



PROJECT GUIDE

Lưu hành nội bộ

Thông tin tài liệu

Phiên bản	Ngày ban hành	Trạng thái	Trang thay đổi
1.0	10/06/2014	Soạn mới	-

MỤC LỤC

1. Giới thiệu	3
2. Lựa chọn nhóm dự án	4
3. Cách chọn một dự án	4
4. Phương thức thực hiện dự án	5
5. Xây dựng kế hoạch cho dự án.....	7
6. Quản lý yêu cầu	9
7. Phân tích và thiết kế.....	14
8. Phát triển dự án	16
9. Kiểm thử (Testing).....	16
10. Viết báo cáo	20
11. Tài liệu hỗ trợ của dự án	23
12. Bảo vệ dự án	23
13. Công cụ phát triển dự án.....	25
14. Form mẫu cho dự án	26
15. Cách đánh giá đồ án.....	26
16. Thời lượng và tiến độ.....	26
17. Tài liệu tham khảo	28

1. Giới thiệu

Đồ án cuối mỗi khóa học là vô cùng quan trọng đối với mỗi học viên, nó giúp học viên tổng hợp, áp dụng các kiến thức đã học vào thực tế công việc cũng như thu thập những kinh nghiệm để sẵn sàng cho công việc sắp tới. Các nhà tuyển dụng có xu hướng sử dụng đồ án cuối khóa của học viên để đánh giá năng lực của các bạn khi tuyển dụng.

Do vậy, việc thực hiện dự án cần được các học viên thực hiện nghiêm túc, có phương pháp tốt để đạt được kết quả tốt nhất khi kết thúc môn học này. Tài liệu này nhằm mục tiêu hướng dẫn học viên thực hiện đồ án một cách hiệu quả nhất dưới sự hướng dẫn của giảng viên tại Trung tâm đào tạo Lập trình viên chuyên nghiệp iViettech.

Tùy theo đặc điểm của chương trình đào tạo mà giáo viên có thể lựa chọn một số phần trong tài liệu này để hướng dẫn cho học viên thực hiện đồ án cho hiệu quả.

Những khó khăn học viên thường gặp khi thực hiện đồ án:

- Chọn một dự án như thế nào là hợp lý?
- Làm thế nào để xây dựng dự án đó?
- Phương thức sản xuất phần mềm nào sử dụng là hợp lý?
- Nên phân chia các giai đoạn như thế nào?
- Công cụ hỗ trợ nào nên được sử dụng?
- Chuẩn bị một báo cáo như thế nào cho hợp lý?
- Tài liệu nào cần phải có?
- Chuẩn bị bảo vệ đồ án như thế nào?

Tài liệu này sẽ giúp học viên hiểu rõ hơn về những yêu cầu đó để giúp các em yên tâm hơn trong quá trình thực hiện dự án của mình.

2. Lựa chọn nhóm dự án

Trong ngành công nghiệp phần mềm bạn có rất ít cơ hội thành công nếu tự mình làm tất cả mọi việc. Do vậy, đồ án này yêu cầu các bạn phải thực hiện theo nhóm. Mỗi nhóm dự án được khuyến khích là 3 người, trong một số điều kiện đặc biệt các bạn có thể chọn 2 hoặc 4 người nhưng nên hạn chế. Vì nhóm nhiều người các bạn sẽ không quản lý được, ít người quá các bạn sẽ không đủ nguồn lực để hoàn thành dự án như ý muốn.

Do vậy, hãy tìm hiểu và lựa chọn các bạn vào nhóm mình cho hợp lý. Cố gắng chọn các bạn có thể mạnh có thể bổ sung cho nhau vào 1 nhóm. Ví dụ: bạn có khả năng code tốt nên chọn làm chung nhóm với bạn có khả năng thiết kế giao diện tốt v.v..

3. Cách chọn một dự án

Đối với mỗi khóa học, tính chất của mỗi dự án khác nhau tùy vào loại ứng dụng của khóa học đó nhắm tới. Việc chọn lựa đúng chủ đề và cấu trúc của dự án sẽ giúp cho các bạn có hứng thú hơn với dự án và đạt được kết quả cao hơn.

Có hai cách để chọn dự án:

1. Chọn từ danh sách của giáo viên

Mỗi giáo viên hướng dẫn môn đồ án đều chuẩn bị một danh sách các dự án theo mô hình thực tế tương ứng với lĩnh vực mà bạn đang học để bạn có thể lựa chọn. Bạn cần trao đổi kỹ với giáo viên khi lựa chọn đề tài/dự án để thực hiện.

2. Tự đề xuất dự án

Học viên có thể thấy hứng thú với dự án nào mình đang ấp ủ và đề xuất với giáo viên. Giáo viên sẽ xem xét góp ý để bạn có lựa chọn đúng chủ đề, phạm vi của dự án. Sau đó bạn hãy viết mô tả chi tiết của đề án trước khi thực hiện.

3. Đặc tả tóm tắt yêu cầu dự án

Khi đã xác định được dự án, các bạn phải tóm tắt đặc tả dự án và trình giáo viên phê duyệt. Đặc tả tóm tắt dự án bao gồm:

1. Tên dự án:
2. Mục đích của dự án:
3. Các chức năng của dự án:
4. Mô tả ngắn gọn yêu cầu của từng chức năng:
5. Các yêu cầu khác:

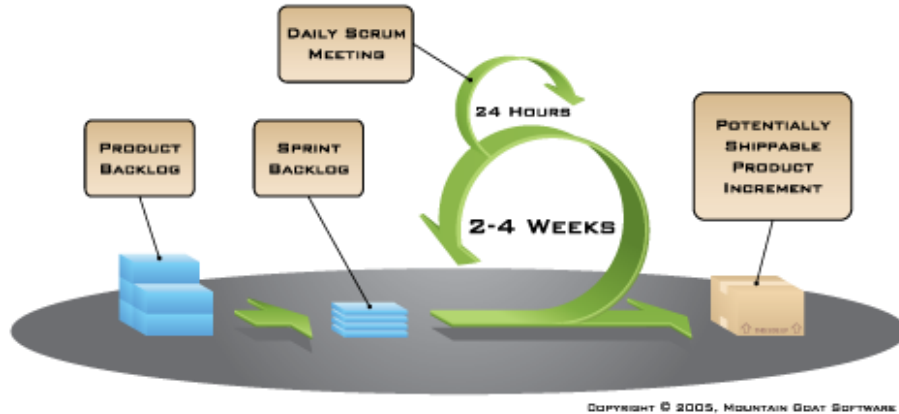
4. Phương thức thực hiện dự án

Phương thức (mô hình) thực hiện dự án sẽ qui định cách thức chúng ta phát triển dự án phần mềm. Có thể kể ra một số mô hình thực hiện dự án phần mềm như sau:

