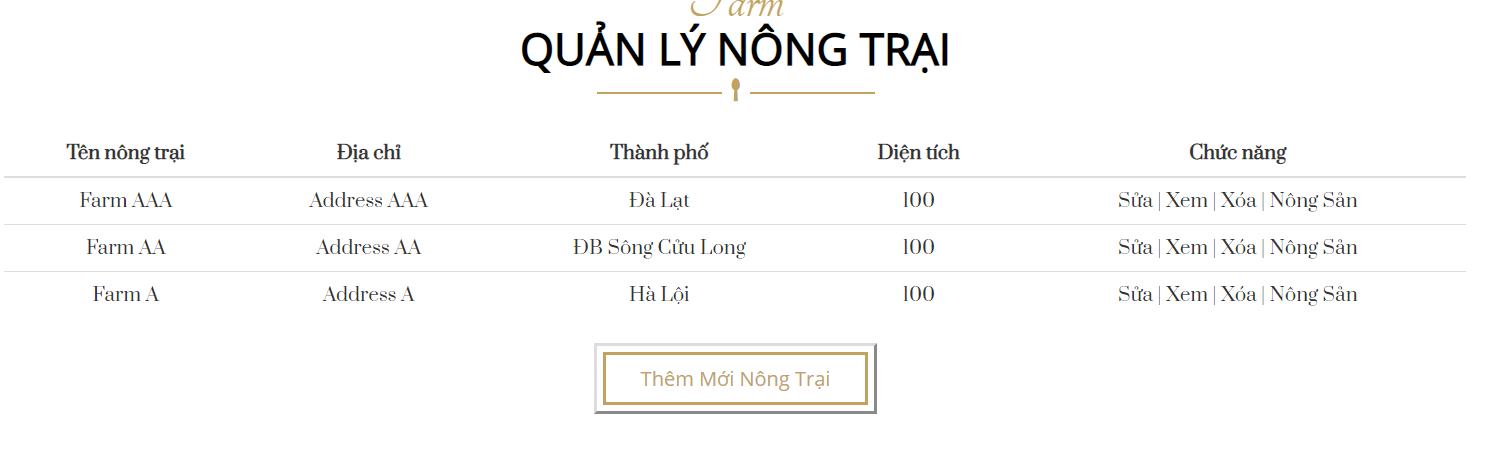
Sau đây là một số hình ảnh minh họa thiết kế giao diện cho một số chức năng quan trọng nhất.

Các trang của website đều sử dụng một layout chung nhằm đồng nhất giao diện



**Hình x** Layout chung của website

Danh sách các dữ liệu sẽ thể hiện ở dạng bảng, các nút chức năng như thêm xóa sửa sẽ hiển thị theo từng dòng dữ liệu



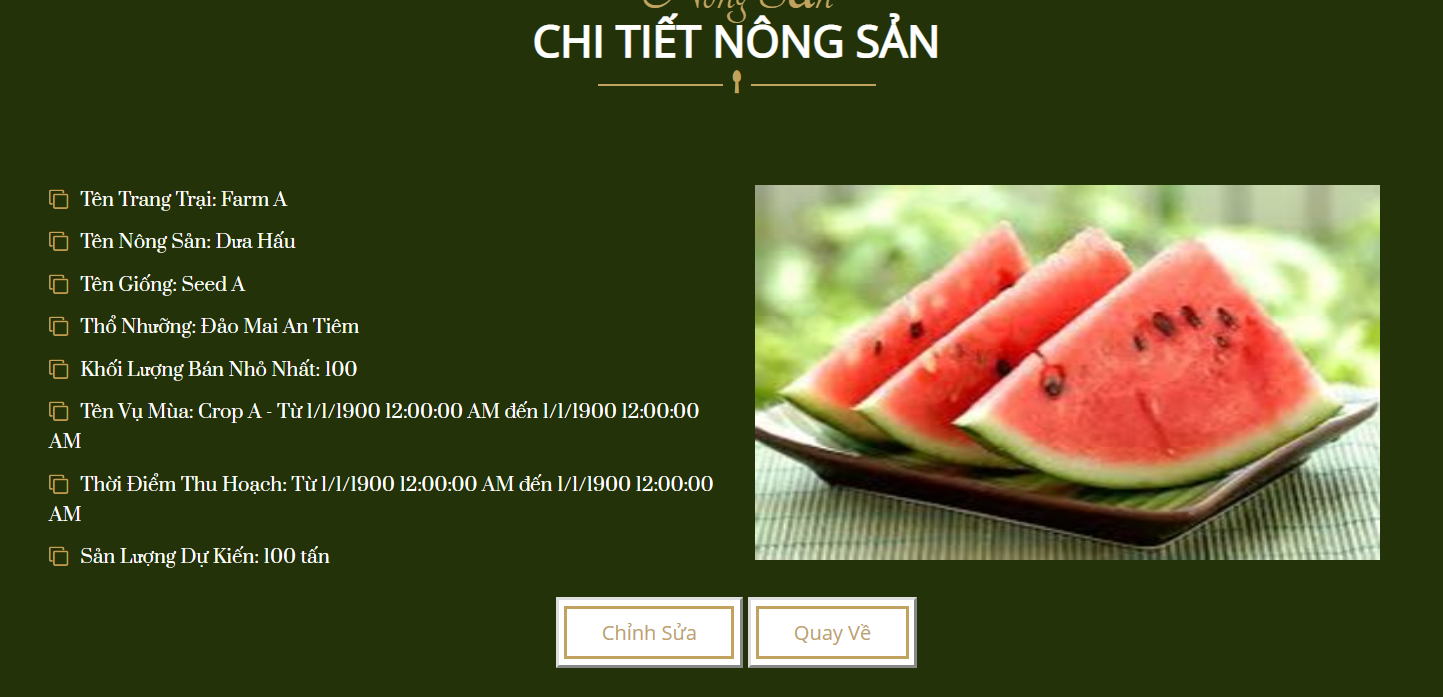
**Hình x** Bảng danh sách dữ liệu

Thông tin chi tiết của một đối tượng sẽ được hiển thị tập trung ở giữa màn hình để dễ nhìn, dễ theo dõi



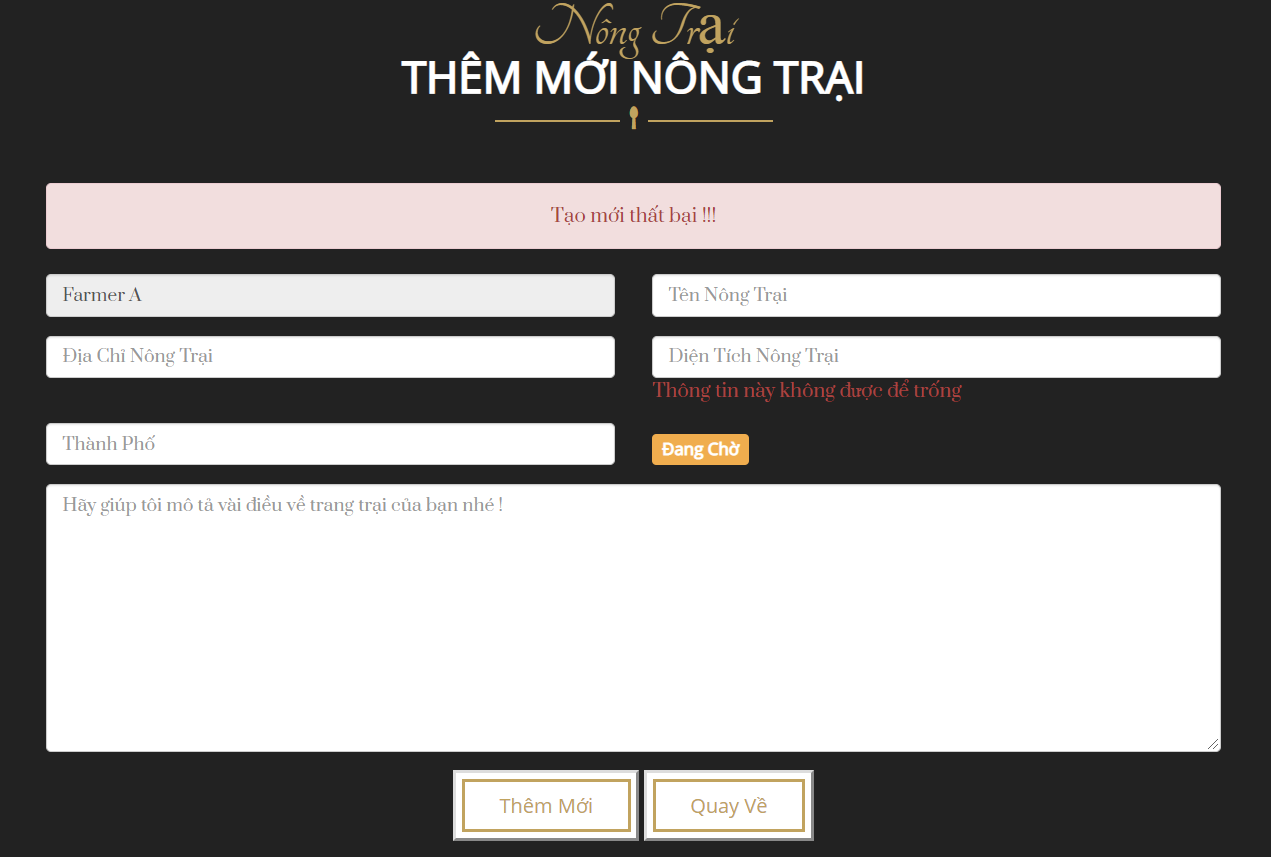
**Hình x** Thông tin chi tiết

Chi tiết nông sản có thêm hình ảnh, vì vậy sẽ hiển thị theo bố cục 5-5. Bên trái là thông tin về tên nông sản, giống, vv...bên trái là hình ảnh của nông sản đó



**Hình x** Chi tiết nông sản

Form tạo mới có các gợi ý để người dùng dễ nhập liệu thông tin. Ngoài ra, các lỗi nhập liệu cũng được thể hiện rõ ràng. Hình dưới là ví dụ cho thấy khi chủ trang trại muốn tạo mới một nông trại, nhưng lại quên điền thông tin diện tích nông trại



