**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

****

**BÀI TẬP LỚN**

**XÂY DỰNG WEBSITE GIỚI THIỆU PHẦN MỀM QUẢN LÝ CÔNG VIỆC SNAP WORK**

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

HƯỚNG CHUYÊN SÂU: CÔNG NGHỆ WEB

SINH VIÊN: **ĐỒNG VĂN CÔNG**

MÃ LỚP: **125201**

HƯỚNG DẪN: **NGUYỄN VĂN QUYẾT**

**HƯNG YÊN – 2023**

NHẬN XÉT

**Nhận xét của giảng viên hướng dẫn:**

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

.......................................................................................................................................

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**LỜI CAM ĐOAN**

Em xin cam đoan bài tập lớn “Xây dựng website giới thiệu phần mềm quản lý công việc Snap ” là kết quả thực hiện của bản thân em dưới sự hướng dẫn của thầy Nguyễn Văn Quyết.

Những phần sử dụng tài liệu tham khảo trong bài tập lớn đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong đồ án và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiển.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

*Hưng Yên, ngày … tháng … năm 2023*

Sinh viên

**LỜI CẢM ƠN**

Để có thể hoàn thành bài tập lớn này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin – Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn thầy Nguyễn Văn Quyết đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiển bài tập lớn vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được đồ án này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

**MỤC LỤC**

[NHẬN XÉT 2](#_Toc65656825)

[DANH MỤC CÁC THUẬT NGỮ 7](#_Toc65656826)

[DANH MỤC CÁC BẢNG 8](#_Toc65656827)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ 9](#_Toc65656828)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 11](#_Toc65656829)

[1.1. Lý do chọn đề tài 11](#_Toc65656830)

[1.2. Mục tiêu của đề tài 11](#_Toc65656831)

[1.2.1 Mục tiêu tổng quát 12](#_Toc65656832)

[1.2.2 Mục tiêu cụ thể 12](#_Toc65656833)

[1.3. Giới hạn và phạm vi của đề tài 13](#_Toc65656834)

[1.3.1 Đối tượng nghiên cứu 13](#_Toc65656835)

[1.3.2 Phạm vi nghiên cứu 14](#_Toc65656836)

[1.4. Nội dung thực hiển 15](#_Toc65656837)

[1.5. Phương pháp tiếp cận 16](#_Toc65656838)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 17](#_Toc65656839)

[2.1. Quy trình phát triển phần mềm 17](#_Toc65656840)

[2.2. Phương pháp phân tích thiết kế hướng đối tượng 19](#_Toc65656841)

[2.3. Hệ quản trị cơ sở dữ liệu SQL/PostgreSQL 22](#_Toc65656842)

[2.4. Thiết kế giao diện website 27](#_Toc65656843)

[2.5. Lập trình phía front-end 28](#_Toc65656844)

[2.6. Lập trình phía back-end 29](#_Toc65656845)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 20](#_Toc65656846)

[3.1 Đặc tả yêu cầu phần mềm 20](#_Toc65656847)

[3.1.1 Các yêu cầu chức năng 21](#_Toc65656848)

[3.1.2 Biểu đồ lớp thực thể 22](#_Toc65656849)

[3.1.3 Các yêu cầu phi chức năng 45](#_Toc65656850)

[3.2 Thiết kế hệ thống 49](#_Toc65656851)

[3.2.1 Thiết kê cơ sở dữ liệu 50](#_Toc65656852)

[3.2.2 Thiết kế lớp đối tượng 52](#_Toc65656853)

[3.2.3 Thiết kế giao diện 52](#_Toc65656854)

[CHƯƠNG 4: TRIỂN KHAI WEBSITE 53](#_Toc65656855)

[4.1 Triển khai các chức năng cho phân hệ người dùng 54](#_Toc65656856)

[4.1.1 Trang chủ 55](#_Toc65656857)

[4.1.2 Trang Giày Giày thể thao 56](#_Toc65656858)

[4.1.3 Trang Quản lý giỏ 57](#_Toc65656859)

[4.2 Triển khai các chức năng cho phân hệ quản trị nội dung 58](#_Toc65656860)

[4.3 Kiểm thử và triển khai ứng dụng 59](#_Toc65656861)

[4.3.1 Kiểm thử](#_Toc65656862) …………………………………………………………………...59

[4.3.2 Đóng gói ứng dụng 60](#_Toc65656863)

[4.3.3 Triển khai ứng dụng 60](#_Toc65656864)

[KẾT LUẬN 61](#_Toc65656865)

[TÀI LIỆU THAM KHẢO 62](#_Toc65656866)

DANH MỤC CÁC THUẬT NGỮ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Từ viết tắt | Cụm từ tiếng anh | Diễn giải |
| 1 | PHP | Hypertext Preprocessor | Ngôn ngữ lập trình kịch bản |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

DANH MỤC CÁC BẢNG

[Bảng 1. 1 Các chức năng của hệ quản trị](#_Toc135842328) 12

[Bảng 1. 2 Sự kiện chính quản lý loại giày thể thao](#_Toc135842329) 24

[Bảng 1. 3 Sự kiện chính của Giày thể thao](#_Toc135842330) 26

[Bảng 1. 4 Dữ liệu Category](#_Toc135842331) 28

[Bảng 1. 5 Dữ liệu bảng Sản phẩm](#_Toc135842332) 29

[Bảng 1. 6 Dữ liệu bảng tin tức](#_Toc135842333) 30

[Bảng 1. 7 Dữ liệu bảng hóa đơn](#_Toc135842334) 56

[Bảng 1. 8 Dữ liệu bảng chi tiết hóa đơn](#_Toc135842335) 65

[Bảng 1. 9 Dữ liệu bảng người dùng](#_Toc135842336) 66

[Bảng 1. 10 Dữ liệu bảng khách hàng](#_Toc135842337) 87

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

[Hình 1. 1 Công ty Viettel SoftWare Service 14](#_Toc153014957)

[Hình 1. 2 Biểu đồ Use Case trang quản trị hệ thống tổng quát 35](#_Toc153014958)

[Hình 1. 3 Biểu đồ use case chức năng quản lý công việc 35](#_Toc153014959)

[Hình 1. 4 Biểu đồ use case thêm công việc 36](#_Toc153014960)

[Hình 1. 5 Biểu đồ use case chức năng sửa công việc 37](#_Toc153014961)

[Hình 1. 6 Biểu đồ use case chức năng xóa công việc 38](#_Toc153014962)

[Hình 1. 7 Biểu đồ use case chức năng tìm kiếm công việc 39](#_Toc153014963)

[Hình 1. 8 Biểu đồ use case quản trị dự án 40](#_Toc153014964)

[Hình 1. 9 Biểu đồ use case thêm dự án 41](#_Toc153014965)

[Hình 1. 10 Biểu đồ use case chức năng sửa dự án 42](#_Toc153014966)

[Hình 1. 11 Biểu đồ use case chức năng xóa dự án 43](#_Toc153014967)

[Hình 1. 12 Biểu đồ chức năng tìm kiếm dự án 44](#_Toc153014968)

[Hình 1. 13 Biểu đồ use case chức năng quản lý người dùng 45](#_Toc153014969)

[Hình 1. 14 Biểu đồ use case chức năng thêm người dùng 45](#_Toc153014970)

[Hình 1. 15 Biểu đồ use case chức năng sửa người dùng 46](#_Toc153014971)

[Hình 1. 16 Biểu đồ use case chức năng xóa người dùng 47](#_Toc153014972)

[Hình 1. 17 Biểu đồ chức năng tìm kiếm người dùng 48](#_Toc153014973)

[Hình 1. 18 Biểu đồ use case chức năng quản lý đội nhóm 49](#_Toc153014974)

[Hình 1. 19 Biểu đồ use case thêm đội nhóm 49](#_Toc153014975)

[Hình 1. 20 Biểu đồ use case chức năng sửa đội nhóm 50](#_Toc153014976)

[Hình 1. 21 Biểu đồ use case chức năng xóa đội nhóm 51](#_Toc153014977)

[Hình 1. 22 Biểu đồ chức năng tìm kiếm đội nhóm 52](#_Toc153014978)

[Hình 1. 23 Biểu đồ use case phân rã tài khoản 55](#_Toc153014979)

[Hình 1. 24 Biểu đồ use case phân rã công việc và nhiệm vụ 56](#_Toc153014980)

[Hình 1. 25 Biểu đồ chức năng quản trị dự án 57](#_Toc153014981)

[Hình 1. 26 Biểu đồ lớp thực thể hệ thống 59](#_Toc153014982)

[Hình 1. 27 Biểu đồ VOPC Đăng nhập 65](#_Toc153014983)

[Hình 1. 28 Biểu đồ VOPC Dự án 65](#_Toc153014984)

[Hình 1. 29 Biểu đồ VOPC Thêm dự án 66](#_Toc153014985)

[Hình 1. 30 Biểu đồ VOPC Sửa dự án 66](#_Toc153014986)

[Hình 1. 31 Biểu đồ VOPC Quản lý công việc 67](#_Toc153014987)

[Hình 1. 32 Biểu đồ VOPC Thêm công việc 67](#_Toc153014988)

[Hình 1. 33 Biểu đồ VOPC Sửa công việc 68](#_Toc153014989)

[Hình 1. 34 Biểu đồ VOPC Xóa công việc 68](#_Toc153014990)

[Hình 1. 35 Biểu đồ tuần tự cho use case đăng nhập 69](#_Toc153014991)

[Hình 1. 36 Biểu đồ tuần tự cho chức năng xem thông tin dự án 69](#_Toc153014992)

[Hình 1. 37 Biểu đồ tuần tự cho use case thêm dự án 70](#_Toc153014993)

[Hình 1. 38 Biểu đồ tuần tự cho use case sửa dự án 70](#_Toc153014994)

[Hình 1. 39 Biểu đồ tuần tự cho use case xóa dự án 71](#_Toc153014995)

[Hình 1. 40 Biểu đồ lớp chi tiết cho use case đăng nhập 72](#_Toc153014996)

[Hình 1. 41 Biểu đồ lớp chi tiết cho use case thêm dự án 72](#_Toc153014997)

[Hình 1. 42 Biểu đồ lớp chi tiết cho use case sửa dự án 73](#_Toc153014998)

[Hình 1. 43 Biểu đồ lớp chi tiết cho use case xóa dự án 73](#_Toc153014999)

[Hình 1. 44 Biểu đồ lớp chi tiết cho use case xem thông tin chi tiết dự án 74](#_Toc153015000)

# TỔNG QUAN VỀ ĐỀ TÀI

1.1. Lý do chọn đề tài

Thế kỷ 21 đã chứng kiến sự bùng nổ của Công nghệ thông tin (CNTT), làm thay đổi bản chất của xã hội kinh tế. Trong bối cảnh này, phát triển hệ thống website trên toàn cầu đã đóng vai trò quan trọng trong việc kết nối con người và biến nền tri thức trở nên gần gũi hơn. Website không chỉ là nguồn giải trí mà còn trở thành nền tảng quan trọng cho quảng cáo, thương mại và quản lý công việc. Hiện nay, việc quản lý công việc trong doanh nghiệp Việt Nam đang đối mặt với nhiều thách thức do sự phân mảnh và rời rạc trong quá trình làm việc. Để giải quyết vấn đề này, em đề xuất xây dựng một trang website giới thiệu về phần mềm quản lý dự án Snap Work. Snap Work không chỉ đơn giản là một công cụ quản lý công việc mà còn là một đối tác đắc lực, giúp doanh nghiệp xây dựng và phân công công việc một cách nhanh chóng, theo dõi tiến độ một cách chính xác và kịp thời, từ đó tối ưu hoá hiệu suất làm việc của nhân viên. Trước đây, doanh nghiệp Việt Nam thường hoạt động không có quy trình hàng loạt, dẫn đến sự nội bộ phân mảnh và khó kiểm soát. Việc này thường diễn ra thủ công và không đảm bảo sự giao tiếp hiệu quả giữa các phòng ban. Snap Work ra đời để giải quyết những thách thức này, tạo ra môi trường làm việc thống nhất và linh hoạt hơn.

SnapWork không chỉ là một phần mềm quản lý công việc, mà là một giải pháp toàn diện giúp doanh nghiệp tận dụng tối đa tiềm năng của công nghệ website. Với những tính năng nổi bật của mình. SnapWork không chỉ giúp doanh nghiệp Việt Nam nhanh chóng thích ứng với môi trường công nghiệp hiện đại mà còn giúp tối ưu hoá quy trình làm việc, góp phần nâng cao hiệu suất làm việc của mọi nhân viên. Với ưu việt của Snap Work trong quản lý công việc, em tin rằng nó sẽ là công cụ quan trọng giúp doanh nghiệp Việt Nam hiện đại hoá quy trình làm việc và tiến lên một cách mạnh mẽ, đồng thời tạo ra môi trường làm việc tích cực và hiệu quả.

1.2. Mục tiêu của đề tài

1.2.1 Mục tiêu tổng quát

Mục tiêu tổng quát của đề tài là nghiên cứu và giới thiệu phần mềm quản lý công việc SnapWork, nhấn mạnh vai trò quan trọng của công nghệ website trong cải thiện quy trình làm việc và hiệu suất của doanh nghiệp Việt Nam. Thông qua việc phân tích thực tế của môi trường làm việc trước đây và thách thức hiện nay, đề tài sẽ tập trung và những ưu điểm đặc biệt của SnapWork trong việc xây dựng, phân công và theo dõi công việc một cách hiệu quả.

1.2.2 Mục tiêu cụ thể

Đó là lý do tại sao chúng ta tập chung vào nghiên cứu về một công cụ quản lý công việc dự án mạnh mẽ, SnapWork để giúp bạn hiểu rõ hơn về cách nó có thể làm cho công việc của bạn trở nên dễ dàng và hiệu quả hơn :

1. Hiểu Rõ Thách Thức Quản Lý Công Việc Trong Doanh Nghiệp Việt Nam : Đánh giá tình trạng làm việc trước đây để xác định những vấn đề cụ thể và thách thức trong quản lý công việc.

2. Phân tích công nghệ Website và ưu điểm trong quản lý công việc : Xem xét cách công nghệ website, đặc biệt là SnapWork, có thể giải quyết các thách thức trên và mang lại những lợi ích đặc biệt trong quản lý công việc.

3. Giới thiệu SnapWork và các tính năng quan trọng : Mô tả chi tiết về SnapWork, từ cách nó hoạt động đến những tính năng đặc biệt giúp doanh nghiệp tối ưu hoá quy trình làm việc và giao tiếp nội bộ.

4. Tích hợp SnapWork vào môi trường doanh nghiệp Việt Nam : Đề xuất cách doanh nghiệp Việt Nam có thể tích hợp SnapWork và hệ thống làm việc hiện tạo và ứng dụng nó một cách linh hoạt và hiệu quả.

5. Phản hồi và hướng phát triển : Thu nhập ý kiến phản hồi từ người sử dụng về trải nghiệm của họ với SnapWork và đề xuất các hướng phát triển tương lai để đáp ứng ngày càng nhiều nhu cầu trong quản lý công việc.

6. Mục tiêu cuối cùng là tạo ra một phần mềm công việc chứng mình cần thiết và hiệu quả của SnapWork trong việc cải thiện quản lý công việc và góp phần nâng cao hiệu suất làm việc trong môi trường doanh nghiệp Việt Nam.

1.3. Giới hạn và phạm vi của đề tài

1.3.1 Đối tượng nghiên cứu

* Đối tượng nghiên cứu : Nhóm quản lý dự án của công ty Viettel SoftWare Services sử dụng Jira.
* Chức vụ vai trò :
* Người quản lý dự án (Project Managers) : Định rõ nhiệm vụ và tiến độ của các công việc trong dự án thông qua Jira.
* Nhân viên theo dõi công việc (Task Trackers) : Sử dụng Jira để theo dõi, cập nhật trạng thái và đánh giá hiệu suất công việc hàng ngày.
* Sử dụng Jira cho giám sát công việc hàng ngày :
* Theo dõi tiến độ : Người quản lý dự án theo dõi tiến độ của các công việc hàng ngày thông qua bảng và biểu đồ Gantt trên Jira.
* Xem tổng quan chi tiết dự án : Nhân viên theo dõi bảng công việc có thể xem tổng quan chi tiết của dự án thông qua các bảng điều khiển và dashboard trên Jira.
* Quản lý trạng thái công việc :
* Chỉnh sửa trạng thái : Người dùng có thể dễ dàng cập nhật trạng thái của công việc “Đang tiến hành “ đến “ Hoàn thành“ thông qua Jira.
* Đánh giá hiệu suất : Các thành viên nhóm có thể sử dụng Jira để đánh giá hiệu suất cá nhân và nhóm dự án.
* Tương tác hàng ngày với giao diện Jira :
* Xem danh sách công việc : Người quản lý dự án và nhân viên theo dõi công việc có thể sử dụng giao diện Jira để xem danh sách công việc chi tiết.
* Tương tác thông qua bảng Scrum : Sử dụng bảng Scrum để tương tác trực tiếp với công việc và theo dõi quá trình phát triển.
* Phản hồi và đề xuất :
* Phản hồi hàng ngày : Những người sử dụng có khả năng đánh giá cụ thể về trải nghiệm sử dụng Jira trong việc giám sát công việc hàng ngày.
* Đề xuất cải tiến : Góp ý về cách Jira có thể được cải thiện để tối ưu hoá quy trình giám sát công việc và theo dõi dự án.

Đối tượng nghiên cứu này tập trung vào nhóm quản lý dự án và những người theo dõi công việc hàng ngày tại công ty Viettel SoftWare Services, sử dụng Jira như một công cụ quan trọng trong quy trình giám sát và quản lý dự án.



Hình 1. 1 Công ty Viettel SoftWare Service

1.3.2 Phạm vi nghiên cứu

* Phạm vi không gian: Nhóm phát triển tại công ty Viettel SoftWare Services, bao gồm các lập trình viên và quản lý dự án chịu trách nhiệm cho các dự án công nghệ.
* Phạm vi thời gian: Từ tháng 10 năm 2023.
* Ý nghĩa khoa học và thực tiễn của đề tài: Đề tài đóng góp vào lĩnh vực nghiên cứu về ứng dụng của công nghệ thông tin trong quản lý dự án, tập trung vào vai trò và ảnh hưởng của Jira đối với hiệu quả làm việc trong nhóm phát triển. Nghiên cứu giúp doanh nghiệp, đặc biệt là những doanh nghiệp hoạt động trong lĩnh vực công nghệ, hiểu rõ hơn về cách Jira có thể hỗ trợ quản lý dự án và đưa ra giải quyết định thông minh trong việc triển khai công nghệ mới.

1.4. Nội dung thực hiển

* Thu nhập thông tin khảo sát :
* Tiến hành cuộc khảo sát trước để hiểu rõ về quy trình quản lý dự án trước khi triển khai SnapWork.
* Xác định mức độ sử dụng công nghệ và các thách thức mà nhóm phát triển đối mặt.
* Quan sát và thu nhập dữ liệu :
* Quan sát trực tiếp hoạt động sử dụng Jira trong quy trình làm việc hàng ngày.
* Thu nhập dữ liệu về sự thay đổi trong quy trình làm việc, thời gian hoàn thành công việc, và tương tác giữa các thành viên nhóm.
* Phân tích dữ liệu và đánh giá hiệu quả :
* Phân tích dữ liệu thu nhập được để đánh giá sự ảnh hưởng của Jira đối với hiệu quả quản lý dự án.
* So sánh các thước đo trước và sau triển khai để đánh giá sự cải thiện.
* Phản hồi từ người dùng và đề xuất cải tiến :
* Thu nhập ý kiến từ người sử dụng về trải nghiệm của họ với Jira.
* Dựa trên phản hồi, đề xuất cải tiến để tối ưu hoá việc sử dụng SnapWork trong môi trường làm việc cụ thể của nhóm phát triển.
* Đánh giá tình ứng dụng và mở rộng :
* Đánh giá tính ứng dụng của kết quả nghiên cứu cho các doanh nghiệp và nhóm phát triển khác.
* Nếu cần, xem xét khả năng mở rộng sự triển khai của SnapWork vào các phòng ban hoặc dự án khác trong công ty.

1.5. Phương pháp tiếp cận

* Tổng hợp thông tin từ các nguồn tài liệu, sách báo và nghiên cứu trước đây về công nghệ quản lý dự án, ưu điểm của SnapWork, và các vấn đề phổ biến trong quản lý công việc dự án.
* Khảo sát tình hình hiện tại :
* Tiến hành cuộc khảo sát và phỏng vấn nhóm phát triển tại công ty Viettel SoftWare Services để đánh giá tình hình hiện tại của quy trình quản lý dự án và nhận định các thách thức.
* Quan sát và ghi chú :
* Quan sát trực tiếp các hoạt động sử dụng SnapWork trong quy trình làm việc hàng ngày.
* Ghi chú về các thay đổi, thách thức xuất hiện, và hiệu suất công việc.
* Phân tích thống kê :
* Phân tích dữ liệu thu nhập được từ sự thay đổi trong quy trình làm việc và hiệu suất trước và sau khi triển khai Snap Work.
* Sử dụng các phương pháp thống kê để xác định sự cải thiện có ý nghĩa.
* Thu nhập phản hồi từ người dùng :
* Tổ chức phiên đối thoại và thu nhập phản hồi từ nhóm phát triển về trải nghiệm của họ với việc sử dụng SnapWork.
* Ghi chú về ý kiến, đánh giá và đề xuất cải tiến.

# CƠ SỞ LÝ THUYẾT

2.1. Quy trình phát triển phần mềm

Quy trình xây dựng phần mềm bao gồm tập hợp các thao tác và kết quả tương quan sử dụng trong việc phát triển để tạo ra phần mềm quản lý công việc Snap. Mỗi một phần mềm lại có đặc điểm và yêu cầu khác nhau, tuy nhiên tất cả đều cần phải trải qua các bước sau:

**Bước 1: Phân tích yêu cầu**

Phân tích yêu cầu là công việc bao gồm các tác vụ xác định yêu cầu cho một hệ thống mới hoặc được thay đổi dựa trên cơ sở là các nhu cầu trong quá trình sử dụng. Việc phân tích yêu cầu có ý nghĩa quan trọng đối với thành công của một dự án. Các yêu cầu phải có tính đo được, kiểm thử được, có liên quan đến các nhu cầu hoặc cơ hội doanh nghiệp đã được xác định, và phải được định nghĩa ở mức độ chi tiết đủ cho việc thiết kế hệ thống.

**Bước 2: Thiết kế phần mềm**

Là một quá trình giải quyết vấn đề và lập kế hoạch cho một giải pháp phần mềm. Sau khi các mục đích và các đặc điểm kĩ thật của phần mềm được giải quyết, lập trình viên sẽ thiết kế hoặc thuê người thiết kế để phát triển một kế hoạch cho giải pháp phần mềm. Nó bao gồm các thành phần cấp thấp, các vấn đề thuật toán cũng như một khung nhìn kiến trúc. Thiết kế chức năng, cơ sở dữ liệu và giao diện.

**Bước 3: Lập trình máy tính**

Lập trình máy tính (gọi tắt là lập trình) là kỹ thuật cài đặt một hoặc nhiều thuật toán trừu tượng có liên quan với nhau bằng một hoặc nhiều ngôn ngữ lập trình để tạo ra một chương trình máy tính có các thành tố nghệ thuật, khoa học, toán học, kỹ nghệ. Các ngôn ngữ lập trình khác nhau hỗ trợ các phong cách lập trình khác nhau. Một phần của công việc lập trình là việc lựa chọn một trong những ngôn ngữ phù hợp nhất với các bài toán cần giải quyết. Các ngôn ngữ lập trình khác nhau đòi hỏi lập trình viên phải xử lý các chi tiết ở mức độ khác nhau khi cài đặt các thuật toán. Sự thống nhất trong các cách xử lý sẽ tạo thuận lợi cho việc lập trình và hiệu quả của chương trình.

**Bước 4: Kiểm thử phần mềm**

Kiểm thử phần mềm là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng hoặc dịch vụ được kiểm thử. Kiểm thử có thể cung cấp cho doanh nghiệp một quan điểm, một cách nhìn độc đáo về phần mềm để từ đó đánh giá và thấu hiểu được những rủi ro trong quá trình triển khai phần mềm. Tùy thuộc vào từng phương pháp, việc kiểm thử có thể được thực hiện bất cứ lúc nào trong quá trình phát triển phần mềm. Theo truyền thống thì các nỗ lực kiểm thử được tiến hành sau khi các yêu cầu được xác định và việc lập trình được hoàn tất trong phương pháp phát triển “Agile” thì việc kiểm thử được tiến hành liên tục trong suốt quá trình xây dựng phần mềm. Như vậy, mỗi một phương pháp kiểm thử bị chi phối theo một quy trình phát triển phần mềm nhất định.

**Bước 5: Triển khai phần mềm**

Sau khi phần mềm được kiểm thử và khắc phục những sai sốt sẽ được triển khai đưa vào sử dụng trong thực tế. Đối với những phần mềm thiết kế theo thỏa thuận với khách hàng, việc triển khai đơn giản chỉ là hướng dẫn cho khách hàng cách sử dụng đạt hiệu quả cao. Với những phần mềm mang tính thông dụng, việc triển khai còn qua các chương trình giới thiệu và đưa giày thể thao ra thị trường. Trong quá trình triển khai cũng luôn đánh giá hiệu quả sử dụng của phần mềm, xét những nhược điểm để lên kế hoạch thiết kế phần mềm hiệu quả hơn.

**Bước 6: Bảo trì phần mềm**

Bảo trì phần mềm bao gồm điều chỉnh các lỗi mà chưa được phát hiện trong các giai đoạn trước chu kỳ sống của phần mềm, nâng cấp tính năng sử dụng và an toàn vận hành của phần mềm. Bảo trì phần mềm có thể chiếm đến 65%-75% công sức chu kỳ sống của một phần mềm. Quá trình phát triển phần mềm bao gồm rất nhiều giai đoạn: thu thập yêu cầu, phân tích, xây dựng, kiểm tra, triển khai và bảo trì phần mềm. Nhiệm vụ của giai đoạn bảo trì phần mềm là giữ cho phần mềm được cập nhật khi môi trường thay đổi và yêu cầu người sử dụng thay đổi. Mỗi một giai đoạn xây dựng phần mềm lại đòi hỏi các kỹ năng phân tích và ứng dụng kiến thức công nghệ khác nhau. Để xây dựng phần mềm thiết thực, mang lại hiệu quả kinh tế cao đòi hỏi con người lập trình viên phải tuân thủ các yêu cầu trong từng giai đoạn thiết kế.