1. **Mô hình thác nước (Waterfall):** thực hiện các bước tuần tự từ Phân tích yêu cầu, Thiết kế, Lập trình, Kiểm thử, triển khai. Mô hình này phù hợp với những dự án nhỏ, mọi thứ đã rõ ràng. Khó thành công với các dự án lớn.
2. **Mô hình xoắn ốc (Spiral):** Thực hiện mô hình thác nước theo từng giai đoạn và hoàn thiện dần dần.
3. **Mô hình RUP (Rational Unified Proccess):** Mô hình tăng trưởng bằng cách chia dự án ra thành nhiều giai đoạn (iteration) để phát triển. Mô hình này hiện đại và phù hợp với những dự án lớn.
4. **Mô hình Agile:** là mô hình phát triển ứng dụng nhanh giảm nhẹ tài liệu và thời gian chuẩn bị dự án. Chúng ta có thể dễ dàng thay đổi yêu cầu cũng như xác nhận kết quả dự án ngay trong quá trình phát triển. Mô hình này được sử dụng rộng rãi hiện nay và rất phù hợp với các dự án mới phát triển

hoặc dành cho học viên. Framework thành công nhất của mô hình này là SCRUM và trong đồ án này chúng ta sẽ áp dụng SCRUM để thực hiện.

4.1 Giới thiệu sơ lược về SCRUM



Hình 1. Mô hình phát triển phần mềm theo SCRUM

- **Product Backlog:** là danh sách các chức năng của sản phẩm mà chúng ta sẽ phát triển.
- **Sprint:** là giai đoạn phát triển được chia ra từ 2-4 tuần. Trong phạm vi đồ án này mỗi sprint đề nghị là 1 tuần. Kết thúc Sprint thì phải có kết quả đầu ra.
- **Sprint Backlog:** là danh sách các chức năng bạn sẽ phát triển trong sprint đó. Nếu Product Backlog cuối cùng được phát triển có nghĩa bạn đã hoàn thành dự án.
- **Daily Scrum Meeting:** là phiên họp hàng ngày để Review công việc. Mỗi phiên họp nên tối đa là 15 phút và mỗi người nên đặt ra 3 câu hỏi:
 - Bạn đã làm gì ngày hôm qua?
 - Bạn gặp trở ngại gì? Hay có kinh nghiệm gì được rút ra?
 - Bạn định làm gì ngày hôm nay?

Bạn có thể đọc thêm về SCRUM tại:

<http://iviettech.vn/category/blog/scrum-methodology>

Và giáo viên sẽ hướng dẫn cách áp dụng Scrum vào dự án cho các bạn.

5. Xây dựng kế hoạch cho dự án

Bất kỳ một dự án nào và làm theo phương pháp nào thì bạn cũng cần hoạch định kế hoạch cho dự án. Có kế hoạch bạn mới biết được bạn sẽ làm như thế nào? Trong bao lâu? Và tốn bao nhiêu tiền (ở đây là công sức bạn bỏ ra). Do vậy, lập kế hoạch là bước vô cùng quan trọng và cần thiết. Trong phần này, chúng ta sẽ bàn về cách lập kế hoạch.

Chúng ta chia kế hoạch ra làm 02 phần:

1. Xây dựng khung kế hoạch (Roadmap) cho dự án: là cách thể hiện các mốc chính theo thời gian và kết quả đạt được để giúp bạn dễ hình dung và kiểm tra tiến độ của dự án.
2. Kế hoạch chi tiết: cho bạn biết cụ thể ai làm việc gì, ngày nào v.v.. Bạn nên làm theo từng giai đoạn khi đã có đầy đủ thông tin.

Trước tiên, để lập được kế hoạch của dự án chúng ta phải xác định được các bước trong quá trình phát triển phần mềm. Thông thường nó bao gồm các bước sau:

1. Thu thập và hiểu yêu cầu
2. Phân tích
3. Thiết kế
4. Kiểm thử
5. Triển khai

Đối với Scrum các bước được thể hiện như sau:

1. Thu thập và hiểu yêu cầu

2. Phân tích, thiết kế tổng thể
3. Phát triển: Sprint
 - a. Phân tích, thiết kế chi tiết
 - b. Phát triển
 - c. Kiểm thử
4. Kiểm thử hệ thống
5. Triển khai

a. Kế hoạch tổng thể của dự án

Mã	Công việc	Thời gian	Ngày bắt đầu	Ngày kết thúc
1	Thu thập và hiểu yêu cầu	?	?	?
2	Phân tích và thiết kế tổng thể	?	?	?
3	Phát triển	?	?	?
4	Kiểm thử hệ thống	?	?	?
5	Viết báo cáo	?	?	?
6	Bảo vệ	?	?	?

Hình 2. Kế hoạch tổng thể dự án

Kế hoạch này mang tính chất ước lượng khi mọi thứ còn chưa rõ ràng. Do vậy, bạn cần tham vấn giáo viên hoặc người có kinh nghiệm hơn để có những điều chỉnh cho hợp lý. Ngoài ra, kế hoạch này cũng cần phải điều chỉnh nhiều lần trong quá trình phát triển khi mọi thứ dần rõ ràng hơn.

b. Kế hoạch chi tiết của dự án

Kế hoạch chi tiết của dự án được thiết lập khi bạn đã thực hiện các bước ban đầu và mọi việc bắt đầu rõ ràng. Lúc này bạn cần xác định:

- Từng công việc chi tiết
- Ai làm
- Thời gian bao lâu

Mã	Công việc	Người thực hiện	Thời gian	Ngày bắt đầu	Ngày kết thúc
1	Phân tích Use Case	?	?	?	?
2	Thiết kế giao diện (Wire Frame)				
3	Thiết kế CSDL				
4	Thiết kế lớp				
5	Thiết kế Sequence Diagram				

Hình 3. Kế hoạch chi tiết của dự án

c. Kiểm soát kế hoạch của dự án

Xây dựng lên kế hoạch đã khó, việc kiểm soát kế hoạch đôi khi còn khó hơn. Do vậy, các bạn phải thường xuyên giám sát tiến độ của dự án, giải quyết các vấn đề phát sinh, đôi khi phải điều chỉnh kế hoạch cho phù hợp để đảm bảo thành công dự án. Giáo viên sẽ từng bước hướng dẫn bạn về việc này.

