**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**



**Báo cáo**

**Project III**

**Đề tài: Xây dựng hệ thống quản lí chuỗi cafe**

**Trịnh Phú Quang - 20205219**

Quang.tp205219@sis.hust.edu.vn

**Ngành Công nghệ thông tin**

**Giáo viên hướng dẫn:** **Lê Đức Trung**

**HÀ NỘI, 09/2023**

# 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.

Project 3 là môn để tạo tiền đề cho sinh viên Trịnh Phú Quang tìm hiểu về ý tưởng cho đồ án tốt nghiệp. Kế thừa từ Project 2 của sinh viên là phân tích và thiết kế hệ thống quản lí chuỗi café. Sinh viên chọn đề tài này vì đề tài mang tính thực tiễn, hiện tại ngoài thị trường dù đã có một vài sản phẩm giúp giải quyết vấn đề nhưng vẫn còn một vài vấn đề tồn đọng. Vì vậy sinh viên mong đợi project 3 sẽ là nền tảng để xây dựng hệ thống nhằm khắc phục các nhược điểm của các sản phẩm hiện có.

Giải pháp hiện tại cho việc quản lí chuỗi café là xây dựng một hệ thống quản lí tiết kiệm, dễ sử dụng, dễ kiểm soát và miễn phí. Trong môn Project 3 này sinh viên đã xây dựng nên những viên gạch đầu tiên trong việc xây dựng hệ thống

# Mục lục

[Lời cam kết ii](#_Toc44922331)

[Lời cảm ơn iii](#_Toc44922332)

[Tóm tắt iv](#_Toc44922333)

[Abstract v](#_Toc44922334)

[Mục lục vi](#_Toc44922335)

[Danh mục hình vẽ ix](#_Toc44922336)

[Danh mục bảng x](#_Toc44922337)

[Danh mục công thức xi](#_Toc44922338)

[Danh mục các từ viết tắt xii](#_Toc44922339)

[Danh mục thuật ngữ xiii](#_Toc44922340)

[Chương 1 Giới thiệu đề tài 1](#_Toc44922341)

[1.1 Đặt vấn đề 1](#_Toc44922342)

[1.2 Mục tiêu và phạm vi đề tài 2](#_Toc44922343)

[1.3 Định hướng giải pháp 2](#_Toc44922344)

[1.4 Bố cục đồ án 2](#_Toc44922345)

[Chương 2 Khảo sát và phân tích yêu cầu 4](#_Toc44922346)

[2.1 Khảo sát hiện trạng 4](#_Toc44922347)

[2.2 Tổng quan chức năng 4](#_Toc44922348)

[2.2.1 Biểu đồ use case tổng quan 4](#_Toc44922349)

[2.2.2 Biểu đồ use case phân rã XYZ 5](#_Toc44922350)

[2.2.3 Quy trình nghiệp vụ 5](#_Toc44922351)

[2.3 Đặc tả chức năng 5](#_Toc44922352)

[2.3.1 Đặc tả use case A 5](#_Toc44922353)

[2.3.2 Đặc tả use case B 5](#_Toc44922354)

[2.4 Yêu cầu phi chức năng 5](#_Toc44922355)

[Chương 3 Công nghệ sử dụng 6](#_Toc44922356)

[Chương 4 Phát triển và triển khai ứng dụng 7](#_Toc44922357)

[4.1 Thiết kế kiến trúc 7](#_Toc44922358)

[4.1.1 Lựa chọn kiến trúc phần mềm 7](#_Toc44922359)

[4.1.2 Thiết kế tổng quan 7](#_Toc44922360)

[4.1.3 Thiết kế chi tiết gói 8](#_Toc44922361)

[4.2 Thiết kế chi tiết 8](#_Toc44922362)

[4.2.1 Thiết kế giao diện 8](#_Toc44922363)

[4.2.2 Thiết kế lớp 9](#_Toc44922364)

[4.2.3 Thiết kế cơ sở dữ liệu 9](#_Toc44922365)

[4.3 Xây dựng ứng dụng 9](#_Toc44922366)

[4.3.1 Thư viện và công cụ sử dụng 9](#_Toc44922367)

