

NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Khóa luận đáp ứng yêu cầu của Khóa luận cử nhân CNTT.

TpHCM, ngày..... tháng.....năm 2012

Giáo viên hướng dẫn

[Ký tên và ghi rõ họ tên]

LỜI CẢM ƠN

Đầu tiên, chúng em xin gửi lời cảm ơn đến Thầy, Cô khoa Công nghệ Thông tin trường Đại học Khoa học Tự nhiên đã tận tình dạy dỗ, dìu dắt chúng em suốt bốn năm đại học.

Chúng em cảm ơn Thầy Nguyễn Văn Vũ, người tận tình hướng dẫn, giúp đỡ, động viên chúng em hoàn thành luận văn này.

Cuối cùng, chúng con cảm ơn Ba, Mẹ và những người thân đã khích lệ, động viên chúng con trong thời gian học tập, nghiên cứu để có được thành quả như ngày nay.

Tháng 6 năm 2012

Sinh viên

Đinh Văn Hoàng – Nguyễn Đức Xuân

ĐỀ CƯƠNG CHI TIẾT

Tên Đề Tài:	
Hệ thống quản lý tương tác trong các dự án phần mềm	
Giáo viên hướng dẫn:	
TS. Nguyễn Văn Vũ	
Thời gian thực hiện:	
Từ 01/12/2011 đến 30/06/2012	
Sinh viên thực hiện:	
Đình Văn Hoàng	0812164
Nguyễn Đức Xuân	0812642
Loại đề tài:	
Xây dựng và triển khai giải pháp	

Nội Dung Đề Tài:
<p>❖ Lý thuyết:</p> <ul style="list-style-type: none"> • Tìm hiểu Collaborative Development Environments (CDE) • Tìm hiểu những vấn đề liên quan đến việc tạo lập và quản lý tương tác của 1 project <p>❖ Đề xuất, lựa chọn giải pháp :</p> <ul style="list-style-type: none"> • Đề xuất các hướng giải quyết trong vấn đề quản lý tương tác giữa các thành viên khi tạo lập 1 project • Lựa chọn phương án tối ưu để thực hiện đề tài

❖ **Thử nghiệm:**

- Triển khai và cài đặt giải pháp trên ứng dụng web kết hợp với web server

Kế Hoạch Thực Hiện:

❖ **Giai đoạn 1** : Từ 01-12-2011 đến 31-12-2011

- Tìm hiểu CDE
- Liệt kê những tính năng cần có, những tính năng có thể thêm nhằm hướng tới mục tiêu của đề tài
- Yêu cầu hệ thống

❖ **Giai đoạn 2**: Từ 01-01-2012 đến 31-01-2012

- Tìm hiểu process template
- Đề xuất hướng giải quyết

❖ **Giai đoạn 3**: Từ 01-02-2012 đến 29-02-2012

- Xây dựng usecase
- Thiết kế kiến trúc, các tổ chức dữ liệu
- Design giao diện ứng dụng
- Define những tính năng cần có

❖ **Giai đoạn 4**: Từ 01-03-2012 đến 31-03-2012

- Cài đặt và kiểm tra các tính năng

❖ **Giai đoạn 5** : Từ 01-04-2012 đến 30-04-2012

- Cài đặt và kiểm tra các tính năng

❖ **Giai đoạn 6** : Từ 01-05-2012 đến 31-05-2012

- Hoàn chỉnh tính năng
- Test để tìm lỗi và fix lỗi

❖ **Giai đoạn 7** : Từ 01-06-2012 đến 30-06-2012

- Viết báo cáo các phần đã tìm hiểu và làm được.

<ul style="list-style-type: none"> • Đưa ra các giả thuyết và nhận định để tìm ra các ưu khuyết điểm. • Tổng kết lại các phần và hướng phát triển của đề tài. • Hoàn tất 	
<p>Xác nhận của GVHD</p> <p>TS. Nguyễn Văn Vũ</p>	<p>Ngày 10 tháng 01 năm 2012</p> <p>SV Thực hiện</p> <p>Đinh Văn Hoàng Nguyễn Đức Xuân</p>



MỤC LỤC

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	i
NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN	ii
LỜI CẢM ƠN	iii
ĐỀ CƯƠNG CHI TIẾT	iv
MỤC LỤC	vii
DANH SÁCH CÁC BẢNG	xi
DANH SÁCH CÁC HÌNH VẼ	xii
Chương 1 MỞ ĐẦU	1
Chương 2 TỔNG QUAN	3
Chương 3 KHẢO SÁT HIỆN TRẠNG	5
3.1 Cơ sở lý thuyết	5
3.2 Tìm hiểu thực tế	7
3.2.1 Team foundation server (TFS)	7
3.2.2 Redmine	10
3.2.3 Assembla	11
Chương 4 HỆ THỐNG QUẢN LÝ TƯƠNG TÁC	13
4.1 Chức năng	13
4.1.1 Quản lý quy trình phát triển phần mềm	13
4.1.2 Quản lý dự án	13
4.1.3 Quản lý lớp học	13

4.1.4	Quản lý lịch cá nhân và lịch dự án.....	13
4.1.5	Quản lý work item và tương tác trong nhóm.....	14
4.2	Yêu cầu.....	14
4.2.1	Yêu cầu chức năng.....	14
4.2.1.1	Danh sách các Actor	15
4.2.1.2	Sơ đồ Use-Case	16
	Mô hình Use-Case.....	16
	Account management.....	17
	Project management	18
	Project activity	19
4.2.1.3	Đặc tả Use-Case chính	19
	Use-Case Create Work Item	19
	Use-Case Update Work Item	20
	Use-Case Create account.....	21
	Use-Case Update account	22
	Use-Case Delete account.....	23
	Use-Case Create user's event.....	24
	Use-Case Update user's event.....	24
4.2.1.4	Sơ đồ lớp mức phân tích	25
4.2.2	Yêu cầu phi chức năng.....	28
4.3	Thiết kế.....	28

4.3.1	Mô hình triển khai.....	29
4.3.2	Kiến trúc tổng quan.....	29
4.3.2.1	Các package chính trong hệ thống	32
4.3.3	Các lớp chính trong domain model.....	32
4.3.3.1	Quản lý nhóm và dự án:	33
4.3.3.2	Quản lý work item và tương tác trong nhóm:	34
4.3.3.3	Quản lý lịch.....	37
4.3.4	Cơ sở dữ liệu.....	37
4.3.5	Cơ chế bảo mật	38
4.3.6	Thiết kế giao diện	39
4.3.6.1	Cấu trúc chung của giao diện.....	39
4.3.6.2	Giao diện khi người dùng ở trang quản lý của admin.....	40
4.3.6.3	Giao diện khi người dùng ở trang của dự án.....	41
4.3.6.4	Giao diện nhà người dùng.....	43
4.3.6.5	Màn hình overview của dự án.....	44
4.3.6.6	Màn hình dashboard của người dùng:.....	44
4.3.6.7	Màn hình account calendar:	44
4.3.6.8	Màn hình project calendar	45
4.3.6.9	Màn hình hiển thị work item.....	46
4.3.7	Thư viện hỗ trợ:	48
4.4	Hướng dẫn sử dụng.....	48

4.4.1	Tạo process	49
4.4.2	Quản lý project.....	50
Chương 5	KẾT LUẬN.....	54
Chương 6	HƯỚNG PHÁT TRIỂN.....	55
Chương 7	DANH MỤC TÀI LIỆU THAM KHẢO.....	56
Chương 8	PHỤ LỤC	57
	Phụ lục A: Các khái niệm cơ bản.....	57
	Phụ lục B: Các màn hình chi tiết.....	58
	Phụ lục C: Các thư viện hỗ trợ	60

DANH SÁCH CÁC BẢNG

Bảng 1 Danh sách actor.....	16
Bảng 2 Các lớp đối tượng mức phân tích	28

DANH SÁCH CÁC HÌNH VẼ

Hình 1 Team Foundation Server Proccess	7
Hình 2 CMMI.....	8
Hình 3 Agile	9
Hình 4 Danh sách actor	15
Hình 5 Mô hình use case trong nhóm chức năng quản lý tài khoản	17
Hình 6 Mô hình use case trong nhóm chức năng quản lý dự án	18
Hình 7 Mô hình use case trong nhóm chức năng quản lý hoạt động trong dự án	19
Hình 8 Sơ đồ lớp mức phân tích	26
Hình 9 Mô hình triển khai	29
Hình 10 Kiến trúc tổng quan	30
Hình 11 Mô hình MVC	31
Hình 12 Data Mapper Pattern	31
Hình 13 Các lớp chính liên quan tới việc quản lý nhóm và dự án.....	34
Hình 14 Các lớp chính liên quan tới quản lý work item và tương tác	36
Hình 15 Các lớp chính liên quan tới quản lý lịch	37
Hình 16 Class Table Inheritance	38
Hình 17 Mô hình vai trò của người dùng trong dự án	39
Hình 18 cấu trúc chung của giao diện.....	39
Hình 19 Giao diện admin	41
Hình 20 Giao diện project.....	42

Hình 21 Giao diện nhà người dùng.....	43
Hình 22 Lịch cá nhân	45
Hình 23 Lịch dự án.....	46
Hình 24 Hiện thị work item	47
Hình 25 Các thư viện hỗ trợ.....	48
Hình 26 Màn hình login	49

Chương 1 MỞ ĐẦU

Đề án được sử dụng thường xuyên trong các môn học thuộc ngành công nghệ thông tin, thường được thực hiện bởi nhóm 2-7 sinh viên nhằm giải quyết một vấn đề. Thông qua việc thực hiện đề án sinh viên có thể áp dụng kiến thức vào thực tế, từ đó hiểu sâu vấn đề hơn. Không những vậy việc thực hiện đề án còn giúp sinh viên rèn luyện kỹ năng làm việc nhóm, một kỹ năng rất quan trọng đối với kỹ sư công nghệ phần mềm.

Bằng cách kiểm tra việc thực hiện đề án của sinh viên, giáo viên có thể đánh giá được sinh viên có đạt được mục tiêu đề ra của môn học hay không, đồng thời có thể đánh giá phương pháp truyền đạt kiến thức của mình có hiệu quả không.

Tuy nhiên hiện nay giáo viên cũng như sinh viên đang gặp nhiều khó khăn trong việc quản lý và thực hiện đề án.

Hiện nay để thực hiện đề án sinh viên phải sử dụng nhiều công cụ khác nhau như Google Code, Google Group, Yahoo và Skype. Tuy nhiên các công cụ này mang tính rời rạc, không thống nhất. Do đó sinh viên khó có thể áp dụng các quy trình phát triển phần mềm để thực hiện đề án..

Bên cạnh đó việc sinh viên sử dụng các công cụ rời rạc, không thống nhất để thực hiện đề án, gây ra nhiều khó khăn cho giáo viên trong việc đánh giá và quản lý quá trình thực hiện đề án của sinh viên. Hiện nay giáo viên đánh giá đề án của sinh viên chỉ dựa trên kết quả sau cùng. Cách thức thực hiện đề án cũng là một phần rất quan trọng, tuy nhiên đã bị bỏ qua khi đánh giá đề án.

Không chỉ vậy do sinh viên phải học cùng lúc nhiều môn và tham gia nhiều đề án nên thời gian rảnh không giống nhau gây khó khăn trong việc lên kế hoạch cho các hoạt động của nhóm.

Từ thực tế trên nhóm em, được sự hướng dẫn của thầy Nguyễn Văn Vũ, đã quyết định thực hiện luận văn với đề tài “**Xây dựng hệ thống quản lý, tương tác cho các đề án môn học**”

với tên gọi “**TeamSpace**” nhằm hỗ trợ sinh viên, giáo viên trong việc quản lý, thực hiện đề án.

Mục tiêu của đề tài là tạo ra một hệ thống chung, thống nhất để các nhóm sinh viên thực hiện đề án. Hệ thống này sẽ cung cấp các chức năng cơ bản để thành lập nhóm, quản lý nhóm, tương tác trong nhóm và quản lý lịch của cá nhân và nhóm. Việc thực hiện đề án trên một hệ thống chung, thống nhất sẽ giúp sinh viên dễ dàng áp dụng các quy trình phát triển phần mềm vào quá trình làm đề án, cho phép các thành viên trong nhóm làm đề án tương tác với nhau dễ dàng hơn. Hệ thống này sẽ giúp giáo viên dễ dàng hơn trong việc quản lý, đánh giá đề án của sinh viên. Việc đánh giá đề án không chỉ dựa trên kết quả sau cùng mà còn dựa trên cách thức thực hiện đề án của sinh viên. Sử dụng TeamSpace, sinh viên trong nhóm dễ dàng lên lịch cho hoạt động của nhóm dựa vào thời gian rảnh của các thành viên.

Chương 2 TỔNG QUAN

Hiện nay, giáo viên sử dụng trang môn học (moodle) để giao đề án cho sinh viên. Đồng thời giáo viên sẽ giải đáp những thắc mắc của sinh viên trực tiếp trên lớp hoặc thông qua diễn đàn trên trang môn học. Khi sinh viên hoàn thành đề án sẽ nộp kết quả, báo cáo cho giáo viên để đánh giá. Giáo viên có thể vấn đáp sinh viên về đề án nếu cần.

Về phía sinh viên, sau khi nhận đề án tùy nhóm sẽ sử dụng các công cụ khác nhau để hỗ trợ quản lý nhóm, tương tác trong nhóm. Sinh viên sẽ sử dụng Google Group, Yahoo, Skype, ... cho việc tương tác giữa các thành viên trong nhóm. Quản lý nhóm, cụ thể là phân chia công việc cho các thành viên trong nhóm, được thực hiện bằng cách họp nhóm sau đó ghi nhận kết quả phân chia và lưu tại một nơi chung của nhóm như Google Group và Google Code.

Cách thực hiện đề án như trên gặp nhiều hạn chế.

Một là, sinh viên sử dụng nhiều công cụ khác nhau. Giáo viên lựa chọn phương pháp, công cụ một cách tùy biến, không thống nhất. Do đó sẽ làm ảnh hưởng đến hiệu quả của việc giảng dạy và thực hiện đề án.

Hai là, đề án được thực hiện bằng nhiều công cụ, phương pháp khác nhau, không thống nhất. Do đó gây khó khăn trong việc đánh giá đề án. Đề án chỉ được đánh giá dựa vào kết quả sau cùng. Quá trình thực hiện đề án không được kiểm tra và đánh giá.

Ba là, với cách thực hiện đề án như hiện tại các nhà nghiên cứu không có một môi trường thống nhất để thu thập dữ liệu dữ liệu phục vụ cho việc phân tích, đánh giá và thử nghiệm các phương pháp, các cách tiếp cận mới trong phát triển phần mềm.

Bốn là việc thực hiện đề án chỉ giới hạn trong nhóm sinh viên, chưa có sự tham gia của công ty bên ngoài. Do đó, sinh viên sẽ không tiếp thu được nhiều kinh nghiệm thực tế khi thực hiện đề án. Việc doanh nghiệp chưa quan tâm tới đề án của sinh viên có thể do quy trình, cách thức thực hiện đề án của sinh viên không sát với những gì công ty phần mềm sử dụng.

Năm là việc tương tác trong nhóm sinh viên phải sử dụng nhiều công cụ không thống nhất như Skype, Yahoo.

Và cuối cùng là việc lên kế hoạch họp nhóm gặp nhiều khó khăn do thời gian rảnh của các thành viên trong nhóm không giống nhau.

Để khắc phục những hạn chế đã phân tích ở trên, đề tài này sẽ tập trung nghiên cứu và xây dựng hệ thống quản lý và tương tác trên môi trường web. Hệ thống này sẽ tạo ra môi trường thống nhất trong việc thực hiện đề án. Hệ thống sẽ có những tính năng cơ bản sau:

Tính năng đầu tiên là cho phép sinh viên hoặc giảng viên thành lập các nhóm làm việc. Trong nhóm sẽ có sự tương tác giữa các thành viên, tương tác giữa giáo viên và thành viên của nhóm. Hệ thống sẽ cung cấp các chức năng nhằm đảm bảo sự tương tác này được diễn ra thuận lợi, giúp giảm thiểu chi phí thời gian trong quá trình thực hiện dự án.

Tính năng thứ hai là cho phép lập kế hoạch cho đề án với những quy trình phần khác nhau. Lên kế hoạch làm việc với những mốc thời gian cụ thể, mục tiêu rõ ràng, đảm bảo mọi thứ đều nằm trong lịch trình định sẵn, giảm thiểu rủi ro, tăng hiệu suất công việc.

Tính năng thứ ba là quản lý và phân công tác vụ cho từng thành viên. Với chức năng này, người quản lý sẽ dễ dàng phân công, theo dõi đánh giá được tiến độ công việc. Đồng thời đánh giá được năng lực của mỗi thành viên.

Tính năng thứ tư là giúp theo dõi tiến độ đề án. Người quản lý sẽ dễ dàng theo dõi được tiến độ công việc, đưa ra những điều chỉnh thích hợp và kịp thời nhất.

Tính năng thứ năm là hỗ trợ lên lịch trình cho hoạt động của dự án. Quản lý dự án sẽ không phải tốn thời gian và chi phí liên hệ với từng thành viên để biết thời gian rảnh của các thành viên.

Cuối cùng là cho phép tương tác trên môi trường Web, đảm bảo được sự nhanh chóng và lưu động. Mỗi thành viên có thể tương tác bất cứ khi nào, ở đâu, chỉ cần có kết nối internet. Nhờ vậy, quy trình xây dựng dự án sẽ được diễn ra một cách liên tục.

Chương 3 KHẢO SÁT HIỆN TRẠNG

3.1 Cơ sở lý thuyết

Con người có những giới hạn ảnh hưởng đến việc làm phần mềm. Đó là khi phải làm việc ở mức độ trừu tượng cao - như viết yêu cầu, thiết kế hệ thống và viết mã nguồn - tốc độ làm việc chậm và khó phát hiện lỗi. Do đó cần phải hợp tác để có thể thực hiện các dự án phần mềm lớn với thời gian hợp lý. Bên cạnh đó khi hợp tác với nhau thì lỗi của một người có thể được phát hiện dễ dàng bởi người khác.

