[Chương 1 GROUPSPACE – HỆ THỐNG QUẢN LÝ TƯƠNG TÁC CHO CÁC ĐỀ ÁN MÔN HỌC](#_Toc329801889)

[1.1 Chức năng](#_Toc329801891)

[1.1.1 Quản lý quy trình phát triển phần mềm](#_Toc329801892)

[1.1.2 Quản lý dự án](#_Toc329801893)

[1.1.3 Quản lý lớp học](#_Toc329801894)

[1.1.4 Quản lý work item và tương tác trong nhóm](#_Toc329801895)

[1.1.5 Quản lý lịch cá nhân và lịch dự án](#_Toc329801896)

[1.2 Yêu cầu](#_Toc329801897)

[1.2.1 Yêu cầu chức năng](#_Toc329801898)

[1.2.2 Yêu cầu phi chức năng](#_Toc329801899)

[1.3 Thiết kế](#_Toc329801900)

[1.3.1 Mô hình triển khai](#_Toc329801901)

[1.3.2 Kiến trúc tổng quan](#_Toc329801902)

[1.3.3 Các lớp chính trong domain model](#_Toc329801903)

[1.3.4 Cơ sở dữ liệu](#_Toc329801904)

[1.3.5 Cơ chế bảo mật](#_Toc329801905)

[1.3.6 Thiết kế giao diện](#_Toc329801906)

[1.3.7 Thư viện hỗ trợ:](#_Toc329801907)

[1.4 Hướng dẫn sử dụng](#_Toc329801908)

[1.4.1 Tạo process](#_Toc329801909)

[1.4.2 Quản lý project](#_Toc329801910)

# GROUPSPACE – HỆ THỐNG QUẢN LÝ TƯƠNG TÁC CHO CÁC ĐỀ ÁN MÔN HỌC



## Chức năng

GroupSpace được thiết kế với những tính năng chính sau:

### 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.

### 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.

### Quản lý lớp học

Một lớp học sẽ dạy một môn học. Mỗi lớp học 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 lớp học sẽ dễ dàng cho việc quản lý khi mà số lượng dự án ngày càng lớn.

### Quản lý work item và tương tác trong nhóm

GroupSpace hỗ trợ quản lý work item của dự án. 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. . 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. Các chức năng của GroupSpace hỗ trợ quản lý work item và tương tác cụ thể như sau:

**Tạo và câp 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 cho biết tiêu đề (title), độ ưu tiên, trạng thái của work item. Người dùng cũng có thể mô tả chi tiết 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.

### 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 và 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 họp thích hợp. Lịch họp 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.

## Yêu cầu

### Yêu cầu chức năng

#### Danh sách các Actor



Hình 4.1 Danh sách actor

|  |  |  |
| --- | --- | --- |
| STT | Actor | Mô tả |
| 1 | User | Người sử dụng chương trình đã có tài khoản trong hệ thống và tài khoản 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 member | Thành viên của nhóm thực hiện đồ án. Thành viên của dự án có thể thực hiện hầu hết các chức năng liên quan tới dự án như xem và tạo công viêc, 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 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 hiên tại. |

Bảng 1 Danh sách actor

#### Sơ đồ Use-Case

##### Mô hình Use-Case

Chia làm 3 nhóm chính là nhóm quản lý tài khoản, nhóm quản lý dự án, nhóm quản lý các hoạt động trong dự án. Nhóm *quản lý tài khoản* liên quan tới việc quản lý tài khoản người dùng và 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 việc quản lý work item, tương tác trong nhóm và quản lý lịch của dự án.

##### Account management



Hình 4.2 Mô hình use case trong nhóm chức năng quản lý tài khoản

##### Project management



Hình 4.3 Mô hình use case trong nhóm chức năng quản lý dự án

##### Project activity



Hình 4.4 Mô hình use case trong nhóm chức năng quản lý hoạt động trong dự án

#### 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 thúc tối thiểu | Khôi phục lại tình trạng hệ thống trước khi tạo work item |
| Trigger | Thành viên chọn chức năng tạo work item |
| Dòng sự kiện chính | 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 | 2a. Thành viên cung cấp thông tin không đúng.  2a1. Hệ thống thông báo đã người dùng đã nhập sai thông tin và yêu cầu nhập lại.  2a2. Thực hiện lại bước 2.  2b. Thành viên muốn hủy bỏ quá trình tạo work item.  2b1. Kết thúc use case và không ghi nhập thông tin về work item người dùng đang tạo.  2c. Thành viên có thể attach file trong khi tạo work item . |

##### 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 | 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 | 2a. Thành viên cung cấp thông tin không đúng.  2a1. Hệ thống thông báo người dùng đã nhập sai thông tin và yêu cầu nhập lại.  2a2. Thực hiện lại bước 2.  2b. Thành viên muốn hủy bỏ quá trình cập nhật work item.  2b1. Kết thúc use case và không ghi nhập thay đổi trên work item.  2c. Thành viên có thể attach file trong khi câp nhật work item.  2d. Thành viên có thể remove file khi câp nhật work item.  2e. Thành viên có thể tải về file đã được đính kèm theo work item trước đó. |