[4.3.2 Kết quả đạt được 9](#_Toc44922368)

[4.3.3 Minh hoạ các chức năng chính 10](#_Toc44922369)

[4.4 Kiểm thử 10](#_Toc44922370)

[4.5 Triển khai 10](#_Toc44922371)

[Chương 5 Các giải pháp và đóng góp nổi bật 11](#_Toc44922372)

[Chương 6 Kết luận và hướng phát triển 12](#_Toc44922373)

[6.1 Kết luận 12](#_Toc44922374)

[6.2 Hướng phát triển 12](#_Toc44922375)

[Tài liệu tham khảo 13](#_Toc44922376)

[Phụ lục A-1](#_Toc44922377)

[A Hướng dẫn viết đồ án tốt nghiệp A-1](#_Toc44922378)

[A.1 Quy định chung A-1](#_Toc44922379)

[A.2 Ngành học A-2](#_Toc44922380)

[A.3 Tạo đề mục A-2](#_Toc44922381)

[A.4 Bảng biểu A-3](#_Toc44922382)

[A.5 Hình vẽ A-4](#_Toc44922383)

[A.6 Tài liệu tham khảo A-4](#_Toc44922384)

[A.7 Công thức toán học A-5](#_Toc44922385)

[A.8 Tham chiếu chéo A-5](#_Toc44922386)

[A.9 Cập nhật mục lục và tham chiếu chéo A-6](#_Toc44922387)

[A.10 In quyển đồ án tốt nghiệp A-6](#_Toc44922388)

[B Đặc tả use case B-8](#_Toc44922389)

[B.1 Đặc tả use case “Thống kê tình hình mượn sách” B-8](#_Toc44922390)

[B.2 Đặc tả use case “Đăng ký làm thẻ mượn” B-8](#_Toc44922391)

[C Công nghệ sử dụng C-8](#_Toc44922392)

[C.1 Công nghệ bảo mật dữ liệu C-8](#_Toc44922393)

[C.2 Công nghệ blockchain C-8](#_Toc44922394)

[D Thiết kế gói D-8](#_Toc44922395)

[D.1 Thiết kế gói cho kiến trúc tổng quan D-8](#_Toc44922396)

[D.2 Thiết kế gói cho chức năng “Trả sách” D-9](#_Toc44922397)

[E Thiết kế lớp E-9](#_Toc44922398)

# Danh mục hình vẽ

[**Hình 1** Ví dụ biểu đồ phụ thuộc gói 7](#_Toc27562454)

[**Hình 2** Ví dụ thiết kế gói 8](#_Toc27562455)

[**Hình 3** Ví dụ hình vẽ A-4](#_Toc27562456)

[**Hình 4** Quy cách đóng quyển A-6](#_Toc27562457)

[**Hình 5** Quy cách ghi chữ phần gáy A-6](#_Toc27562458)

[**Hình 6** Hướng dẫn thiết lập in hai mặt A-7](#_Toc27562459)

# Danh mục bảng

[**Bảng 1** Danh sách thư viện và công cụ sử dụng 9](#_Toc27562460)

[**Bảng 2** Ví dụ sử dụng bảng A-2](#_Toc27562461)

# Giới thiệu đề tài

## Đặt vấn đề

Hiện nay có nhiều cửa hàng café nhỏ lẻ và chuỗi café với quy mô nhỏ thường vẫn dùng những phương pháp thủ công như bút, giấy để quản lí những việc như kiểm soát các vấn đề nguồn cung nguyên vật liệu đầu vào, nguyên vật liệu cần cho 1 sản phẩm, doanh số bán hàng hàng ngày, thời gian làm việc hàng ngày của nhân viên, thông tin cá nhân của nhân viên.