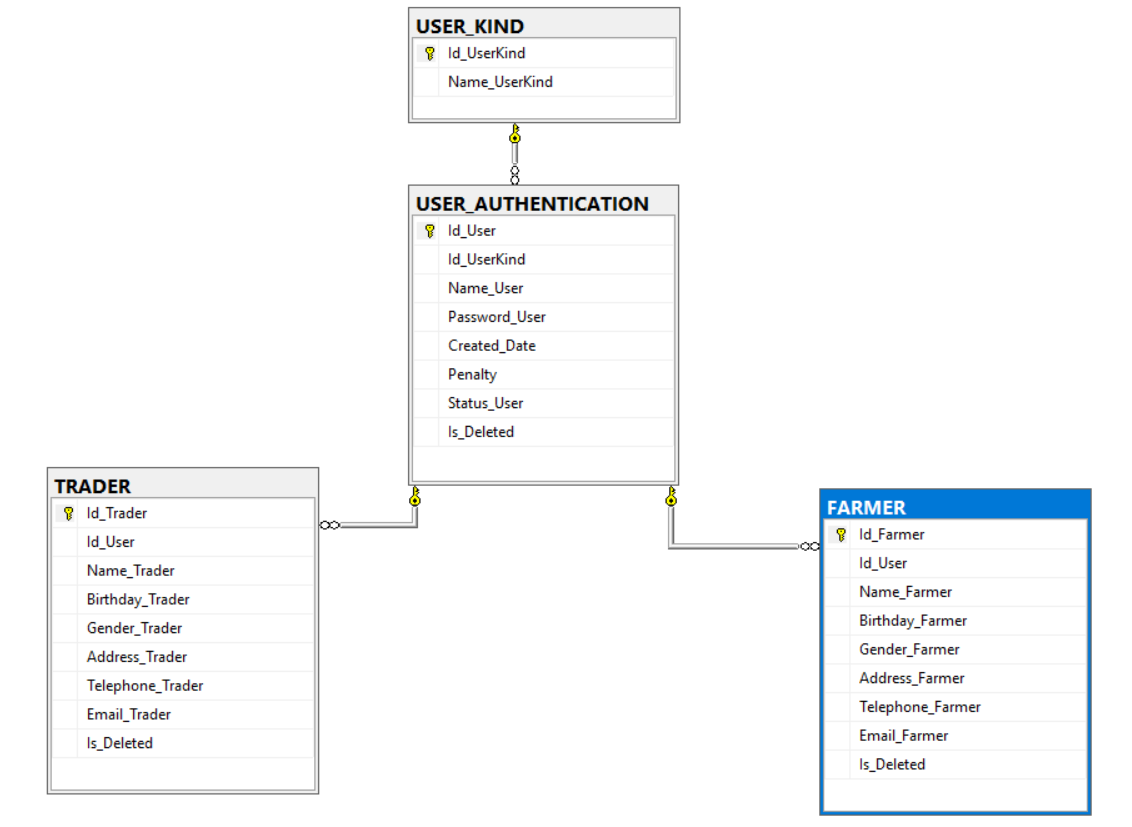
**Hình x** Thêm mới một đối tượng

### Thiết kế lớp

Phần này có độ dài từ ba đến bốn trang. Sinh viên trình bày thiết kế chi tiết các thuộc tính và phương thức cho một số lớp chủ đạo/quan trọng nhất của ứng dụng (từ 2-4 lớp). Thiết kế chi tiết cho các lớp khác, nếu muốn trình bày, sinh viên đưa vào phần phụ lục.

Để minh họa thiết kế lớp, sinh viên thiết kế luồng truyền thông điệp giữa các đối tượng tham gia cho 2 đến 3 use case quan trọng nào đó bằng biểu đồ trình tự (hoặc biểu đồ giao tiếp).

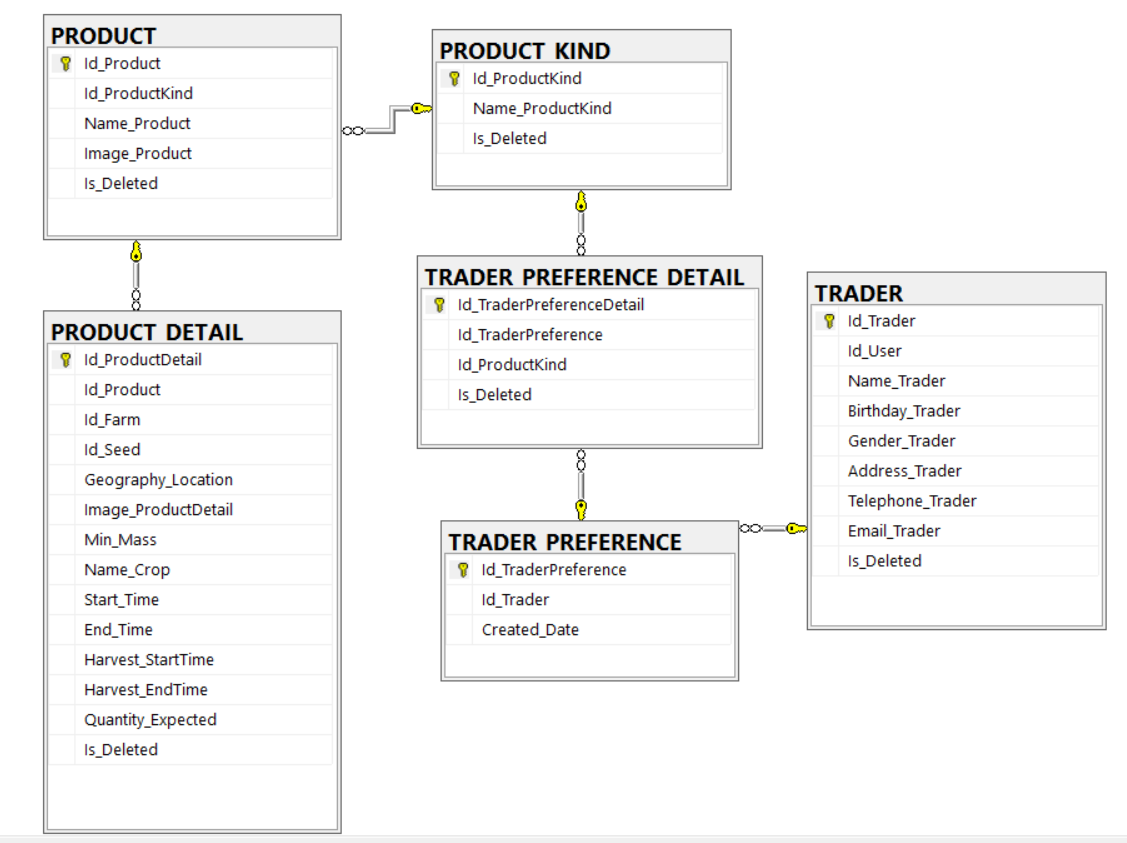
### Thiết kế cơ sở dữ liệu



**Hình x** Đăng ký thông tin người dùng

Giải thích:

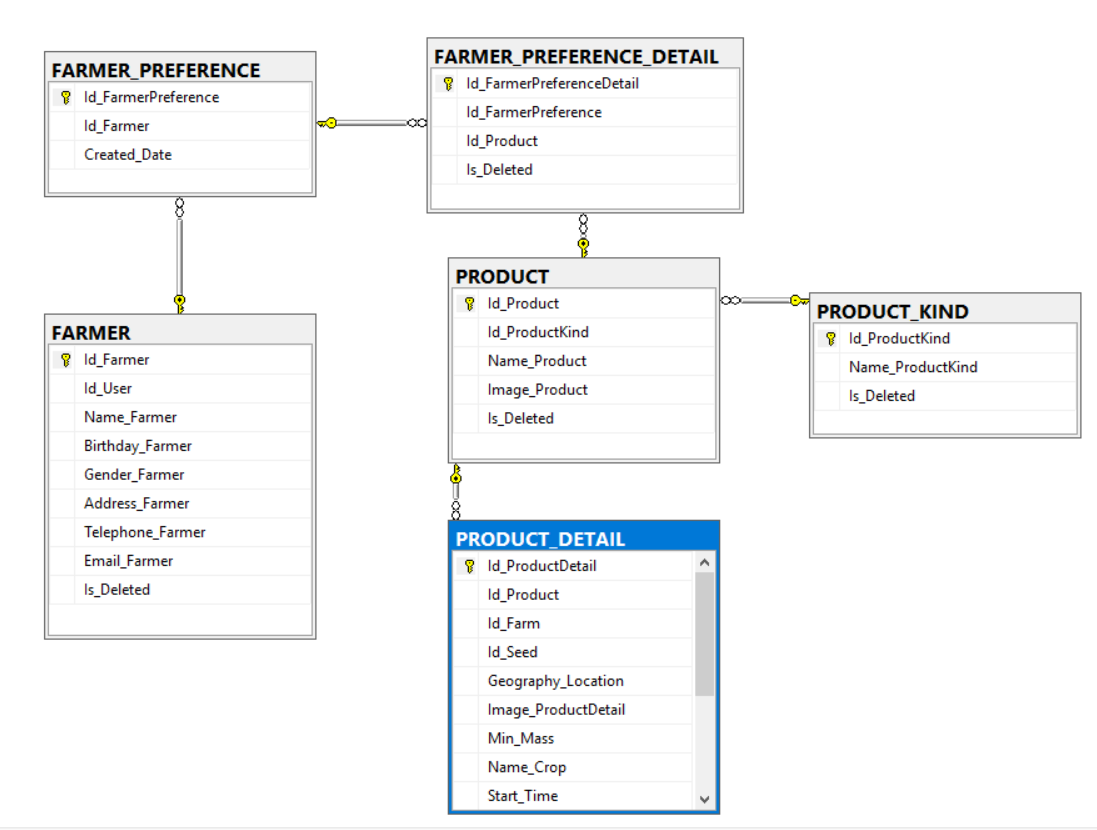
* Bảng USER\_KIND lưu thông tin các loại người dùng, có ba loại người dùng chính: Admin, Doanh nghiệp và Chủ trang trại
* USER\_AUTHENTICATION lưu thông tin của các tài khoản đã đăng ký, nối khóa ngoại đến bảng TRADER và FARMER



**Hình x** Hiển thị nông sản ưu tiên của doanh nghiệp

Giải thích:

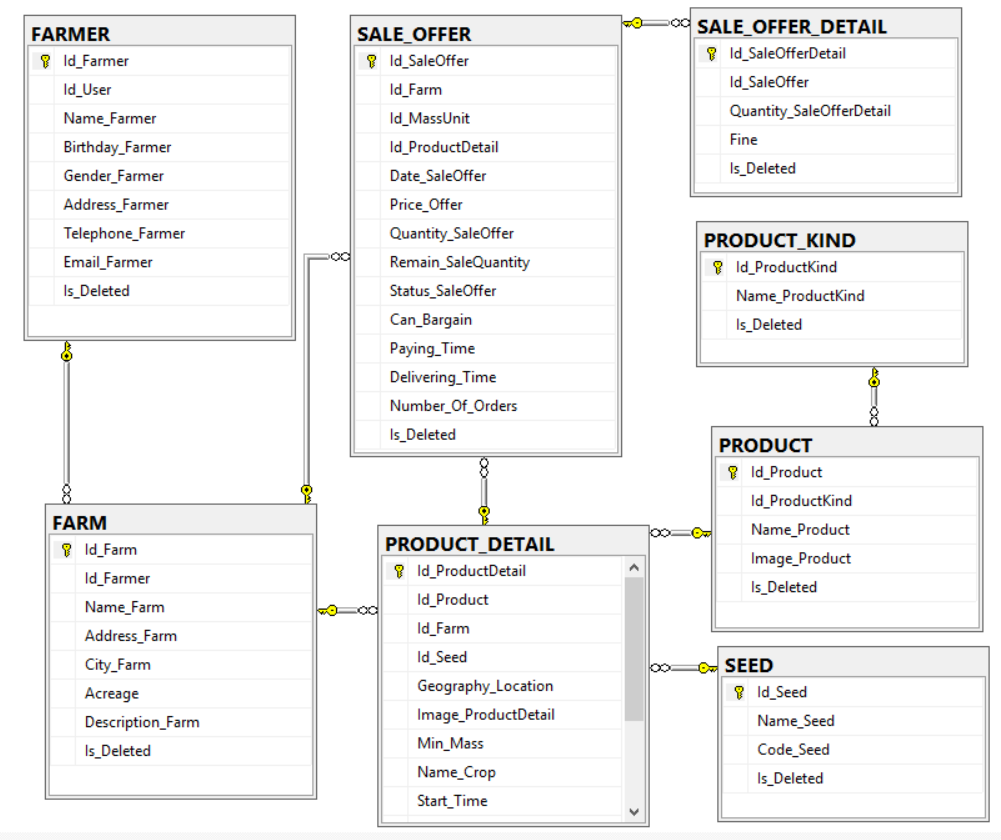
* Sau khi đăng ký tài khoản, TRADER (doanh nghiệp) có thể chọn những nông sản chính mà mình quan tâm. Những thông tin nông sản được lưu trong bảng sẽ ưu tiên hiển thị lên trang chủ.



**Hình x** Hiển thị nông sản ưu tiên của chủ trang trại

Giải thích:

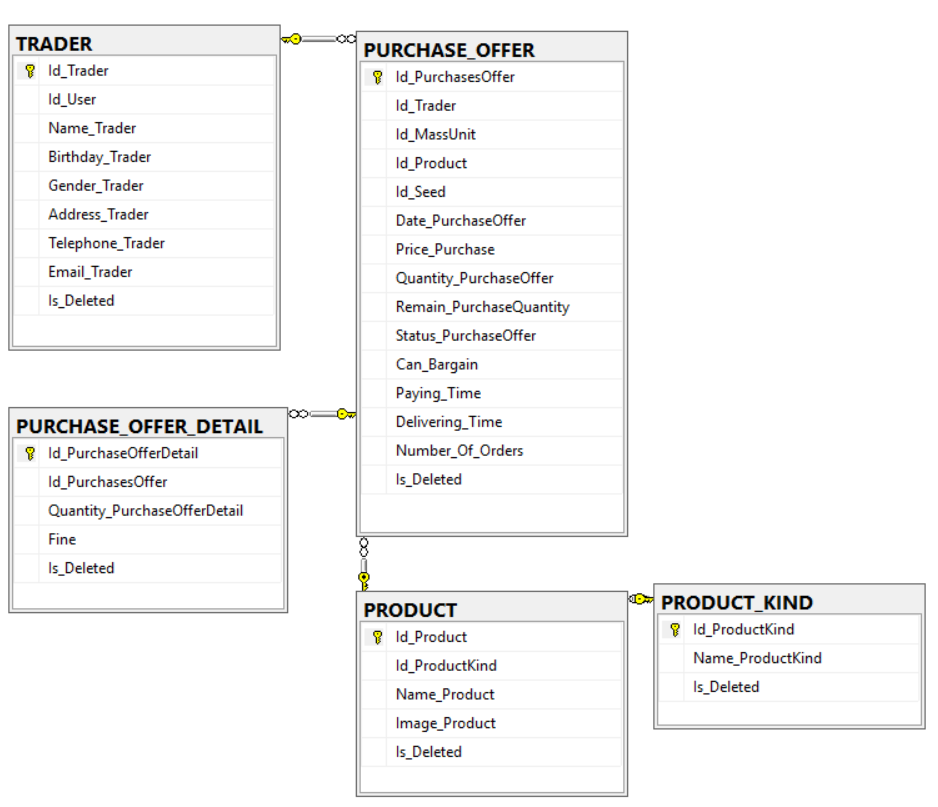
* Ý 1
* Ý 2



**Hình x** Nông trại – Nông sản – Nhu cầu bán

Giải thích

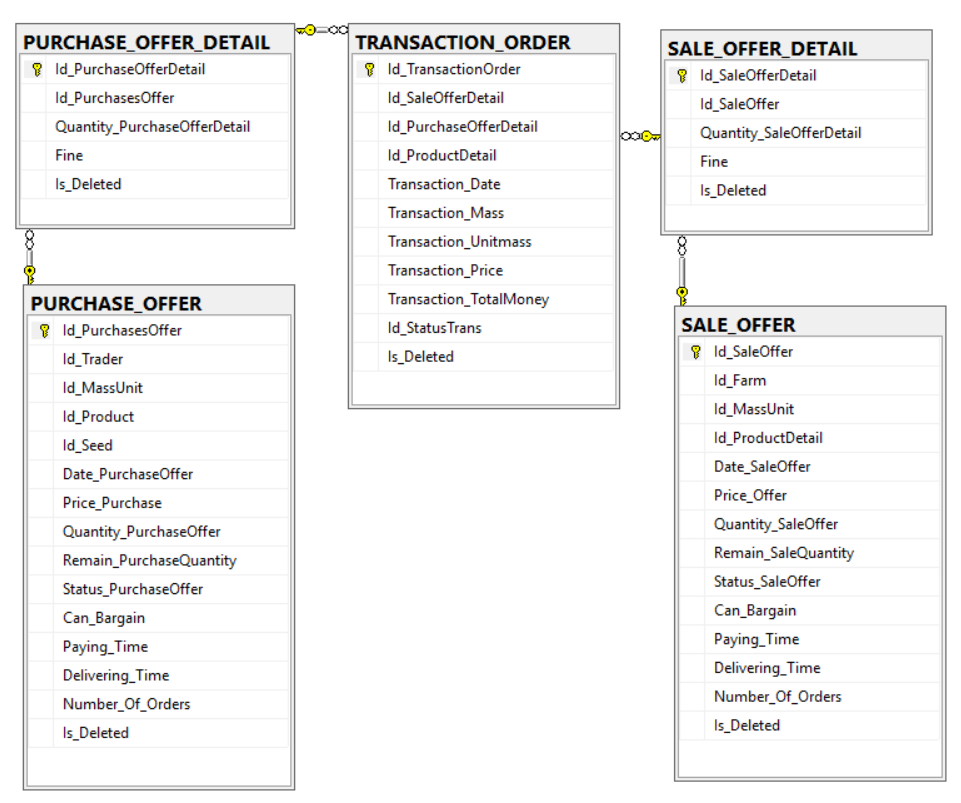
* Một chủ trang trại có thể sở hữu nhiều trang trại, một trang trại chỉ có thể thuộc sở hữu của một người. Trong mỗi trang trại có thể trồng nhiều loại nông sản (giống, mùa vụ, ...) khác nhau.
* Khi đăng bán nông sản, chủ trang trại sẽ đăng bán theo từng nông trại của mình. Thông tin nhu cầu bán sẽ cho biết nông trại nào, bán loại nông sản gì, giống gì, giá bao nhiêu,, vv...



**Hình x** Doanh nghiệp – Nhu cầu mua

Giải thích

* Khi đăng mua, nhu cầu mua sẽ cho biết doanh nghiệp muốn mua loại nông sản nào, giống gì, giá bao nhiêu



**Hình x** Giao dịch

Giải thích

* Ý 1
* Ý 2

## Xây dựng ứng dụng

### Thư viện và công cụ sử dụng

Sinh viên liệt kê các công cụ, ngôn ngữ lập trình, API, thư viện, IDE, công cụ kiểm thử, v.v. mà mình sử dụng để phát triển ứng dụng. Mỗi công cụ phải được chỉ rõ phiên bản sử dụng. SV nên kẻ bảng mô tả tương tự như Bảng 1. Nếu có nhiều nội dung trình bày, sinh viên cần xoay ngang bảng.

**Bảng 1** Danh sách thư viện và công cụ sử dụng

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| IDE lập trình | Visual Studio 2017 | <https://visualstudio.microsoft.com/> |
| Hệ quản trị CSDL | SQL Server 2017 | <https://www.microsoft.com/en-us/sql-server/sql-server-downloads-free-trial> |
| Front-end | Bootstrap, JavaScripts, jQuerry | <https://getbootstrap.com/> |
| Back-end | Nền tảng ASP.NET | <https://dotnet.microsoft.com/apps/aspnet> |
| Test | BitBucket | [https://bitbucket.org](https://bitbucket.org/) |

### Kết quả đạt được

Sinh viên trước tiên mô tả kết quả đạt được của mình là gì, ví dụ như các sản phẩm được đóng gói là gì, bao gồm những thành phần nào, ý nghĩa, vai trò?