6. Quản lý yêu cầu

Quản lý và hiểu được yêu cầu là một việc làm vô cùng quan trọng để đảm bảo cho sự thành công của một dự án phần mềm. Theo mô hình Scrum việc quản lý yêu cầu được Product Owner chịu trách nhiệm chính. Tuy nhiên, các thành viên tham gia phát triển dự án cũng đóng một vai trò quan trọng. Bạn phải có cách để phân tích, hiểu và quản lý một cách hiệu quả các yêu cầu của dự án.

Trong phần này chúng ta sẽ làm rõ:

- Cách đặc tả yêu cầu
- Các loại yêu cầu
- Cách quản lý chúng

Ghi chú: Trong đồ án này Product Owner sẽ được giáo viên đảm nhận.

6.1 Đặc tả yêu cầu (Requirement Specification)

Đặc tả yêu cầu là công việc vô cùng quan trọng nhằm làm rõ các yêu cầu cần phát triển của hệ thống. Trong thực tế có rất nhiều cách đặc tả yêu cầu và đôi khi rất chi

tiết để nắm được. Tuy nhiên, trong phạm vi của đồ án này chúng ta chọn cách mô tả đơn giản nhất.

Bằng cách mô tả đơn giản nhất, mỗi yêu cầu nên được tóm tắt bằng một câu mà ở đó nó mô tả được nội dung mà hệ thống chúng ta muốn phát triển. Trong trường hợp những yêu cầu lớn chúng ta nên tách chúng ta thành những hệ thống con và sau đó phân tích chúng thành các yêu cầu đơn để phát triển. Cách phân tích này bạn có thể dùng Use Case Diagram hoặc Work Breakdown Structure để phân tích.

Ví dụ: Đặc tả yêu cầu phát triển hệ thống máy rút tiền tự động ATM

- Tạo và quản lý người sử dụng hệ thống.
- Xây dựng chức năng đăng nhập để người dùng đăng nhập khi sử dụng hệ thống.
- Hệ thống cho người dùng rút tiền từ ATM.
- Hệ thống cho người dùng xem số dư của tài khoản.
- Hệ thống cho người dùng in sao kê.
- Hệ thống cho người dùng chuyển khoản cho các tài khoản trong phạm vi của ngân hàng với số lượng không quá 20 triệu/ mỗi giao dịch và không quá 50 triệu/ngày
- Hệ thống có khả năng chạy một lúc đồng thời 100 máy ATM

Ghi chú: Bạn có thể tham khảo thêm cách sử dụng Use Case Diagram để phân tích hệ thống tại: <http://iviettech.vn/blog/543-ban-ve-use-case-use-case-diagram.html>

6.2 Các loại yêu cầu

Có rất nhiều cách phân loại yêu cầu nhưng có một phương thức được sử dụng phổ biến hiện nay có tên là FURPS. Viết tắt của các từ sau:

- Functionality: Yêu cầu chức năng.

- Usability: Yêu cầu tiện dụng.
- Reliability: Yêu cầu về độ tin cậy.
- Performance: Yêu cầu về hiệu năng.
- Supportability: Yêu cầu về hỗ trợ.

Một số ví dụ sau về các loại yêu cầu:

- Functionality: Yêu cầu chức năng.
 - Hệ thống yêu cầu chức năng đăng nhập cho người dùng.
 - Hệ thống cho người dùng rút tiền.
- Usability: Yêu cầu tiện dụng.
 - Giao diện người dùng phải thân thiện theo chuẩn giao tiếp GUI.
 - Người dùng không phải kích quá 3 lần để đến bất kỳ chức năng nào của hệ thống.
- Reliability: Yêu cầu về độ tin cậy.
 - Hệ thống phải phục hồi được khi mất điện hoặc máy bị sự cố.
- Performance: Yêu cầu về hiệu năng.
 - Hệ thống phải chạy được khi có 100 máy ATM dùng đồng thời.
- Supportability: Yêu cầu về hỗ trợ.
 - Hệ thống phải có hệ thống ghi lại log để xử lý khi có sự cố.

Trong thực tế, những yêu cầu này đôi khi bị trộn lẫn với nhau. Việc đáp ứng tất cả các yêu cầu ở mức cao nhất có thể làm tăng chi phí sản xuất. Do vậy, chúng ta phải cân nhắc khi đáp ứng các yêu cầu. Trong phạm vi đồ án này chúng ta chỉ chú trọng đến yêu cầu chức năng. Bạn có thể thêm một vài yêu cầu khác nhưng phải đảm bảo đúng tiến độ.

6.3 Yêu cầu chức năng

Yêu cầu chức năng là những chức năng cần có của hệ thống mà chúng ta phải phát triển. Việc phân tích các yêu cầu này bạn có thể sử dụng Use Case Diagram hoặc Work Breakdown Structure để thực hiện.

6.4 Quản lý các yêu cầu

Sau khi xác định danh sách các yêu cầu, việc tiếp theo chúng ta phải quản lý các yêu cầu đó để đảm bảo chúng được thực hiện đúng. Các hoạt động quản lý yêu cầu bao gồm:

- Lập danh sách các yêu cầu và đánh số ID.
- Xác định mức độ ưu tiên.
- Cập nhật hồ sơ phân tích, thiết kế cho mỗi yêu cầu.
- Kiểm tra kết quả thực hiện của các yêu cầu.