Việc làm các công đoạn trên là một việc tốn khá nhiều thười gian và không hiệu quả. Điều này đặt ra vấn đề về bài toán làm thế nào để giúp giảm thời gian và chi phí quản lí các quán café, từ đó nâng cao hiệu suất hoạt động

## Mục tiêu và phạm vi đề tài

Mục tiêu của đề tài này là xây dựng hệ thống mà sẽ được sử dụng để giải quyết vấn đề được nói đến ở trên với chi phí thấp, dễ dàng tiếp cận và sử dụng. Hiện nay ở Việt Nam có nhiều cửa hàng và chuỗi cửa hàng bán café vừa và nhỏ nhưng vẫn sử dụng cách quản lí thủ công. Việc mong muốn một hệ thống giúp quản lí là hoàn toàn chính đáng và hợp lí, tuy nhiên vì nhiều lí do mà các hệ thống hiện nay trên thị trường chưa được triển khai rộng rãi ở các cửa hàng và chuỗi cửa hàng vừa và nhỏ này.

Ở thời điểm hiện tại, ứng dụng đã có thể chạy được nhưng ở dưới dạng vẫn cần các phần mềm chuyên dụng của lập trình viên để chạy, chưa thể chạy trực tiếp trên máy khách hàng vì các vấn đề về kĩ thuật và thời gian phát triển sản phẩm ngắn.

Hệ thống hiện tại gồm các chức năng chính là: đăng nhập, tạo tài khoản để quản lí người dùng trong hệ thống (nhân viên), tạo đơn hàng, nhập và kiểm soát nguyên liệu trong kho, thống kê thu chi của quán, thay đổi menu quán.

## Định hướng giải pháp

Thiết kế ra một hệ thống giúp cho việc quản lí thu chi, tính toán thời gian làm việc và tiền lương của từng nhân viên. Hệ thống cần có thể dễ dàng cài đặt trên nhiều hệ thống máy tính thấp cấp để tiết kiệm chi phí, dễ dàng sử dụng, có giao diện thân thiện với người dùng, dễ hiểu. Từ đó có thể giúp tiết kiệm thời gian, chi phí và nâng cao hiệu suất của doanh nghiệp sử dụng hệ thống.

Hệ thống sẽ được xây dựng sử dụng ngôn ngữ Java vì đây là ngôn ngữ có thể chạy trên nhiều hệ điều hành khác nhau, miễn là có môi trường Java và MySql cho cơ sở dữ liệu

## 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

## Khảo sát hiện trạng

Hiện nay, theo tôi tìm hiểu, có một vài các ứng dụng giúp quản lí cho chuỗi café trên cả nền tảng máy tính, tablet và smartphone, nhiều ứng dụng còn có thể sử dụng đa nền tảng, có thể kể đến như **KiotViet, SoftNet Restaurant, CukCuk,** … các hệ thống này đều cho dùng thử từ 15 tới 30 ngày rồi sau đó sẽ cần phải phí giao động vào khoảng 99.000/tháng cho 1 thiết bị. Đây đều là những lựa chọn có sẵn phù hợp đối với nhiều đối tượng khách hàng, tuy nhiên nó cũng có nhiều hạn chế. Những hạn chế của những hệ thống có sẵn được kể trên có thể nói tới bao gồm giá thành, giao diện, kết nối và lưu trữ.

Đối với giá thành, 99.000k trên 1 thiết bị mỗi tháng không phải quá lớn, tuy nhiên một quán nhỏ lẻ có thể muốn có nhiều hơn một thiết bị cho quầy thu ngân, chưa kể đến những thiết bị rời như tablet và smartphone, cũng như việc một số thiết bị đời cũ có thể không tương thích được với ứng dụng, tạo nên chi phí không nhỏ.

Tiếp đến là giao diện, vì để tăng cường nhiều chức năng, nhiều ứng dụng trên thị trường hiện tại có rất nhiều tính năng và giao diện chưa hợp lí hoặc thừa thãi.

Cuối cùng là về kết nối và lưu trữ, phần lớn các ứng dụng trên đều cần kết nối mạng đến máy server của ứng dụng để hoạt động, việc này có thể đem đến rất nhiều phiền toái khi kết nối internet của quán có vấn đề hoặc bên phía server có vấn đề gây ra hiện tượng không thể quản lí trong một thời gian ngắn.