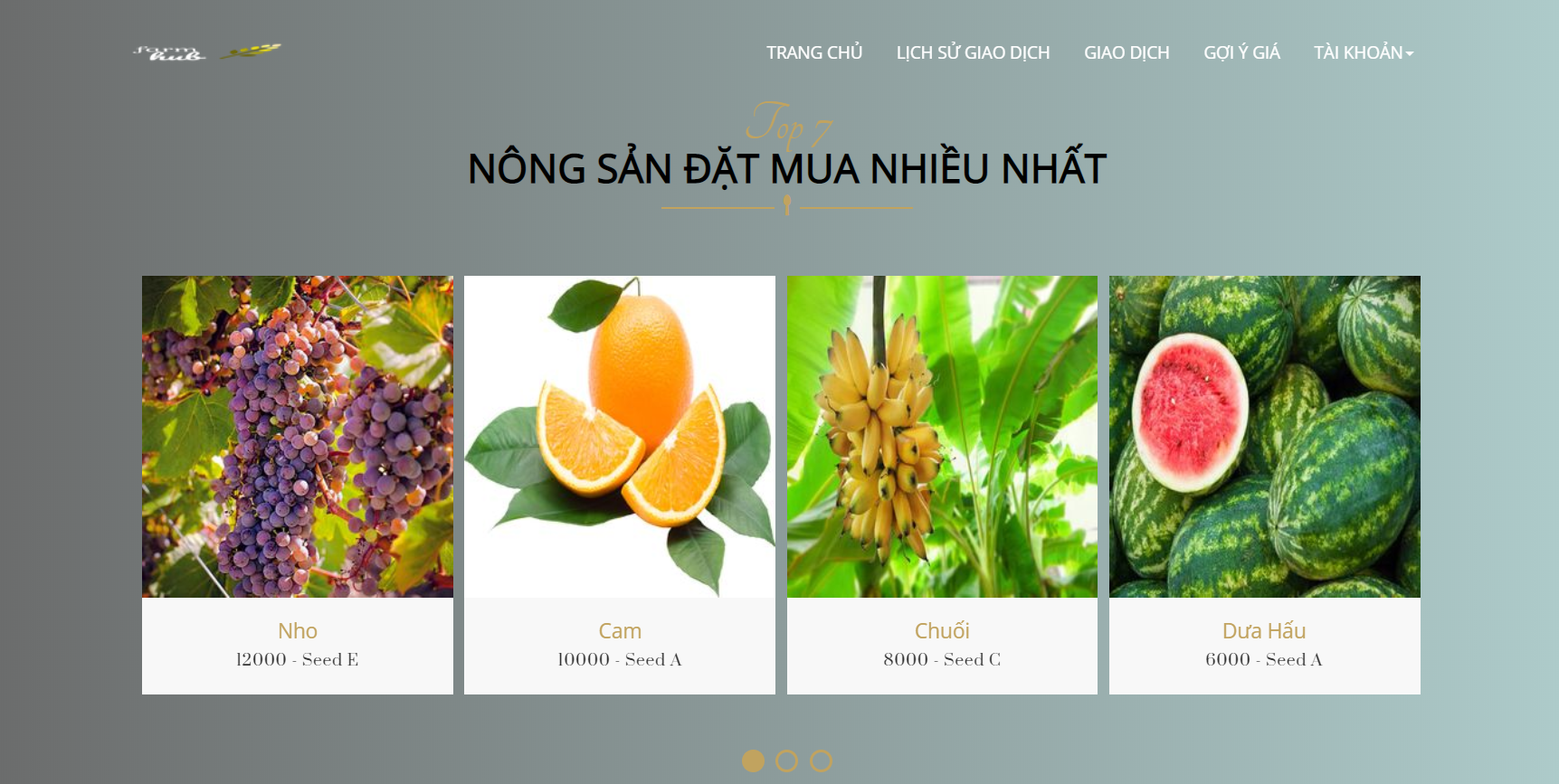
Sinh viên cần thống kê các thông tin về ứng dụng của mình như: số dòng code, số lớp, số gói, dung lượng toàn bộ mã nguồn, dung lượng của từng sản phẩm đóng gói, v.v. Tương tự như phần liệt kê về công cụ sử dụng, sinh viên cũng nên dùng bảng để mô tả phần thông tin thống kê này.

### Minh hoạ các chức năng chính

Sinh viên lựa chọn và đưa ra màn hình cho các chức năng chính, quan trọng, và thú vị nhất. Mỗi giao diện cần phải có lời giải thích ngắn gọn. Khi giải thích, sinh viên có thể kết hợp với các chú thích ở trong hình ảnh giao diện.

Trang chủ:

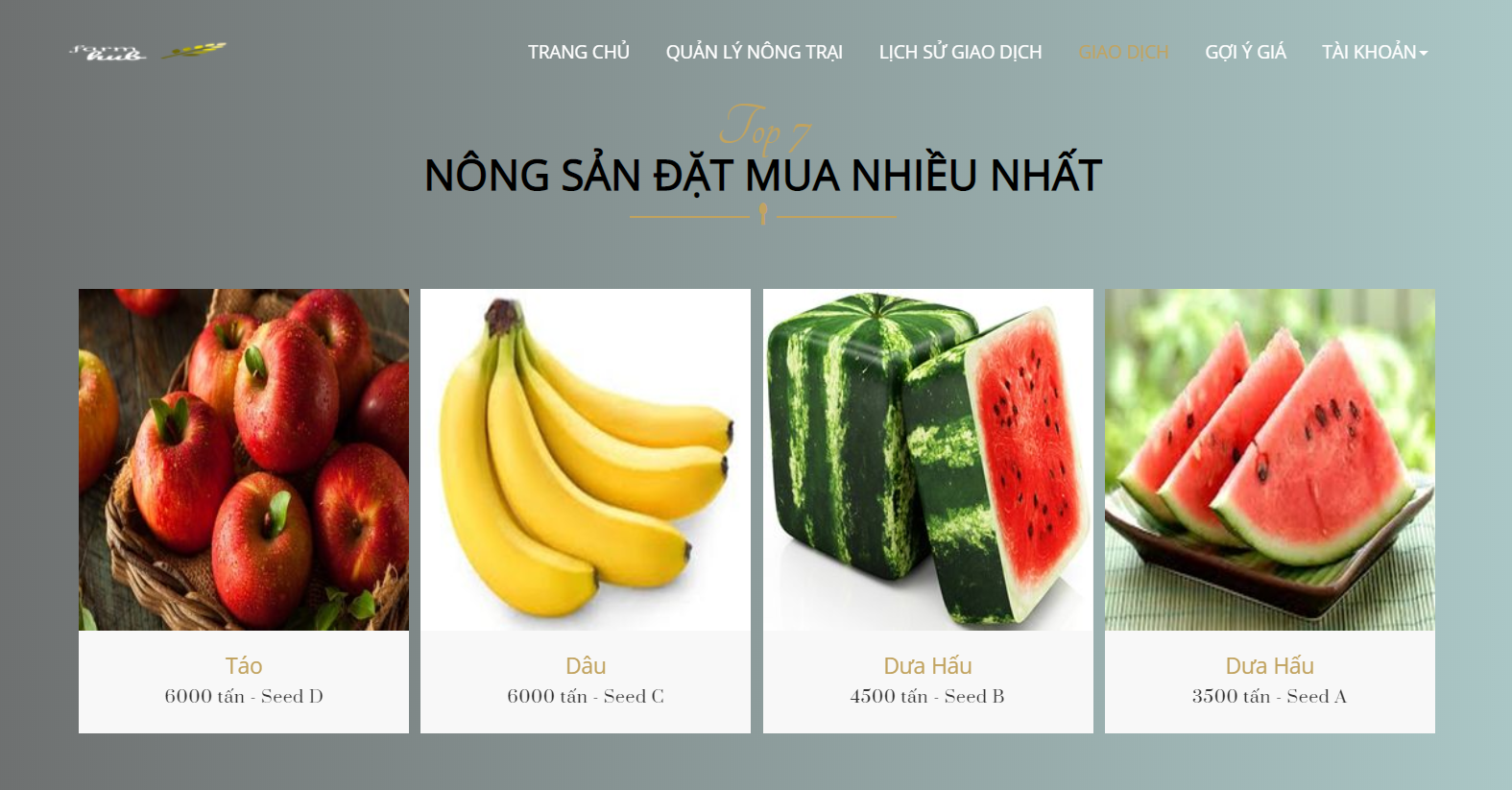
* Khi vừa truy cập website, đây sẽ là trang giao diện mà người dùng nhìn thấy đầu tiên
* Trang chủ chứa các thông tin của thị trường như: top các loại nông sản được mua nhiều nhất, các giao dịch đang diễn ra trên thị trường, vv....



**Hình x** Trang chủ

Trang chủ Chủ trang trại:

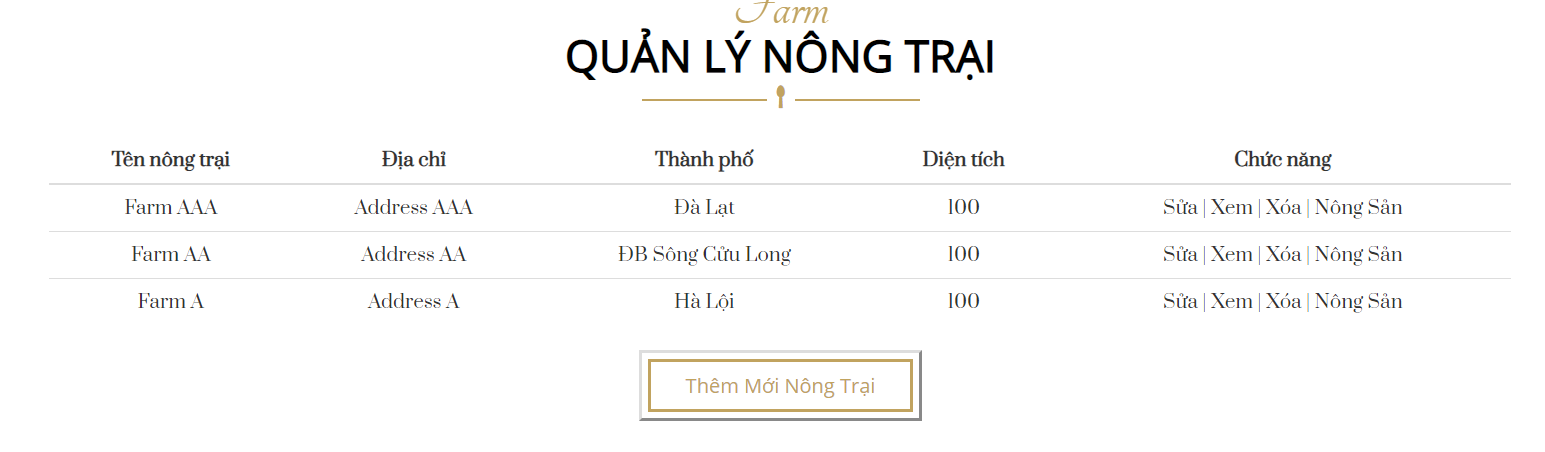
* Sau khi đăng nhập với tư cách là chủ trang trại, đây sẽ là giao diện mà người sử dụng nhìn thấy
* Trang này tập trung các thông tin như: top các loại nông sản đang được mua nhiều nhất, các thông tin về nông sản mà người dùng đã đăng bán cùng với những nhu cầu mua phù hợp