Tuy nhiên khi hợp tác với nhau để làm phần mềm lại nảy sinh vấn đề khác. Đó là việc sử dụng ngôn ngữ tự nhiên dễ gây ra cách hiểu không thống nhất giữa các bên. Bên cạnh đó, trí nhớ của con người là có hạn nên không thể ghi nhớ, quản lý tất cả các hoạt động của dự án, đặc biệt khi dự án lớn, có nhiều người tham gia.

Trong ngành công nghệ phần mềm đã nghiên cứu, áp dụng nhiều kỹ thuật tương tác để giải quyết những giới hạn của con người, làm tăng năng suất phát triển phần mềm. Các nhà phát triển đã sử dụng các phương thức như điện thoại, thư điện tử, tin nhắn để tương tác với nhau. Bên cạnh đó các nhà phát triển cũng có thể tương tác với nhau dựa trên các sản phẩm trong quá trình phát triển phần mềm như đặc tả yêu cầu, mô hình UML và mã nguồn.

Có thể chia các công cụ hỗ trợ tương tác trong công nghệ phần mềm thành 4 nhóm chính:

Nhóm công cụ tương tác dựa trên mô hình (model-based collaboration) hỗ trợ tương tác để tạo nên các sản phẩm của dự án như đặc tả yêu cầu, thiết kế hệ thống, kế hoạch kiểm thử. Những sản phẩm này tạo nên mô hình của phần mềm. Các thành viên tương tác dựa trên mô hình của phần mềm.

Nhóm công cụ hỗ trợ tương tác dựa trên quy trình (Process centered collaboration) cho phép mô hình hóa toàn bộ hoặc một phần của quy trình phát triển phần mềm. Các

thành viên tương tác với nhau theo quy trình đã được mô hình hóa. Ví dụ công cụ có thể quản lý việc phân công công việc cho thành viên và theo dõi tiến độ thực hiện công việc.

Nhóm công cụ hỗ trợ nhận thức (Awareness tools) giúp thành viên ý thức được các hành động đang được thực hiện bởi các thành viên khác nhằm tránh xung đột.

Nhóm hạ tầng tương tác được phát triển nhằm tăng khả năng phối hợp giữa các công cụ hỗ trợ tương tác khác. Nhóm hạ tầng tương tác tập trung vào tích hợp dữ liệu và điều khiển giữa các công cụ.

Việc tương tác trong nhóm làm phần mềm thường được hỗ trợ bởi các thành phần riêng lẻ như hệ thống quản lý cấu hình, hệ thống quản lý vấn đề (issue-tracking system), hệ thống tin nhắn tức thời và trang web của dự án. Được kết hợp lại một cách có hệ thống các thành phần trên tạo thành môi trường phát triển tương tác (CDE).

CDE là một không gian ảo nơi các bên liên quan - các bên liên quan có thể có khoảng cách về không gian, thời gian - gặp nhau, chia sẻ, thảo luận, hợp tác cùng nhau để thực hiện công việc nhằm tạo ra sản phẩm. Mục đích của CDE là tạo ra môi trường tốt cho việc phát triển phần mềm bằng cách loại bỏ hoặc tự động hóa các hoạt động bình thường, không sáng tạo của cá nhân và nhóm, cung cấp cơ chế nhằm khuyến khích sự trao đổi giữa các bên liên quan.

Tương tác không phải là chủ đề mới. Cùng với sự phát triển của internet đã có nhiều công cụ hỗ trợ tương tác như thư điện tử, hệ thống tin nhắn tức thời và wiki. Các công cụ trên đã được sử dụng để hỗ trợ tương tác trong nhóm. Tuy nhiên CDE không giống những công cụ trên. Có hai yếu tố tạo nên sự khác biệt của CDE. Thứ nhất bởi vì nhà phát triển phần mềm cần phải theo tác với những sản phẩm mang tính ngữ nghĩa cao. Bên cạnh đó thì mối quan hệ giữa các sản phẩm đó cũng có tính ngữ nghĩa cao. Thứ hai, web giống như là môi trường sống của nhà phát triển. Nó làm cho khoảng cách giữa các nhà phát triển trở nên gần hơn cho dù họ cách xa nhau về mặt địa lý.

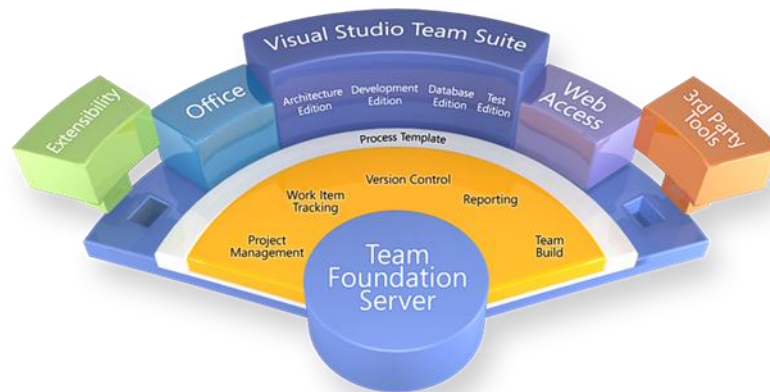
CDE không phải là một hệ thống mới, tuy nhiên nó là một hệ thống dễ thay đổi, bởi vì CDE liên quan đến các yếu tố xã hội của quá trình phát triển phần mềm. CDE cần phải đơn giản, dễ sử dụng, phù hợp với sở thích của cá nhân và văn hóa của nhóm làm phần mềm. CDE cung cấp môi trường cho các cá nhân, nhóm hoạt động. Tuy nhiên nó không được ảnh hưởng đến quá trình làm việc của cá nhân và của nhóm.

3.2 Tìm hiểu thực tế

Hiện nay trên thị trường có một số công cụ hỗ trợ việc tương tác trong nhóm như Team Foundation Server của Microsoft, Assembla. Bên cạnh đó thì cũng có những công cụ mã nguồn mở hỗ trợ việc tương tác như Redmine.

3.2.1 Team foundation server (TFS)

Là một sản phẩm của Microsoft hỗ trợ việc phát triển dự án phần mềm, cho phép các thành viên trong dự án tương tác với nhau



Hình 1 Team Foundation Server Process

(Nguồn: Team Foundation Server Process Templates for effective Project Management, *Aaron Bjork, Kimberly Walters*)

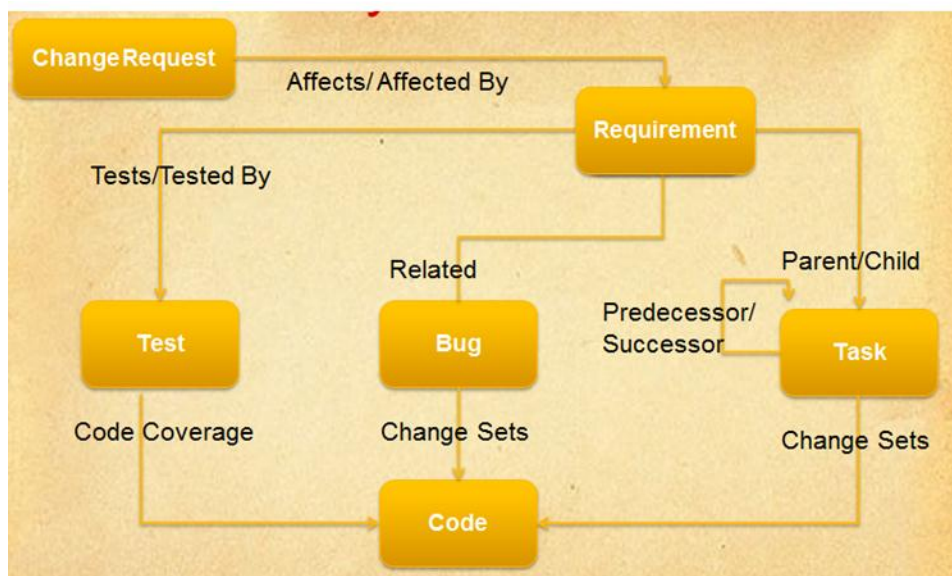
TFS có các 5 nhóm chức năng chính là quản lý dự án, quản lý công việc, quản lý cấu hình, tạo báo cáo, xây dựng nhóm.

Nhóm chức năng *Quản lý dự án* cho phép chọn quy trình phát triển cho dự án. Việc quản lý dự án sẽ phụ thuộc vào quy trình phát triển được chọn. Quy trình phát triển định nghĩa các loại công việc, mối liên hệ giữa các công việc, truy vấn công việc và báo cáo. Hiện tại TFS hỗ trợ sẵn 2 quy trình là CMMI và Agile. Tuy nhiên có thể bổ sung các quy trình khác từ bên thứ ba. Nhóm chức năng *xây dựng nhóm* cho phép tạo và quản lý nhóm làm việc trong dự án.

Tùy thuộc vào quy trình được chọn mà dự án có các loại công việc khác nhau, cũng như mối quan hệ giữa các công việc. TFS có các chức năng cho phép *quản lý công việc*. TFS cho phép tạo công việc, quản lý tình trạng của công việc cũng như quản lý mối quan hệ giữa các công việc.

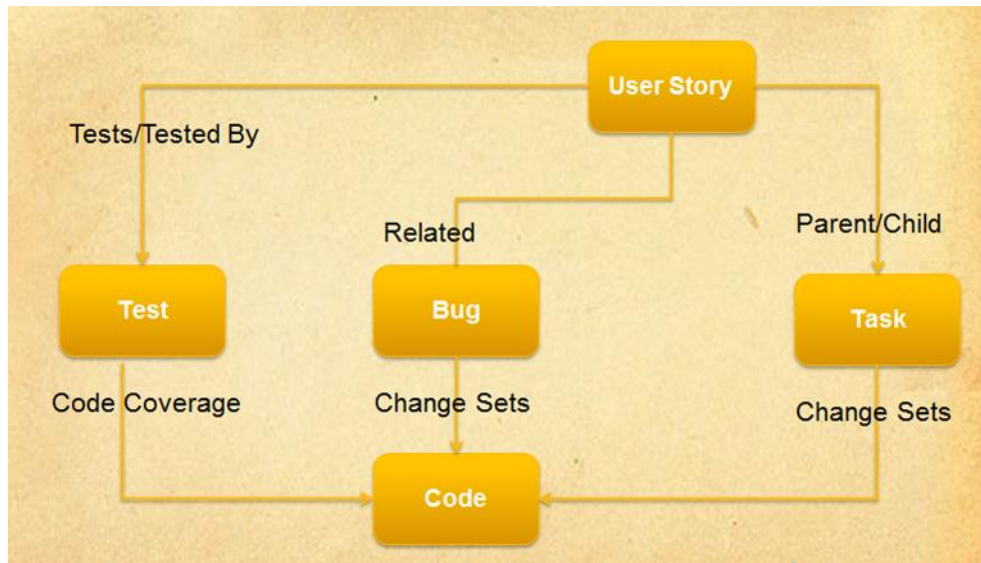
Mã nguồn của dự án được quản lý bởi hệ thống *quản lý phiên bản*. Có thể cấu hình để mã nguồn được tự động biên dịch theo một lịch trình định trước hoặc khi có thay đổi được đưa lên hệ thống quản lý phiên bản.

Tình trạng, tiến độ của dự án có thể theo dõi dễ dàng nhờ vào chức năng *báo cáo*. Mỗi quy trình phát triển sẽ có các loại báo cáo khác nhau để theo dõi tiến độ của dự án.



Hình 2 CMMI

(Nguồn: *Planning and Tracking Projects with VSTS 2010 by Ahmed Nasr*)



Hình 3 Agile

(Nguồn : *Planning and Tracking Projects with VSTS 2010 by Ahmed Nasr*)

Phần mềm Team Foundation Server có một số những điểm ưu việt so với những phần mềm khác như sau:

Ưu điểm đầu tiên là hỗ trợ tốt việc phát triển phần mềm theo quy trình như CMMI và Agile. Những phần mềm phát triển theo những quy trình này sẽ được cung cấp đầy đủ những tính năng hỗ trợ trong quá trình phát triển phần mềm.

Ưu điểm tiếp theo là bao quát tất cả các hoạt động trong quá trình phát triển phần mềm. TFS có thể được sử dụng trong các giai đoạn như lên kế hoạch, thiết kế, cài đặt và kiểm tra. Các sản phẩm được tạo ra trong từng giao đoạn có thể được liên kết với nhau như có thể liên kết bug với mã nguồn và liên kết mã nguồn với yêu cầu.

Và TFS có liên kết với nhiều công cụ thông dụng trong quá trình phát triển phần mềm như Visual Studio Team Suite, MS Office và SharePoint.

Tuy nhiên, Team Foundation Server cũng có những mặt hạn chế đối với người dùng:

Một là không miễn phí và có giá khá cao. Điều này gây khó khăn đối với nhóm người dùng không có điều kiện chi trả cho bản quyền, đặc biệt là đối với sinh viên nước ta.

Hai là quá phức tạp. Để có thể sử dụng một cách thành thạo TFS, sử dụng được hết các chức năng của TFS thì cần phải có một thời gian dài nghiên cứu và thực hành. Ngoài ra, TFS còn có nhiều chức năng mà người dùng, đặc biệt là sinh viên không dùng tới.

Không thích hợp với các dự án nhỏ như các dự án được thực hiện bởi sinh viên trong lớp học. TFS hỗ trợ xây dựng phần mềm theo các quy trình chuẩn, điều này có nghĩa là gồm nhiều giai đoạn mà đối với những dự án nhỏ thì không cần thiết. Trong môi trường lớp học, những dự án nhỏ của sinh viên có thể chỉ thực hiện trong vòng một tuần hoặc một tháng, những quy trình mà TFS đưa ra sẽ không phù hợp với những dự án kiểu này.

3.2.2 Redmine

Redmine là một công cụ quản lý dự án, quản lý vấn đề cho phép tạo nhiều dự án. Vấn đề là một thứ được quan tâm trong dự án. Vấn đề có thể là yêu cầu, bug, công việc, ... Trong mỗi dự án có thể thêm thành viên, phân quyền cho thành viên. Redmine cung cấp nhiều lựa chọn về hệ thống quản lý phiên bản để quản lý mã nguồn. Bên cạnh đó thì redmine cũng hỗ trợ quản lý tài liệu của dự án.

Mỗi dự án có một lịch và biểu đồ grant nhằm thể hiện một cách hình ảnh về tiến độ của dự án cũng như các các thời điểm hạn cuối (deadline) của dự án. Redmine có chức năng quản lý thời gian nhằm theo dõi thời gian các thành viên đã giành cho dự án.

Để các thành viên trong dự án có thể tương tác với nhau dễ dàng hơn, trong dự án có thể tạo trang wiki và diễn đàn để các thành viên tương tác với nhau.

Redmine được viết bằng ngôn ngữ Ruby, sử dụng Ruby on Rails framework. Redmine không phụ thuộc vào nền tảng (cross-platform) và cơ sở dữ liệu. Redmine là một phần của Bitnami app library cho phép triển khai dễ dàng.

Redmine có những điểm mạnh sau:

Ưu điểm thứ nhất là mã nguồn mở và miễn phí. Người dùng có thể tùy chỉnh lại theo mục đích sử dụng của mình. Hơn nữa Redmine miễn phí, cho nên cộng đồng sử dụng redmine cũng rất lớn, tài liệu tham khảo, nghiên cứu về Redmine phong phú và đầy đủ..

Ưu điểm thứ hai là hệ thống quản lý vấn đề mạnh. Đây là thế mạnh chính của redmine, giúp quản lý quy trình phát triển phần mềm một cách hiệu quả.

Ưu điểm thứ ba là hỗ trợ tương tác trong nhóm thông qua wiki và diễn đàn. Nhờ tính năng này, thành viên trong dự án có thể tương tác với nhau dễ dàng hơn.

Ưu điểm thứ tư là hỗ trợ đa ngôn ngữ. Với điểm này, Redmine thu hút được đông đảo người dùng tại nhiều quốc gia, vùng miền khác nhau.

Ưu điểm thứ năm là không phụ thuộc vào nền tảng và cơ sở dữ liệu.

Tuy nhiên, Redmine cũng có những điểm yếu:

Đầu tiên đó là không hỗ trợ quản lý sự kiện của cá nhân và dự án. Người quản lý dự án không thể biết lịch của các cá nhân trong dự án để đưa ra lịch gặp mặt, hội thảo phù hợp.

Tiếp theo là không định nghĩa sẵn các quy trình phát triển. Người dùng phải tự cấu hình hệ thống quản lý vấn đề, thông qua các trường tùy chỉnh (custom fields), và workflow cho phù hợp với quy trình phát triển được sử dụng.

3.2.3 Assembla

Đây là một công cụ rất hữu ích cho ai thường xuyên phải làm việc nhóm. Nó giúp cho quá trình làm việc nhóm trở lên hiệu quả hơn.

Assembla tạo ra một workspace trên mạng chung cho cả nhóm. Đây sẽ là nơi lưu trữ sản phẩm làm việc của nhóm (source code, tài liệu, báo cáo)...). Tất cả các thành viên trong nhóm có thể truy cập vào workspace này để tải về, tải lên, sửa, xóa file...

Đặc biệt assembla được sử dụng cùng với một phần mềm client. Phần mềm này cho phép download và upload một cách dễ dàng và nhanh chóng các file trên workspace, giúp cho tất cả các thành viên trong nhóm có thể biết được về tiến độ làm việc của nhóm một cách và đồng bộ và cập nhật nhất.

Ngoài ra còn các tính năng khác như phân công công việc, chat.