2.2. Phương pháp phân tích thiết kế hướng đối tượng

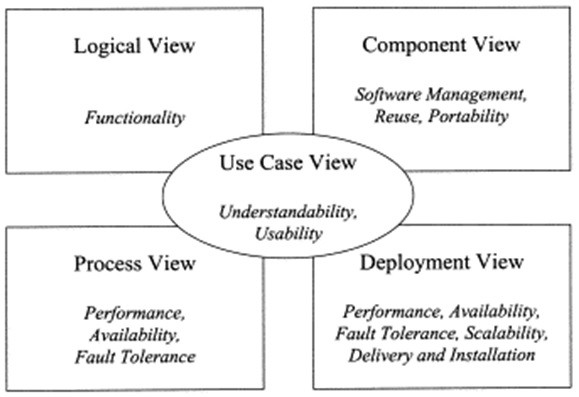
* Khái niệm OOAD (Object Oriented Analysis and Design)

Phân tích và thiết kế hướng đối tượng là một kỹ thuật tiếp cận phổ biến dùng để phân tích, thiết kế một ứng dụng, hệ thống. Nó dựa trên bộ các nguyên tắc chung, đó là một tập các hướng dẫn để giúp chúng ta tránh khỏi một thiết kế xấu. 5 nguyên tắc SOLID trong thiết kế hướng đối tượng:

* Một lớp chỉ nên có một lý do để thay đổi, tức là một lớp chỉ nên xử lý một chức năng đơn lẻ, duy nhất thôi. Nếu đặt nhiều chức năng vào trong một lớp sẽ dẫn đến sự phụ thuộc giữa các chức năng với nhau và mặc dù sau đó ta chỉ thay đổi ở một chức năng thì cũng phá vỡ các chức năng còn lại.
* Các lớp, module, chức năng nên dễ dàng Mở (Open) cho việc mở rộng (thêm chức năng mới) và Đóng (Close) cho việc thay đổi.
* Lớp dẫn xuất phải có khả năng thay thế được lớp cha của nó.
* Chương trình không nên buộc phải cài đặt một interface mà nó không sử dụng đến.
* Các module cấp cao không nên phụ thuộc vào các module cấp thấp. Cả hai nên phụ thuộc thông qua lớp trừu tượng. Lớp trừu tượng không nên phụ thuộc vào chi tiết. Chi tiết nên phụ thuộc vào trừu tượng.
* **Khái niệm UML**

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư v.v..

* Tại sao lại là OOAD và UML? OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiển. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.
* OOAD sử dụng UML
* UML sử dụng để vẽ cho nhiều lĩnh vực khác nhau như phần mềm, cơ khí, xây dựng v… trong phạm vi các bài viết này chúng ta chỉ nghiên cứu cách sử dụng UML cho phân tích và thiết kế hướng đối tượng trong ngành phần mềm. OOAD sử dụng UML bao gồm các thành phần sau:
* View (góc nhìn)
* Mỗi góc nhìn như thầy bói xem voi, nó không thể hiển hết hệ thống nhưng thể hiện rõ hệ thống ở một khía cạnh. Chính vì thế trong xây dựng có bản vẽ kiến trúc (nhìn về mặt kiến trúc), bản vẽ kết cấu (nhìn về mặt kết cấu), bản vẽ thi công (nhìn về mặt thi công). Trong phần mềm cũng như vậy, OOAD sử dụng UML có các góc nhìn sau:



Trong đó,

* Use Case View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.
* Logical View: cung cấp góc nhìn về cấu trúc hệ thống, nó được tổ chức như thế nào. Bên trong nó có gì.
* Process View: cung cấp góc nhìn động về hệ thống, các thành phần trong hệ thống tương tác với nhau như thế nào.
* Component View: Cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.
* Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Trong hình trên chúng ta thấy góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của Use Case View.

* Diagram (bản vẽ)

Diagram chúng ta có thể dịch là sơ đồ. Tuy nhiên ở đây chúng ta sử dụng từ bản vẽ cho dễ hình dung. Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.

* Trong đó :
* Use Case Diagram: bản vẽ mô tả về ca sử dụng của hệ thống. Bản vẽ này sẽ giúp chúng ta biết được ai sử dụng hệ thống, hệ thống có những chức năng gì. Lập được bản vẽ này chúng ta sẽ hiểu được yêu cầu của hệ thống cần xây dựng.
* Class Diagram: bản vẽ này mô tả cấu trúc của hệ thống, tức hệ thống được cấu tạo từ những thành phần nào. Nó mô tả khía cạnh tĩnh của hệ thống.
* Object Diagram: Tương tự như Class Diagram nhưng nó mô tả đến đối tượng thay vì lớp (Class).
* Sequence Diagram: là bản vẽ mô tả sự tương tác của các đối tượng trong hệ thống với nhau được mô tả tuần tự các bước tương tác theo thời gian.
* Collaboration Diagram: tương tự như sequence Diagram nhưng nhấn mạnh về sự tương tác thay vì tuần tự theo thời gian.
* State Diagram: bản vẽ mô tả sự thay đổi trạng thái của một đối tượng. Nó được dùng để theo dõi các đối tượng có trạng thái thay đổi nhiều trong hệ thống.
* Activity Diagram: bản vẽ mô tả các hoạt động của đối tượng, thường được sử dụng để hiểu về nghiệp vụ của hệ thống.
* Component Diagram: bản vẽ mô tả về việc bố trí các thành phần của hệ thống cũng như việc sử dụng các thành phần đó.
* Deployment Diagram: bản vẽ mô tả việc triển khai của hệ thống như việc kết nối, cài đặt, hiệu năng của hệ thống v.v…
* Notations (ký hiệu)

Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Chúng ta phải biết từ vựng thì mới ghép thành câu, thành bài được. Chúng ta sẽ tìm hiểu kỹ các notations trong từng bản vẽ sau này. Dưới đây là vài ví dụ về notation.

* Mechanisms (qui tắc, cơ chế)

Mechanisms là các qui tắc để lập nên bản vẽ, mỗi bản vẽ có qui tắc riêng và chúng ta phải nắm được để tạo nên các bản vẽ thiết kế đúng. Các qui tắc này chúng ta sẽ bàn kỹ trong các bài về các bản vẽ.

2.3. Hệ quản trị cơ sở dữ liệu MySQL

2.3.1 Hệ quản trị cơ sở dữ liệu

* **Hệ quản trị cơ sở dữ liệu quan hệ**

Hệ quản trị cơ sở dữ liệu dựa trên mô hình quan hệ để làm việc với dữ liệu. Mô hình quan hệ mô hình bất cứ thông tin nào được lưu trữ bằng việc xác định chúng như các thực thể liên quan tới các thuộc tính của các bảng. Những loại hệ quản trị cơ sở dữ liệu đòi hỏi các cấu trúc (ví dụ như một bảng) phải được xác định để chứa và làm việc với dữ liệu. Với mỗi bảng, mỗi cột (ví dụ như một thuộc tính) giữ một kiểu dữ liệu khác của thông tin. Mỗi bản ghi trong cơ sở dữ được định nghĩa duy nhất với các key và tương ứng với một hàng trong một bảng. Với thuộc tính của các hàng được thể hiển như một cột trong một bảng.

## Các quan hệ và các dạng dữ liệu

Các quan hệ có thể được coi như là một tập hợp có chứa một loạt các thuộc tính mà đại diện cho cơ sở dữ liệu và thông tin được lưu trữ. Khi đang định nghĩa một bảng để thêm các bản ghi thi mỗi phần tử trong một bản ghi phải tương ứng với kiểu dữ liệu được định nghĩa (như kiểu integer, date,...). Hệ quản trị cơ sở dữ liệu quan hệ khác nhau triển khai các loại dữ liệu khác nhau - đó không phải là luôn luôn trực tiếp hoán đổi cho nhau.

## Các cơ sở dữ liệu quan hệ phổ biến và quan trọng

* **MySQL:** Một hệ quản trị cơ sở dữ liệu quan hệ mạnh mẽ và phổ biến hiện nay.
* **PostgreSQL:** Với tuyên ngôn rằng "Cơ sở dữ liệu mã nguồn mở tiên tiến nhất thế giới".

Sau đây, chúng ta sẽ tìm hiểu chi tiết về 2 hệ quản trị cơ sở dữ liệu quan hệ ở trên.

2.3.2 MySQL

MySQL là một trong những cơ sở dữ liệu có khả năng mở rộng phổ biến nhất hiển nay. Nó giàu các tính năng, là một giày thể thao mã nguồn mở mạnh mẽ trên các website và các ứng dụng online. Việc bắt đầu với MySQL là cực kì dễ dàng và các nhà phát triển dễ dàng tiếp cận với một lượng lớn các thông tin về cơ sở dữ liệu trên internet.

* **Dạng dữ liệu :**

| **Kiểu dữ liệu** | **Kích thước** |
| --- | --- |
| TINYINT | -128 - 127 |
| SMALLINT | -32768 - 32767 |
| MEDIUMINT | -8388608 - 8388607 |
| INT | -2147483648 - 2147483647 |
| BIGINT | - 2^64 − 1 - 26^3 − 1 |
| FLOAT | 4 bytes |
| DOUBLE | 8 bytes |
| DECIMAL, NUMERIC | 4 bytes |
| DATE | 3 bytes |
| DATETIME | 8 bytes |
| TIMESTAMP | 4 bytes |
| TIME | 3 bytes |
| YEAR | 1 byte |
| CHAR | M × w bytes |
| VARCHAR | L + 2 bytes |
| TINYBLOB, TINYTEXT | L + 1 bytes |
| BLOB, TEXT | L + 2 bytes |
| MEDIUMBLOB, MEDIUMTEXT | L + 3 bytes |
| LONGBLOB, LONGTEXT | L + 4 bytes |
| ENUM | An enumeration |
| SET | A set |

## Ưu điểm của SQL :

* **Dễ dàng sử dụng: My**SQL có thể dễ dàng cài đặt. Với các công cụ bên thứ 3 làm cho nó càng dễ đơn giản hơn để có thể sử dụng.
* **Giàu tính năng: My**SQL hỗ trợ rất nhiều chức năng MySQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ-cả trực tiếp lẫn gián tiếp.
* **Bảo mật:** Có rất nhiều tính năng bảo mật, một số ở cấp cao đều được xây dựng trong MySQL.
* **Khả năng mở rộng và mạnh mẽ: My**SQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.
* **Nhanh:** Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.

## Nhược điểm của SQL :

* **Giới hạn:** Theo thiết kế, MySQL không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vào ứng dụng có thể cần.
* **Độ tin cậy:** Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,...) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.
* **Sự phát triển chậm:** Mặc dù MySQL vẫn là công nghệ mã nguồn mở phổ biến nhưng có những phàn nàn liên quan đến quá trình phát triển kể từ khi nó được mua lại.

## Khi nào nên sử dụng MySQL?

* **Các hoạt động phân tán:** Khi bạn cần nhiều hơn những gì MySQ có thể cung cấp, bao gồm cả MySQL để ngăn xếp triển khai của bạn, giống như bất kỳ máy chủ cơ sở dữ liệu độc lập nào mà mang lại nhiều tự do hoạt động cùng với một số tính năng tiên tiến.
* **Tính bảo mật cao:** Tính năng bảo mật của MySQL cung cấp sự bảo vệ đáng tin cậy cho các dữ liệu truy nhập (và sử dụng) theo một cách đơn giản.
* **Các ứng dụng web:** Phần lớn các websites có thể hoạt động đơn giản trên MySQL. Sự linh hoạt và có phần mở rộng của các công cụ là dễ dàng sử dụng và dễ dàng quản lý.
* **Các giải pháp tùy chỉnh:** Nếu bạn đang làm trên một giải pháp rất cụ thể và có tính tùy chỉnh, thì MySQL có thể thực hiện theo các quy tắc của bạn nhờ vào các thiết lập cấu hình phong phú và các phương thức hoạt động.

## Khi nào không nên sử dụng MySQL?

* **Tuân thủ theo MySQL:** Bởi vì MySQL không thi hành theo đầy đủ chuẩn của MySQL nên công cụ này không hoàn toàn tuân thủ theo MySQL.
* **Sự đồng thời:** Mặc dù MySQL và một số công cụ thực hiện tốt việc đọc nhưng đồng thời việc đọc-viết có thể có vấn đề.
* **Thiếu tính năng:** Một lần nữa, tùy thuộc vào sự lựa chọn của các cơ sở dữ liệu, MySQL có thể thiếu một số tính năng, chẳng hạn như tìm kiếm văn bản đầy đủ.

2.4. Thiết kế giao diện website

* Ngôn ngữ HTML
* HTML là một loại ngôn ngữ đánh dấu siêu văn bản, thế nên các chức năng của nó cũng xoay quanh yếu tố này. Cụ thể, HTML giúp cấu thành các cấu trúc cơ bản trên một website (chia khung sườn, bố cục các thành phần trang web) và góp phần hỗ trợ khai báo các tập tin kỹ thuật số như video, nhạc, hình ảnh.
* Các trang HTML được quy định bằng các thẻ tag. Những thẻ này được chứa trong các dấu ngoặc đơn dạng: <tên thẻ> . Trừ một vài thẻ đặc biệt, hầu hết các thẻ cơ bản đều có các thẻ đóng tương ứng với nó. Ví dụ, thẻ <html> có thẻ đóng tương ứng là </html>, thẻ <body> có thẻ đóng tương ứng là </body>... Dưới đây là các mẫu tag thường gặp trong HTML.
* Một số thẻ trong html:
* <html>, <head>, <body>, <title>, <h1> - <h6>, <div>, <span>, <p>, …
* Ngôn ngữ CSS
* CSS (Cascading Style Sheets) là mã bạn sử dụng để tạo kiểu cho trang web của mình, Giống như HTML, CSS không thực sự là một ngôn ngữ lập trình. Nó không phải là một ngôn ngữ đánh dấu - đó là một ngôn ngữ định kiểu. Điều này có nghĩa là nó cho phép bạn áp dụng kiểu có chọn lọc cho các phần tử trong tài liệu HTML.
* Cơ cấu bộ quy tắc CSS:
* **Selector**: Tên phần tử HTML bắt đầu của bộ quy tắc. Nó chọn (các) phần tử được tạo kiểu (trong trường hợp này là phần tử p). Để tạo kiểu cho một phần tử khác, chỉ cần thay đổi bộ chọn.
* **Declaration**: Một quy tắc duy nhất như: color: red; xác định **thuộc tính** phần tử mà bạn muốn tạo nên.
* **Properties**: Những cách mà bạn có thể tạo kiểu cho một phần tử HTML. (Trong trường hợp này, color là một thuộc tính của phần tử <p>.) Trong CSS, bạn chọn thuộc tính nào bạn muốn tác động trong quy tắc của mình.
* **Giá trị thuộc tính**: Ở bên phải của thuộc tính sau dấu hai chấm(:), chúng ta có **giá trị thuộc tính**, mà chọn một trong số nhiều lần xuất hiện có thể cho một thuộc tính cụ thể (color có rất nhiều giá trị ngoài red).
* Chọn nhiều phần tử: Bạn cũng có thể chọn nhiều kiểu phần tử và áp dụng một quy tắc duy nhất được đặt cho tất cả các yếu tố đó. Bao gồm nhiều bộ chọn được phân biệt bởi dấu phẩy (,).

2.5. Lập trình phía front-end

* Javascript:

Javascript (*viết tắt là JS*) là một ngôn ngữ lập trình kịch bản phía máy khách, mã lệnh được thực thi bởi trình duyệt người dùng .

Javascript cho phép bạn có thể tạo ra rất nhiều tính năng tương tác cho trang web. Giúp người dùng dễ dàng sử dụng website hơn.

* Angular :

Angular là một javascript framework do google phát triển để xây dựng các Single Page Application (SPA) bằng JavaScript, HTML và TypeScript. Angular cung cấp các tính năng tích hợp cho animation, http service và có các tính năng như auto-complete, navigation, toolbar, menus, … Code được viết bằng TypeScript, biên dịch thành JavaScript và hiển thị tương tự trong trình duyệt.

2.6. Lập trình phía back-end

* **C Sharp :**

C Sharp (C#) là một ngôn ngữ lập trình hiện đại, hướng đối tượng và an toàn kiểu. C Sharp cho phép các nhà phát triển xây dựng nhiều loại ứng dụng an toàn và mạnh mẽ chạy trong .NET . C Sharp có nguồn gốc từ họ ngôn ngữ C và quen thuộc ngày với các lập trình viên C, C ++, Java và JavaScript.

C Sharp là ngôn ngữ lập trình hướng đối tượng . C Sharp cung cấp các cấu trúc ngôn ngữ để hỗ trợ trực tiếp các khái niệm này, biến C Sharp trở thành ngôn ngữ tự nhiên để tạo và sử dụng các thành phần phần mềm. Kể từ khi ra đời, C Sharp đã bổ xung thêm các tính năng để hỗ trợ khối lượng công việc mới và các phương pháp thiết kế phần mềm mới nổi. Các chương trình C# chạy trên .NET, một hệ thống thực thi ảo được gọi là thời gian chạy ngôn ngữ chung (CLR) và một tập hợp các thư viện lớp. CLR là việc Microsoft triển khai cơ sở hạ tầng ngôn ngữ chung (CLI), một tiêu chuẩn quốc tế. CLI là cơ sở để tạo môi trường thực thi và phát triển trong đó các ngôn ngữ và thư viện hoạt động liền mạch với nhau.

* **ASP .NET :**

ASP.NET Core là một tập hợp các thư viện chuẩn như một framework để xây dựng ứng dụng web. ASP.NET Core không phải là phiên bản tiếp theo của ASP.NET. Nó là một cái tên mới được xây dựng từ đầu. Nó có một sự thay đổi lớn về kiến trúc và kết quả là nó gọn hơn, phân chia module tốt hơn. ASP.NET Core có thể chạy trên cả .NET Core hoặc full .NET Framework.

* Các đặc tính quan trọng của ASP .NET Core :
* Bạn có thể xây dựng và chạy ứng dụng ASP .NET đa nền tảng trên Windows, Mac và Linux.
* ASP .NET Core hợp nhất ASP .NET MVC và ASP .NET Web API .
* Có thể host trên IIS hoặc tự host.
* Có sẵn Dependency Injection.
* Dễ dàng tích hợp với các framework frontend như Angular, Knockout...
* Hỗ trợ cấu hình cho nhiều môi trường.
* Cơ chế HTTP Request mới.
* Hỗ trợ quản lý phiên bản.
* Dùng chung toàn bộ Nuget Package.
* **NodeJS :**

NodeJS là một môi trường runtime chạy JavaScript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phát triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ.

* **Ưu điểm :**
* IO hướng sự kiện không đồng bộ, cho phép xử lý nhiều yêu cầu đồng thời.
* Sử dụng JavaScript – một ngôn ngữ dễ học.
* Chia sẻ cùng node ở cả phía client và server.
* NPM (Node Package Manager) và module Node đang ngày càng phát triển mạnh mẽ.
* Cộng đồng hỗ trợ tích cực.
* Cho phép stream các file có kích thước lớn.
* **Nhược điểm :**
* Không có khả năng mở rộng, vì vậy không thể tận dụng lợi thế mô hình đa lõi trong các phần cứng cấp server hiện nay.
* Khó thao tác với cơ sở dữ liệu quan hệ.
* Mỗi callback sẽ đi kèm với rất nhiều callback lồng nhau khác.
* Cần có kiến thức tốt về JavaScript.
* Không phù hợp với các tác vụ đòi hỏi nhiều CPU.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

* 1. Phát biểu bài toán

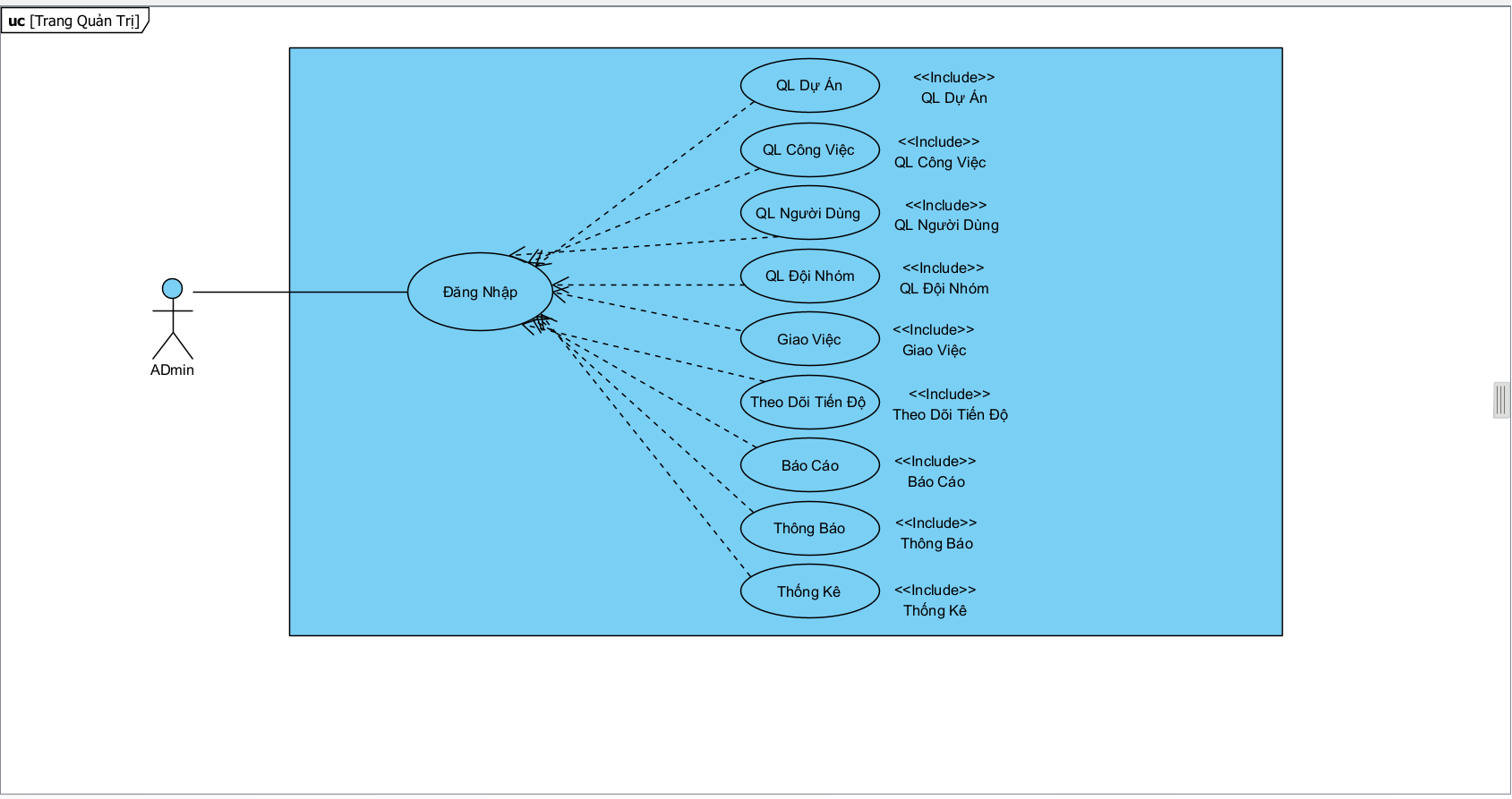
Quá trình phát triển phần mềm quản lý công việc Snap Work, một công cụ quan trọng để tối ưu hóa quá trình làm việc trong tổ chức. Dưới đây là yêu cầu chi tiết của hệ thống quản lý công việc :

* **Quản lý dự án :**
* Thêm dự án : Người dùng có thể thêm mới các dự án vào hệ thống cung cấp.
* Sửa và Xóa dự án : Cung cấp khả năng sửa đổi thông tin hoặc xóa các dự án đã tồn tại trong hệ thống.
* **Quản lý công việc :**
* Tạo công việc : Người dùng có thể tạo công việc mới, cung cấp Mã công việc, Tên công việc, Đơn vị thời gian, Mô tả và Thiết lập trạng thái (đang tiến hành, hoàn thành, chờ xử lý).
* Sửa và xóa công việc : Cho phép sửa thông tin chi tiết hoặc xóa các công việc trong hệ thống.
* **Quản lý người dùng :**
* Thêm người dùng : Người quản trị có thể thêm mới người dùng với Mã người dùng, Tên vai trò (nhân viên, quản trị viên), và trạng thái hoạt động.
* Sửa và Xóa người dùng : Cho phép sửa thông tin chi tiết hoặc xóa người dùng trong hệ thống.
* **Giao việc và theo dõi tiến độ :**
* Giao công việc : Người dùng có thể giao công việc cho người khác, xác định thời gian bắt đầu, thời hạn hoàn thành, và theo dõi trạng thái công việc.
* Theo dõi tiến độ : Người dùng có thể theo dõi tiến độ của công việc, xem công việc đang tiến hành, công việc đã hoàn thành, và công việc chờ xử lý.
* **Báo cáo và thống kê :**
* Báo cáo tiến độ : Hệ thống cung cấp báo cáo về tiến độ công việc dựa trên các tiêu chí như dự án, người dùng hoặc thời gian.
* Thống kê hiệu suất : Người dùng có thể xem thống kê về nguồn lực sử dụng và hiệu suất làm việc của nhóm.
* **Thông báo và nhắc nhở :**
* Thông báo tự động : Hệ thống sẽ tự động gửi thông báo và nhắc nhở về các công việc sắp đến hạn hoặc cần chú ý.
* Tùy chọn thông báo : Người dung có thể tùy chọn cách họ muốn nhận thông báo (email, thông báo trực tiếp trên ứng dụng).
* **Quản lý đội nhóm :**
* Tạo và quản lý đội nhóm : Người quản trị có khả năng tạo mới, chỉnh sửa và xóa đội nhóm. Người dung có thể tham gia và một hoặc nhiều đội nhóm.
* **Hệ thống giao tiếp nội bộ :**
* Tích hợp chức năng bình luận cung cấp môi trường giao tiếp nội bộ giữa các thành viên trong dự án.
  1. Đặc tả yêu cầu phần mềm
     1. Các yêu cầu chức năng
        1. **Chức năng của phân hệ quản trị nội dung**
* Các chức năng của trang quản trị :

|  |  |  |
| --- | --- | --- |
| STT | Chức năng | Mô tả |
| 1 | Quản lý dự án | Người dùng có thể tạo và quản lý các dự án. Mỗi dự án có thông tin như Tên dự án, Mô tả, Ngày bắt đầu, Ngày kết thúc.  Chức năng chính :   * Tạo mới dự án với các thông tin chi tiết * Hiển thị danh sách các dự án đang hoạt động.   Xem thông tin chi tiết của từng dự án. |
| 2 | Quản lý công việc | Công việc được tạo và gán cho thành viên trong dự án. Mỗi công việc có Tiêu đề, Mô tả, Ngày bắt đầu, Ngày kết thúc, Trạng thái.  Chức năng chính :   * Tạo công việc mới và gán cho thành viên. * Xem danh sách công việc theo dự án và thành viên.   Cập nhật trạng thái của công việc (đang thực hiện, đã hoàn thành hoặc chưa bắt đầu). |
| 3 | Quản lý đội người dùng | Người dùng có khả năng đăng nhập để truy cập vào hệ thống. Cung cấp khả năng quản lý và cập nhập thông tin cá nhân của người dùng.  Chức năng chính :  Xem và chỉnh sửa thông tin cá nhân. |
| 4 | Quản lý đội nhóm | Người quản lý có quyền quản lý thành viên trong dự án và gán vai trò cho họ.  Chức năng chính :   * Thêm và xóa thành viên khỏi dự án.   Gán vai trò cho từng thành viên như quản lý dự án, thành viên thực hiện công việc . |
| 5 | Báo cáo và thống kê | Hệ thống cung cấp báo cáo và thống kê về tiến độ, công việc đã hoàn thành và tài nguyên đã sử dụng.  Chức năng chính :   * Tạo báo cáo tổng quan về dự án và công việc.   Xem biểu đồ thống kê về nguồn lực, thời gian và chi phí. |
| 6 | Giao việc và theo dõi tiến độ | Người quản lý có khả năng giao việc cho các thành viên trong dự án và đặt mức độ ưu tiên.  Chức năng chính :   * Gán công việc cho thành viên cụ thể. * Đặt mức độ ưu tiên cho từng công việc.   Theo dõi tiến độ thực hiện công việc. |
| 7 | Hệ thống giao tiếp nội bộ | Các thành viên có khả năng tương tác và giao tiếp trong nội dung công việc.  Chức năng chính :  Bình luận và thảo luận các thông tin dự án. |
| 8 | Thông báo và nhắc nhở | Cung cấp các thông báo và nhắc nhở nội bộ giữa người dùng.  Chức năng chính :  Gửi các thông báo về các sự kiện quan trọng. |