**Hình x** Trang chủ chủ trang trại

Quản lý nông trại (danh sách nông trại):

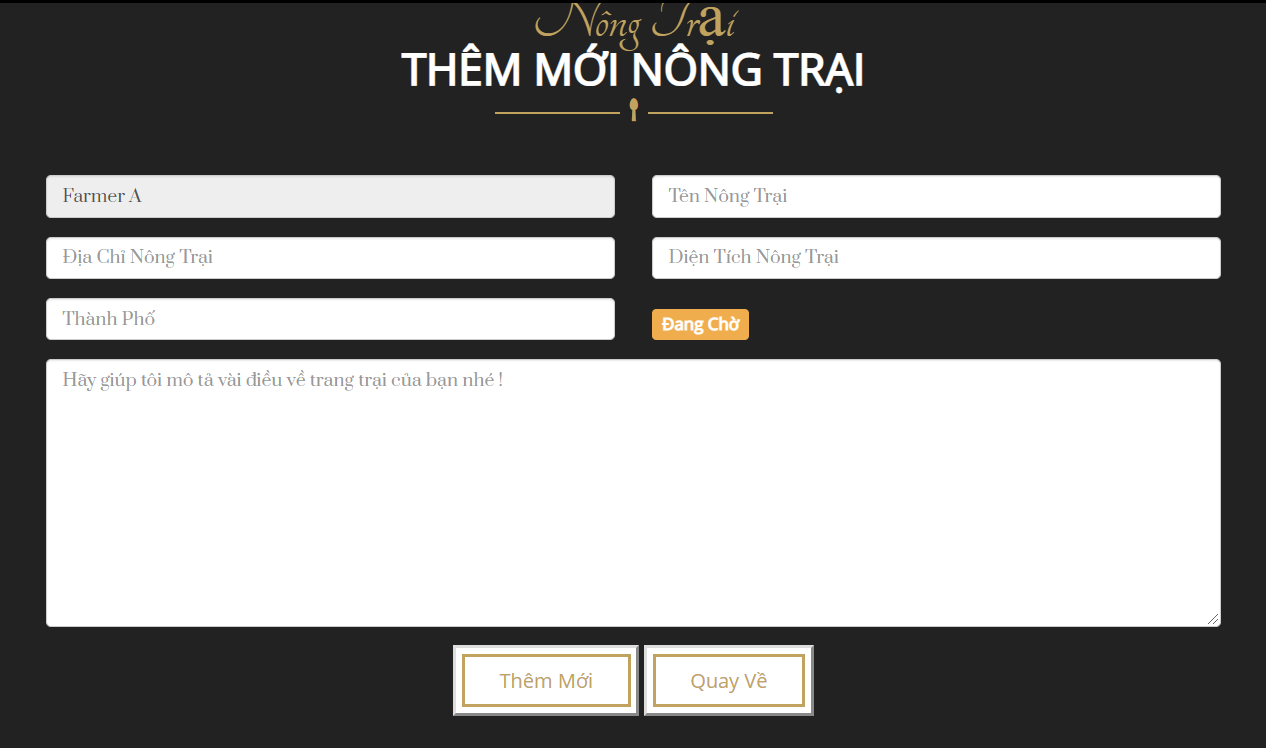
* Một chủ trang trại có thể sở hữu một hoặc nhiều trang trại, trang này cho phép họ quản lý thông tin của những trang trại mà mình đã đăng kí
* Ngoài những thông tin về trang trại như tên trang trại, địa chỉ, diện tích, vv..., mỗi trang trại sẽ đăng ký và quản lý thông tin về những nông sản trồng ở trang trại đó (**Hình x** Danh sách nông sản )



**Hình x** Quản lý nông trại

Thêm mới nông trại

* Đây là màn hình nhập thông tin khi chủ trang trại muốn tạo mới một trang trại
* Các trường thông tin phải được ghi đầy đủ
* Chủ trang trại có thể thêm vài điều mô tả về trang trại của mình



**Hình x** Thêm mới nông trại

Chỉnh sửa nông trại:

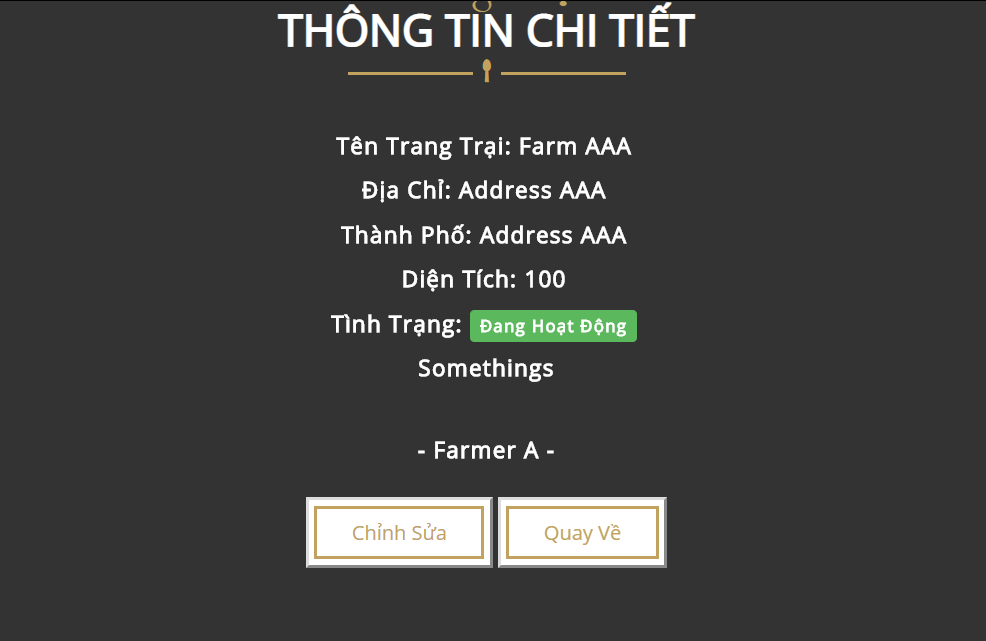
* Đây là màn hình chỉnh sửa thông tin trang trại đã đăng ký
* Nếu một trường dữ liệu không được chỉnh sửa gì, thì sẽ lưu dữ liệu cũ trước đó



**Hình x** Chỉnh sửa thông tin nông trại

Chi tiết nông trại:

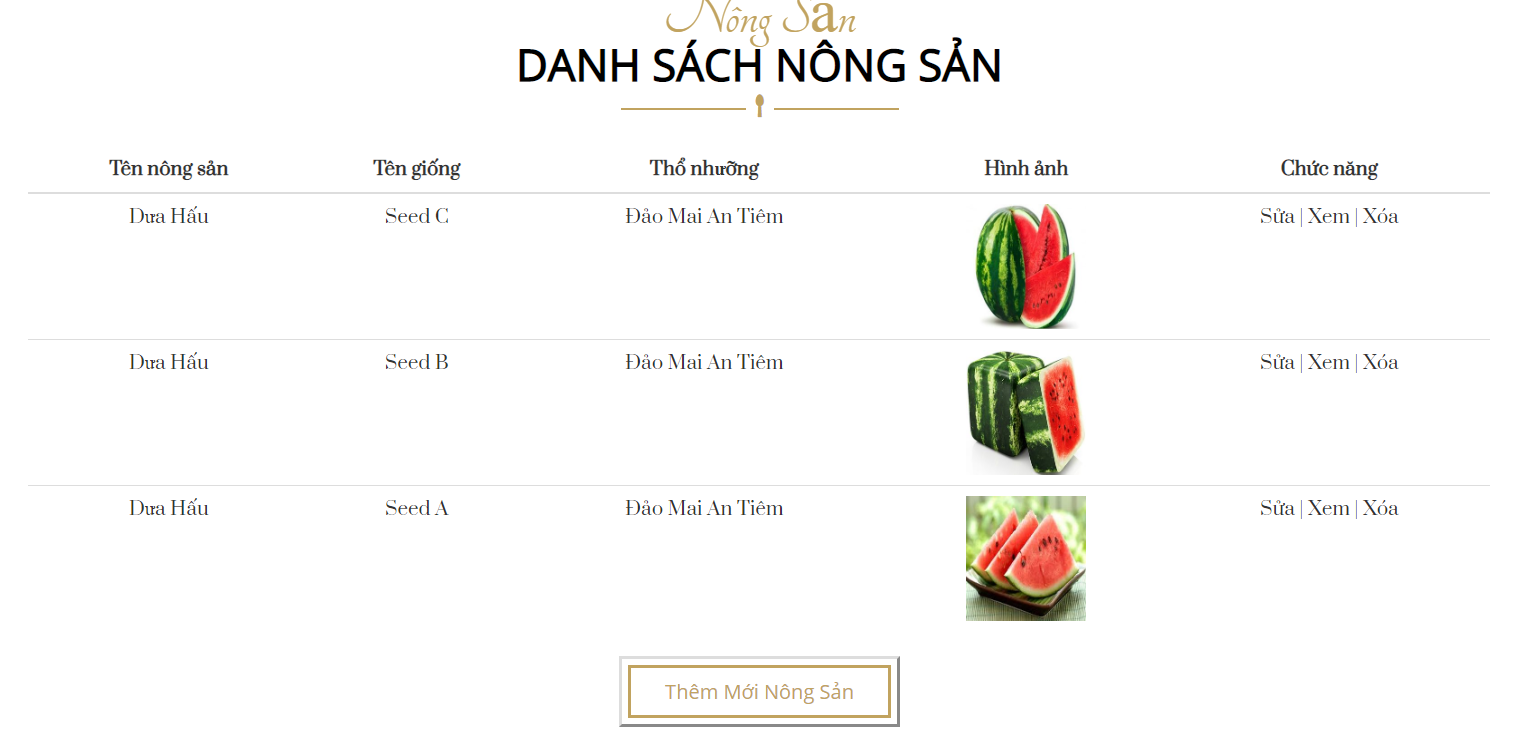
* Đây là màn hình xem thông tin chi tiết về trang trại
* Hiển thị đầy đủ tất cả các thông tin của trang trại