Mỗi dự án có thể có đến hàng trăm yêu cầu do vậy việc nhớ từng yêu cầu thực tế là một khó khăn cho nhóm phát triển dự án. Do vậy, chúng ta cần đánh số các yêu cầu và sử dụng công cụ để quản lý chúng.

Xác định mức độ ưu tiên: việc xác định mức độ ưu tiên sẽ giúp chúng ta xác định được nên làm cái nào trước, nên làm cái nào sau nhằm thuận lợi cho việc phát triển ứng dụng sau này. Mức độ ưu tiên có thể phân loại thành “Phải có”, “Nên có”, “Có thì tốt” để lập kế hoạch cho hợp lý.

ID	Yêu cầu	Mức độ ưu tiên
1	Tạo và quản lý người sử dụng hệ thống.	Phải có
2	Xây dựng chức năng đăng nhập để người dùng đăng nhập khi sử dụng hệ thống.	Phải có
3	Hệ thống cho người dùng rút tiền từ ATM.	Phải có
4	Hệ thống cho người dùng xem số dư của tài khoản.	Phải có
5	Hệ thống cho người dùng in sao kê.	Phải có
6	Hệ thống có khả năng chạy một lúc đồng thời 100 máy ATM	Nên có
7	Ghi log lại hệ thống giao dịch	Phải có
8	In biên lại ra giấy	Nên có
9	Hiển thị thông tin khuyến mãi của ngân hàng đến người dùng	Có thì tốt

Hình 4. Danh sách yêu cầu và mức độ ưu tiên

Ngay cả, trong danh sách yêu cầu phải có cũng cần có mức độ ưu tiên cái nào cao hơn hoặc cái nào cần phát triển trước để có kế hoạch thực hiện. Bạn cần làm việc thêm với giáo viên về vấn đề này.

Cập nhật hồ sơ phân tích, thiết kế cho mỗi yêu cầu: khi các yêu cầu được phân tích, thiết kế cụ thể bạn nên tham chiếu nó trong bản quản lý yêu cầu để dễ dàng tìm kiếm, kiểm tra sau này.

Kiểm tra kết quả thực hiện của các yêu cầu: định kỳ trong quá trình phát triển cần tổ chức xem xét, kiểm tra xem các yêu cầu đã thực hiện đến đâu nhằm biết được tình trạng của các yêu cầu.

ID	Yêu cầu	Mức độ ưu tiên	Tài liệu đính kèm	Tình trạng	Ghi chú
1	Tạo và quản lý người sử dụng hệ thống.	Phải có	Trang 12 tài liệu "Đặc tả yêu cầu phần mềm"	Hoàn tất	
2	Xây dựng chức năng đăng nhập để người dùng đăng nhập khi sử dụng hệ	Phải có			
3	Hệ thống cho người dùng rút tiền từ	Phải có			
4	Hệ thống cho người dùng xem số dư của tài khoản.	Phải có			
5	Hệ thống cho người dùng in sao kê.	Phải có			
6	Hệ thống có khả năng chạy một lúc đồng thời 100 máy ATM	Nên có			
7	Ghi log lại hệ thống giao dịch	Phải có			
8	In biên lại ra giấy	Nên có			
9	Hiển thị thông tin khuyến mãi của ngân hàng đến người dùng	Có thì tốt			

Hình 5. Bảng đặc tả yêu cầu chi tiết

Tóm lại, quản lý yêu cầu là một việc quan trọng và cần thiết trong quá trình phát triển dự án, dù là dự án lớn hay nhỏ. Vì vậy, bạn cần bỏ thời gian để đầu tư cho công việc này để đạt được kết quả tốt trong quá trình phát triển dự án.

7. Phân tích và thiết kế

Sau khi đặc tả cơ bản được các yêu cầu của hệ thống cần phát triển, bước tiếp theo chúng ta cần phân tích và thiết kế hệ thống. Có thể một số tài liệu tách biệt phân tích và thiết kế ra riêng nhưng trong thực tế rất khó tách biệt 02 hoạt động này. Chi tiết về phân tích và thiết kế hệ thống phần mềm các bạn được học trong môn “Phân tích và thiết kế hệ thống phần mềm theo mô hình hướng đối tượng sử dụng UML”, các bạn nên đọc thêm tài liệu này. Trong phần này chỉ giới thiệu một số nét cơ bản mang tính gợi ý.

7.1 Use Case Modeling

Use Case Modeling là một trong những công cụ phân tích rất mạnh được sử dụng rộng rãi hiện nay. Chúng bao gồm các thành tố sau:

- Use Case Diagram: giúp phân tích hệ thống và các ca sử dụng của hệ thống.
- Class Diagram: giúp thiết kế kiến trúc hệ thống.
- Activity Diagram: giúp phân tích nghiệp vụ của hệ thống.
- Sequence Diagram: giúp thiết kế chi tiết chức năng của hệ thống.

Bạn hãy tham khảo giáo trình môn học “Phân tích và thiết kế hệ thống phần mềm theo mô hình hướng đối tượng sử dụng UML” hoặc đọc trên Blog của website iviettech.vn để nắm thêm chi tiết.