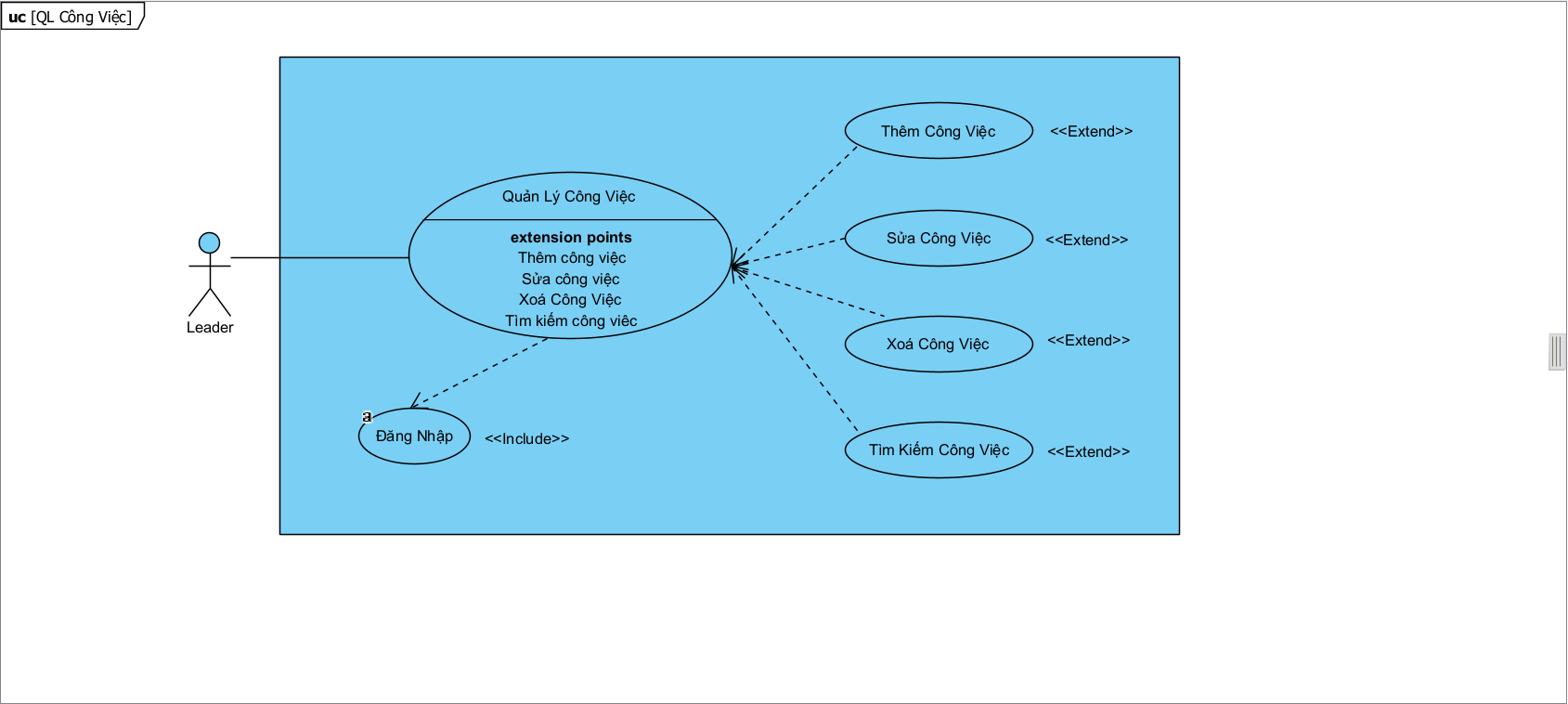
Bảng 1. 1 Chức năng phân hệ của quản trị hệ thống

* **Biểu đồ Use Case tổng quát của trang quản trị**



Hình 1. 2 Biểu đồ Use Case trang quản trị hệ thống tổng quát

* **Các biểu đồ Use Case phân rã của trang quản trị**
* Biểu đồ use case chức năng Quản lý công việc :

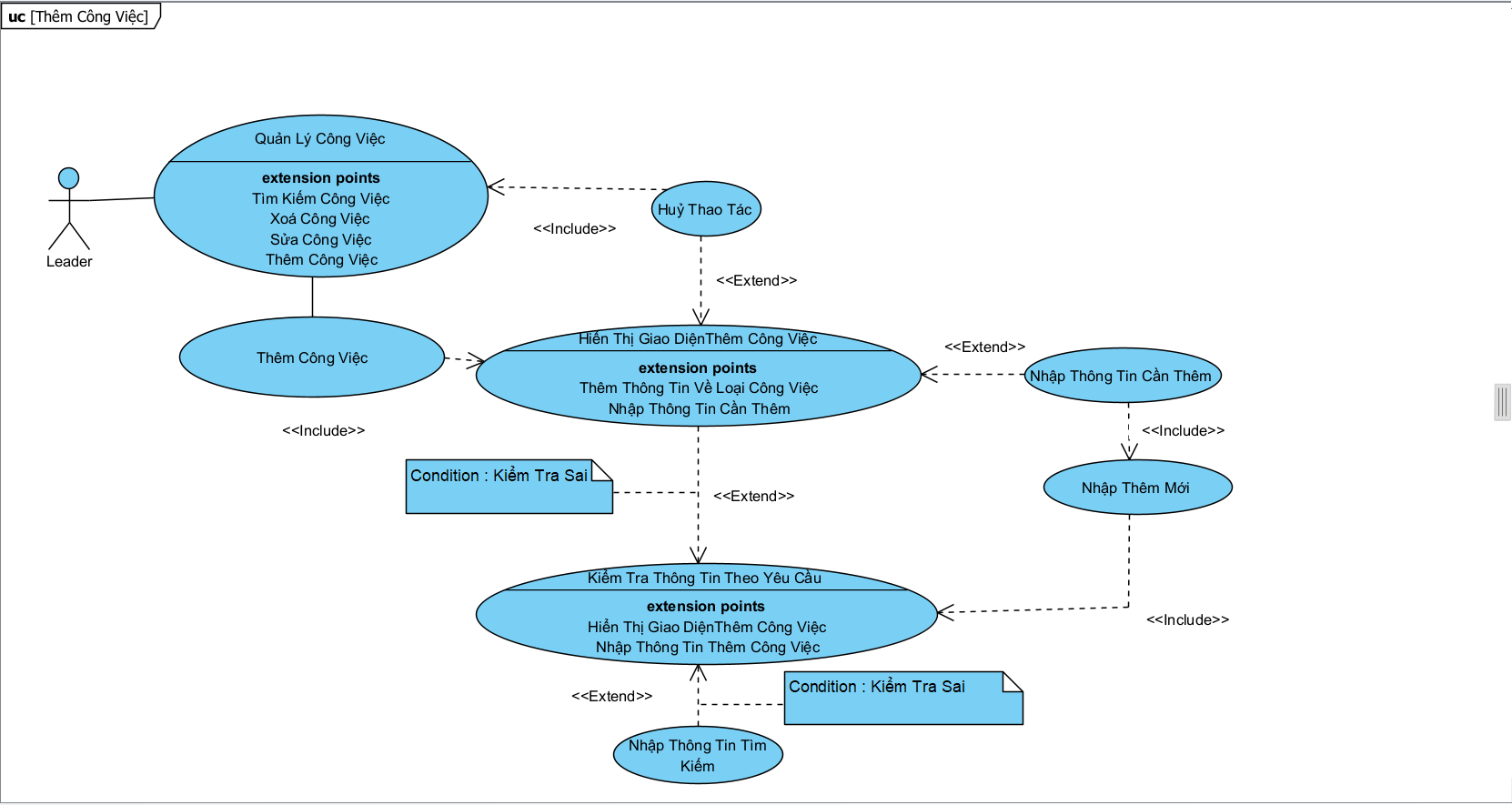


Hình 1. 3 Biểu đồ use case chức năng quản lý công việc

* Luồng sự kiện quản lý công việc :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng yêu cầu chức năng quản lý công việc | Hiển thị giao diện lựa chọn các chức năng quản lý công việc |
| Người dùng yêu cầu chức năng thêm công việc | Hệ thống hiển thị giao diện trang quản lý thêm công việc |
| Người dùng yêu cầu chức năng sửa công việc | Hệ thống hiển thị giao diện quản lý chức năng sửa thông tin công việc |
| Người dùng yêu cầu chức năng xóa công việc | Hệ thống hiển thị giao diện quản lý chức năng xóa thông tin công việc |
| Người dùng yêu cầu chức năng tìm kiếm công việc | Hệ thống hiển thị giao diện quản lý chức năng tìm kiếm thông tin công việc |

* Use case chức năng thêm công việc :

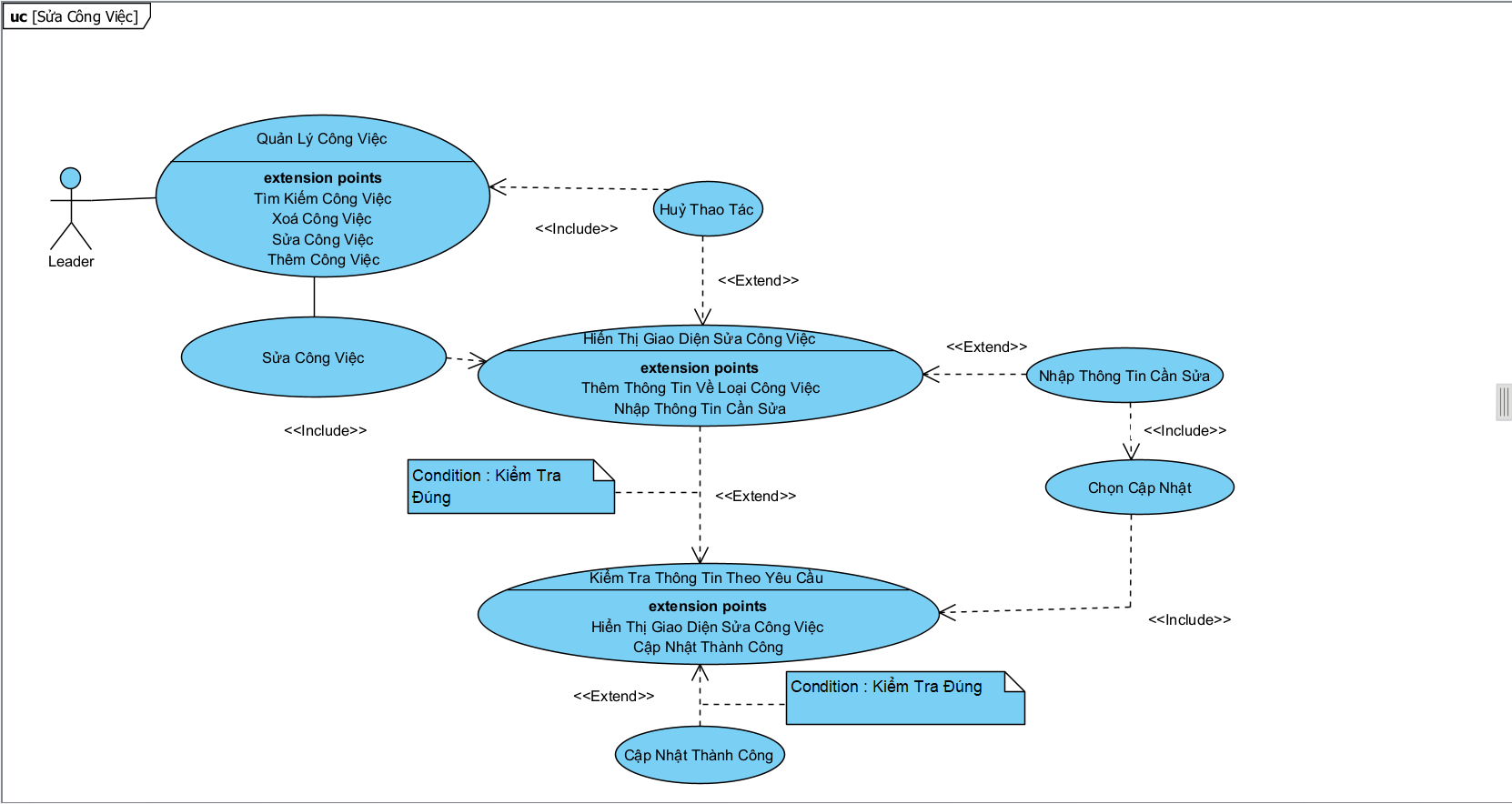


Hình 1. 4 Biểu đồ use case thêm công việc

* Luồng sự kiện thêm công việc :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã công việc | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng nhập tiêu đề công việc | Hệ thống kiểm tra tính hợp lệ của tiêu đề công việc nhập vào |
| Người dùng nhập mô tả công việc |  |
| Người dùng xác nhận thêm thông tin công việc | Hệ thống lưu thông tin công việc thêm mới vào CSDL |

* Use case chức năng sửa công việc :

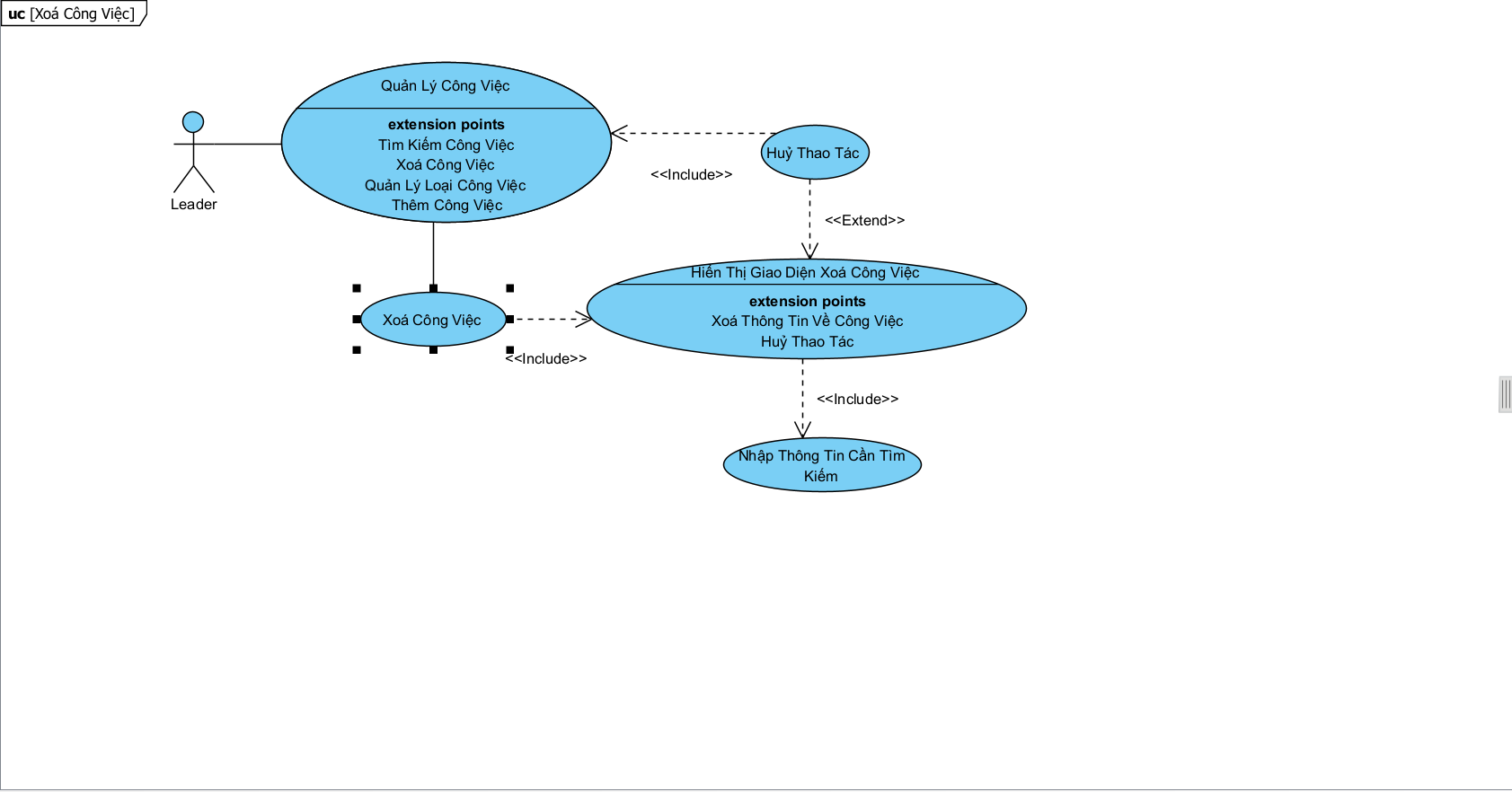


Hình 1. 5 Biểu đồ use case chức năng sửa công việc

* Luồng sự kiện sửa công việc :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã công việc cần sửa | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng nhập tên công việc | Hệ thống kiểm tra tính hợp lệ của tên công việc nhập vào |
| Người dùng nhập mô tả công việc |  |
| Người dùng xác nhận sửa thông tin công việc | Hệ thống lưu thông tin công việc sau khi sửa mới vào CSDL |

* Use case chức năng xóa công việc :

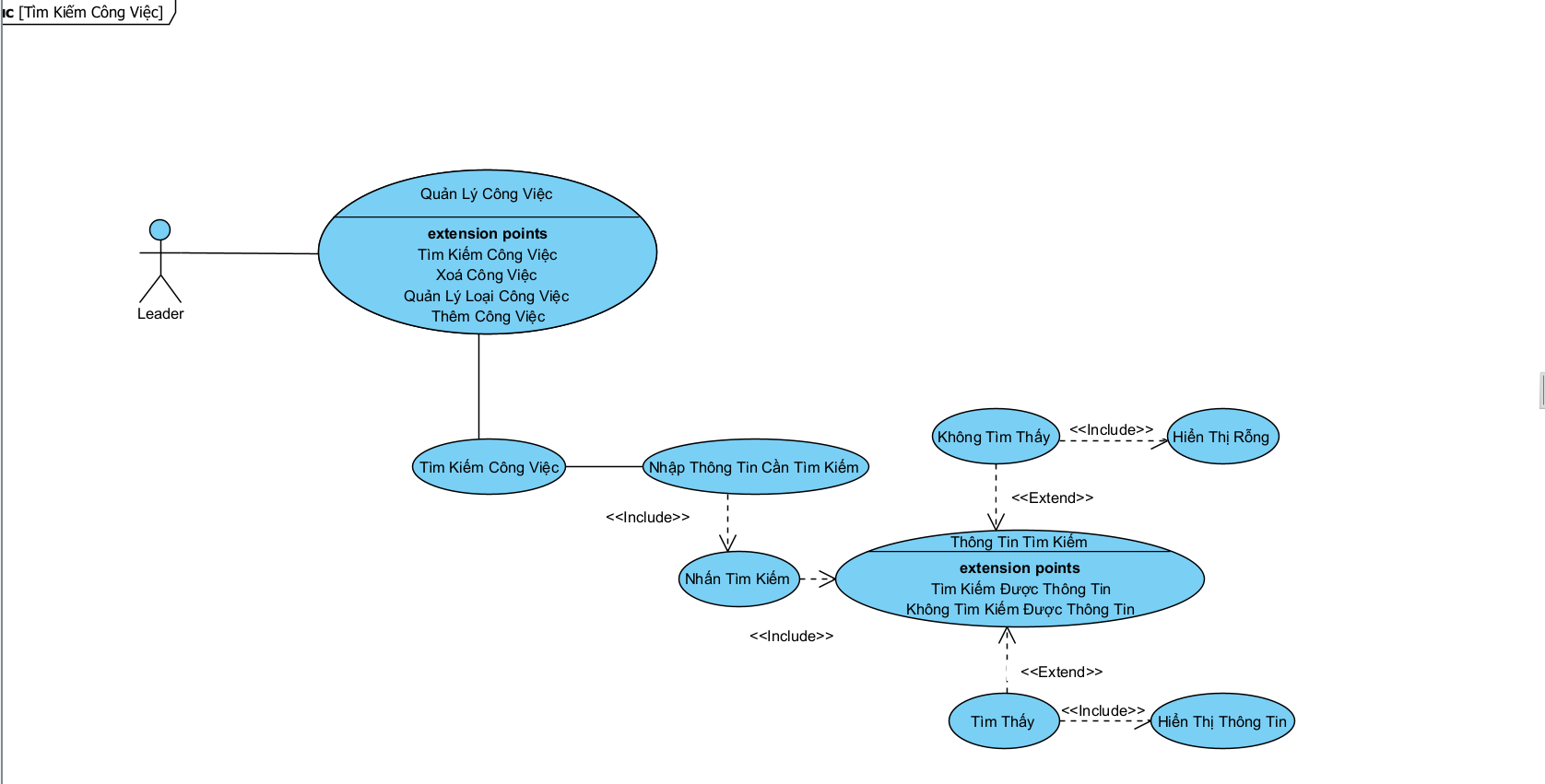


Hình 1. 6 Biểu đồ use case chức năng xóa công việc

* Luồng sự kiện xóa công việc :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã công việc cần xóa | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng xác nhận xóa thông tin công việc | Hệ thống lưu thông tin công việc sau khi xóa mới vào CSDL |

* Use case chức năng tìm kiếm công việc :

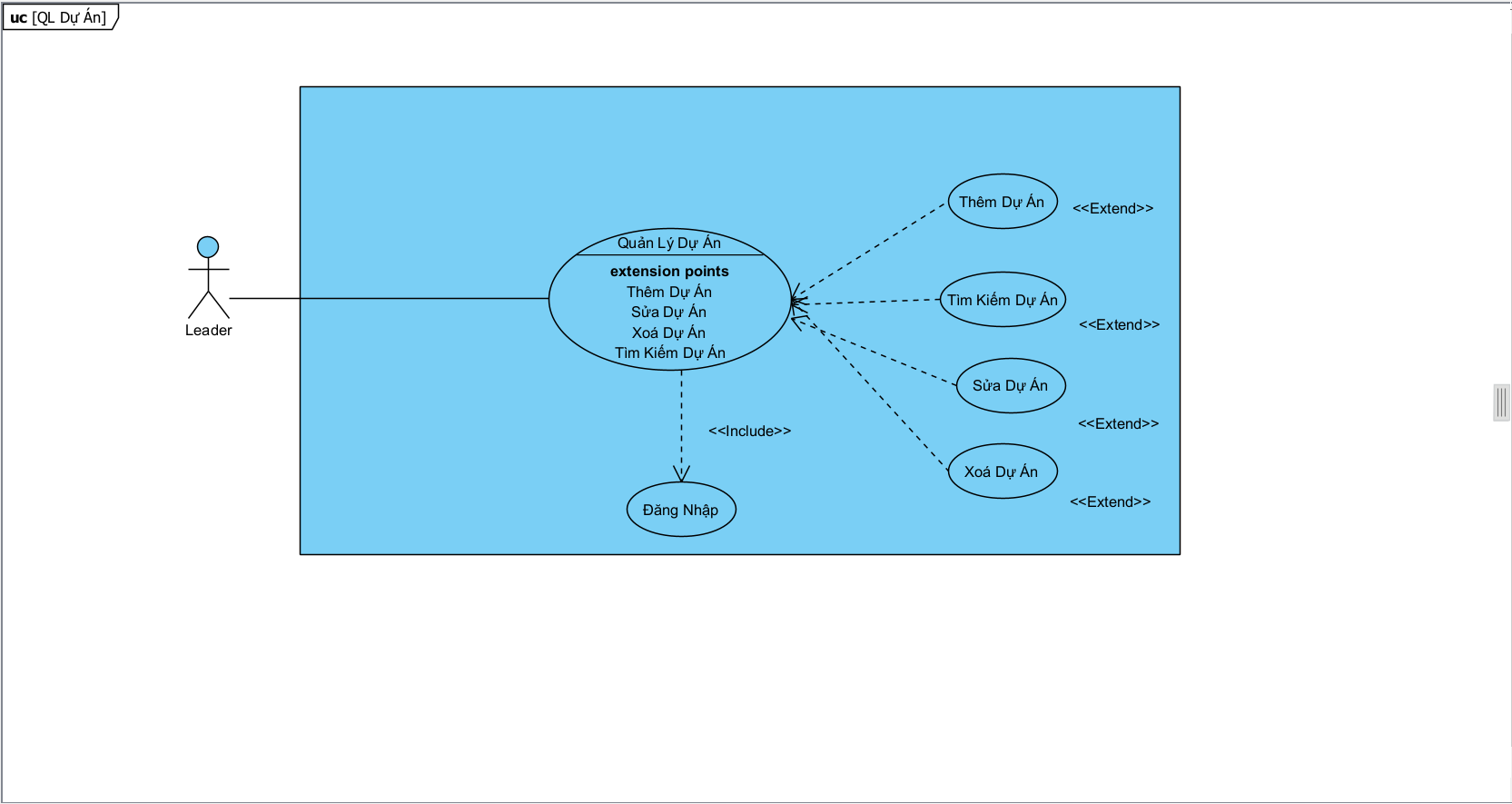


Hình 1. 7 Biểu đồ use case chức năng tìm kiếm công việc

* Luồng sự kiện tìm kiếm công việc :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng tìm kiếm bằng mã công việc | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng tìm kiếm mã công việc có tồn tại | Hệ thống hiển thị thông tin công việc ra màn hình |
| Người dùng tìm kiếm mã công việc không tồn tại | Hệ thống hiển thị rỗng |

