**TRƯỜNG ĐẠI HỌC PHENIKAA**

**BÁO CÁO BÀI TẬP LỚN**

**[TÊN BTL]**

**20010000 Nguyễn Văn A** nguyenvanabc@st.phenikaa-uni.edu.vn

**20010010 Nguyễn Thị B** nguyenvanabc@st.phenikaa-uni.edu.vn

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | ThS. Nguyễn Thành Trung |
| **Khoa:** | Công nghệ thông tin |
| **HÀ NỘI, 06/2021** | |

# Lời cam kết

Họ và tên nhóm sinh viên:

* Phan Thành Đạt

Điện thoại liên lạc: . . . .0342611368. . . . Email: . . . Phandat2002h@gmail.com

Lớp: . . Công nghệ thông tin Việt Nhật. . . . . . . Hệ đào tạo: . . Chính quy. .

Tôi cam kết Bài tập lớn (BTL) là công trình nghiên cứu của bản thân tôi. Các kết quả nêu trong BTL 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 BTL – 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.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 08 tháng 08 năm 2024*  Tác giả BTL  Phan Thành Đạt |

# Tóm tắt

Trong thời đại khoa học công nghệ và y học phát triển, tuổi thọ của con người ngày càng được kéo dài, dẫn đến số lượng người cao tuổi ngày càng tăng. Tuy nhiên, tuổi tác cao thường kéo theo sự suy giảm của một số chức năng trên cơ thể, đặc biệt là não bộ. Một trong những vấn đề phổ biến nhất mà người cao tuổi phải đối mặt là chứng suy giảm trí nhớ. Hiện tại, việc kiểm tra và đánh giá tình trạng này cần sự can thiệp của các chuyên gia chuyên nghiệp, đòi hỏi người cao tuổi phải đến trực tiếp các cơ sở y tế. Để giải quyết vấn đề này, tôi đã lựa chọn phát triển một hệ thống chuẩn đoán sơ bộ từ xa. Hệ thống này mang lại nhiều lợi ích, chẳng hạn như tiết kiệm chi phí và thời gian cho người dùng. Hệ thống chuẩn đoán từ xa hoạt động bằng cách sử dụng các công nghệ hiện đại để thu thập và phân tích dữ liệu về tình trạng trí nhớ của người dùng thông qua các bài kiểm tra trực tuyến. Sau đó, kết quả sẽ được gửi đến các chuyên gia y tế để đánh giá và đưa ra khuyến nghị. BTL này áp dụng những kiến thức đã học để cải thiện chất lượng sống của mọi người, đặc biệt là người cao tuổi. Kết quả cuối cùng của dự án là xây dựng được một ứng dụng đánh giá sa sút trí tuệ từ xa, giúp người cao tuổi dễ dàng tiếp cận dịch vụ y tế và nâng cao chất lượng cuộc sống.

# Mục lục

[Lời cam kết ii](#_Toc74298230)

[Tóm tắt iii](#_Toc74298231)

[Mục lục iv](#_Toc74298232)

[Danh mục hình vẽ vii](#_Toc74298233)

[Danh mục bảng viii](#_Toc74298234)

[Danh mục công thức ix](#_Toc74298235)

[Danh mục các từ viết tắt x](#_Toc74298236)

[Danh mục thuật ngữ xi](#_Toc74298237)

[Chương 1 Giới thiệu đề tài 1](#_Toc74298238)

[1.1 Đặt vấn đề 1](#_Toc74298239)

[1.2 Mục tiêu và phạm vi đề tài 2](#_Toc74298240)

[1.3 Định hướng giải pháp 2](#_Toc74298241)

[1.4 Bố cục bài tập lớn 2](#_Toc74298242)

[Chương 2 Khảo sát và phân tích yêu cầu 4](#_Toc74298243)

[2.1 Khảo sát hiện trạng 4](#_Toc74298244)

[2.2 Tổng quan chức năng 4](#_Toc74298245)

[2.2.1 Biểu đồ use case tổng quan 4](#_Toc74298246)