##### 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 công | Hê thống lưu trữ vào database |
| 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 | 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 | 2a. Admin nhập sai định dạng email.  2a1. Hệ thống báo sai định dạng.  2a2. Thực hiện lại bước 2. |

##### 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 chính | 1. Admin thực hiện chức năng tìm kiếm 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 | 1. Người dùng cung cấp thông tin về sứ 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. |
| 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 | 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. |

#### Sơ đồ lớp mức phân tích



Hình 4.5 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 work item (có thể là iteration hoặc project) |
| 2 | iteration | Một giai đoạn 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 | 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 work item. Lưu lại lịch sử các lần thay đổi trên work item. |
| 6 | attachment | Tập tin đính kèm. Ứng với mỗi work item người dùng có thể kèm theo các file để mô tả work item rõ ràng hơn. |
| 7 | comment | Bình luận. Tương ứng với bình luận của người dùng đối với work item. |
| 8 | account | Tài khoản của người dùng trong hệ thống. |
| 9 | proccess | Quy trình phát triển phần mềm. Quy trình phát triển sẽ quy định các loại work item 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 | 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 hiên. Có thể là yêu cầu chức năng hoặc phi chức năng. |

Bảng 2 Các lớp đối tượng mức phân tích

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

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

Hệ thống sử dụng Ajax tuy nhiên vẫn phải cho người dùng sử dụng các chức năng history, bookmarks của trình duyệt.

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.

## 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 gởi và nhận dữ liệu với server dùng kĩ thuật *Ajax*. Tạimáy client sử dụng javascript framework JQuery, SmartClient nhằm thể hiện các đối tượng giao diện phức tạp và hỗ trợ việc gởi và nhận dữ liệu giữa client và server dùng Ajax.

### Mô hình triển khai

Người dùng có thể truy cập vào ứng dùng 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, gởi mail thông qua Mail Server.



Hình 4.6 Mô hình triển khai

### 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 4.7 Kiến trúc tổng quan

**Presentation Layer**

Tầng xử lý tương tác giữa người dùng với hệ thống. Trong hệ thống GroupSpace 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 chứa 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 4.8 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 nàydược 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 trao đổi dữ liệu và tách biệt giữa các đối tượng trong bộ nhớ và cơ sở dữ liệu.



Hình 4.9 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 và 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.

#### 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 gởi mail và hỗ trợ đọc file.

### 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* 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 lýsự kiện của cá nhân cũng như của cả dự án.

*Các lớp trong Domain model đều có thuộc tính id và version để có thể lưu vô 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.*

#### 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*. Hệ thống sẽ kiểm tra việc truy cập của người dùng dựa trên quyền này.



Hình 4.10 Các lớp chính liên quan tới việc quản lý nhóm và dự án

#### 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 *getAdditionFields* 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>*

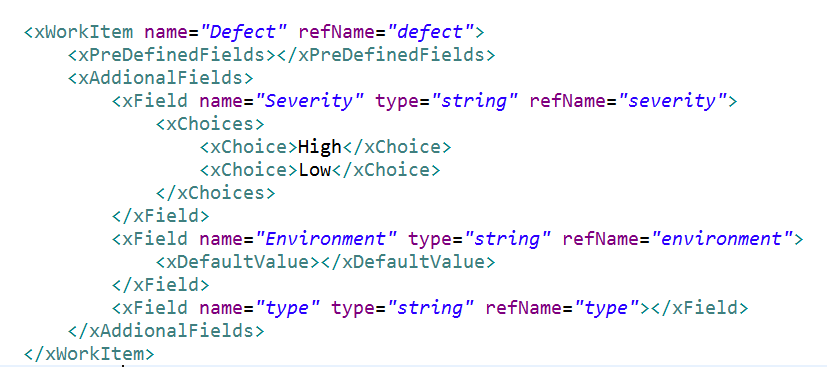
*<xField ref="environment">windows 7</xField>*

*<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





Hình 4.11 Các lớp chính liên quan tới quản lý work item và tương tác

#### 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 4.12 Các lớp chính liên quan tới quản lý lịch

### 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 4.13 Class Table Inheritance

### 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 2 loại vai trò là vai trò trong hệ thống và vai trò trong dự án cụ thể. 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”.