* Biều đồ use case quản lý dự án :

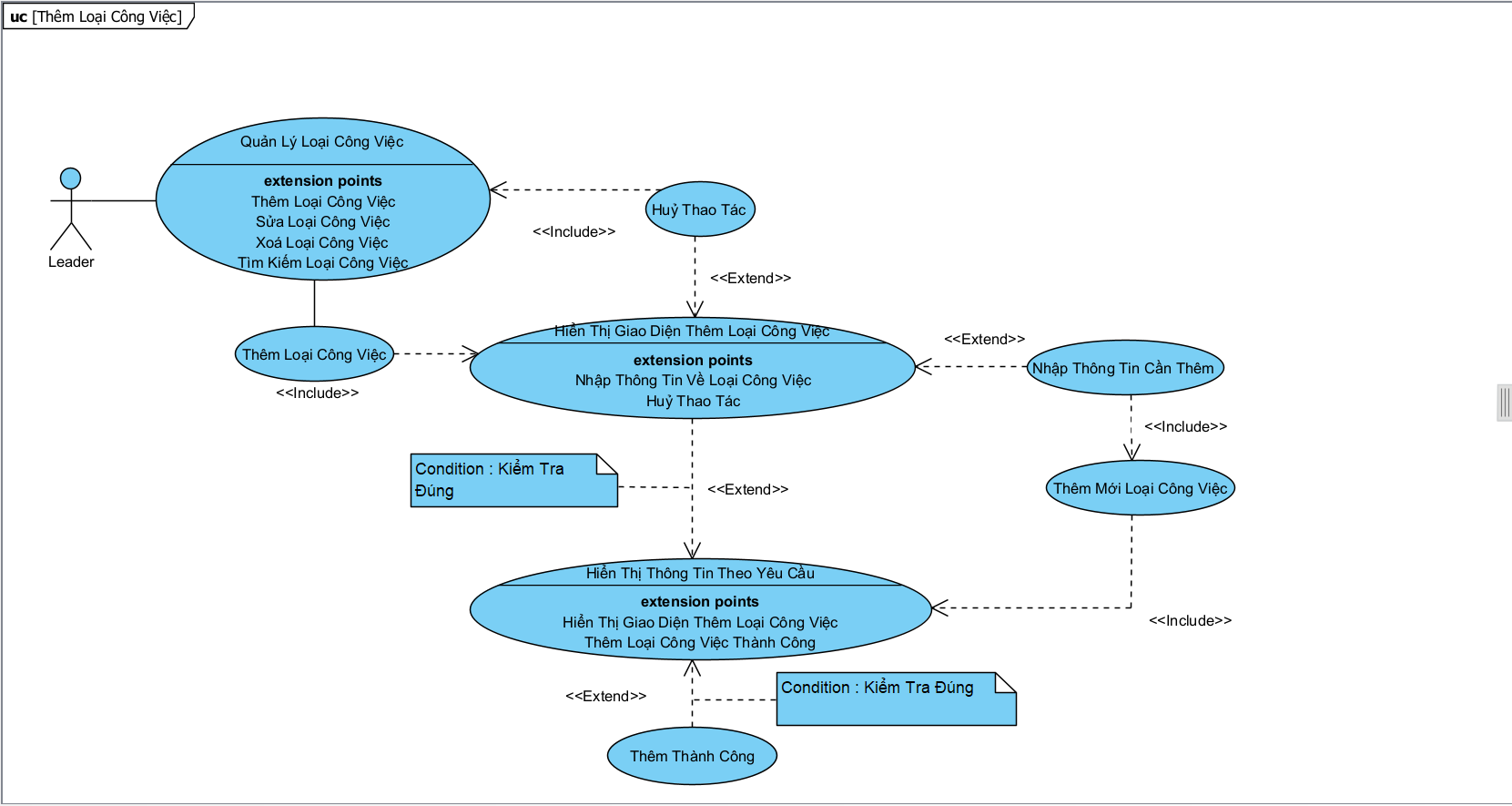


Hình 1. 8 Biểu đồ use case quản trị dự án

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng yêu cầu chức năng quản lý dự án | Hiển thị giao diện lựa chọn các chức năng quản lý dự án. |
| Người dùng yêu cầu chức năng thêm dự án | Hệ thống hiển thị giao diện trang quản lý thêm dự án. |
| Người dùng yêu cầu chức năng sửa dự án | Hệ thống hiển thị giao diện quản lý chức năng sửa thông tin dự án. |
| Người dùng yêu cầu chức năng xóa dự án | Hệ thống hiển thị giao diện quản lý chức năng xóa thông tin dự án. |
| Người dùng yêu cầu chức năng tìm kiếm dự án | Hệ thống hiển thị giao diện quản lý chức năng tìm kiếm thông tin dự án. |

* Use case chức năng thêm dự án :

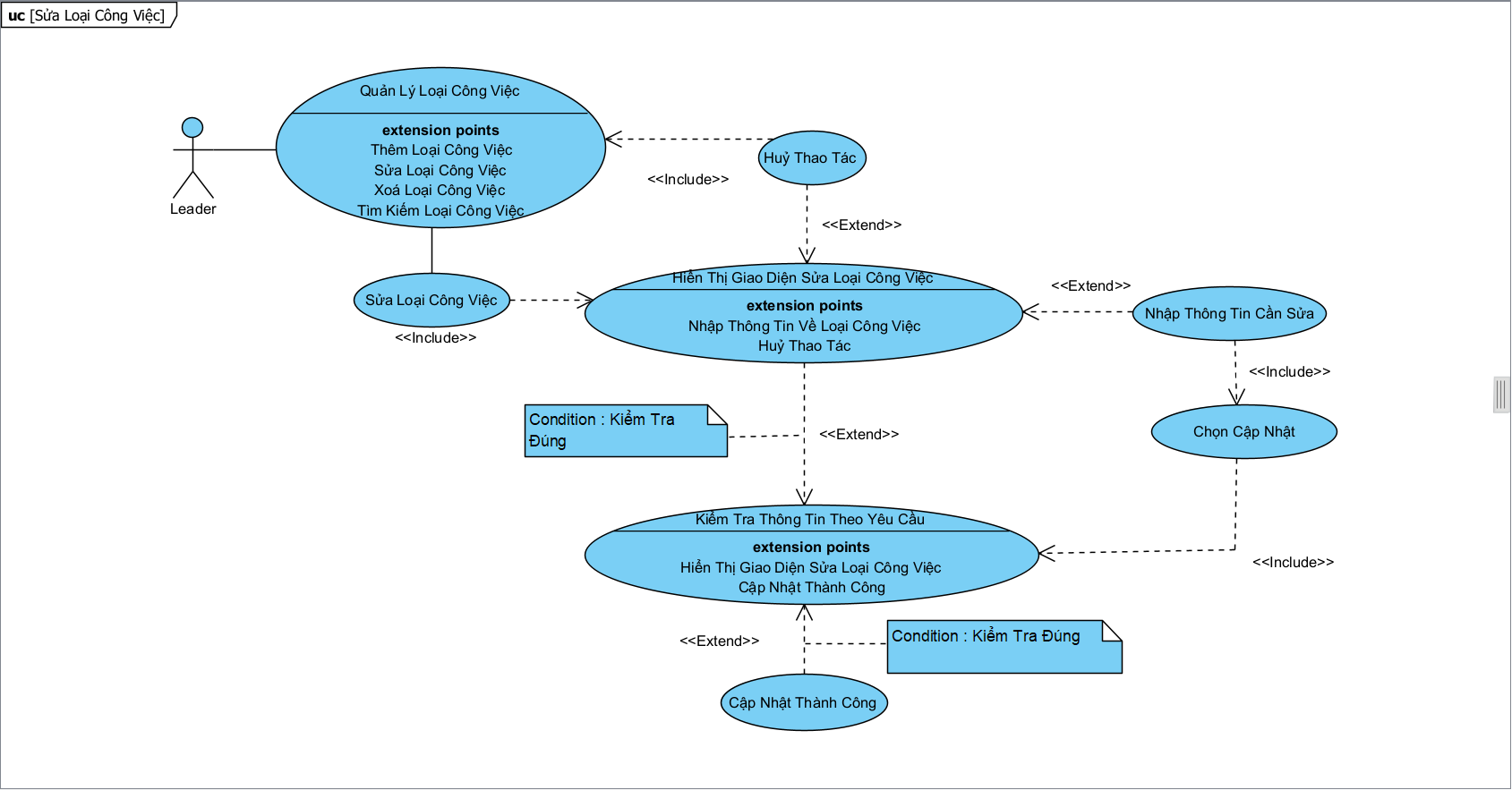


Hình 1. 9 Biểu đồ use case thêm dự án

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã dự án | Hệ thống kiểm tra tính hợp lệ của mã dự án nhập vào |
| Người dùng nhập tiêu đề dự án | Hệ thống kiểm tra tính hợp lệ của tiêu đề dự án nhập vào |
| Người dùng nhập mô tả dự án |  |
| Người dùng xác nhận thêm thông tin dự án | Hệ thống lưu thông tin dự án thêm mới vào CSDL |

* Use case chức năng sửa dự án :

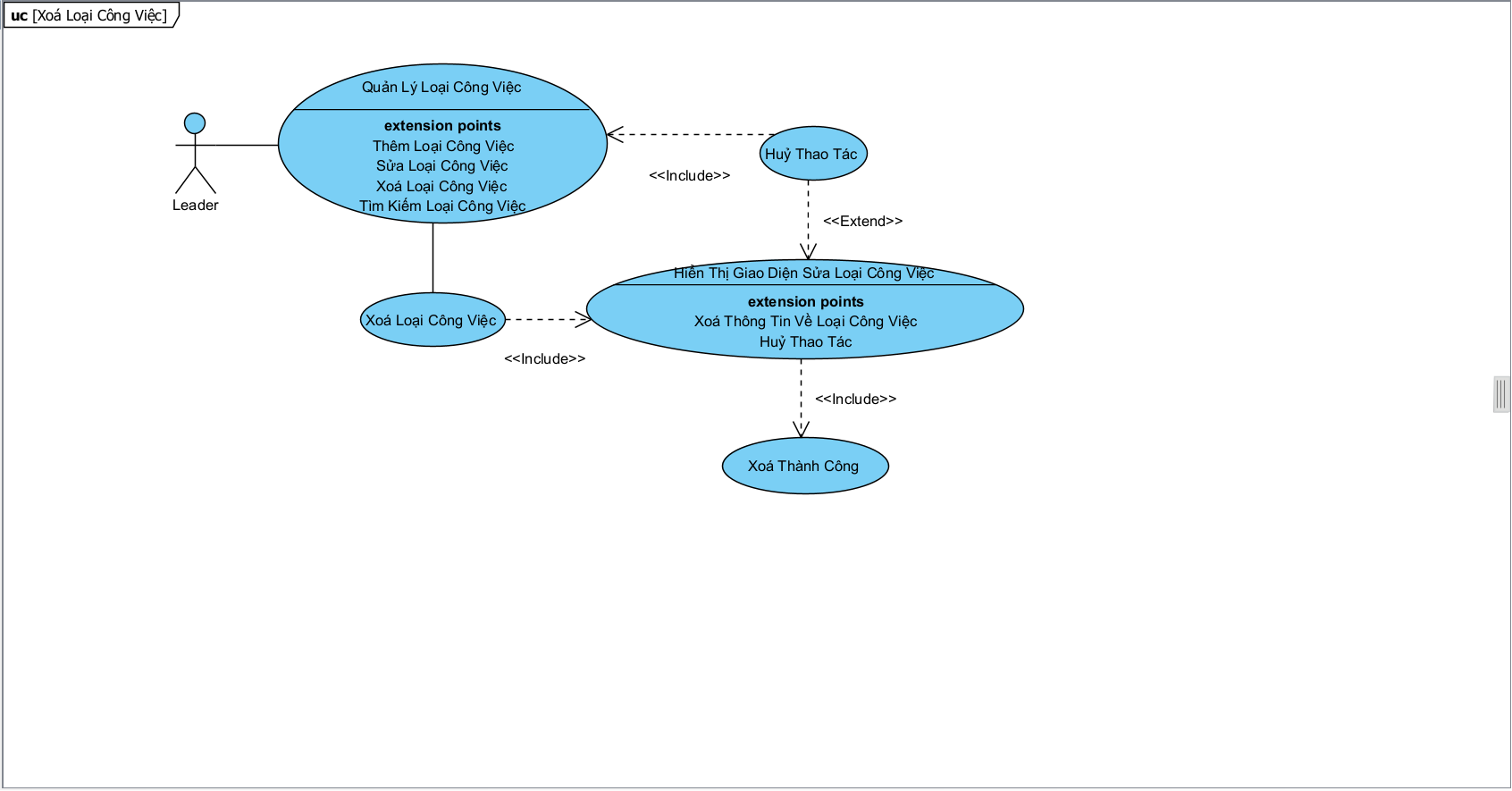


Hình 1. 10 Biểu đồ use case chức năng sửa dự án

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã dự án cần sửa thông tin | Hệ thống kiểm tra tính hợp lệ của mã dự án nhập vào |
| Người dùng nhập tiêu đề dự án sau khi sửa thông tin | Hệ thống kiểm tra tính hợp lệ của tiêu đề dự án nhập vào |
| Người dùng nhập mô tả dự án |  |
| Người dùng xác nhận sửa thông tin dự án | Hệ thống lưu thông tin dự án sau khi sửa vào CSDL |

* Use case chức năng xóa dự án :

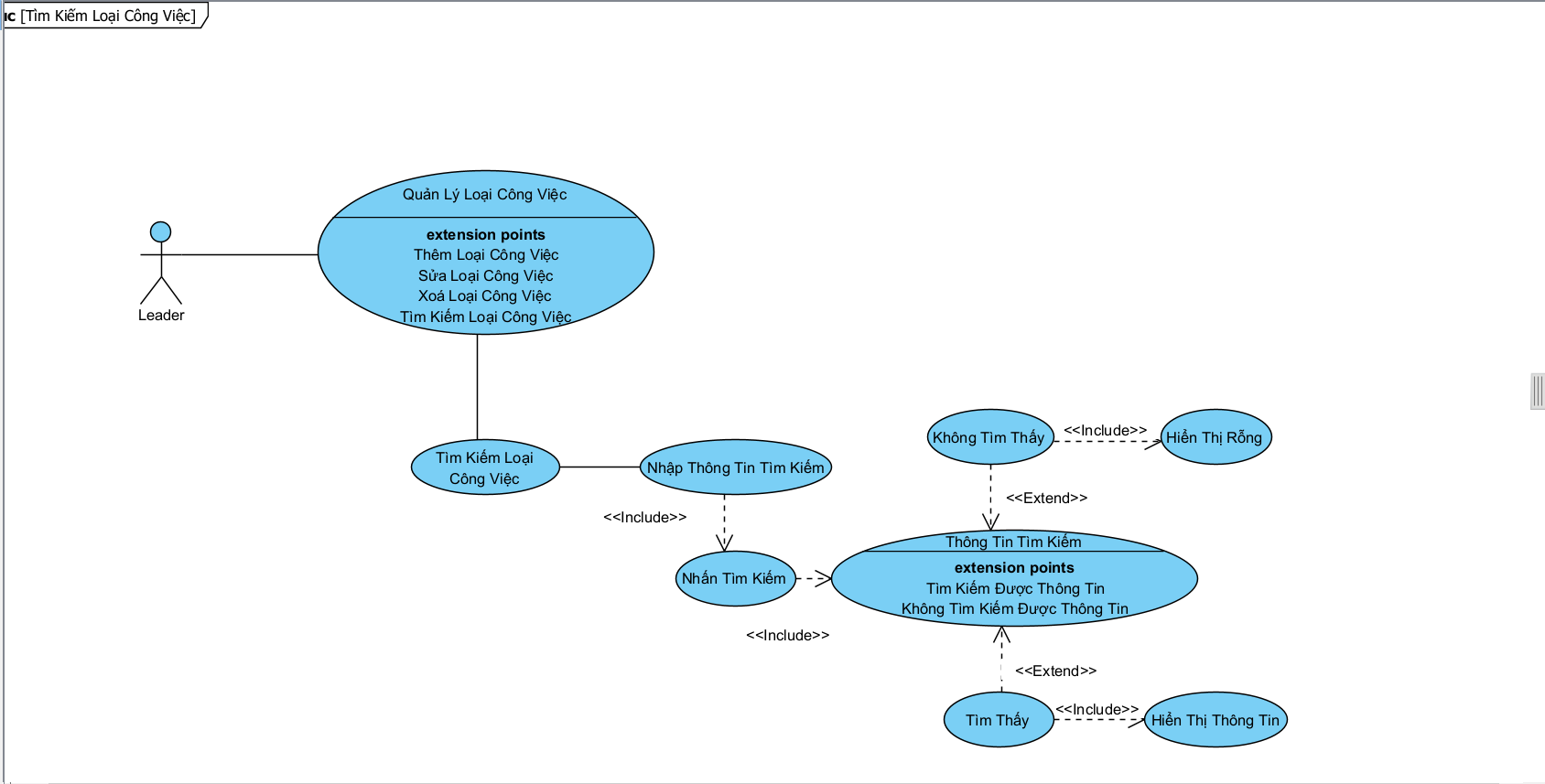


Hình 1. 11 Biểu đồ use case chức năng xóa dự án

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã dự án cần xóa thông tin | Hệ thống kiểm tra tính hợp lệ của mã dự án nhập vào |
| Người dùng xác nhận xóa thông tin dự án | Hệ thống lưu thông tin dự án sau khi sửa vào CSDL |

* Use case chức năng tìm kiếm dự án :

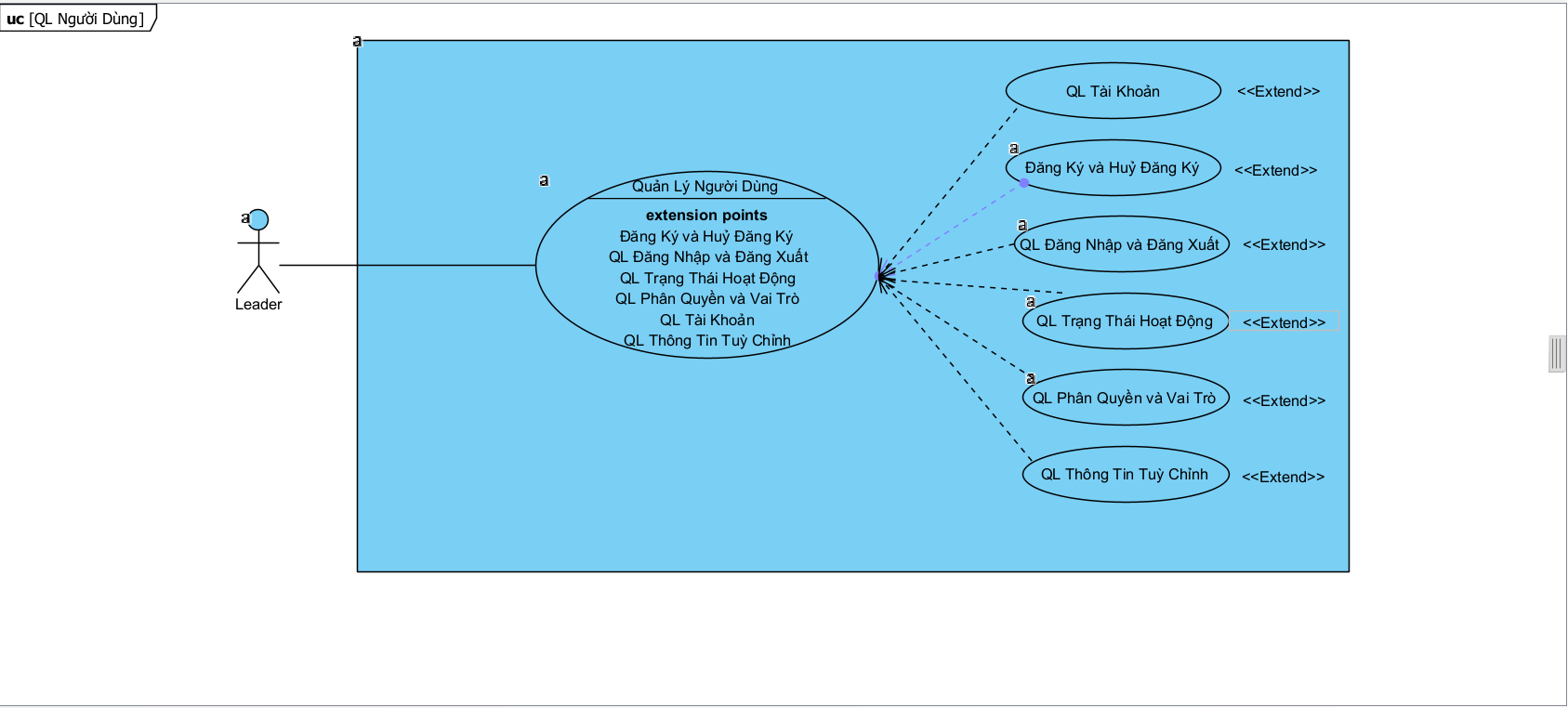


Hình 1. 12 Biểu đồ chức năng tìm kiếm dự án

* Luồng sự kiện chính :

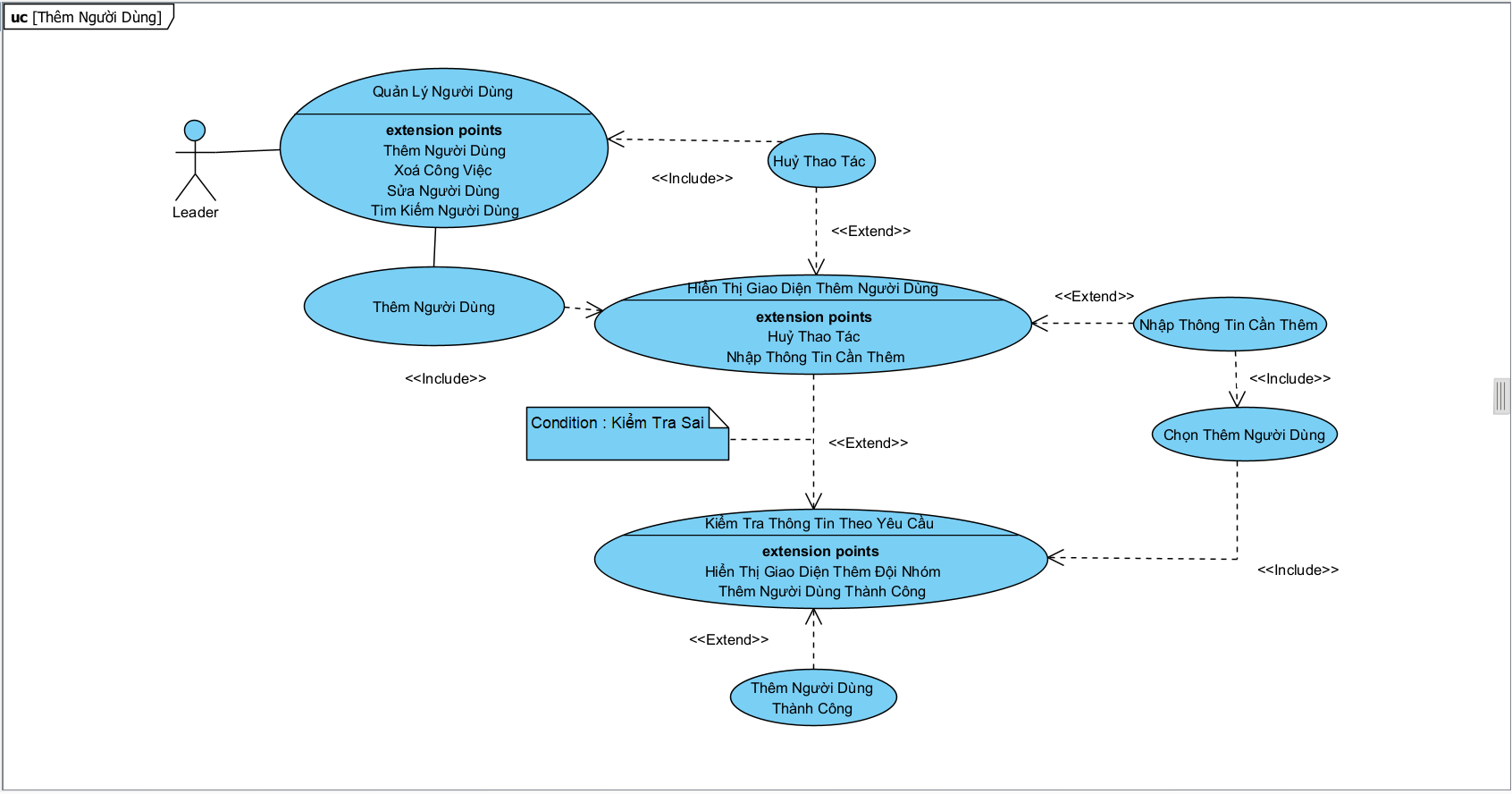
|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng tìm kiếm bằng mã dự án | Hệ thống kiểm tra tính hợp lệ của mã dự án nhập vào |
| Người dùng tìm kiếm mã dự án có tồn tại | Hệ thống hiển thị thông tin dự án ra màn hình |
| Người dùng tìm kiếm mã dự án không tồn tại | Hệ thống hiển thị rỗng |

* Biểu đồ use case chức năng quản lý người dùng



Hình 1. 13 Biểu đồ use case chức năng quản lý người dùng

* Use case chức năng thêm người dùng :

