

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**

**NGHIÊN CỨU VÀ PHÁT TRIỂN HỆ THỐNG QUẢN LÝ KỸ NĂNG CỦA NHÂN VIÊN**

# MỤC LỤC

[MỤC LỤC 1](#_Toc5920693)

[DANH MỤC CÁC HÌNH VẼ 4](#_Toc5920694)

[DANH MỤC CÁC BẢNG BIỂU 8](#_Toc5920695)

[LỜI MỞ ĐẦU 10](#_Toc5920696)

[CHƯƠNG 1 : GIỚI THIỆU 11](#_Toc5920697)

[1.1 Tổng quan 11](#_Toc5920698)

[1.2 Mục tiêu đề tài 12](#_Toc5920699)

[1.3 Phạm vi đề tài 12](#_Toc5920700)

[1.4 Mô tả yêu cầu chức năng 13](#_Toc5920701)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 15](#_Toc5920702)

[2.1 Spring Boot 15](#_Toc5920703)

[2.1.1 Kiến trúc và các thành phần của Spring Boot: 15](#_Toc5920704)

[2.1.2 Lợi ích sử dụng Spring Boot : 17](#_Toc5920705)

[2.1.3 Các dependency sử dụng trong project: 19](#_Toc5920706)

[2.2 Angular 20](#_Toc5920707)

[2.2.1. Giới thiệu về Angular 20](#_Toc5920708)

[2.2.2. Tổng quan về kiến trúc 20](#_Toc5920709)

[2.2.3. Angular CLI(Command-line Interface) 23](#_Toc5920710)

[2.2.4. Các chức năng chính của Angular 23](#_Toc5920711)

[2.2.5. So sánh Angular với các công nghệ front-end khác: 24](#_Toc5920712)

[2.3 MySQL 25](#_Toc5920713)

[2.3.1 Cơ sở dữ liệu là gì? 25](#_Toc5920714)

[***2.3.2*** ***Thuật ngữ RDBMS*** 26](#_Toc5920715)

[***2.3.3*** ***MySQL và lợi ích khi sử dụng MySQL*** 26](#_Toc5920716)

[2.4 Maven 27](#_Toc5920717)

[***2.4.1*** ***Tổng quan về Apache Maven*** 27](#_Toc5920718)

[***2.4.2*** ***Lợi ích khi sử dụng Apache Maven:*** 28](#_Toc5920719)

[***2.4.3*** ***Ứng dụng Apache Maven vào dự án:*** 28](#_Toc5920720)

[2.5 Nghiệp vụ quản lý nhân sự 30](#_Toc5920721)

[CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ 32](#_Toc5920722)

[3.1 Mô tả tổng quan ứng dụng 32](#_Toc5920723)

[3.1.1 Mô hình Usecase tổng quát 32](#_Toc5920724)

[3.1.2 Danh sách các tác nhân và mô tả 33](#_Toc5920725)

[3.1.3 Danh sách Usecase và mô tả 33](#_Toc5920726)

[3.2 Đặc tả các yêu cầu chức năng 37](#_Toc5920727)

[3.2.1 UC001 Ho tro 37](#_Toc5920728)

[3.2.2 UC001a Tai tai lieu huong dan 40](#_Toc5920729)

[3.2.3 UC001b Xuat ho so ca nhan 42](#_Toc5920730)

[3.2.4 UC001c Upload ho so ca nhan 45](#_Toc5920731)

[3.2.5 UC002 Xem thong tin ca nhan 46](#_Toc5920732)

[3.2.6 UC002a Cap nhat thong tin ca nhan 48](#_Toc5920733)

[3.2.7 UC002b Cap nhat summary ca nhan 51](#_Toc5920734)

[3.2.8 UC002c Cap nhat avatar ca nhan 54](#_Toc5920735)

[3.2.9 UC003 Them chung chi 57](#_Toc5920736)

[3.2.10 UC004 Xem thong tin chung chi 60](#_Toc5920737)

[3.2.11 UC004a Xoa chung chi 62](#_Toc5920738)

[3.2.12 UC004b Cap nhat chung chi 65](#_Toc5920739)

[3.2.13 UC005 Them kinh nghiem lam viec, UC005a Them vai tro vao kinh nghiem lam viec 69](#_Toc5920740)

[3.2.14 UC006 Xem thong tin kinh nghiem lam viec, UC006a Xem thong tin vai tro trong kinh nghiem lam viec 74](#_Toc5920741)

[3.2.15 UC006a1 Cap nhat thong tin ve vai tro trong kinh nghiem lam viec 76](#_Toc5920742)

[3.2.16 UC006a2 Xoa vai tro trong kinh nghiem lam viec 80](#_Toc5920743)

[3.2.17 UC006b Cap nhat thong tin kinh nghiem lam viec 84](#_Toc5920744)

[3.2.18 UC006c Xoa kinh nghiem lam viec 88](#_Toc5920745)

[3.2.19 UC007 Them ngoai ngu 90](#_Toc5920746)

[3.2.20 UC008 Xem thong tin ngoai ngu 93](#_Toc5920747)

[3.2.21 UC008a Xoa ngoai ngu 95](#_Toc5920748)

[3.2.22 UC008b Cap nhat ngoai ngu 98](#_Toc5920749)

[3.2.23 UC009 Them ky nang ky thuat 102](#_Toc5920750)

[3.2.24 UC0010 Xem thong tin ky nang ky thuat 105](#_Toc5920751)

[3.2.25 UC010a Xoa ky nang ky thuat 107](#_Toc5920752)

[3.2.26 UC010b Cap nhat ky nang ky thuat 110](#_Toc5920753)

[3.2.27 UC0011 Them hoc van 113](#_Toc5920754)

[3.2.28 UC0012 Xem thong tin hoc van 116](#_Toc5920755)

[3.2.29 UC012a Xoa hoc van 118](#_Toc5920756)

[3.2.30 UC012b Cap nhat hoc van 121](#_Toc5920757)

[3.2.31 UC013 Cap nhat trang thai ho so nhan vien 125](#_Toc5920758)

[3.2.32 UC014 Xuat danh sách nhan vien 125](#_Toc5920759)

[3.2.33 UC0015 Quan ly ho so nhan vien 125](#_Toc5920760)

[3.2.34 UC016 Them du an 125](#_Toc5920761)

[3.2.35 UC017 Xem du an cua cong ty 125](#_Toc5920762)

[3.2.36 UC017a Cap nhat thong tin du an 125](#_Toc5920763)

[3.2.37 UC018 Tim kiem 125](#_Toc5920764)

[3.2.38 UC019 Thong ke 125](#_Toc5920765)

[3.2.39 UC020 Xem lich su cap nhat 125](#_Toc5920766)

[3.2.40 UC021 Quan ly du lieu hien thi cua he thong 125](#_Toc5920767)

[3.2.41 UC021a Them du lieu 125](#_Toc5920768)

[3.2.42 UC021b Xoa du lieu 125](#_Toc5920769)

[3.2.43 UC021c Cap nhat du lieu 125](#_Toc5920770)

[3.3 Mô hình ERD 126](#_Toc5920771)

[3.4 Class diagram 127](#_Toc5920772)

[CHƯƠNG 4: HIỆN THỰC 127](#_Toc5920773)

[4.1 Tiêu đề mục 127](#_Toc5920774)

[4.1.1 Tiêu đề tiểu mục 127](#_Toc5920775)

[4.1.2 Tiêu đề tiểu mục 127](#_Toc5920776)

[4.2 Tiêu đề mục 128](#_Toc5920777)

[4.2.1 Tiêu đề tiểu mục 128](#_Toc5920778)

[4.2.2 Tiêu đề tiểu mục 128](#_Toc5920779)

[CHƯƠNG 5: KẾT LUẬN 129](#_Toc5920780)

[5.1 Kết quả đạt được 129](#_Toc5920781)

[5.2 Hạn chế của đồ án 129](#_Toc5920782)

[5.3 Hướng phát triển 129](#_Toc5920783)

[TÀI LIỆU THAM KHẢO 129](#_Toc5920784)

[PHỤ LỤC 131](#_Toc5920785)

# DANH MỤC CÁC HÌNH VẼ

[Hình 2‑1 Kiến trúc tổng thể Spring FrameWork 16](#_Toc5920430)

[Hình 2‑2 Mô hình Spring Boot trong spring io 18](#_Toc5920431)

[Hình 2‑3 Module mẫu của một Angular App 21](#_Toc5920432)

[Hình 2‑4 Một Angular Component mẫu với metadata khai báo template và service provider 21](#_Toc5920433)

[Hình 2‑5 Một Angular Service mẫu với metadata với khai báo cho phép cung cấp service tại root application 22](#_Toc5920434)

[Hình 2‑6 Kiến trúc cơ bản của một Angular App 22](#_Toc5920435)

[Hình 2‑7 Lược đồ miêu tả luồng dữ liệu qua lại giữa DOM và Component với các cú pháp tương ứng 24](#_Toc5920436)

[Hình 2‑8 File POM cha hỗ trợ quản lý thông tin dự án, thư viện và phân tách module 29](#_Toc5920437)

[Hình 2‑9 Cấu hình file POM hỗ trợ tự động cài đặt môi trường và build tự động cho Front-end 30](#_Toc5920438)

[Hình 2‑10 Maven tự động build mã nguồn front-end Angular tự động trong quá trình đóng gói dự án. 30](#_Toc5920439)

[Hình 3‑1Usecase tổng quát 32](#_Toc5920440)

[Hình 3‑2 Activity diagram UC001 39](#_Toc5920441)

[Hình 3‑3 Sequence diagram UC001 39](#_Toc5920442)

[Hình 3‑4 Activity diagram UC001a 41](#_Toc5920443)

[Hình 3‑5 Sequence diagram UC001a 42](#_Toc5920444)

[Hình 3‑6 Activity diagram UC001b 44](#_Toc5920445)

[Hình 3‑7 Sequence diagram UC001b 44](#_Toc5920446)

[Hình 3‑8 Activity diagram UC002 47](#_Toc5920447)

[Hình 3‑9 Sequence diagram UC002 48](#_Toc5920448)

[Hình 3‑10 Activity diagram UC002a 50](#_Toc5920449)

[Hình 3‑11 Sequence diagram UC002a 51](#_Toc5920450)

[Hình 3‑12 Activity diagram UC002b 53](#_Toc5920451)

[Hình 3‑13 Sequence diagram UC002b 54](#_Toc5920452)

[Hình 3‑14 Activity diagram UC002c 56](#_Toc5920453)

[Hình 3‑15 Sequence diagram UC002c 57](#_Toc5920454)

[Hình 3‑16 Activity diagram UC003 59](#_Toc5920455)

[Hình 3‑17 Sequence diagram UC003 60](#_Toc5920456)

[Hình 3‑18 Activity diagram UC004 61](#_Toc5920457)

[Hình 3‑19 Sequence diagram UC004 62](#_Toc5920458)

[Hình 3‑20 Activity diagram UC004a 64](#_Toc5920459)

[Hình 3‑21 Sequence diagram UC004a 65](#_Toc5920460)

[Hình 3‑22 Activity diagram UC004b 67](#_Toc5920461)

[Hình 3‑23 Sequence diagram UC004b 68](#_Toc5920462)

[Hình 3‑24 Activity diagram UC005 72](#_Toc5920463)

[Hình 3‑25 Sequence diagram UC005 74](#_Toc5920464)

[Hình 3‑26 Activity diagram UC006 75](#_Toc5920465)

[Hình 3‑27 Sequence diagram UC006 76](#_Toc5920466)

[Hình 3‑28 Activity diagram UC006a1 78](#_Toc5920467)

[Hình 3‑29 Sequence diagram UC006a1 80](#_Toc5920468)

[Hình 3‑30 Activity diagram UC006a2 82](#_Toc5920469)

[Hình 3‑31 Sequence diagram UC006a2 83](#_Toc5920470)

[Hình 3‑32 Activity diagram UC006b 86](#_Toc5920471)

[Hình 3‑33 Sequence diagram UC006b 87](#_Toc5920472)

[Hình 3‑34 Activity diagram UC006c 89](#_Toc5920473)

[Hình 3‑35 Sequence diagram UC006c 90](#_Toc5920474)

[Hình 3‑36 Activity diagram UC007 92](#_Toc5920475)

[Hình 3‑37 Sequence diagram UC007 93](#_Toc5920476)

[Hình 3‑38 Activity diagram UC008 94](#_Toc5920477)

[Hình 3‑39 Sequence diagram UC008 95](#_Toc5920478)

[Hình 3‑40 Activity diagram UC008a 97](#_Toc5920479)

[Hình 3‑41 Sequence diagram UC008a 98](#_Toc5920480)

[Hình 3‑42 Activity diagram UC008b 100](#_Toc5920481)

[Hình 3‑43 Sequence diagram UC008b 101](#_Toc5920482)

[Hình 3‑44 Activity diagram UC009 104](#_Toc5920483)

[Hình 3‑45 Sequence diagram UC009 105](#_Toc5920484)

[Hình 3‑46 Activity diagram UC010 106](#_Toc5920485)

[Hình 3‑47 Sequence diagram UC010 107](#_Toc5920486)

[Hình 3‑48 Activity diagram UC010a 109](#_Toc5920487)

[Hình 3‑49 Sequence diagram UC010a 110](#_Toc5920488)

[Hình 3‑50 Activity diagram UC010b 112](#_Toc5920489)

[Hình 3‑51 Sequence diagram UC010b 113](#_Toc5920490)

[Hình 3‑52 Activity diagram UC011 115](#_Toc5920491)

[Hình 3‑53 Sequence diagram UC011 116](#_Toc5920492)

[Hình 3‑54 Activity diagram UC012 117](#_Toc5920493)

[Hình 3‑55 Sequence diagram UC012 118](#_Toc5920494)

[Hình 3‑56 Activity diagram UC012a 120](#_Toc5920495)

[Hình 3‑57 Sequence diagram UC012a 121](#_Toc5920496)

[Hình 3‑58 Activity diagram UC012b 123](#_Toc5920497)

[Hình 3‑59 Sequence diagram UC012b 124](#_Toc5920498)

[Hình 3‑60 ERD của hệ thống skill inventory 126](#_Toc5920499)

[Hình 3‑61 Class diagram hệ thống skill incentory 127](#_Toc5920500)

# DANH MỤC CÁC BẢNG BIỂU

[Bảng 3‑1 Danh sách tác nhân và mô tả 33](#_Toc5920501)

[Bảng 3‑2 Danh sách Usecase và mô tả 37](#_Toc5920502)

[Bảng 3‑3 Đặc tả UC001 38](#_Toc5920503)

[Bảng 3‑4 Đặc tả UC001a 41](#_Toc5920504)

[Bảng 3‑5 Đặc tả UC001b 43](#_Toc5920505)

[Bảng 3‑6 Đặc tả UC001c 46](#_Toc5920506)

[Bảng 3‑7 Đặc tả UC002 47](#_Toc5920507)

[Bảng 3‑8 Đặc tả UC002a 49](#_Toc5920508)

[Bảng 3‑9 Đặc tả UC002b 52](#_Toc5920509)

[Bảng 3‑10 Đặc tả UC002c 55](#_Toc5920510)

[Bảng 3‑11 Đặc tả UC003 58](#_Toc5920511)

[Bảng 3‑12 Đặc tả UC004 61](#_Toc5920512)

[Bảng 3‑13 Đặc tả UC004a 63](#_Toc5920513)

[Bảng 3‑14 Đặc tả UC004b 66](#_Toc5920514)

[Bảng 3‑15 Đặc tả UC005, UC005a 71](#_Toc5920515)

[Bảng 3‑16 Đặc tả UC006, UC006a 75](#_Toc5920516)

[Bảng 3‑17 Đặc tả UC006a1 78](#_Toc5920517)

[Bảng 3‑18 Đặc tả UC006a2 81](#_Toc5920518)

[Bảng 3‑19 Đặc tả UC006b 86](#_Toc5920519)

[Bảng 3‑20 Đặc tả UC006c 89](#_Toc5920520)

[Bảng 3‑21 Đặc tả UC007 91](#_Toc5920521)

[Bảng 3‑22 Đặc tả UC008 94](#_Toc5920522)

[Bảng 3‑23 Đặc tả UC008a 96](#_Toc5920523)

[Bảng 3‑24 Đặc tả UC008b 99](#_Toc5920524)

[Bảng 3‑25 Đặc tả UC009 103](#_Toc5920525)

[Bảng 3‑26 Đặc tả UC010 106](#_Toc5920526)

[Bảng 3‑27 Đặc tả UC010a 108](#_Toc5920527)

[Bảng 3‑28 Đặc tả UC010b 111](#_Toc5920528)

[Bảng 3‑29 Đặc tả UC011 114](#_Toc5920529)

[Bảng 3‑30 Đặc tả UC012 117](#_Toc5920530)

[Bảng 3‑31 Đặc tả UC012a 119](#_Toc5920531)

[Bảng 3‑32 Đặc tả UC012b 122](#_Toc5920532)

# LỜI MỞ ĐẦU

**1. Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài**

Hiện nay việc số hoá, áp dụng công nghệ thông tin thay thế cho các hình thức truyền thống được chú trọng và áp dụng trong nhiều lĩnh vực của đời sống từ thương mại, giáo dục cho đến hành chính, mang lại nhiều lợi ích, tăng năng suất làm việc, tự động hoá các tác vụ và đơn giản hoá quy trình làm việc. Đặc biệt là trong lĩnh vực quản lý việc áp dụng công nghệ thông tin giúp lĩnh vực quản lý đạt hiệu quả cao hơn, quản lý thông tin trên quy mô lớn hơn, thống kê được dữ liệu và giúp công tác quản lý đơn giản hơn, từ quản lý hàng hoá, dữ liệu hành chính, sinh viên,.. đều mang lại những lợi ích rất lớn. Bên cạnh các lĩnh vực quản lý đó còn có một mục tiêu tiềm năng về quản lý vẫn đang trong quá trình phát triển và chưa được chú trọng nhiều đó là việc quản lý nhân sự, cụ thể đó là quản lý kỹ năng của nhân viên trong một công ty.

Đa số các doanh nghiệp hiện nay chưa chú trọng đến việc quản lý kỹ năng của nhân viên trong công ty, và các công tác quản lý bằng giấy tờ hành chính, quy trình còn khá phức tạp, khó quản lý hiệu quả với quy mô lớn. Điều này còn chính xác hơn nữa với các doanh nghiệp công nghệ thông tin, khi các kỹ năng về công nghệ thông tin rất đa dạng và ở nhiều cấp độ khác nhau thì việc tìm hiểu, điều tra năng lực của nhân viên, sắp xếp vị trí phù hợp cho nhân viên, tiến hành đào tạo thêm tăng chất lượng cho nhóm kỹ năng phù hợp với nhân viên hoặc sắp xếp nhân viên vào dự án phù hợp để đạt được năng suất cao còn rất khó khăn. Việc tìm kiếm dữ liệu kỹ năng nhân viên và tuyển dụng vào các dự án công nghệ thông tin mất rất nhiều thời gian và chi phí.

Khi xây dựng được hệ thống quản lý kỹ năng của nhân viên, về ngắn hạn việc quản lý, thu thập thông tin của nhân viên trong công ty sẽ dễ dàng hơn, cho phép thực hiện trên quy mô lớn hơn và có thể sử dụng thông tin giúp việc tuyển dụng, sắp xếp vào các dự án hiệu quả, tiêu tốn ít thời gian hơn, mang lại lợi ích về chi phí, tăng lợi nhuận cho công ty và cũng giúp cho nhân viên có thể phát triển hết tiềm năng khi được sắp xếp vào đúng dự án phù hợp với kỹ năng. Về dài hạn, các thông tin về kỹ năng của nhân viên có thể phục vụ cho việc phân tích, thống kê nhằm lên kế hoạch bổ sung các nguồn lực, bồi dưỡng các kỹ năng mà công ty cần, định hướng phát triển cho nguồn nhân lực của công ty.

**2. Ý nghĩa khoa học và thực tiễn của đề tài**

Theo nghiên cứu bởi viện nghiên cứu khoa học James Martin tại trường University of Oxford’s Saïd Business School [1], khi khảo sát các CEO của các công ty, các CEO đều có nhận định chung về việc tìm kiếm các kỹ năng phù hợp mà họ cần đang là một thách thức lớn đối với doanh nghiệp của họ, trong một thị trường doanh nghiệp cạnh tranh từng giờ, cần nhân viên tài năng, phù hợp.

Hiểu được những bất cập, phức tạp trong quy trình quản lý kỹ năng nội bộ, để đáp ứng nhu cầu thu thập, quản lý các kỹ năng nội bộ của công ty FPT Software, nhóm quyết định nghiên cứu và phát triển hệ thống quản lý kỹ năng của nhân viên. Hệ thống nhằm giải quyết vấn đề cấp thiết trong khâu quản lý kỹ năng mà nhân viên có, từ đó góp phần giúp công ty đưa ra phương án đào tạo nhân lực hợp lý cũng như tuyển dụng nhân sự vào dự án phù hợp với kỹ năng.

# : GIỚI THIỆU

## Tổng quan

Đề tài “NGHIÊN CỨU VÀ PHÁT TRIỂN HỆ THỐNG QUẢN LÝ KỸ NĂNG CỦA NHÂN VIÊN” giúp các công ty, doanh nghiệp về lĩnh vực công nghệ thông tin nói chung và các doanh nghiệp về phần mềm nói riêng có thể quản lý nội bộ thông tin cá nhân cũng như các kỹ năng mà nhân viên có một cách dễ dàng, trực quan, đáp ứng nhu cầu thu thập, quản lý, phân tích các kỹ năng của nhân viên trong công ty. Đối với doanh nghiệp không có hệ thống quản lý nội bộ riêng, nhân viên quản lý dự án hay nhân viên nhân sự có thể phải trực tiếp đi đến từng bộ phận để xem xét, lựa chọn nhân viên có kỹ năng phù hợp hoặc quản lý thông tin trên giấy tờ, gây ra sự phức tạp và mất thời gian trong việc quản lý cũng như tìm nhân sự cho dự án. Khi áp dụng hệ thống vào doanh nghiệp sẽ hỗ trợ quá trình ứng tuyển nhân viên có kỹ năng phù hợp vào dự án của công ty nhanh chóng hơn vì dữ liệu các kỹ năng của nhân viên đã được lưu trữ trong hệ thống và khả năng tìm kiếm và gợi ý của hệ thống giúp lọc ứng viên phù hợp với yêu cầu của dự án.

Hệ thống hỗ trợ nhân viên của công ty có thể tùy chỉnh hồ sơ của bản thân với các thông tin về các kỹ năng kỹ thuật, chứng chỉ, học vấn, trình độ ngoại ngữ, kinh nghiệm làm việc, effort bản thân, một số thông tin cá nhân một cách đơn giản, nhanh chóng. Các thông tin này có thể chưa chính xác nên sẽ được bộ phận nhân viên nhân sự xác thực lại. Điều này giúp nhân viên có cái nhìn tổng quan về trình độ của bản thân, có thể đưa ra định hướng trau dồi kinh nghiệm, kiến thức trong tương lai tốt hơn.

Nhân viên nhân sự có thể quản lý toàn bộ hồ sơ của nhân viên, chỉnh sửa thông tin trong hồ sơ của toàn bộ nhân viên, đặc biệt là chức năng xác thực hồ sơ xem thông tin có chính xác và phù hợp không. Thống kê hồ sơ để nắm bắt được tình hình các kỹ năng của nhân viên trên nhiều tiêu chí kỹ năng. Quản lý thông tin dữ liệu hiển thị trong hệ thống, quản lý lịch sử cập nhật hồ sơ của nhân viên Nhân viên nhân sự có thể hỗ trợ nhân viên quản lý dự án tìm kiếm các CV có kỹ năng phù hợp với dự án cần tuyển.

Nhân viên quản lý dự án có thể quản lý thông tin các dự án mà mình đang phụ trách, hệ thống hỗ trợ tìm kiếm nhân viên phù hợp với dự án thông qua chức năng tìm kiếm, hỗ trợ đề xuất để tìm ra nhân viên có nhiều tiêu chí đáp ứng nhu cầu tìm kiếm nhất.

## Mục tiêu đề tài

Mục tiêu xây dựng được hệ thống quản lý thông tin kỹ năng của nhân viên trong nội bộ công ty với các chức năng:

* Chức năng dành cho nhân viên công ty: có các thao tác cập nhật một số thông tin cá nhân; các thao tác(thêm, chỉnh sửa, xóa) các thông tin về kỹ năng kỹ thuật, bằng cấp, chứng chỉ, trình độ ngoại ngữ, kinh nghiệm làm việc, effort của bản thân.
* Chức năng dành cho nhân viên nhân sự của công ty: có thể chỉnh sửa thông tin hồ sơ của toàn bộ của nhân viên. Xác thực thông tin trong hồ sơ bằng cách cập nhật trạng thái trong hồ sơ của nhân viên thành “Approved” hoặc “Reject”, mặc định hồ sơ sẽ có trạng thái “Pending”. Nhân viên nhân sự có thể thống kê dữ liệu về kỹ năng của toàn bộ nhân viên theo nhiều tiêu chí, để phục vụ kế hoạch định hướng và đào tạo nhân viên phù hợp theo xu hướng thị trường của công ty. Có thể thực hiện các chức năng quản lý thông tin dữ liệu hiển thị của hệ thống(thêm, chỉnh sửa, xóa). Có thể truy xuất nhật ký thay đổi hồ sơ của các nhân phục vụ vấn đề truy cứu trách nhiệm nếu thông tin trong hồ sơ có vấn đề bất thường. Nhân viên nhân sự có thể hỗ trợ tìm kiếm ứng viên phù hợp với dự án đang cần tuyển thông qua chức năng tìm kiếm hỗ trợ đề xuất của hệ thống phục vụ quá trình quản lý chuyên nghiệp, nhanh gọn và tiện lợi hơn so với các làm thủ công.
* Chức năng dành cho nhân viên quản lý dự án của công ty: có thể thao tác dữ liệu (thêm, chỉnh sửa, xóa) thông tin các dự án của công ty mà mình đang phụ trách; có thể tìm kiếm nhân viên trong hệ thống phù hợp với dự án đang cần tuyển thêm nhân lực phục vụ quá trình phỏng vấn ứng viên nhanh chóng.

## Phạm vi đề tài

* Ứng dụng cung cấp RESTful API viết trên phần mềm Eclipse, sử dụng code Java, framework Spring boot 2.1.3, thiết kế giao diện trên phần mềm Webstorm, sử dụng framework Angular 7, Maven 3.6.0, chạy trên server công ty FPT Software, sử dụng hệ cơ sở dữ liệu MySQL 5 để lưu trữ dữ liệu.
* Thiết kế bao gồm các chức năng đã được nêu ở phần mục tiêu đề tài, sử dụng trên phạm vi có nhiều truy cập đồng thời. Tốc độ phụ thuộc đường truyền internet sử dụng và cấu hình của server.
* Hệ thống không quản lý hồ sơ của cá nhân không phải là nhân viên trong công ty. Hệ thống không hỗ trợ quy trình, nghiệp vụ kiểm tra, xét duyệt hồ sơ của nhân viên. Hệ thống không quản lý thông tin tài khoản đăng nhập của nhân viên.

## Mô tả yêu cầu chức năng

Nhân viên sử dụng tài khoản đã được đăng ký với công ty đăng nhập vào hệ thống để tạo hồ sơ phục vụ cho việc quản lý thông tin và tuyển dụng vào dự án nội bộ của các bộ phận trong công ty.

Hồ sơ này quản lý các thông tin về: thông tin cá nhân; giới thiệu chung; các kỹ năng về kỹ thuật; trình độ ngoại ngữ; bằng cấp; chứng chỉ; kinh nghiệm làm việc; effort (nỗ lực hiện tại - từ 0% đến 100% thể hiện khối lượng công việc và thời gian dành cho các dự án hiện tại mà nhân viên đang tham gia - effort này do nhân viên tự ước tính) dựa trên tiêu chí này nhân viên nhân sự/nhân viên quản lý dự án sẽ xem xét nhân viên này có thể tham gia vào các dự án khác nữa được không - đáp ứng được khối lượng công việc, thời gian của dự án mới đó không. Mặc định hồ sơ effort sẽ là 0%. Ví dụ: Một nhân viên đang tham gia vào dự án A với vai trò support và tự ước tính khối lượng công việc hiện tại chỉ tiêu tốn khoảng 20% effort và có thể tham gia thêm vào dự án khác, từ đó nhân viên nhân sự/quản lý khi tìm kiếm xem xét nhân viên có thể tham gia vào dự án hay không.