**Hình x** Chi tiết nông trại

Danh sách nông sản

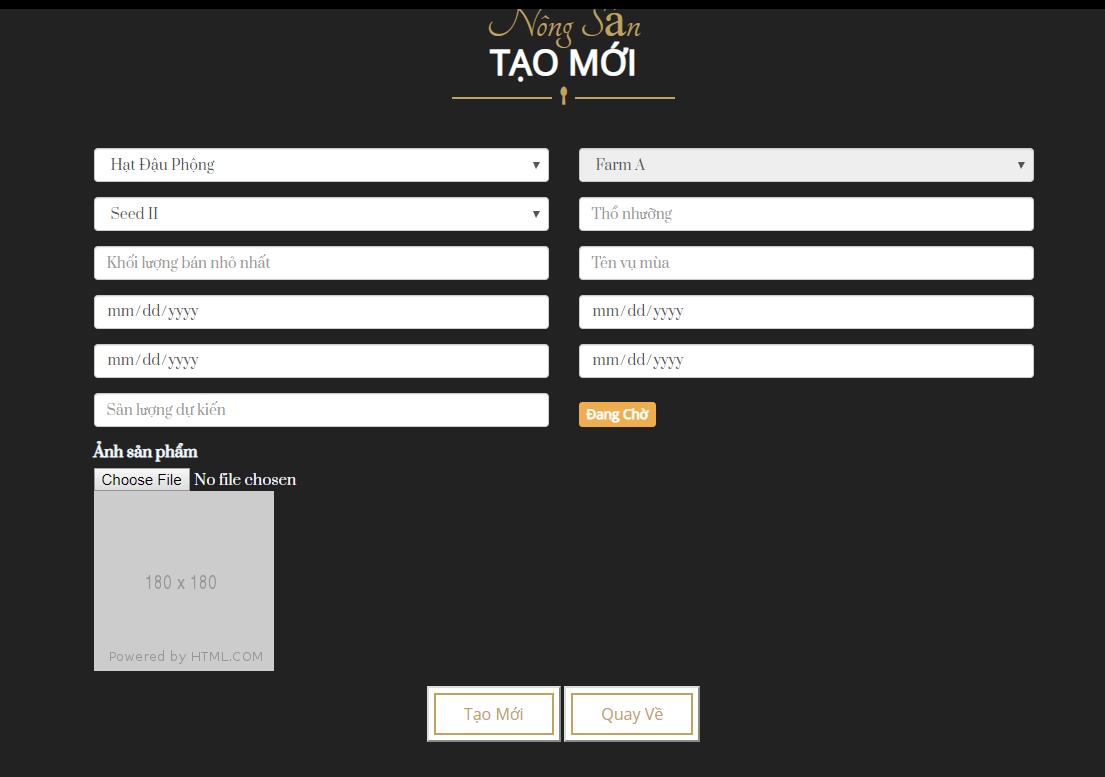
* Mỗi trang trại có thể trồng nhiều loại nông sản khác nhau
* Đây là màn hình danh sách các nông sản được đăng ký là trồng ở một nông trại xác định



**Hình x** Danh sách nông sản

Thêm mới nông sản

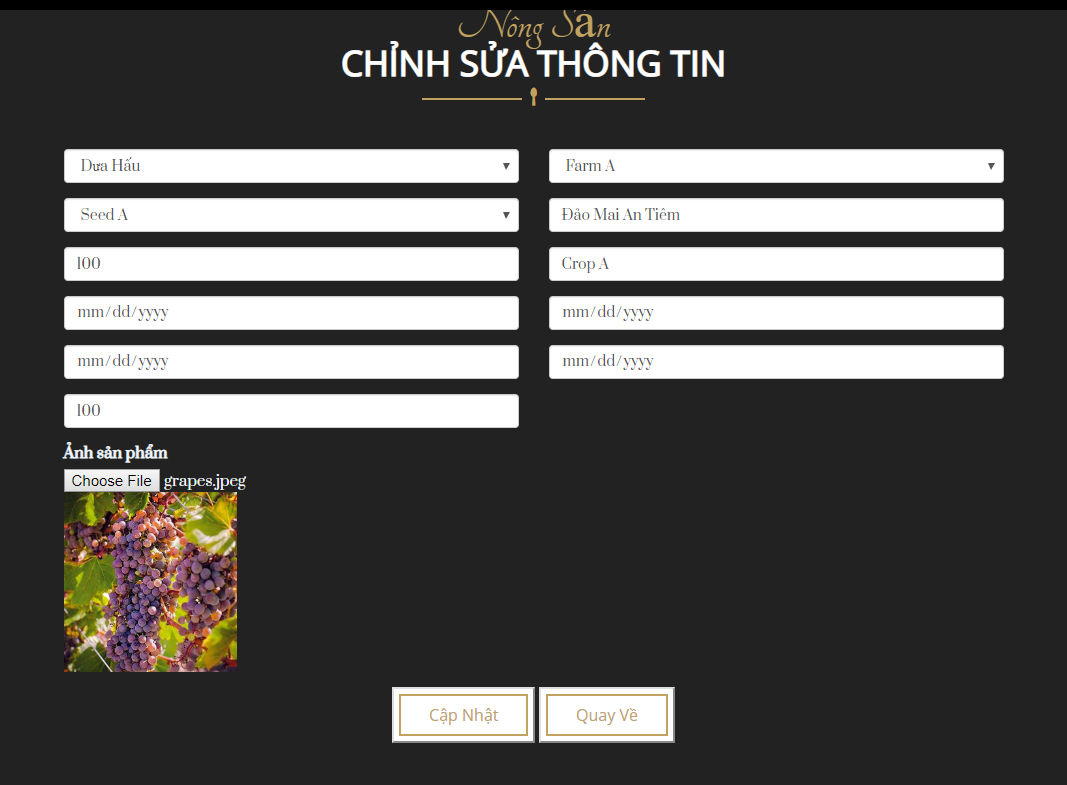
* Đây là màn hình nhập liệu khi tạo mới một nông sản
* Chủ trang trại chọn loại nông sản, giống, và nhập đầy đủ các thông tin khác cũng như chọn hình ảnh



**Hình x** Thêm mới nông sản

Chỉnh sửa nông sản:

* Đây là màn hình chỉnh sửa thông tin nông sản
* Nếu một trường dữ liệu không được chỉnh sửa gì, thì sẽ lưu dữ liệu cũ trước đó



**Hình x** Chỉnh sửa nông sản

Chi tiết nông sản

* Đây là màn hình chi tiết nông sản



**Hình x** Chi tiết nông sản

## Kiểm thử

Phần này có độ dài từ hai đến ba trang. Sinh viên thiết kế các trường hợp kiểm thử cho hai đến ba chức năng quan trọng nhất. Sinh viên cần chỉ rõ các kỹ thuật kiểm thử đã sử dụng. Chi tiết các trường hợp kiểm thử khác, nếu muốn trình bày, sinh viên đưa vào phần phụ lục.

Sinh viên sau cùng tổng kết về số lượng các trường hợp kiểm thử và kết quả kiểm thử. Sinh viên cần phân tích lý do nếu kết quả kiểm thử không đạt.

## Triển khai

Sinh viên trình bày mô hình và/hoặc cách thức triển khai thử nghiệm/thực tế. Ứng dụng của sinh viên được triển khai trên server/thiết bị gì, cấu hình như thế nào. Kết quả triển khai thử nghiệm nếu có (số lượng người dùng, số lượng truy cập, thời gian phản hồi, phản hồi người dùng, khả năng chịu tải, các thống kê, v.v.)

# Các giải pháp và đóng góp nổi bật

Chương này có độ dài tối thiểu 5 trang, tối đa không giới hạn. Sinh viên cần trình bày tất cả những nội dung đóng góp mà mình thấy tâm đắc nhất trong suốt quá trình làm ĐATN. Đó có thể là một loạt các vấn đề khó khăn mà sinh viên đã từng bước giải quyết được, là giải thuật cho một bài toán cụ thể, là giải pháp tổng quát cho một lớp bài toán, hoặc là mô hình/kiến trúc hữu hiệu nào đó được sinh viên thiết kế.

Chương này là **cơ sở quan trọng** để các thầy cô đánh giá sinh viên. Vì vậy, sinh viên cần phát huy tính sáng tạo, khả năng phân tích, phản biện, lập luận, tổng quát hóa vấn đề và tập trung viết cho thật tốt.

Mỗi giải pháp hoặc đóng góp của sinh viên cần được trình bày trong một mục độc lập bao gồm ba mục con: (i) dẫn dắt/giới thiệu về bài toán/vấn đề, (ii) giải pháp, và (iii) kết quả đạt được (nếu có).

Sinh viên lưu ý **không trình bày lặp lại nội dung**. Những nội dung đã trình bày chi tiết trong các chương trước không được trình bày lại trong chương này. Vì vậy, với nội dung hay, mang tính đóng góp/giải pháp, sinh viên chỉ nên tóm lược/mô tả sơ bộ trong các chương trước, đồng thời tạo tham chiếu chéo tới đề mục tương ứng trong Chương 5 này. Chi tiết thông tin về đóng góp/giải pháp được trình bày trong mục đó.

Ví dụ, trong Chương 4, sinh viên có thiết kế được kiến trúc đáng lưu ý gì đó, là sự kết hợp của các kiến trúc MVC, MVP, SOA, v.v. Khi đó, sinh viên sẽ chỉ mô tả ngắn gọn kiến trúc đó ở Chương 4, rồi thêm các câu có dạng: “Chi tiết về kiến trúc này sẽ được trình bày trong phần 5.1”.