<http://iviettech.vn/category/blog/phan-tich-thiet-ke-he-thong>

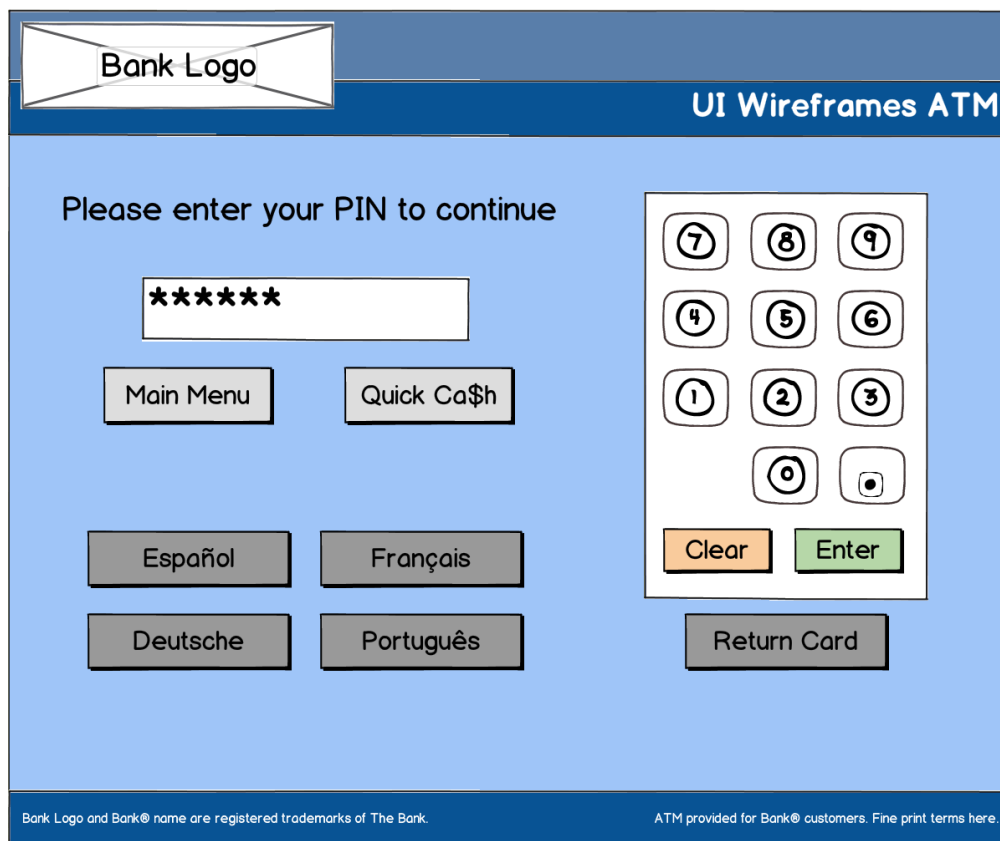
7.2 Giao diện người sử dụng (User Interface)

User Interface là phần không thể thiếu của các ứng dụng có tương tác với người sử dụng, nó cũng thường là phần mất nhiều thời gian nhất trong quá trình thiết kế nếu

chúng ta không phân tích và thiết kế một cách kỹ lưỡng và làm việc chặt chẽ với Product Owner và người dùng đầu cuối.

User Interface có quan hệ mật thiết với Use Case, thông thường nó được thiết kế để đáp ứng các Use Case. Vì vậy, nó được thiết kế dựa theo các Use Case và chi tiết các biểu mẫu áp dụng trong thực tế của hệ thống.

Để đảm bảo giao tiếp tốt giữa Product Owner, Người dùng cuối và nhóm phát triển người ta thường dùng bảng mẫu của giao diện người dùng thường được gọi là Prototype hay Wireframe.



Hình 6. Ví dụ về ATM Wire Frame

Bạn có thể dùng tay hoặc các phần mềm có sẵn trên mạng để vẽ các Wire Frame một cách nhanh chóng để giao tiếp với khách hàng trước khi tiến hành phát triển. Trong phạm vi của đồ án này, Giáo viên sẽ đóng vai trò là khách hàng và Product Owner.

7.3 Thiết kế CSDL (Database Design)

Cơ sở dữ liệu là một phần rất quan trọng nữa trong việc phát triển ứng dụng. Do vậy, bạn cần chú ý và bỏ thời gian để thiết kế kỹ CSDL để khỏi mất nhiều thời gian về sau. Chúng ta đã bàn kỹ vấn đề này trong môn học “Cơ sở dữ liệu dành cho Lập trình viên”. Nếu khóa học của bạn không học môn này bạn có thể tham khảo giáo trình môn học này hoặc nhờ sự giúp đỡ của giáo viên hướng dẫn.

8. Phát triển dự án

Phát triển dự án hay còn gọi là Coding chính là phần mà các bạn đã học trong suốt khóa học. Do vậy, ở đây không trình bày chi tiết về vấn đề này mà chỉ lưu ý các bạn một số điểm sau:

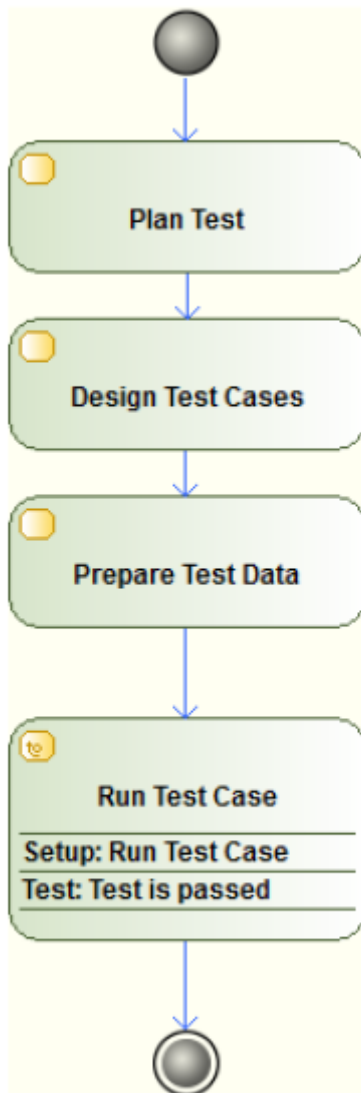
- Bám sát yêu cầu và thiết kế
- Tuân thủ Coding Convention
- Kiểm tra (Testing) một cách cẩn thận

Căn cứ vào danh sách yêu cầu và kết quả phân tích chúng ta lập ra danh sách các yêu cầu cần phát triển và được gọi là Product Backlog. Sau đó chúng ta phân vào từng giai đoạn phát triển gọi là Sprint Backlog.

9. Kiểm thử (Testing)

Sản phẩm của bạn sẽ thế nào? Nếu bạn không kiểm tra một cách cẩn thận. Tuy nhiên, trong thực tế các Lập trình viên hay bỏ qua giai đoạn này hoặc chỉ làm cho có. Vì nghĩ rằng đã có tester nên các bạn rất hay viết cầu thả chương trình của mình. Điều này sẽ gây ra nhiều vấn đề nghiêm trọng, ảnh hưởng đến năng suất cũng như con đường tiến thân của bạn. Trong phần này chúng ta sẽ tiếp cận kiểm thử theo hướng của một lập trình viên. Nếu bạn muốn tiếp cận theo hướng Tester chuyên nghiệp hãy tham khảo giáo trình kiểm thử chuyên nghiệp.