Nhân viên có thể thực hiện các thao tác quản lý (thêm, xóa, sửa) các thông tin về kỹ năng kỹ thuật, ngoại ngữ, chứng chỉ, học vấn, kinh nghiệm làm việc của hồ sơ cá nhân và thông tin cá nhân. Có thể thay đổi avatar cá nhân. Tuy nhiên, đối với mục kinh nghiệm làm việc nhân viên chỉ được phép thao tác với các dữ liệu mà nhân viên tự thêm vào, các dữ liệu được hệ thống thêm vào nhân viên không được phép chỉnh sửa.

Mỗi hồ sơ của nhân viên sẽ có thuộc tính là trạng thái thuộc một trong các giá trị: Pending(chờ duyệt), Rejected(Từ chối), Approved(Thông qua) Mặc định trạng thái của hồ sơ sẽ là Pending. Sau khi nhân viên nhân sự xem xét, nếu toàn bộ thông tin trong hồ sơ của nhân viên là chính xác, nhân viên nhân sự có thể cập nhật trạng thái hồ sơ thành Approved. Nếu thông tin trong hồ sơ có vấn đề như sai lệch so với khảo sát thực tế hoặc thiếu thông tin, nhân viên nhân sự có thể cập nhật trạng thái thành Rejected. Trường hợp sau khi hồ sơ nhân viên đã được nhân viên nhân sự xét duyệt (Approved hoặc Reject) nếu nhân viên tiến hành chỉnh sửa, trạng thái hồ sơ sẽ tự động chuyển trạng thái thành Pending, và hệ thống tự động gửi mail thông báo cho nhân viên nhân sự.

Nhân viên có thể tiến hành nhập nhanh các thông tin kỹ năng qua giao diện mô phỏng spreadsheet; có thể xuất thông tin hồ sơ dưới dạng file word .doc hoặc .pdf theo template mặc định của hệ thống; có thể tải tài liệu hướng dẫn sử dụng hệ thống và có thể gửi mail cho nhân viên admin yêu cầu thêm dữ liệu về một trong các mục dữ liệu hiển thị của hồ sơ(kỹ năng, ngoại ngữ, ..) mà hệ thống không có.

Nhân viên nhân sự có tất cả các tính năng của nhân viên bình thường về quản lý hồ sơ cá nhân và có thể xem và chỉnh sửa hồ sơ của nhân viên khác. Nhân viên nhân sự có thể sử dụng tính năng tìm kiếm theo username hoặc tìm kiếm nâng cao. Chức năng tìm kiếm nâng cao cho phép nhân viên nhân sự có thể tìm theo nhiều tiêu chí khác khác nhau (kĩ năng kĩ thuật, ngôn ngữ, chứng chỉ, phòng ban, trạng thái hồ sơ) qua hệ thống checkbox, filter, kết hợp với thuật toán để đưa ra đề xuất thông minh phù hợp nhất với các tiêu chí tìm kiếm. Sau khi tìm kiếm ra danh sách nhân viên, nhân viên nhân sự có thể chọn các nhân viên bằng checkbox và xuất danh sách ra file spreadsheet, hỗ trợ việc làm hồ sơ, sắp xếp, lựa chọn tuyển dụng nhân sự cho dự án.

Nhân viên nhân sự sau khi xem xét thông tin hồ sơ của một nhân viên, tiến hành xác thực (nghiệp vụ bên ngoài hệ thống), có thể thay đổi trạng thái hồ sơ: hồ sơ đúng với kết quả xác thực, nhân viên nhân sự sẽ cập nhật trạng thái thành Approved, và hệ thống sẽ gửi mail thông báo tới nhân viên được duyệt. Nếu thông tin hồ sơ không chính xác, nhân viên nhân sự sẽ cập nhật trạng thái thành Rejected, kèm lý do Reject, hệ thống sẽ gửi mail thông báo tới nhân viên kèm lý do. Nhân viên nhân sự không được phép tự duyệt cho bản thân. Có thể thống kê được số hồ sơ mà nhân viên nhân sự đã duyệt. Thống kê tổng số lượng hồ sơ, số lượng hồ sơ Pending, Approved, Rejected, thống kê về số người có một ngoại ngữ, một kỹ năng hay một chứng chỉ cụ thể.  Có thể thực hiện các chức năng quản lý dữ liệu hiển thị của hệ thống (data dùng để load lên form) về danh sách chứng chỉ, ngoại ngữ, kỹ năng kỹ thuật, trường học, quốc gia, bằng cấp, lĩnh vực của bằng cấp, vị trí công việc. Có thể thêm mới một dữ liệu hệ thống; xóa hoặc sửa các dữ liệu trên, với điều kiện chưa có nhân viên nào sử dụng dữ liệu đó vào hồ sơ. Có thể kiểm tra nhật ký thay đổi của các hồ sơ hệ thống nhật ký.

Nhân viên quản lý dự án có tất cả các tính năng của nhân viên bình thường về quản lý hồ sơ cá nhân và có thể xem thông tin hồ sơ của nhân viên khác.

Nhân viên quản lý dự án có thể thực hiện các chức năng: thêm, thay đổi trạng thái, xóa, sửa thông tin các dự án của công ty mà mình đang phụ trách. Trạng thái của một dự án có thể là Waiting(đang đợi tuyển thành viên), On-going(đang tiến hành), Finished(đã hoàn thành), Cancelled(bị huỷ). Khi một dự án thay đổi trạng thái sang Finished hồ sơ của tất cả các ứng cử viên với trạng thái “Joined” sẽ được thêm thông tin vào project và role trong mục Project Experience (kinh nghiệm làm việc).

Nhân viên quản lý dự án chỉ có thể xem danh sách dự án mà mình tạo ra và có thể tìm kiếm nhân sự cho dự án qua chức năng gợi ý tự động của hệ thống. Hệ thống sẽ dựa trên các thông tin yêu cầu về kĩ năng kĩ thuật, ngoại ngữ, chứng chỉ, phòng ban của từng vị trí, tính toán đưa ra đề xuất và hiển thị danh sách được sắp xếp theo mức độ phù hợp từ cao đến thấp của vị trí đó.

Nhân viên quản lý dự án có thể sử dụng tính năng tìm kiếm theo user name và tìm kiếm nâng cao như của nhân viên nhân sự. Sau khi tìm kiếm được hồ sơ phù hợp, nhân viên quản lý dự án có thể thêm nhân viên đó vào danh sách ứng cử viên của dự án. và có thể thay đổi trạng thái của ứng cử viên trong danh sách ứng cử viên trong dự án. Mặc định khi được thêm vào danh sách, ứng cử viên sẽ có trạng thái Pending(Chờ), nhân viên quản lý dự án có thể thay trạng thái ứng cử viên thành Interview (Phỏng vấn) khi được thay đổi sang trạng thái này hệ thống sẽ gửi mail tới nhân viên, Joined(Tham gia) khi được thay đổi sang trạng thái này hệ thống sẽ gửi mail tới nhân viên.

Nhân viên quản lý dự án có thể thực hiện các chức năng quản lý vị trí tuyển dụng trong dự án như: thêm một vị trí tuyển dụng, cập nhật thông tin vị trí tuyển dụng, thay đổi trạng thái vị trí tuyển dụng Có các trạng thái(Open - Đang tuyển, Close - đủ người hoặc kết thúc tuyển, Cancelled - Huỷ vị trí). Mặc định sẽ là Open. Khi trạng thái đổi sang “Close” sẽ không thể thêm nhân viên vào danh sách ứng cử viên của vị trí này. Khi trạng thái đổi sang Cancelled tất cả nhân viên trong danh sách ứng cử viên của vị trí này sẽ đổi trạng thái thành Cancelled.

# : CƠ SỞ LÝ THUYẾT

## Spring Boot

### 2.1.1 Kiến trúc và các thành phần của Spring Boot:

Nền tảng của Spring Boot là Spring Framework. Spring Framework [3] được xây dựng dựa trên 2 nguyên tắc design chính là: Dependency Injection (cho phép xóa bỏ sự phụ thuộc hard-code và làm cho ứng dụng dễ mở rộng và bảo trì hơn) và Aspect Oriented Programming (để tách biệt các thành phần business).

Ưu và nhược điểm của Dependency Injection:

Ưu điểm:

* Giảm sự kết dính giữa các module.
* Code dễ bảo trì, dễ thay thế module.
* Rất dễ test và viết Unit Test.
* Dễ dàng thấy quan hệ giữa các module (Vì các dependency đều được inject vào constructor).

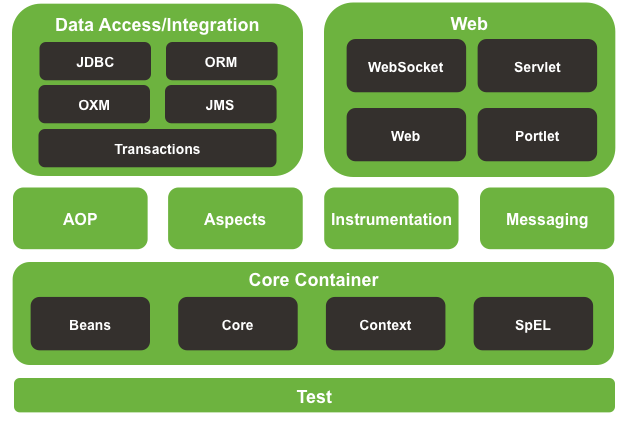
Nhược điểm:

* Khái niệm DI hơi khó hiểu với người mới.
* Khó debug vì không biết implements nào của interface được gọi đến.
* Các object được khởi tạo từ đầu làm giảm performance.
* Làm tăng độ phức tạp của code.

Những tính năng core (cốt lõi) của Spring có thể được sử dụng để phát triển Java Desktop, ứng dụng mobile, Java Web. Mục tiêu chính của Spring là giúp phát triển các ứng dụng J2EE một cách dễ dàng hơn dựa trên mô hình sử dụng POJO (Plain Old Java Object) Spring được chia làm nhiều module khác nhau, tùy theo mục đích phát triển ứng dụng mà ta dùng 1 trong các module đó. Một số lợi ích của Spring Framework:

* Spring cho phép lập trình viên sử dụng POJOs. Việc sử dụng POJOs giúp bạn không phải làm việc với EJB, ứng dụng, các luồng chạy, cấu hình… đơn giản hơn rất nhiều.
* Spring được tổ chức theo kiểu module. Số lượng các gói và các lớp khá nhiều, nhưng chỉ cần quan tâm đến những gì cần và không cần quan tâm đến phần còn lại.
* Spring hỗ trợ sử dụng khá nhiều công nghệ như ORM Framework, các logging framework, JEE, các thư viện tạo lịch trình (Quartz và JDK timer)…
* Module Web của Spring được thiết kế theo mô hình MVC nên nó cung cấp đầy đủ các tính năng giúp thay thế các web framework khác như Struts.

Dưới đây là kiến trúc tổng thể của Spring Framework:



Hình ‑ Kiến trúc tổng thể Spring FrameWork

* Test: Tầng này cung cấp khả năng hỗ trợ kiểm thử với JUnit và TestNG.
* Spring Core Container: Bao gồm các module spring core, beans, context và expression language (EL)
  + Spring core, bean cung cấp tính năng IOC và Dependency Injection.
  + Spring Context hỗ trợ đa ngôn ngữ (internationalization), các tính năng Java EE như EJB, JMX.
  + Expression Language được mở rộng từ Expression Language trong JSP. Nó cung cấp hỗ trợ việc setting/getting giá trị, các method cải tiến cho phép truy cập collections, index, các toán tử logic…
  + AOP, Aspects and Instrumentation: Những module này hỗ trợ cài đặt lập trình hướng khía cạnh (Aspect Oriented Programming), hỗ trợ tích hợp với AspectJ.
  + Data Access / Integration: Nhóm này bao gồm JDBC, ORM, OXM, JMS và module Transaction. Những module này cung cấp khả năng giao tiếp với database
* Web: Hay còn gọi là Spring MVC Nhóm này gồm Web, Web-Servlet… hỗ trợ việc tạo ứng dụng web.

Tính năng quan trọng nhất của Spring Framework là Dependency Injection. Cốt lõi của tất cả các module Spring là Dependency Injection hoặc IOC Inversion of Control. Tại sao nó lại quan trọng? Bởi vì, khi DI hoặc IOC được sử dụng đúng cách, chúng ta có thể phát triển các ứng dụng ghép lỏng lẻo. Và các ứng dụng ghép lỏng lẻo có thể dễ dàng được bảo trì cũng như dễ kiểm thử đơn vị.

### 2.1.2 Lợi ích sử dụng Spring Boot :

Spring Boot [4] ra đời để cải thiện một số vấn đề của Spring Framework, đặc biệt là bước cấu hình, vì Spring Framework đòi hỏi cấu hình khá phức tạp. Spring Boot là một project trong Spring io platform nằm ở layer IO Execution, cung cấp tính năng RAD (Rapid Application Development) - Phát triển ứng dụng nhanh. Nó giảm thiểu effort cần thiết để tạo production-ready, DevOps-friendly, XML-free Spring applications. Đơn giản hóa bootstrapping của các dự án Spring với source code tối thiểu, thực hiện khả năng mở rộng các tính năng hoạt động như tự động kiểm tra metrics, endpoints,... và hỗ trợ embedded containers cho phép tạo ra các tập lệnh có khả năng tự thực thi. Làm cho project dùng spring trở nên độc lập và gọn nhẹ.

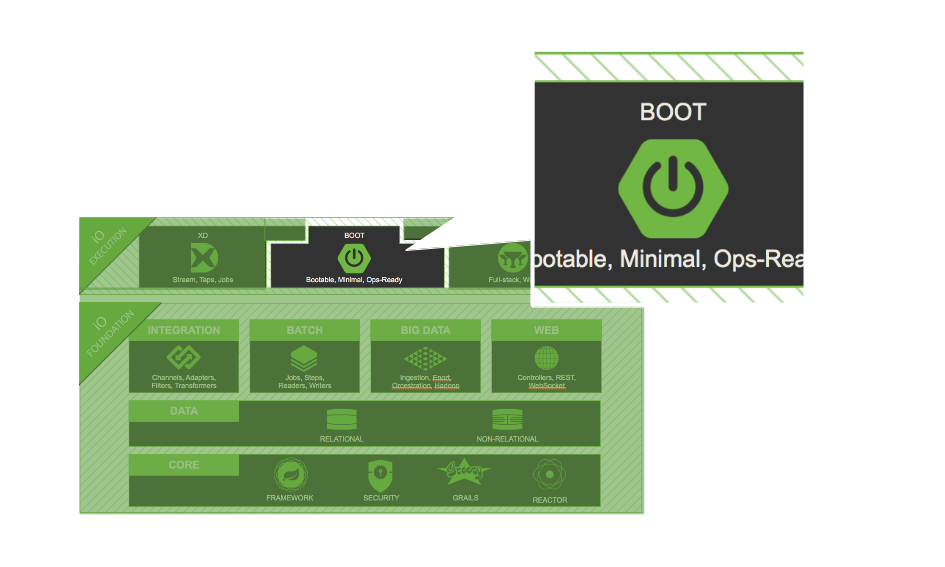
Ví dụ cần phát triển một ứng dụng web cơ bản Hello World sử dụng Spring framework. Các công đoạn lập trình ứng dụng này sẽ ít nhất bao gồm các công đoạn - ít nhất 5 bước sau:

* Tạo một project sử dụng Maven với các dependency cần thiết của Spring MVC và Servlet API.
* Một tập tin web.xml để khai báo DispatcherServlet của Spring MVC.
* Một tập tin cấu hình của Spring MVC.
* Một class Controller trả về một trang “Hello World” khi có request đến.
* Cuối cùng là phải có một web server dùng để triển khai ứng dụng lên chạy.

Để giảm bớt các công đoạn có thể giống nhau này, đối với ứng dụng Spring Boot chỉ cần triển khai qua các bước đơn giản sau:

* Tạo một project sử dụng Maven với các dependency cần thiết của Spring MVC và Servlet API.
* Một class Controller trả về một trang “Hello World” khi có request đến.

Mô hình Spring Boot trong spring io:



Hình ‑ Mô hình Spring Boot trong spring io

Yêu cầu hệ thống: Spring Boot 2.1.3.RELEASE yêu cầu JDK 8 - 11 và Spring Framework 5.1.5 hoặc cao hơn. Yêu cầu maven (3.3+) hoặc Gradle (4.4+).

Có thể dùng spring boot để tạo ứng dụng Java chạy bằng command line 'java -jar' hoặc export gói war để deploy lên server như thông thường.

Ưu điểm của Spring boot:

* Có các tính năng của Spring Framework.
* Tạo ứng dụng độc lập, có thể chạy bằng java -jar (cho cả java web).
* Nhúng trực tiếp các ứng dụng server (Tomcat, Jetty…) do đó không cần phải triển khai file WAR.
* Cấu hình ít, tự động cấu hình bất kỳ khi nào có thể (Giảm thời gian viết code, tăng năng suất).
* Không yêu cầu XML config.
* Cung cấp nhiều plugin.

### 2.1.3 Các dependency sử dụng trong project:

* spring-boot-starter-web: ứng dụng web project Spring boot sử dụng module này để khởi động và chạy một cách nhanh chóng. Bao gồm các thư viện để xây dựng một ứng dụng web sử dụng Spring MVC, và sử dụng tomcat (spring-boot-starter-tomcat) như là một Web Container mặc định được nhúng vào (embedded). Nó bao gồm cả các thư viện cho ứng dụng RESTful.
* spring-boot-starter-data-jpa: Spring Data JPA, giúp dễ dàng thực hiện lưu trữ dữ liệu dựa trên JPA. Dependency này liên quan đến hỗ trợ nâng cao cho các lớp truy cập dữ liệu dựa trên JPA. Nó giúp việc xây dựng các ứng dụng hỗ trợ Spring sử dụng các công nghệ truy cập dữ liệu dễ dàng hơn.

Việc triển khai một lớp riêng để hỗ trợ truy cập dữ liệu của một ứng dụng đã khá cồng kềnh. Quá nhiều code được viết để thực hiện các truy vấn đơn giản cũng như thực hiện phân trang và kiểm soát dữ liệu. Spring Data JPA nhằm mục đích cải thiện đáng kể việc thực hiện các lớp truy cập dữ liệu bằng cách giảm công việc phải làm để thực hiện truy vấn cần thiết. Có thể viết các xử lý cho việc lưu trữ dữ liệu của project, bao gồm các phương thức tìm kiếm tùy chỉnh và Spring sẽ tự động cung cấp việc triển khai. Ngoài ra dependency này cho phép thao tác với database một cách hoàn toàn tự nhiên thông qua các đối tượng. Hoàn toàn không cần quan tâm đến loại database sử dụng là gì.

* mysql-connector-java: thư viện jdbc hỗ trợ kết nối tới cơ sở dữ liệu MySQL. Sử dụng version 5. Khai báo các thông tin cấu hình cần thiết để kết nối với MySQL trong file application.properties của ứng dụng Spring Boot.
* spring-boot-devtools: DevTools là một bộ công cụ của Spring Boot giúp cho quá trình phát triển ứng dụng Spring Boot được dễ dàng và tiện lợi hơn. Khi sử dụng DevTools của Spring Boot chúng ta không cần phải restart lại ứng dụng mỗi khi có sự thay đổi về code, chỉ cần save lại code vừa viết sẽ tự động restart lại ứng dụng. Hay nếu đang làm việc với một ứng dụng web có UI, mỗi lần cập nhật code để thay đổi giao diện, bình thường phải refresh lại browser để thấy sự thay đổi này, sử dụng DevTools có thể cài đặt để browser có thể tự động reload để thấy sự thay đổi.
* spring-security-cas và spring-boot-starter-security: Dependency hỗ trợ việc chứng thực thông tin khi truy cập vào ứng dụng.

## Angular

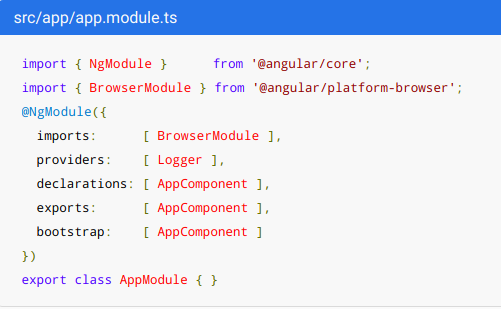
### 2.2.1. Giới thiệu về Angular

Angular [5] là một framework front-end Javascript hoàn chỉnh được xây dựng và bảo trì bởi Google và sử dụng TypeScript một ngôn ngữ mở rộng của JavaScript được xây dựng bởi Microsoft, hỗ trợ xây dựng ứng dụng web, mobile, hoặc desktop phía client, và thường được dùng để xây dựng Single-page Application (web application tương tác với người dùng và load động các các phần trong web mà không cần tải lại toàn bộ trang) .

Khác với AngularJS (hay còn gọi là Angular1) là một thư viện viết dựa trên JavaScript thuần theo mô hình MVC tập trung chủ yếu vào xử lý controller, và chưa được cấu trúc theo các best-practice (các kỹ thuật tốt nhất được sử dụng rộng rãi) gây khó khăn trong quá trình bảo trì, mở rộng. Angular ( hay còn còn gọi là Angular2 cho các version từ 2 trở lên ) gần như được viết lại hoàn toàn theo các best-practice tập trung vào việc xử lý logic của view và việc tương tác, quản lý các component(thành phần) và directive(chỉ thị).

### 2.2.2. Tổng quan về kiến trúc

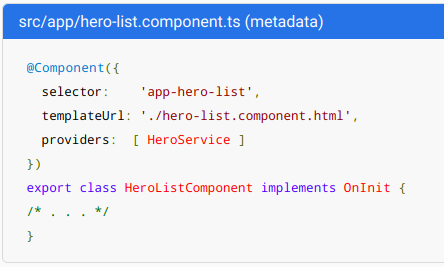
Angular được xây dựng dựa khối cơ bản được gọi là NgModule chứa một tập các Component, Service và luồng xử lý liên quan, một Angular Application cần có ít nhất một NgModule được gọi là root module cho phép việc tự động khởi chạy, biên soạn ngữ cảnh cho các Component và có thể có thêm nhiều module tính năng khác.



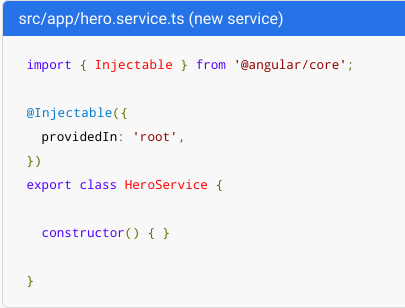
Hình ‑ Module mẫu của một Angular App

Trong Angular một Component đại diện cho view(giao diện) là một tập hợp các thành phần của màn hình mà Angular có thể chọn và thay đổi logic, dữ liệu theo chương trình nhà phát triển viết ra. Component sẽ sử dụng các Service - một class cung cấp các chức năng với mục đích cụ thể, không liên quan trực tiếp tới view  - mà các Service provider sẽ inject vào Component trở thành một Component Dependency làm cho code trở nên module hoá, có thể tái sử dụng nhiều lần, hiệu quả hơn trong việc mở rộng, bảo trì.

Mỗi Component và Service đều chỉ là một class với decorators(chú thích - tương tự annotation) để đánh dấu kiểu của chúng và cung cấp một số metadata để Angular biết được các sử dụng của chúng. Các metadata của một class Component giúp liên kết với template và định nghĩa nên view. Các metadata của class Service cung cấp cho Angular biết các thông tin để cung cấp sẵn class Service đó cho các Component thông qua dependency injection.



Hình ‑ Một Angular Component mẫu với metadata khai báo template và service provider



Hình ‑ Một Angular Service mẫu với metadata với khai báo cho phép cung cấp service tại root application

Một Component định nghĩa nhiều view và xác định thứ bậc giữa các view có thể sử dụng Router được Angular cung cấp để định nghĩa các đường dẫn giữa các view.



Hình ‑ Kiến trúc cơ bản của một Angular App

Sự kết hợp giữa Template và Component định nghĩa nên view, trong đó:

* Decorator trên một class Component thêm metadata vào class, bao gồm thông tin trỏ tới template tương ứng.
* Derectives và binding markup trong template của component chỉnh sửa view dựa trên dữ liệu và logic lập trình

Dependency Injector cung cấp Service cần thiết đến Component, ví dụ như Router Service cho phép định nghĩa chuyển hướng giữa các view.

### 2.2.3. Angular CLI(Command-line Interface)

Angular cung cấp một ứng dụng command-line tiện dụng tên là Angular Cli với tên đại diện là "ng" với nhiều tính năng:

* Hỗ trợ khởi tạo khung project nhanh chóng với lệnh "ng new TênApp", điều này còn giúp cho cấu trúc của project Angular được tiêu chuẩn hoá giúp các nhà phát triển làm việc với nhau dễ dàng hơn
* Hỗ trợ khởi tạo nhanh các component, class, service,... với "ng generate" tăng năng suất lập trình
* Cung cấp server tự khởi chạy hỗ trợ môi trường development với "ng serve"
* Hỗ trợ các thao tác với test, build, update,...

Đây là một công cụ mạnh mẽ tăng tính tiêu chuẩn, giảm thời gian phát triển phần mềm và hỗ trợ tăng năng suất lập trình.

### 2.2.4. Các chức năng chính của Angular

Templates, directives :

* Một template là sự kết hợp của HTML thông thường và các markup binding(đánh dấu ràng buộc dữ liệu) và directive(chỉ thị) của Angualar, cho phép Angular sửa đổi HTML theo logic lập trình trước khi hiển thị chúng. Angular sử dụng dấu "{{ }}" để đánh dấu nơi cần hiển thị data tương ứng vào HTML.
* Derective trong Angular như \*ngIf cung cấp logic quyết định, \*ngFor cho cấu trúc lặp,... cung cấp logic cho việc render dữ liệu.
* Một điểm đặc biệt mạnh mẽ trong Angular đó là data binding:
* Có 2 cách binding dữ liệu trong Angular đó là Event binding và Property binding.
* Event binding cho phép ứng dụng phản hồi lại với input từ người dùng bằng cách cập nhật lại dữ liệu.
* Property binding cho phép nội suy(nhúng) dữ liệu đã được tính toán từ ứng dụng vào HTML.
* Khi kết hợp cả 2 cách trên Angular tạo nên Two-way data binding kết nối 2 chiều giữa component và HTML template cho phép tạo nên luồng dữ liệu qua lại cập nhật liên tục để nhà phát triển có thể tạo nên ứng dụng tương tác mang trải nghiệm hoàn toàn khác biệt. Chỉ cần sử dụng cú pháp [(ng-model)]="property" dữ liệu giữa DOM và Component sẽ được kết nối.



Hình ‑ Lược đồ miêu tả luồng dữ liệu qua lại giữa DOM và Component với các cú pháp tương ứng

Template-driven form và dynamic form:

* Hỗ trợ việc khởi tạo form, quản lý trạng thái form, validation dữ liệu hiệu quả, nhanh chóng. Đối với template-driven form qua việc kết nối dữ liệu input với Variable template kết hợp với two-way binding hỗ trợ việc validation và tương tác form ngay khi người dùng bắt đầu nhập liệu . Đối với reactive form cho phép khởi tạo form động, quản lý vòng đời form khi người dùng tiến hành tương tác.

HttpClientModule:

* Là một HTTP API cung cấp thư viện lightweight giao tiếp với backend API qua XMLHttpRequest interface mà browser cho phép request, respone object, bắt chặn tin, và hỗ trợ bất đồng bộ.