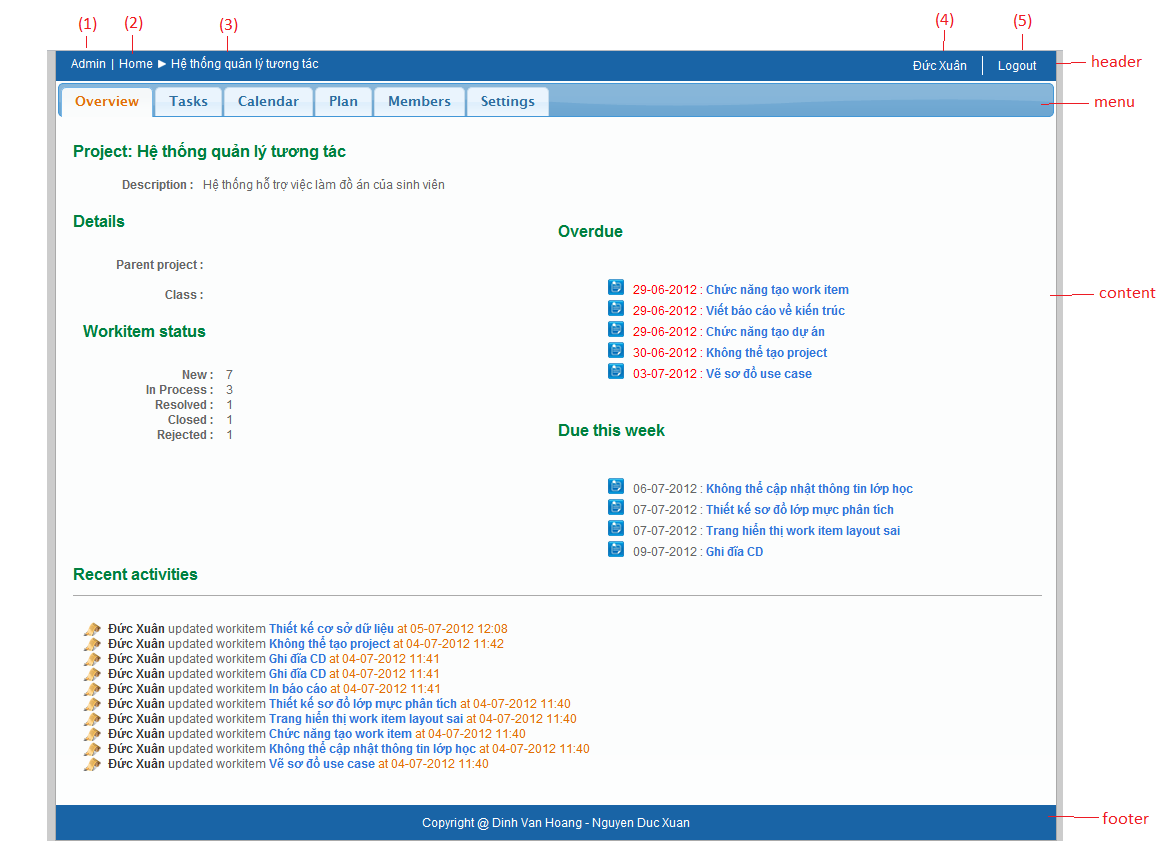
Quyền của người dùng trong hệ thống có dạng “domain : operation”. Trong đó domain là lĩnh vực thao tác được thực hiện. Operation là thao tác được thực hiện trên lĩnh vực. Ví dụ “project : list” thì người dùng sẽ có quyền xem danh sách các project trong hệ thống”. Bên cạnh đó để đảm bảo người dùng chỉ có thể cập nhật được tài khoản của mình hệ thống có kiểm tra quyền truy ở cấp độ thể hiện, 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 4.14 Mô hình vai trò của người dùng trong dự án

### Thiết kế giao diện

#### Cấu trúc chung của giao diện



Hình 4.15 cấu trúc chung của giao diện

(1)Liên kết tới trang quản lý của admin, (2) Liên kết tới trang nhà của người dùng,

(3)Tên dự án đang thao tác, (4) Tên người dùng hiện tại,

(5)Liên kết cho phép người dùng đăng xuất

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 và 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.