9.1 Testing Process



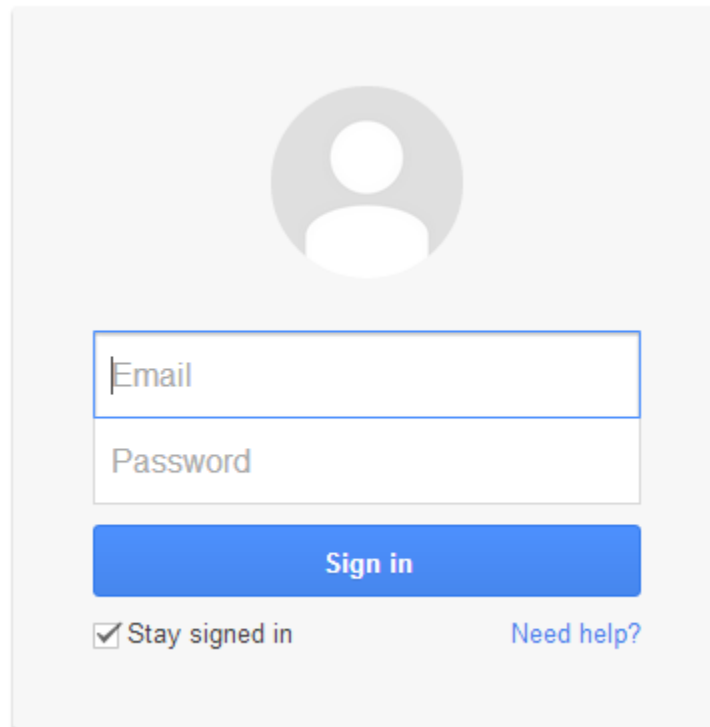
Hình 7. Testing Process

9.2 Test Plan

Trước tiên bạn cần lập một kế hoạch test để đảm bảo bạn có thời gian để thực hiện việc đó một cách nghiêm túc. Kế hoạch này phải được tích hợp vào kế hoạch tổng thể của dự án để mọi người đều nhận biết. Để lập được kế hoạch kiểm thử cho dự án, bạn cần phải hiểu rõ yêu cầu Test cho dự án, thời gian cho phép và nguồn lực của dự án. Về biểu mẫu thì phần này cũng tương tự như biểu mẫu kế hoạch của dự án.

9.3 Design Test Case

Test Case là các trường hợp Test cho một chức năng hay một yêu cầu Test nào đó.
Ví dụ để Test chức năng Login của gmail bạn phải có những Test Case sau:



Hình 8. Chức năng Login của Gmail

Các Test case cho chức năng Login của Gmail:

- Kiểm tra việc nhập đúng định dạng của Email
- Kiểm tra Email và Password đúng
- Kiểm tra Email hoặc Password sai
- Kiểm tra trường hợp sai quá 3 lần thì phải xác minh (để tránh đăng nhập tự động)

Việc thiết kế Website đòi hỏi người có kinh nghiệm và hiểu rõ yêu cầu đặt ra.

9.4 Test Data

Test Case chỉ ra các trường hợp cần Test còn Test Data là dữ liệu được sử dụng để Test các trường hợp đó. Nếu bạn không chuẩn bị tốt dữ liệu thì khả năng bạn phát hiện lỗi cũng rất kém. Ví dụ để kiểm tra việc nhập đúng định dạng Email là Gmail bạn có thể chuẩn bị dữ liệu như sau:

Test Data		0	
Test Case ID	Test Objective	Data	Expectation
1	Nhập sai địa chỉ email	Johnsmith	Thông báo "Bạn nhập sai địa chỉ email"
1	Nhập sai địa chỉ Gmail	Johnsmith@yahoo.com	Thông báo "Bạn phải sử dụng địa chỉ Gmail"

Hình 9. Ví dụ về Test Data

Bạn cần phải chuẩn bị dữ liệu sao cho khả năng phát hiện ra lỗi là lớn nhất.

9.4 Thực hiện Test

Bạn có thể thực hiện Test bằng tay (manual) hoặc tự động bằng chương trình (Automation Test) tùy vào khả năng cũng như yêu cầu của giáo viên.

9.5 Test Result

Kết quả test phải được ghi rõ và được báo cáo đến Product Owner để giám sát và kiểm tra. Kết quả test được thể hiện bằng việc đúng hay sai. Nếu có thể bạn nên chụp hình lại hoặc phán đoán lý do để thuận lợi cho quá trình Debug và Fix lỗi sau này.

No. of Test Cases		0			
Test Case ID	Test Objective	Steps	Expectation	Results	Notes
1	Kiểm tra trường hợp đăng ký kí danh trống	1. Nhập ký danh trống 2. Kích nút đăng ký	Thông báo lỗi "Kí danh không được trống, vui lòng nhập kí danh"	Pass	
2	Kiểm tra trường hợp đăng ký kí danh trùng	1. Nhập ký danh đã có trong hệ thống 2. Kích nút đăng ký	Thông báo lỗi "Kí danh này đã tồn tại vui lòng nhập kí danh khác"	Pass	
3	Kiểm tra trường hợp đăng ký kí danh không có độ dài không quá 50 ký tự.	1. Nhập tên ký danh có độ dài hơn 50 ký tự 2. Kích nút đăng ký	Thông báo lỗi "hợp lệ"	Fail	Vẫn chấp nhận trường hợp quá 50 ký tự

Hình 10. Mẫu báo cáo kết quả Test

10. Viết báo cáo

Trong khuôn khổ của đồ án này bạn phải viết một báo cáo tương đối hoàn chỉnh và nó là một phần quan trọng để bạn có đủ điều kiện để bảo vệ đồ án cuối khóa. Theo cách nghĩ thông thường thì tài liệu là phần bạn hoàn thành cuối cùng, tuy nhiên bạn không nên để quá trễ nó sẽ gây cản trở cho quá trình hoàn thành đồ án của bạn. Bạn nên bắt đầu sớm phần viết tài liệu này ngay khi giáo viên thông qua mỗi phần. Bạn nên sử dụng mẫu tài liệu của dự án mà Trung tâm cung cấp ở phía sau của tài liệu này để việc chuẩn bị của bạn được dễ dàng.