Angular animations:

* Angular cho phép khởi tạo và định nghĩa các trạng thái, cũng như thuộc tính chuyển trạng thái các Component trong ứng dụng cho phép tạo nên các hiệu ứng đặc biệt để tăng trải nghiệm người dùng và đồng thời giản thiểu độ khó cho các nhà phát triển

### 2.2.5. So sánh Angular với các công nghệ front-end khác:

Hiện nay về phía front-end Javascript có 2 cái tên đứng đầu đó là Angular và React, khi so sánh với React (phát triển bởi Facebook) Angular vượt trội hơn [7] về các mặt như:

* Có cấu trúc project tiêu chuẩn hoá, sử dụng TypeScript cũng hỗ trợ về mặt tiêu chuẩn coding convention giảm thiểu khả năng tạo ra lỗi và tạo điều kiện cho các nhà phát triển làm việc với nhau dễ dàng hơn. Trong khi React được viết với JavaScript thuần và không có cấu trúc project tiêu chuẩn, sẽ dễ dàng gây nên lỗi không mong muốn nếu không định nghĩa các quy định cấu hình và convention ngay từ đầu.
* Angular cung cấp command-line interface Angular CLI giúp nhanh chóng khởi tạo project và các thành phần cần thiết trong quá trình phát triển, tăng năng suất lập trình trong khi React cần phải khời tạo thủ công.
* Angular là một framework hoàn chỉnh hỗ trợ thư viện đồ sộ và các built-in giúp cho các nhà phát triển có nhiều nguồn tài nguyên được xây dựng theo các tiêu chuẩn tốt nhất, React chỉ là một thư viện JavaScript, khi lập trình với React các nhà phát triển thường phải mất khá nhiều thời gian xác định các thành phần cần thiết trong dự án để tích hợp vào.
* Khi sử dụng Angular lập trình viên có thể sử dụng với HTML và CSS thuần trong khi React sử dụng JSX không sử dụng HTML và tương thích với các CSS pre-processor.
* Về giấy phép bản quyền Angular là một mã nguồn mở miễn phí, React thuộc quyền sáng chế và giấy phép của Facebook (gần đây Facebook mới đổi sang giấy phép MIT cho phép sử dụng miễn phí).
* Tuy nhiên Angular vẫn có một số hạn chế như đòi hỏi lập trình viên phải tìm hiểu nhiều kiến thức hơn về TypeScript và các pattern có trong Angular, cũng như các thư viện trong Angular trước khi bắt đầu. Về dung lượng do Angular framework lớn nên sẽ tốn nhiều dung lượng hơn React. Khi sử dụng các thư viện ngoài Angular do viết bởi TypeScript nên tương thích với ít thư viện bên ngoài hơn React.
* React với JSX có thể sử dụng server side rendering còn Angular thì chỉ có thể dùng client side rendering.

Bên cạnh jQuery cũng là một công nghệ được nhiều người sử dụng khi phát triển front-end, tuy nhiên khi so sánh Angular  [7]vẫn nổi trội hơn về nhiều mặt:

* Angular hỗ trợ two-way binding dữ liệu giữa các class model và UI, jQuery chỉ hỗ trợ one-way binding nhưng chỉ với các thành phần UI với nhau.
* Mã nguồn jQuery khó bảo trì và nâng cấp hơn Angular vì không hỗ trợ các module và component.
* Việc viết và sử dụng AJAX của jQuery dài dòng và tốn nhiều công sức, hơn việc sử dụng HttpClient của Angular, lập trình viên mới bắt đầu có thể mất nhiều thời gian hơn khi sử dụng AJAX.
* Tuy nhiên thì việc tìm hiểu và học jQuery sẽ dễ dàng và đòi hỏi ít thời gian hơn tìm hiểu Angular.

## MySQL

### Cơ sở dữ liệu là gì?

Một Database (Cơ sở dữ liệu) là một ứng dụng riêng rẽ mà lưu trữ một tập hợp dữ liệu. Mỗi cơ sở dữ liệu có một hoặc nhiều API riêng biệt để tạo, truy cập, quản lý, tìm kiếm và tái tạo dữ liệu đang giữ.

Một số loại kho lưu dữ liệu khác có thể được sử dụng, chẳng hạn như file trên hệ thống file hoặc các Hash Table lớn, nhưng việc lấy và ghi dữ liệu không thể nhanh và dễ dàng với các loại kho lưu dữ liệu này của các hệ thống.

Vì thế, ngày nay, chúng ta sử dụng các Hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) lưu giữ và quản lý khối lượng lớn dữ liệu. RDBMS được gọi là cơ sở dữ liệu quan hệ, bởi vì tất cả dữ liệu được lưu giữ trong các bảng dữ liệu khác nhau và các mối quan hệ được thành lập bởi sử dụng các Primary Key (khóa chính) và một số khóa khác được biết đến như là Foreign Key.

Một Hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) là một phần mềm:

* Cho phép triển khai một Database với các bảng dữ liệu, cột (column), và các chỉ mục (Index).
* Bảo đảm Referential Integrity (có thể dịch là toàn vẹn quan hệ) giữa các hàng và các bảng đa dạng.
* Cập nhật tự động các chỉ mục.
* Thông dịch một truy vấn SQL và tổ hợp thông tin từ các bảng khác nhau.
  + 1. ***Thuật ngữ RDBMS***

Trước khi đi vào khái niệm hệ thống cơ sở dữ liệu MySQL, chúng ta cùng xem lại một số định nghĩa liên quan tới cơ sở dữ liệu:

* Database: Một cơ sở dữ liệu là một tập hợp các bảng dữ liệu, với dữ liệu có liên quan.
* Bảng dữ liệu: Một bảng là một ma trận dữ liệu. Một bảng trong một cơ sở dữ liệu trông giống như một bảng tính đơn giản.
* Cột: Một cột chứa cùng một kiểu dữ liệu, ví dụ như tên khách hàng.
* Hàng: Một hàng (row, entry, record) là một nhóm dữ liệu có liên quan.
* Redundancy: (có thể hiểu là dữ liệu dự phòng) Dữ liệu được lưu giữ hai lần, để làm cho hệ thống nhanh hơn.
* Primary Key: Một Primary Key (Khóa chính) là duy nhất. Một giá trị key không thể xuất hiện hai lần trong một bảng. Với một key, bạn có thể tìm thấy phần lớn trên một hàng.
* Foreign Key: tạo liên kết giữa hai bảng.
* Compound Key: Một Compound Key (hay composite key) là một key mà gồm nhiều cột, bởi vì một cột là không duy nhất.
* Index: Một chỉ mục trong một cơ sở dữ liệu tương tự như chỉ mục trong một cuốn sách.
* Referential Integrity: Đảm bảo rằng một giá trị Foreign Key luôn luôn trỏ tới một hàng đang tồn tại.
  + 1. ***MySQL và lợi ích khi sử dụng MySQL***

MySQL là một RDBMS nhanh và dễ dàng để sử dụng. MySQL đang được sử dụng cho nhiều công việc kinh doanh từ lớn tới nhỏ. MySQL được phát triển, được công bố, được hỗ trợ bởi MySQL AB, là một công ty của Thụy Điển. MySQL trở thành khá phổ biến vì nhiều lý do:

* MySQL là mã nguồn mở. Vì thế, khi sử dụng MySQL sẽ không tốn chi phí cho license.
* MySQL sử dụng một Form chuẩn của ngôn ngữ dữ liệu nổi tiếng là SQL.
* MySQL làm việc trên nhiều Hệ điều hành và với nhiều ngôn ngữ như PHP, PERL, C, C++, Java, …
* MySQL làm việc nhanh và khỏe ngay cả với các tập dữ liệu lớn.
* MySQL hỗ trợ các cơ sở dữ liệu lớn, lên tới 50 triệu hàng hoặc nhiều hơn nữa trong một bảng. Kích cỡ file mặc định được giới hạn cho một bảng là 4 GB, nhưng MySQL cho phép tăng kích cỡ này (nếu hệ điều hành của có thể xử lý được) để đạt tới giới hạn lý thuyết là 8 TB.
* MySQL là cho phép điều chỉnh theo nhu cầu lập trình viên. Giấy phép GPL mã nguồn mở cho phép lập trình viên sửa đổi phần mềm MySQL để phù hợp với môi trường cụ thể của họ.

## Maven

* + 1. ***Tổng quan về Apache Maven***

Apache Maven [8] (gọi tắt là Maven) được xây dựng bởi công ty The Apache Software Foundation là một chương trình quản lý dự án mã nguồn Java và là một build tool hỗ trợ xây dựng quản lý vòng đời mã nguồn Java hoàn chỉnh. Được xây dựng nhằm mục đích thay thế công cụ Apache Ant do việc sử dụng Apache Ant không có một tiêu chuẩn chung trong việc cấu hình buid dự án, mỗi một nhà phát triển có thể cấu hình theo các lệnh khác nhau để buid dự án gây khó khăn trong quá trình làm việc nhóm (ví dụ đối với lệnh xoá mã nguồn build cũ thì một người có thể khai báo là "clear", người khác khai báo là "clean" và cấu trúc dự án, cũng như việc đặt tên, sắp xếp thư mục cũng không có một tiêu chuẩn chung).

Khác với Apache Ant, Apache Maven được xây dựng theo các tiêu chuẩn, định nghĩa có sẵn về cấu trúc dự án, các giai đoạn(phases) trong việc xây dựng dự án và các lệnh(goals) hỗ trợ việc build dự án, bên cạnh đó cho phép xây dựng và tích hợp thêm các plugin và câu lệnh plugin khác. Maven giúp cho việc xây dựng dự án dễ dàng hơn nhờ các lệnh được định nghĩa sẵn, tuy nhiên không che giấu chi tiết mà cho thể hiện một cách rõ ràng những cơ chế bên dưới các lệnh. Maven cung cấp một hệ thông xây dựng thống nhất dựa trên mô hình POM và một tập các plugin được sử dụng chung giữa các dự án, một khi đã quen với một dự án xây dựng với Maven thì khi chuyển sang các project khác cũng xây dựng bởi Maven sẽ không mất thêm thời gian để tìm hiểu. Maven cung cấp nhiều thông tin dự án hữu ích lấy từ file POM và mã nguồn dự án, Maven có thể lấy được thông tin phiên bản trực tiếp từ mã nguồn, danh sách các thư viện cần thiết, kết quả kiểm thử mã nguồn.

* + 1. ***Lợi ích khi sử dụng Apache Maven:***

Apache Maven dựa trên mô hình đối tượng dự án (project object model) hay thường được viết tắt là POM. Maven có thể quản lý quá trình xây dựng, báo cáo, tích hợp, đóng gói dựa trên một file thông tin khai báo dưới dạng xml là POM.xml. Trong file POM.xml này các nhà phát triển có thể tiến hành khai báo các thông tin của dự án bao gồm các thông tin cơ bản về tên dự án, phiên bản, mô tả, các thông tin về thư viện (dependency) cần thiết để xây dựng dự án, các module trong dự án, các plugin và gắn kết các giai đoạn (phases) với các công việc cụ thể nhằm tự động hóa việc build mã nguồn.

* + 1. ***Ứng dụng Apache Maven vào dự án:***

Trong dự án "Nghiên cứu và Phát triển Hệ thống quản lý Kỹ năng của Nhân viên" Maven được sử dụng cho việc:

Khởi tạo cấu trúc ban đầu của dự án, Maven sẽ tự động khởi tạo cấu trúc mặc định của project giúp thống nhất cấu trúc và giảm thời gian chuẩn bị cấu trúc nền tảng (code base).

Quản lý và ràng buộc tự động các các thư viện cần thiết để xây dựng dự án.

Phân tách và quản lý các module nhỏ trong dự án, bằng việc khởi tạo một file POM cha cho các module nhỏ Maven giúp chia nhỏ mã nguồn thành các module độc lập nhau nhằm dễ dàng quản lý và mở rộng dự án.

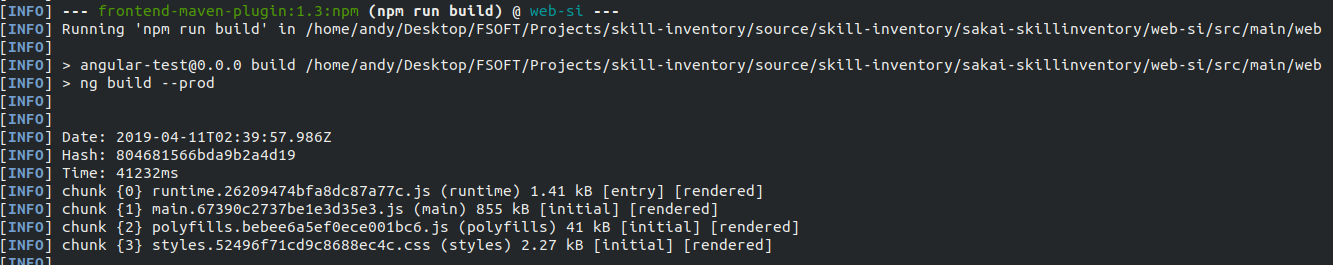


Hình ‑ File POM cha hỗ trợ quản lý thông tin dự án, thư viện và phân tách module

Hỗ trợ việc cài đặt tự động các thư viện và môi trường cần thiết, cũng như build tự động các mã nguồn ngôn ngữ khác(JavaScript) và đóng gói tự động mã nguồn, qua việc gắn kết các công việc cụ thể với các giai đoạn chạy của Maven, giúp dự án có thể cài đặt các công cụ cần thiết (như nodejs và npm) để hỗ trợ build front-end một các tự động để tiết kiệm thời gian và không cần qua các bước phức tạp, tự động gọi  build mã nguồn front-end để tích hợp vào mã nguồn Java cũng như đóng gói các module một các tự động thành một gói WAR để triển khai.



Hình ‑ Cấu hình file POM hỗ trợ tự động cài đặt môi trường và build tự động cho Front-end



Hình ‑ Maven tự động build mã nguồn front-end Angular tự động trong quá trình đóng gói dự án.

## Nghiệp vụ quản lý nhân sự

Khi một nhân viên được tuyển dụng và tiến hành đào tạo cơ bản xong, các đơn vị của công ty FPT Software sẽ tiến hành phỏng vấn vào đơn vị, hoặc phỏng vấn đưa vào các dự án đang thiếu nhân lực của đơn vị, hoặc nếu chưa có dự án phù hợp, dự án trong quá trình chuẩn bị, nhân viên sẽ được đưa về bộ phận nhân lực để tiếp tục bồi dưỡng các kỹ năng đợi lần phỏng vấn tiếp theo.

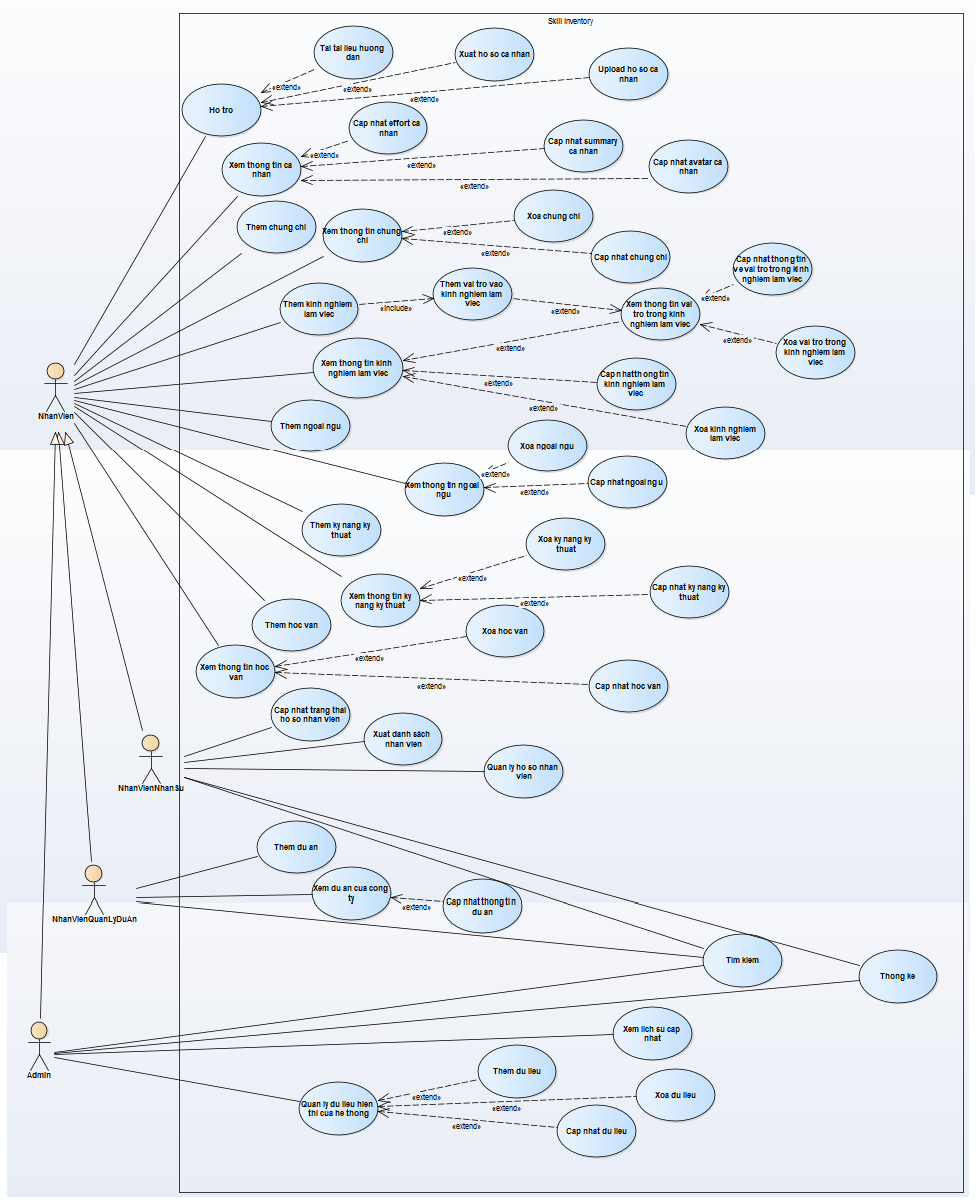
Khi có dự án cần nhân lực nhân viên quản lý dự án sẽ chuẩn bị các thông tin về yêu cầu kỹ năng, chứng chỉ, ngoại ngữ, cũng như thông tin về dự án,  chuyển giao cho nhân viên nhân sự để tiền hành tuyển chọn hồ sơ các nhân viên trong đơn vị nhân lực tìm kiếm các hồ sơ phù hợp sau đó tiến hành phỏng vấn, nếu phỏng vấn thành công nhân viên sẽ được sắp xếp vào dự án, nếu không phù hợp sẽ được đưa về đơn vị nhân lực chuẩn bị cho phỏng vấn dự án khác.

Khi một dự án kết thúc, nếu vẫn chưa có dự án tiếp theo, hoặc chưa có dự án cần nhân lực thì nhân viên cũng sẽ được đưa về bộ phận nhân lực

# : PHÂN TÍCH VÀ THIẾT KẾ

## Mô tả tổng quan ứng dụng

### Mô hình Usecase tổng quát



Hình ‑Usecase tổng quát

### Danh sách các tác nhân và mô tả

|  |  |  |
| --- | --- | --- |
| Tác nhân | Mô tả tác nhân | Ghi chú |
| Nhân viên | Nhân viên của công ty đăng nhập vào hệ thống và dùng các chức năng dành cho nhân viên. |  |
| Nhân viên nhân sự | Là nhân viên của công ty và có thêm các chức năng dành cho quản lý nhân sự. |  |
| Nhân viên quản lý dự án | Là nhân viên của công ty và có thêm các chức năng dành cho quản lý dự án của công ty. |  |
| Nhân viên admin | Là nhân viên của công ty và có thêm các chức năng dành cho quản lý dữ liệu của hệ thống. |  |

Bảng ‑ Danh sách tác nhân và mô tả

### Danh sách Usecase và mô tả

|  |  |  |
| --- | --- | --- |
| ID | Tên Use case | Mô tả ngắn gọn Use case |
| UC001 | Ho tro | Nhóm chức năng hỗ trợ người dùng bao gồm tải tài liệu hướng dẫn, xuất thông tin hồ sơ cá nhân, upload hồ sơ cá nhân. |
| UC001a | Tai tai lieu huong dan | Tải tài liệu hướng dẫn sử dụng hệ thống về. |
| UC001b | Xuat ho so ca nhan | Xuất thông tin hồ sơ ra file excel. |
| UC001c | Upload ho so ca nhan | Upload file thông tin hồ sơ theo mẫu lên hệ thống. |
| UC002 | Xem thong tin ca nhan | Xem thông tin cá nhân trong hồ sơ của bản thân. |
| UC002a | Cap nhat thong tin ca nhan | Cập nhật lại thông tin cá nhân trong hồ sơ của bản thân. |
| UC002b | Cap nhat summary ca nhan | Cập nhật lại thông tin giới thiệu bản thân trong hồ sơ của. |
| UC002c | Cap nhat avatar ca nhan | Cập nhật lại ảnh đại diện cá nhân. |
| UC003 | Them chung chi | Thêm thông tin chứng chỉ của bản thân vào hồ sơ. |
| UC004 | Xem thong tin chung chi | Xem chi tiết thông tin các chứng chỉ có trong hồ sơ của bản thân. |
| UC004a | Xoa chung chi | Xóa thông tin chứng chỉ trong hồ sơ của bản thân. |
| UC004b | Cap nhat chung chi | Cập nhật thông tin chứng chỉ trong hồ sơ của bản thân. |
| UC005 | Them kinh nghiem lam viec | Thêm thông tin các dự án đã làm vào hồ sơ. |
| UC005a | Them vai tro vao kinh nghiem lam viec | Thêm thông tin các vai trò trong dự án đã làm vào hồ sơ. |
| UC006 | Xem thong tin kinh nghiem lam viec | Xem chi tiết thông tin dự án đã làm trong hồ sơ cá nhân. |
| UC006a | Xem thong tin vai tro trong kinh nghiem lam viec | Xem chi tiết thông tin các vai trò trong dự án đã làm. |
| UC006a1 | Cap nhat thong tin ve vai tro trong kinh nghiem lam viec | Cập nhật thông tin về vai trò trong dự án đã làm trong hồ sơ cá nhân. |
| UC006a2 | Xoa vi tri trong kinh nghiem lam viec | Xóa thông tin vai trò trong dự án đã làm. |
| UC006b | Cap nhat thong tin kinh nghiem lam viec | Cập nhật thông tin các dự án đã làm trong hồ sơ cá nhân. |
| UC006c | Xoa kinh nghiem lam viec | Xóa thông tin về dự án đã làm trong hồ sơ cá nhân. |
| UC007 | Them ngoai ngu | Thêm thông tin kỹ năng ngoại ngữ của bản thân vào hồ sơ. |
| UC008 | Xem thong tin ngoai ngu | Xem chi tiết thông tin kỹ năng ngoại ngữ của bản thân. |
| UC008a | Xoa ngoai ngu | Xóa thông tin kỹ năng ngoại ngữ của bản thân. |
| UC008b | Cap nhat ngoai ngu | Cập nhật thông tin kỹ năng ngoại ngữ của bản thân trong hồ sơ. |
| UC009 | Them ky nang ky thuat | Thêm thông tin về kỹ năng kỹ thuật của bản thân vào hồ sơ. |
| UC010 | Xem thong tin ky nang ky thuat | Xem chi tiết thông tin kỹ năng kỹ thuật của bản thân trong hồ sơ. |
| UC010a | Xoa ky nang ky thuat | Xóa thông tin kỹ năng kỹ thuật của bản thân. |
| UC010b | Cap nhat ky nang ky thuat | Cập nhật thông tin kỹ năng kỹ thuật của bản thân trong hồ sơ. |
| UC011 | Them hoc van | Thêm thông tin về học vấn của bản thân vào hồ sơ. |
| UC012 | Xem thong tin hoc van | Xem chi tiết thông tin trình độ học vấn của bản thân trong hồ sơ. |
| UC012a | Xoa hoc van | Xóa thông tin học vấn của bản thân. |
| UC012b | Cap nhat hoc van | Cập nhật thông tin học vấn của bản thân trong hồ sơ. |
| UC013 | Cap nhat trang thai ho so nhan vien | Cập nhật trạng thái hồ sơ của nhân viên. |
| UC014 | Xuat danh sach nhan vien | Xuất danh sách thông tin nhân viên ra file excel. |
| UC015 | Quan ly ho so nhan vien | Cập nhật thông tin về thông tin cá nhân, kỹ năng kỹ thuật, kỹ năng ngoại ngữ, chứng chỉ, bằng cấp, kinh nghiệm làm việc của nhân viên. |
| UC016 | Them du an | Thêm thông tin dự án của công ty vào hệ thống. |
| UC017 | Xem du an cua cong ty | Xem chi tiết thông tin dự án của công ty mà nhân viên đang phụ trách. |
| UC017a | Cap nhat thong tin du an | Cập nhật thông tin dự án của công ty mà nhân viên đang phụ trách. |
| UC018 | Tim kiem | Tìm kiếm nhân viên theo nhiều tiêu chí. |
| UC019 | Thong ke | Thống kê dữ liệu theo nhiều tiêu chí. |
| UC020 | Xem lich su cap nhat | Xem thông tin về lịch sử cập nhật, sửa, xóa dữ liệu trong hồ sơ của nhân viên. |
| UC021 | Quan ly du lieu hien thi cua he thong | Quản lý dữ liệu hiển thị của hệ thống. |
| UC021a | Them du lieu | Thêm thông tin dữ liệu hiển thị của hệ thống. |
| UC021b | Xoa du lieu | Xóa thông tin dữ liệu hiển thị của hệ thống. |
| UC021c | Cap nhat du lieu | Cập nhật thông tin dữ liệu hiển thị của hệ thống. |

Bảng ‑ Danh sách Usecase và mô tả

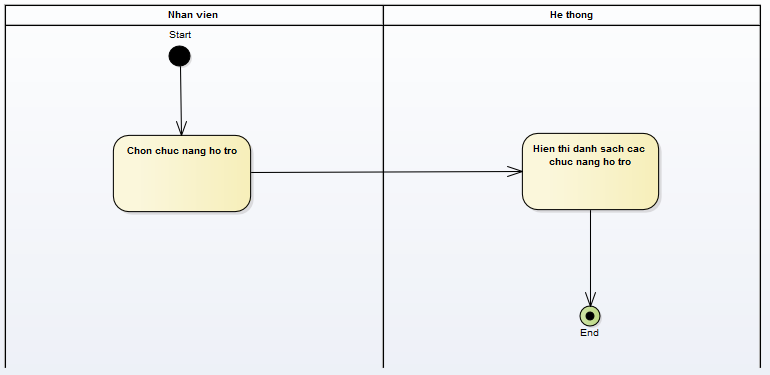
## Đặc tả các yêu cầu chức năng

### UC001 Ho tro

|  |  |  |
| --- | --- | --- |
| Use case: UC001 – Ho tro | | |
| Mục đích: | Hỗ trợ nhân viên một số tính năng mở rộng mà hệ thống cung cấp. | |
| Tác nhân: | Nhân viên. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Truy cập được mục hỗ trợ của hệ thống. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên truy cập vào hệ thống chọn chức năng hỗ trợ. |  |
|  | 2. Hệ thống hiển thị các chức năng hỗ trợ nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

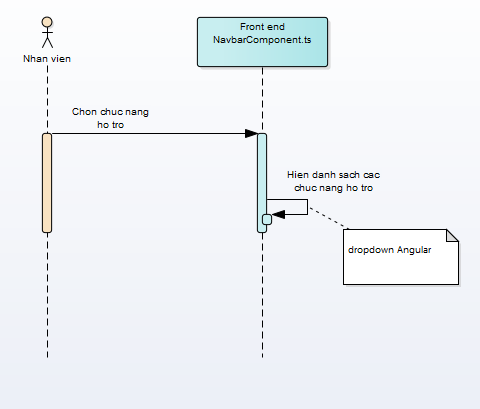
Bảng 3‑3 Đặc tả UC001

* Activity diagram:



Hình ‑ Activity diagram UC001

* Sequence diagram:



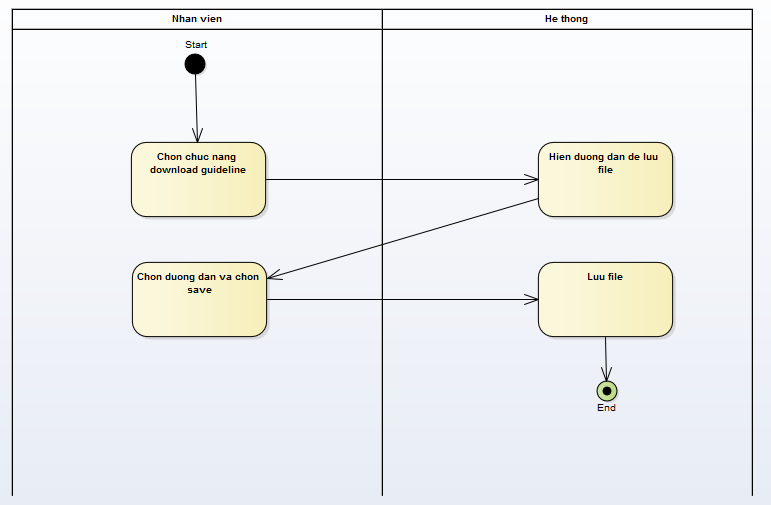
Hình ‑ Sequence diagram UC001

### UC001a Tai tai lieu huong dan

|  |  |  |
| --- | --- | --- |
| Use case: UC001a – Tai tai lieu huong dan | | |
| Mục đích: | Hỗ trợ nhân viên tải về tài liệu hướng dẫn sử dụng hệ thống, file định dạng là .pptx – powerpoint. | |
| Tác nhân: | Nhân viên. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Tải thành công file hướng dẫn sử dụng hệ thống. | |
| Luồng sự kiện chính (Basic flows) | Nhân viên | Hệ thống |
| 1. Nhân viên chọn chức năng download guideline. |  |
|  | 1. Hệ thống hiện đường dẫn để lưu file. |
| 1. Nhân viên chọn đường dẫn để lưu file và chọn save. |  |
|  |  | 1. Lưu file. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