Chương 4 HỆ THỐNG QUẢN LÝ TƯƠNG TÁC

4.1 Chức năng

“Hệ thống quản lý tương tác” thiết kế với những tính năng chính sau:

4.1.1 Quản lý quy trình phát triển phần mềm

Tính năng này giúp người dùng có thể tự định nghĩa quy trình phần mềm để phù hợp với từng dự án đặc thù, từ đơn giản tới phức tạp. Nhờ vậy, khả năng quản lý dự án của hệ thống linh hoạt hơn, không phụ thuộc vào một quy trình phần mềm nhất định nào đó.

4.1.2 Quản lý dự án

Giảng viên sẽ tạo các dự án, phân chia nhóm bằng cách thêm thành viên vào dự án. Thông qua quá trình làm việc với dự án của các thành viên, giảng viên có thể đánh giá quá trình làm việc của sinh viên, qua đó có được đánh giá chính xác hơn cho kết quả cuối cùng của môn học.

4.1.3 Quản lý lớp học

Một lớp học sẽ được dạy một môn học. Mỗi lớp học sẽ có nhiều dự án được thực hiện bởi sinh viên. Giảng viên có thể tạo các lớp học, tạo dự án trong từng lớp học. Quản lý các dự án theo dễ dàng sẽ dễ dàng cho việc quản lý khi mà số lượng dự án ngày càng lớn.

4.1.4 Quản lý lịch cá nhân và lịch dự án

Đây là chức năng mới so với các hệ thống tương tác đã có. Tính năng này giúp người dùng tự tạo lịch cá nhân cho mình, đưa ra thời gian rảnh rỗi, thời gian bận của mình trong ngày, tuần, tháng, năm.

Hệ thống sẽ tổng hợp lịch cá nhân của các thành viên trong dự án để tạo thành lịch dự án. Thông qua lịch dự án, người quản lý sẽ biết được thời gian rảnh rỗi chung của toàn

bộ thành viên, từ đó đưa ra lịch hợp thích hợp. Lịch hợp đưa ra sẽ cập nhật vào lịch cá nhân của từng thành viên trong dự án. Nhờ đó mỗi thành viên cũng sẽ biết được lịch chung của dự án.

4.1.5 Quản lý work item và tương tác trong nhóm

Work item là một yếu tố được quan tâm trong quá trình phát triển phần mềm. Work item có thể là yêu cầu, báo cáo lỗi, công việc. TeamSpace hỗ trợ quản lý work item trong dự án. Bên cạnh đó thành viên cũng có thể tương tác dựa trên work item như thảo luận về work item, nhận thông báo khi có thay đổi trên work item. Chức năng của TeamSpace hỗ trợ quản lý work item và tương tác cụ thể như sau:

Tạo và cập nhật work item

Trong dự án có nhiều loại work item, tùy thuộc vào quy trình phát triển phần mềm được sử dụng. Thành viên sẽ chọn loại work item phù hợp với mục đích của mình để tạo. Khi tạo work item người dùng sẽ nhập tiêu đề, độ ưu tiên, trạng thái của work item. Người dùng cũng có thể mô tả thêm về work item, giao trách nhiệm xử lý work item hiện tại cho một thành viên trong dự án. Tùy thuộc vào loại work item được chọn có thể có thêm các thông tin khác về work item.

Khi trạng thái của work item thay đổi như đã xử lý xong work item, work item tạo không đúng, thành viên có thể thay đổi trạng thái của work item. Bên cạnh đó thì người dùng cũng có thể thay đổi các thông tin khác như tiêu đề, mô tả, độ ưu tiên, người chịu trách nhiệm.

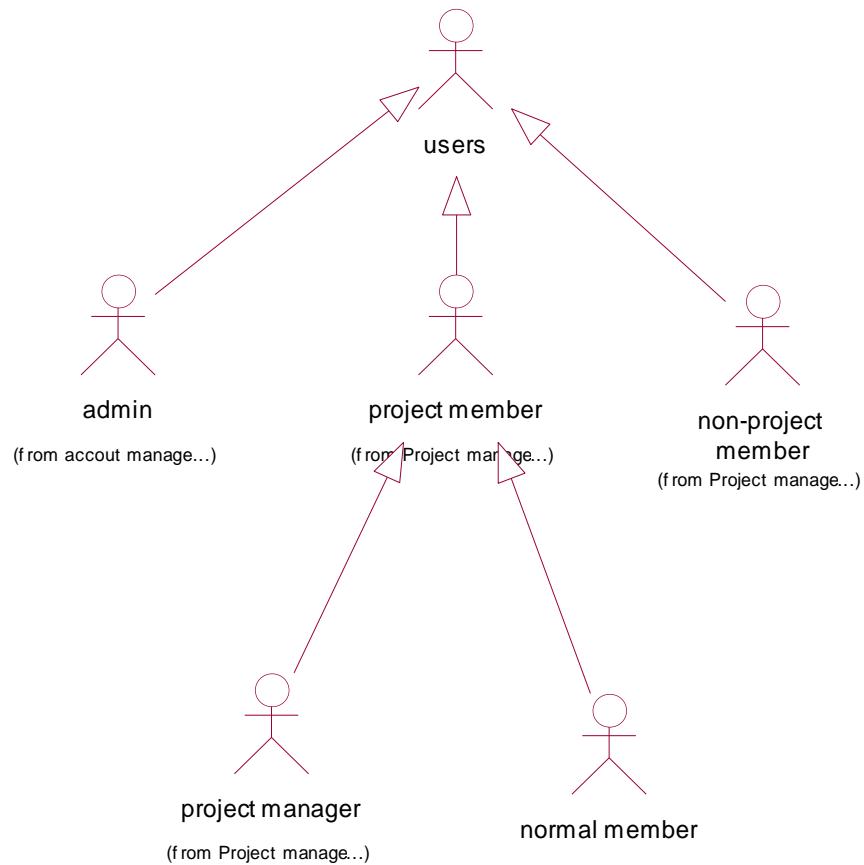
Bình luận về work item và nhận thông báo khi có thay đổi

Thành viên trong nhóm có thể thảo luận về work item hoặc đăng kí (subscribe) để nhận email thông báo khi có thay đổi trên work item.

4.2 Yêu cầu

4.2.1 Yêu cầu chức năng

4.2.1.1 Danh sách các Actor



Hình 4 Danh sách actor

STT	Actor	Mô tả
1	User	Người sử dụng chương trình đã có tài khoản đăng nhập đang ở trạng thái hoạt động bình thường
2	Admin	Quản trị của ứng dụng. Quản trị có quyền quản lý quy trình phát triển, tài khoản, lớp, dự án.
3	Project	Thành viên của nhóm thực hiện đồ án. Thành viên của dự án

	member	có thể thực hiện hầu hết các chức năng liên quan tới dự án như xem và tạo công việc, xem và tạo sự kiện của dự án, ... Tuy nhiên thành viên dự án không thể thay đổi các thiết đặt của dự án.
4	Project manager	Người quản lý dự án. Có thể thực hiện tất cả các chức năng liên quan tới project đang quản lý.
5	Normal member	Người có quyền thao tác trên project nhưng không được phép thay đổi thiết lập của project
6	Non-project member	Người dùng có tài khoản trong hệ thống nhưng không phải là thành viên của dự án hiện tại.

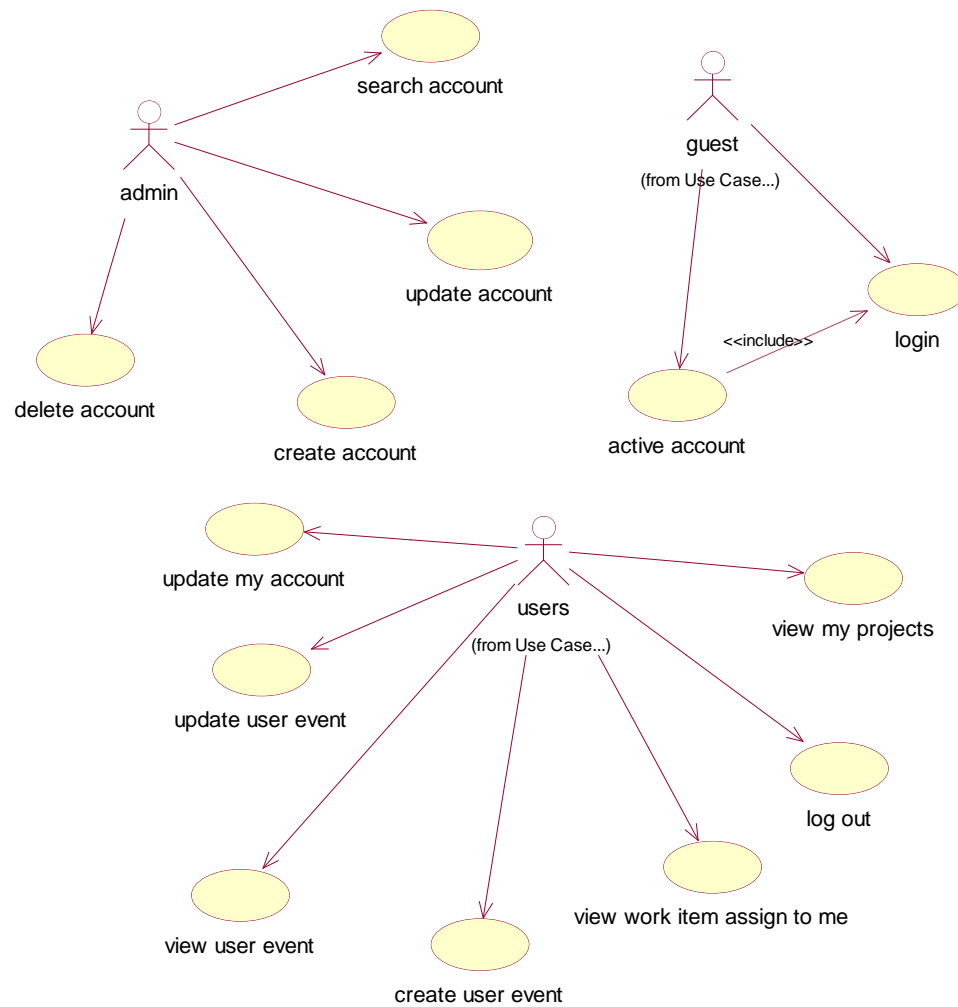
Bảng 1 Danh sách actor

4.2.1.2 Sơ đồ Use-Case

Mô hình Use-Case

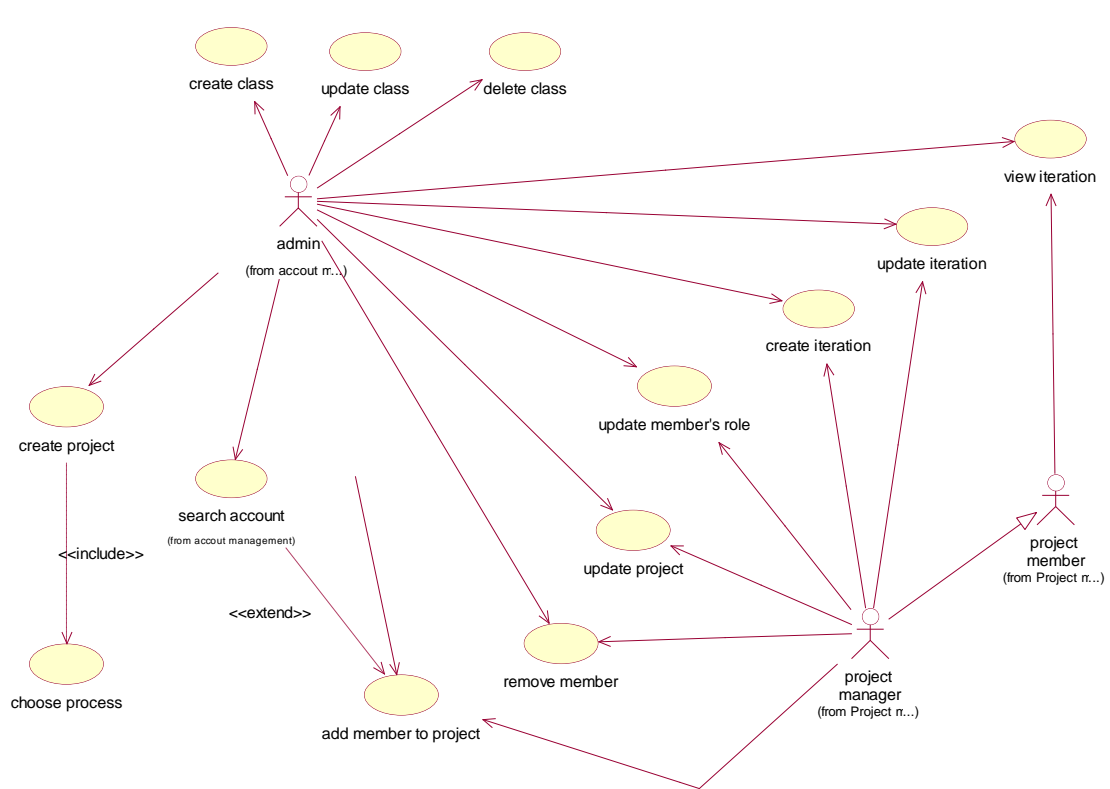
Chia làm 3 nhóm chính là nhóm quản lý tài khoản, nhóm quản dự án, nhóm quản lý quản lý các hoạt động trong dự án. Nhóm *quản lý tài khoản* liên quan tới quản lý tài khoản người dùng và quản lý các thông tin liên quan tới tài khoản. Nhóm *quản lý dự án* liên quan tới việc tạo và quản lý lớp học, tạo và quản lý dự án, quản lý thành viên trong dự án, quản lý các giai đoạn nhỏ trong dự án. Nhóm *quản các hoạt động trong dự án* liên quan tới work item và tương tác trong nhóm.

Account management



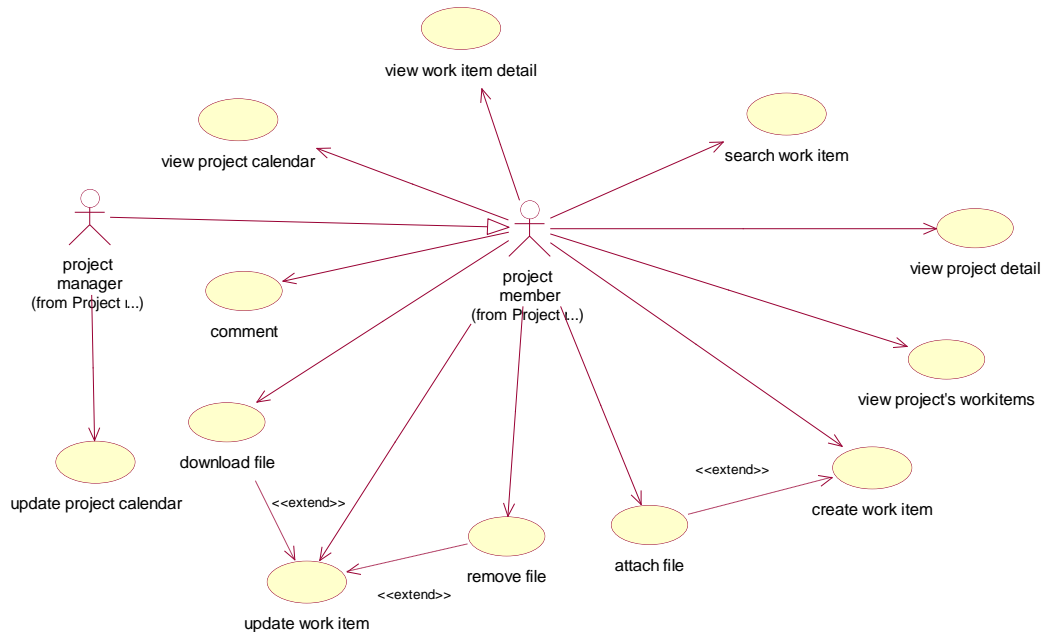
Hình 5 Mô hình use case trong nhóm chức năng quản lý tài khoản

Project management



Hình 6 Mô hình use case trong nhóm chức năng quản lý dự án

Project activity



Hình 7 Mô hình use case trong nhóm chức năng quản lý hoạt động trong dự án

4.2.1.3 Đặc tả Use-Case chính

Use-Case Create Work Item

Tóm tắt	Thành viên dự án tạo work item cho dự án.
Actor chính	Project member
Điều kiện tiên quyết	Người dùng đã đăng nhập và là thành viên của dự án
Điều kiện kết thúc thành công	Hệ thống ghi nhận work item vừa tạo
Điều kiện kết	Khôi phục lại tình trạng hệ thống trước khi tạo work item

thúc tối thiểu	
Trigger	Thành viên chọn chức năng tạo work item
Dòng sự kiện chính	<ol style="list-style-type: none"> 1. Thành viên dự án chọn loại work item cần tạo. 2. Thành viên dự án cung cấp thông tin của work item cần tạo. 3. Hệ thống ghi nhận. 4. Kết thúc
Dòng sự kiện phụ	<p>2a. Thành viên cung cấp thông tin không đúng.</p> <p>2a1. Hệ thống thông báo đã người dùng đã nhập sai thông tin và yêu cầu nhập lại.</p> <p>2a2. Thực hiện lại bước 2.</p> <p>2b. Thành viên muốn hủy bỏ quá trình tạo work item.</p> <p>2b1. Kết thúc use case và không ghi nhập thông tin về work item người dùng đang tạo.</p> <p>2c. Thành viên có thể attach file trong khi tạo work item .</p>

Use-Case Update Work Item

Tóm tắt	Thành viên dự án cập nhật work item của dự án
Actor chính	Project member
Điều kiện tiên quyết	Người dùng đã đăng nhập và là thành viên của dự án
Điều kiện kết thúc thành công	Hệ thống ghi nhận thay đổi
Điều kiện kết thúc tối thiểu	Khôi phục lại tình trạng hệ thống trước khi tạo work item

Trigger	Thành viên chọn chức năng cập nhật work item
Dòng sự kiện chính	<ol style="list-style-type: none"> 1. Thành viên dự án chọn work item cần cập nhật. 2. Thành viên dự án cung cấp thông tin mới của work item 3. Hệ thống ghi nhận. 4. Kết thúc
Dòng sự kiện phụ	<p>2a. Thành viên cung cấp thông tin không đúng.</p> <p>2a1. Hệ thống thông báo đã người dùng đã nhập sai thông tin và yêu cầu nhập lại.</p> <p>2a2. Thực hiện lại bước 2.</p> <p>2b. Thành viên muốn hủy bỏ quá trình cập nhật work item.</p> <p>2b1. Kết thúc use case và không ghi nhập thay đổi trên work item.</p> <p>2c. Thành viên có thể attach file trong khi cập nhật work item.</p> <p>2d. Thành viên có thể attachremove file khi cập nhật work item.</p> <p>2e. Thành viên có thể tải về file đã được đính kèm theo work item trước đó.</p>

Use-Case Create account

Tóm tắt	Admin tạo lập tài khoản mới dựa theo email
Actor chính	Admin
Điều kiện tiên quyết	Đăng nhập với quyền admin
Điều kiện kết thúc thành	Hệ thống lưu trữ vào database

công	
Điều kiện kết thúc tối thiểu	Hệ thống báo lỗi sai định dạng email.
Trigger	Admin chọn chức năng tạo tài khoản.
Dòng sự kiện chính	<ol style="list-style-type: none"> 1. Admin chọn chức năng tạo mới account. 2. Admin nhập email và tạo tài khoản. 3. Hệ thống gửi email kích hoạt tới thành viên, đồng thời lưu vào database. 4. Hệ thống trả về thông tin trạng thái người dùng mới khởi tạo.
Dòng sự kiện phụ	<p>2a. Admin nhập sai định dạng email.</p> <p>2a1. Hệ thống báo sai định dạng.</p> <p>2a2. Thực hiện lại bước 2.</p>

Use-Case Update account

Tóm tắt	Admin thực hiện chức năng update account (thay đổi trạng thái).
Actor chính	Admin
Điều kiện tiên quyết	Đăng nhập với quyền admin
Điều kiện kết thúc thành công	Hệ thống cập nhật những thay đổi
Điều kiện kết thúc tối thiểu	Hệ thống khôi phục lại trạng thái ban đầu
Trigger	Admin chọn chức năng update account
Dòng sự kiện	<ol style="list-style-type: none"> 1. Admin thực hiện chức năng tìm kiếm

chính	2. Admin chọn account và chọn tác vụ update 3. Hệ thống cập nhật thông tin mới về account 4. Hệ thống trả về danh sách account với những cập nhật mới thay đổi
Dòng sự kiện phụ	Không có.