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

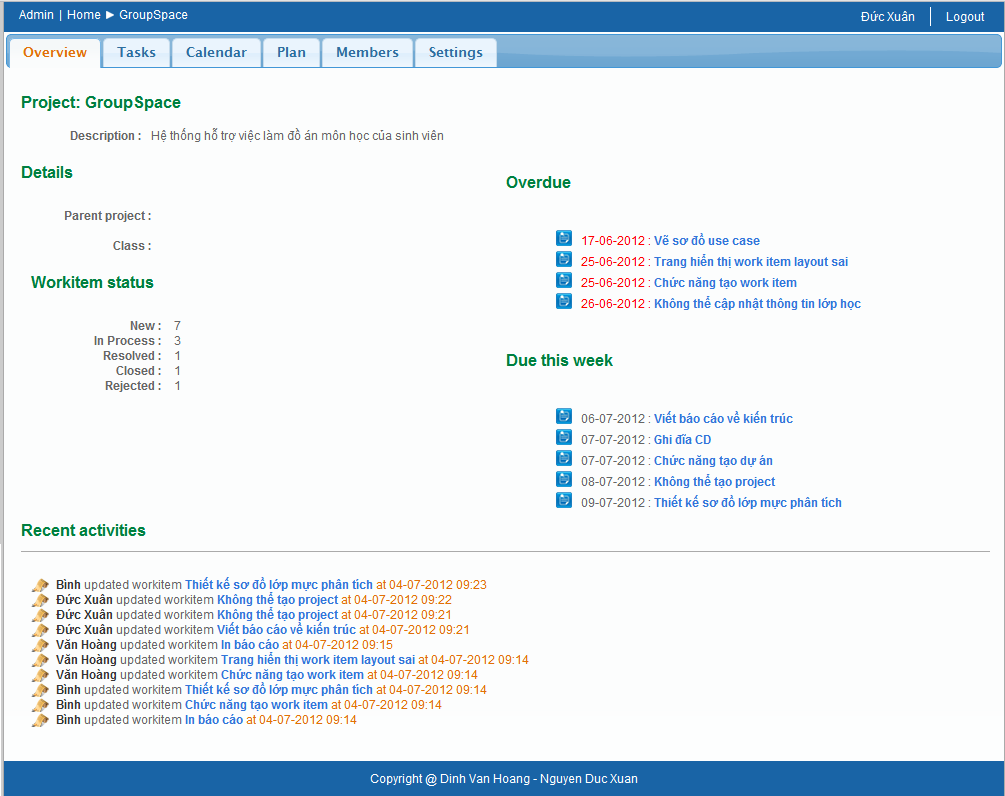


Hình 4.16 Giao diện admin

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 quy trình phát triển phần mềm (upload file XML mô tả process template), xóa quy trình phát triển phần mềm.

#### Giao diện khi người dùng ở trang của dự án

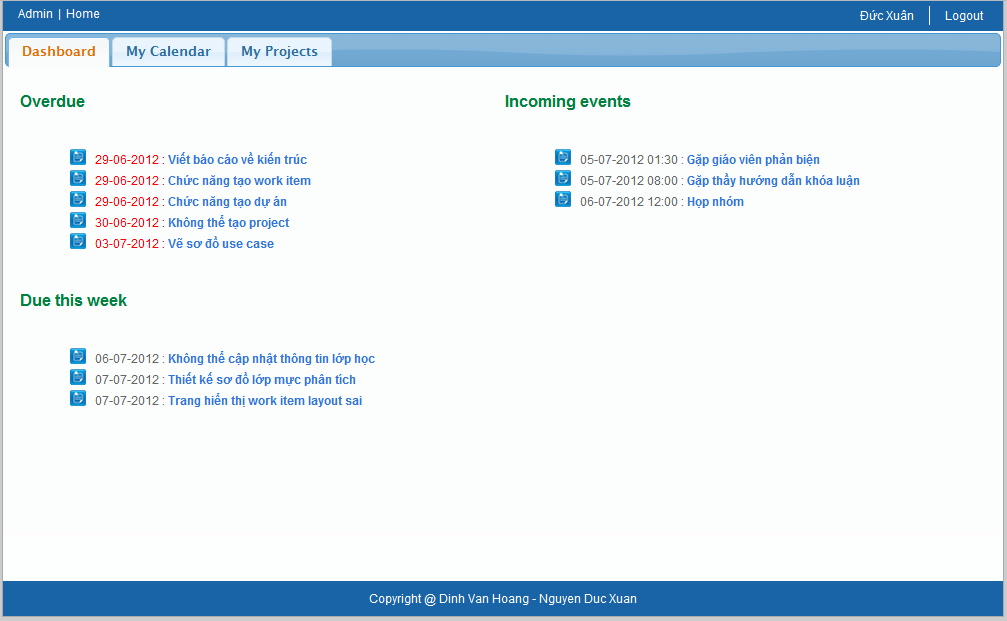


Hình 4.17 Giao diện project

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

* **Overview**: thông tin chung về project, các work item đã quá hạn, work item gần tới hạn cuối, hoạt động gần đây trong dự án.
* **Tasks**: quản lý work item. 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, cập nhật và xóa sự kiện của 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.

#### Giao diện nhà người dùng



Hình 4.18 Giao diện nhà người dùng

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

* **Dashboard**: 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 sự kiện 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.