Thông thường, cấu trúc một tài liệu đồ án bao gồm những phần sau:

- Bìa
- Giới thiệu và cảm ơn
- Mục lục
- Giới thiệu về đề tài
- Giải pháp
- Đặc tả và quản lý yêu cầu
- Phương thức quản lý
 - + Nhóm dự án
 - + Phương thức quản lý

- + Kế hoạch thực hiện
- Phân tích
- Thiết kế
 - + Thiết kế giao diện
 - + Thiết kế CSDL
 - + Thiết kế các chức năng
 - + Cài đặt
- Kiểm thử
- Kết luận và mở rộng
- Tài liệu tham khảo

10.1 Bìa

Là trang đầu tiên của tài liệu, mô tả thông tin về Trung tâm, học viên, khóa học, giáo viên hướng dẫn ...

10.2 Giới thiệu và cảm ơn

Trang này giới thiệu về cảm nghĩ của bạn khi thực hiện đồ án và cảm ơn Giáo viên hướng dẫn cũng như những người đã giúp đỡ bạn hoàn thành đồ án này.

10.3 Mục lục

Phần mục lục bao gồm mục lục của phần nội dung, ảnh, bảng biểu.

10.4 Giới thiệu về đề tài

Phần này bạn giới thiệu về yêu cầu của đề tài của bạn.

10.5 Giải pháp

Từ yêu cầu bạn định nghĩa ra giải pháp bạn định dùng để xử lý như bạn chọn phát triển ứng dụng đa tầng, ứng dụng Web với kiến trúc như thế nào.

Bạn cần có lý luận vì sao bạn chọn giải pháp đó và nó thỏa mãn với yêu cầu được đưa ra ở phần 10.4.

10.5 Phương thức quản lý

Bạn cần nêu rõ phương pháp quản lý, thông thường ở đây chúng ta sử dụng SCRUM để quản lý nên bạn chỉ cần ghi là sử dụng phương thức quản lý nào mà thôi, không cần giải thích kỹ hơn. Bên cạnh đó, bạn cần mô tả nhóm dự án, vai trò của từng người cũng như bản phân công của các thành viên trong nhóm. Cuối cùng là bảng kế hoạch thực hiện dự án chi tiết.

Ngoài ra, bạn cũng cần định nghĩa các công cụ để quản lý Yêu cầu, Task, Bug, Source Code Control, công cụ phát triển và một số qui định khác.

10.6 Phân tích và thiết kế

Phần này bạn mô tả những bảng vẽ phân tích, thiết kế lên giao diện, chức năng, CSDL. Ưu tiên sử dụng hình vẽ và giải thích một cách ngắn gọn.

10.7 Cài đặt

Nêu ra những chức năng đã thực hiện, kết quả thế nào? Và nơi lưu trữ mã lệnh.

10.8 Kiểm thử

Bạn nêu rõ bạn đã kiểm thử được những nội dung nào, kết quả ra sao.

10.9 Kết luận và mở rộng

Bạn nêu rõ kết quả đạt được qua đồ án như kết quả làm được, kết quả học được và định hướng mở rộng hay áp dụng sản phẩm trong tương lai.

10.10 Tài liệu tham khảo

Liệt kê tài liệu tham khảo.

11. Tài liệu hỗ trợ của dự án

Tài liệu hỗ trợ dự án là các thành phần để giúp người dùng, người đánh giá, người được chuyển giao ... hiểu được sản phẩm phần mềm của bạn. Tài liệu này thể hiện qua các tài liệu sau:

- Mã nguồn
- Tài liệu hướng dẫn cài đặt
- Tài liệu hướng dẫn sử dụng

11.1 Mã nguồn

Phần mã lệnh cũng là một tài liệu rất quan trọng của dự án phục vụ cho việc bảo trì, nâng cấp sau này. Do vậy, cần viết mã lệnh rõ ràng, trong sáng và có chú thích cần thiết để người đọc có thể hiểu được.

11.2. Tài liệu hướng dẫn cài đặt

Tài liệu này hướng dẫn chi tiết yêu cầu phần cứng, phần mềm, cách cấu hình, các bước cài đặt để hệ thống có thể chạy được.

11.4. Tài liệu hướng dẫn sử dụng

Tài liệu này hướng dẫn từng bước cách sử dụng các chức năng của sản phẩm.

12. Bảo vệ dự án

Bảo vệ đồ án là khâu cuối cùng trong hành trình thực hiện đồ án của bạn và nó cũng là khâu quan trọng nhất. Nó cũng giống như cách các bạn đi bán sản phẩm cho khách hàng. Bạn phải làm sao để trong một khoảng thời gian cho phép khoảng 30 phút, các bạn có thể giới thiệu được đầy đủ tính năng, ưu điểm của dự án đến khách hàng. Do vậy, bạn phải chuẩn bị rất kỹ và tập đi, tập lại bài trình bày trước khi tiến hành buổi bảo vệ chính thức.

Dưới đây là cấu trúc của một bài trình bày khoảng 20 phút chưa bao gồm phần hỏi đáp.

Phân bổ thời gian: 20 phút

- Chuẩn bị: 5 phút
- Trình bày: 10 phút
- Demo: 5 phút

Số Slide: 18

Cấu trúc của bài trình bày:

- Slide 1: Thông tin về đề án và nhóm dự án (30 giây)
- Slide 2: Cấu trúc nội dung sẽ trình bày (30 giây)
- Slide 3: Nêu nội dung của đề tài (1 phút)
- Slide 4: Nêu những yêu cầu chính của đề tài (1 phút)
- Slide 5: Tóm tắt giải pháp (1 phút)
- Slide 6-10: Kết quả phân tích, thiết kế (5 phút)
- Slide 11-15: Kết quả thực hiện được (5 phút)
- Slide 16: Demo (5 phút)
- Slide 17: Kết luận và hướng mở rộng (1 phút)
- Slide 18: Cảm ơn và hỏi đáp

12.1 Công tác chuẩn bị

Việc chuẩn bị được chia làm 2 giai đoạn. Giai đoạn 1 bạn phải chuẩn bị tất cả mọi thứ từ sản phẩm, tài liệu, bài trình bày, diễn tập.. Các công việc này phải hoàn thành trước ít nhất 1 ngày trước khi bảo vệ.