## Tổng quan chức năng

### Biểu đồ use case tổng quan

Các chức năng của hệ thống sẽ được phân quyền, có 3 phân lớp quyền cơ bản phụ thuộc vào tác nhân, đó quyền của nhân viên (staff), quyền của quản lí (manager) và quyền của giám đốc / chủ doanh nghiệp (CEO). Mỗi quyền sau đều sử dụng các tùy chọn của quyền trước đó và có thêm những lựa chọn tùy chọn mới mà quyền thấp cấp hơn trước đó không sử dụng được. Bao gồm:

A diagram of a company

Description automatically generatedStaff:

+ Đăng nhập

+ Đăng xuất

+ Kiểm tra thông tin cá nhân

+ Nhập hàng

+ Tạo đơn hàng

+ Kiểm tra/thay đổi thông tin kho hàng

Manager:

+ Tạo tài khoản

+ Thống kê số liệu

Director/Admin:

+ Thay đổi menu UseCase Tổng quan

### Biểu đồ use case phân rã XYZ

#### Biểu đồ phân rã usecase của Staff (Nhân viên)

A diagram of a person with many circles

Description automatically generated

#### Biểu đồ phân rã usecase của Manager (quản lí)

A diagram of a person with many circles

Description automatically generated

#### Biểu đồ phân rã usecase của CEO (Giám đốc)

A diagram of a person with many circles

Description automatically generated

### 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.

# Công nghệ sử dụng

Các công nghệ sử dụng cho chương trình vẫn hiện đang dừng lại ở mức độ dành cho nhà phát triển, chưa thể trực tiếp triển khai trên máy khách thông qua Java runtime environment. Vậy nên trừ khi máy khách cũng cài đặt các phần mềm như trên máy của nhà phát triển mới có thể chạy chương trình. Các phần mềm đó bao gồm:  
Ngôn ngữ lập trình: Java

Thư viện sử dụng: Thư viện java tiêu chuẩn JDK 20, mysql-connector-j-8.0.33.jar

IDE: Intellij

Database: MySQL

Xây dựng UI: Scene Builder

Trong đó Code khi được chạy sẽ đọc thông tin về cơ sở dữ liệu trong file config.txt và từ đó kết nối tới cơ sở dữ liệu MySQl thông qua hàm DatabaseUtil.java.

# Phát triển và triển khai ứng dụng

## Thiết kế kiến trúc

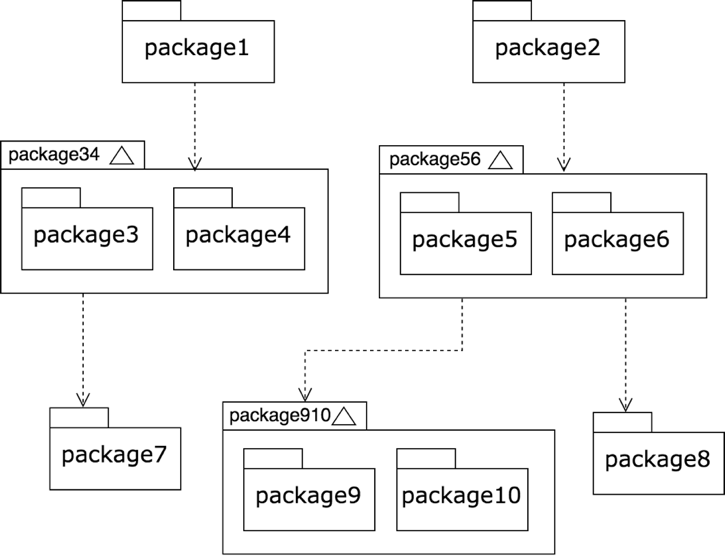
### Lựa chọn kiến trúc phần mềm

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.

### 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.

### 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

Phần này có độ dài từ hai đến bốn trang. Sinh viên thiết kế, vẽ và giải thích biểu đồ thực thể liên kết (E-R diagram). Từ đó, sinh viên thiết kế cơ sở dữ liệu tùy theo hệ quản trị cơ sở dữ liệu mà mình sử dụng (SQL, NoSQL, Firebase, v.v.)