[2.2.2 Biểu đồ use case phân rã XYZ 5](#_Toc74298247)

[2.2.3 Quy trình nghiệp vụ 5](#_Toc74298248)

[2.3 Đặc tả chức năng 5](#_Toc74298249)

[2.3.1 Đặc tả use case A 5](#_Toc74298250)

[2.3.2 Đặc tả use case B 5](#_Toc74298251)

[2.4 Yêu cầu phi chức năng 5](#_Toc74298252)

[Chương 3 Công nghệ sử dụng 6](#_Toc74298253)

[Chương 4 Phát triển và triển khai ứng dụng 7](#_Toc74298254)

[4.1 Thiết kế kiến trúc 7](#_Toc74298255)

[4.1.1 Lựa chọn kiến trúc phần mềm 7](#_Toc74298256)

[4.1.2 Thiết kế tổng quan 7](#_Toc74298257)

[4.1.3 Thiết kế chi tiết gói 8](#_Toc74298258)

[4.2 Thiết kế chi tiết 8](#_Toc74298259)

[4.2.1 Thiết kế giao diện 8](#_Toc74298260)

[4.2.2 Thiết kế lớp 9](#_Toc74298261)

[4.2.3 Thiết kế cơ sở dữ liệu 9](#_Toc74298262)

[4.3 Xây dựng ứng dụng 9](#_Toc74298263)

[4.3.1 Thư viện và công cụ sử dụng 9](#_Toc74298264)

[4.3.2 Kết quả đạt được 9](#_Toc74298265)

[4.3.3 Minh hoạ các chức năng chính 10](#_Toc74298266)

[4.4 Kiểm thử 10](#_Toc74298267)

[4.5 Triển khai 10](#_Toc74298268)

[Chương 5 Các giải pháp và đóng góp nổi bật 11](#_Toc74298269)

[Chương 6 Kết luận và hướng phát triển 12](#_Toc74298270)

[6.1 Kết luận 12](#_Toc74298271)

[6.2 Hướng phát triển 12](#_Toc74298272)

[Tài liệu tham khảo 13](#_Toc74298273)

[Phụ lục A-1](#_Toc74298274)

[A Hướng dẫn viết Bài tập lớn A-1](#_Toc74298275)

[A.1 Quy định chung A-1](#_Toc74298276)

[A.2 Tạo đề mục A-2](#_Toc74298277)

[A.3 Bảng biểu A-2](#_Toc74298278)

[A.4 Hình vẽ A-3](#_Toc74298279)

[A.5 Tài liệu tham khảo A-4](#_Toc74298280)

[A.6 Công thức toán học A-4](#_Toc74298281)

[A.7 Tham chiếu chéo A-5](#_Toc74298282)

[A.8 Cập nhật mục lục và tham chiếu chéo A-5](#_Toc74298283)

[A.9 In quyển Bài tập lớn A-6](#_Toc74298284)

[B Đặc tả use case B-6](#_Toc74298285)

[B.1 Đặc tả use case “Thống kê tình hình mượn sách” B-6](#_Toc74298286)

[B.2 Đặc tả use case “Đăng ký làm thẻ mượn” B-6](#_Toc74298287)

[C Công nghệ sử dụng C-6](#_Toc74298288)

[C.1 Công nghệ bảo mật dữ liệu C-6](#_Toc74298289)

[C.2 Công nghệ blockchain C-6](#_Toc74298290)

[D Thiết kế gói D-6](#_Toc74298291)

[D.1 Thiết kế gói cho kiến trúc tổng quan D-6](#_Toc74298292)

[D.2 Thiết kế gói cho chức năng “Trả sách” D-6](#_Toc74298293)

[E Thiết kế lớp E-6](#_Toc74298294)

# 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)

# Danh mục công thức

[**Công thức 1** Khai triển Newton A-5](#_Toc27562462)

Lưu ý: Nếu BTL 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 |
| **BTL** | Bài tập lớn |
| **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