Còn thời gian chuẩn bị ở đây để bạn sử dụng cho việc kết nối máy tính vào máy chiếu, chỉnh đốn lại vị trí của các thành viên trong nhóm, chỉnh đốn trang phục, hít sâu để lấy tinh thần v.v..

12.2 Trình bày phần phân tích, thiết kế

Phần này bạn nên trình bày từ 3-5 Slide, nên bạn cần lựa chọn những nội dung quan trọng nhất để trình bày. Nên sử dụng hình, hạn chế chữ.

12.3 Tóm tắt kết quả đạt được

Tương tự như phần trên, bạn cần chọn ra những chức năng chất lượng nhất của bạn để trình bày và cũng chỉ nói sơ qua để giám khảo thấy được kết quả sản phẩm của bạn. Tránh đưa vào quá nhiều, không cần thiết.

12.4 Demo

Bạn có 5 phút để Demo nên bạn cần chọn những chức năng quan trọng để Demo nhằm thu hút người nghe. Ngoài ra, nếu các thành viên nhóm dự án thực hiện các chức năng khác nhau thì bạn cũng nên Demo các chức năng để thể hiện được công việc của các thành viên trong nhóm.

13. Công cụ phát triển dự án

Trong quá trình phát triển dự án, bạn có thể dùng nhiều công cụ hỗ trợ để phát triển và quản lý. Trong phần này sẽ liệt kê một số công cụ để giúp bạn có lựa chọn tốt khi phát triển dự án.

- IDE: Phần này các bạn đã học trên lớp nên không đề cập ở đây.
- Công cụ quản lý dự án: bạn có thể sử dụng Redmine hoặc Jira là hai công cụ mã nguồn mở và có phiên bản Online miễn phí để sử dụng trong khoảng thời gian bạn phát triển dự án. Bạn có thể xem thêm tại: <https://www.hostedredmine.com/> hoặc <https://www.atlassian.com/software/jira/try/>. Jira yêu cầu phải có thể tín dụng nên có thể khó hơn.
- Source Code Control: phần mềm này sẽ giúp bạn làm việc theo nhóm, chia sẻ code mà không lo cập nhật nhầm hay mất nhiều thời gian để nhập code

lại sau khi làm xong. Hiện tại bitbucket đang miễn phí 5 Users có thể là lựa chọn rất tốt cho các bạn. Xem tại: <https://bitbucket.org> để biết thêm chi tiết.

- Hosting: Thông thường các bạn sẽ cài đặt Localhost để chủ động. Tuy nhiên, bạn có thể chọn một số dịch vụ Hosting miễn phí để đỡ mất công cài đặt và thuận lợi cho quá trình triển khai.

14. Form mẫu cho dự án

Phần này sẽ mô tả chi tiết ở phần phụ lục của tài liệu này.

15. Cách đánh giá đồ án

Việc đánh giá đồ án được chia làm 2 phần:

- Tài liệu và quá trình thực hiện đồ án
- Bảo vệ đồ án

15.1 Tài liệu và bảo vệ đồ án

Phần này do giáo viên hướng dẫn đánh giá dựa trên kết quả trong suốt quá trình thực hiện đồ án của bạn cũng như kết quả thể hiện qua tài liệu của đồ án.

15.2 Bảo vệ đồ án

Phần này do hội đồng bảo vệ đánh giá, điểm của bạn sẽ được tính theo giá trị trung bình của các giám khảo tham gia chấm bảo vệ.

Nếu không đạt phần tài liệu bạn sẽ không được tham gia bảo vệ. Nếu không đạt 1 trong 2 phần bạn sẽ phải làm lại đồ án.

16. Thời lượng và tiến độ

Session	Nội dung
S1	<ul style="list-style-type: none">- Giới thiệu về dự án- Qui trình, công cụ khi thực hiện dự án- Các yêu cầu đầu vào, đầu ra khi thực hiện dự án- Cách đánh giá, Phân chia nhóm, Hướng dẫn làm việc nhóm- Hướng dẫn lựa chọn đề tài
S2	<ul style="list-style-type: none">- Hiệu chỉnh, thông qua đề tài- Hướng dẫn, phân tích thiết kế dự án
S3	<ul style="list-style-type: none">- Phân tích và thiết kế tổng thể dự án:<ul style="list-style-type: none">o Phân tích yêu cầuo Thiết kế hệ thống: Thiết kế chức năng, Thiết kế giao diện, Thiết kế CSDL.
S4	<ul style="list-style-type: none">- Bắt đầu Sprint 1: Mỗi Sprint 1 tuần<ul style="list-style-type: none">o Họp Sprint Planningo Tiến hành thực hiện
S5	<ul style="list-style-type: none">- Sprint 1<ul style="list-style-type: none">o Daily Meetingo Tiếp tục phát triển
S6	<ul style="list-style-type: none">- Sprint 1<ul style="list-style-type: none">o Daily Meetingo Tiếp tục phát triển
S7	<ul style="list-style-type: none">- Bắt đầu Sprint 2: Mỗi Sprint 1 tuần<ul style="list-style-type: none">o Họp Review Sprint 1 và Họp Sprint Planning Sprint 2
S8	<ul style="list-style-type: none">- Sprint 2<ul style="list-style-type: none">o Daily Meetingo Tiếp tục phát triển
S9	<ul style="list-style-type: none">- Sprint 2<ul style="list-style-type: none">o Daily Meetingo Tiếp tục phát triểno Sprint 2 Review

Sau khi kết thúc Session 9, giáo viên cho thêm khoảng 1 tuần để học viên hoàn thành tài liệu và sản phẩm và nộp lại cho giáo viên chấm và ấn định ngày bảo vệ. Bảng tiến độ trên chỉ mang tính tham khảo. Tùy theo yêu cầu của môn học giáo viên hướng dẫn sẽ điều chỉnh tiến độ cho phù hợp.

17. Tài liệu tham khảo

- How to do the final year projects, Hossein Hassani