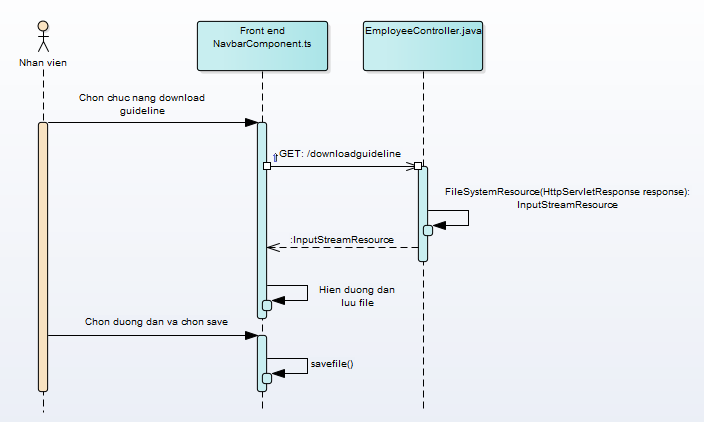
Bảng ‑ Đặc tả UC001a

* Activity diagram:



Hình ‑ Activity diagram UC001a

* Sequence diagram:



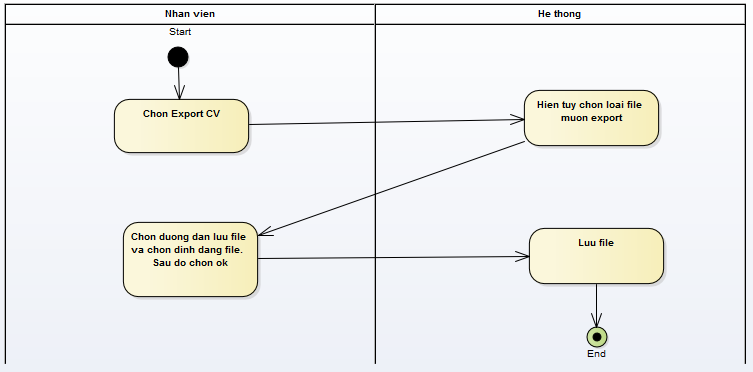
Hình ‑ Sequence diagram UC001a

### UC001b Xuat ho so ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC001b – Xuat ho so ca nhan | | |
| Mục đích: | Nhân viên xuất thông tin về các kỹ năng, thông tin cá nhân ra file mềm như pdf, excel, word. | |
| Tác nhân: | Nhân viên. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xuất thành công thông tin hồ sơ ra file. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn export cv. |  |
|  | 1. Hệ thống hiện tùy chọn loại file muốn export. |
| 1. Nhân viên chọn loại file và chọn ok. |  |
|  | 1. Lưu file. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

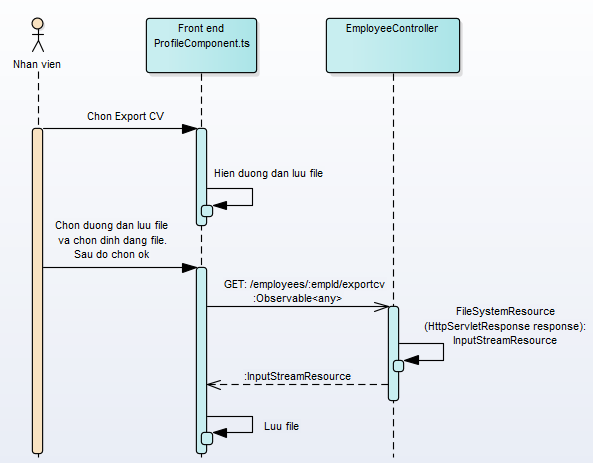
Bảng ‑ Đặc tả UC001b

* Activity diagram:



Hình ‑ Activity diagram UC001b

* Sequence diagram:



Hình ‑ Sequence diagram UC001b

### UC001c Upload ho so ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC001c – Upload ho so ca nhan | | |
| Mục đích: | Nhân viên upload thông tin hồ sơ từ file mềm theo template vào hệ thống. | |
| Tác nhân: | Nhân viên. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Tải lên thành công file hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn chức năng upload CV |  |
|  | 1. Hệ thống hiện đường dẫn để chọn file muốn upload. |
| 1. Nhân viên chọn file cần upload và chọn Open |  |
|  | 1. Hệ thống cập nhật dữ liệu trong hồ sơ. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xảy ra lỗi trong quá trình đọc dữ liệu từ file sẽ thông báo lỗi và reset dữ liệu ban đầu của hồ sơ. |

Bảng ‑ Đặc tả UC001c

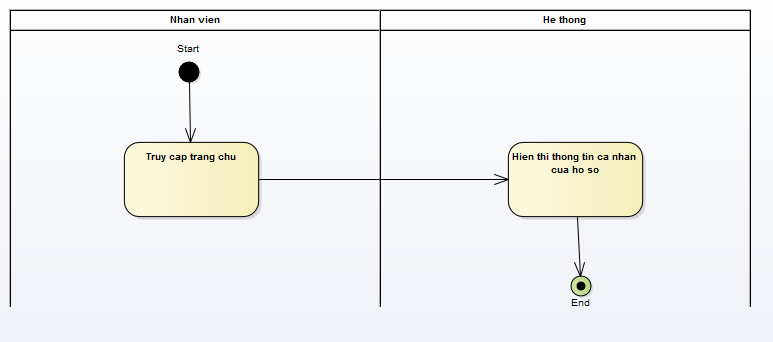
* Activity diagram:
* Sequence diagram:

### UC002 Xem thong tin ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC002 – Xem thong tin ca nhan | | |
| Mục đích: | Nhân viên xem thông tin cá nhân của mình trong hồ sơ.  Nếu quyền là nhân viên nhân sự hoặc nhân viên quản lý dự án sẽ xem được thông tin cá nhân của các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin cá nhân trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiển thị thông tin cá nhân của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

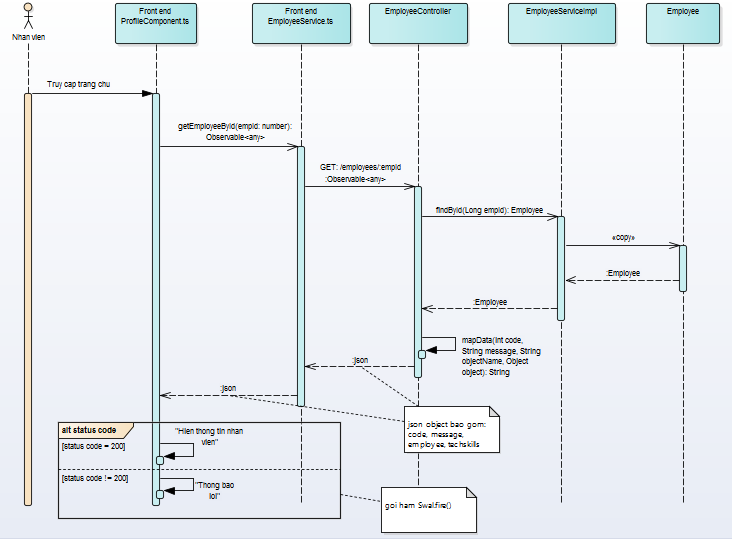
Bảng ‑ Đặc tả UC002

* Activity diagram:



Hình ‑ Activity diagram UC002

* Sequence diagram:



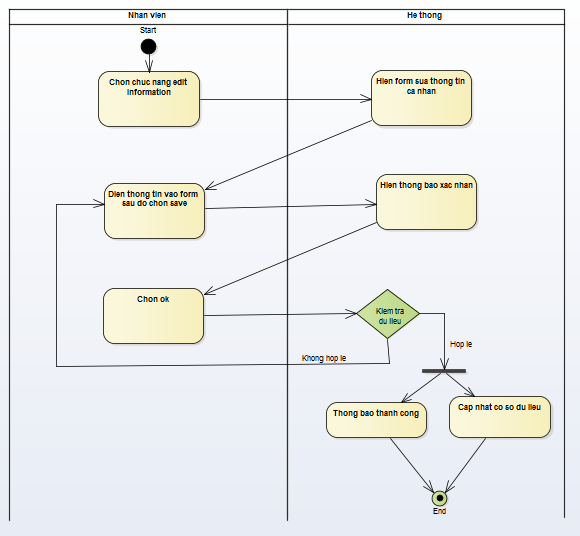
Hình ‑ Sequence diagram UC002

### UC002a Cap nhat thong tin ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC002a – Cap nhat thong tin ca nhan | | |
| Mục đích: | Nhân viên chỉnh sửa thông tin cá nhân của mình trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể sửa thông tin cá nhân này của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin cá nhân trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn vào chức năng Edit Information. |  |
|  | 1. Hệ thống hiện form sửa thông tin cá nhân. |
| 1. Nhân viên điền thông tin vào form sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

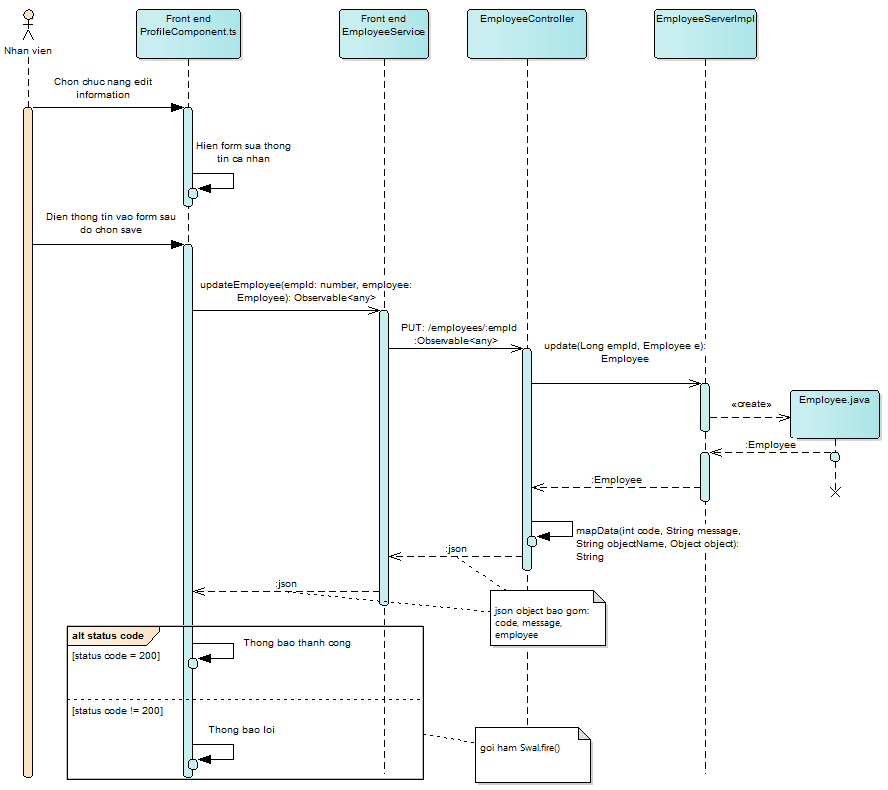
Bảng ‑ Đặc tả UC002a

* Activity diagram:



Hình ‑ Activity diagram UC002a

* Sequence diagram:



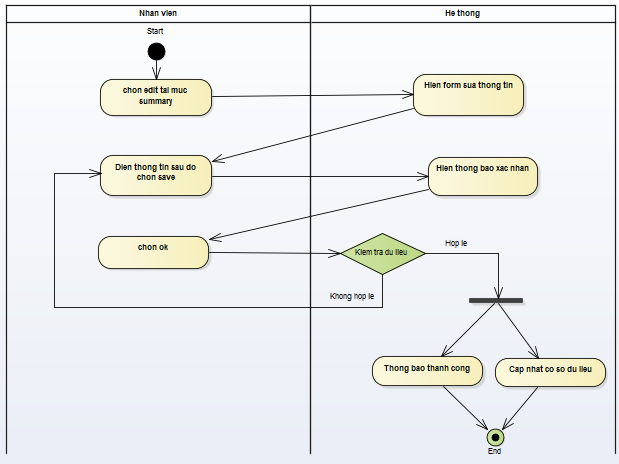
Hình ‑ Sequence diagram UC002a

### UC002b Cap nhat summary ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC002b – Cap nhat summary ca nhan | | |
| Mục đích: | Nhân viên chỉnh sửa thông tin summary của mình trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể sửa thông tin cá nhân này của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công summary trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn vào icon edit tại mục summary. |  |
|  | 1. Hệ thống hiện form chỉnh sửa summary. |
| 1. Nhân viên điền thông tin sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn OK. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

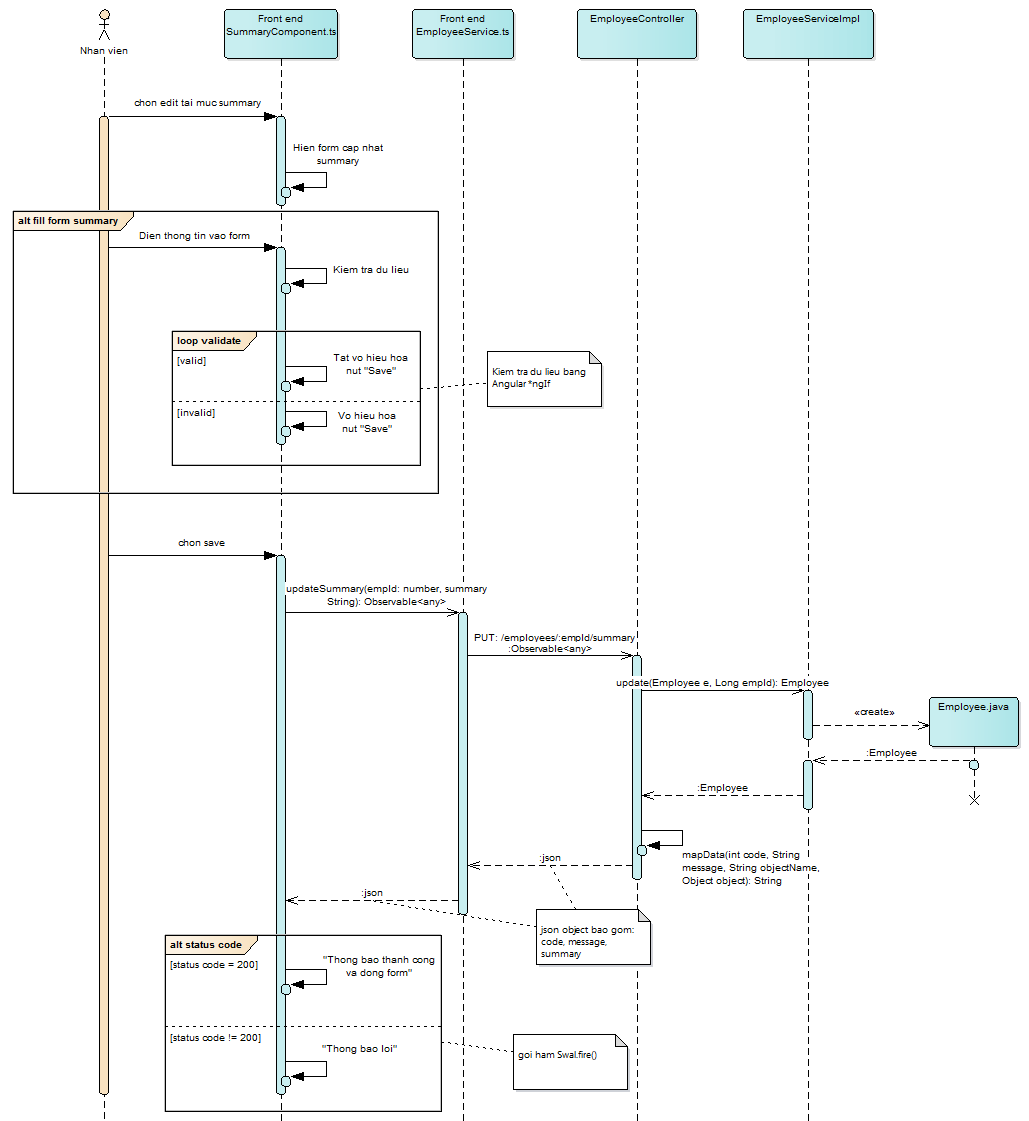
Bảng ‑ Đặc tả UC002b

* Activity diagram:



Hình ‑ Activity diagram UC002b

* Sequence diagram:



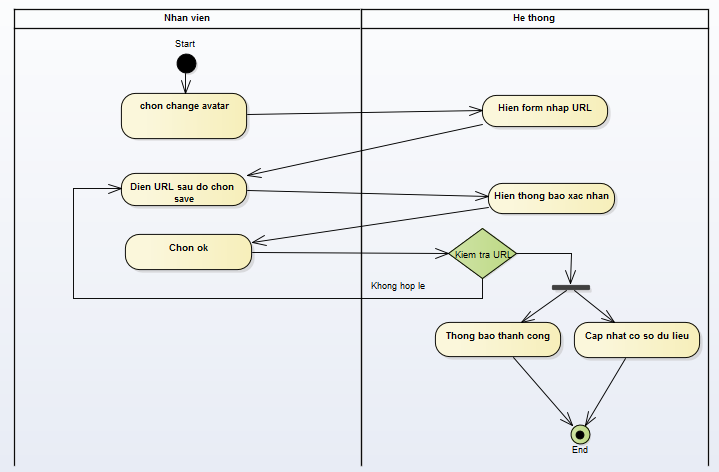
Hình ‑ Sequence diagram UC002b

### UC002c Cap nhat avatar ca nhan

|  |  |  |
| --- | --- | --- |
| Use case: UC002c – Cap nhat avatar ca nhan | | |
| Mục đích: | Nhân viên chỉnh sửa hình đại diện của mình trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể sửa thông tin cá nhân này của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công ảnh đại diện trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn change avatar. |  |
|  | 1. Hệ thống hiện form nhập URL hình ảnh(online). |
| 1. Nhân viên điền URL ảnh sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

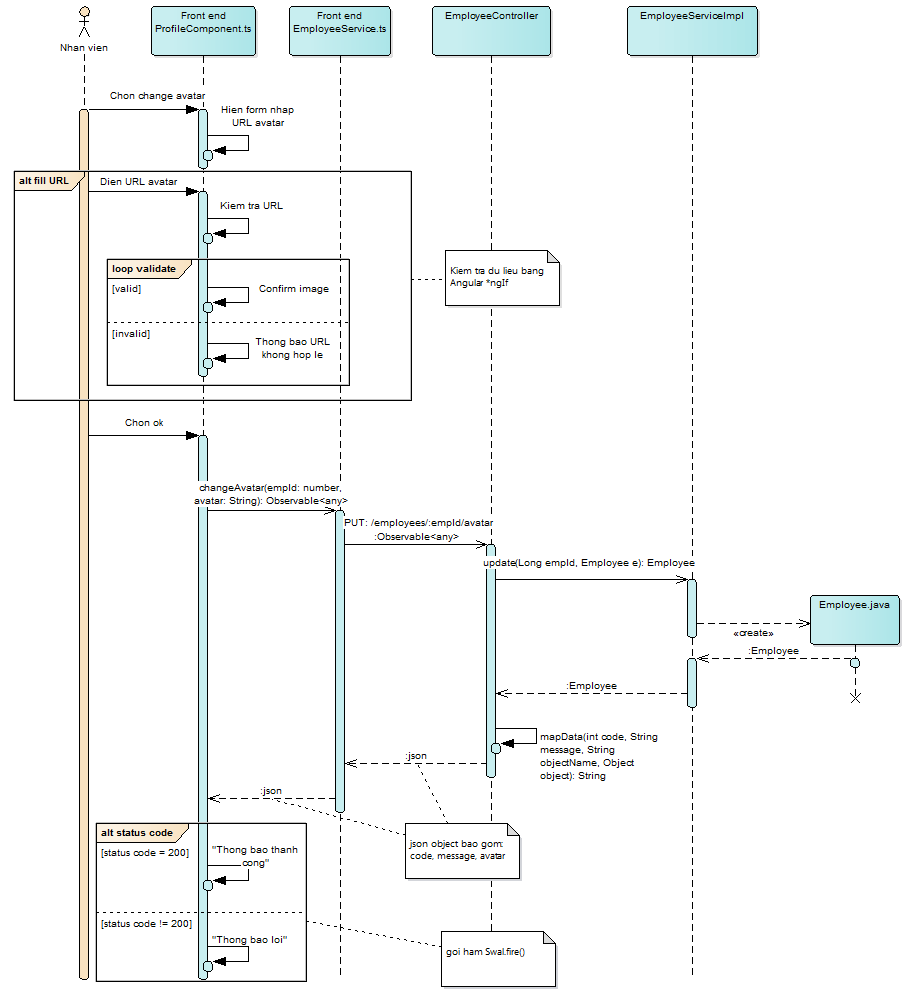
Bảng ‑ Đặc tả UC002c

* Activity diagram:



Hình ‑ Activity diagram UC002c

* Sequence diagram:



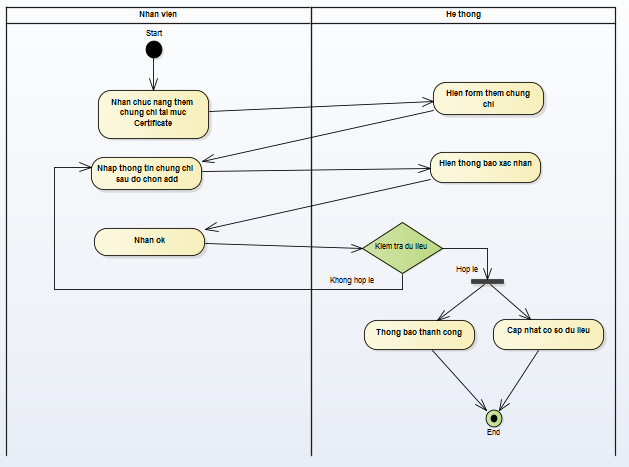
Hình ‑ Sequence diagram UC002c

### UC003 Them chung chi

|  |  |  |
| --- | --- | --- |
| Use case: UC003 – Them chung chi | | |
| Mục đích: | Nhân viên thêm thông tin chứng chỉ vào hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể thêm chứng chỉ vào hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Thêm thành công thông tin chứng chỉ vào hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn chức năng thêm chứng chỉ tại mục Certificate. |  |
|  | 1. Hệ thống hiện form thêm chứng chỉ. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add. |  |
|  | 1. Hệ thống hiện thông báo xác nhận thêm hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Thêm không thành công sẽ thông báo lỗi. |

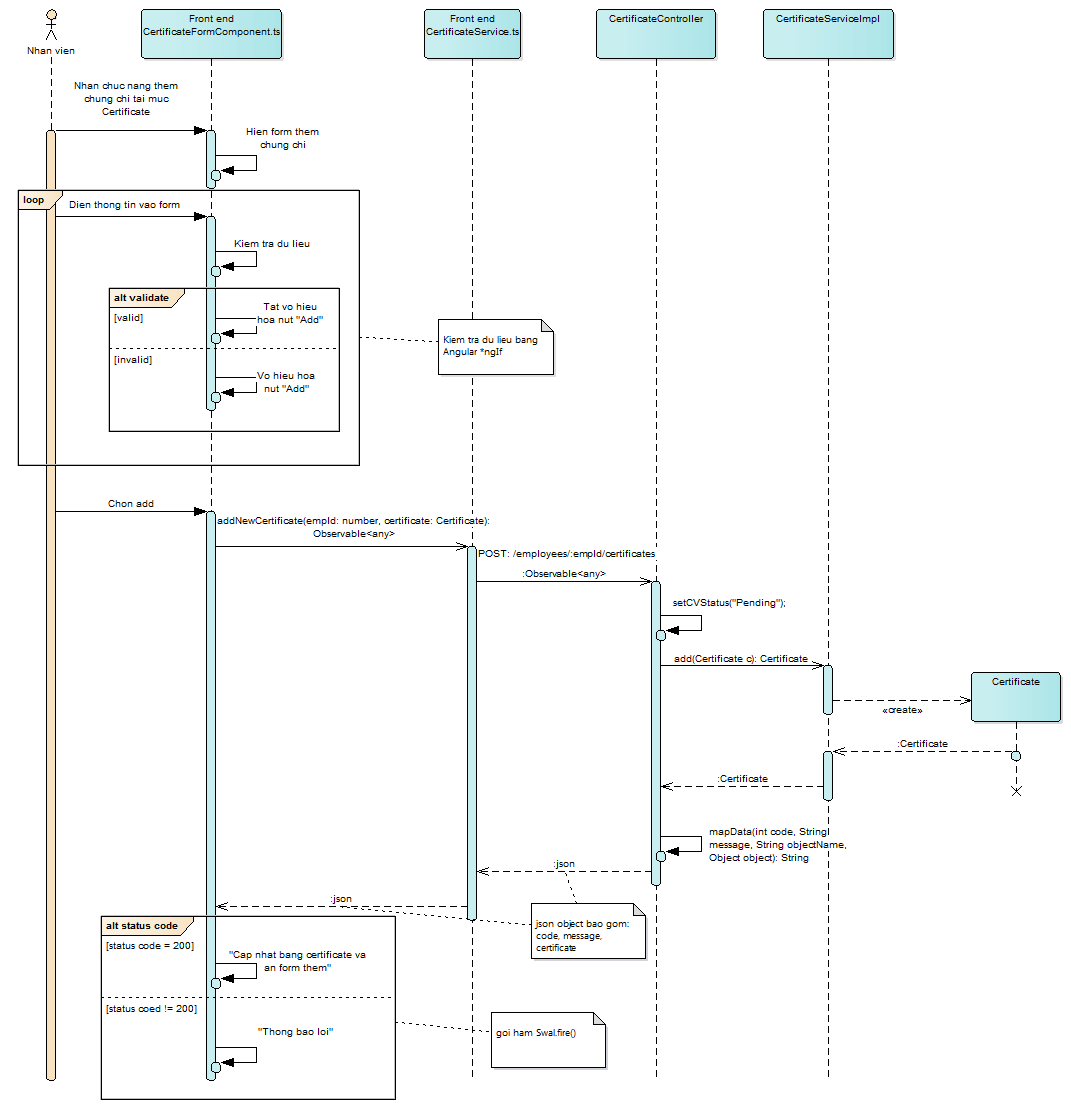
Bảng ‑ Đặc tả UC003

* Activity diagram:



Hình ‑ Activity diagram UC003

* Sequence diagram:



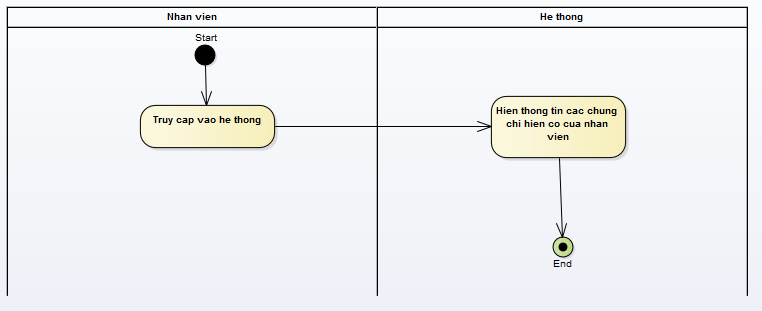
Hình ‑ Sequence diagram UC003

### UC004 Xem thong tin chung chi

|  |  |  |
| --- | --- | --- |
| Use case: UC004 – Xem thong tin chung chi | | |
| Mục đích: | Nhân viên xem chi tiết thông tin chứng chỉ trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể xem chứng chỉ trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin chứng chỉ của hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiện thông tin các chứng chỉ hiện có của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