Chương 1 có độ dài từ 3 đến 6 trang với các nội dung sau đây

## Đặt vấn đề

Hiện nay, khi khoa học công nghệ và y học ngày càng phát triển, tuổi thọ của con người đã được kéo dài đáng kể. Điều này dẫn đến sự gia tăng đáng kể số lượng người cao tuổi trong xã hội. Tuy nhiên, cùng với sự gia tăng này là sự xuất hiện phổ biến của các vấn đề về sức khỏe, đặc biệt là suy giảm trí nhớ và sa sút trí tuệ. Đây là một vấn đề nghiêm trọng, ảnh hưởng đến chất lượng cuộc sống của người cao tuổi, gây khó khăn trong sinh hoạt hàng ngày và tạo gánh nặng cho gia đình và xã hội. Nếu vấn đề này được giải quyết, nó sẽ mang lại lợi ích to lớn, không chỉ cải thiện chất lượng cuộc sống cho người cao tuổi mà còn giảm bớt áp lực cho gia đình và hệ thống y tế. Hơn nữa, giải pháp cho vấn đề này có thể được áp dụng rộng rãi trong nhiều lĩnh vực khác, như chăm sóc sức khỏe từ xa, hỗ trợ tâm lý và tư vấn y tế, góp phần nâng cao chất lượng dịch vụ y tế và phát triển các ứng dụng công nghệ trong y học.

## Mục tiêu và phạm vi đề tài

Ở Việt Nam, hiện chưa có nhiều sản phẩm và dịch vụ hỗ trợ đánh giá tình trạng sa sút trí tuệ ở người cao tuổi. Vậy nên mọi người thường đến trực tiếp các cơ sở y tế để thực hiện các bài kiểm tra và đánh giá bởi các chuyên gia. Nhu cầu của người dùng hiện nay là có một hệ thống tiện lợi, tiết kiệm thời gian và chi phí, đồng thời vẫn đảm bảo được độ tương đối chính xác trong việc đánh giá tình trạng trí tuệ.

Qua việc so sánh và đánh giá tình trạng thực tế, tôi nhận thấy rằng cần phải có một hệ thống có thể đáp ứng được đầy đủ nhu cầu của người dùng. Hệ thống phải đáp ứng được một số hạn chế chính bao gồm: yêu cầu người dùng phải đến trực tiếp các cơ sở y tế, chi phí cao, thời gian chờ đợi lâu, và sự phức tạp trong việc sử dụng công nghệ.

Dựa trên những phân tích và đánh giá này, đề tài hướng tới việc giải quyết cụ thể các hạn chế hiện tại bằng cách phát triển một phần mềm đánh giá tình trạng sa sút trí tuệ từ xa cho người cao tuổi. Phần mềm này sẽ có các chức năng chính như: Trực tiếp được đánh giá từ các chuyên gia, thực hiện các bài kiểm tra trí nhớ trực tuyến, thu thập và phân tích dữ liệu tự động, và cung cấp kết quả đánh giá sơ bộ nhanh chóng. Mục tiêu của đề tài là tạo ra một đột phá trong việc sử dụng công nghệ để nâng cao chất lượng sống của người cao tuổi, giúp họ dễ dàng tiếp cận dịch vụ y tế mà không cần phải di chuyển đến các cơ sở y tế.

## Định hướng giải pháp

Để giải quyết vấn đề đã xác định, tôi sẽ áp dụng công nghệ web để phát triển một hệ thống đánh giá tình trạng sa sút trí tuệ từ xa. Công nghệ cho phép hệ thống thực hiện các bài kiểm tra trí nhớ trực tuyến và phân tích dữ liệu thu thập được một cách tự động và chính xác, từ đó đưa ra các đánh giá sơ bộ nhanh chóng. Hệ thống này cũng sẽ tích hợp với các chuyên gia y tế, cho phép họ trực tiếp đánh giá và theo dõi tình trạng sức khỏe của người cao tuổi dựa trên dữ liệu được thu thập.