# Kết luận và hướng phát triển

## Kết luận

Sinh viên so sánh kết quả nghiên cứu hoặc sản phẩm của mình với các nghiên cứu hoặc sản phẩm tương tự.

Sinh viên phân tích trong suốt quá trình thực hiện ĐATN, mình đã làm được gì, chưa làm được gì, các đóng góp nổi bật là gì, và tổng hợp những bài học kinh nghiệm rút ra nếu có.

Meicai – web bán rau của Trung Quốc

## Hướng phát triển

Trong phần này, sinh viên trình bày định hướng công việc trong tương lai để hoàn thiện sản phẩm hoặc nghiên cứu của mình.

Trước tiên, sinh viên trình bày các công việc cần thiết để hoàn thiện các chức năng/nhiệm vụ đã làm. Sau đó sinh viên phân tích các hướng đi mới cho phép cải thiện và nâng cấp các chức năng/nhiệm vụ đã làm.

Trong quá trình sản xuất, trái cây và rau quả cần phải có chứng chỉ “an toàn nông nghiệp” (GAP – Good Agricultural Practices) và rất nhiều chứng chỉ khác như chứng chỉ xác nhận nguồn gốc giống (GMO), chứng chỉ báo cáo chất lượng (hàm lượng protein, chống oxy hóa, vitamine, đồng bộ về giống, kích cỡ, màu sắc). Để giải quyết vấn đề này, có thể kết hợp IOT vào để theo dõi quá trình sản xuất nông sản (chăm tưới, bón phân,vv...) và các thông tin khác như nhiệt độ, độ ẩm, sâu hại, vv...

# Tài liệu tham khảo

Lưu ý: Sinh viên không được đưa bài giảng/slide, các trang Wikipedia, hoặc các trang web thông thường làm tài liệu tham khảo.

Một trang web được phép dùng làm tài liệu tham khảo **chỉ khi** nó là công bố chính thống của cá nhân hoặc tổ chức nào đó. Ví dụ, trang web đặc tả ngôn ngữ XML của tổ chức W3C https://www.w3.org/TR/2008/REC-xml-20081126/ là TLTK hợp lệ.

Có năm loại tài liệu tham khảo mà sinh viên phải tuân thủ đúng quy định về cách thức liệt kê thông tin như sau. Lưu ý: các phần văn bản trong cặp dấu < > dưới đây chỉ là hướng dẫn khai báo cho từng loại tài liệu tham khảo; sinh viên cần xóa các phần văn bản này trong ĐATN của mình.

<**Bài báo đăng trên tạp chí khoa học**: Tên tác giả, tên bài báo, tên tạp chí, volume, từ trang đến trang (nếu có), nhà xuất bản, năm xuất bản>

1. Hovy E. H., Automated Discourse Generation Using Discourse Structure Relations, Artificial Intelligence, Elsevier Science Publishers, 63: 341-385, 1993.

<**Sách**: Tên tác giả, tên sách, volume (nếu có), lần tái bản (nếu có), nhà xuất bản, năm xuất bản>

1. Peterson L. L. and Davie B. S., Computer Networks: A Systems Approach, 2nd ed., Mogran-Kaufmann, 1999.
2. Nguyễn Thúc Hải, Mạng máy tính và các hệ thống mở, Nhà xuất bản giáo dục, 1999.

<**Tập san Báo cáo Hội nghị Khoa học**: Tên tác giả, tên báo cáo, tên hội nghị, ngày (nếu có), địa điểm hội nghị, năm xuất bản>

1. Poesio M. and Di Eugenio B., Discourse Structure and Anaphoric Accessibility, In Proc. of the ESSLLI Workshop on Information Structure, Discourse Structure and Discourse Semantics, Helsinki, 2001.

<**Đồ án tốt nghiệp, Luận văn Thạc sĩ, Tiến sĩ**: Tên tác giả, tên đồ án/luận văn, loại đồ án/luận văn, tên trường, địa điểm, năm xuất bản>

1. Knott D., A Data-Driven Methodology for Motivating a Set of Coherence Relations, Ph.D. Thesis, University of Edinburgh, UK, 1996.

<**Tài liệu tham khảo từ Internet**: Tên tác giả (nếu có), tựa đề, cơ quan (nếu có), địa chỉ trang web, thời gian lần cuối truy cập trang web>

1. Berners-Lee T., Hypertext Transfer Protocol (HTTP), CERN, ftp:/info.cern.ch/pub/www/doc/http-spec.txt.Z, last visited May 2010.
2. Princeton University, WordNet, http://www.cogsci.princeton.edu/~wn/index.shtml, last visited May 2010.

# Phụ lục

Phần phụ lục là không bắt buộc. Nếu sinh viên không có nhu cầu trình bày thêm, có thể xóa bỏ phần này. Lưu ý là phần phụ lục chỉ được đánh chỉ mục đến cấp 2, sinh viên không được phép chia nhỏ hơn nữa.

Hướng dẫn viết đồ án tốt nghiệp

Quy định chung

Dưới đây là một số quy định và hướng dẫn viết đồ án tốt nghiệp mà bắt buộc sinh viên phải **đọc kỹ và tuân thủ nghiêm ngặt**.

Sinh viên cần đảm bảo tính thống nhất toàn báo cáo (font chữ, căn dòng hai bên, hình ảnh, bảng, margin trang, đánh số trang, v.v.). Để làm được như vậy, sinh viên chỉ cần sử dụng các định dạng theo đúng template ĐATN này. Khi paste nội dung văn bản từ tài liệu khác của mình, sinh viên cần chọn kiểu Copy là “Text Only” để định dạng văn bản của template không bị phá vỡ/vi phạm.

Tuyệt đối cấm sinh viên đạo văn. Sinh viên cần ghi rõ nguồn cho tất cả những gì không tự mình viết/vẽ lên, bao gồm các câu trích dẫn, các hình ảnh, bảng biểu, v.v. Khi bị phát hiện, sinh viên sẽ **không được phép bảo vệ ĐATN**.

Tất cả các hình vẽ, bảng biểu, công thức, và tài liệu tham khảo trong ĐATN nhất thiết phải được SV giải thích và tham chiếu tới ít nhất một lần. Không chấp nhận các trường hợp sinh viên đưa ra hình ảnh, bảng biểu tùy hứng và không có lời mô tả/giải thích nào.

Sinh viên tuyệt đối không trình bày ĐATN theo kiểu viết ý hoặc gạch đầu dòng. ĐATN không phải là một slide thuyết trình; khi người đọc không hiểu sẽ không có ai giải thích hộ. Sinh viên cần viết thành các đoạn văn và phân tích, diễn giải đầy đủ, rõ ràng. Câu văn cần đúng ngữ pháp, đầy đủ chủ ngữ, vị ngữ và các thành phần câu.

Khi thực sự cần liệt kê, sinh viên nên liệt kê theo phong cách khoa học với các ký tự La Mã. Ví dụ, nhiều sinh viên luôn cảm thấy hối hận vì (i) chưa cố gắng hết mình, (ii) chưa sắp xếp thời gian học/chơi một cách hợp lý, (iii) chưa tìm được người yêu để chia sẻ quãng đời sinh viên vất vả, và (iv) viết ĐATN một cách cẩu thả.

Trong một số trường hợp nhất thiết phải dùng các bullet để liệt kê, sinh viên cần thống nhất Style cho toàn bộ các bullet các cấp mà mình sử dụng đến trong báo cáo. Nếu dùng bullet cấp 1 là hình tròn đen, toàn bộ báo cáo cần thống nhất cách dùng như vậy; ví dụ như sau:

* Đây là mục 1 – Thực sự không còn cách nào khác tôi mới dùng đến việc bullet trong báo cáo.
* Đây là mục 2 – Nghĩ lại thì tôi có thể không cần dùng bullet cũng được. Nên tôi sẽ xóa bullet và tổ chức lại hai mục này trong báo cáo của mình cho khoa học hơn. Tôi muốn thầy cô và người đọc cảm nhận được tâm huyết của tôi trong từng trang báo cáo ĐATN.

Tạo đề mục

Đề mục giúp tạo bố cục cho tài liệu. Để các tính năng tự động – ví dụ tính năng cập nhật mục lục, hoặc tính năng tham chiếu chéo – của Word hoạt động được, sinh viên cần tuân thủ theo các style đã tạo trong tài liệu này. Để hiển thị các style này, sinh viên vào tab Home trong thanh Ribbon của Word.

Để tạo đề mục cấp 1, 2, 3, 4, 5, sinh viên gõ tiêu đề cho đề mục của mình rồi chọn các style là Heading 1, 2, 3, 4, 5 tương ứng. Sinh viên hạn chế dùng tới đề mục cấp 4, và phải trong trường hợp thực sự cần thiết mới dùng đến đề mục cấp 5.

Phần phụ lục chỉ cho phép có hai cấp tiêu đề. Hai style tương ứng với hai cấp này là “Heading 7, Phụ lục cấp 1” và “Heading 8, Phụ lục cấp 2”.

Bảng biểu

Sinh viên lưu ý không để bảng tràn ra lề (margin) trên, dưới, trái hoặc phải của trang. Do không gian nhỏ hẹp, bảng nên có font là 12pt, nhỏ hơn một chút so với font thông thường (13pt) của ĐATN. Độ dãn dòng của bảng nên là 1 line. Căn lề bảng là căn giữa, nhưng nội dung văn bản trong bảng nên được căn lề trái.

Sinh viên có thể viết tắt các từ trong bảng để tiết kiệm không gian nhưng phải giải thích các từ viết tắt này ở phần Chú thích bảng. Ví dụ áp dụng được minh họa trong Bảng 2.