#### 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.

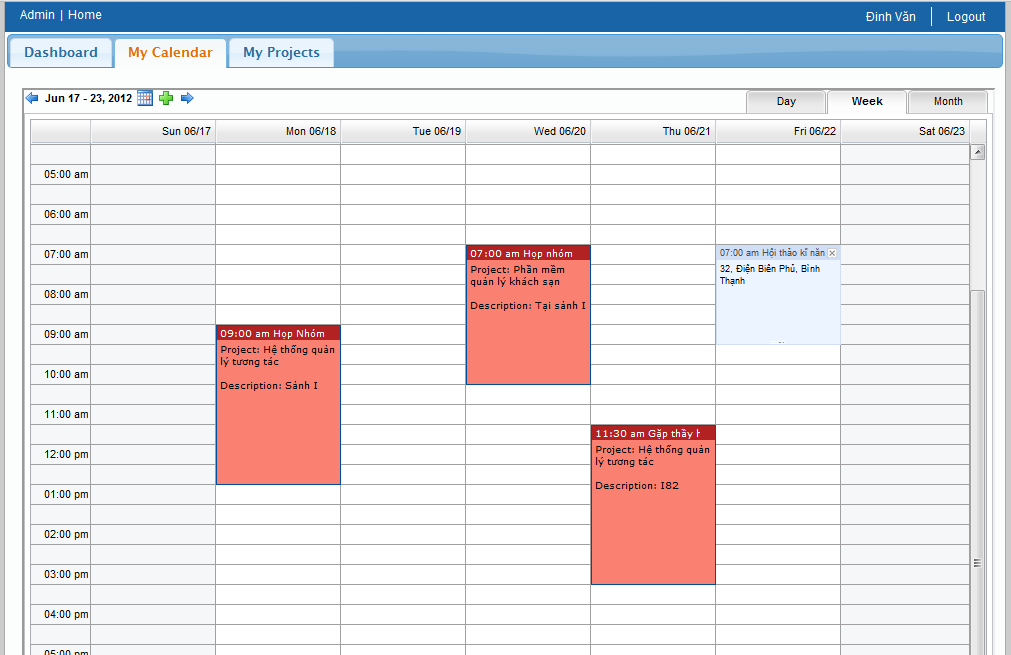
#### Màn hình dashboard của người dùng:

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.

#### 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.

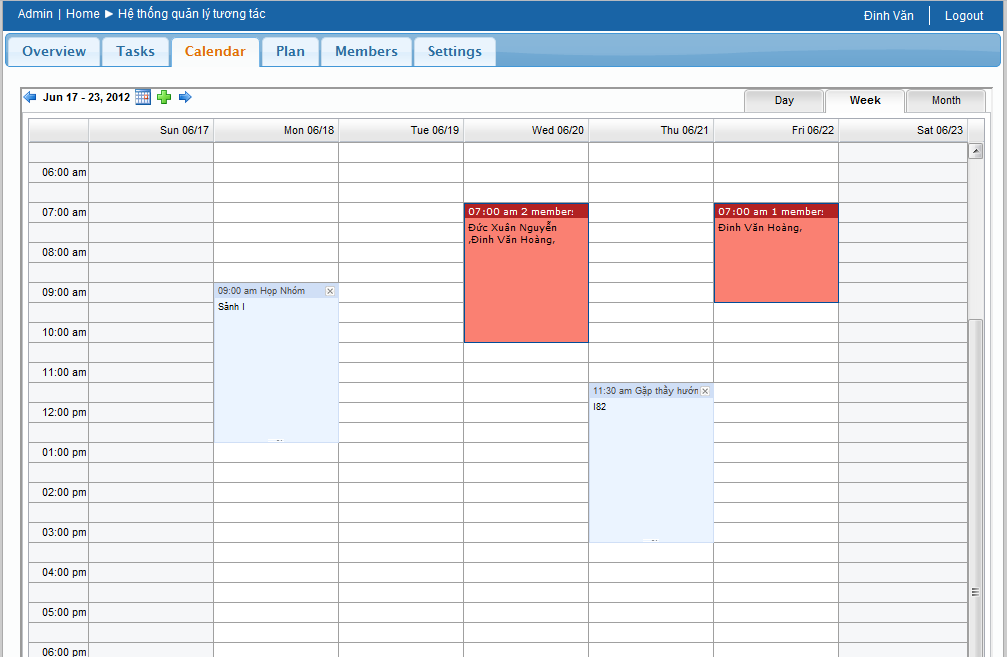
Người dùng có thể tạo, chỉnh sửa hoặc xóa các sự kiện cá nhân.