Giải pháp của tôi là thiết kế và phát triển một ứng dụng có khả năng thực hiện các bài kiểm tra trí nhớ trực tuyến, thu thập dữ liệu, và cung cấp kết quả sơ bộ nhanh chóng, đồng thời tạo điều kiện cho các chuyên gia y tế tiếp cận dữ liệu này để đưa ra những đánh giá chuyên sâu hơn. Ứng dụng này sẽ giúp tiết kiệm thời gian, chi phí cho người dùng và nâng cao khả năng tiếp cận dịch vụ y tế cho người cao tuổi.

Với giải pháp được đưa ra tôi mong rằng mình có thể hoàn thiện một giải pháp toàn diện từ thiết kế đến triển khai một ứng dụng đánh giá tình trạng sa sút trí tuệ từ xa. Kết quả đạt được là xây dựng một ứng dụng hoạt động hiệu quả, đáp ứng đúng nhu cầu của người cao tuổi và giúp cải thiện chất lượng cuộc sống của họ.

## Bố cục bài tập lớn

Phần còn lại của báo cáo Bài tập lớn này được tổ chức như sau.

Về tổng quan chương 2 của bài tập lớn tập trung vào việc khảo sát và phân tích yêu cầu để xây dựng ứng dụng đánh giá sa sút trí tuệ từ xa cho người cao tuổi. Nội dung chương này bao gồm hai phần chính: (i) Khảo sát hiện trạng và (ii) Tổng quan chức năng. Trong phần này, tôi đã tiến hành phân tích, so sánh và đánh giá chi tiết ưu nhược điểm của các sản phẩm hiện có, qua đó xác định những tính năng quan trọng cần phát triển trong ứng dụng của mình. Ứng dụng sẽ có các chức năng chính như: thực hiện các bài kiểm tra trí nhớ trực tuyến, thu thập và phân tích dữ liệu tự động, cung cấp kết quả đánh giá sơ bộ, và cho phép các chuyên gia y tế trực tiếp tham gia đánh giá và theo dõi tình trạng trí nhớ của người cao tuổi.

Sau khi đã có thể xác định được hết những chức năng cần thiết thì bước tiếp theo cần làm là tìm hiểu và lựa chọn những công nghệ cần thiết để sử dụng trong dự án. Và đây cũng là nội dung chính mà chương 3 đề cập tới. Một số công nghệ được áp dụng vào đề tài ở Chương 3 có đề cập đến như back-end Springboot, font-end Angular, database MySql. Dựa vào nền tảng các công nghệ trên để có thể lựa chọn những kiến trúc hệ thống phù hợp để phát triển và triển khai ứng dụng ở chương số 4.

Với những công nghệ đã đề cập ở trên thì tôi đã lựa chọn được mô hình kiến trúc phù hợp cho đề tài của mình. Với Font-end thì tôi sử dụng MVVM( Model-view-view/model). Với springboot thì kiến trúc được lựa chọn là Microservices do ưu điểm rất dễ dàng nâng cấp và có thể hoạt động khi mà 1 service nào đó bị trục trặc mà không ảnh hưởng đến toàn bộ hệ thống.

# ­­Khảo sát và phân tích yêu cầu

Chương này có độ dài từ 9 đến 11 trang. Với bài tập lớn 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 bài tập lớn 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.

## 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.

### 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 bài tập lớn 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

Chương này có độ dài không quá 10 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 bài tập lớn 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 bài tập lớn. Sinh viên cũng có thể trình bày thêm nền tảng lý thuyết nào đó nếu cần dùng tới.

Với bài tập lớn 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 BTL 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).