**Bảng 2** Ví dụ sử dụng bảng

Chú thích: Y: Year; RS: Risk Set; G: Graduated; AB: Academically Excluded;C: Censored; HRG: Hazard Ratio – Graduated

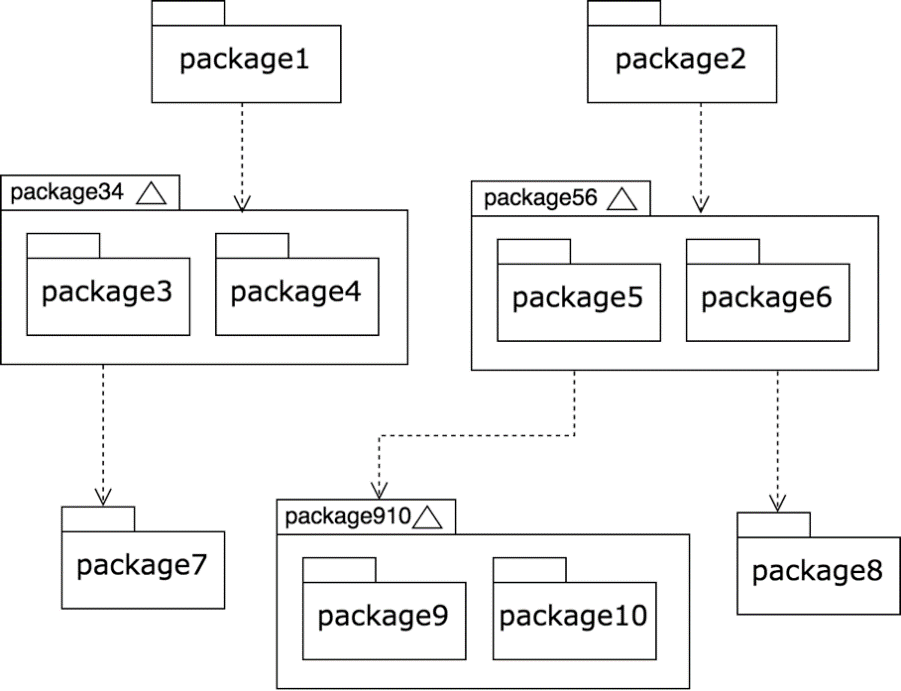
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Y** | **RS** | **G** | **AE** | **C** | **HRG**  **(%)** |
| **1** | 11.959 | 0 | 725 | 1619 | 0,0 |
| **2** | 10.457 | 0 | 474 | 1513 | 0,0 |
| **3** | 7.365 | 1213 | 335 | 966 | 16,7 |
| **4** | 900 | 599 | 145 | 405 | 55,3 |

Sinh viên được tùy ý lựa chọn định dạng (template) cho các bảng trong ĐATN của mình, nhưng phải đảm bảo tính nhất quán trên toàn ĐATN. Template bảng phải đảm bảo phần heading cột trong bảng có font in đậm và nổi bật (highlighted) hơn so với các nội dung khác trong bảng.

Vì bảng có thể kéo dài nhiều trang, tiêu đề của bảng nên để ở phần đầu của bảng. Sinh viên không thêm tiêu đề bảng bằng tay. Để thêm tiêu đề bảng tự động, sinh viên nhấn chuột phải vào bảng, chọn “Insert Caption”, chọn “Label” là “Bảng”, rồi nhấn nút “OK”. Sau đó, sinh viên nhập vào nội dung tiêu đề và căn chỉnh “Center” cho tiêu đề này. Lưu ý, sinh viên cần bôi đậm bằng tay cụm từ **Bảng n**. Kết quả thu được có dạng như sau “**Bảng 2** Ví dụ sử dụng bảng”. Sinh viên nên xoay ngang trang giấy trong trường hợp bảng có nhiều cột với nhiều nội dung văn bản.

Hình vẽ

Tương tự như bảng, sinh viên không được để hình vẽ tràn lề trang. Căn lề cho hình vẽ là căn giữa (Center). Cách thêm tiêu đề hình vẽ tương tự như cách thêm tiêu đề bảng, nhưng sinh viên chọn “Label” là “Hình” thay vì “Bảng”.



**Hình 3** Ví dụ hình vẽ

Tiêu đề hình vẽ phải đặt ở dưới hình vẽ. Nếu hình vẽ được copy từ trên mạng, sinh viên bắt buộc phải ghi rõ nguồn. Sinh viên nên thống nhất công cụ sử dụng và style cho hình vẽ trong toàn ĐATN. Các chi tiết trong hình vẽ phải được bố trí gọn gàng; chữ trong hình phải đảm bảo nhìn được rõ nét khi in báo cáo trên giấy A4. Khi resize ảnh, cần giữ nguyên tỷ lệ dài rộng, tránh làm méo hoặc vỡ hình. Ví dụ hình vẽ được minh họa trong Hình 3.

Tài liệu tham khảo

Sinh viên cần hạn chế tối đa dùng trang Web làm tài liệu tham khảo. Chỉ chấp nhận trang Web làm TLTK khi trang đó là nơi công bố chính thức của tổ chức hoặc cá nhân nào đó. Ví dụ, trang web đặc tả ngôn ngữ XML của tổ chức W3C https://www.w3.org/TR/2008/REC-xml-20081126/ là TLTK hợp lệ.

Khi giới thiệu về công cụ, API, thư viện, hoặc nền tảng nào đó, sinh viên có thể đưa ra địa chỉ URL của các tiện ích này. Sinh viên lưu ý địa chỉ URL đó không phải là tài liệu tham khảo. Trong các trường hợp tương tự như vậy, sinh viên nên tạo “Footnote”. Sinh viên tạo “Footnote” bằng cách vào mục “References”, chọn “Insert Footnote”. Ví dụ tạo Footnote như sau: TensorFlow[[1]](#footnote-1) là nền tảng học máy mã nguồn mở đang được sử dụng rộng rãi hiện nay. Lưu ý: số Footnote phải đặt sát với từ được mô tả. Như trong ví dụ trên, số 1 được đặt ngay cạnh chữ TensorFlow (không có dấu cách).

Công thức toán học

Giống như bảng, hình vẽ, và tài liệu tham khảo, công thức toán học cần được đánh số, giải thích, và tham chiếu đầy đủ.

Để thêm tiêu đề (caption) cho công thức, sinh viên đặt con trỏ văn bản vào dòng văn bản dưới công thức. Sau đó vào menu “Insert” (lưu ý là menu Insert, không phải là tab Insert trong thanh Ribbon) và chọn mục “Caption”. Từ Popup Menu hiện ra, sinh viên chọn “Label” là “Công thức”, rồi nhấn nút “OK”. Sinh viên bôi đậm chữ **Công thức n**, và căn tiêu đề ra giữa. Công thức 1 là ví dụ mẫu cho sinh viên tham khảo.

**Công thức 1** Khai triển Newton

Tham chiếu chéo

Tham chiếu chéo (Cross-reference) là tiện ích hữu hiệu cho người viết báo cáo. Nó giúp tạo các liên kết tham chiếu (hyperlink) tới các hình ảnh, bảng biểu, tài liệu tham khảo, và các đề mục một cách tự động. Ví dụ, ngay trong câu này, một tham chiếu đã được tạo ra tới mục 2.1. Người đọc dễ dàng nhấp chuột vào liên kết 2.1 để ngay lập tức chuyển đến mục đó.

Để tạo tham chiếu chéo tới các đề mục (heading), sinh viên vào tab “References”, rồi tìm và nhấn chọn “Cross-refrence”. SV chọn “Refrence type” là “Heading” và chọn “Insert Reference to” là “Heading number (no context)”. Sau đó, SV chọn phần đề mục muốn tham chiếu rồi bấm “Insert”.

Để tạo tham chiếu chéo tới các hình vẽ, bảng biểu và công thức, sinh viên cũng vào tab “References” và chọn “Cross-refrence”. SV chọn “Refrence type” là “Hình”, “Bảng”, hoặc “Công thức’. SV chọn “Insert Reference to” là “Only label and number”, trỏ đến phần muốn tham chiếu rồi bấm “Insert”. Nếu font chữ trong liên kết tham chiếu tạo ra được in đậm (bold), SV chuyển về dạng thường cho chuẩn tắc.

Thực hiện tương tự các bước mô tả ở trên, sinh viên có thể tạo tham chiếu chéo tới các tài liệu tham khảo. Sinh viên chọn “Refrence type” là “Numbered item”, chọn “Insert Reference to” là “Paragraph number”, trỏ đến phần tài liệu tham khảo muốn tham chiếu rồi bấm “Insert”. Ví dụ, tham chiếu chéo tới tài liệu tham khảo [2], [3], [4] đã được tạo.

Cập nhật mục lục và tham chiếu chéo

Trong suốt quá trình viết ĐATN, sinh viên sẽ tạo ra nhiều xáo trộn như thay đổi vị trí hình và bảng, thay đổi thứ tự tài liệu tham khảo, thay đổi tên đề mục, v.v. Vì vậy, để hoàn thành ĐATN, sinh viên cần cập nhật lại các thành phần mục lục, danh sách hình ảnh/bảng/công thức và tham chiếu chéo được tạo tự động trong ĐATN của mình.

Để cập nhật các thành phần này, sinh viên bấm Ctrl+A để chọn toàn báo cáo, nhấn chuột phải và chọn “Update Field”, rồi liên tục chọn mục “Entire table” khi được hỏi trong hộp thoại Popup Menu. Sau đó sinh viên search chữ “Error” trên toàn báo cáo để kiểm tra xem có lỗi đánh chỉ mục hoặc lỗi tham chiếu nào không. Đồng thời, sinh viên nên tự soát lại bằng tay toàn bộ nội dung quyển ĐATN để tránh mọi sai sót.

In quyển đồ án tốt nghiệp

Do hiện nay có nhiều phiên bản Word cho nhiều nền tảng máy tính, sinh viên nhất thiết phải xuất ĐATN ra định dạng PDF rồi mang tới cửa hàng in ấn để tránh sai sót.

Sinh viên chỉ cần in bìa giấy cứng mà không cần đóng quyển để tiết kiệm chi phí và giúp việc vận chuyển ĐATN dễ dàng hơn.

Quyển ĐATN nên được in một mặt trên các trang giấy A4.

Đặc tả use case

Nếu trong nội dung chính không đủ không gian cho các use case khác (ngoài các use case nghiệp vụ chính) thì đặc tả thêm cho các use case đó ở đây.

Đặc tả use case “Thống kê tình hình mượn sách”

Cách trình bày tương tự như phần hướng dẫn ở mục 2.3.

Đặc tả use case “Đăng ký làm thẻ mượn”

Cách trình bày tương tự như phần hướng dẫn ở mục 2.3.

Công nghệ sử dụng

Công nghệ bảo mật dữ liệu

Công nghệ blockchain

Thiết kế gói

Thiết kế gói cho kiến trúc tổng quan

Thiết kế gói cho chức năng “Trả sách”

Thiết kế lớp

1. https://www.tensorflow.org/, lần truy cập cuối: 28/06/2018 [↑](#footnote-ref-1)