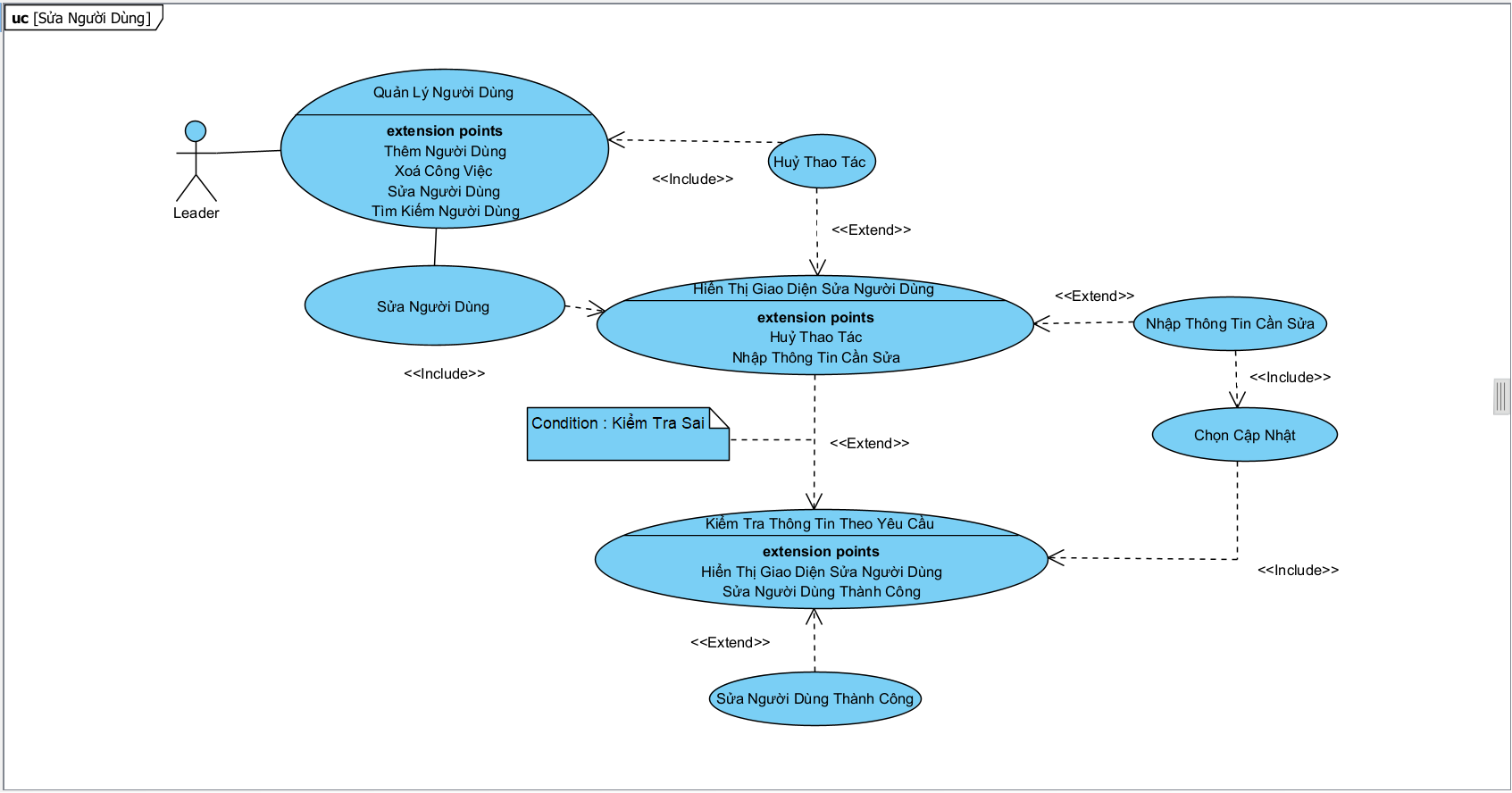


Hình 1. 14 Biểu đồ use case chức năng thêm người dùng

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã người dùng cần thêm | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng nhập tiêu tên và họ đầy đủ | Hệ thống kiểm tra tính hợp lệ của tên công việc nhập vào |
| Người dùng điền thêm thông tin người dùng | Hệ thống lưu thông tin người dùng thêm mới vào CSDL |

* Use case chức năng sửa người dùng

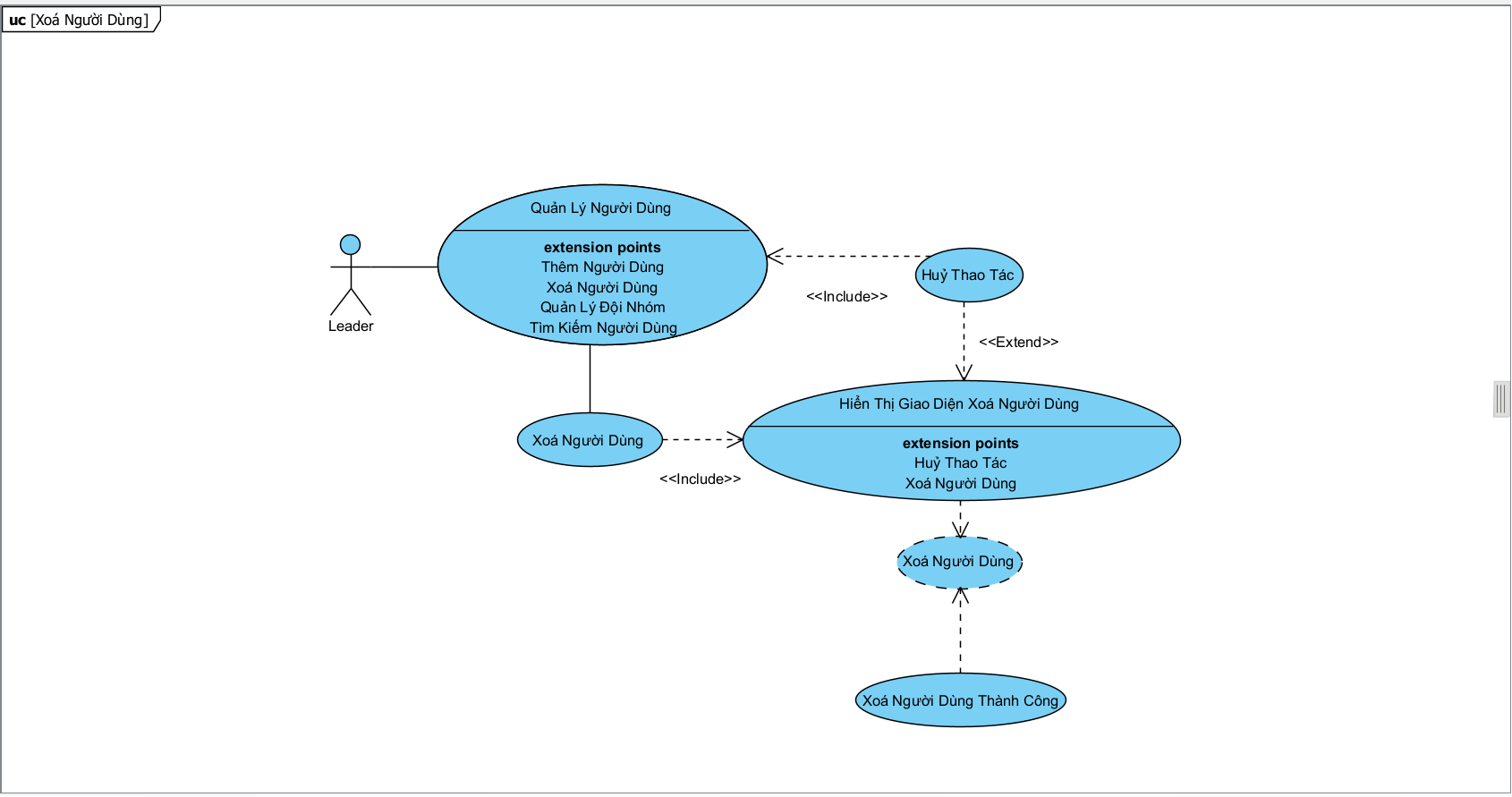


Hình 1. 15 Biểu đồ use case chức năng sửa người dùng

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã người dùng cần sửa | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng nhập tên người dùng | Hệ thống kiểm tra tính hợp lệ của tên người dùng nhập vào |
| Người dùng nhập mô tả người dùng |  |
| Người dùng xác nhận sửa thông tin người dùng | Hệ thống lưu thông tin người dùng sau khi sửa mới vào CSDL |

* Use case chức năng xóa người dùng :

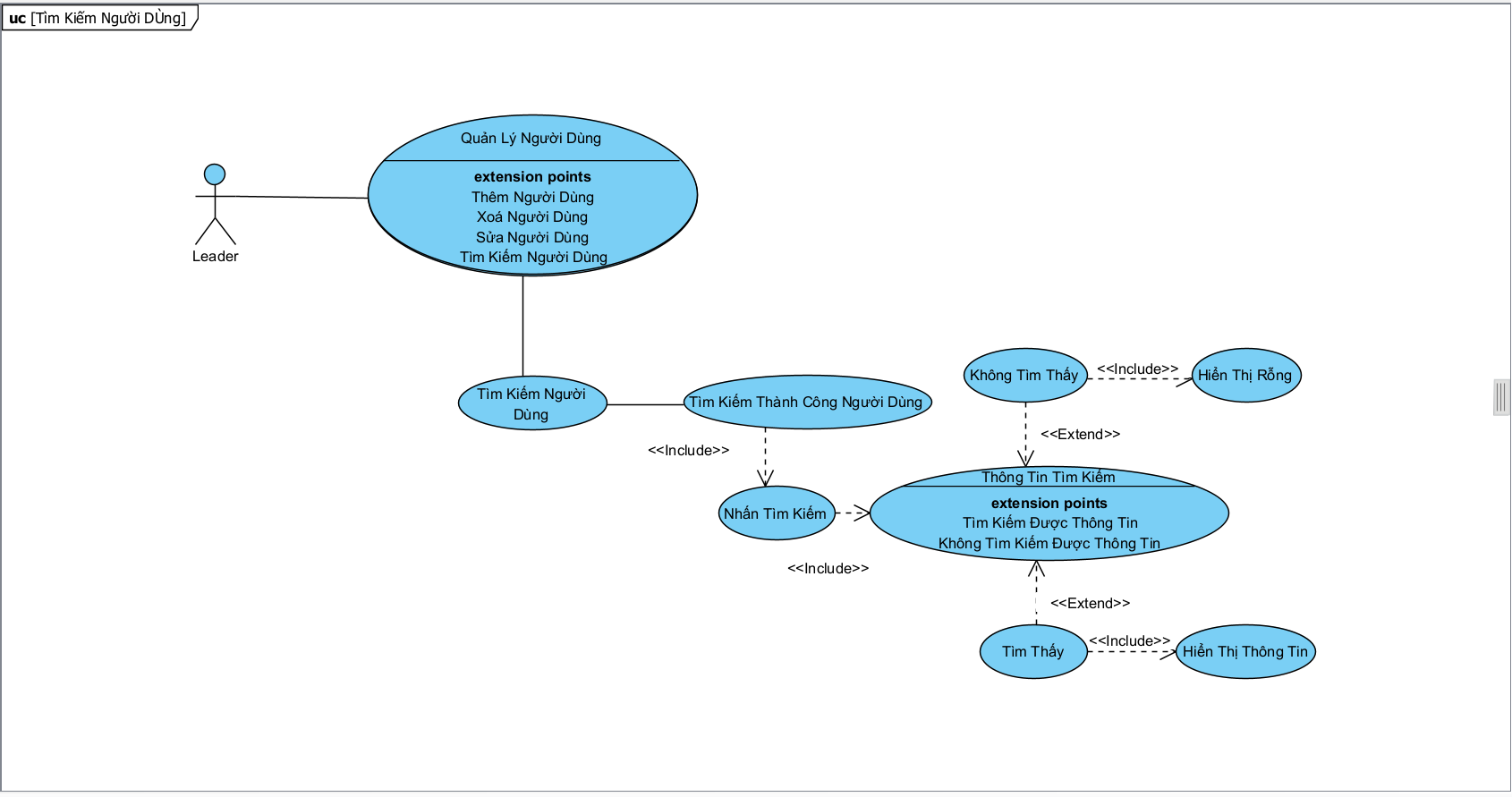


Hình 1. 16 Biểu đồ use case chức năng xóa người dùng

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã công việc cần xóa | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng xác nhận xóa thông tin công việc | Hệ thống lưu thông tin công việc sau khi xóa mới vào CSDL |

* Use case chức năng tìm kiếm người dùng :

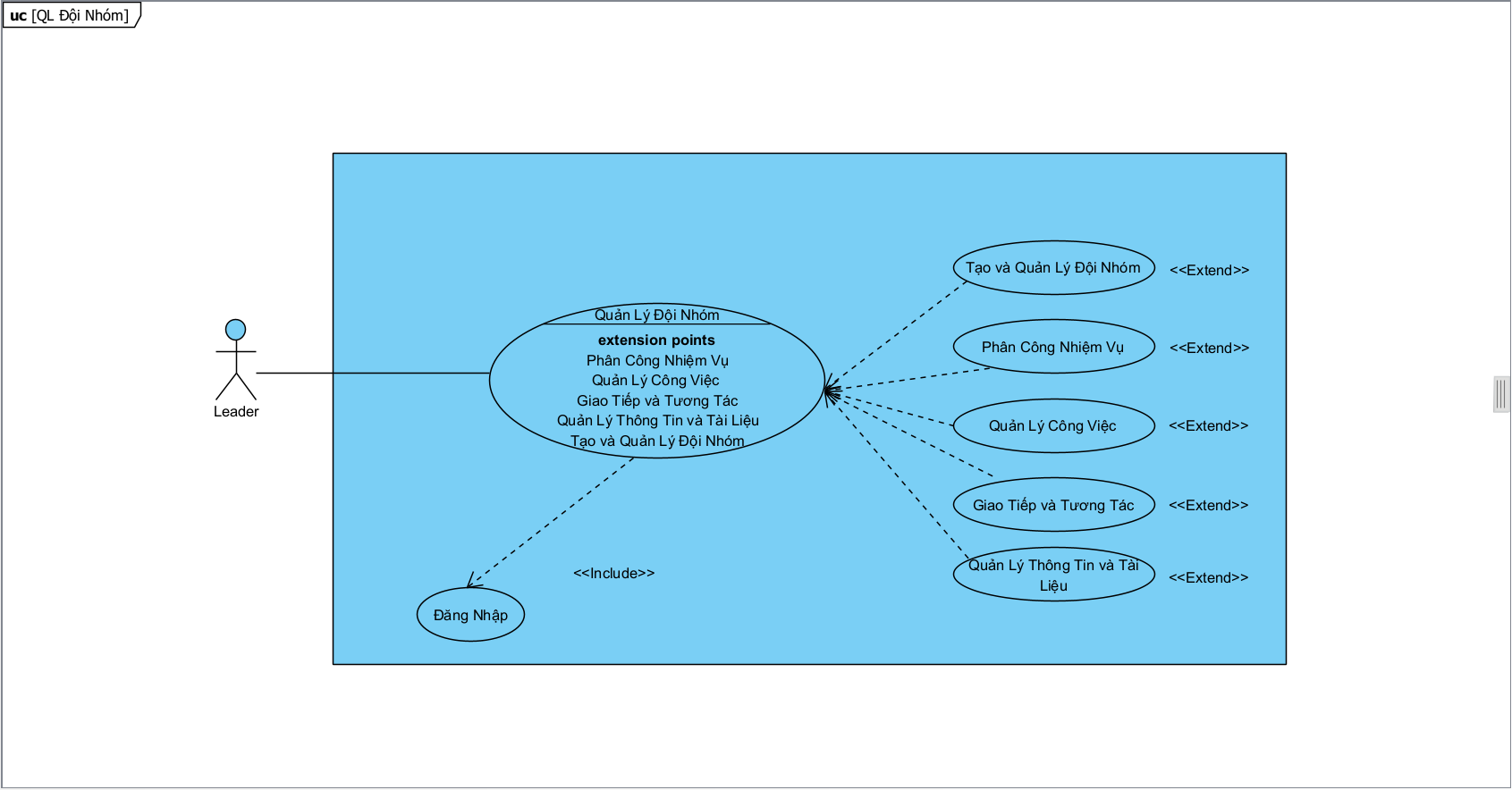


Hình 1. 17 Biểu đồ chức năng tìm kiếm người dùng

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng tìm kiếm bằng mã người dùng | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng tìm kiếm mã người dùng có tồn tại | Hệ thống hiển thị thông tin người dùng ra màn hình |
| Người dùng tìm kiếm mã người dùng không tồn tại | Hệ thống hiển thị rỗng |

* Biểu đồ use case chức năng quản lý đội nhóm :



Hình 1. 18 Biểu đồ use case chức năng quản lý đội nhóm

* Use case chức năng thêm đội nhóm



Hình 1. 19 Biểu đồ use case thêm đội nhóm

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã người dùng cần thêm | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng nhập tiêu tên và họ đầy đủ | Hệ thống kiểm tra tính hợp lệ của tên công việc nhập vào |
| Người dùng điền thêm thông tin người dùng | Hệ thống lưu thông tin người dùng thêm mới vào CSDL |

* Use case chức năng sửa đội nhóm

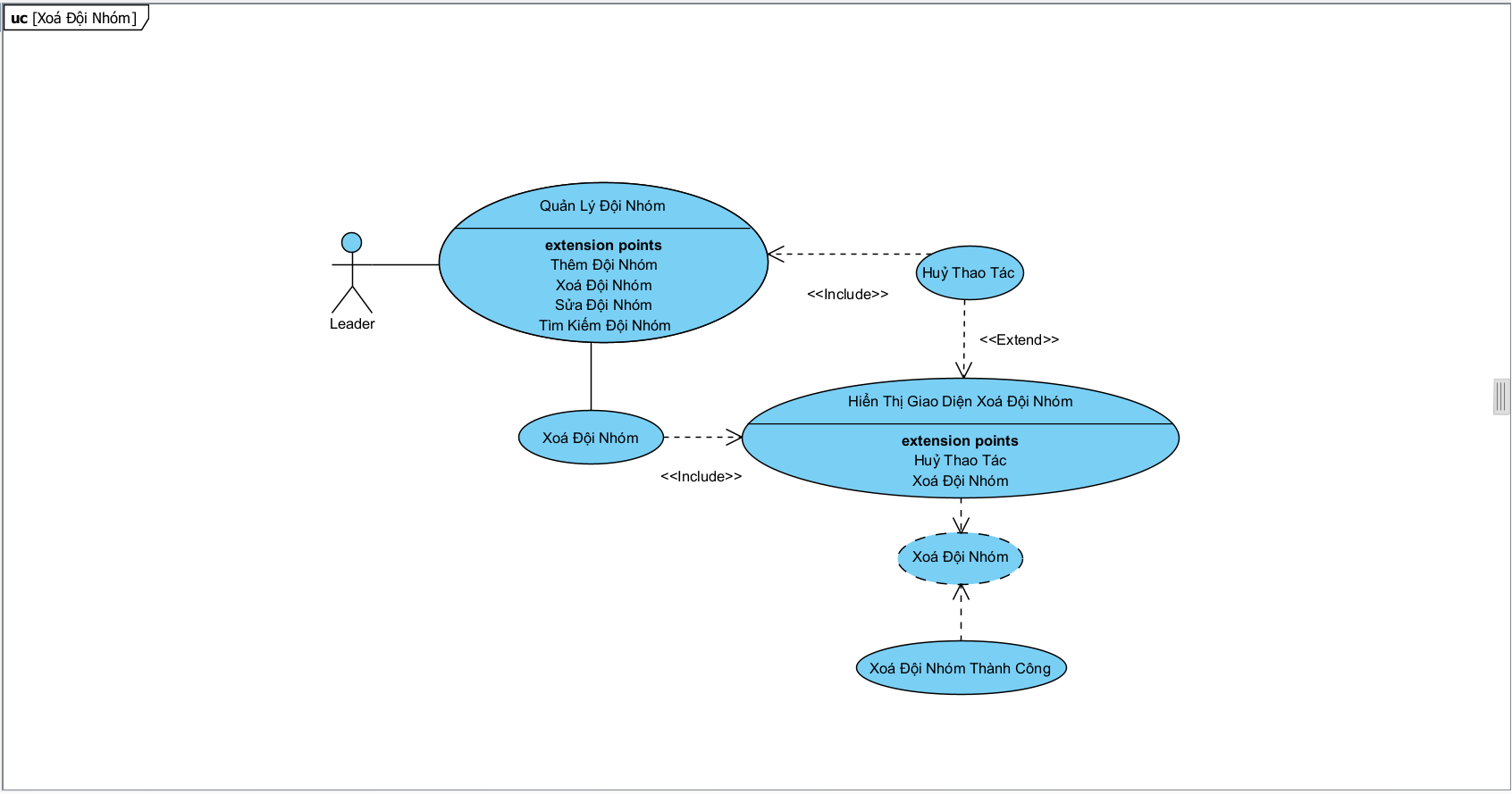


Hình 1. 20 Biểu đồ use case chức năng sửa đội nhóm

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã người dùng cần sửa | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng nhập tên người dùng | Hệ thống kiểm tra tính hợp lệ của tên người dùng nhập vào |
| Người dùng nhập mô tả người dùng |  |
| Người dùng xác nhận sửa thông tin người dùng | Hệ thống lưu thông tin người dùng sau khi sửa mới vào CSDL |

* Use case chức năng xóa đội nhóm

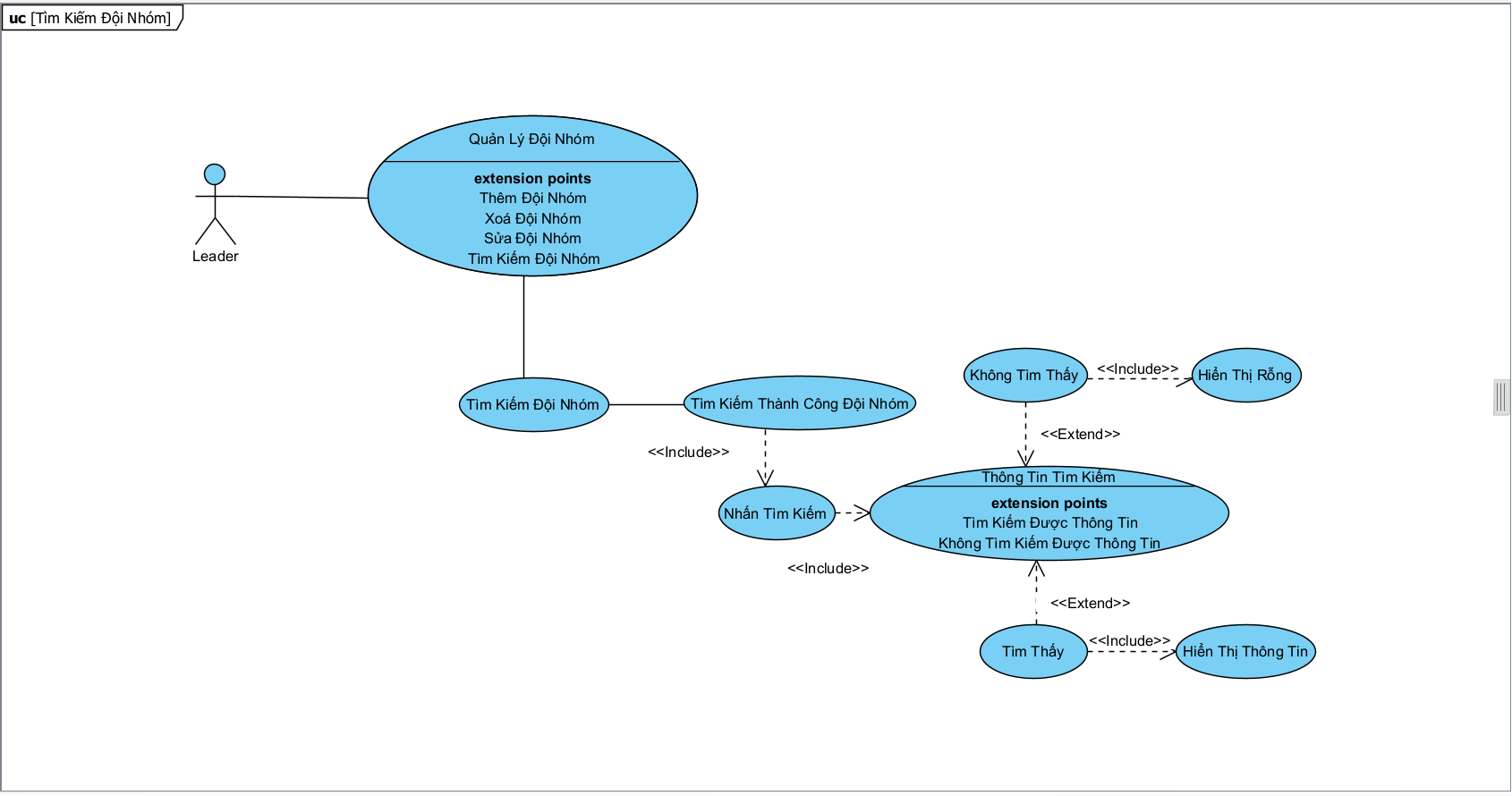


Hình 1. 21 Biểu đồ use case chức năng xóa đội nhóm

* Luồng sự kiến chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng nhập mã công việc cần xóa | Hệ thống kiểm tra tính hợp lệ của mã công việc nhập vào |
| Người dùng xác nhận xóa thông tin công việc | Hệ thống lưu thông tin công việc sau khi xóa mới vào CSDL |

* Use case chức năng tìm kiếm đội nhóm



Hình 1. 22 Biểu đồ chức năng tìm kiếm đội nhóm

* Luồng sự kiện chính :