Hình 4.19 Màn hình lịch cá nhân

#### 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.

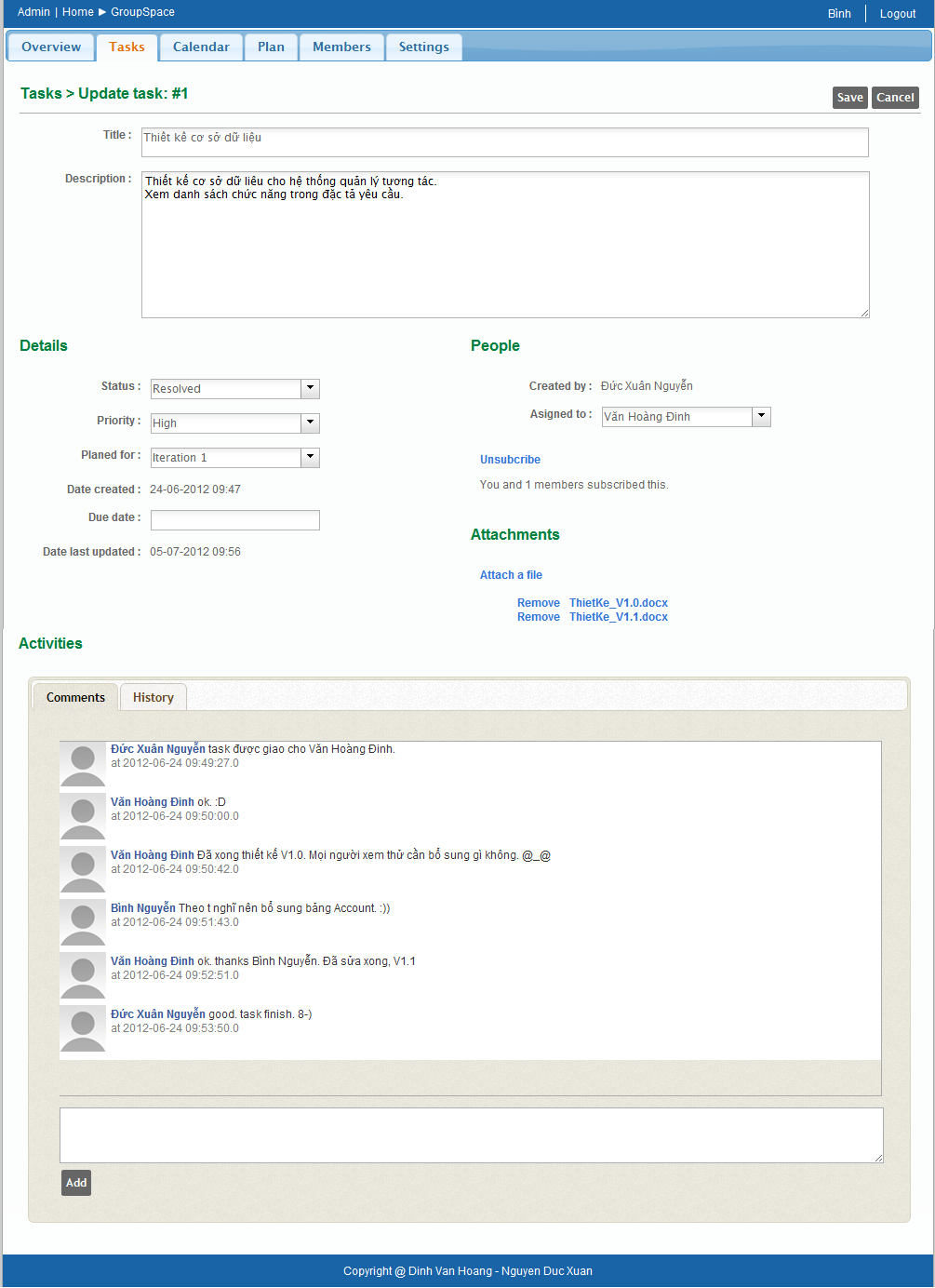


Hình 4.20 Màn hình lịch dự án

#### 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 hoặc bình luận về work item.



Hình 4.21 Màn hình hiển thị work item

### Thư viện hỗ trợ:

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

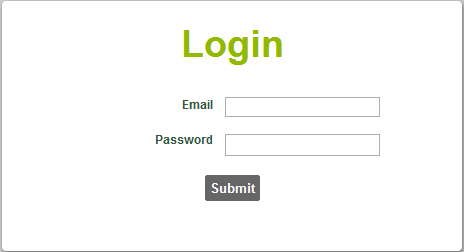
Tại server, GroupSpace sử dụng Spring framework. Bên cạnh đó thì tại server cũng có sử dụng các thư viện bổ sung khác như Apache Shiro nhằm hỗ trợ bảo mật hệ thống và sử dụng Spring Data, Java Persistence API và Hibernate để hỗ trợ thao tác với cơ sở dữ liệu quan hệ.



Hình 4.22 Các thư viện hỗ trợ

## Hướng dẫn sử dụng

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

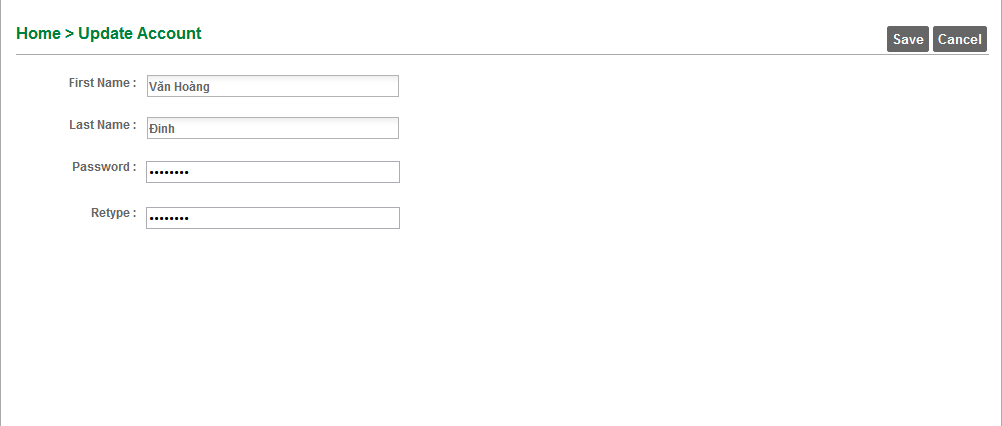


Hình 4.23 Hướng dẫn login

Username mặc định ban đầu là [admin@tis.GroupSpace.com](mailto: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.