# Phát triển và triển khai ứng dụng

## Thiết kế kiến trúc

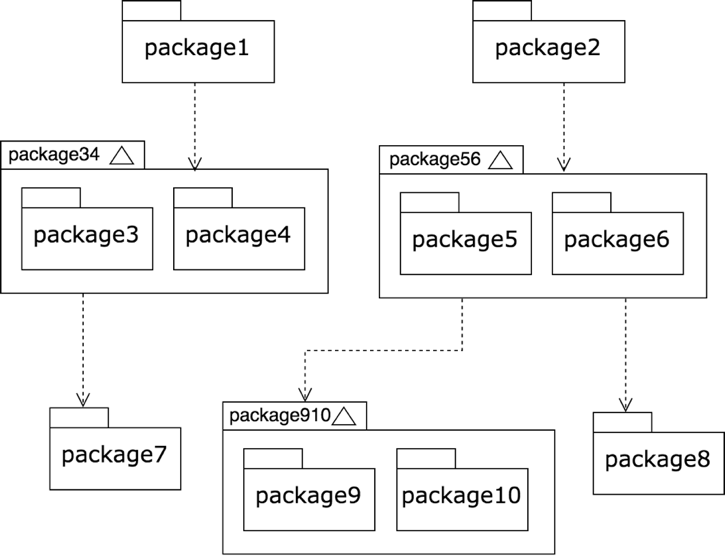
### Lựa chọn kiến trúc phần mềm

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 BTL. 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 BTL, 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 BTL 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.

<**Bài tập lớn, Luận văn Thạc sĩ, Tiến sĩ**: Tên tác giả, tên bài tập lớn/luận văn, loại bài tập lớn/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 Bài tập lớn

Quy định chung

Dưới đây là một số quy định và hướng dẫn viết Bài tập lớn 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 BTL 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 BTL**.

Tất cả các hình vẽ, bảng biểu, công thức, và tài liệu tham khảo trong BTL 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 BTL theo kiểu viết ý hoặc gạch đầu dòng. BTL 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 BTL 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 BTL.

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 |

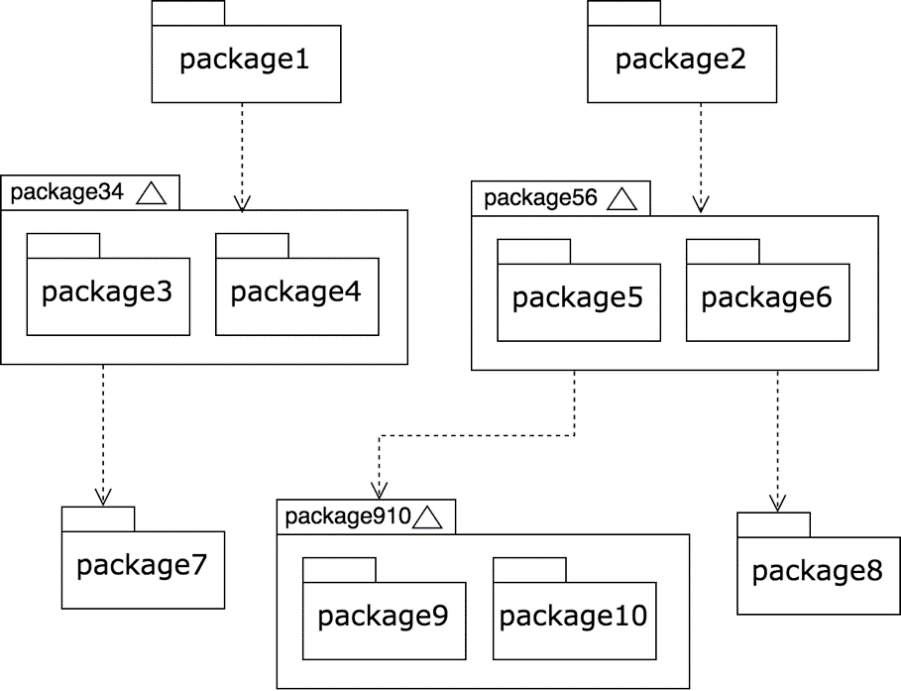
Sinh viên được tùy ý lựa chọn định dạng (template) cho các bảng trong BTL của mình, nhưng phải đảm bảo tính nhất quán trên toàn BTL. Template bảng phải đảm bảo phần heading cột trong bảng có font in đậm và nổi bật (highlighted) hơn so với các nội dung khác trong bảng.

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 BTL. 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 BTL, 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 BTL, 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 BTL 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 BTL để tránh mọi sai sót.

In quyển Bài tập lớn

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