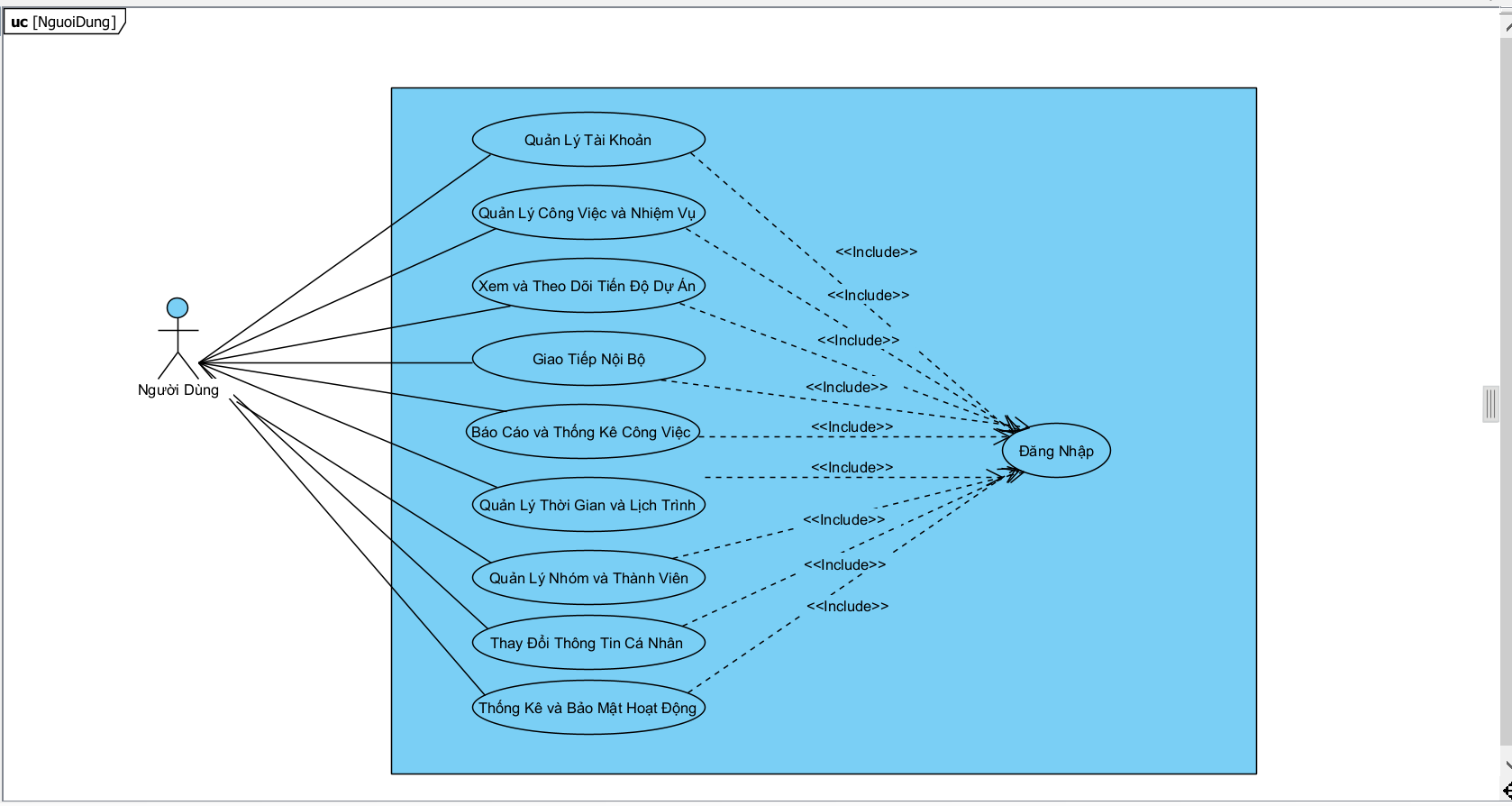
|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng tìm kiếm bằng mã người dùng | Hệ thống kiểm tra tính hợp lệ của mã người dùng nhập vào |
| Người dùng tìm kiếm mã người dùng có tồn tại | Hệ thống hiển thị thông tin người dùng ra màn hình |
| Người dùng tìm kiếm mã người dùng không tồn tại | Hệ thống hiển thị rỗng |

* + - 1. **Chức năng của phân hệ người dùng**

|  |  |  |
| --- | --- | --- |
| **STT** | **Vai trò** | **Mô tả** |
| 1 | Quản lý tài khoản | Người dùng có thể truy cập hệ thống bằng cách đăng nhập bằng tên đăng nhập và mật khẩu. Nếu là người dùng mới, có chức năng đăng ký tài khoản với thông tin cần thiết. |
| 2 | Quản lý công việc và nhiệm vụ | Người dùng có thể tạo, sửa đổi, và xoá công việc và nhiệm vụ. Có khả năng gán công việc cho bản thân hoặc các thành viên khác trong dự án. |
| 3 | Xem và Theo dõi tiến độ dự án | Cung cấp biểu đồ và báo cáo trực quan về trạng thái và tiến độ của dự án. Người dùng có thể theo dõi và quản lý công việc và nhiệm vụ một cách hiệu quả. |
| 4 | Giao tiếp nội bộ | Hỗ trợ và nhận tin nhắn nội bộ, thông báo dự án và bình luận trên công việc. tạo ra một không gian giao tiếp linh hoạt. |
| 5 | Báo cáo và Thống kê công việc | Cho phép tạo báo cáo tiến độ, hiệu suất cá nhân và nhóm. Thống kê giúp đưa ra quyết định dựa trên dữ liệu. |
| 6 | Quản lý thời gian và Lịch trình | Đặt lịch làm việc cá nhân và theo dõi lịch trình của dự án. Hệ thống cung cấp thông báo và nhắc nhở về các sự kiện quan trọng. |
| 7 | Quản lý nhóm và Thành viên | Người quản trị có thể tạo và quản lý nhóm công việc. Gán và quản lý công việc và nhiệm vụ cho từng thành viên trong nhóm. |
| 8 | Thay đổi thông tin cá nhân | Người dùng có thể cập nhật thông tin cá nhân như mật khẩu, ảnh đại diện, thông tin liên lạc. |
| 9 | Thống kê bảo mật và Hoạt động | Hiển thị lịch sử đăng nhập và các hoạt động khác trong hệ thống bảo vệ an toàn thông tin và tăng cường bảo mật. |

Bảng 1. 2 Bảng chức năng phân hệ quản trị người dùng

* **Use case tổng quát phân hệ người dùng :**



Bảng 1. 3 Biểu đồ use case chức năng quản lý người dùng

* Use case phân rã quản lý tài khoản :

***Diagram

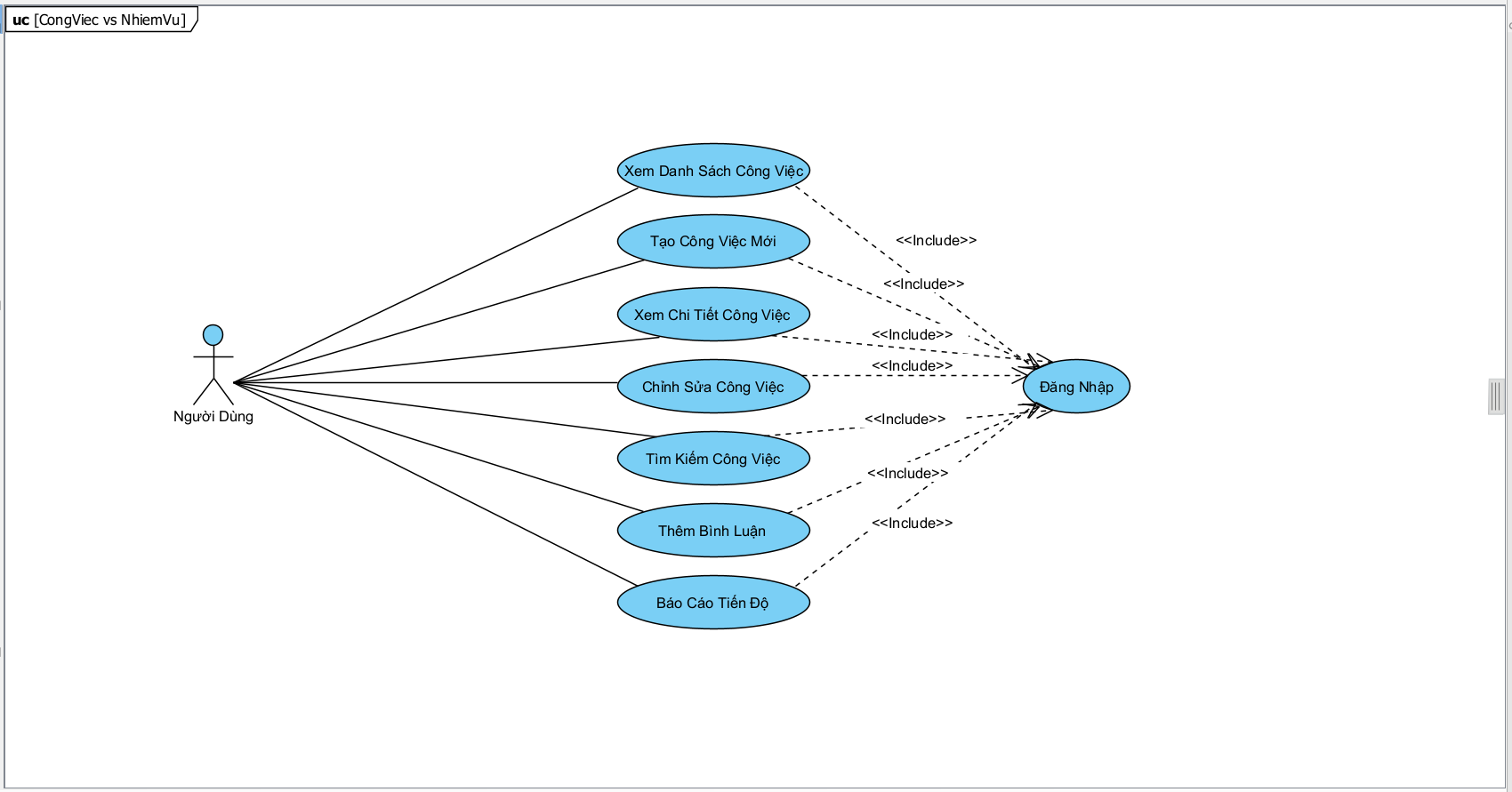
Description automatically generated***

Hình 1. 23 Biểu đồ use case phân rã tài khoản

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng yêu cầu chức năng đăng nhập | Hệ thống hiển thị ra giao diện đăng nhập |
| Người dùng nhập các thông tin yêu cầu của hệ thống | Hệ thống yêu cầu kiểm tra thông tin tài khoản. |

* Use case phân rã quản lý công việc :



Hình 1. 24 Biểu đồ use case phân rã công việc và nhiệm vụ

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Người dùng sau khi đăng nhập thành công, di chuyển đến mục “ Danh Sách Công Việc” hoặc tìm kiếm chức năng tương ứng trong giao diện người dùng. | Hệ thống phản ứng cách hiển thị danh sách công việc đầy đủ thông tin tên giao diện người dùng. Mỗi công việc sẽ có các thông tin như tên công việc, người thực hiện, thời hạn, trạng thái và các chi tiết khác. |
| Người dùng chọn “Xem Danh Sách Công Việc” để hiển thị thông tin về các công việc có trong dự án. | Hệ thống hiển thị thông tin người dùng ra màn hình |
| Tạo công việc : Nhập thông tin công việc mới (tên, mô tả, ngày hết hạn, ưu tiên, danh sách công việc liên quan) | Lưu thông tin công việc vào cơ sở dữ liệu.  Hiển thị thông tin công việc mới trong danh sách công việc của người dùng. |
| Người dùng muốn sửa đổi thông tin của công việc (tên, mô tả, ngày hết hạn, ưu tiên) | Hệ thống sẽ cập nhật thông tin công việc trong cơ sở dữ liệu và Hiển thị thông tin cập nhật trong danh sách công việc. |
| Người dùng muốn xoá công việc không cần thiết | Hệ thống xoá thông tin công việc khỏi cơ sở dữ liệu. Cập nhật danh sách công việc để không hiển thị công việc đã bị xoá. |
| Gửi bình luận về công việc hoặc đọc bình luận từ thành viên khác. | Hệ thống sẽ lưu thông tin bình luận trong cơ sở dữ liệu và đồng thời sẽ hiển thị bình luận mới và thông báo cho người dùng liên quan. |
| Người dùng sẽ tạo báo cáo về tiến độ công việc hoặc hiệu suất cá nhân | Hệ thống sẽ tính toán dữ liệu từ cơ sở dữ liệu để tạo báo cáo và Hiển thị và cung cấp tải xuống nếu cần. |

* Use case phân rã quản trị dự án :



Hình 1. 25 Biểu đồ chức năng quản trị dự án

* Luồng sự kiện chính :

|  |  |
| --- | --- |
| Hành động của tác nhân | Phản ứng của hệ thống |
| Quản trị dự án đăng nhập vào hệ thống bằng tên đăng nhập và mật khẩu được cấp. |  |
| Quản trị dự án di chuyển đến mục “Quản lý dự án” hoặc tìm kiếm chức năng tương ứng trong giao diện người dùng. | Hệ thống hiển thị thông tin chi tiết về dự án, bao gồm danh sách các công việc, thành viên tham gia, tiến độ, trạng thái |
| Quản trị dự án chọn dự án cụ thể mà họ muốn quản lý từ danh sách dự án có sẵn | Hệ thống cung cấp giao diện quản lý dự án cho quản trị, nơi họ có thể hiện các hoạt động như gán công việc, theo dõi tiến độ và thay đổi trạng thái. |
| Quản trị dự án gán công việc cụ thể cho từng thành viên trong nhóm, đảm bảo mỗi người có nhiệm vụ rõ ràng. | Mở giao diện quản lý công việc, cho phép quản trị dự án gán công việc cho các thành viên trong nhóm. |
| Quản trị dự án có thể đặt ưu tiên cho các công việc trong dự án dựa trên mức độ quan trọng và khẩn cấp. | Cập nhật ưu tiên của công việc trong cơ sở dữ liệu, đồng thời cập nhật trên giao diện người dùng. |
| Quản trị dự án theo dõi tiến độ tổng thể của dự án thông qua bảng điều khiển, đảm bảo rặng dự án tiến triển đúng đắn. | Hiển thị biểu đồ và báo cáo tiến độ dự án, được cập nhật theo thời gian thực. |

* + 1. Biểu đồ lớp thực thể



Hình 1. 26 Biểu đồ lớp thực thể hệ thống

* + 1. Các yêu cầu phi chức năng

Yêu cầu phi chức năng là những yêu cầu không liên quan trực tiếp đến việc thực hiện các nhiệm vụ cụ thể trong hệ thống, nhưng lại quan trọng để đảm bảo hiệu suất, an toàn và sự thân thiện của người dùng.

1. Tính bảo mật và Quyền riêng tư :

* Hệ thống cung cấp cơ chế xác thực mạnh mẽ để bảo vệ thông tin nhạy cảm.
* Quản lý quyền hạn để đảm bảo người dùng chỉ có thể truy cập thông tin chức năng tương ứng với vai trò của họ.

1. Hiệu suất và Tốc độ :

* Hệ thống phải đảm bảo hiệu suất cao, đặc biệt là khi có nhiều người dùng đồng thời sử dụng.
* Thời gian phản hồi của hệ thống cần làm giảm thiểu cải thiện trải nghiệm người dùng.

1. **Độ tin cậy và Sẵn sàng :**

* Hệ thống cần đảm bảo tính sẵn sàng liên tục để người dùng có thể truy cập mọi lúc.
* Sao lưu dữ liệu định kỳ và khả năng khôi phục dữ liệu nhanh chóng khi có sự cố.

1. **Hỗ trợ đa nền tảng và Đa thiết bị :**

* Hệ thống nên tương thích với nhiều nền tảng (Web, di động, desktop) và nhiều thiết bị khác .

1. **Dễ dàng sử dụng và Giao diện người dùng thân thiện :**

* Giao diện người dùng phải được thiết kế sao cho dễ sử dụng và hiểu.
* Cung cấp các hướng dẫn sử dụng hoặc tài liệu hỗ trợ để giúp người dùng mới làm quen với hệ thống.

1. **Hỗ trợ đa ngôn ngữ và Đa vùng :**

* Cung cấp hỗ trợ cho nhiều ngôn ngữ và vùng để phục vụ người dùng trên khắp thế giới.
  1. Thiết kế hệ thống
     1. Thiết kê cơ sở dữ liệu
* **Mô tả cấu trúc bảng :**
* **Bảng dự án :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| projectID | INT | Không rỗng | Mã dự án |
| projectName | NVARCHAR(200) | Không rỗng | Tên dự án |
| projectKey | NVARCHAR(10) | Không rỗng | Mã định danh |
| projectLead | INT | Không rỗng | ID người quản lý |
| createdDate | DATETIME | Không rỗng | Ngày tạo dự án |
| endDate | DATETIME | Không rỗng | Ngày kết thúc |
| progress | INT | Không rỗng | Tiến độ |
| estimatedEndDate | DATETIME | Không rỗng | Ngày dự kiến |
| budget | DECIMAL(18, 2) | Không rỗng | Ngân sách |

* **Bảng chi tiết dự án :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| projectID | INT | Khóa chính | Mã dự án |
| projectDescription | NVARCHAR(200) | Không rỗng | Mô tả dự án |
| clientName | NVARCHAR(10) | Không rỗng | Tên đối tác |
| clientContactName | NVARCHAR(200) | Không rỗng | Tên người liên hệ |
| clientContactEmail | NVARCHAR(200) | Không rỗng | Email của người liên hệ |
| clientContactPhone | NVARCHAR(10) | Không rỗng | Số điện thoại của người liên hệ |
| projectManagerID | INT | Không rỗng | Người quản lý dự án |

* **Bảng công việc :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| taskID | INT | Khoá chính | Mã công việc |
| projectID | INT | Khoá ngoại | Mã dự án |
| taskType | VARCHAR(10) | Khoá ngoại | Loại công việc |
| summary | INT | Không rỗng | Nội dung công việc |
| assigneeID | INT | Không rỗng | Người được giao |
| status | VARCHAR(20) | Không rỗng | Trạng thái |
| createdDate | DATETIME | Không rỗng | Ngày tạo |
| description | TEXT | Không rỗng | Mô tả |
| startDate | DATETIME | Không rỗng | Ngày bắt đầu |
| priority | INT | Không rỗng | Mức độ |

* **Bảng chi tiết công việc :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| taskID | INT | Khóa chính | Mã công việc |
| taskDescription | NVARCHAR(200) | Không rỗng | Mô tả công việc |
| estimatedHours | INT | Không rỗng | Số giờ đăng ký |
| actualHoursSpent | INT | Không rỗng | Số giờ hoàn thành |
| taskManagerID | INT | Không rỗng | Ngày tạo dự án |

* **Bảng phân công việc :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| taskAssignmentID | INT | Khoá chính | Mã nhân viên |
| taskID | NVARCHAR(200) | Không rỗng | Mã công việc |
| assigneeID | INT | Không rỗng | Người được giao công việc |
| hoursSpent | INT | Không rỗng | Số giờ đã tiêu thụ cho công việ |
| completionDate | INT | Không rỗng | Ngày hoàn thành công việc |

* **Bảng bình luận :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| commentID | INT | Khoá chính | Mã bình luận |
| taskID | INT | Khoá ngoại | Công việc mà bình luận |
| userID | INT | Không rỗng | Người gửi bình luận |
| commentText | TEXT | Không rỗng | Nội dung bình luận |
| attachmentData | VARBINARY(MAX) | Không rỗng | Bình luận cha nếu có |
| createdDate | DATETIME | Không rỗng | Ngày tạo bình luận |

* **Bảng người dùng :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| userID | INT | Khoá chính | Mã người dùng |
| picture | VARBINARY(MAX) | Không rỗng | Hình ảnh |
| fullName | VARCHAR(150) | Khoá ngoại | Họ tên |
| passwordHash | VARCHAR(255) | Không rỗng | Mật khẩu |
| emailAddress | VARCHAR(255) | Không rỗng | Email |
| phoneNumber | VARCHAR(11) | Không rỗng | Số điện thoại |
| roleID | INT | Không rỗng | Vai trò |
| lastLogin | DATETIME | Không rỗng | Thời điểm |
| createdDate | DATETIME | Không rỗng | Ngày tạo |

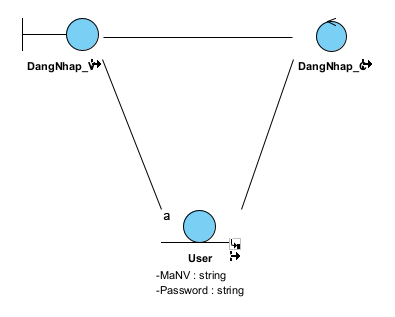
* **Bảng thành viên :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| roleID | INT | Khoá chính | ID Người dùng |
| rolesName | VARCHAR(20) | Không rỗng | Tên vai trò |

* **Bảng thông báo :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| notificationID | INT | Khoá chính | ID thông báo |
| userID | INT | Không rỗng | ID người dùng |
| message | NVARCHAR(200) | Rỗng | Nội dung |
| isRead | BIT | Rỗng | Trạng thái |

* + 1. Thiết kế lớp đối tượng
       1. Biểu đồ lớp VOPC của các ca sử dụng
* Biểu đồ lớp VOPC cho Use Case Đăng nhập :



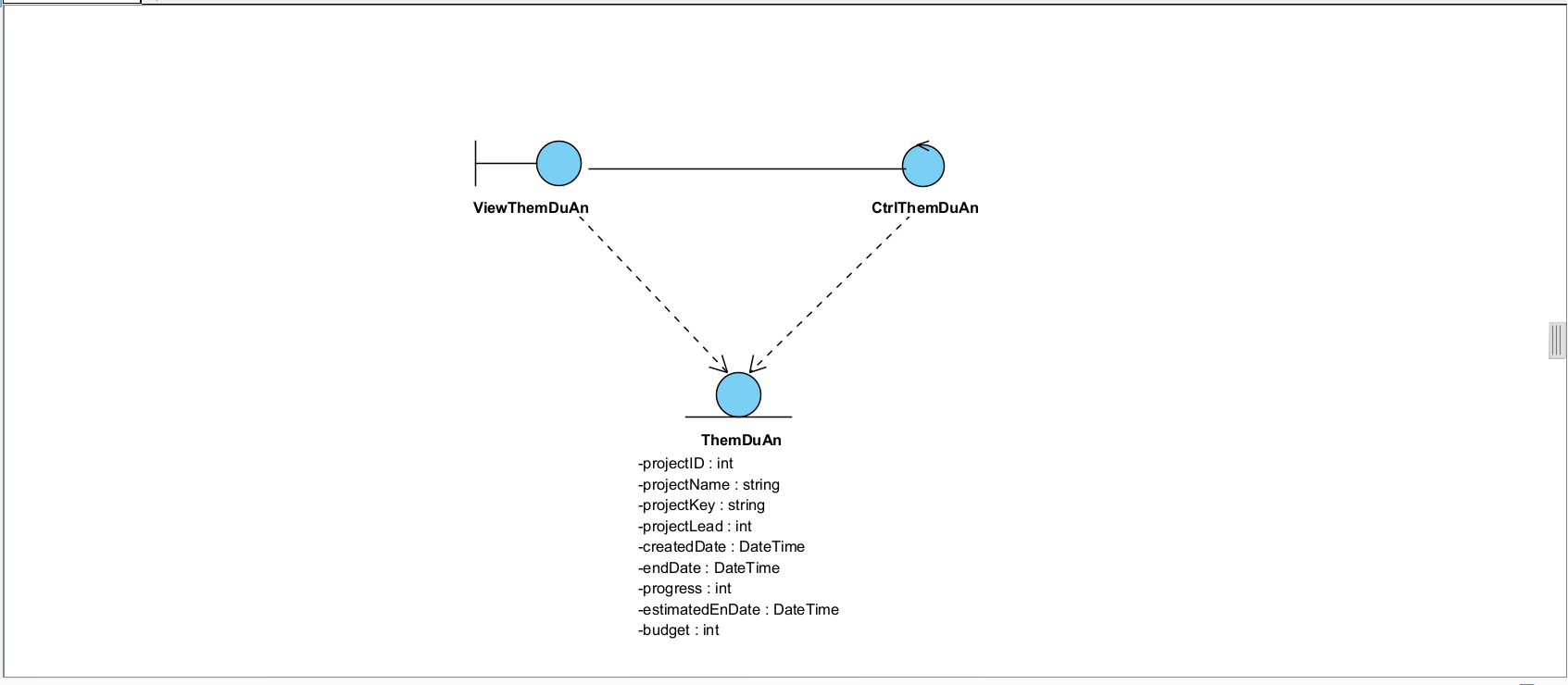
Hình 1. 27 Biểu đồ VOPC Đăng nhập

* Biểu đồ lớp VOPC cho Use Case Quản lý Dự án :



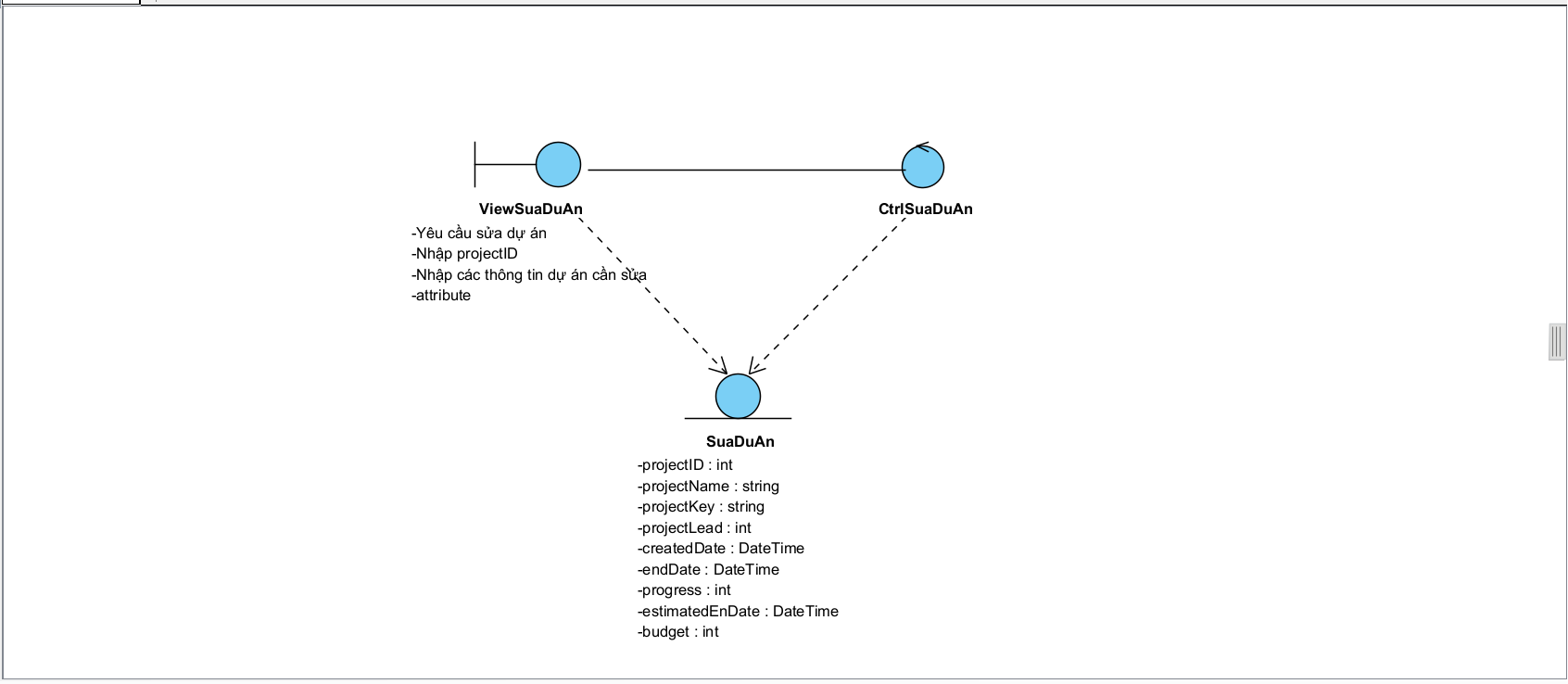
Hình 1. 28 Biểu đồ VOPC Dự án

* Biểu đồ lớp VOPC cho Use Case Thêm Dự án :