Hình 4.24 Hướng dẫn thay đổi thông tin cá nhân

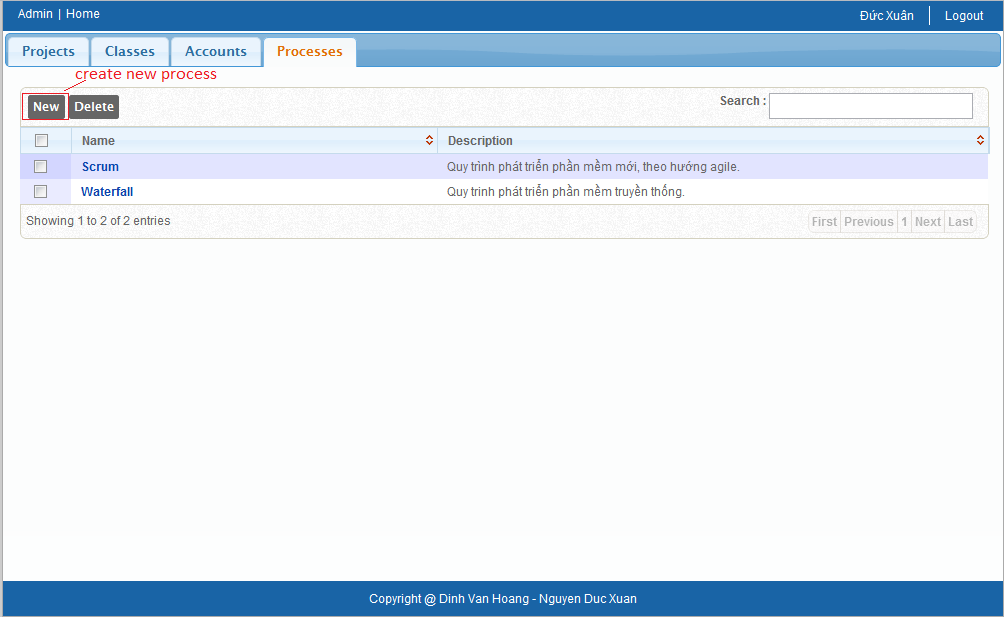
### Tạo process

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



Hình 4.25 Hướng dẫn chuyển sang giao diện admin

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



Hình 4.26 Hướng dẫn tạo quy trình phát triển phần mềm

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



Hình 4.27 Hướng dẫn upload file mô tả quy trình phát triển

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.