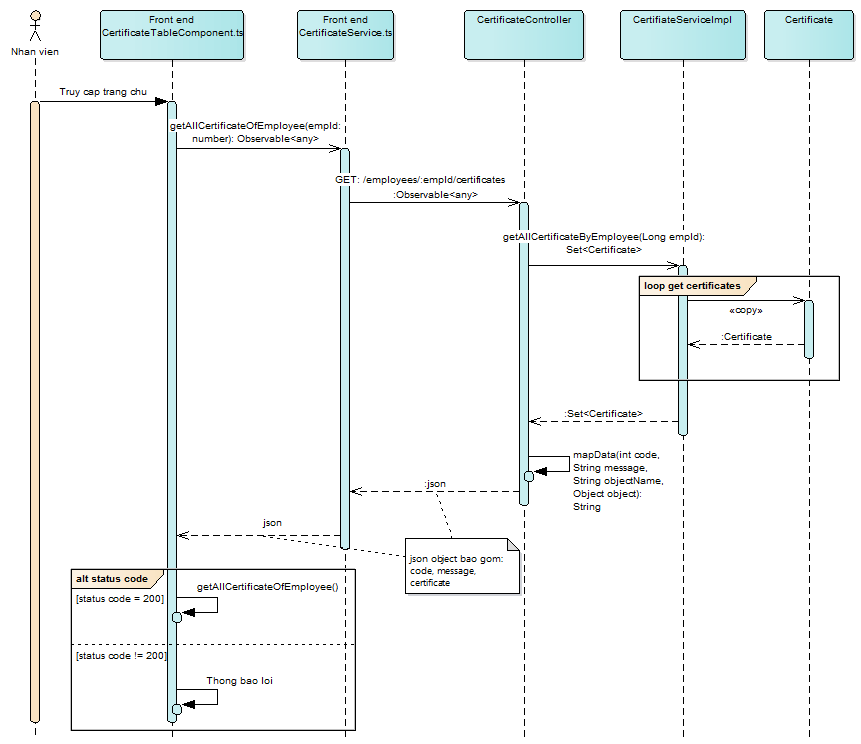
Bảng ‑ Đặc tả UC004

* Activity diagram:



Hình ‑ Activity diagram UC004

* Sequence diagram:



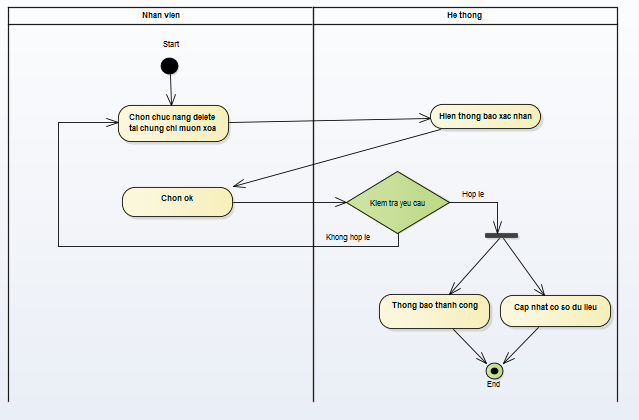
Hình ‑ Sequence diagram UC004

### UC004a Xoa chung chi

|  |  |  |
| --- | --- | --- |
| Use case: UC004a – Xoa chung chi | | |
| Mục đích: | Nhân viên xóa thông tin chứng chỉ trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin chứng chỉ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công chứng chỉ trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn chọn chức năng delete tại chứng chỉ muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi. |

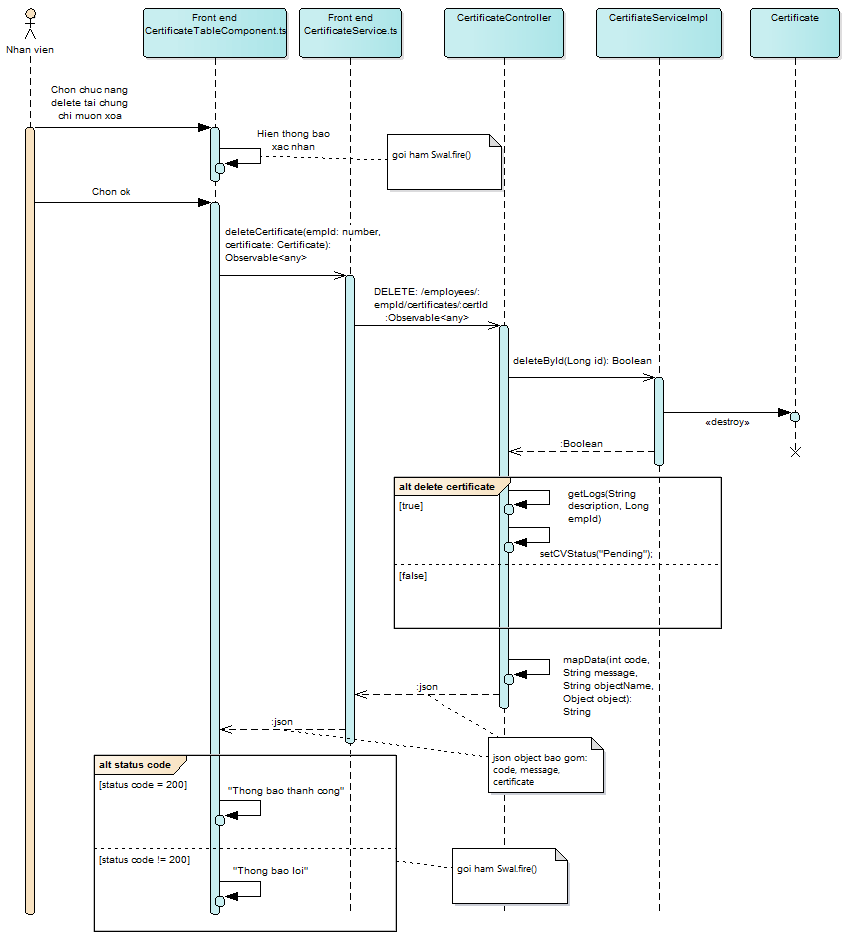
Bảng ‑ Đặc tả UC004a

* Activity diagram:



Hình ‑ Activity diagram UC004a

* Sequence diagram:



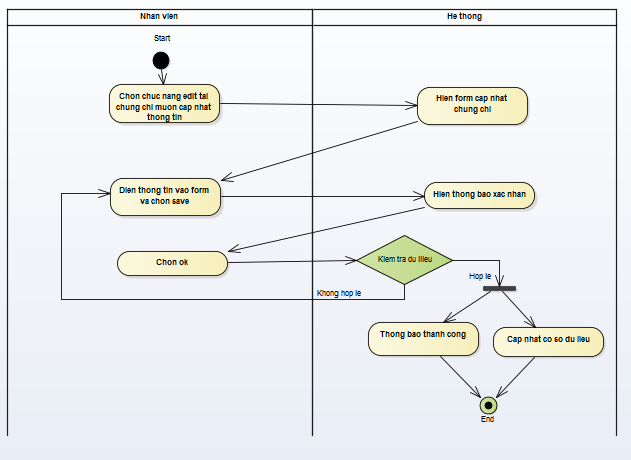
Hình ‑ Sequence diagram UC004a

### UC004b Cap nhat chung chi

|  |  |  |
| --- | --- | --- |
| Use case: UC004b – Cap nhat chung chi | | |
| Mục đích: | Nhân viên cập nhật thông tin chứng chỉ trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin chứng chỉ trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin chứng chỉ trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn chức năng edit tại chứng chỉ muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật chứng chỉ. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

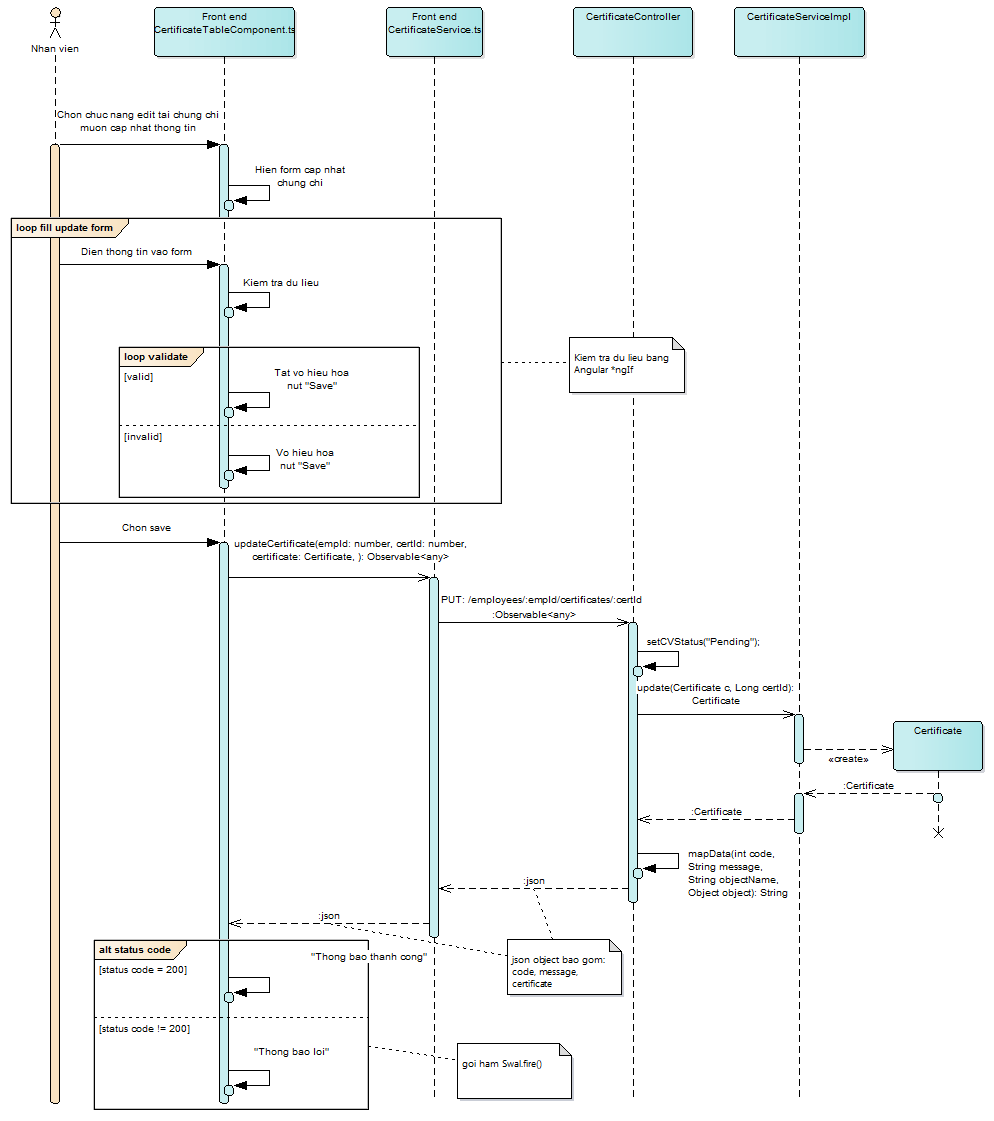
Bảng ‑ Đặc tả UC004b

* Activity diagram:



Hình ‑ Activity diagram UC004b

* Sequence diagram:



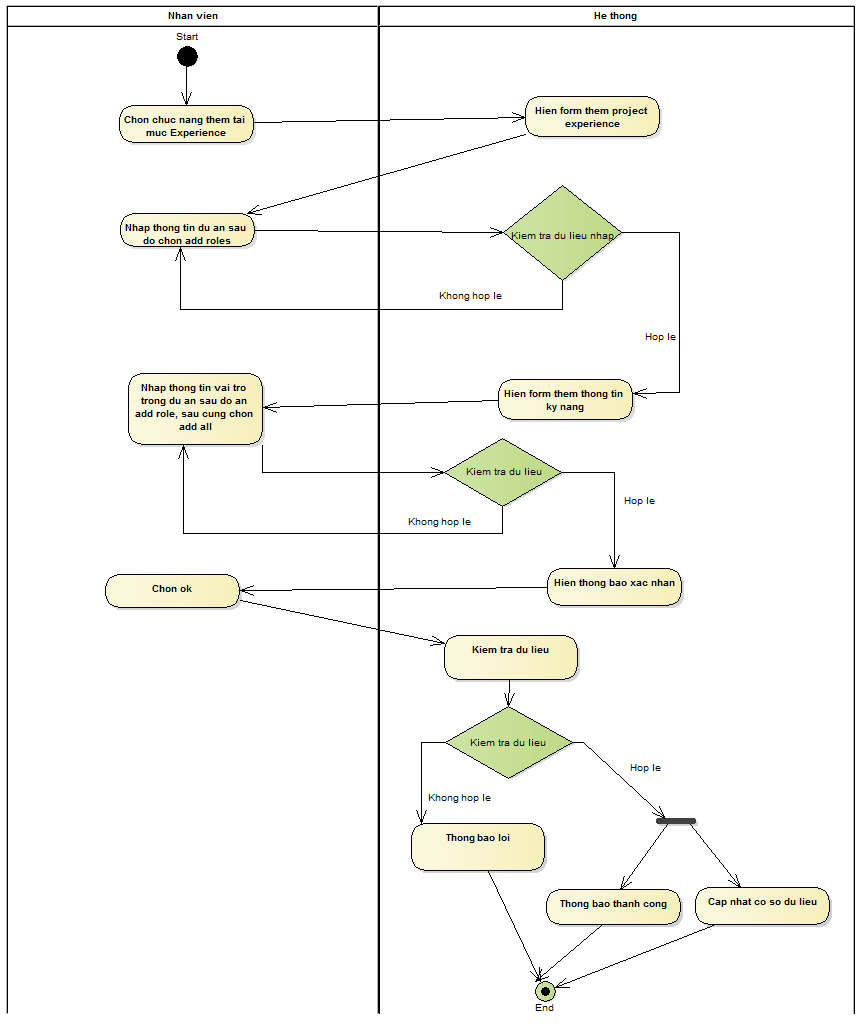
Hình ‑ Sequence diagram UC004b

### UC005 Them kinh nghiem lam viec, UC005a Them vai tro vao kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC005 – Them kinh nghiem lam viec  Use case: UC005a – Them vai tro vao kinh nghiem lam viec | | |
| Mục đích: | Nhân viên thêm thông tin kinh nghiệm làm việc và vai trò về các dự án đã làm vào hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể thêm thông tin này vào hồ sơ của bản thân và các hồ sơ nhân viên khác.  Kinh nghiệm làm việc cũng có thể được hệ thống tự động thêm vào nếu nhân viên có tham gia dự án của công ty và trạng thái của dự án đó là “Fineshed”. Chi tiết ở UC017a. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Thêm thành công thông tin kinh nghiệm và các vai trò trong dự án đã làm vào hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào chức năng thêm tại mục Experience. |  |
|  | 1. Hệ thống hiện form thêm project experience. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add roles. |  |
|  | 1. Hệ thống hiện form thêm vai trò trong dự án vừa nhập. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add roles. Có thể điền nhiều role. Sau cùng chọn add all. |  |
|  | 1. Hệ thống hiện thông báo xác nhận thêm hay không? |
| 1. Nhân viên ấn “OK”. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Thêm không thành công sẽ thông báo lỗi. |

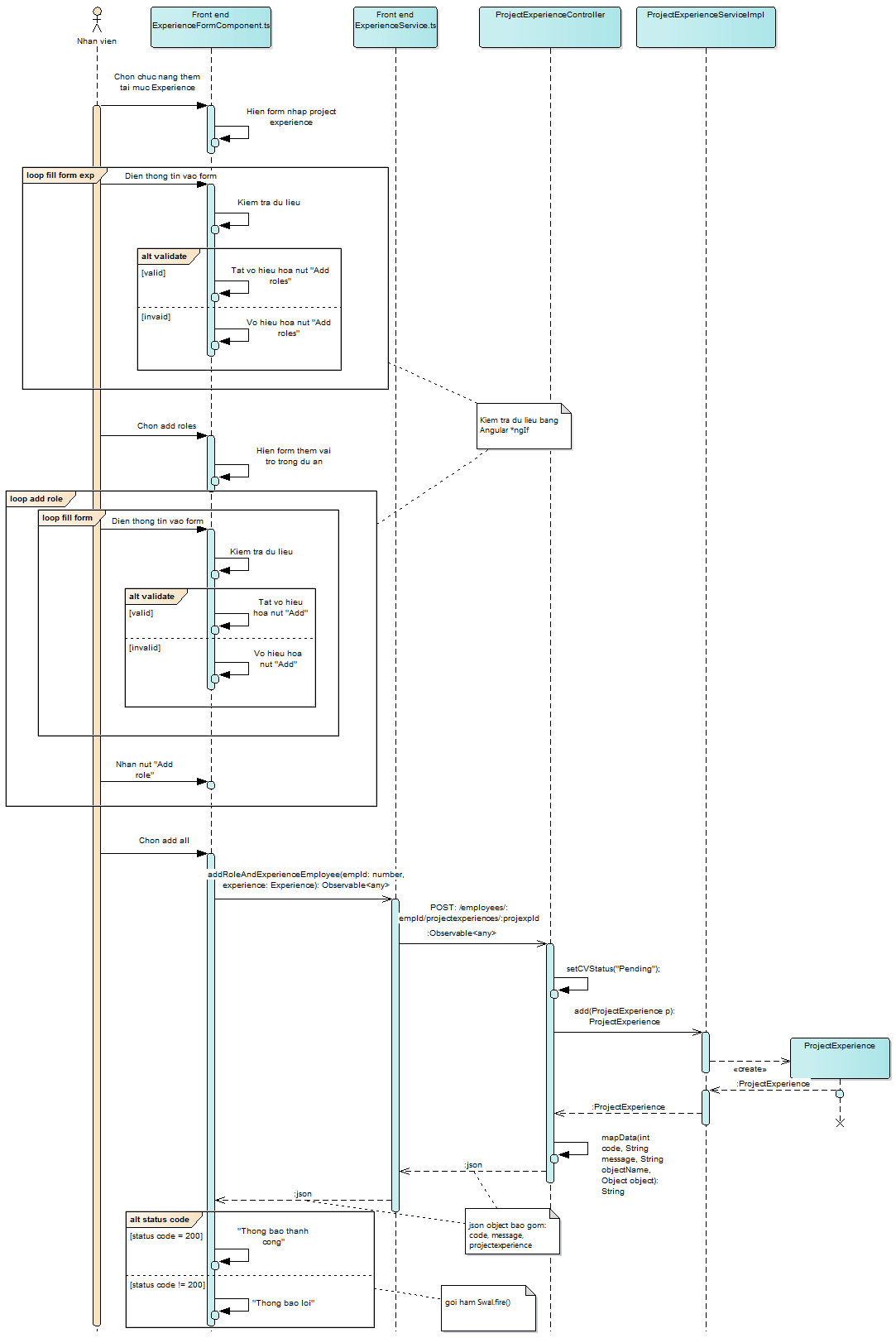
Bảng ‑ Đặc tả UC005, UC005a

* Activity diagram:



Hình ‑ Activity diagram UC005

* Sequence diagram:



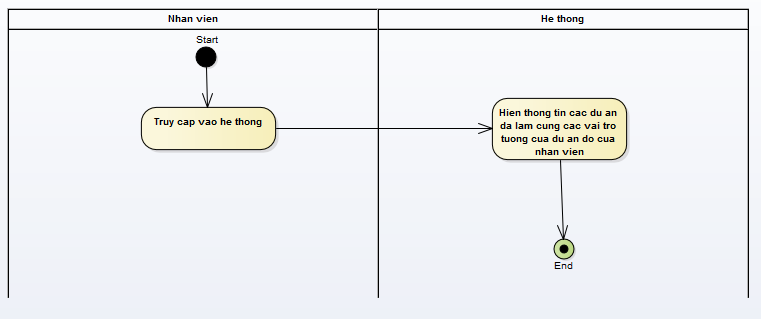
Hình ‑ Sequence diagram UC005

### UC006 Xem thong tin kinh nghiem lam viec, UC006a Xem thong tin vai tro trong kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC006 – Xem thong tin kinh nghiem lam viec  Use case: UC006a – Xem thong tin vai tro trong kinh nghiem lam viec | | |
| Mục đích: | Nhân viên xem chi tiết thông tin các dự án và các vai trò tương ứngg trong dự án đó trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể xem kinh nghiệm làm việc trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin kinh nghiệm làm việc của hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiện thông tin các dự án cùng các vai trò tương ứng trong dự án đã làm hiện có của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

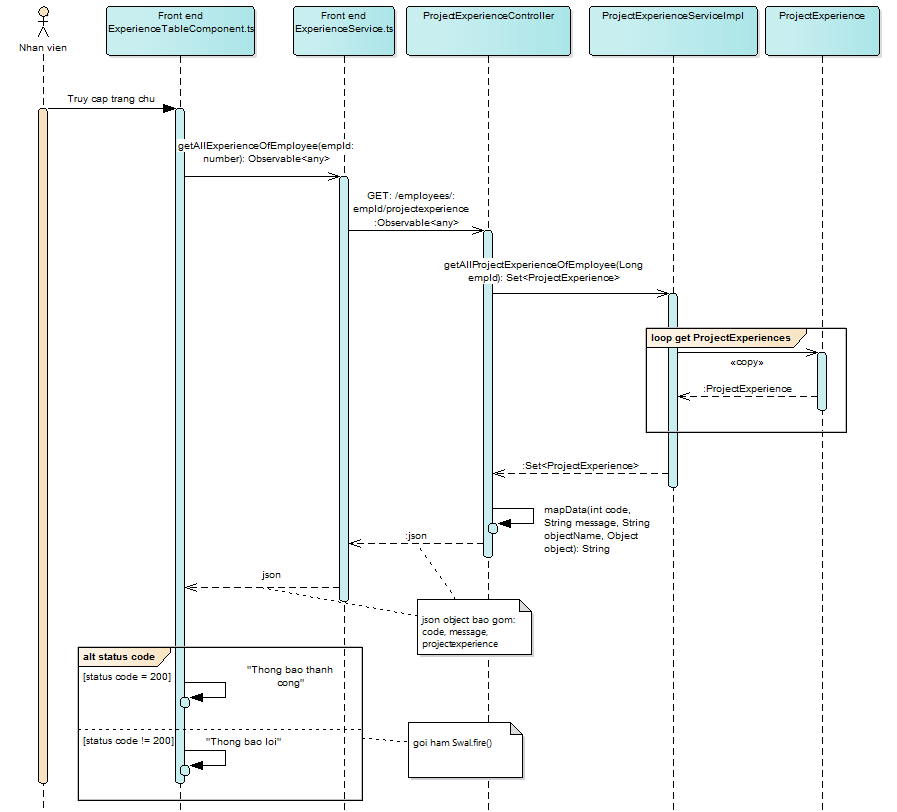
Bảng ‑ Đặc tả UC006, UC006a

* Activity diagram:



Hình ‑ Activity diagram UC006

* Sequence diagram:



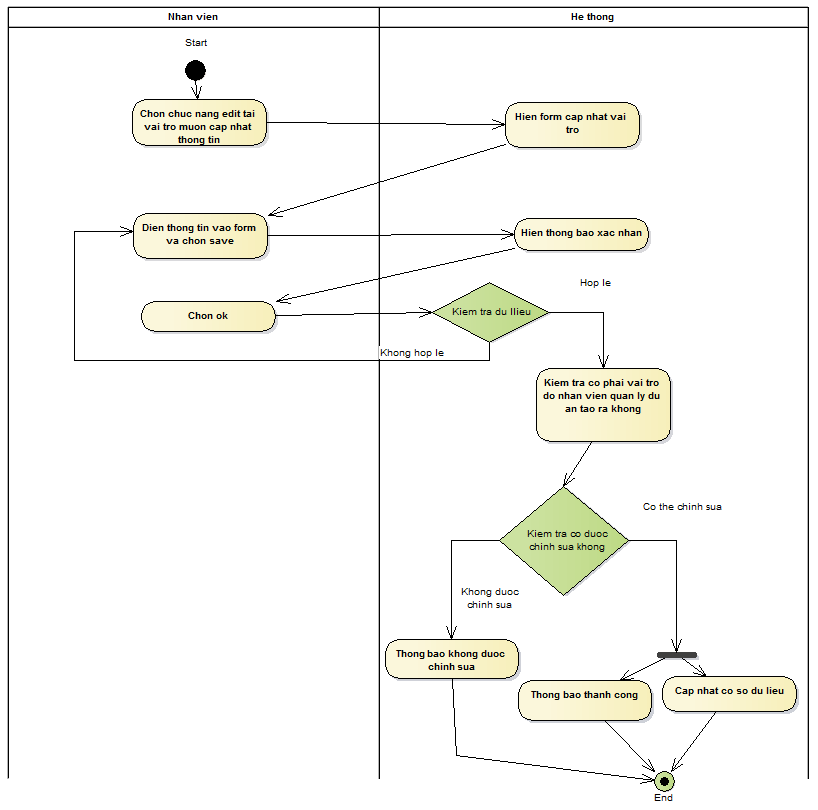
Hình ‑ Sequence diagram UC006

### UC006a1 Cap nhat thong tin ve vai tro trong kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC006a1 – Cap nhat thong tin ve vai tro trong kinh nghiem lam viec | | |
| Mục đích: | Nhân viên cập nhật thông tin vai trò của dự án kinh nghiệm làm việc trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin vai trò trong hồ sơ của bản thân và các hồ sơ nhân viên khác.  Nếu dự án kinh nghiệm làm việc do hệ thống tự động thêm bởi sự quản lý của nhân viên quản lý dự án, thì nhân viên không cập nhật được vai trò trong dự án này. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin vai trò trong dự án trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào icon edit tại vai trò muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật vai trò. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi.   2. Dự án do nhân viên quản lý dự án quản lý sẽ thông báo không có quyền cập nhật. |

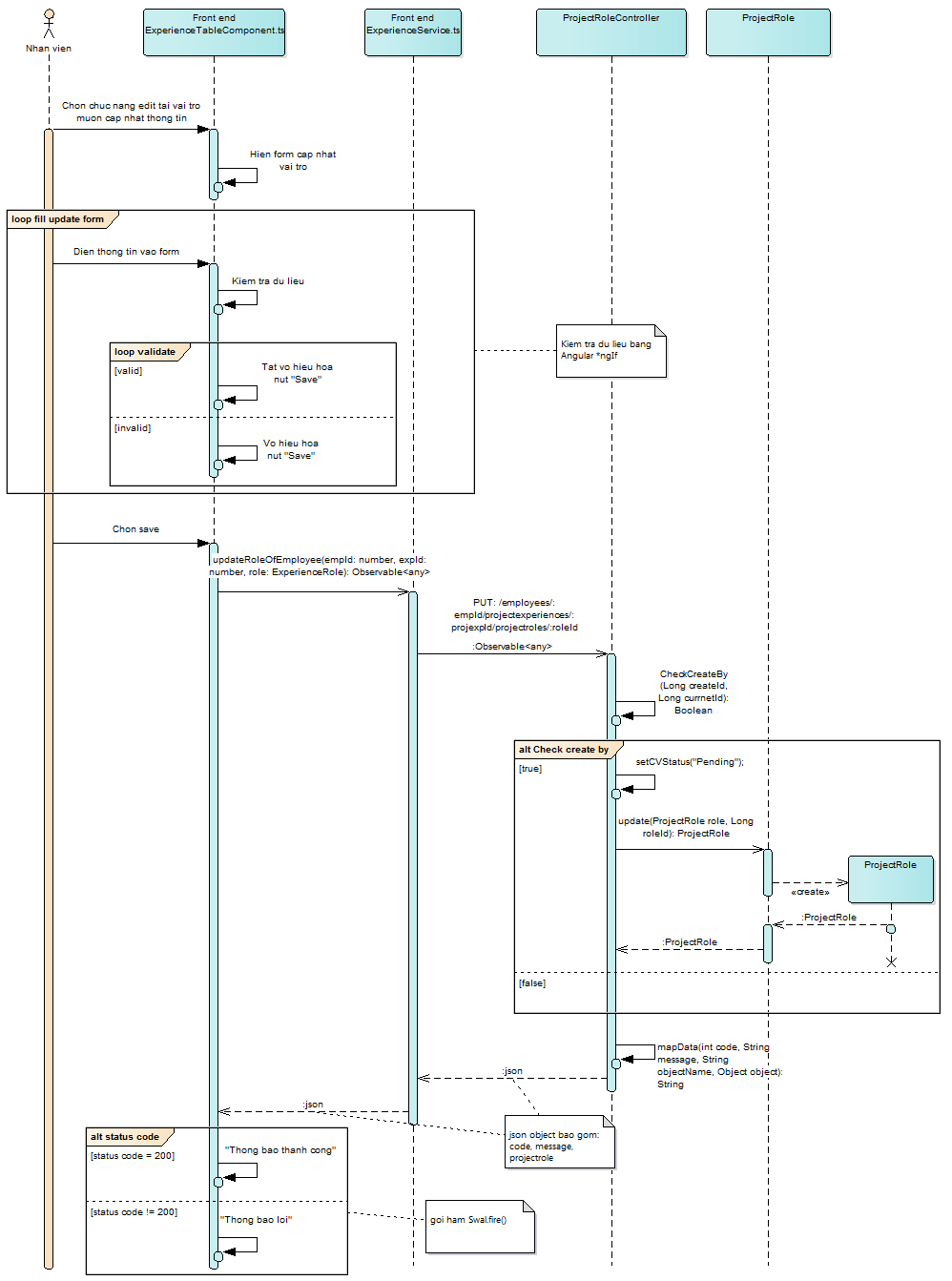
Bảng ‑ Đặc tả UC006a1

* Activity diagram:



Hình ‑ Activity diagram UC006a1

* Sequence diagram:



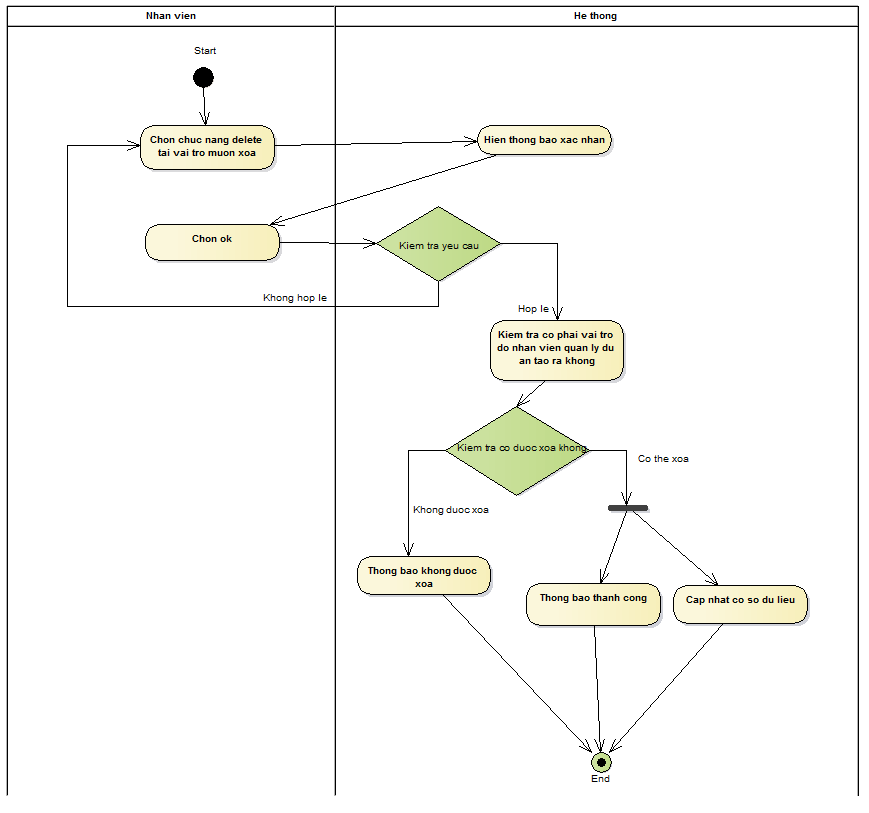
Hình ‑ Sequence diagram UC006a1

### UC006a2 Xoa vai tro trong kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC006a2 – Xoa vai tro trong kinh nghiem lam viec | | |
| Mục đích: | Nhân viên xóa thông tin vai trò của dự án đã làm trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin vai trò của bản thân và các hồ sơ nhân viên khác.  Nếu dự án kinh nghiệm làm việc do hệ thống tự động thêm bởi sự quản lý của nhân viên quản lý dự án, thì nhân viên không xóa được vai trò trong dự án này. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công vai trò trong dự án đã làm trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn chưc năng xóa tại vai trò muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi.   2. Dự án do nhân viên quản lý dự án quản lý sẽ thông báo không có quyền xóa. |

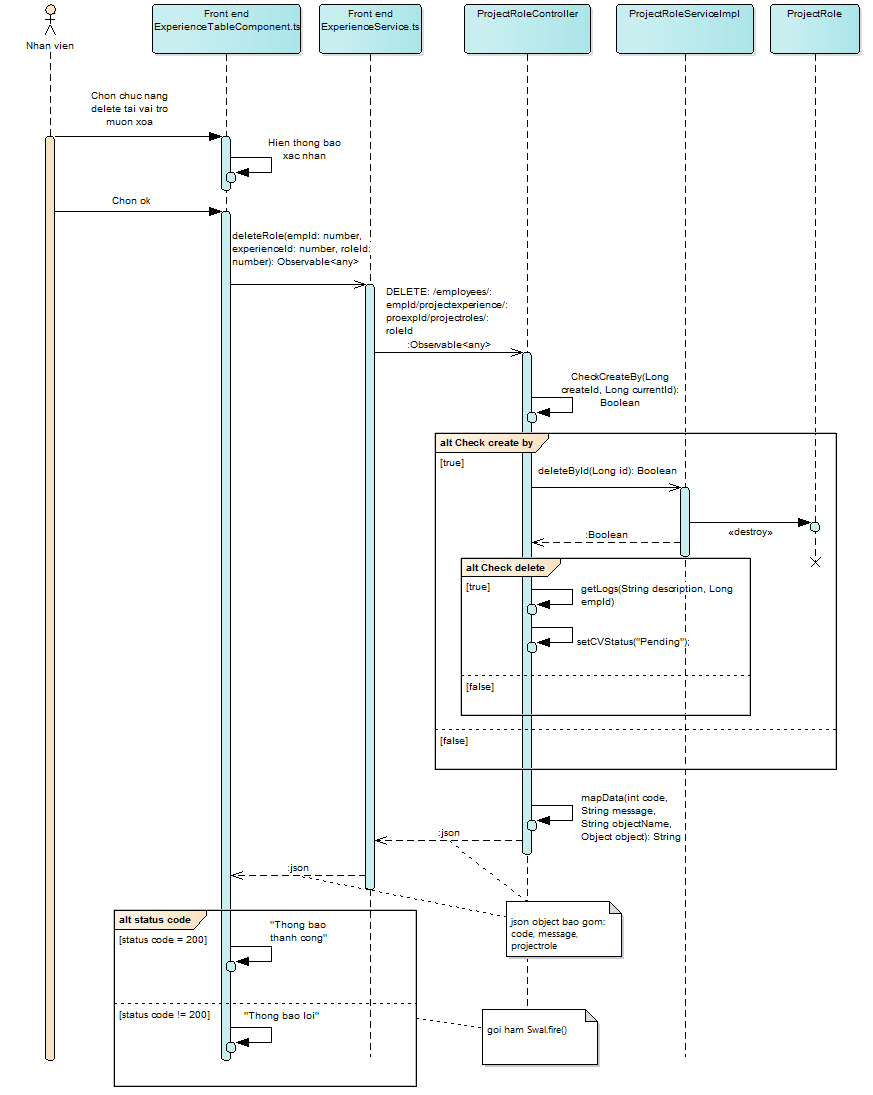
Bảng ‑ Đặc tả UC006a2

* Activity diagram:



Hình ‑ Activity diagram UC006a2

* Sequence diagram:



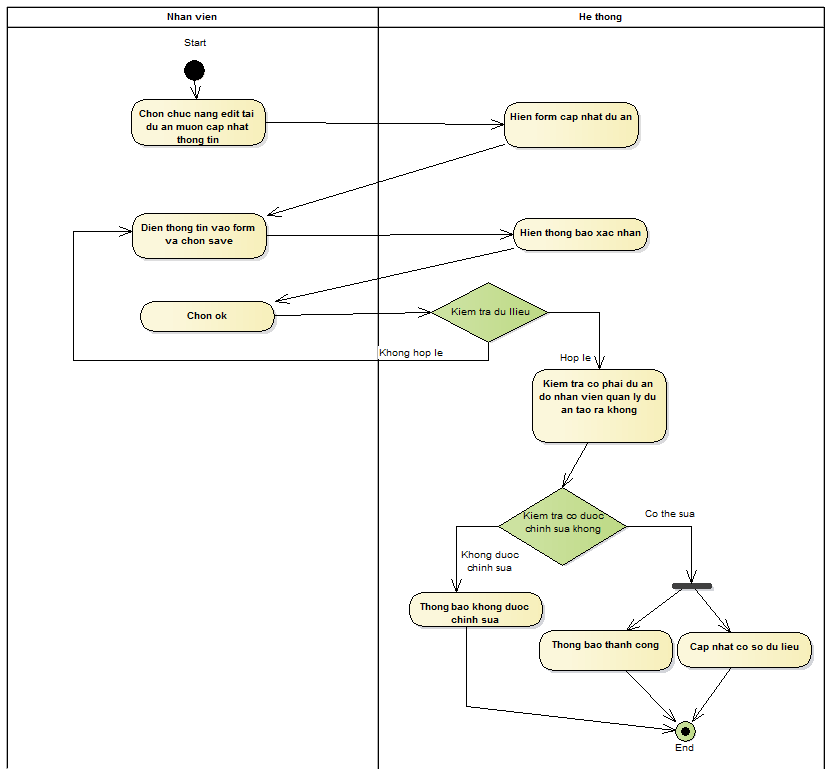
Hình ‑ Sequence diagram UC006a2

### UC006b Cap nhat thong tin kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC006b – Cap nhat thong tin kinh nghiem lam viec | | |
| Mục đích: | Nhân viên cập nhật thông tin về dự án có kinh nghiệm đã làm trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin dự án kinh nghiệm đã làm trong hồ sơ của bản thân và các hồ sơ nhân viên khác.  Nếu dự án kinh nghiệm làm việc do hệ thống tự động thêm bởi sự quản lý của nhân viên quản lý dự án, thì nhân viên không cập nhật được thông tin trong dự án này. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông dự án đã làm trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn chức năng edit tại dự án muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật dự án. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi.   2. Dự án do nhân viên quản lý dự án quản lý sẽ thông báo không có quyền cập nhật. |

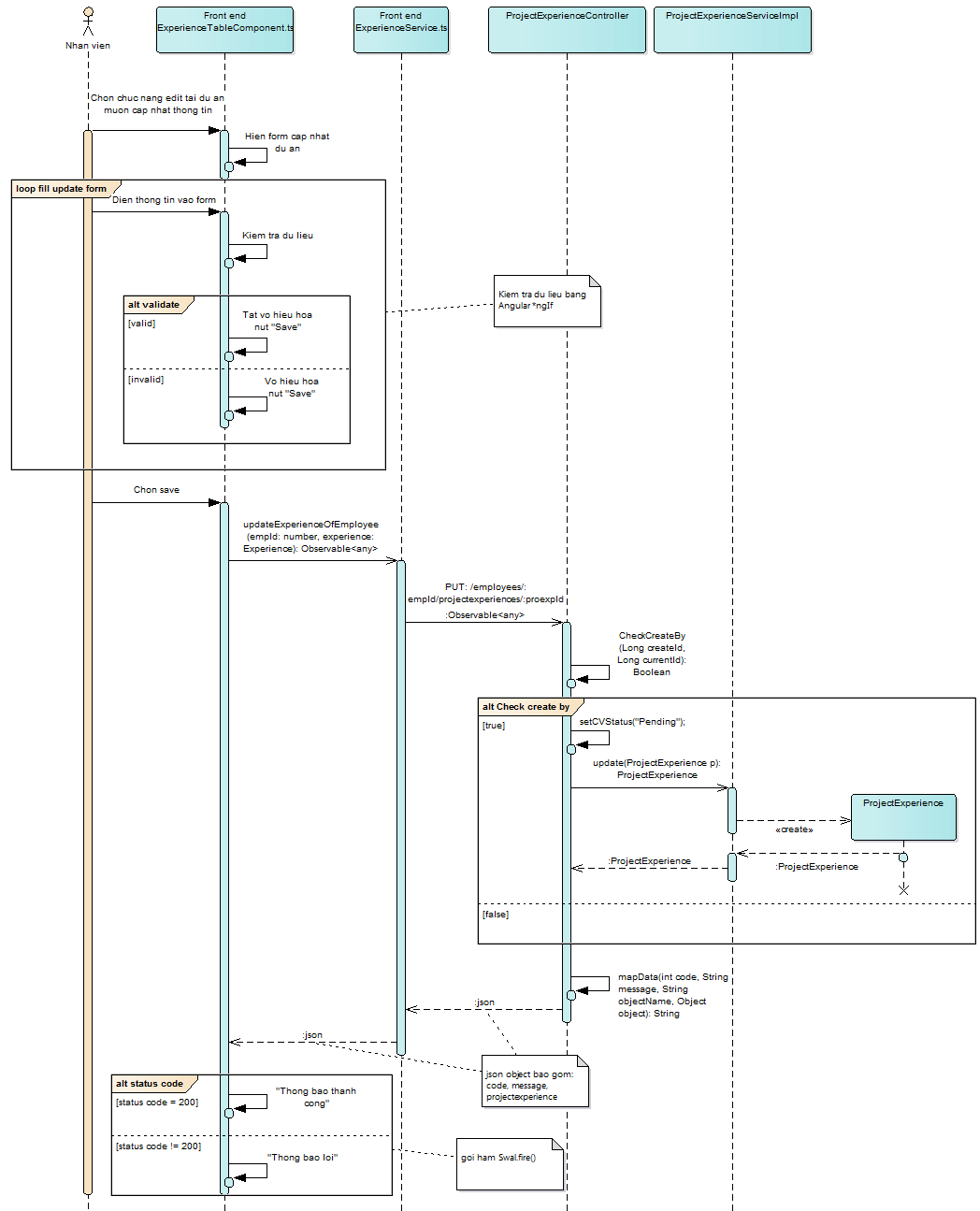
Bảng ‑ Đặc tả UC006b

* Activity diagram:



Hình ‑ Activity diagram UC006b

* Sequence diagram:



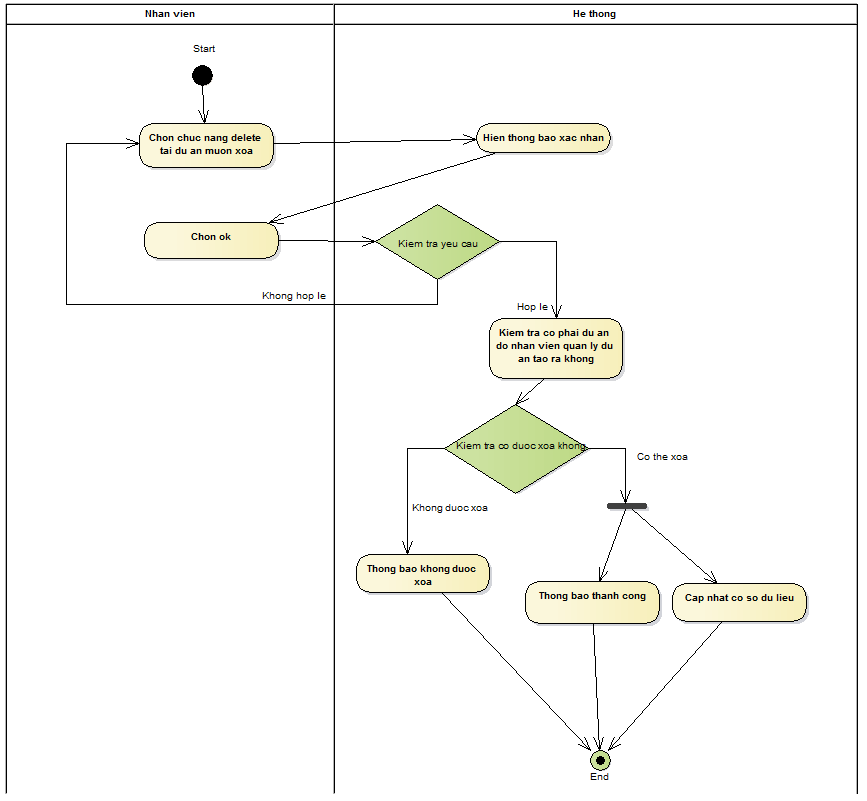
Hình ‑ Sequence diagram UC006b

### UC006c Xoa kinh nghiem lam viec

|  |  |  |
| --- | --- | --- |
| Use case: UC006c – Xoa kinh nghiem lam viec | | |
| Mục đích: | Nhân viên xóa thông tin dự án đã làm trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin vai trò của bản thân và các hồ sơ nhân viên khác.  Nếu dự án kinh nghiệm làm việc do hệ thống tự động thêm bởi sự quản lý của nhân viên quản lý dự án, thì nhân viên không xóa được dự án này. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công dự án đã làm trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên chọn chức năng xóa tại dự án muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi.   2. Dự án do nhân viên quản lý dự án quản lý sẽ thông báo không có quyền xóa. |

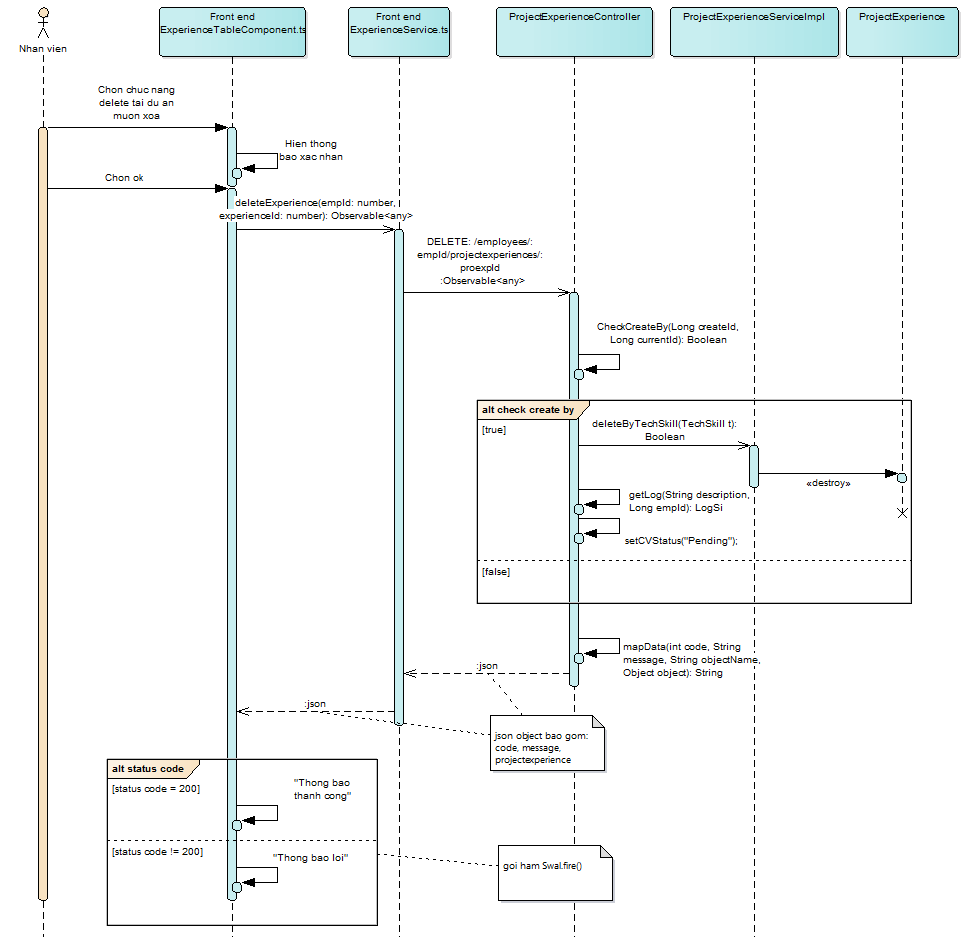
Bảng ‑ Đặc tả UC006c

* Activity diagram:



Hình ‑ Activity diagram UC006c

* Sequence diagram:



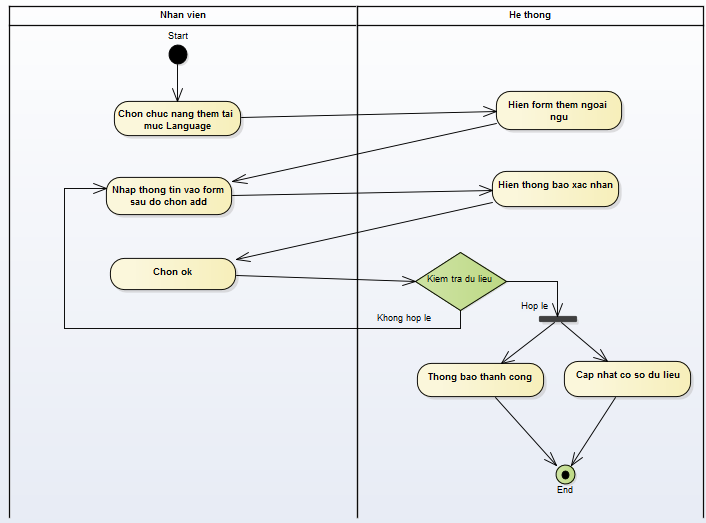
Hình ‑ Sequence diagram UC006c

### UC007 Them ngoai ngu

|  |  |  |
| --- | --- | --- |
| Use case: UC007 – Them ngoai ngu | | |
| Mục đích: | Nhân viên thêm thông tin trình độ ngoại ngữ vào hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể thêm thông tin này vào hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Thêm thành công thông tin trình độ ngoại ngữ vào hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào chức năng thêm tại mục Language. |  |
|  | 1. Hệ thống hiện form thêm language. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add. |  |
|  | 1. Hệ thống hiện thông báo xác nhận thêm hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Thêm không thành công sẽ thông báo lỗi. |

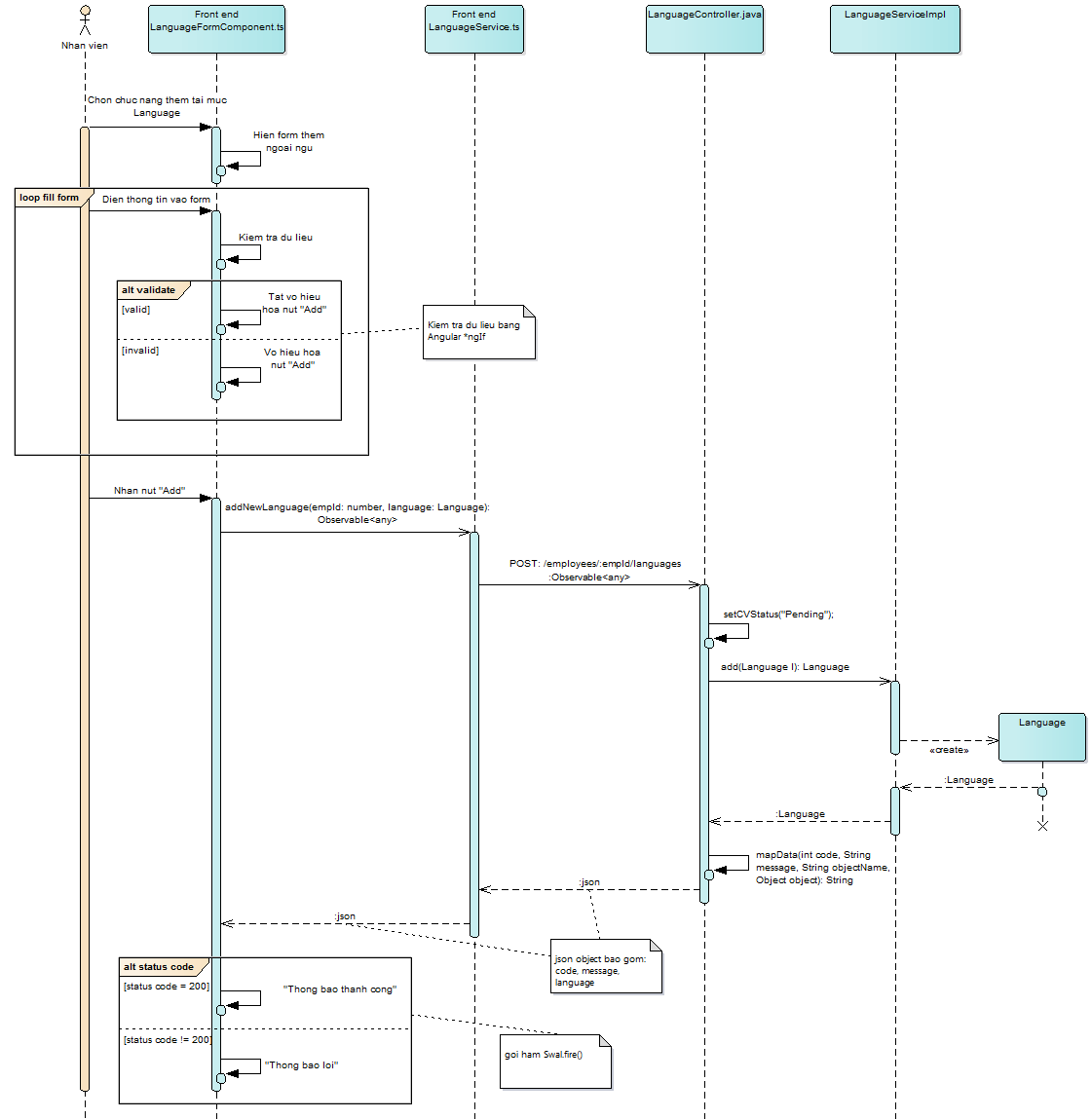
Bảng ‑ Đặc tả UC007

* Activity diagram:



Hình ‑ Activity diagram UC007

* Sequence diagram:



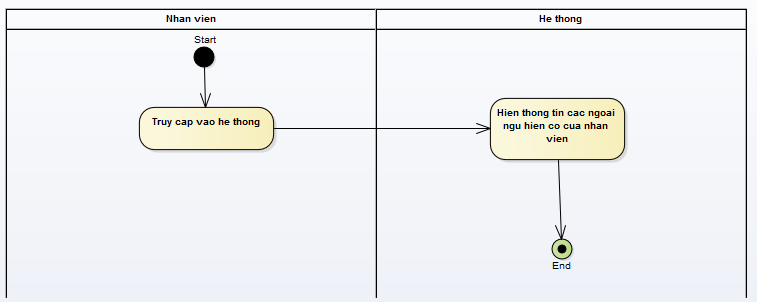
Hình ‑ Sequence diagram UC007

### UC008 Xem thong tin ngoai ngu

|  |  |  |
| --- | --- | --- |
| Use case: UC008 – Xem thong tin ngoai ngu | | |
| Mục đích: | Nhân viên xem chi tiết thông tin về trình độ ngoại ngữ trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể xem thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin các ngoại ngữ của hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiện thông tin các ngoại ngữ hiện có của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

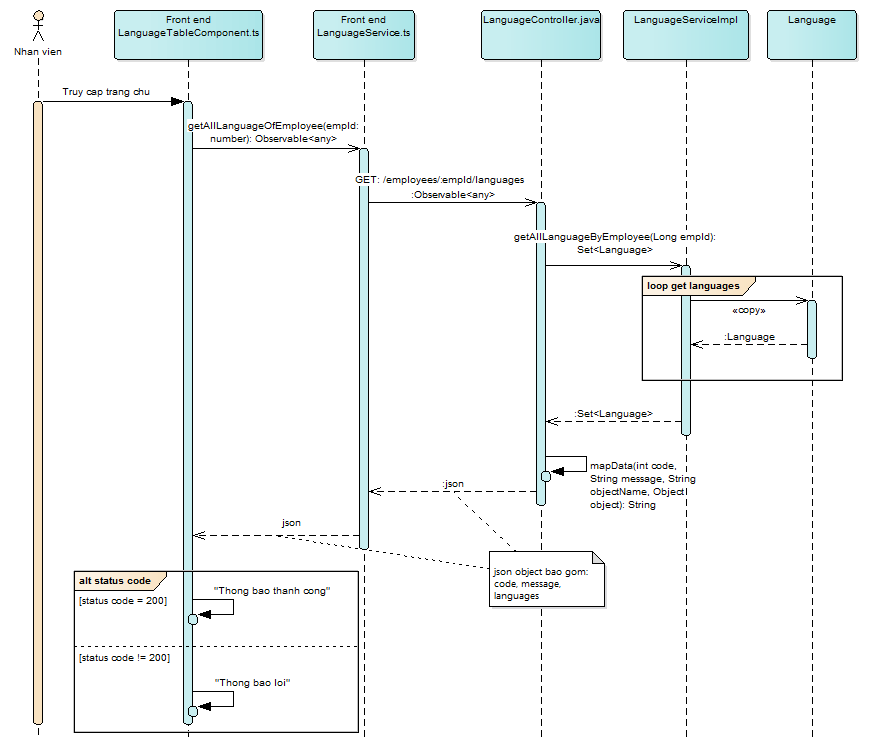
Bảng ‑ Đặc tả UC008

* Activity diagram:



Hình ‑ Activity diagram UC008

* Sequence diagram:



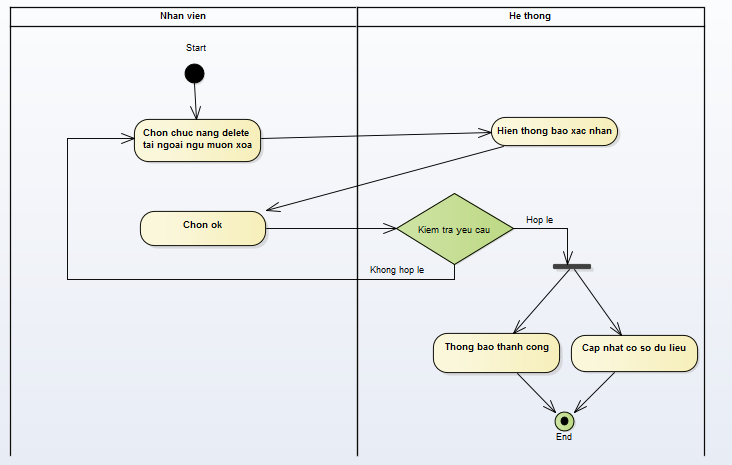
Hình ‑ Sequence diagram UC008

### UC008a Xoa ngoai ngu

|  |  |  |
| --- | --- | --- |
| Use case: UC008a – Xoa ngoai ngu | | |
| Mục đích: | Nhân viên xóa thông tin trình độ ngoại ngữ trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin ngoại ngữ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công ngoại ngữ trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn vào chức năng xóa tại ngoại ngữ muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi. |

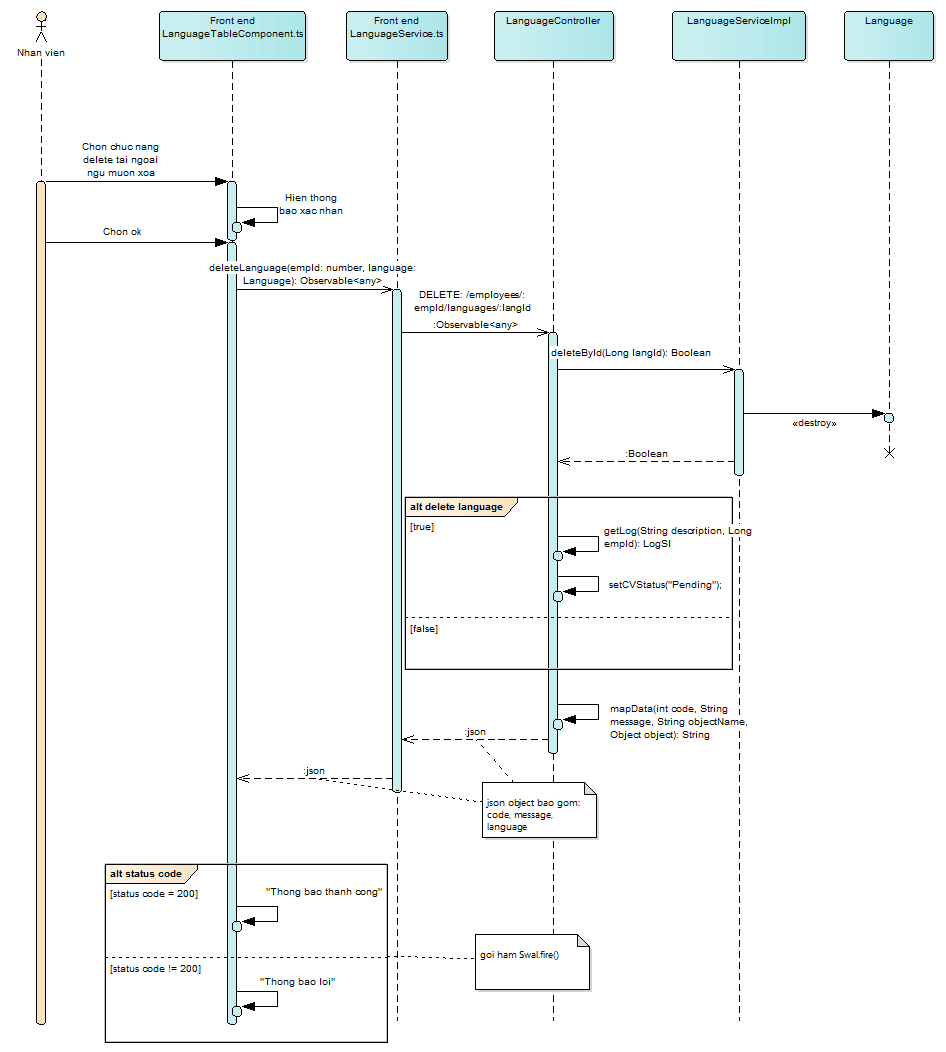
Bảng ‑ Đặc tả UC008a

* Activity diagram:



Hình ‑ Activity diagram UC008a

* Sequence diagram:



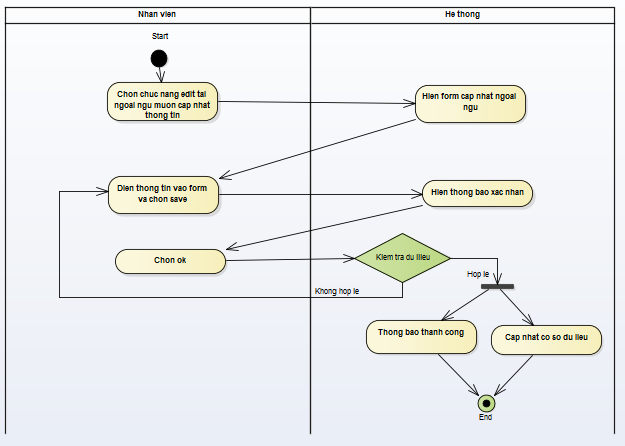
Hình ‑ Sequence diagram UC008a

### UC008b Cap nhat ngoai ngu

|  |  |  |
| --- | --- | --- |
| Use case: UC008b – Cap nhat ngoai ngu | | |
| Mục đích: | Nhân viên cập nhật thông tin trình độ ngoại ngữ trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin trình độ ngoại ngữ trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn chức năng edit tại ngoại ngữ muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật ngoại ngữ. |
| 1. Nhân viên điền thông tin form yêu cầu sau chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

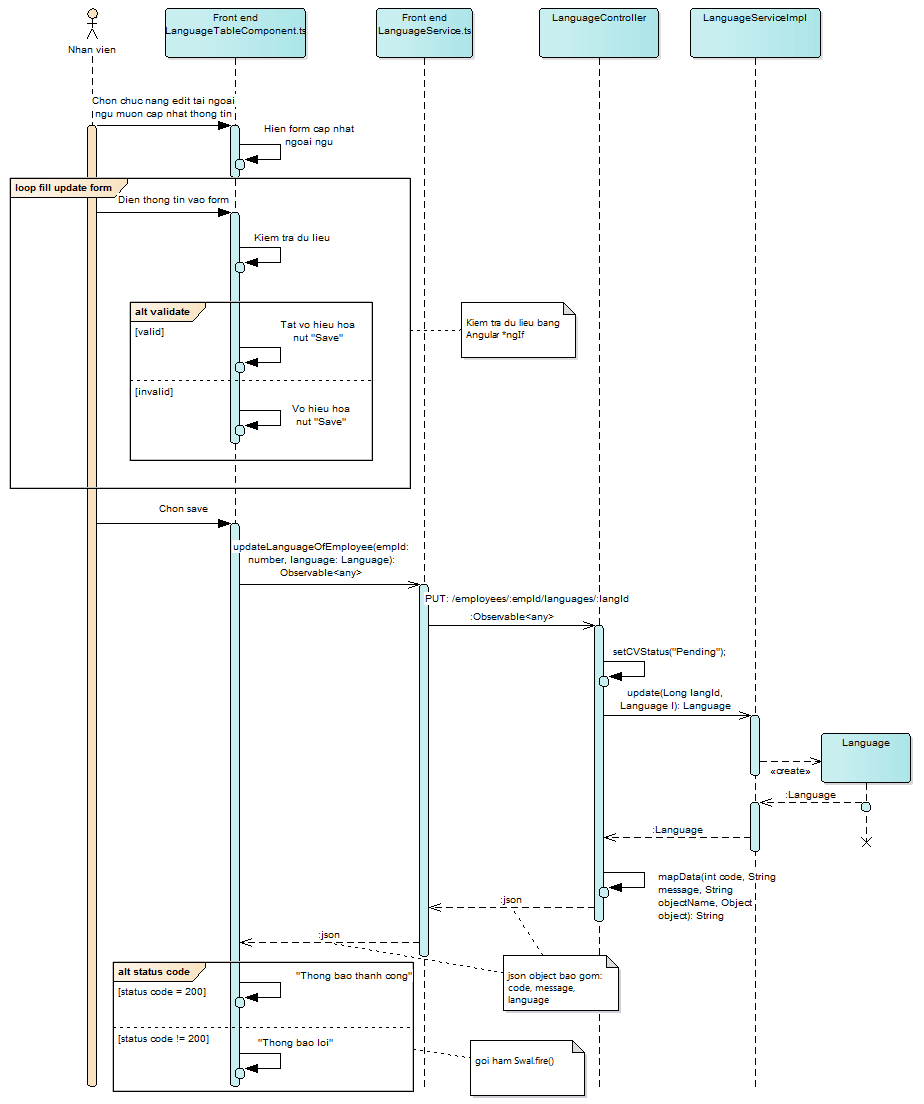
Bảng ‑ Đặc tả UC008b

* Activity diagram:



Hình ‑ Activity diagram UC008b

* Sequence diagram:



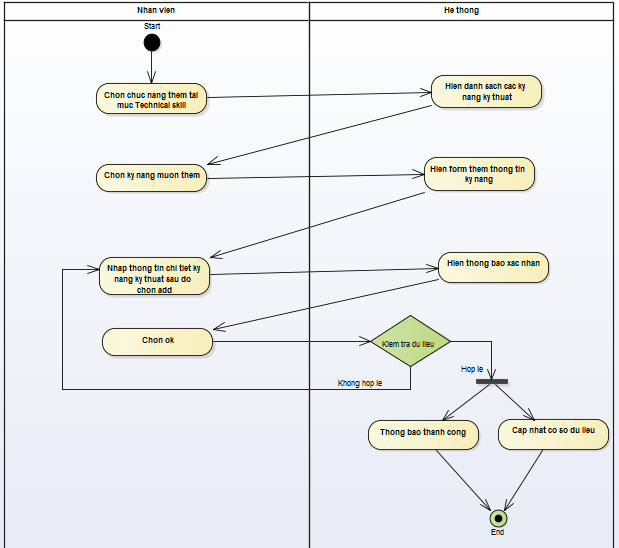
Hình ‑ Sequence diagram UC008b

### UC009 Them ky nang ky thuat

|  |  |  |
| --- | --- | --- |
| Use case: UC009 – Them ky nang ky thuat | | |
| Mục đích: | Nhân viên thêm thông tin trình độ kỹ năng kỹ thuật vào hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể thêm thông tin này vào hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Thêm thành công thông tin kỹ năng kỹ thuật vào hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào chức năng thêm tại mục Technical Skill. |  |
|  | 1. Hệ thống hiện danh sách các kỹ năng kỹ thuật. |
| 1. Nhân viên chọn kiểu kỹ năng, loại kỹ năng và tên kỹ năng cần thêm. |  |
|  | 1. Hệ thống hiện form thêm chi tiết trình độ kỹ năng vừa chọn. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add. |  |
|  | 1. Hệ thống hiện thông báo xác nhận thêm hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Thêm không thành công sẽ thông báo lỗi. |

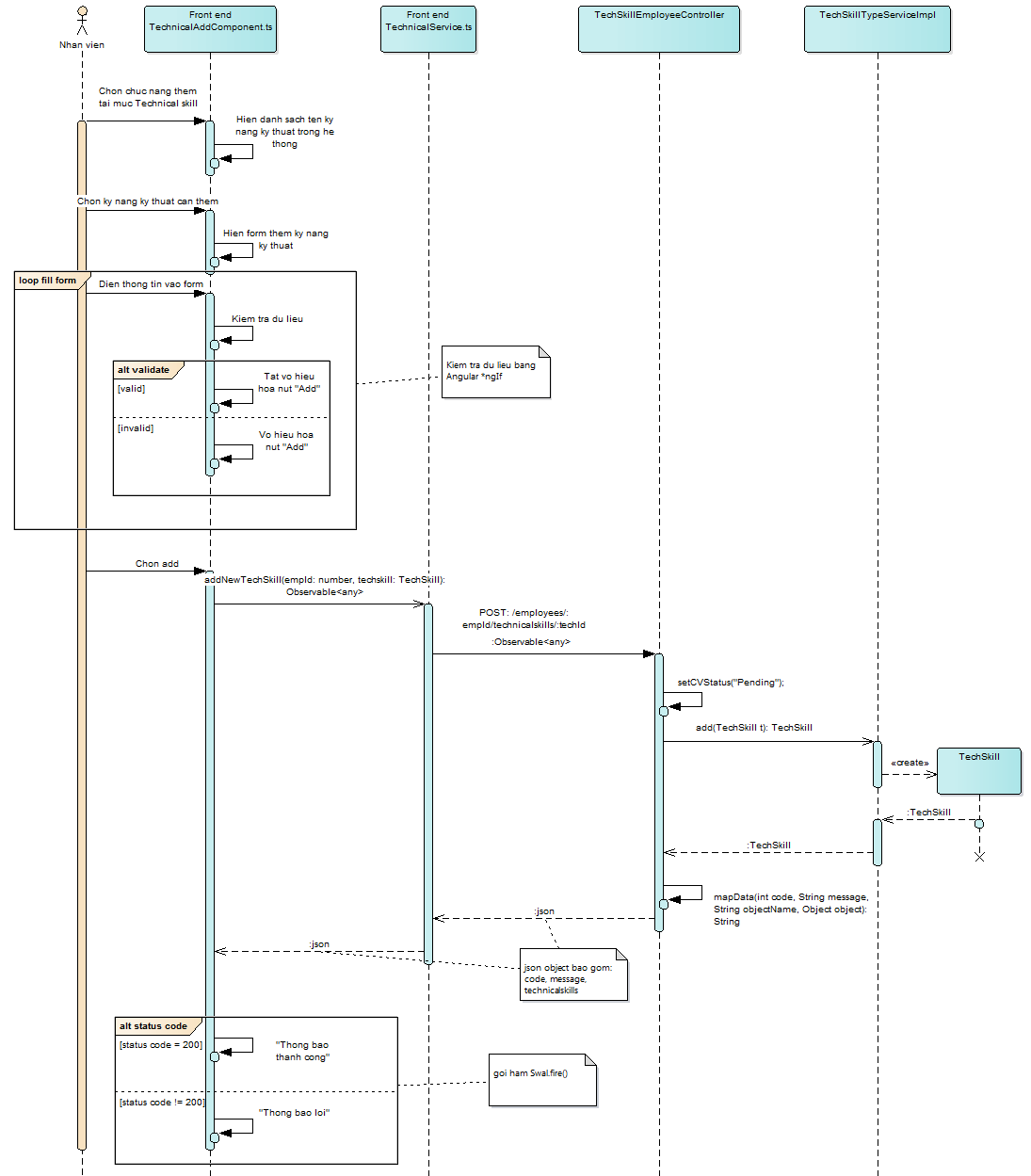
Bảng ‑ Đặc tả UC009

* Activity diagram:



Hình ‑ Activity diagram UC009

* Sequence diagram:



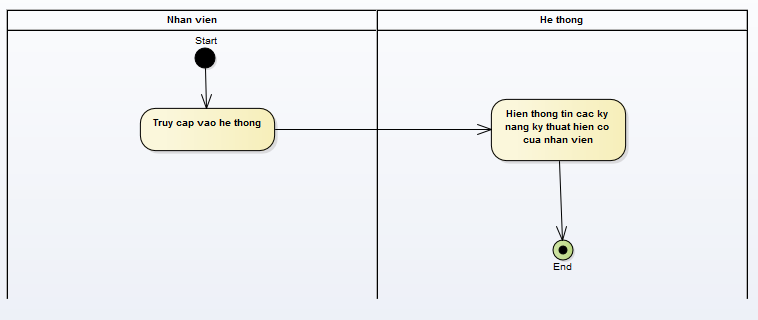
Hình ‑ Sequence diagram UC009

### UC0010 Xem thong tin ky nang ky thuat

|  |  |  |
| --- | --- | --- |
| Use case: UC0010 – Xem thong tin ky nang ky thuat | | |
| Mục đích: | Nhân viên xem chi tiết thông tin về trình độ kỹ năng kỹ thuật trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể xem thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin các kỹ năng kỹ thuật của hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiện thông tin các kỹ năng kỹ thuật hiện có của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

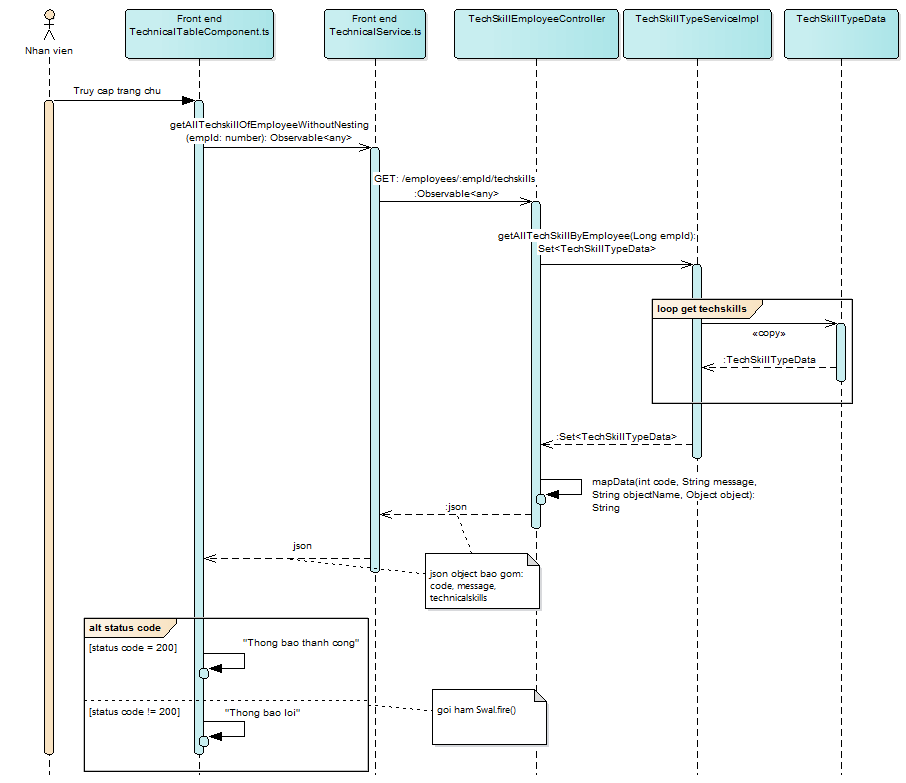
Bảng ‑ Đặc tả UC010

* Activity diagram:



Hình ‑ Activity diagram UC010

* Sequence diagram:



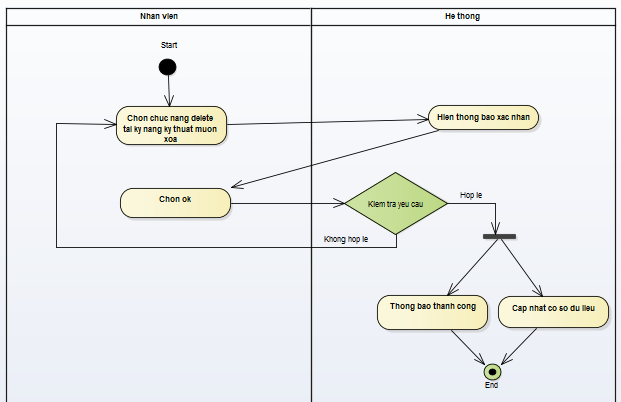
Hình ‑ Sequence diagram UC010

### UC010a Xoa ky nang ky thuat

|  |  |  |
| --- | --- | --- |
| Use case: UC010a – Xoa ky nang ky thuat | | |
| Mục đích: | Nhân viên xóa thông tin trình độ kỹ năng kỹ thuật trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin kỹ năng kỹ thuật của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công kỹ năng kỹ thuật trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn vào chức năng delete tại kỹ năng kỹ thuật muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi. |

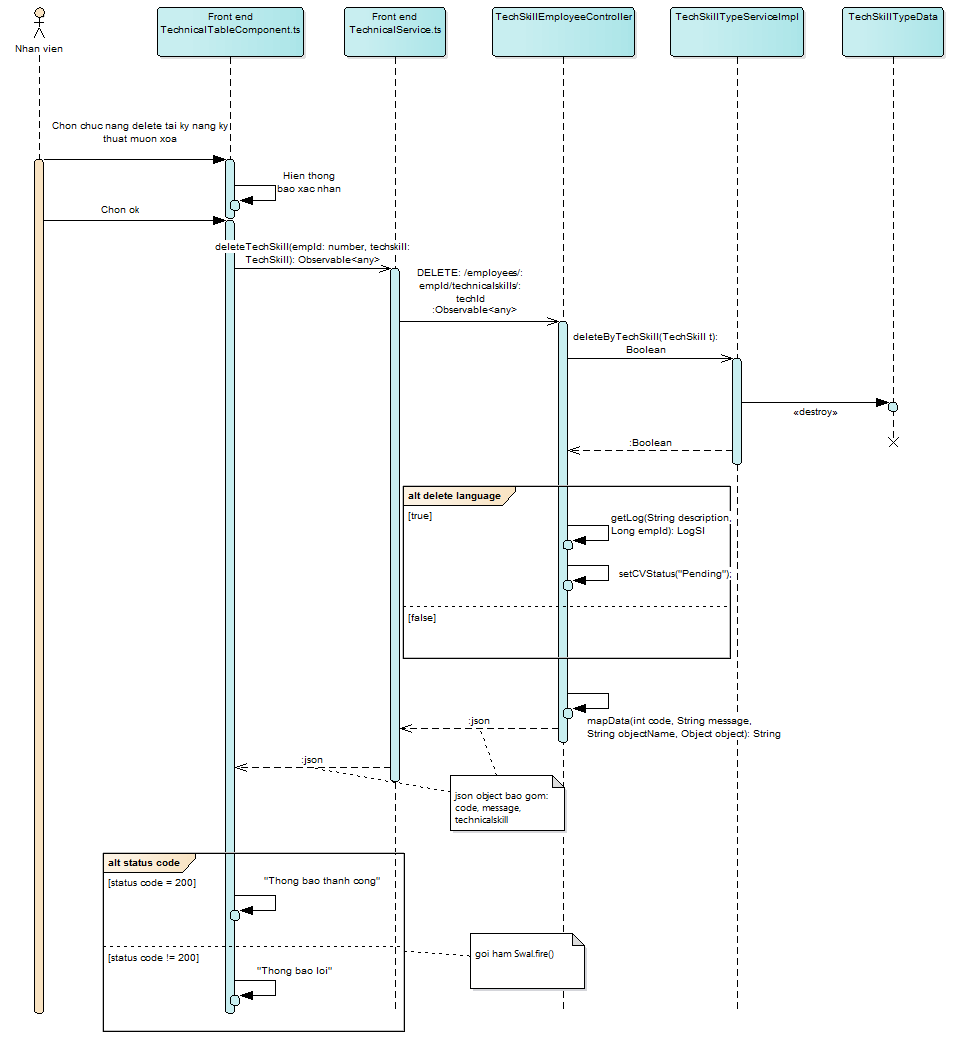
Bảng ‑ Đặc tả UC010a

* Activity diagram:



Hình ‑ Activity diagram UC010a

* Sequence diagram:



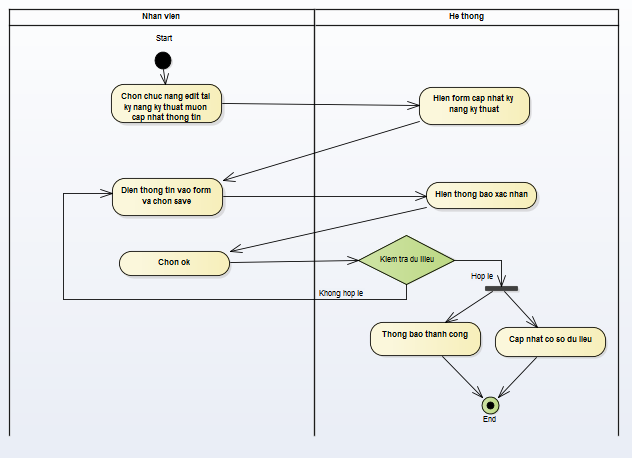
Hình ‑ Sequence diagram UC010a

### UC010b Cap nhat ky nang ky thuat

|  |  |  |
| --- | --- | --- |
| Use case: UC010b – Cap nhat ky nang ky thuat | | |
| Mục đích: | Nhân viên cập nhật thông tin trình độ kỹ năng kỹ thuật trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin trình độ kỹ năng kỹ thuật trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào chức năng edit tại kỹ năng kỹ thuật muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật kỹ năng kỹ thuật . |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

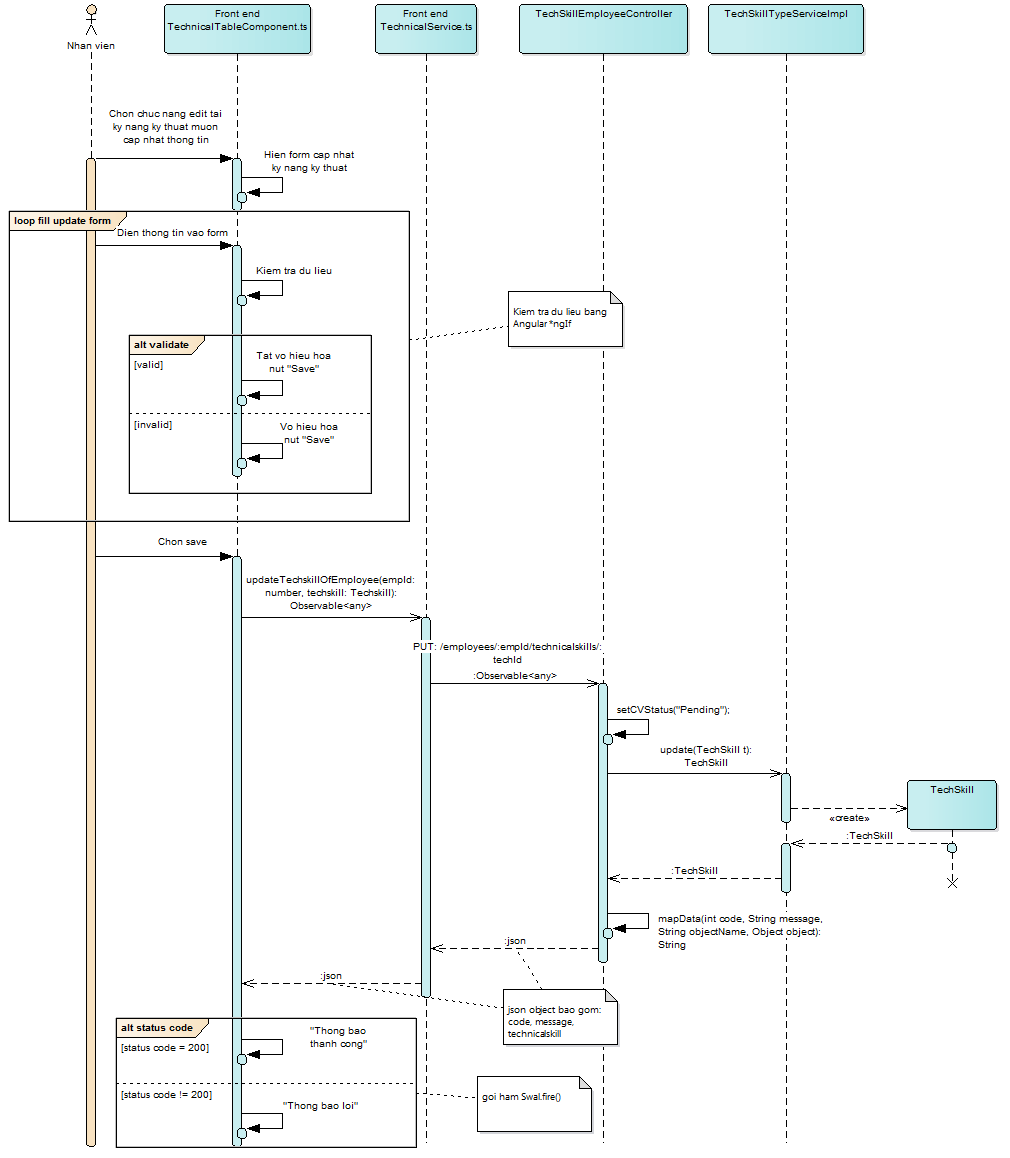
Bảng ‑ Đặc tả UC010b

* Activity diagram:



Hình ‑ Activity diagram UC010b

* Sequence diagram:



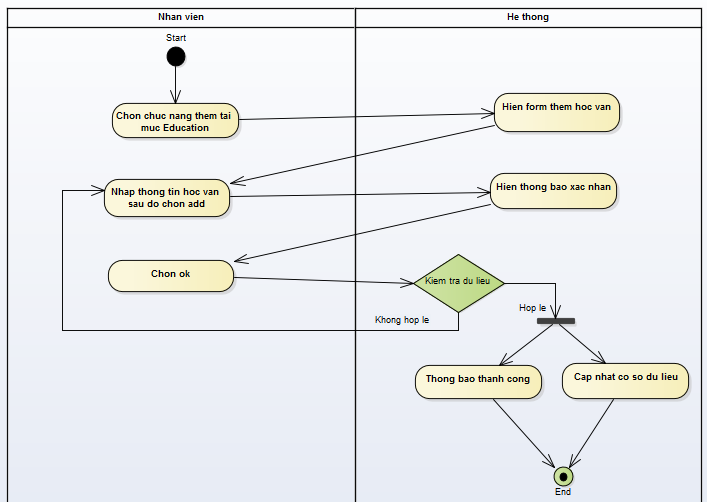
Hình ‑ Sequence diagram UC010b

### UC0011 Them hoc van

|  |  |  |
| --- | --- | --- |
| Use case: UC0011 – Them hoc van | | |
| Mục đích: | Nhân viên thêm thông tin trình độ học vấn vào hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể thêm thông tin này vào hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Thêm thành công thông tin trình độ học vấn vào hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn chức năng thêm tại mục Education. |  |
|  | 1. Hệ thống hiện form thêm học vấn . |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn add. |  |
|  | 1. Hệ thống hiện thông báo xác nhận thêm hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Thêm không thành công sẽ thông báo lỗi. |

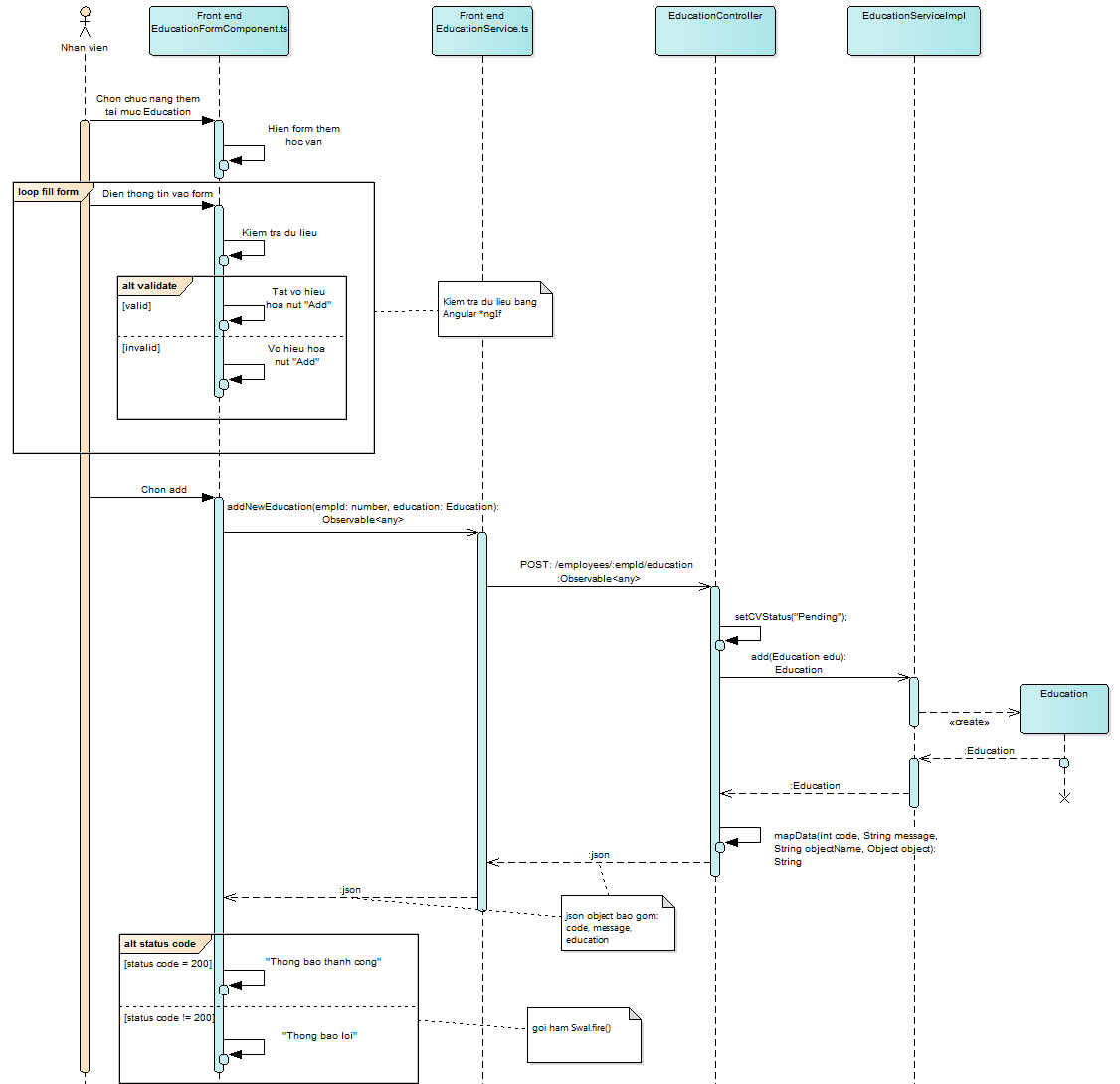
Bảng ‑ Đặc tả UC011

* Activity diagram:



Hình ‑ Activity diagram UC011

* Sequence diagram:



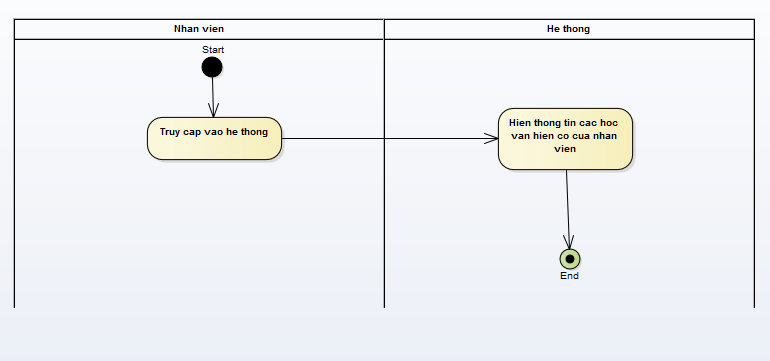
Hình ‑ Sequence diagram UC011

### UC0012 Xem thong tin hoc van

|  |  |  |
| --- | --- | --- |
| Use case: UC0012 – Xem thong tin hoc van | | |
| Mục đích: | Nhân viên xem chi tiết thông tin về trình độ học vấn trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể xem thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xem được thông tin các học vấn của hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
|  | 1. Hệ thống hiện thông tin các học vấn hiện có của nhân viên. |
| Luồng sự kiện phụ (Alternative Flow); |  |  |

Bảng ‑ Đặc tả UC012

* Activity diagram:



Hình ‑ Activity diagram UC012

* Sequence diagram:



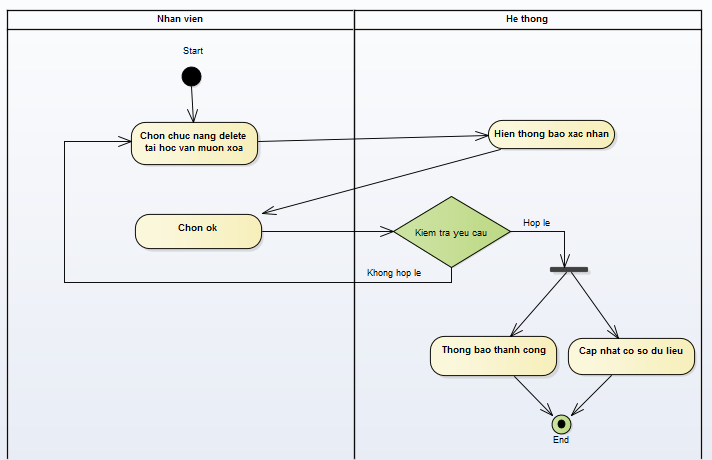
Hình ‑ Sequence diagram UC012

### UC012a Xoa hoc van

|  |  |  |
| --- | --- | --- |
| Use case: UC0012a – Xoa hoc van | | |
| Mục đích: | Nhân viên xóa thông tin trình độ học vấn trong hồ sơ của bản thân.  Nếu quyền là nhân viên nhân sự sẽ có thể xóa thông tin học vấn của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Xóa thành công học vấn trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên ấn vào chức năng delete tại học vấn muốn xóa. |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Xóa không thành công sẽ thông báo lỗi. |

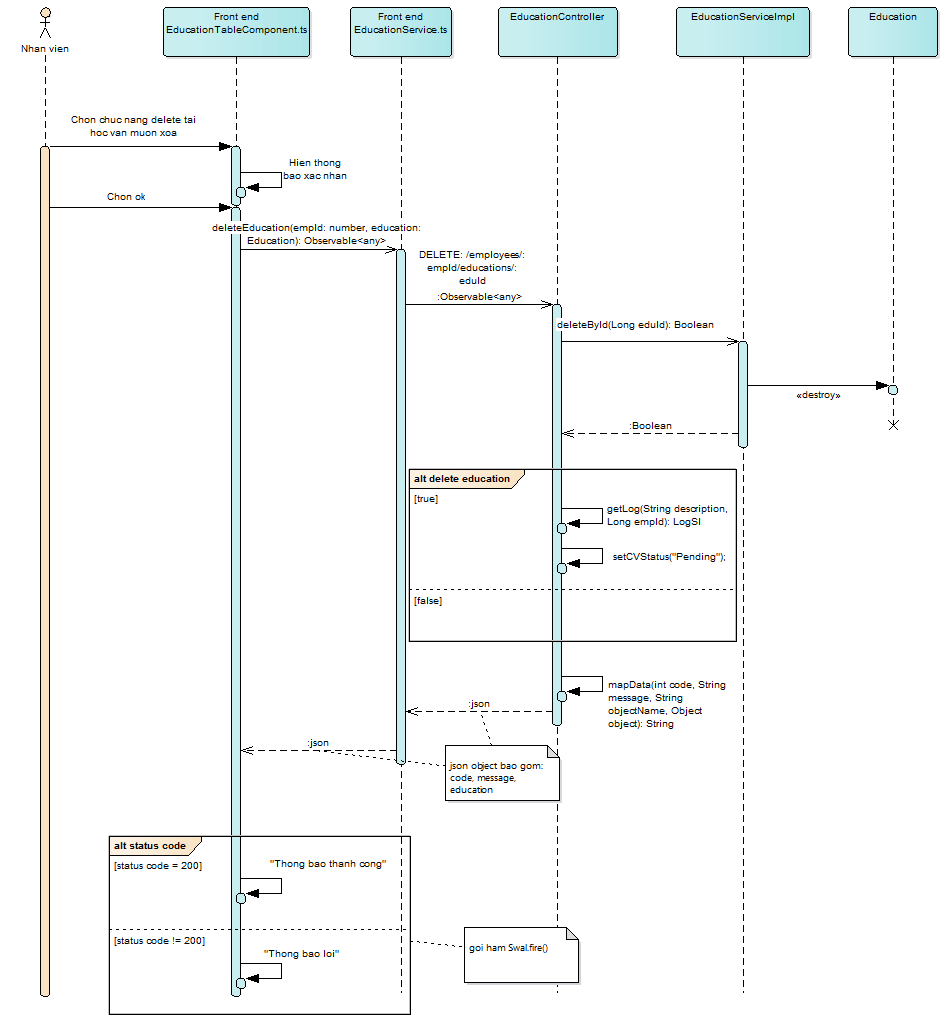
Bảng ‑ Đặc tả UC012a

* Activity diagram:



Hình ‑ Activity diagram UC012a

* Sequence diagram:



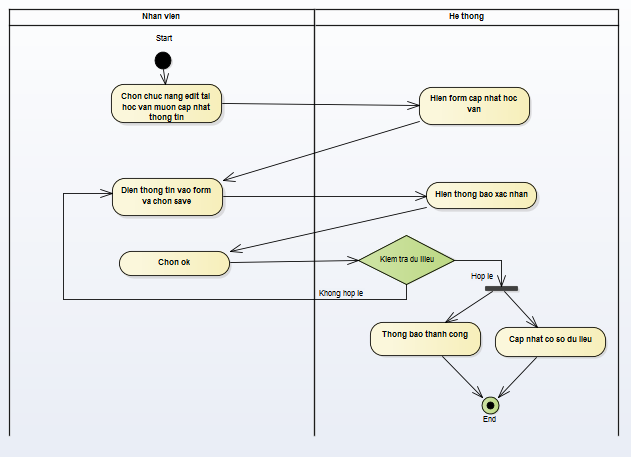
Hình ‑ Sequence diagram UC012a

### UC012b Cap nhat hoc van

|  |  |  |
| --- | --- | --- |
| Use case: UC0012b – Cap nhat hoc van | | |
| Mục đích: | Nhân viên cập nhật thông tin trình độ học vấn trong hồ sơ.  Nếu quyền là nhân viên nhân sự sẽ có thể cập nhật thông tin này trong hồ sơ của bản thân và các hồ sơ nhân viên khác. | |
| Tác nhân: | Nhân viên, nhân viên nhân sự. | |
| Điều kiện trước: | Đã đăng nhập thành công vào hệ thống. | |
| Điều kiện sau: | Cập nhật thành công thông tin trình độ học vấn trong hồ sơ. | |
| Luồng sự kiện chính (Basic flows) | Actor | Hệ thống |
| 1. Nhân viên nhấn vào chức năng edit tại học vấn muốn cập nhật thông tin. |  |
|  | 1. Hệ thống hiện form cập nhật học vấn. |
| 1. Nhân viên điền thông tin form yêu cầu sau đó chọn save. |  |
|  | 1. Hệ thống hiện thông báo xác nhận lưu hay không? |
| 1. Nhân viên chọn ok. |  |
|  | 1. Hệ thống update dữ liệu vào cơ sở dữ liệu và thông báo thành công. |
| Luồng sự kiện phụ (Alternative Flow); |  | * 1. Cập nhật không thành công sẽ thông báo lỗi. |

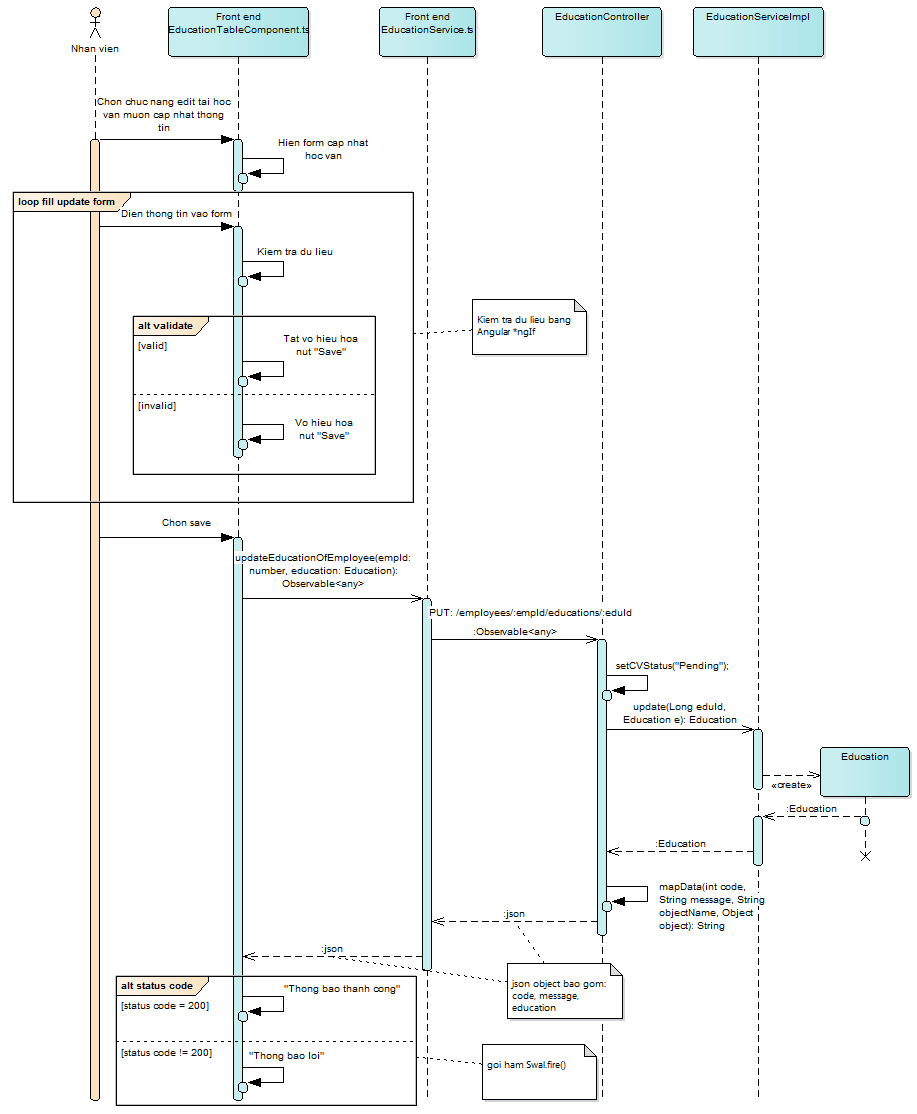
Bảng ‑ Đặc tả UC012b

* Activity diagram:



Hình ‑ Activity diagram UC012b

* Sequence diagram:



Hình ‑ Sequence diagram UC012b

### UC013 Cap nhat trang thai ho so nhan vien

### UC014 Xuat danh sách nhan vien

### UC0015 Quan ly ho so nhan vien

### UC016 Them du an

### UC017 Xem du an cua cong ty

### UC017a Cap nhat thong tin du an

### UC018 Tim kiem

### UC019 Thong ke

### UC020 Xem lich su cap nhat

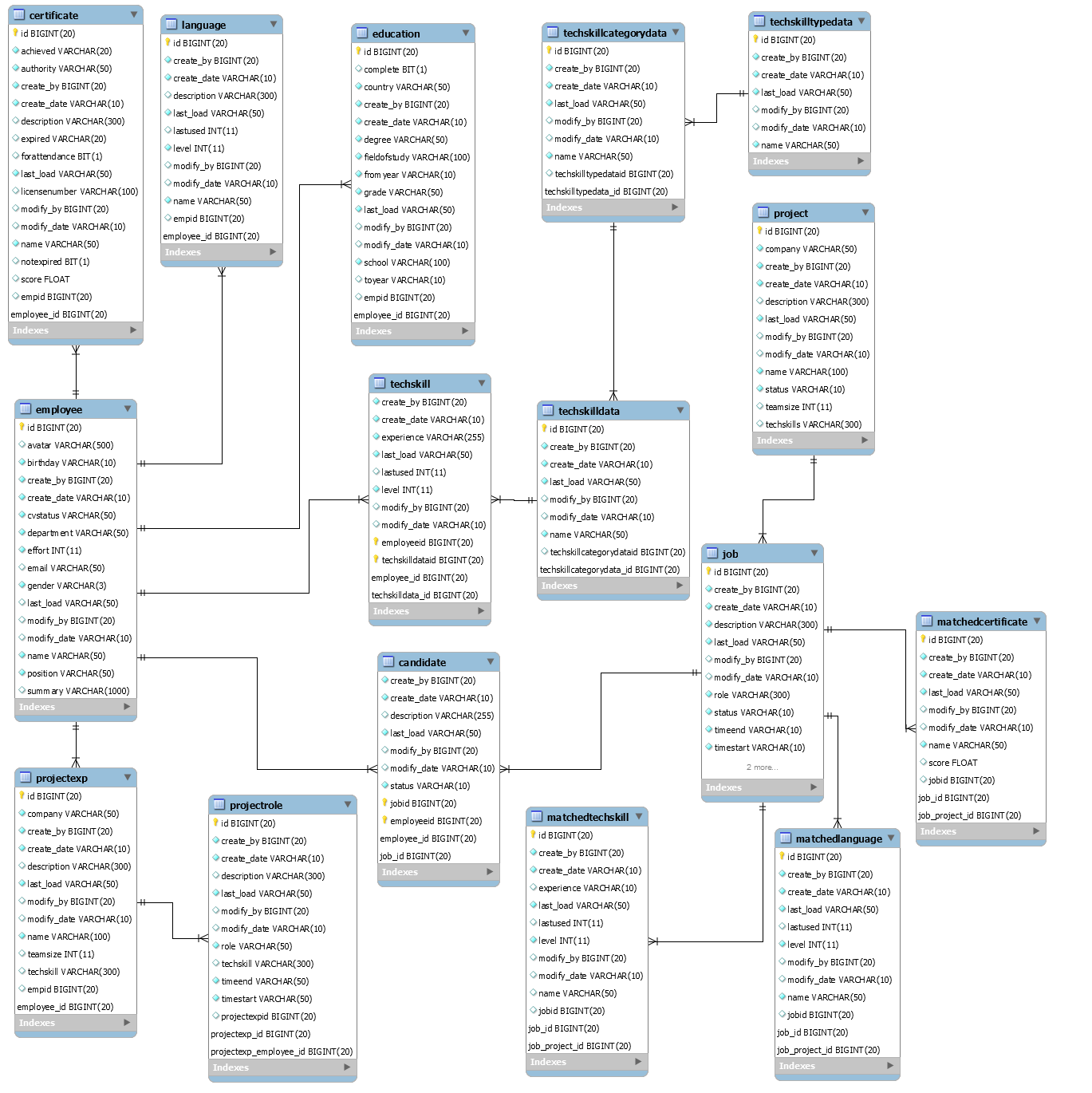
### UC021 Quan ly du lieu hien thi cua he thong

### UC021a Them du lieu

### UC021b Xoa du lieu

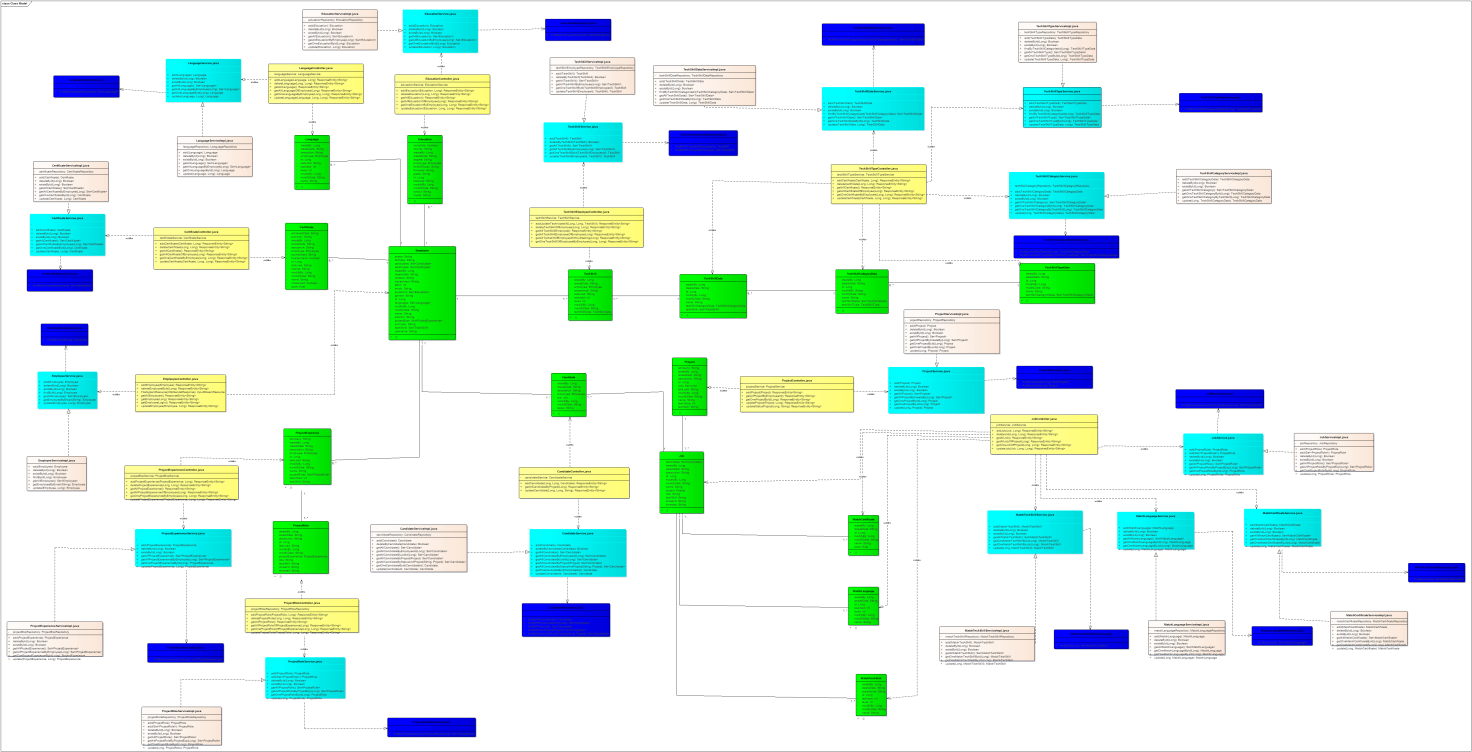
### UC021c Cap nhat du lieu

## Mô hình ERD



Hình ‑ ERD của hệ thống skill inventory

## Class diagram



Hình ‑ Class diagram hệ thống skill incentory

# CHƯƠNG 4: HIỆN THỰC

Chương này sinh viên trình bày quá trình thiết kế, thử nghiệm và hiện thực hệ thống hoặc bài toán.

## Tiêu đề mục

### 4.1.1 Tiêu đề tiểu mục

Nội dung tiểu mục

### Tiêu đề tiểu mục

Nội dung tiểu mục

## Tiêu đề mục

### Tiêu đề tiểu mục

Nội dung tiểu mục

### Tiêu đề tiểu mục

Nội dung tiểu mục

# CHƯƠNG 5: KẾT LUẬN

## Kết quả đạt được

Trình bày các chức năng đã hoàn thành trong đồ án hoặc kết quả nghiên cứu.

## Hạn chế của đồ án

Trình bày những hạn chế, công việc chưa hoàn tất trong đồ án

## Hướng phát triển

Trình bày các định hướng phát triển cho hệ thống hoặc hướng nghiên cứu trong tương lai.

# TÀI LIỆU THAM KHẢO

**Các tài liệu từ internet**

[1].

[2].

[3]. Website <https://stackjava.com/spring/spring-la-gi-gioi-thieu-spring-framework.html>

[4]. Website <https://docs.spring.io/spring-boot/docs/2.1.3.RELEASE/reference/htmlsingle/>

[5].

[6].

[7].

[8].

[9].

[10].

[11].

[12].

[13].

[14].

[15].

**Các tài liệu Tiếng Anh**

[16]. Amy Apon. Lecture for Cluster and Grid Computing. University of Arkansas, 2004.

**Các tài liệu Tiếng Việt**

[17]. Họ và Tên tác giả (Thứ tự theo Tên). Tên sách. Tên nhà xuất bản, năm xuất bản.

# PHỤ LỤC