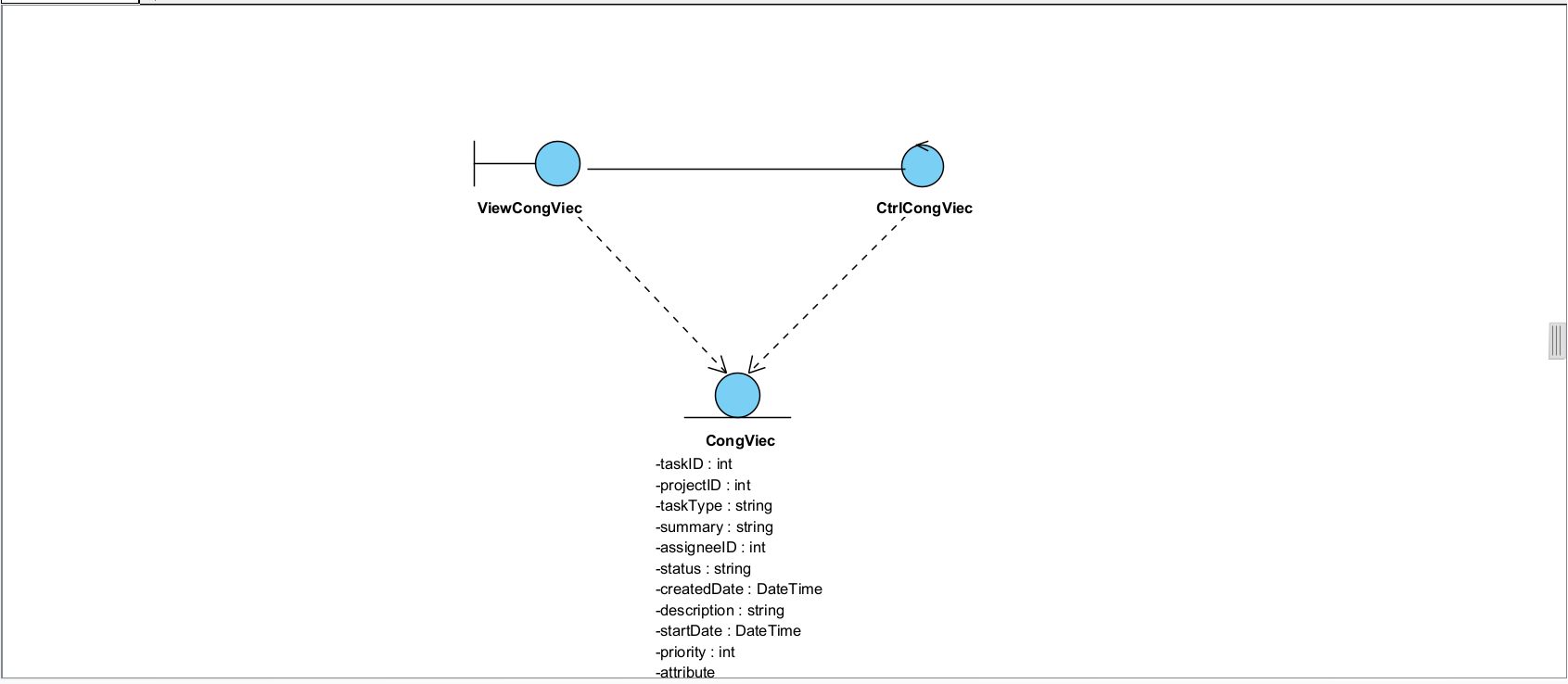
Hình 1. 29 Biểu đồ VOPC Thêm dự án

* Biểu đồ VOPC cho Use Case Sửa Dự án :



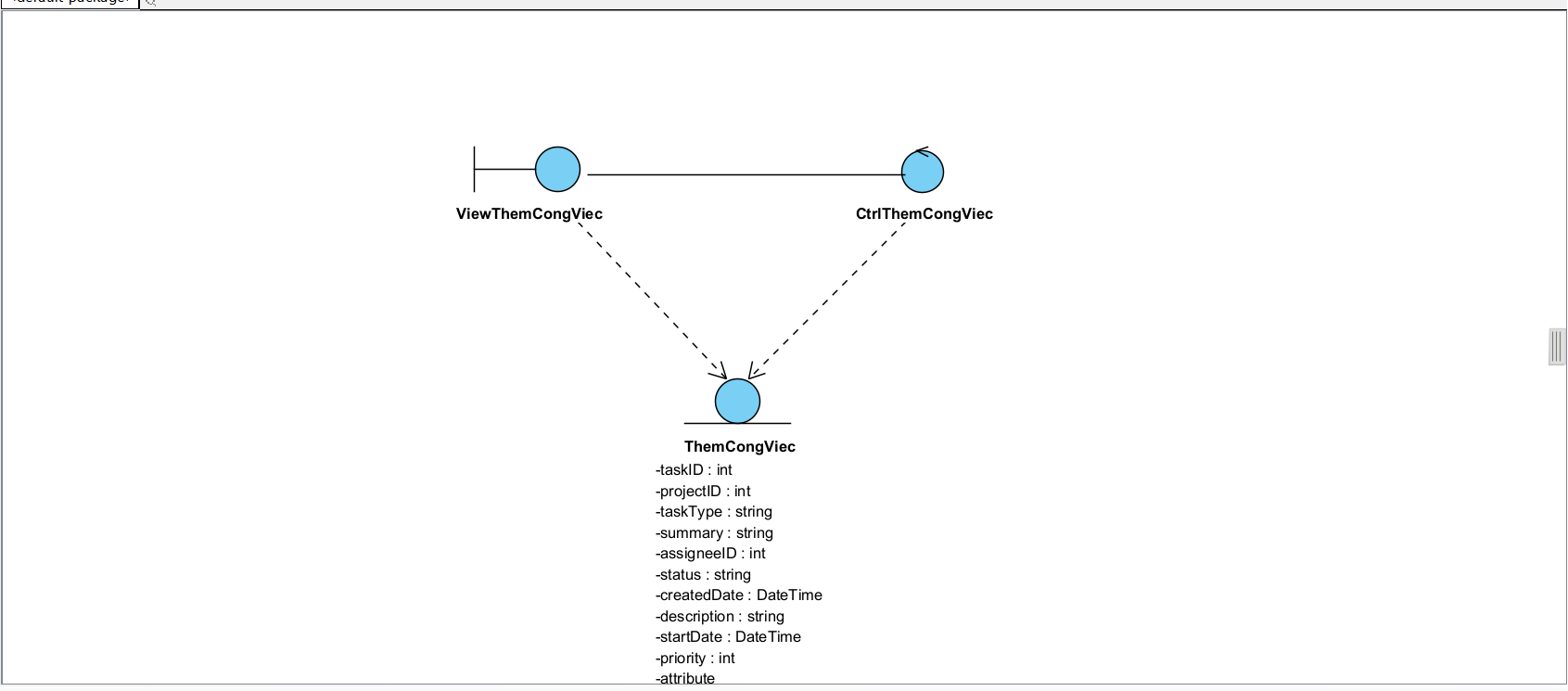
Hình 1. 30 Biểu đồ VOPC Sửa dự án

* Biểu đồ lớp VOPC cho Use Case Quản lý công việc :



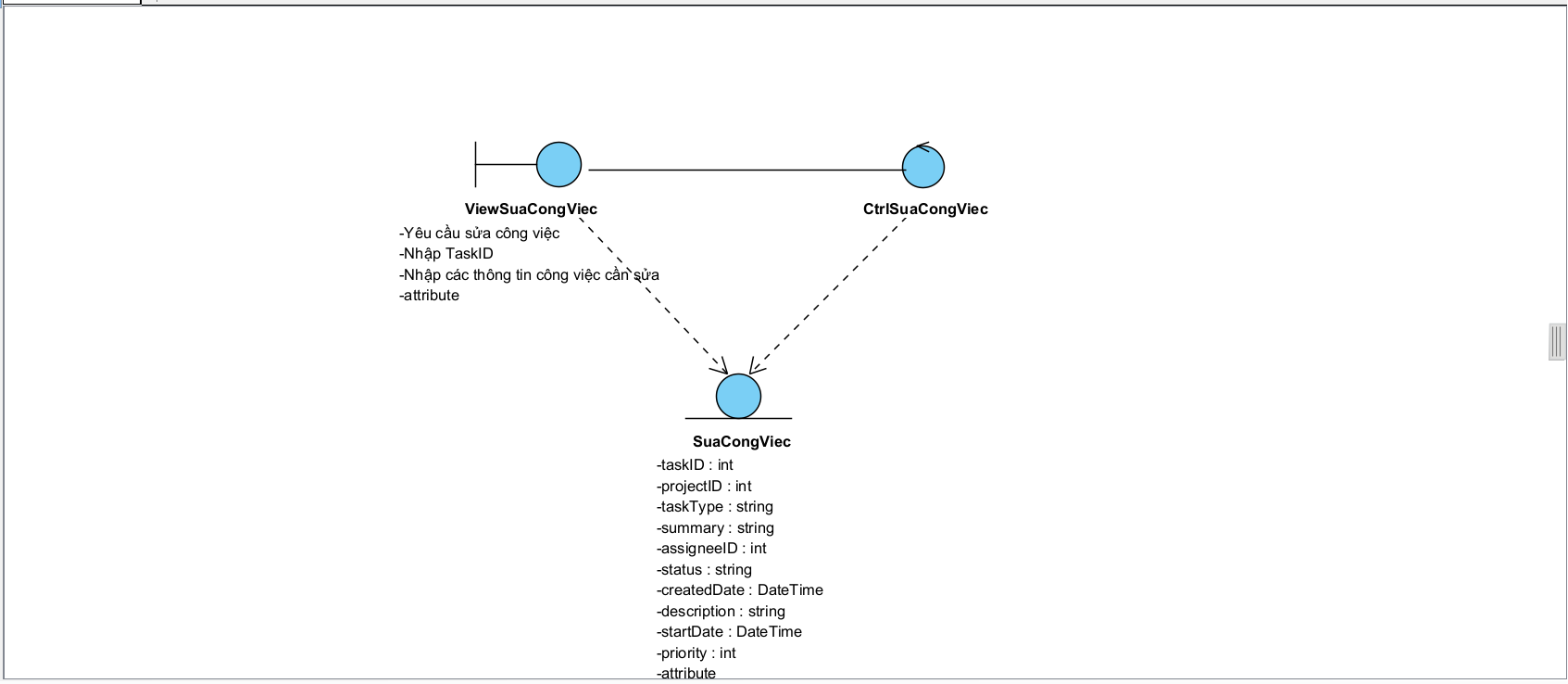
Hình 1. 31 Biểu đồ VOPC Quản lý công việc

* Biểu đồ VOPC của Use Case Thêm công việc :



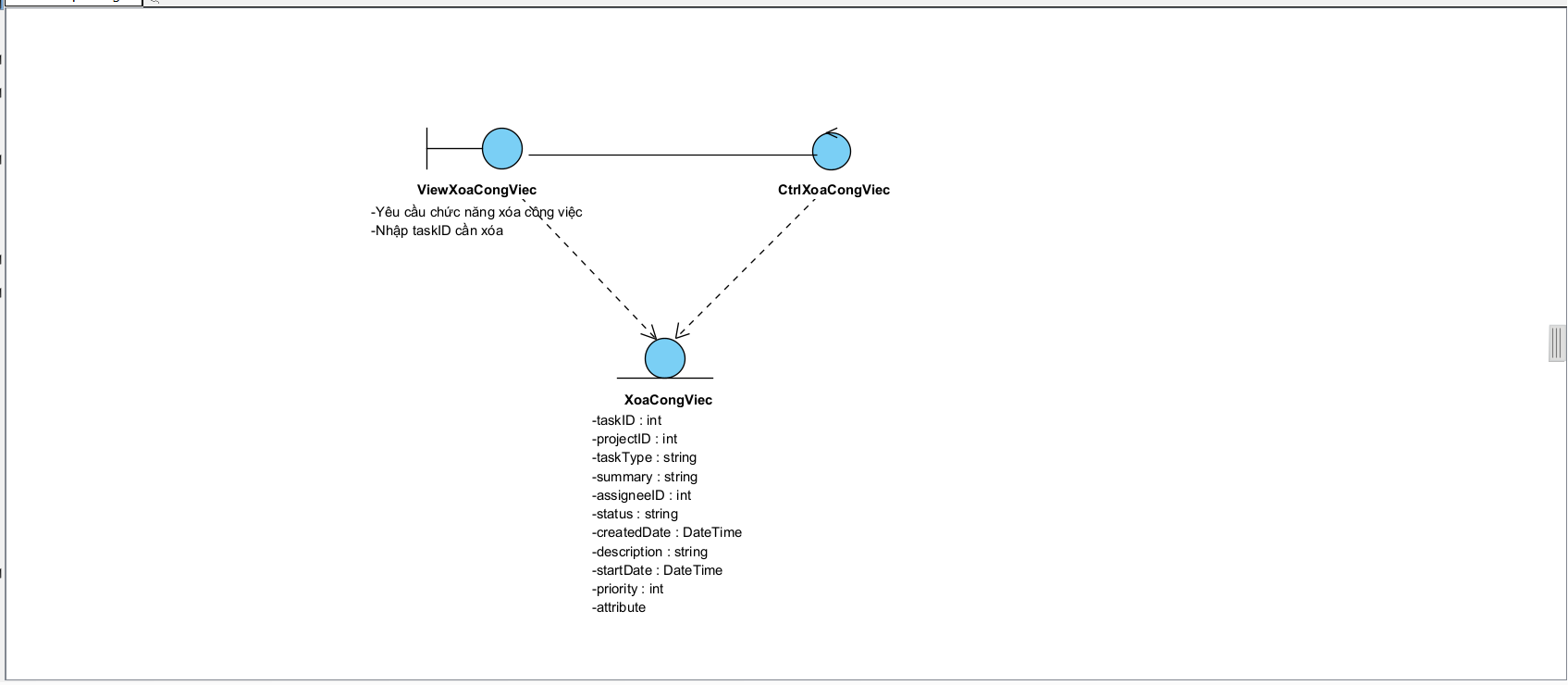
Hình 1. 32 Biểu đồ VOPC Thêm công việc

* Biểu đồ VOPC của Use Case Sửa công việc :



Hình 1. 33 Biểu đồ VOPC Sửa công việc

* Biểu đồ VOPC của Use Case Xoá công việc :



Hình 1. 34 Biểu đồ VOPC Xóa công việc

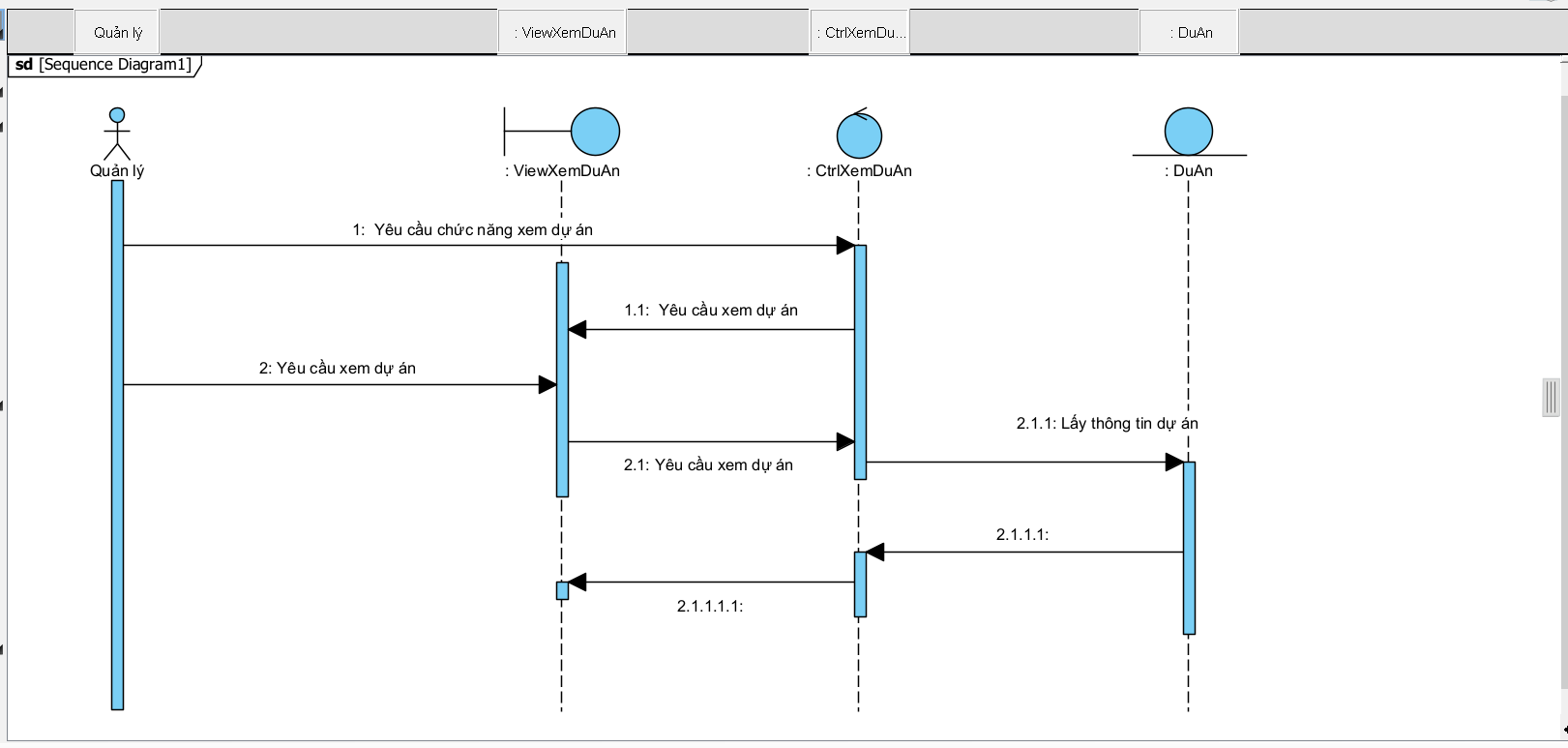
* + - 1. Biểu đồ tuần tự
* Biểu đồ tuần tự cho Use Case Đăng nhập :

A picture containing diagram

Description automatically generated

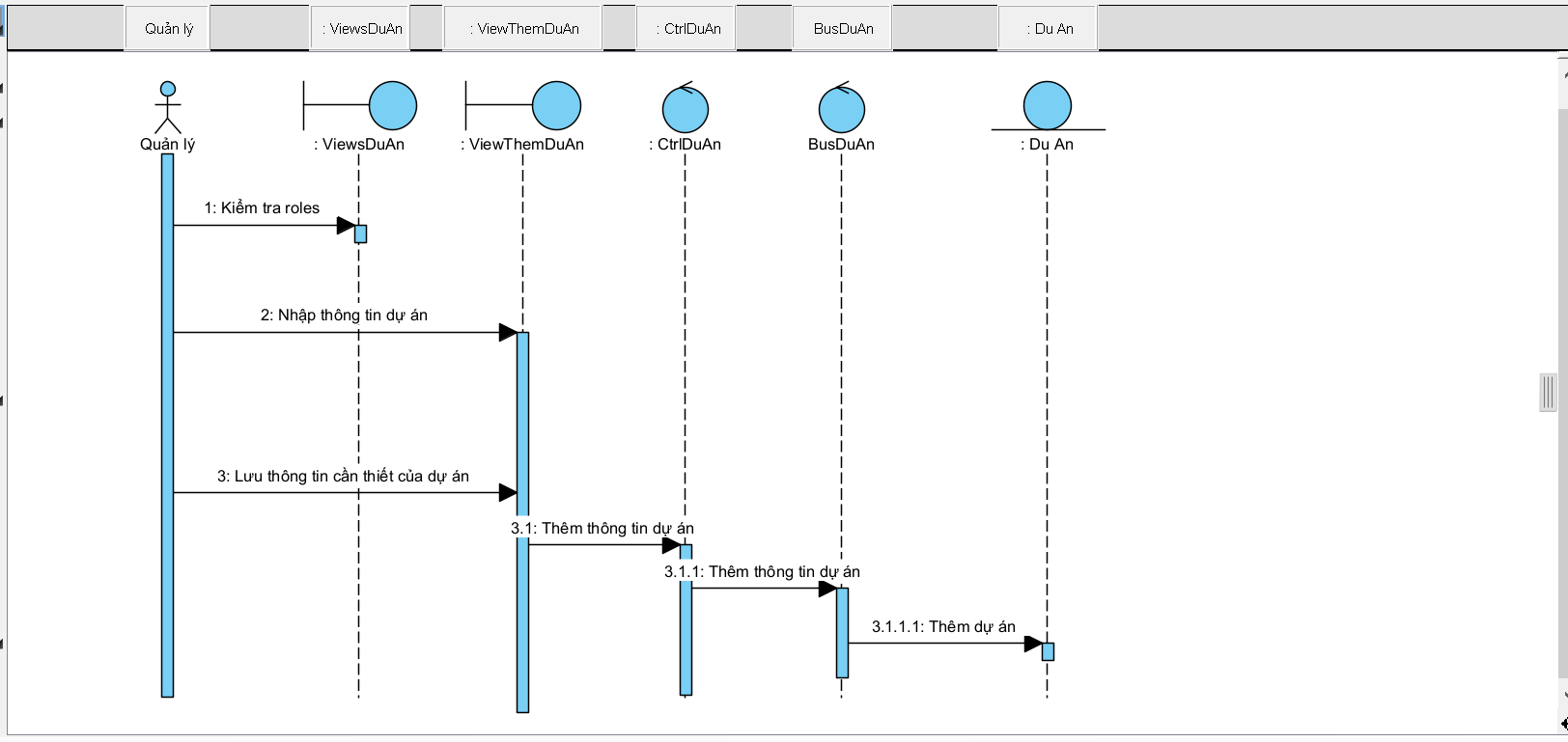
Hình 1. 35 Biểu đồ tuần tự cho use case đăng nhập

* Biểu đồ tuần tự cho use case xem thông tin dự án :



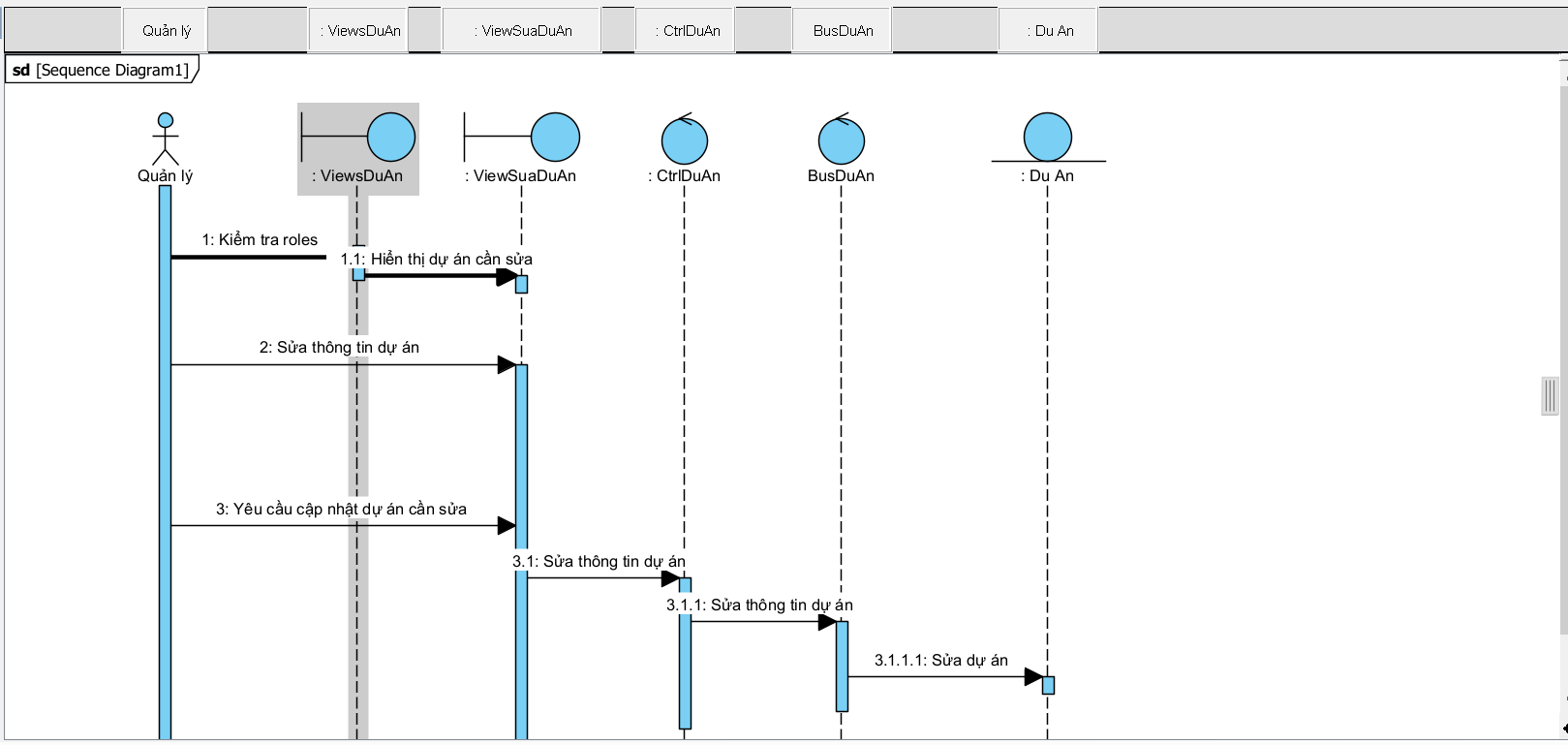
Hình 1. 36 Biểu đồ tuần tự cho chức năng xem thông tin dự án

* Biểu đồ tuần tự cho Use Case Thêm dự án :