## 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 | Eclipse Oxygen 64 bit | http://www.eclipse.org/ |
| v.v. | v.v. | v.v. |

### 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.

## 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.

## 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.

# 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.

Ngành học

Sinh viên lưu ý viết đúng ngành/chuyên ngành trên bìa và trên gáy theo đúng quy định của Trường. Ngành học hay chuyên ngành học phụ thuộc vào ngành học mà sinh viên đăng ký. Sinh viên có thể đăng nhập trên trang quản lý học tập của mình để xem lại chính xác ngành học của mình.

Một số ví dụ sinh viên có thể tham khảo dưới đây, trong trường hợp có chuyên ngành thì sinh viên không cần ghi chuyên ngành:

* Đối với kỹ sư chính quy:
  + Từ K61 trở về trước: Ngành Kỹ thuật phần mềm
  + Từ K62 trở về sau: Ngành Khoa học máy tính
* Đối với cử nhân:
  + Ngành Công nghệ thông tin
* Đối với chương trình EliteTech:
  + Chương trình Việt Nhật/KSTN: Ngành Công nghệ thông tin
  + Chương trình ICT Global: Ngành Information Technology
  + Chương trình DS&AI: Ngành Khoa học dữ liệu
* Đối với kỹ sư chương trình đào tạo quốc tế SIE
  + Chương trình LTU: Ngành Công nghệ thông tin/Information Technology
  + Chương trình VUW: Ngành Kỹ thuật Phần mềm
  + Chương trình G-INP: Ngành Hệ thống thông tin

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 hoặc nhỏ hơn. 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 |

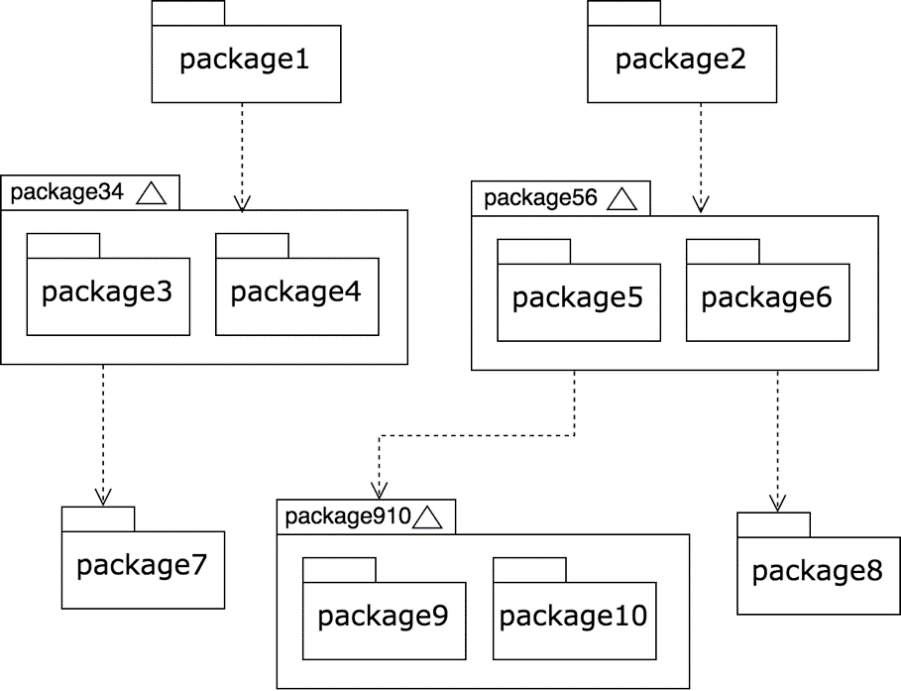
Sinh viên được tùy ý lựa chọn định dạng (template) cho các bảng trong ĐATN của mình, nhưng phải đảm bảo tính nhất quán trên toàn ĐATN. Template bảng phải đảm bảo phần heading cột trong bảng có font in đậm và nổi bật (highlighted) hơn so với các nội dung khác trong bảng.

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”.

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.



**Hình 3** Ví dụ hình vẽ

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

Khi đóng quyển, bìa trước và bìa sau là giấy liền khổ. **Sử dụng keo nhiệt để dán gáy khi đóng quyển thay vì sử dụng băng dính và dập ghim** (Xem Hình 4).