Use-Case Delete account

Tóm tắt	Admin thực hiện chức năng xóa account
Actor chính	Admin
Điều kiện tiên quyết	Đăng nhập với quyền admin.
Điều kiện kết thúc thành công	Hệ thống cập nhật trạng thái của account
Điều kiện kết thúc tối thiểu	Hệ thống khôi phục về trạng thái ban đầu.
Trigger	Admin chọn chức năng delete account.
Dòng sự kiện chính	1. Admin thực hiện chức năng tìm kiếm account. 2. Admin chọn account và chọn tác vụ delete. 3. Hệ thống thay đổi trạng thái của account thành đã xóa, đồng thời xóa email của account trong hệ thống.
Dòng sự kiện phụ	2a. Admin chọn xóa tài khoản của chính mình. 2a1. Hệ thống báo lỗi.

	2a2. Thực hiện lại bước 2.
--	----------------------------

Use-Case Create user's event

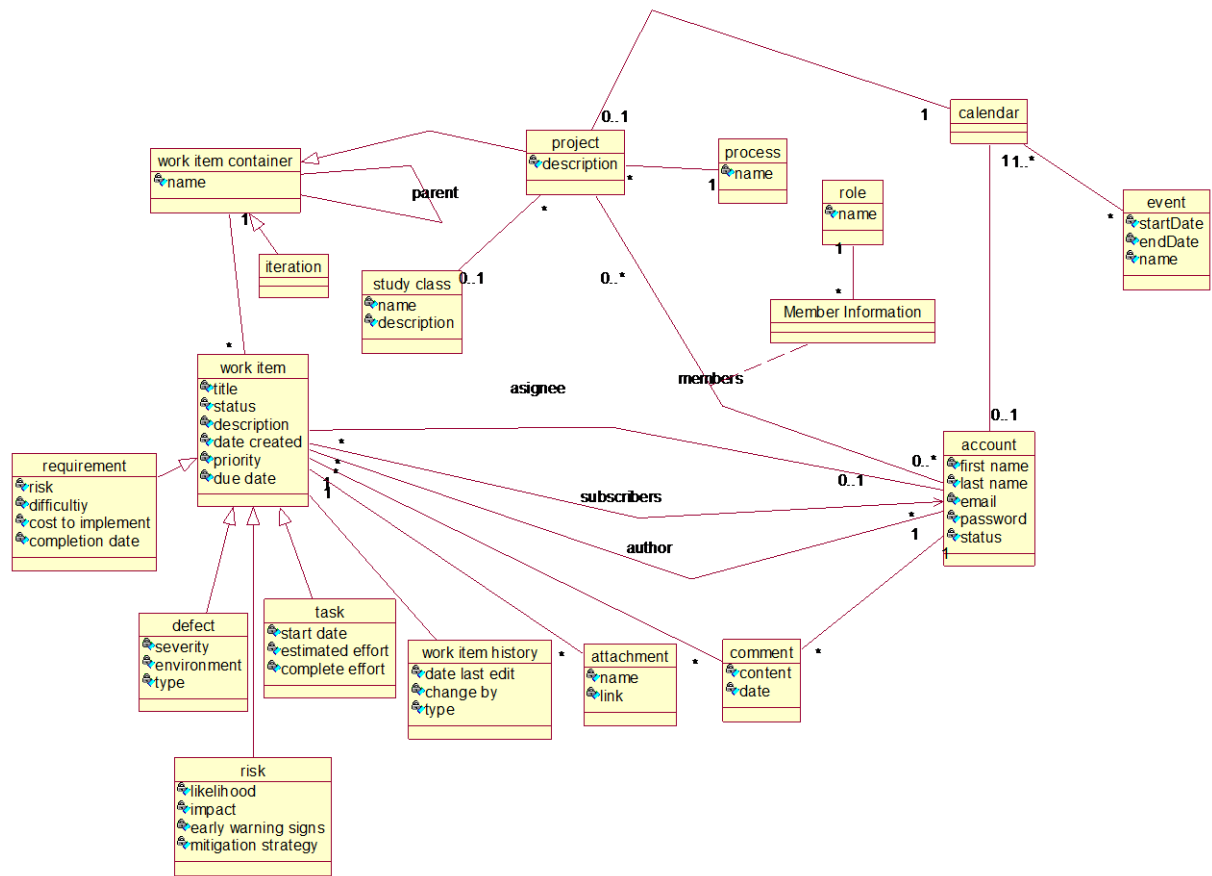
Tóm tắt	User thực hiện chức năng tạo sự kiện cá nhân
Actor chính	User
Điều kiện tiên quyết	User đã đăng nhập.
Điều kiện kết thúc thành công	Hệ thống ghi nhận sự kiện người dùng vừa tạo
Điều kiện kết thúc tối thiểu	Hệ thống khôi phục lại trạng thái trước đó.
Trigger	User chọn chức năng tạo sự kiện cá nhân.
Dòng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng cung cấp thông tin về sự kiện bao gồm tên sự kiện, thời gian bắt đầu, thời gian kết thúc 2. Hệ thống ghi nhận sự kiện
Dòng sự kiện phụ	Không có.

Use-Case Update user's event

Tóm tắt	User chọn chức năng update sự kiện.
Actor chính	User
Điều kiện tiên quyết	User đã đăng nhập.
Điều kiện kết thúc thành công	Hệ thống cập nhật những thay đổi.

công	
Điều kiện kết thúc tối thiểu	Hệ thống khôi phục lại trạng thái trước đó.
Trigger	User chọn chức năng update user's event.
Dòng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn sự kiện cần cập nhật 2. Người dùng cung cấp thông tin mới về sự kiện như tên, thời gian bắt đầu, thời gian kết thúc 3. Hệ thống cập nhật thay đổi
Dòng sự kiện phụ	Không có.

4.2.1.4 Sơ đồ lớp mức phân tích



Hình 8 Sơ đồ lớp mức phân tích

Ghi chú : số loại work item là không cố định. Tùy quy trình phát triển phần mềm sẽ có các loại work item khác nhau.

STT	Lớp đối tượng	Mô tả
1	work item container	Nơi chứa các tác vụ (có thể là iteration hoặc project)
2	iteration	Một vòng lặp nhỏ được tiến hành trong quá trình phát triển dự án.
3	project	Dự án phát triển phần mềm. Trong dự án có nhiều

		thành viên cùng tham gia phát triển.
4	work item	Tác vụ. Là một thứ cần quan tâm, theo dõi trong quá trình thực hiện dự án. Có thể có nhiều loại như là yêu cầu, lỗi, công việc, ...
5	work item history	Lịch sử tác vụ. Lưu lại lịch sử các lần thay đổi trên tác vụ.
6	attachment	Tập tin đính kèm. Ứng với mỗi tác vụ người dùng có thể kèm theo các file để mô tả tác vụ rõ ràng hơn.
7	comment	Bình luận. Tương ứng với bình luận của người dùng đối với tác vụ.
8	account	Tài khoản của người dùng trong hệ thống.
9	process	Quy trình phát triển phần mềm. Quy trình phát triển sẽ quy định các loại tác vụ có thể tạo trong dự án.
10	study class	Lớp học. Trong một lớp học có thể có nhiều dự án.
11	calendar	Lịch. Mỗi người dùng và dự án có một lịch nhằm hỗ trợ lên lịch cho sự kiện của cá nhân hay dự án.
12	role	Vai trò của thành viên trong dự án.
13	member information	Tương ứng với thành viên của dự án.
14	event	Sự kiện của cá nhân, dự án.
15	task	Công việc được phân công cho các thành viên trong dự án.
16	defect	Lỗi. Đây là các lỗi được phát hiện trong quá trình phát triển phần mềm. Có thể là lỗi trong thiết kế, lỗi trong mã nguồn, ...
17	risk	Rủi ro. Các nguy cơ mà dự án có thể gặp phải trong

		quá trong quá trình thực hiện.
18	requirement	Yêu cầu đối với phần mềm đang được thực hiện. Có thể là yêu cầu chức năng hoặc phi chức năng.

Bảng 2 Các lớp đối tượng mức phân tích

4.2.2 Yêu cầu phi chức năng

Hệ thống sử dụng hệ quản trị cơ sở dữ liệu MySQL, sử dụng web server Apache Tomcat.

Hệ thống hiển thị tốt trên trình duyệt IE8+, FireFox 12+, Chrome 19+.

Hệ thống cần phải không phụ thuộc vào hệ quản trị cơ sở dữ liệu cũng như hệ điều hành.

Hệ thống có giao diện tiện dụng. Đối với người dùng có hiểu biết về máy tính, cụ thể là sinh viên công nghệ thông tin thì *thời gian học* để có thể sử dụng các chức năng cơ bản của hệ thống nhỏ hơn 15 phút.

Hệ thống có tính bảo mật tốt. Người dùng không thể thay đổi những thông tin của người khác. Thành viên dự án nếu không phải project manager thì không được thay đổi thiết lập của dự án.

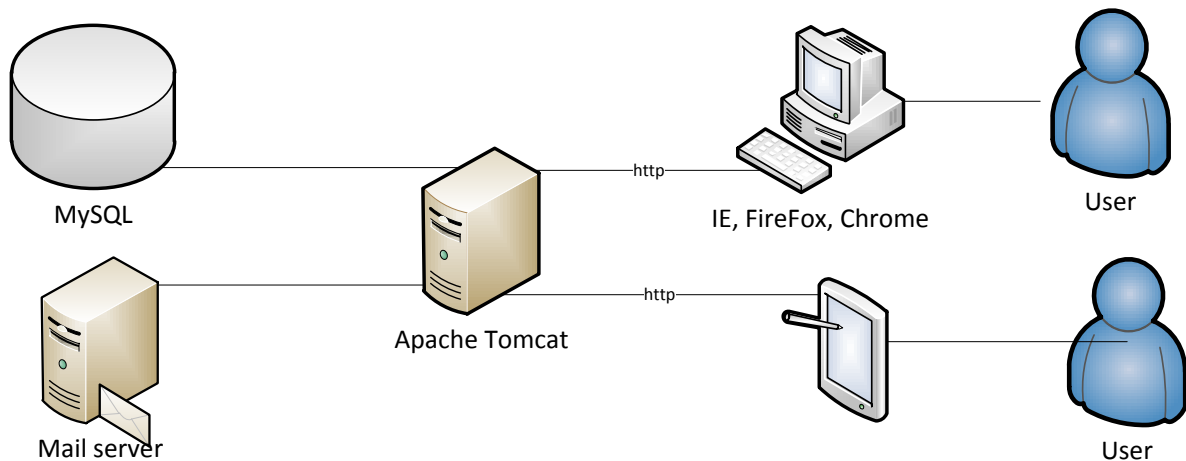
Hệ thống cần có khả năng xử lý nhanh, đưa kết quả trả về người dùng trong khoảng thời gian có thể chấp nhận được.

4.3 Thiết kế

Hệ thống được viết bằng ngôn ngữ *Java*, sử dụng web server *Apache Tomcat*. Dữ liệu được lưu trữ trong hệ quản trị cơ sở dữ liệu *MySQL*. Máy client gửi và nhận dữ liệu với server dùng kỹ thuật *Ajax*. Tại máy client sử dụng javascript framework JQuery, dojo, SmartClient nhằm để hiển thị các đối tượng giao diện phức tạp và hỗ trợ việc gửi và nhận dữ liệu giữa client và server dùng Ajax.

4.3.1 Mô hình triển khai

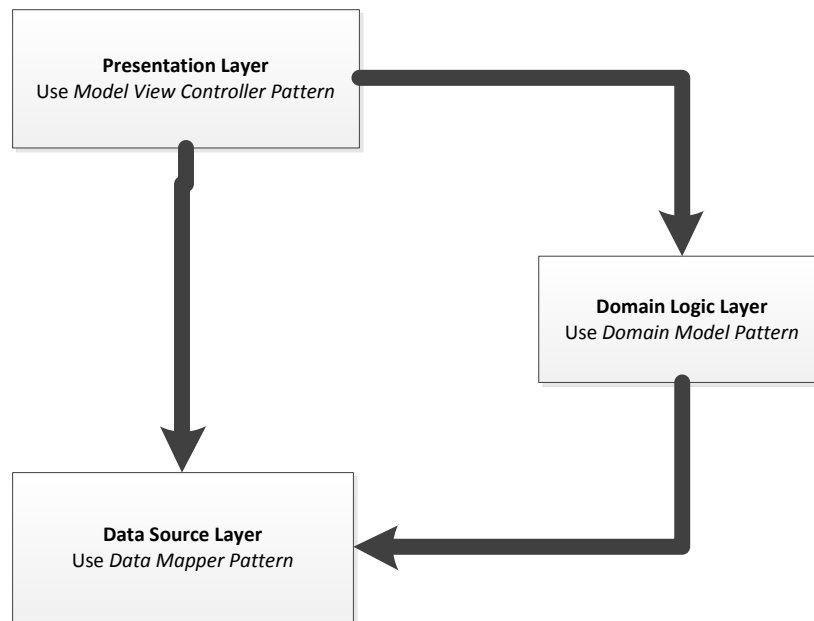
Người dùng có thể truy cập vào ứng dụng từ nhiều thiết bị khác nhau như máy tính, máy tính bảng và smartphone bằng trình duyệt. Hệ thống sẽ xử lý các yêu cầu từ người dùng, truy cập vào cơ sở dữ liệu MySQL để lấy và lưu dữ liệu, gửi mail thông qua Mail Server.



Hình 9 Mô hình triển khai

4.3.2 Kiến trúc tổng quan

Hệ thống được chia thành 3 tầng chính là presentation, domain logic và data source. Trong đó tầng presentation sẽ xử lý các lệnh từ người dùng, gọi tới tầng data source để lấy các thông tin liên quan và yêu cầu tầng domain logic xử lý dữ liệu đó trước khi thể hiện cho người dùng.



Hình 10 Kiến trúc tổng quan

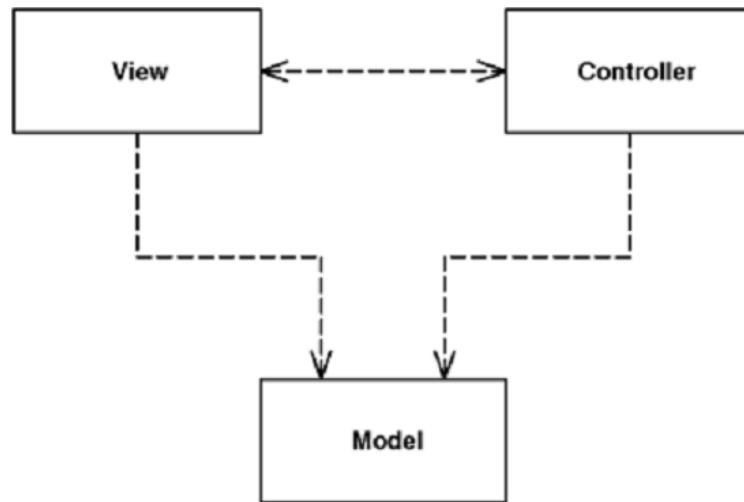
Presentation Layer

Tầng xử lý tương tác giữa người dùng với hệ thống. Trong hệ thống TeamSpace tầng này được cài đặt theo mô hình *Model View Controller* (MVC), gồm 3 thành phần chính là model, view, controller.

Trong đó thành phần *model* là đối tượng thể hiện thông tin của lĩnh vực. Các đối tượng này chứa thông tin và hành động được sử dụng bởi giao diện người dùng. Model là đối tượng trong *domain model*.