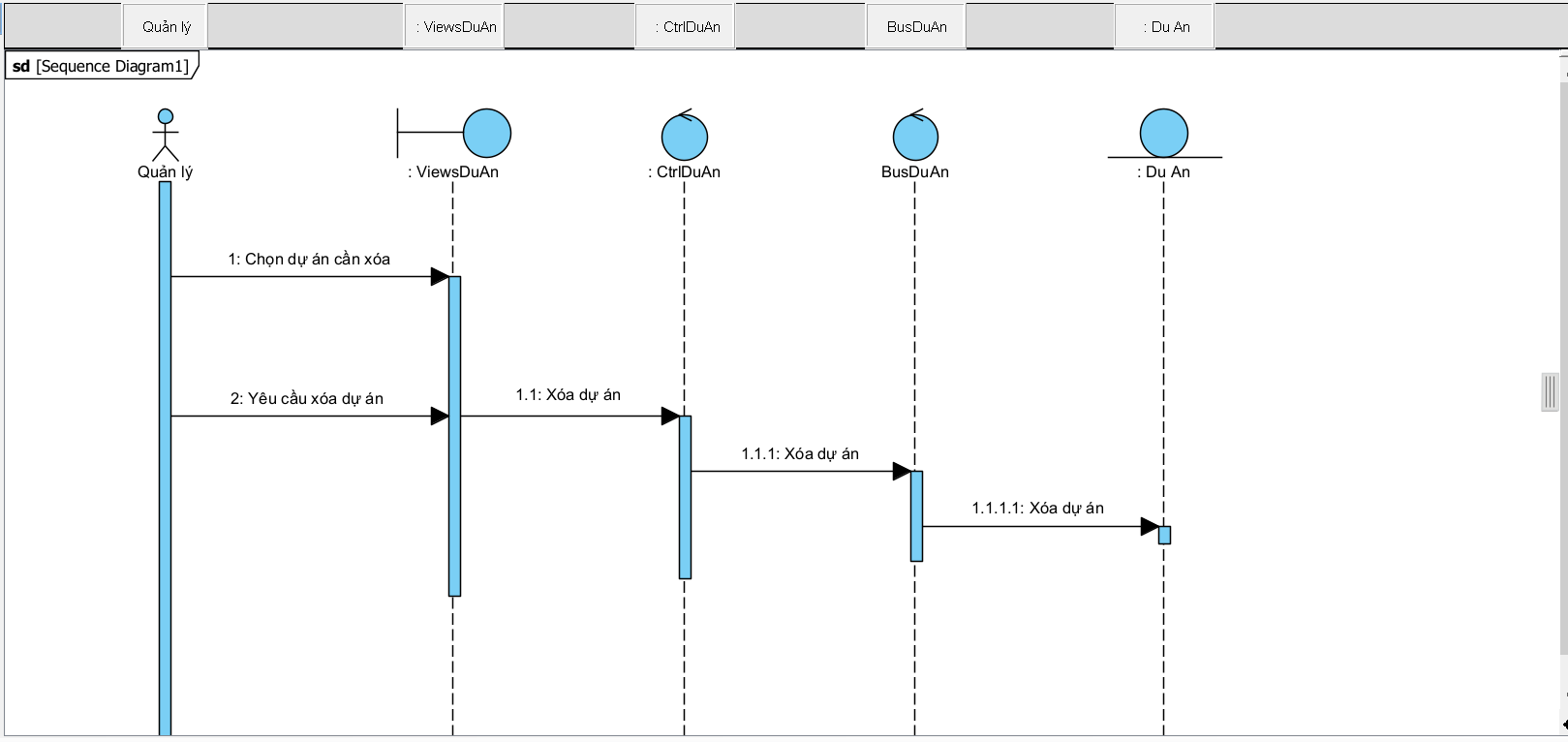
Hình 1. 37 Biểu đồ tuần tự cho use case thêm dự án

* Biểu đồ tuần tự cho Use Case Sửa dự án :



Hình 1. 38 Biểu đồ tuần tự cho use case sửa dự án

* Biểu đồ tuần tự cho Use Case Xoá dự án :



Hình 1. 39 Biểu đồ tuần tự cho use case xóa dự án

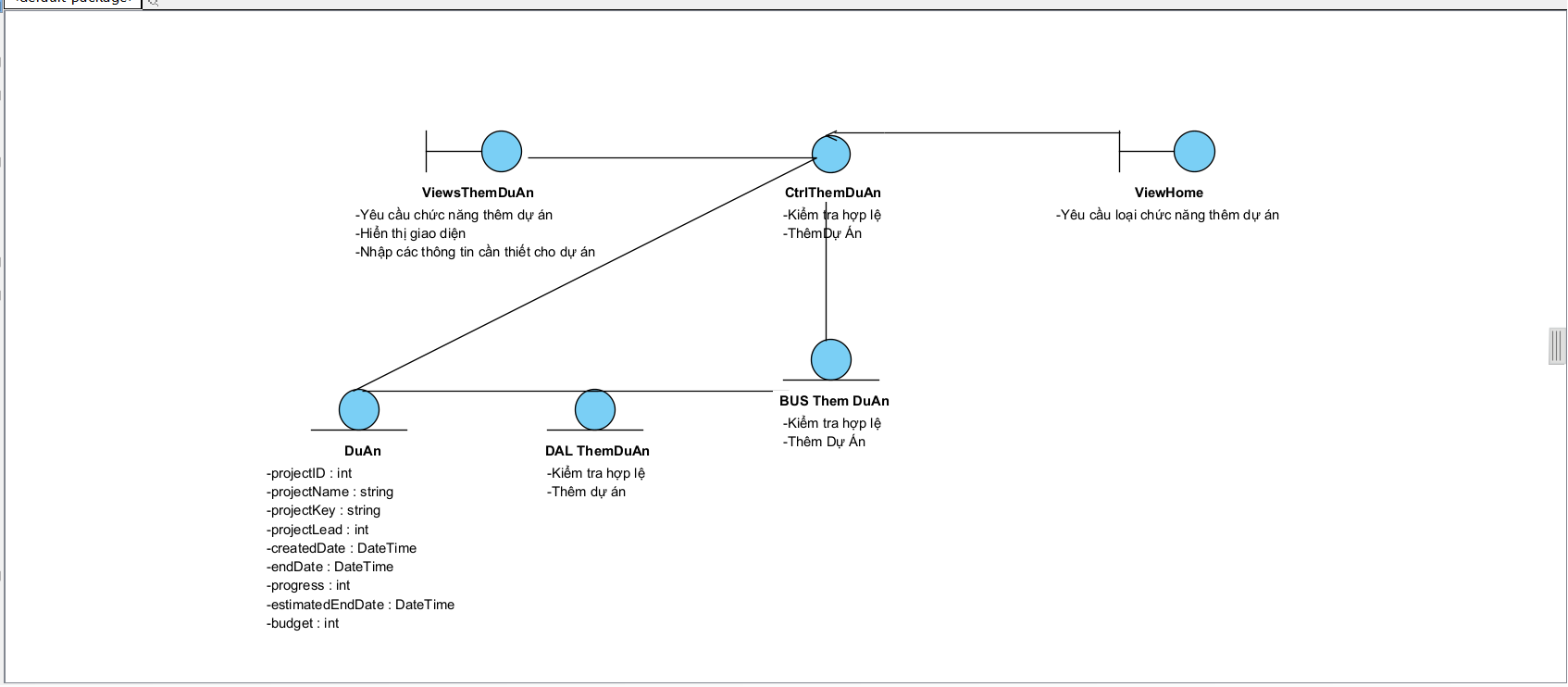
* + - 1. Biểu đồ lớp chi tiết
* Biểu đồ lớp chi tiết cho Use Case Đăng nhập :

Diagram

Description automatically generated

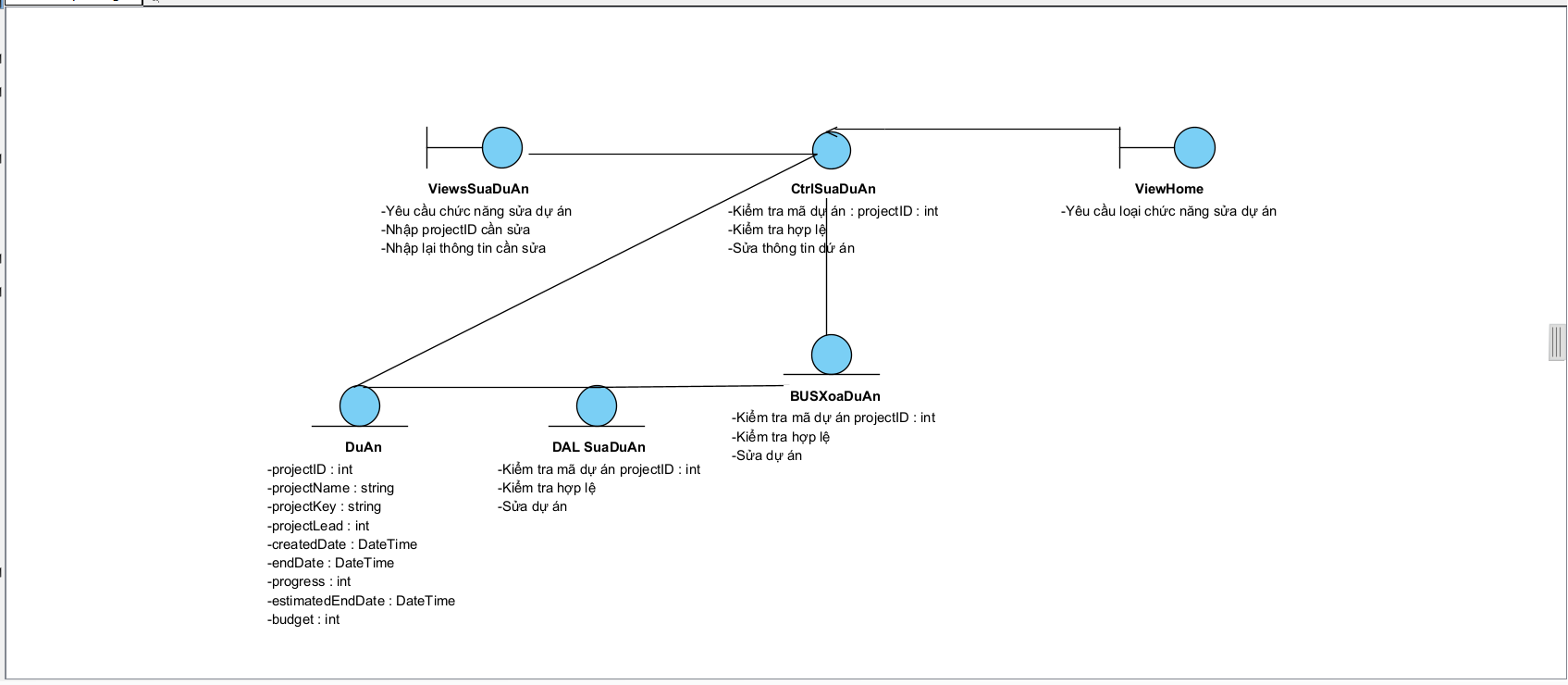
Hình 1. 40 Biểu đồ lớp chi tiết cho use case đăng nhập

* Biểu đồ lớp chi tiết cho use case thêm dự án :



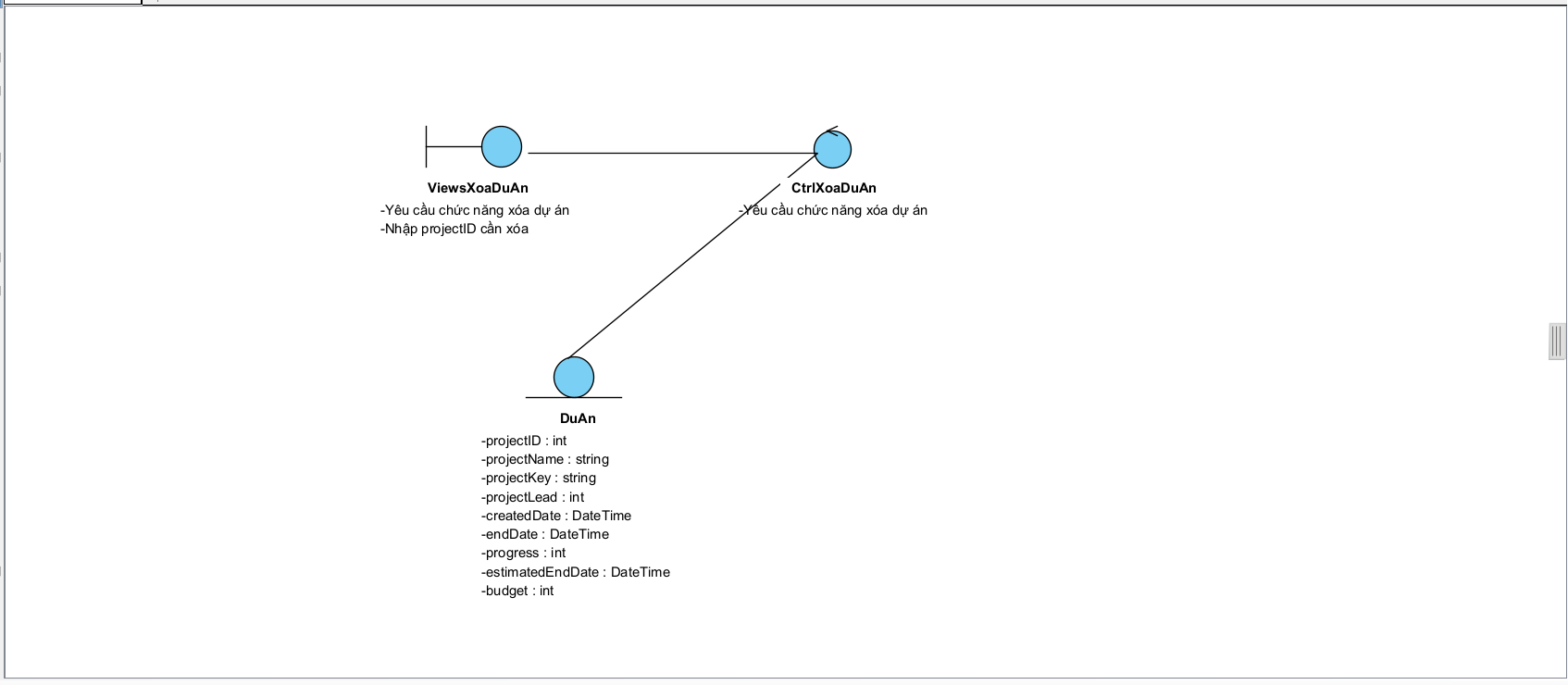
Hình 1. 41 Biểu đồ lớp chi tiết cho use case thêm dự án

* Biểu đồ lớp chi tiết cho use case Sửa dự án :



Hình 1. 42 Biểu đồ lớp chi tiết cho use case sửa dự án

* Biểu đồ lớp chi tiết cho use case xoá dự án :



Hình 1. 43 Biểu đồ lớp chi tiết cho use case xóa dự án

* Biểu đồ lớp chi tiết cho use case xem chi tiết dự án :



Hình 1. 44 Biểu đồ lớp chi tiết cho use case xem thông tin chi tiết dự án

* + 1. Thiết kế giao diện

# TRIỂN KHAI WEBSITE

* 1. Triển khai các chức năng cho phân hệ người dùng

Để xây dựng được các chức năng của trang người dùng theo thiết kế đã được trình bày ở chương 3, đồ án đã sử dụng HTML, CSS, và JavaScript, Jquery, AngularJS để thiết kế giao diện và thao tác dữ liệu các trang theo yêu cầu. Tiếp theo, đồ án sẽ trình bày các kỹ thuật được sử dụng để xây dựng các trang.

* + 1. Trang chủ

a) Phía font end

* Xây dựng bố cục trang Home bằng các thẻ HTML

<Phần này trình bày cách thức sử dụng các thẻ HTML để phân chia được bố cục trang Chủ>.

* Kỹ thuật định dạng bằng CSS

<Phần này trình bày kỹ thuật sử dụng CSS để định các phần của trang Home hiển thị theo bố cục đã thiết kế>.

* Sử dụng Javascript, Jquery, AngularJS để lập trình các chức năng

<Phần này trình bày thuật toán hoặc các bước xử lý để có được các chức năng theo yêu cầu, ví dụ chức năng chọn mua hàng, tìm kiếm, ….>

b) Phía backend

* Triển khai các lớp tầng DataAccess

<Phần này trình bày cách thức xử lý để thực hiện được các thao tác trên cơ sở dữ liệu để phục vụ các xử lý ở trang chủ>

* Triển khai lớp tầng Bussiness

<Phần này trình bày về cách thức cài đặt lớp ở tầng Bussiness để đáp ứng được các xử lý được gọi ở tầng Controller>

* Triển khai lớp tầng Controller

<Phần này trình bày về cách thức cài đặt các Action để đáp ứng được các xử lý được gọi ở tầng View cho trang chủ>

* + 1. Trang Giày thể thao
  + Trang chủ (index):

A person in a turtleneck

Description automatically generated with low confidence

Hình 2-73: Header và slideshow

Mô tả:

* Header gồm 3 phần: Phần trên gồm địa chỉ, các liên kết tài khoản mạng xã hội, tài khoản, dịch vụ, hỗ trợ và giỏ hàng. Phần giữa gồm logo shop, menu. Phần dưới là các dịch vụ của shop
* Tiếp theo là slideshow gồm các hình ảnh về lợi ích của việc đi xe đạp và dịch vụ trả góp, mỗi hình ảnh trình chiếu sau 6s.

Phân loại sản phẩm:

Contact và footer:

A screenshot of a computer

Description automatically generated with medium confidence

Hình 2-80: Contact và footer

* + Trang sản phẩm:

A screenshot of a computer game

Description automatically generated with low confidence

Hình 2-81: Trang sản phẩm giày thể thao

Mô tả: Người dùng có thể thêm sản phẩm vào giỏ hàng khi muốn mua. Ngoài ra người dùng có thể sắp xếp các sản phẩm theo tên, sản phẩm nổi bật, theo giá.

* + Chi tiết sản phẩm:

A screenshot of a computer

Description automatically generated with low confidence

Hình 2-82: Trang chi tiết sản phẩm

Mô tả: Giúp khách hàng có thêm thông tin về các chi tiết, thông số kỹ thuật của sản phẩm và có thể chọn màu sắc, số lượng muốn mua và thêm vào giỏ hàng để tiến hành mua sản phẩm

* + Giỏ hàng:

A screenshot of a website

Description automatically generated with medium confidence

Hình 2-84: Trang giỏ hàng

Gồm danh sách sản phẩm được thêm vào giỏ hàng, nút xoá giỏ hàng, cập nhật giỏ hàng và nút thanh toán, combo box cho phép chọn số lượng, bên phải là số tiền mà người dùng cần thanh toán.

* + Trang thanh toán:

A screenshot of a computer

Description automatically generated with medium confidence

Hình 2-85: Trang thanh toán

Gồm danh sách các sản phẩm sau khi đã được thêm vào giỏ hàng và ấn nút thanh toán. Trang thanh toán sẽ hiện lên với yêu cầu ng dùng nhập thông tin địa chỉ liên hệ và thông tin đơn hàng.

* + Trang tài khoản

Graphical user interface, application

Description automatically generated

Hình 2-86: Trang tài khoản (đăng nhập )

Người dùng có thể đăng nhập vào khi đã có tài khoản để lưu trữ các thông tin và các sản phẩm muốn mua

Graphical user interface

Description automatically generated

Hình 2-87: Trang tài khoản(Đăng kí)

Người dùng có thể đăng kí tài khoản để sử dụng trang web 1 cách thuận tiện hơn

* + 1. Trang Quản lý giỏ
* Giỏ hàng:

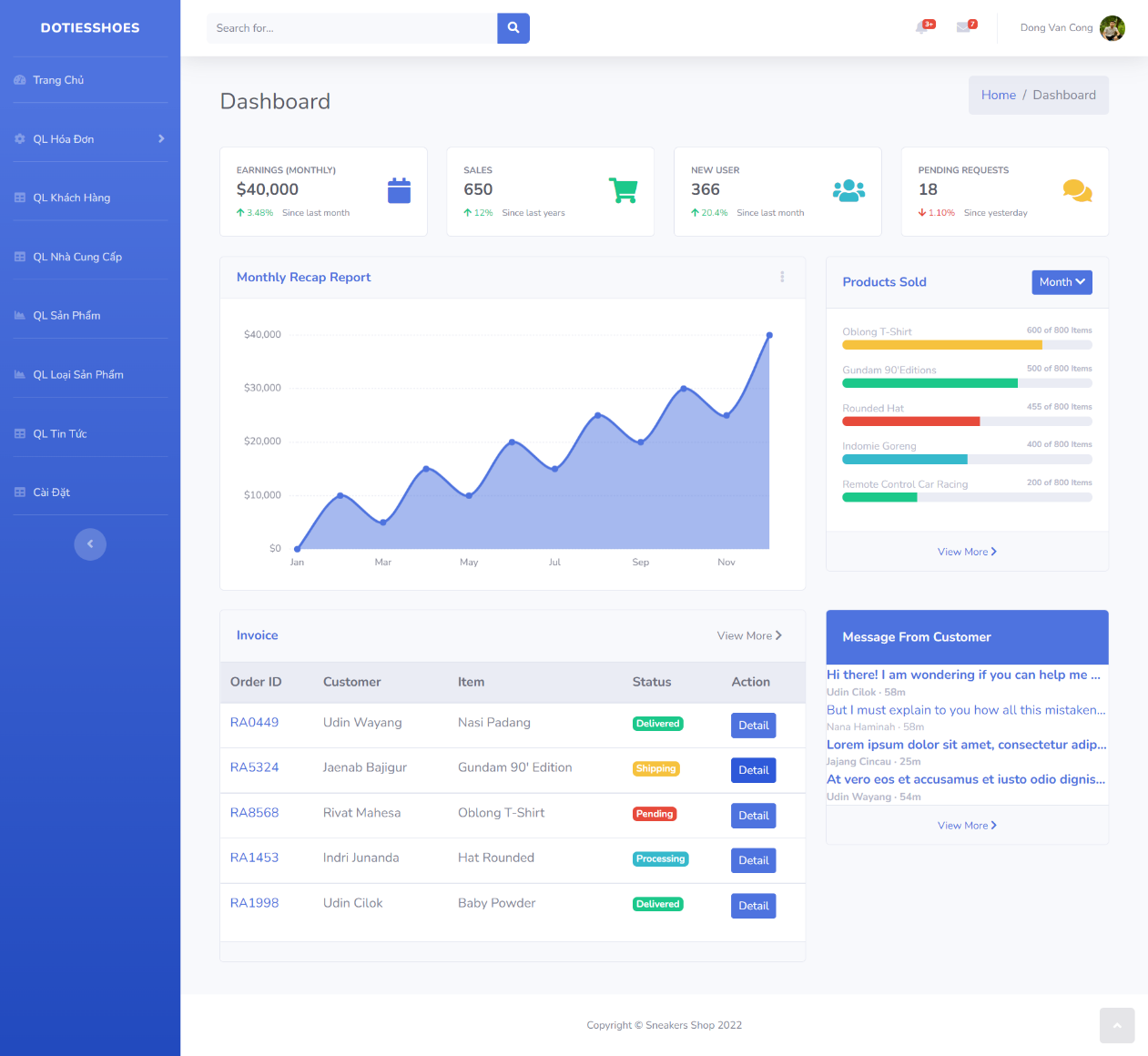
A screenshot of a website

Description automatically generated with medium confidence

Hình 2-84: Trang giỏ hàng

Gồm danh sách sản phẩm được thêm vào giỏ hàng, nút xoá giỏ hàng, cập nhật giỏ hàng và nút thanh toán, combo box cho phép chọn số lượng, bên phải là số tiền mà người dùng cần thanh toán.

* 1. Triển khai các chức năng cho phân hệ quản trị nội dung
* Trang quản trị:
  + Trang tổng quan:



Hình 2-89: Trang tổng quan

* + Trang loại sản phẩm

Graphical user interface, text, email

Description automatically generated

***Hình 2-90: Trang loại sản phẩm***

Trang này cung cấp cho người quản trị danh sách thông tin loại sản phẩm của cửa hàng, cùng với đó là các chức năng thêm, sửa, xoá loại sản phẩm

* + Trang sản phẩm

Graphical user interface, text, application

Description automatically generated

***Hình 2-91: Trang sản phẩm***

Trang này cung cấp cho người quản trị danh sách thông tin sản phẩm của cửa hàng và các chức năng thêm, sửa, xoá sản phẩm

* + Trang nhà cung cấp

Graphical user interface, text, application, email

Description automatically generated

***Hình 2-92: Trang nhà cung cấp***

Người quản lý có thể dễ dàng quản lý các nhà cung cấp ở trang này

* + Trang khách hàng

Graphical user interface, text, application, email

Description automatically generated

***Hình 2-93: Trang khách hàng***

Trang này cung cấp cho người quản lý danh sách nhưng khách hàng đã từng mua sản phẩm và các chức năng thêm, sửa, xoá khách hàng

* + Trang nhân viên

Graphical user interface

Description automatically generated

***Hình 2-94: Trang nhân viên***

Trang này cung cấp cho người quản lý danh sách thông tin các nhân viên cùng với đó là các chức năng thêm, sửa, xoá

* 1. Kiểm thử và triển khai ứng dụng
     1. Kiểm thử

Graphical user interface, text, application

Description automatically generated

***Hình 4-4: Kiểm thử - trang chi tiết sản phẩm***

Input: Màu sắc: Trắng, hình ảnh: Đường dẫn, số lượng: 2, tên sản phẩm: XE ĐẠP 20”…

* Thực hiện:
* Khách hàng chọn màu sắc và số lượng mong muốn, sau đó click vào button “Thêm vào giỏ hàng”.
* Trang web sẽ alert lên 1 thông báo “Đã thêm vào giỏ hàng thành công”.
* Các thông tin sản phẩm sẽ được lưu vào localStorage và hiển thị trong giỏ hàng.

Graphical user interface, text, application

Description automatically generated

***Hình 4-5: Output kiểm thử - trang chi tiết sản phẩm***

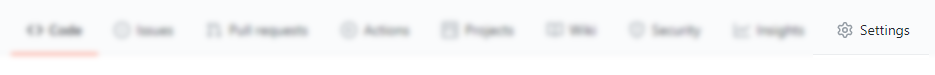
* + 1. Đóng gói ứng dụng

# Đầu tiên cần đẩy bài tập lớn lên Github.

# Tiếp theo truy cập [*http://github.com*](http://github.com), đăng nhập. Chọn bài tập lớn



# Chuyển sang “Setting” trên thanh công cụ



# Tìm đến “Github Pages” và chọn “Check it out here!”

Graphical user interface, application

Description automatically generated

# Ở option trong “Sourse” chọn brand là master sau đó nhấn nút “Save”

Graphical user interface, text, application, email

Description automatically generated

# Khi có thông báo như dưới đây thì đã thành công và có thể truy cập bằng đường dẫn bên dưới.

Graphical user interface, text

Description automatically generated

Lưu ý: Cần đặt trang index.html ra bên ngoài cùng. Nếu không thì phải truy cập đúng vào thư mục chứa trang index. Ví dụ:



* + 1. Triển khai ứng dụng

# Phần cứng:

Máy tính có thể truy cập internet.

Thanh ghi RAM trên 4G.

# Phần mềm:

Trình duyệt phổ biến như: Google Chrome, Egde, FireFox…

# KẾT LUẬN

* Kết quả đạt được:
* Kỹ năng lập trình API, Angular.
* Kiến thức và kỹ năng cấu hình Server và triển khai Website.
* Những hạn chế của đề tài:
* Tốc độ Website chưa nhanh.
* Bảo mật chưa cao.
* Hướng phát triển của đề tài:
* Giải quyết những hạn chế của đề tài.

TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Trường ĐHSPKT Hưng Yên, Lập trình web API |
| [2] | HIENLTH , Web-API , YouTube |

x