**Hình 4** Quy cách đóng quyển

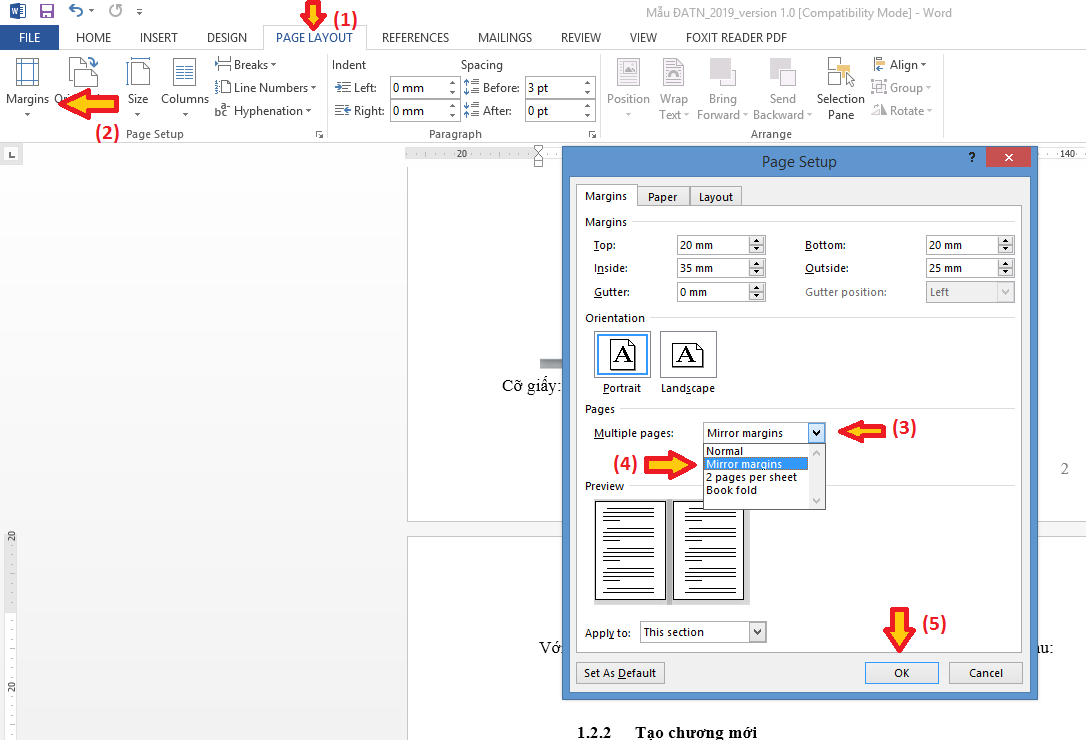
Phần gáy ĐATN được đóng theo quy cách như Hình 5, với các thông tin cần ghi bao gồm *Kỳ làm ĐATN - Ngành đào tạo - Họ và tên sinh viên - Mã số sinh viên*



**Hình 5** Quy cách ghi chữ phần gáy

Sinh viên có thể in một mặt hoặc hai mặt. Template này mặc định ở chế độ in một mặt. Để chuyển sang chế độ in hai mặt, sinh viên phải thiết lập lại page layout như sau. Sinh viên vào tab Page Layout, chọn Margins, chọn Custom Margin, trong popup hiện ra, ở mục Multiple pages, chọn Mirror margins, và nhấn OK (Xem **Hình 6**). Do trang đầu tiên là trang bìa của đồ án, nên khi in chế độ hai mặt có thể cần chèn (insert) 1 trang trắng sau trang bìa để đảm bảo trang “Lời cam kết” sẽ là trang lẻ. Để tránh sai sót gây lãng phí, sinh viên cần kiểm tra kỹ trang chẵn lẽ trước khi in.

Sinh viên lưu ý, để đóng quyển theo đúng quy cách với keo nhiệt và chữ in trên gáy, độ dày tối thiểu cho tổng các trang là 0.5cm. Sinh viên tự cân nhắc in một mặt hoặc hai mặt.



**Hình 6** Hướng dẫn thiết lập in hai mặt

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)