View là thành phần thể hiện model dưới dạng giao diện người dùng. View chỉ làm nhiệm vụ hiển thị thông tin việc xử lý thay đổi trên thông tin được thực hiện bởi thành phần thứ 3 của mô hình này controller.

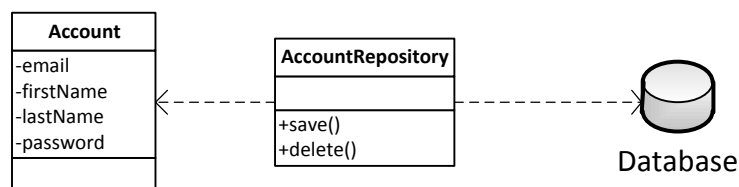
Controller nhận dữ liệu từ người dùng, thay đổi model và yêu cầu view cập nhật lại thay đổi.



Hình 11 Mô hình MVC

Data Source Layer

Xử lý tương tác giữa hệ thống với các hệ thống khác cụ thể là hệ quản trị cơ sở dữ liệu MySQL. Trong hệ thống đang phát triển tầng này được cài đặt theo mẫu *Data Mapper* (Data Mapper pattern). Data Mapper là một tầng nằm giữa các đối tượng trong bộ nhớ và cơ sở dữ liệu. Tầng này làm nhiệm vụ trao đổi dữ liệu và tách biệt giữa các đối tượng trong bộ nhớ và cơ sở dữ liệu.



Hình 12 Data Mapper Pattern

Domain Logic Layer

Tầng thực hiện các xử lý logic của lĩnh vực (domain) như thực hiện tính toán dựa trên dữ liệu nhập vào dữ liệu đã được lưu trữ trước đó. Được cài đặt theo mẫu *Domain Model* (Domain Model Pattern). Từ nghiệp vụ của người dùng mô hình hóa

thành các đối tượng nghiệp vụ. Các đối tượng này bao gồm cả dữ liệu cũng như quy định của nghiệp vụ. Các đối tượng này tạo nên domain model.

4.3.2.1 Các package chính trong hệ thống

org.hcmus.tis.controller:

Package này chứa các lớp xử lý tương tác giữa người dùng với hệ thống. Package này cài đặt thành phần controller của tầng presentation.

org.hcmus.tis.model:

Chứa các lớp đặc thù của lĩnh vực. Đối tượng của các lớp thuộc package này tạo nên domain model của hệ thống. Các đối tượng này thực hiện việc xử lý logic của lĩnh vực. Package này cài đặt tầng domain model, đồng thời cũng là thành phần model của tầng presentation.

org.hcmus.tis.repository:

Chứa các lớp làm nhiệm vụ trao đổi dữ liệu giữa chương trình và cơ sở dữ liệu. Package này cài đặt tầng data source.

org.hcmus.tis.dto:

Chứa các lớp làm nhiệm vụ bao gói dữ liệu để trả về client. Các đối tượng trong package này sẽ trả về client dưới định dạng json.

org.hcmus.tis.util:

Chứa các lớp hỗ trợ như hỗ trợ gửi mail và hỗ trợ đọc file.

4.3.3 Các lớp chính trong domain model

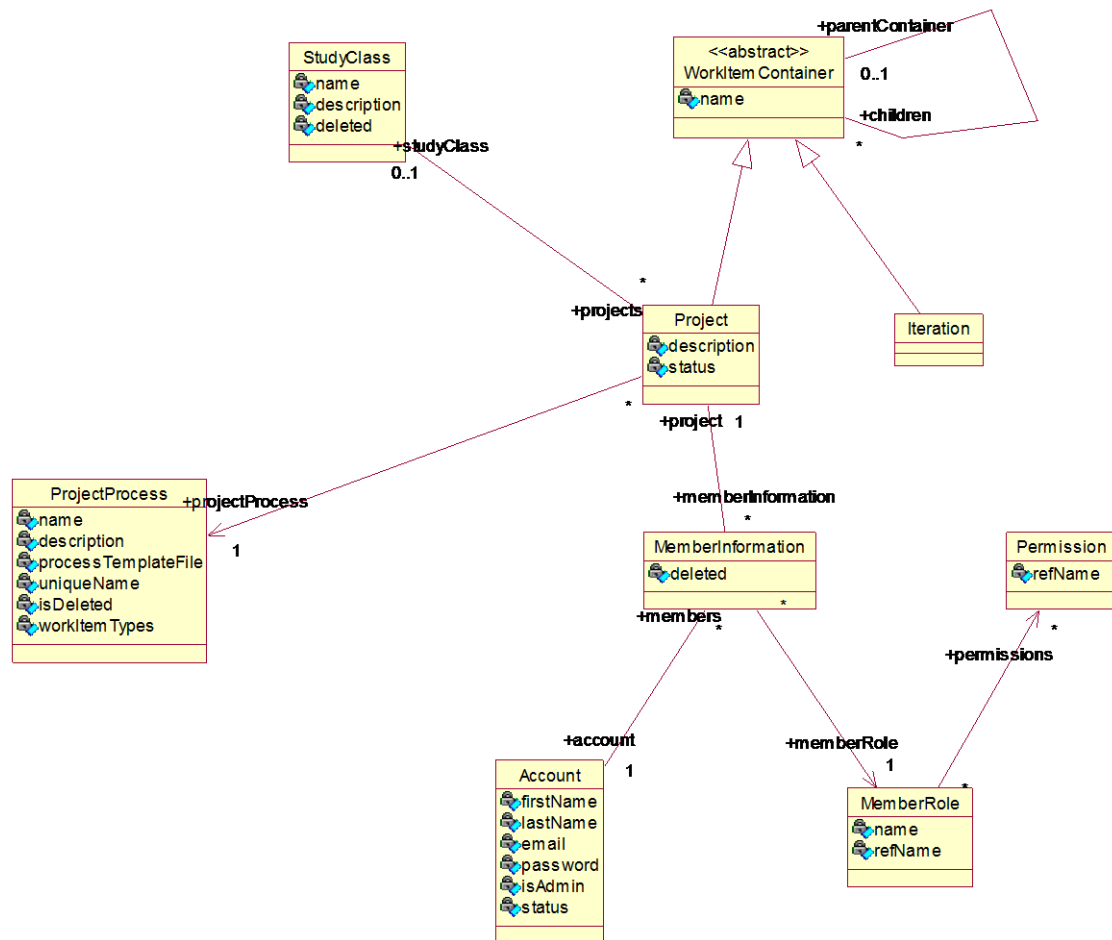
Domain model bao gồm các đối tượng được mô hình hóa từ nghiệp vụ thực tế. Các đối tượng này bao gồm cả dữ liệu cũng như quy định của nghiệp vụ thực tế. Các lớp đối tượng trong domain model được chia thành 3 nhóm chính. Nhóm *quản lý nhóm và dự*

án bao gồm các lớp đối tượng liên quan tới nghiệp vụ quản lý dự án phần mềm và nhóm làm phần mềm. Nhóm *quản lý work item* và *tương tác* liên quan tới nghiệp vụ quản lý work item. Work item là đối tượng được quan tâm trong dự án. Tùy theo dự án có thể có nhiều loại work item khác nhau như bug và user story. Bên cạnh đó thì nhóm này cũng thực hiện các xử lý logic liên quan tới việc tương tác giữa các thành viên trong nhóm. Nhóm *quản lý lịch* liên quan tới việc quản lý sự kiện của từng cá nhân cũng như của cả dự án.

Các lớp trong Domain model đều có thuộc tính id và version để có thể lưu vào cơ sở dữ liệu và hỗ trợ truy xuất đồng thời. Tuy nhiên để dễ theo dõi trong các mô hình những thuộc tính này sẽ không được đề cập tới.

4.3.3.1 Quản lý nhóm và dự án:

Mỗi dự án cần phải tham chiếu tới một quy trình phát triển phần mềm. Tùy thuộc vào quy trình mà dự án sẽ có các loại work item khác nhau. Dự án có thể thuộc về một lớp học. Trong dự án có thể có các dự án con, hoặc là các giai đoạn nhỏ hơn. Tương ứng với thành viên của dự án ta có các đối tượng thuộc lớp *MemberInformation*. Các đối tượng này có chứa tham chiếu tới dự án đang tham gia *project*, người dùng tương ứng *account* và vai trò trong dự án *role*. Trong từng role sẽ có danh sách các quyền *permissions* của role. Hệ thống sẽ kiểm tra việc truy cập của người dùng dựa trên quyền này.



Hình 13 Các lớp chính liên quan tới việc quản lý nhóm và dự án

4.3.3.2 Quản lý work item và tương tác trong nhóm:

Hệ thống hỗ trợ nhiều loại work item. Tất cả các work item đều có các thuộc tính chung như title, description, status và priority. Bên cạnh đó thì mỗi loại work item còn có các thuộc tính riêng. Ví dụ Requirement có thêm thuộc tính risk, difficulty, costToImplement. Risk có thêm thuộc tính likelihood, impact.

Để giải quyết vấn đề trên, mỗi thuộc tính chung của work item sẽ tương ứng với một thuộc tính trong lớp WorkItem. Tất cả các thuộc tính riêng của work item sẽ được

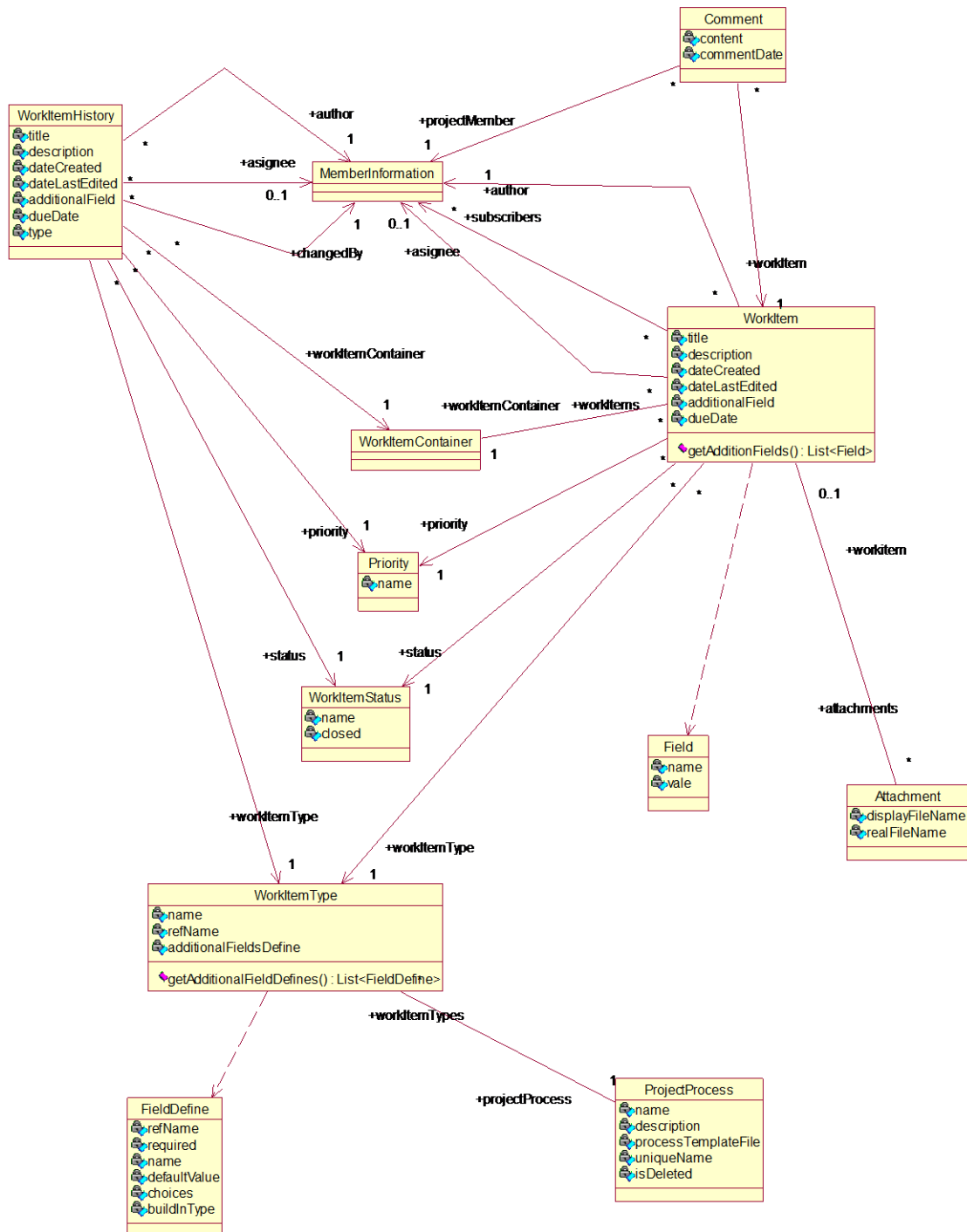
chuyển thành một chuỗi xml và lưu trong thuộc tính *additionalField* của lớp *WorkItem*. Lớp *WorkItem* có phương thức *getAdditionalFields* thực hiện việc phân tích chuỗi này và trả về danh sách các thuộc tính riêng của work item.

```
<xAdditionalFields xmlns="http://www.w3schools.com">
  <xField ref="severity">High</xField>
  <xField ref="environment">windows 7</xField>
  <xField ref="type"></xField>
</xAdditionalFields>
```

Trong ví dụ trên là giá trị trường *additionalField* của work item. Work item đó ngoài các thuộc tính chung còn có thêm thuộc tính *severity*, *environment*, *type* với giá trị lần lượt là *High*, *window 7* và *null*.

Thuộc tính *additionalField* của lớp *WorkItem* chỉ lưu giá trị của các thuộc tính riêng, còn các ràng buộc trên thuộc tính riêng như kiểu giá trị, có được phép bằng *null* hay không, giá trị mặc định được định nghĩa trong file mô tả quy trình phát triển. Ví dụ đoạn file xml mô tả loại work item defect

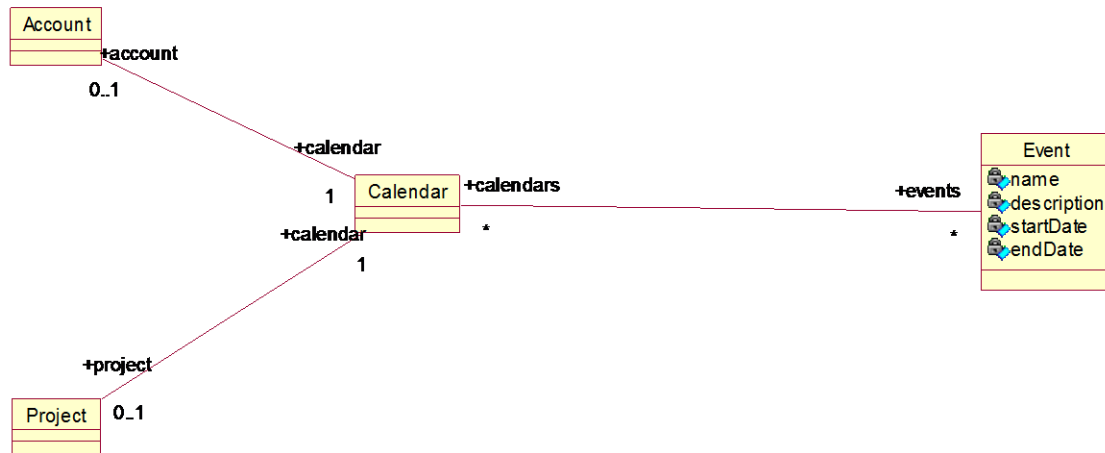
```
<xWorkItem name="Defect" refName="defect">
  <xPreDefinedFields></xPreDefinedFields>
  <xAdditionalFields>
    <xField name="Severity" type="string" refName="severity">
      <xChoices>
        <xChoice>High</xChoice>
        <xChoice>Low</xChoice>
      </xChoices>
    </xField>
    <xField name="Environment" type="string" refName="environment">
      <xDefaultValue>fas</xDefaultValue>
    </xField>
    <xField name="type" type="string" refName="type"></xField>
  </xAdditionalFields>
</xWorkItem>
```

Hình 14 Các lớp chính liên quan tới quản lý work item và tương tác

4.3.3.3 Quản lý lịch

Mỗi account và project đều có một calendar. Trong calendar sẽ có nhiều event, đồng thời một event cũng có thể thuộc nhiều calendar.

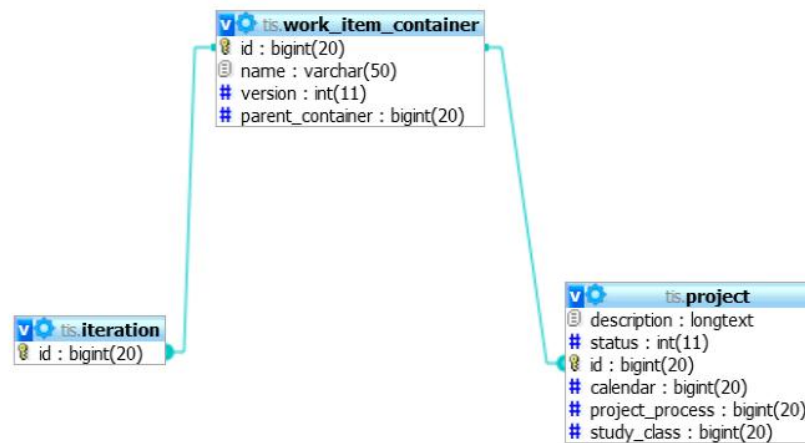


Hình 15 Các lớp chính liên quan tới quản lý lịch

4.3.4 Cơ sở dữ liệu

Hệ thống có cơ sở dữ liệu tương đối giống domain model. Mỗi một lớp trong domain model tương ứng với một bảng trong cơ sở dữ liệu.