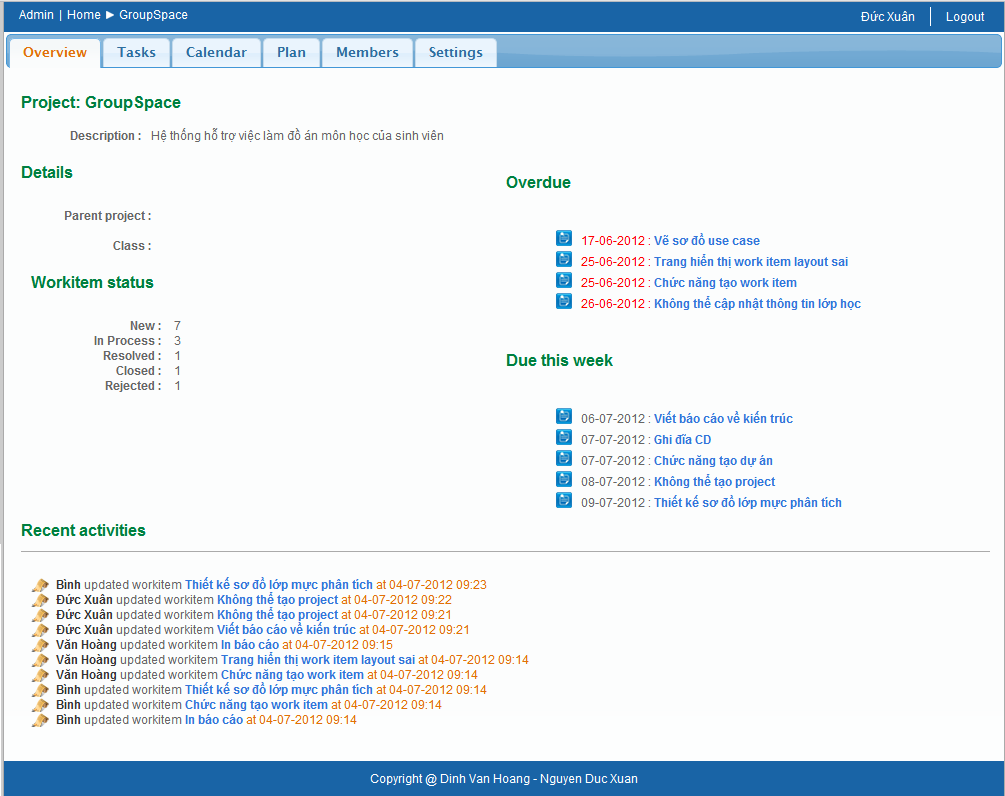
### Quản lý project

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



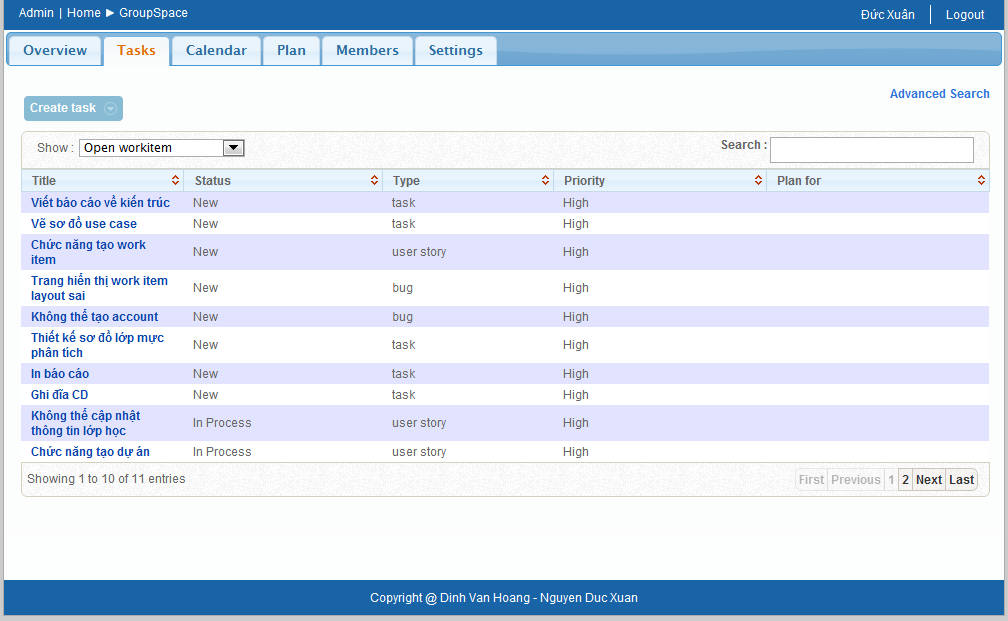
Hình 4.28 Hướng dẫn quản lý các dự án của hệ thống

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 activities.



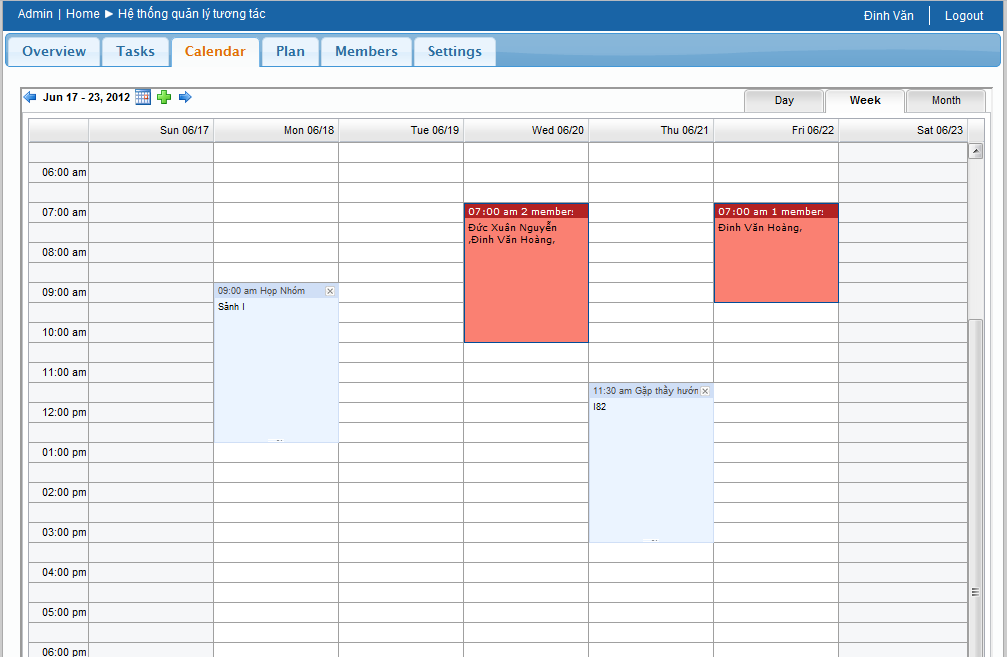
Hình 4.29 Hướng dẫn xem thông tin của dự án

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.



Hình 4.30 Hướng dẫn quản lý work item của dự án

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



Hình 4.31 Hướng dẫn quản lý lịch của dự án