Trong domain model, Project và Iteration kế thừa từ WorkItemContainer. Trong khi đó, cơ sở dữ liệu quan hệ không hỗ trợ kế thừa. Vấn đề được giải quyết bằng cách sử dụng cách tiếp cận “**Class Table Inheritance**”. Với cách tiếp cận này sẽ tạo ra một bảng tương ứng với một lớp trong cây kế thừa. Trong bảng của lớp con có khóa ngoại trỏ đến bảng của lớp cha nhằm liên kết các dòng có quan hệ lại với nhau.

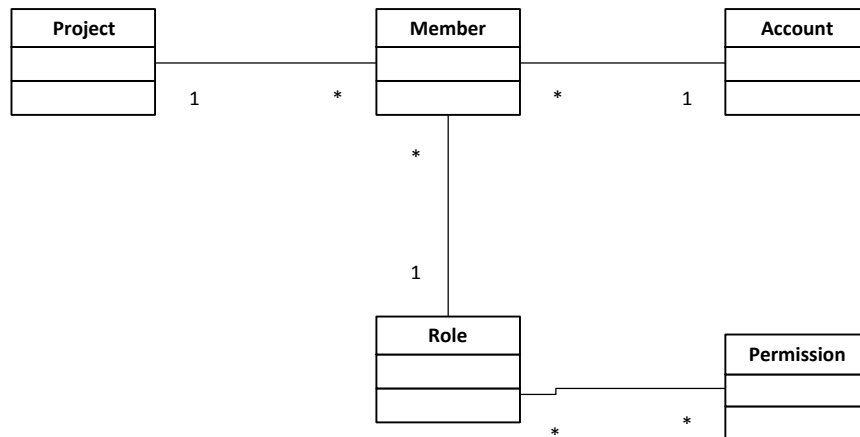


Hình 16 Class Table Inheritance

4.3.5 Cơ chế bảo mật

Hệ thống được bảo mật thông qua cơ chế kiểm soát truy cập dựa trên vai trò (role based access control) . Mỗi người dùng sẽ có các vai trò khác nhau tùy thuộc vào dự án. Mỗi vai trò sẽ được gán quyền (permission). Để thực hiện được hành động trên hệ thống thì người dùng cần có quyền tương ứng với hành động đó đó. Ví dụ để có thể xem được thông tin của dự án, người dùng cần có quyền “project : read”.

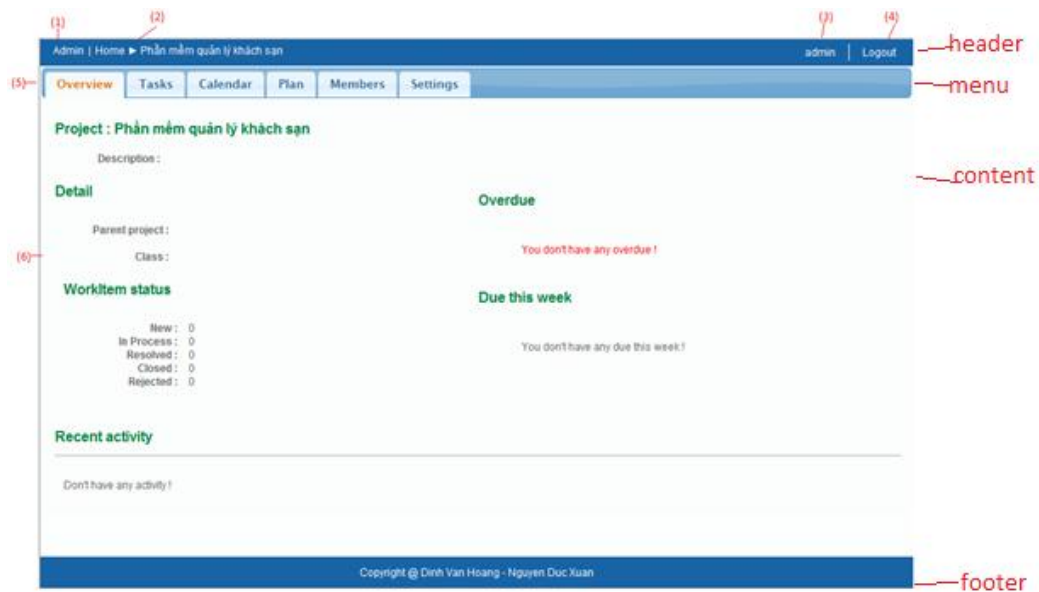
Quyền của người dùng trong hệ thống có dạng “domain : operation”. Trong đó domain là lĩnh vực thao tác được thực hiện. Ví dụ “project : list” thì người dùng sẽ có quyền xem danh sách các project trong hệ thống”. Bên cạnh đó để đảm bảo người dùng bình thường chỉ có thể cập nhật được tài khoản của mình hệ thống có kiểm tra quyền truy ở cấp độ một thể hiện cụ thể, lúc này quyền người dùng có dạng “domain : operation : instance”. Ví dụ người dùng có id là 1 thì sẽ có quyền là “account : update : 1”.



Hình 17 Mô hình vai trò của người dùng trong dự án

4.3.6 Thiết kế giao diện

4.3.6.1 Cấu trúc chung của giao diện



Hình 18 cấu trúc chung của giao diện

(1)Liên kết tới trang quản lý của admin, (2) Liên kết tới trang home,

(3)Tên của người dùng hiện tại, (4) Liên kết cho phép đăng xuất,

(5) Các tab chức năng, (6) Nội dung chính cần hiển thị

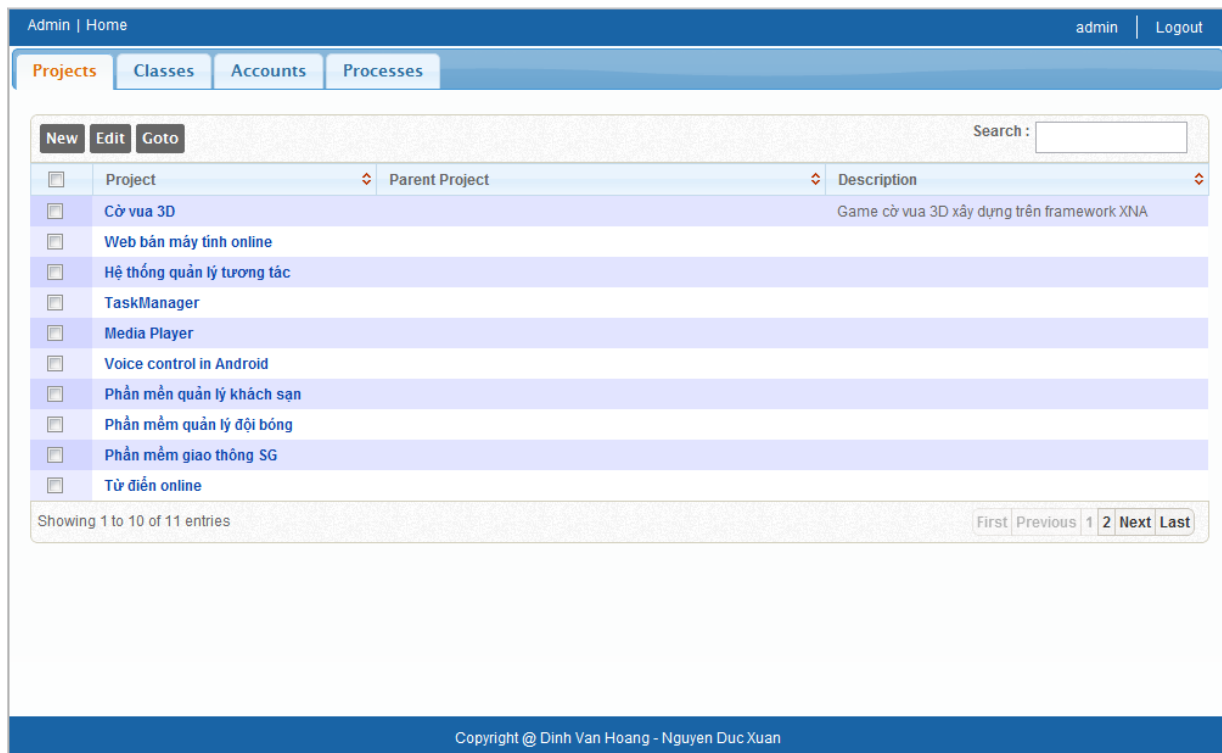
Cấu trúc của giao diện bao gồm 4 thành phần chính là phần header, phần menu, phần content và phần footer.

Phần *header* có link cho phép người dùng đăng nhập, đăng xuất và hiển thị tên người dùng đang đăng nhập. Đồng thời cũng cho biết vị trí hiện tại của người dùng trong website và có liên kết cho phép người dùng chuyển đổi giữa trang nhà của người dùng, trang quản lý của quản trị nếu người dùng có quyền quản trị.

Phần *menu* chứa các tab chức năng. Tùy thuộc vào vị trí hiện tại của người dùng trong website mà các tab chức năng này sẽ khác nhau. Ví dụ khi người dùng đang ở trang quản lý của admin thì sẽ có các tab chức năng là projects, classes, accounts, proceses. Khi người dùng ở trang của project thì sẽ có các tab chức năng là overview, tasks, members, calendar và settings.

Phần *content* chứa nội dung chính cần hiển thị. Phần *footer* chứa các thông tin chung của hệ thống như tên hệ thống, phiên bản...

4.3.6.2 Giao diện khi người dùng ở trang quản lý của admin



Hình 19 Giao diện admin

Trong giao diện trang quản lý của admin phần menu có 4 tab.

- **Projects:** các chức năng liên quan tới quản lý dự án, bao gồm tạo mới, chỉnh sửa thông tin và chuyển đổi trạng thái của project.
- **Classes:** chức năng liên quan tới quản lý lớp học bao gồm tạo mới, chỉnh sửa, cập nhật thông tin và xóa lớp học.
- **Accounts:** các chức năng liên quan tới quản lý tài khoản như thêm, cập nhật và xóa tài khoản.
- **Processes:** các chức năng liên quan tới quản lý các quy trình phát triển phần mềm như thêm process (upload file XML mô tả process template).

4.3.6.3 Giao diện khi người dùng ở trang của dự án

Hình 20 Giao diện project

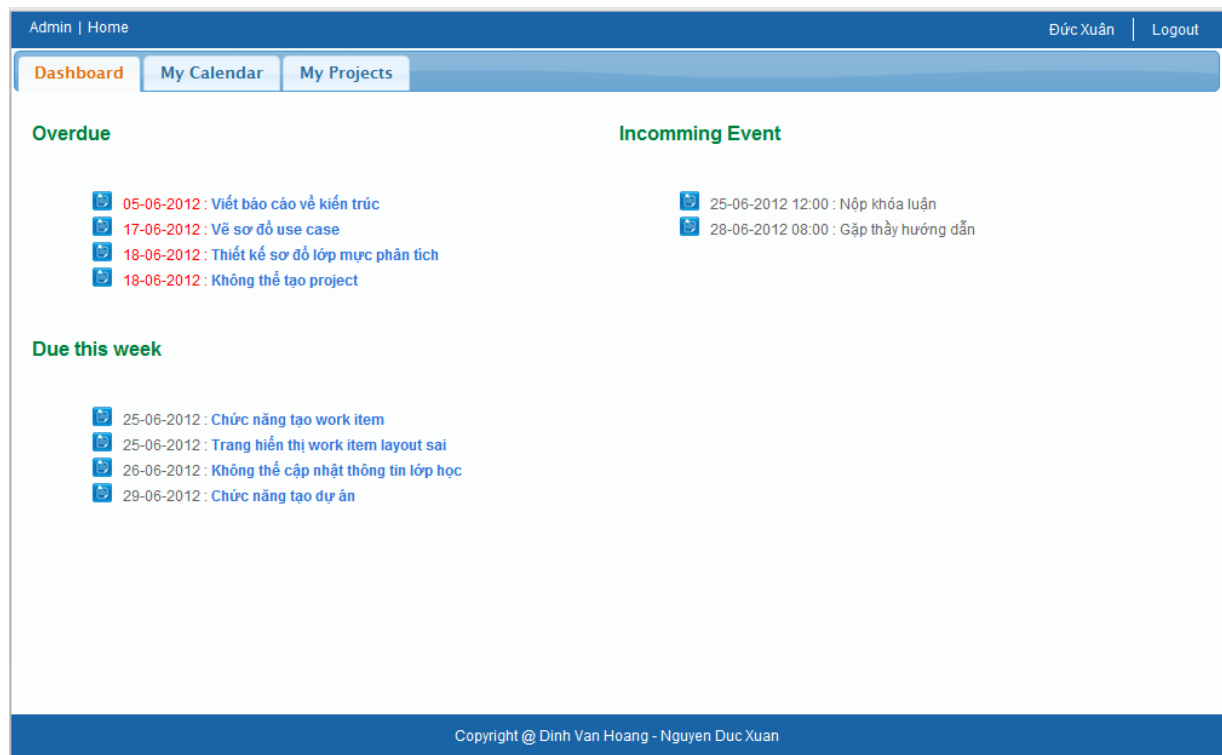
(1) Tên project hiện thời

Trong giao diện trang của project, phần menu có 6 tab:

- **Overview:** thông tin chung về project, các tác vụ đã quá hạn, tác vụ gần tới hạn cuối, hoạt động gần đây trong dự án.
- **Tasks:** quản lý tác vụ. Tại đây, người dùng có thể thêm mới work item, phân chia công việc cho các thành viên dự án cũng như theo dõi tiến độ hoàn thành của dự án.
- **Calendar:** quản lịch của project. Người dùng sẽ xem lịch làm việc tổng hợp của tất cả các thành viên, đồng thời cũng có khả năng tạo mới lịch chung cho dự án.

- **Plan:** quản lý các giai đoạn nhỏ hơn trong dự án, thường là các vòng lặp phát triển trong quy trình phần mềm như Requirement, Analysis and Design, Implement, Configuration and Change Management, Document, Deployment và Maintain.
- **Members:** quản lý thành viên của dự án. Tại tab này, Project manager sẽ thêm thành viên vào dự án hoặc xóa bớt thành viên ra khỏi dự án.
- **Settings:** thiết đặt các thông tin của dự án, bao gồm các thông tin như tên, mô tả, dự án lớp trên và lớp học.

4.3.6.4 Giao diện nhà người dùng



Hình 21 Giao diện nhà người dùng

Trong giao diện trang nhà của người dùng phần menu có 3 tab.

- **Dashboad:** hiển thị thông tin về những hoạt động cần làm của người dùng.

- **My Calendar:** lịch cá nhân của người dùng. Người dùng sẽ thêm, xóa, sửa lịch cá nhân của mình tại tab này, đồng thời cũng có thể thấy được lịch làm việc chung của dự án được hiển thị khác màu với lịch cá nhân của mình.

My Projects: danh sách các project mà người dùng tham gia.

4.3.6.5 Màn hình overview của dự án

Xem chi tiết trong *Hình 20*. Trang overview của dự án sẽ hiển thị các thông tin về dự án như tên, mô tả, lớp học, dự án cha. Trang này cũng hiển thị các thông tin về trạng thái của dự án như hiển thị thống kê về tình trạng của work item trong dự án, danh sách work item đã quá hạn, work item sắp hết hạn. Đồng thời trang overview cũng hiển thị các hoạt động gần đây của dự án.

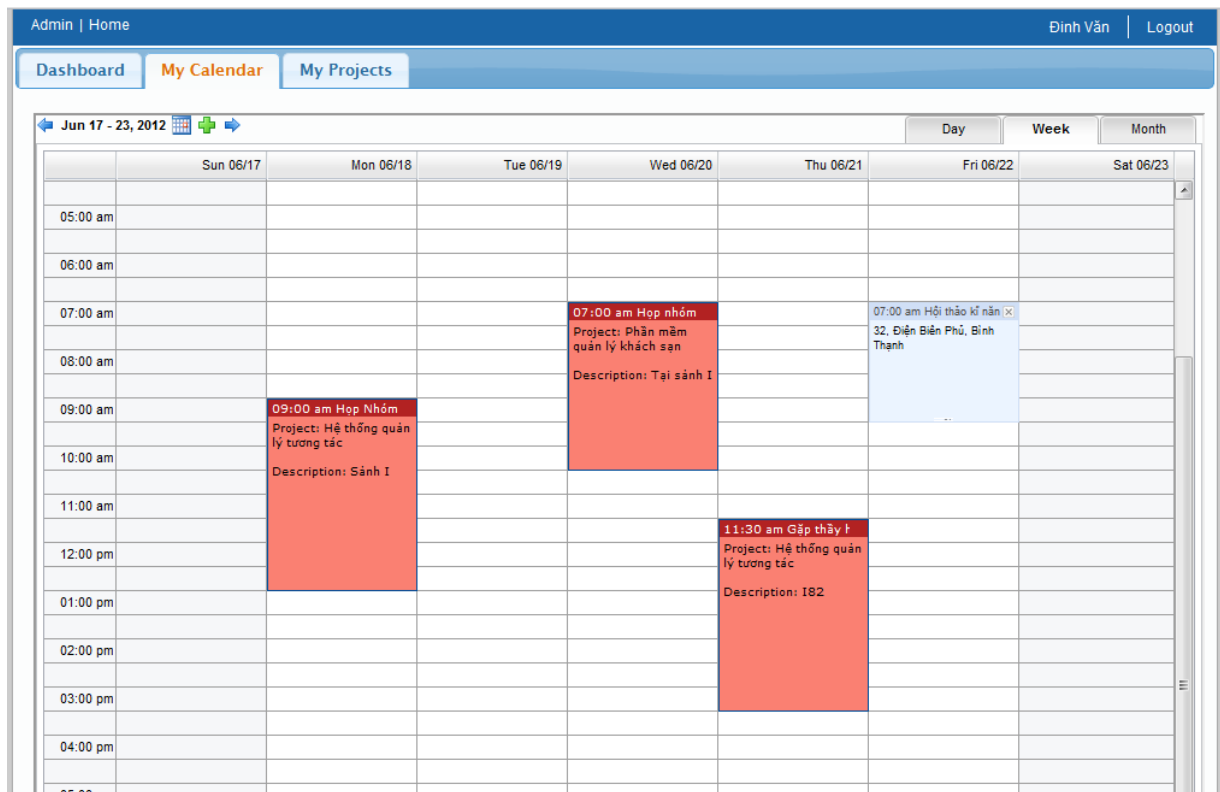
4.3.6.6 Màn hình dashboard của người dùng:

Xem *hình 21*. Trong màn hình này sẽ hiển thị các work item được giao cho người dùng. Các work item được chia thành 2 nhóm đã quá hạn và sắp hết hạn. Màn hình này cũng hiển thị các sự kiện sắp diễn ra của người dùng.

4.3.6.7 Màn hình account calendar:

Trong giao diện này sẽ hiển thị tất cả các sự kiện có liên quan tới người dùng hiện tại. Trong đó những sự kiện cá nhân của người dùng thì có thể chỉnh sửa và có màu xanh. Những sự kiện của dự án thì người dùng không thể chỉnh sửa và hiển thị màu đỏ. Giao diện có thể hiển thị theo tuần, theo ngày hoặc theo tháng.

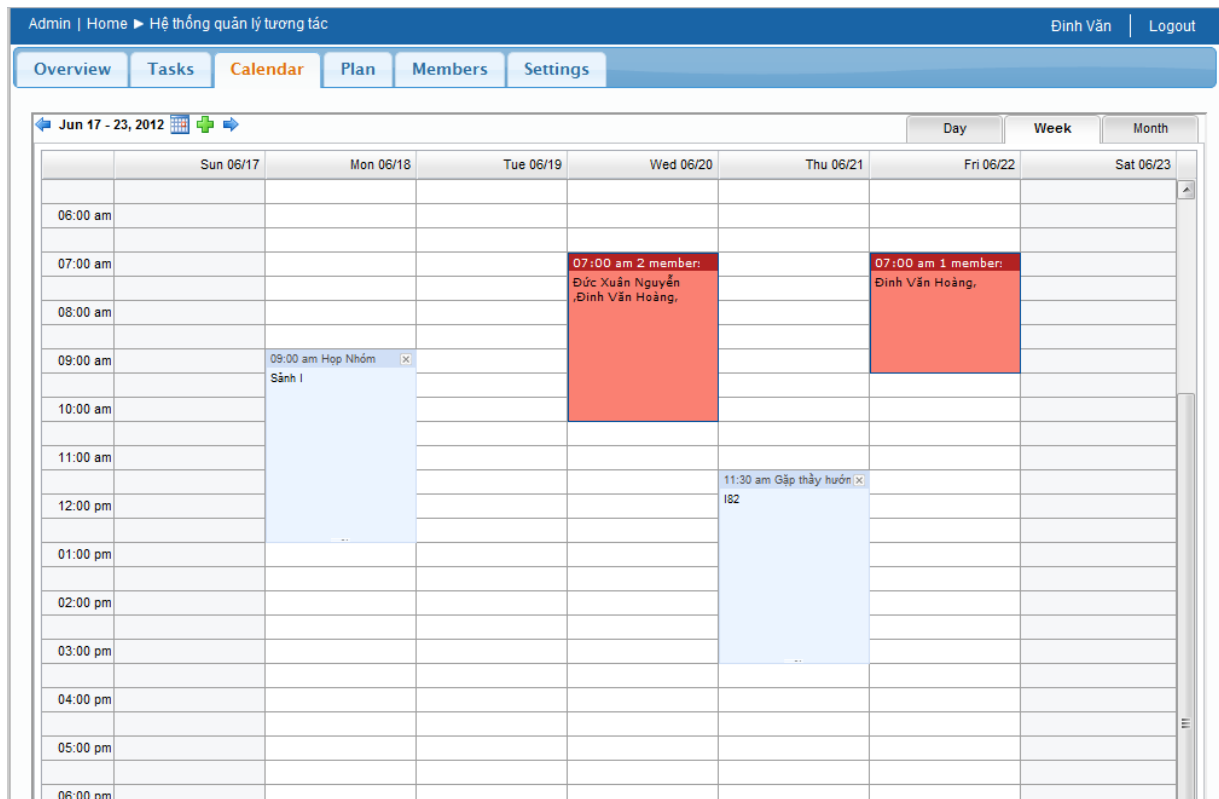
Người dùng có thể tạo, chỉnh sửa hoặc xóa các sự kiện cá nhân.



Hình 22 Lịch cá nhân

4.3.6.8 Màn hình project calendar

Giao diện project calendar sẽ hiển thị các sự kiện của dự án, sự kiện có màu xanh, và cho phép tạo, chỉnh sửa, xóa sự kiện của dự án. Bên cạnh đó thì hệ thống cũng sẽ hiển thị sự kiện của các thành viên dự án, sự kiện có màu đỏ. Tuy nhiên để đảm bảo quyền riêng tư của từng thành viên, hệ thống sẽ không hiển thị thông tin chi tiết về sự kiện của thành viên. Trong dự án không thể thay đổi sự kiện của thành viên.



Hình 23 Lịch dự án

4.3.6.9 Màn hình hiển thị work item

Thành viên có thể xem hoặc cập nhật thông tin của work item. Trong khi tạo hoặc cập nhật work item người dùng có thể kèm theo file liên quan tới work item.

Bên cạnh đó để hỗ trợ tương tác, hệ thống cho phép thành viên thảo luận về work item hoặc đăng kí để nhận thông báo qua email khi có thành viên thay đổi work item hoặc bình luận về work item.

Admin | Home ► Hệ thống quản lý tương tác

Bình

Logout

Overview

Tasks

Calendar

Plan

Members

Settings

Tasks > Update Task : #1

Save

Cancel

Title :

Thiết kế cơ sở dữ liệu

Description :

Thiết kế cơ sở dữ liệu cho hệ thống quản lý tương tác.
Xem danh sách chức năng trong đặc tả yêu cầu.

Details

Status :

New

Priority :

High

Planned for :

Please choose ...

Date Created :

24-06-2012 09:47

Due Date :

Date Last Edit :

24-06-2012 09:53

Peoples

Created by :

Đức Xuân Nguyễn

Assigned to :

Văn Hoàng Đình

Unsubscribed

You and 1 members subscribed this.

Attachments

Attach a file

Remove

ThietKe_V1.0.docx

Remove

ThietKe_V1.1.docx

Activity

Comments

History

Đức Xuân Nguyễn task được giao cho Văn Hoàng Đình.
at 2012-06-24 09:49:27.0

Văn Hoàng Đình ok. :D
at 2012-06-24 09:50:00.0

Văn Hoàng Đình Đã xong thiết kế V1.0. Mọi người xem thử cần bổ sung gì không. @_@
at 2012-06-24 09:50:42.0

Bình Nguyễn Theo t nghĩ nên bổ sung bảng Account. :))
at 2012-06-24 09:51:43.0

Văn Hoàng Đình ok. thanks Bình Nguyễn. Đã sửa xong, V1.1
at 2012-06-24 09:52:51.0

Đức Xuân Nguyễn good. task finish. 8-)
at 2012-06-24 09:53:50.0

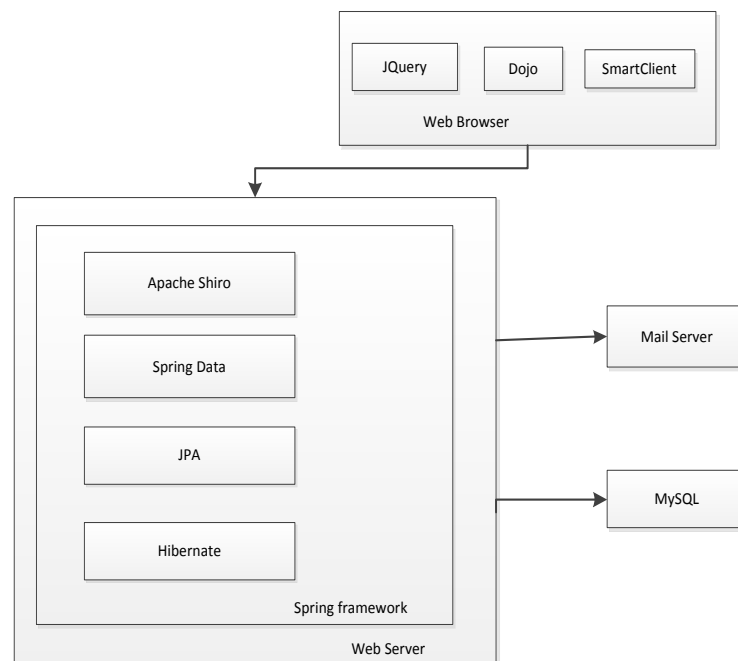
Add

Hình 24 Hiển thị work item

4.3.7 Thư viện hỗ trợ:

Tại trình duyệt của máy người dùng TeamSpace sử dụng các thư viện javascript là JQuery , Dojo, SmartClient để thể hiện các thành phần giao diện phức tạp và thực hiện việc gửi và nhận dữ liệu với server dùng kĩ thuật Ajax.

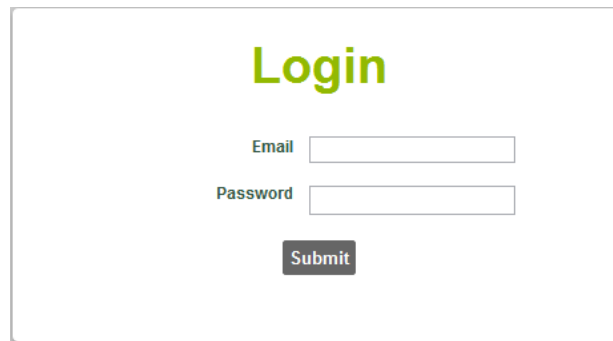
Tại server, TeamSpace sử dụng Spring framework. Bên cạnh đó thì tại server cũng có sử dụng các thư viện bổ sung khác như Apache Shiro nhằm hỗ trợ bảo mật hệ thống và sử dụng Spring Data, Java Persistence API và Hibernate để hỗ trợ thao tác với cơ sở dữ liệu quan hệ.



Hình 25 Các thư viện hỗ trợ

4.4 Hướng dẫn sử dụng

Sau khi triển khai ứng dụng lên server, khởi động ứng dụng để thấy màn hình đăng nhập.



Login

Email

Password

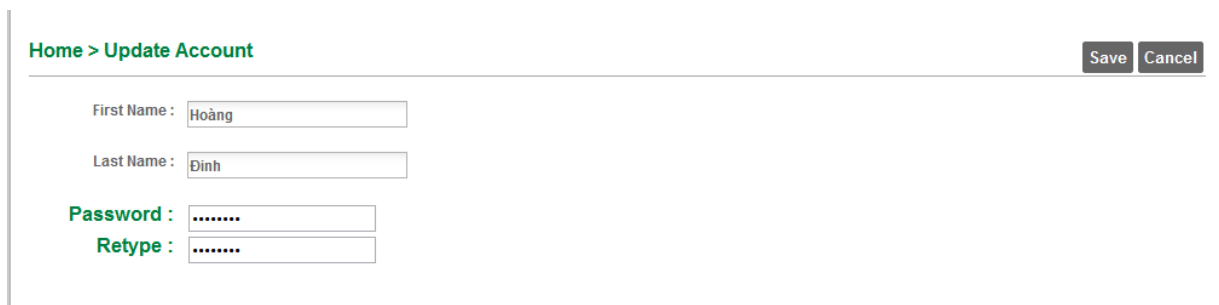
Submit

Hình 26 Màn hình login

Username mặc định ban đầu là admin@tis.teamspace.com và password : 12345.

Sau khi đăng nhập với quyền admin. Người dùng sẽ tới màn hình dashboard. Đây là giao diện giành cho người dùng. Admin cũng có thể được coi như 1 người dùng bình thường.

Admin nên đổi password ngay sau khi đăng nhập lần đầu.



Home > Update Account **Save** **Cancel**

First Name :

Last Name :

Password :

Retype :

4.4.1 Tạo process

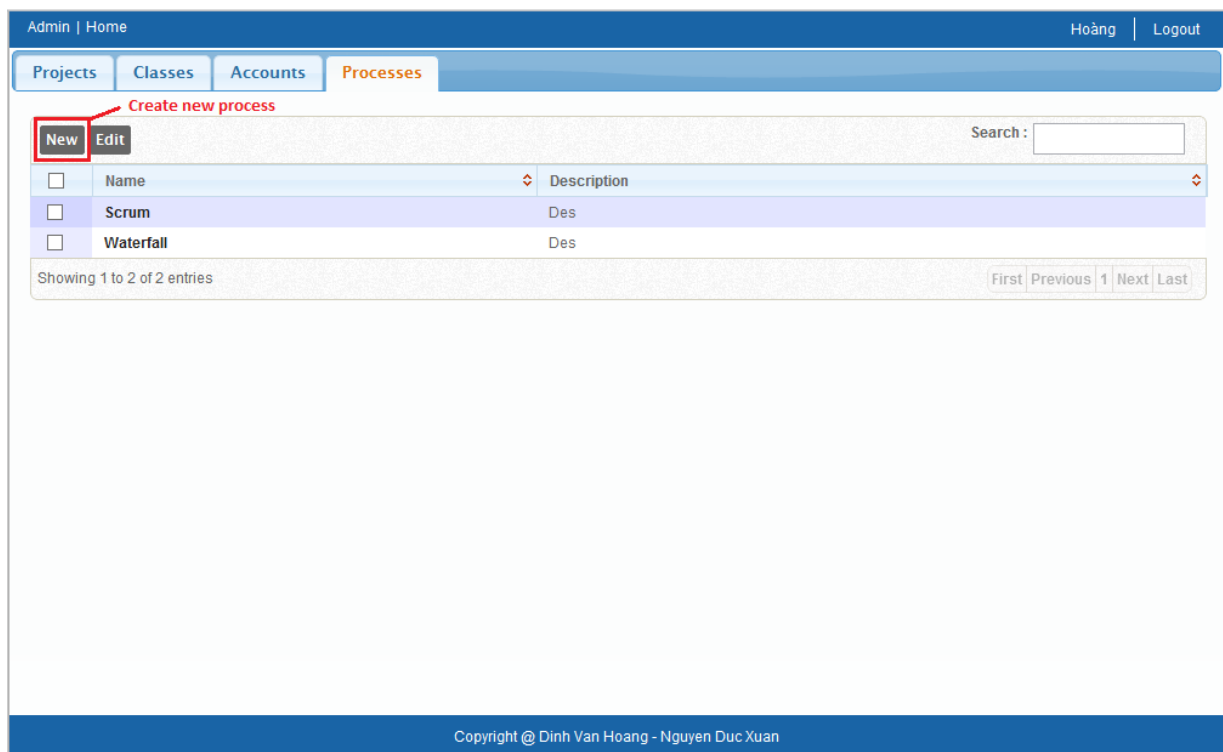
Đây là bước đầu tiên 1 admin nên làm. Bởi process là điều kiện đầu tiên trước khi tạo các project. Click vào link admin trên thanh menu để chuyển sang màn hình admin.



Admin | Home Hoàng | Logout

Dashboard | My Calendar | My Projects

Sau khi đã ở màn hình admin. Chuyển sang tab Processes để tạo processes mới.



Click button *New* để thực hiện tác vụ tạo mới process.

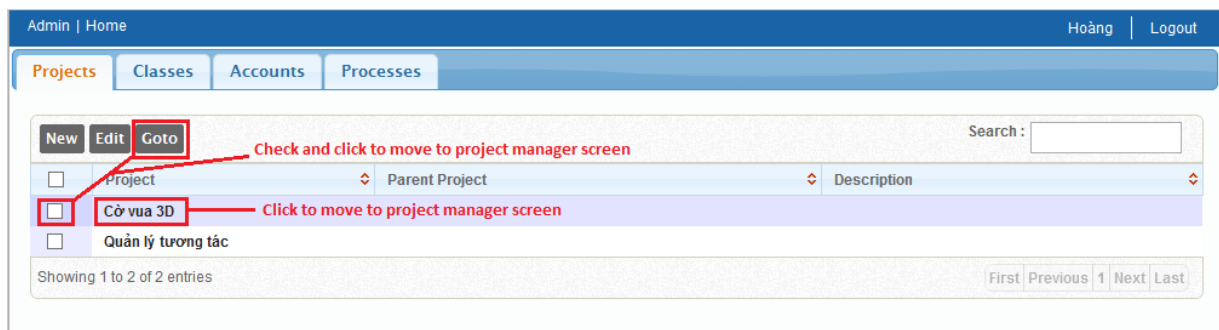


Chọn file XML mô tả process và upload lên server để kết thúc quá trình tạo process.

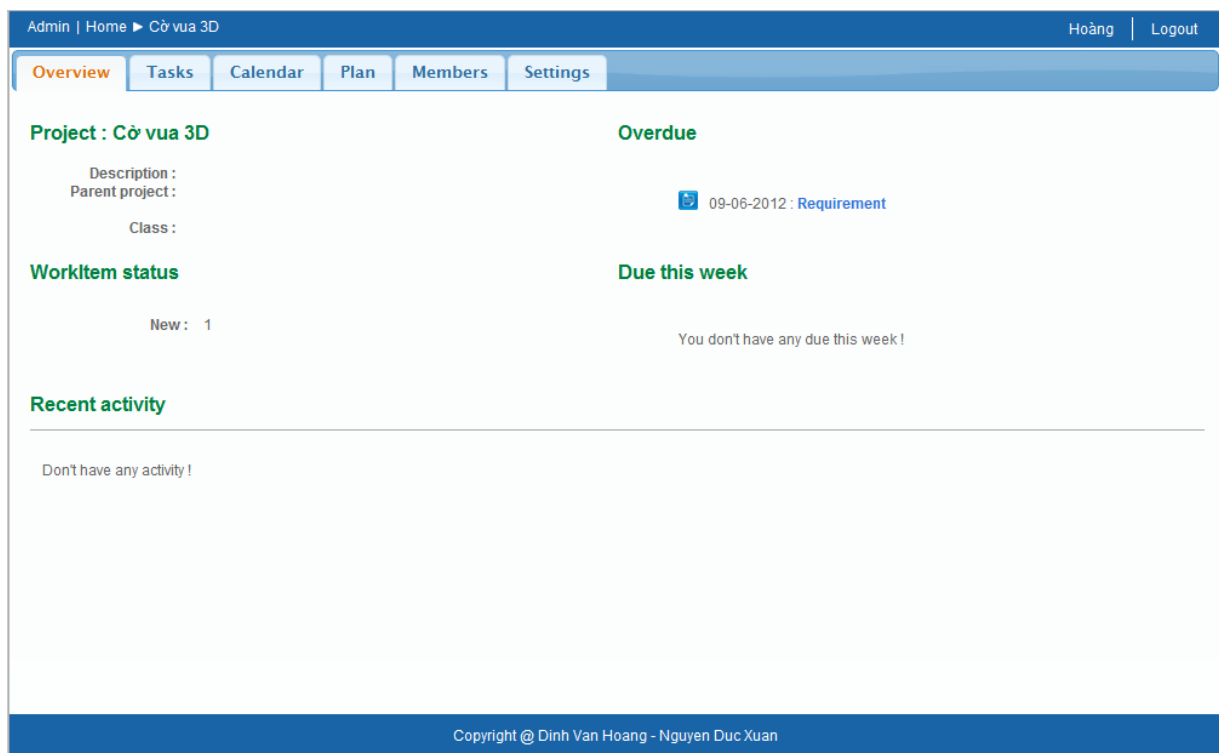
Sau khi tạo process, admin có thể tạo class, project, account và tiến hành các thao tác quản lý như thêm, xóa, sửa.

4.4.2 Quản lý project

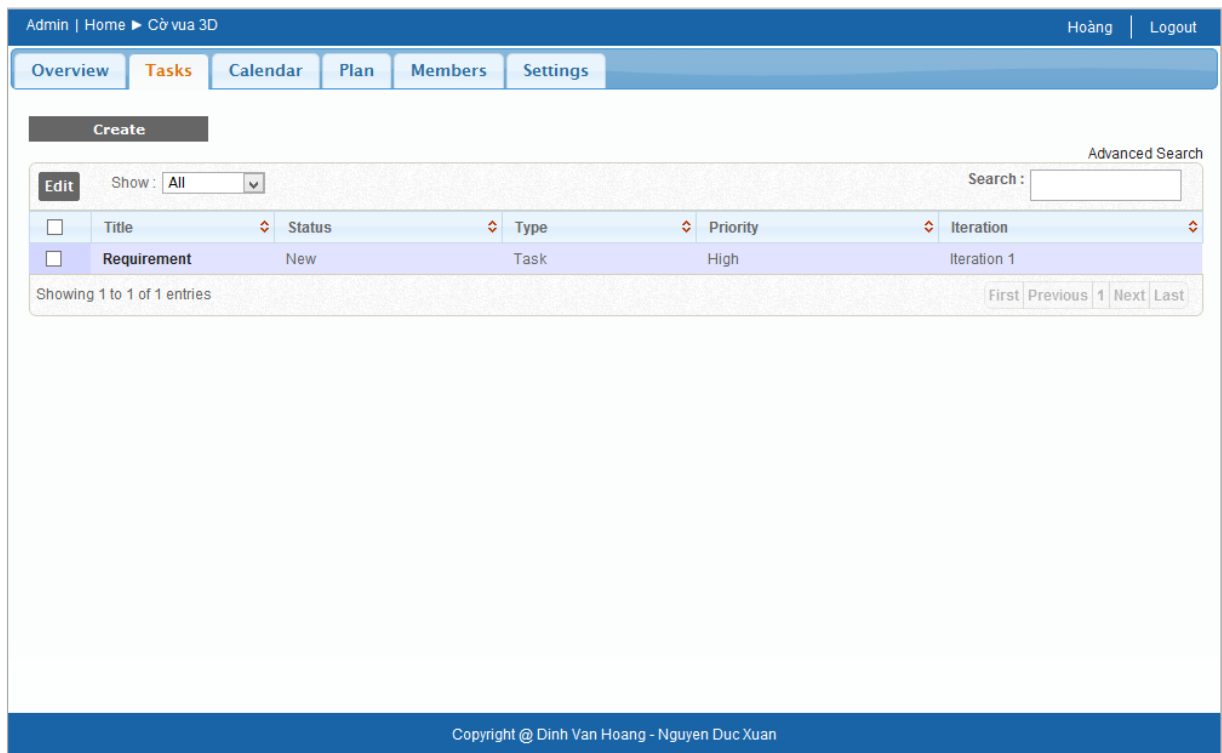
Click vào tên project hoặc check project và chọn Goto để chuyển tới giao diện quản lý project.



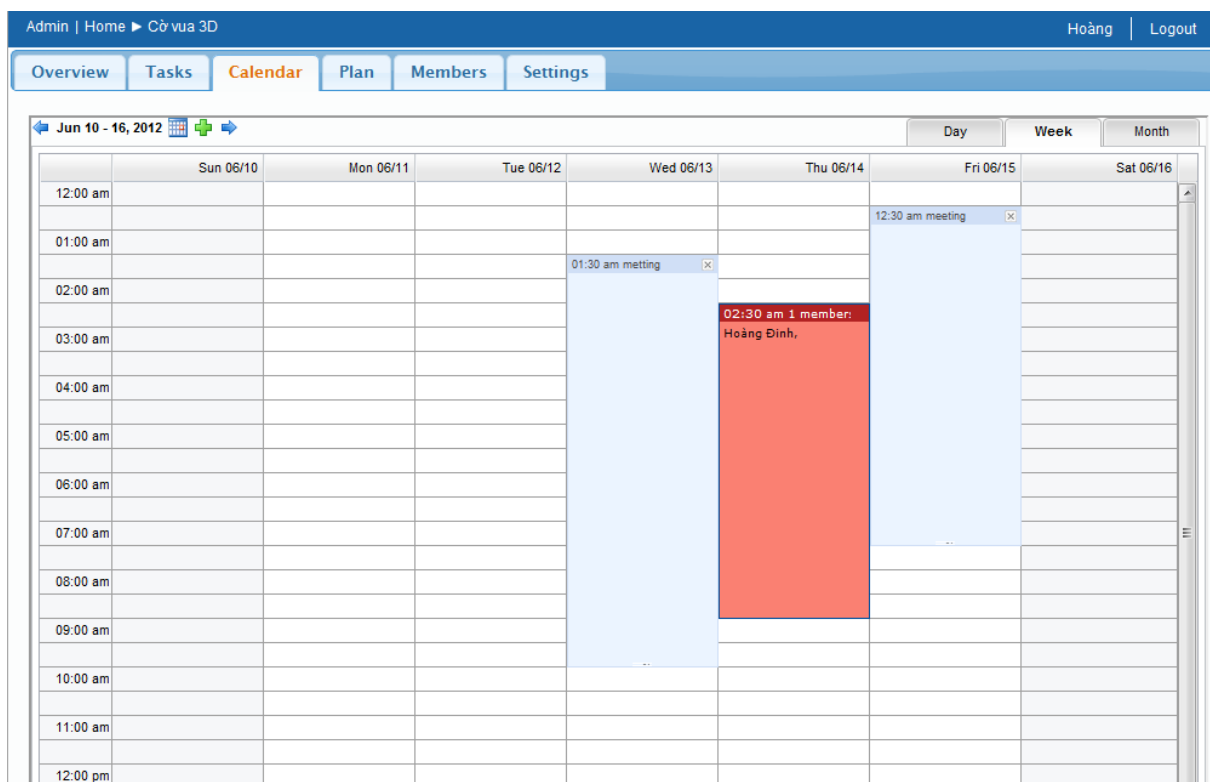
Tại giao diện quản lý project, người dùng có thể xem những thông tin tổng quát về project như tên, mô tả, những lớp học, project liên quan, overdue, due this week, recent activity.



Admin sẽ thêm thành viên của project tại tab Members. Chỉ có thành viên đã được thêm vào project thì mới có thể tạo thao tác tại tab Tasks.



Thành viên sẽ xem và tạo lịch tại tab Calendar. Những sự kiện chung của project sẽ có màu xanh nhạt, của các thành viên sẽ có màu đỏ sậm.



Chương 5 KẾT LUẬN

Qua quá trình tìm hiểu về một số hệ thống hỗ trợ tương tác sẵn có, tìm hiểu về các quy trình xây dựng phần mềm, chúng em đã hoàn thành đề tài “Xây dựng hệ thống quản lý tương tác”. Hệ thống có thể hỗ trợ sinh viên và giáo viên trong quá trình thực hiện đề án.

Một là sinh viên viên có thể tham gia vào các dự án phát triển phần mềm. Có thể tạo và theo dõi các vấn đề trong dự án.

Hai là hỗ trợ sinh viên tương tác với nhau trong quá trình thực hiện, không cần phải phải gặp mặt trực tiếp. Giảm thời gian và chi phí đi lại.

Ba là hỗ trợ lên kế hoạch cho sự kiện của dự án như lên họp nhóm, đi chơi,... Việc này đặc biệt quan trọng đối với sinh viên nhằm giảm thiểu việc xung đột về thời gian rảnh giữa các thành viên trong nhóm.

Bốn là hỗ trợ cho giảng viên trong quản lý các nhóm trong lớp, phân chia nhóm, phân chia dự án, theo dõi, quản lý tiến độ thực hiện dự án của các nhóm, kịp thời điều chỉnh, hỗ trợ kịp thời nếu cần thiết.

Tuy nhiên, do xây dựng trong một thời gian ngắn, hệ thống vẫn còn nhiều vấn đề.

Một là chưa kiểm tra một cách đầy đủ các chức năng đã cài đặt. Do đó có thể có những lỗi chưa được phát hiện trong quá trình phát triển.

Hai là các yêu cầu phi chức năng như hiệu năng, độ tin cậy của hệ thống, tính tiện dụng của giao diện vẫn chưa được kiểm tra, đánh giá.

Ba là các tính năng vẫn chưa hoàn toàn đầy đủ như chưa tự động xếp lịch cho dự án, chưa có các tính năng thực hiện khảo sát trong nhóm và chưa có wiki của dự án.

Chương 6 HƯỚNG PHÁT TRIỂN

Hệ thống quản lý tương tác đang xây dựng là một phần nhỏ trong đề tài “*xây dựng môi trường tích hợp trên Web hỗ trợ cho đào tạo, nghiên cứu, và phát triển dự án trong công nghệ phần mềm*”. Đây là đề án đang được thực hiện tại khoa công nghệ thông tin, trường Đại Học Khoa Học Tự Nhiên. Trong tương lai sẽ tiến hành kiểm tra các chức năng đã được cài đặt đồng thời nghiên cứu tích hợp vào hệ thống “môi trường tích hợp trên Web hỗ trợ cho đào tạo, nghiên cứu, và phát triển dự án trong công nghệ phần mềm”.

Mạng xã hội facebook và Skype đang được dùng phổ biến trong sinh viên. Việc kết hợp hệ thống quản lý tương tác với các mạng xã hội sẽ giúp các thành viên tương tác dễ dàng hơn. Người dùng có thể truy cập vào hệ thống dùng tài khoản facebook và twitter, có thể có sự trao đổi dữ liệu và thông báo giữa hệ thống quản lý tương tác và các mạng xã hội.

Hệ thống tin nhắn tức thời như Skype và Yahoo cũng được sử dụng rộng rãi để tương tác trong nhóm phát triển phần mềm. Có thể kết hợp các hệ thống này với hệ thống quản lý tương tác để hỗ trợ tương tác tốt hơn. Nhóm là dự án có thể hợp nhóm trực tuyến dùng tài khoản Skype và Yahoo.

Mã nguồn, các tài liệu là những sản phẩm rất quan trọng trong quá trình phát triển phần mềm. Hệ thống trong tương lai sẽ kết hợp với hệ thống quản lý cấu hình phần mềm (software configuration management) để quản lý mã nguồn và tài liệu của dự án.

Chương 7 DANH MỤC TÀI LIỆU THAM KHẢO

1. Grady Booch and Alan W. Brown, *Collaborative Development Environments*
2. Jim Whitehead, *Collaboration in Software Engineering: a Roadmap*
3. Martin Fowler, David Rice, Matthew Foemmel, Edward Hieatt, Robert Mee, Randy Stafford, *Patterns of Enterprise Application Architecture*, Addison Wesley, 2002
4. <http://www.drdobbs.com/architecture-and-design/196900222>

Chương 8 PHỤ LỤC

Phụ lục A: Các khái niệm cơ bản

Work item: Là một vấn đề cần quan tâm trong quá trình phát triển phần mềm. work item có thể là yêu cầu, báo cáo lỗi, công việc, ...

Iteration: là giai đoạn nhỏ trong dự án. Ví dụ như trong dự án phát triển theo mô hình Scrum thì iteration tương đương với sprint.

Process: là quy trình phát triển phần mềm. Có nhiều quy trình như Water Fall, Scrum, XP...

Process Template: là file định nghĩa quy trình phát triển phần mềm. File này định nghĩa các loại work item, mối quan hệ giữa các work item.

Phụ lục B: Các màn hình chi tiết

Màn hình Classes

Admin | Home

admin | Logout

Projects

Classes

Accounts

Processes

New

Edit

Delete

Search :

<input type="checkbox"/>	Name	Description
<input type="checkbox"/>	Lập trình Game	TH2008
<input type="checkbox"/>	Lập trình hướng đối tượng	TH2009
<input type="checkbox"/>	Lập trình Windows	TH2010
<input type="checkbox"/>	Lập trình Java	TH2008
<input type="checkbox"/>	Java phân tán	TH2009
<input type="checkbox"/>	Lập trình trên thiết bị di động	TH2009
<input type="checkbox"/>	Hệ điều hành	
<input type="checkbox"/>	Cấu trúc dữ liệu và giải thuật	
<input type="checkbox"/>	Trí tuệ nhân tạo	
<input type="checkbox"/>	Thiết kế giao diện	

Showing 1 to 10 of 11 entries

First

Previous

1

2

Next

Last

Copyright @ Dinh Van Hoang - Nguyen Duc Xuan

Màn hình Accounts

Admin | Home

admin | Logout

Projects

Classes

Accounts

Processes

New

Edit

Delete

Search :

<input type="checkbox"/>	First Name		Last Name		Email		Status	
<input type="checkbox"/>	admin		admin		admin@yahoo.com		ACTIVE	
<input type="checkbox"/>	Hoàng		Đinh		kydrenw@yahoo.com		ACTIVE	
<input type="checkbox"/>	Ca		Nguyễn		canh2000@yahoo.com		ACTIVE	
<input type="checkbox"/>	Hiếu		Nguyễn		hieu@gmail.com		INACTIVE	
<input type="checkbox"/>	Phước		Tăng		phuoc@gmail.com		INACTIVE	
<input type="checkbox"/>	Huyền		Nguyễn		huyen@gmail.com		INACTIVE	
<input type="checkbox"/>	Việt Anh		Nguyễn		a@gmail.com		ACTIVE	
<input type="checkbox"/>	Bích Thủy		Phạm		sdf@gmail.com		BLOCKED	
<input type="checkbox"/>	Thu Hằng		Trần		dfhrth@gmail.com		ACTIVE	
<input type="checkbox"/>	Bích Thảo		Lương		hieua@gmail.com		ACTIVE	

Showing 1 to 10 of 11 entries

First

Previous

1

2

Next

Last

Copyright @ Đinh Văn Hoàng - Nguyễn Đức Xuân

Màn hình Processes

Admin | Home

admin | Logout

Projects

Classes

Accounts

Processes

New

Edit

Search :

<input type="checkbox"/>	Name	↕	Description	↕
<input type="checkbox"/>	Scrum		Des	
<input type="checkbox"/>	Waterfall		Des	

Showing 1 to 2 of 2 entries

First

Previous

1

Next

Last

Phụ lục C: Các thư viện hỗ trợ

Spring framework

Website: <http://www.springsource.org/spring-framework>

Spring Framework là một framework cung cấp mô hình lập trình và cấu hình cho các ứng dụng doanh nghiệp hiện đại dựa trên Java

Giấy phép: Apache License 2.0

Hibernate

Website: <http://www.hibernate.org>

Hibernate là một thư cho phép kết nối từ mô hình đối tượng sang cơ sở dữ liệu quan hệ truyền thống cho ngôn ngữ Java.

Giấy phép: GNU Lesser General Public License

SmartClient

Website: <http://www.smartclient.com/product/smartclient.jsp>

SmartClient kết hợp các thành phần giao diện không phụ thuộc vào trình duyệt với framework chạy trên máy ảo java ở server nhằm cung cấp giải pháp xây dựng ứng dụng web thương mại.

SmartClient có nhiều bản thích hợp với từng mục đích khác nhau như LGPL edition, Professional Edition và Power Edition.

Trong ứng dụng **hệ thống quản lý tương tác** phiên bản “LGPL edition” được sử dụng để xây dựng chức năng hỗ trợ việc lên kế hoạch của cá nhân và dự án.

DataTables

Website: <http://www.datatables.net/>

Đây là một plugin của *jquery* hỗ trợ thêm các thành phần cho phép thao tác dễ dàng hơn vào thẻ table của html.

DataTables là một plugin mã nguồn mở. DataTables có 2 giấy phép là GPL v2 *license* và *BSD (3-point) license*

Trong ứng dụng “**Hệ thống quản lý tương tác**”, DataTables được dùng để hiển thị dữ liệu dưới dạng bảng.

Apache SHIRO

Website: <http://shiro.apache.org/>

Apache SHIRO một “Java security framework” mạnh mẽ và dễ dùng, thực hiện việc xác thực (authentication), ủy quyền (authorization), mã hóa và quản lý session.

Giấy phép: *Apache License 2.0*

Trong ứng dụng “**Hệ thống quản lý tương tác**”, Apache Shiro được dùng để thực hiện việc xác thực và phân quyền.