**Trần Ngọc Quốc XÂY DỰNG HỆ THỐNG QUẢN LÍ, GIÁM SÁT TIẾN ĐỘ CÔNG VIỆC VÀ DỰ ÁN TẠI CÔNG TY 2019**

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG QUẢN LÍ TIẾN ĐỘ CÔNG VIỆC VÀ DỰ ÁN TẠI CÔNG TY**

Người hướng dẫn: **PGS. TS. NGUYỄN TẤN KHÔI**

Sinh viên thực hiện: **TRẦN NGỌC QUỐC**

Số thẻ sinh viên: **102150127**

Lớp**: 15T2**

**Đà Nẵng, 12/2019**

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG QUẢN LÍ TIẾN ĐỘ CÔNG VIỆC VÀ DỰ ÁN TẠI CÔNG TY**

Người hướng dẫn: **PGS. TS. NGUYỄN TẤN KHÔI**

Sinh viên thực hiện: **TRẦN NGỌC QUỐC**

Số thẻ sinh viên: **102150127**

Lớp**: 15T2**

**Đà Nẵng, 12/2019**

**TÓM TẮT**

Tên đề tài: Xây dựng hệ thống quản lí, giám sát tiến độ công việc và dự án tại công ty

Sinh viên thực hiện: Trần Ngọc Quốc

Số thẻ SV: 102150127 Lớp: 15T2

Người hướng dẫn: PGS.TS. Nguyễn Tấn Khôi

Hệ thống bao gồm:

* Website quản lí, giám sát tiến độ công việc và dự án tại công ty

Các chức năng chính:

* Quản lí thông tin nhân viên hiện có trong công ty
* Quản lí dự án của công ty
* Quản lí tiến độ công việc của dự án theo mô hình scrum-agile
* Quản lí thời gian biểu, chấm công của nhân viên

|  |  |
| --- | --- |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc |

**NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP**

**1. Thông tin chung:**

1. Họ và tên sinh viên: Trần Ngọc Quốc

2. Lớp: 15T2 Số thẻ SV: 102150127

3. Tên đề tài: Xây dựng hệ thống quản lí, giám sát tiến độ công việc và dự án tại công ty

4. Người hướng dẫn: Nguyễn Tấn Khôi Học hàm/ học vị: Tiến sĩ

**II. Nhận xét, đánh giá đồ án tốt nghiệp:**

1. Về tính cấp thiết, tính mới, khả năng ứng dụng của đề tài: (điểm tối đa là 2đ)

………………………………………………………………………………………..

………………………………………………………………………………………..

2. Về kết quả giải quyết các nội dung nhiệm vụ yêu cầu của đồ án: (điểm tối đa là 4đ)

………………………………………………………………………………………..

………………………………………………………………………………………..

3. Về hình thức, cấu trúc, bố cục của đồ án tốt nghiệp: (điểm tối đa là 2đ)

………………………………………………………………………………………..

………………………………………………………………………………………..

4. Đề tài có giá trị khoa học/ có bài báo/ giải quyết vấn đề đặt ra của doanh nghiệp hoặc nhà trường: (điểm tối đa là 1đ)

………………………………………………………………………………………..

………………………………………………………………………………………..

5. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:

………………………………………………………………………………………..

………………………………………………………………………………………..

**III. Tinh thần, thái độ làm việc của sinh viên:** (điểm tối đa 1đ)

………………………………………………………………………………………..

**IV. Đánh giá:**

1. Điểm đánh giá: ……../10 (lấy đến 1 số lẻ thập phân)

2. Đề nghị:  Được bảo vệ đồ án  Bổ sung để bảo vệ  Không được bảo vệ

|  |  |  |
| --- | --- | --- |
|  | | *Đà Nẵng, ngày 20 tháng 12 năm 2019* |
|  | | **Người hướng dẫn**  PGS.TS. Nguyễn Tấn Khôi |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc | |

**NHẬN XÉT PHẢN BIỆN ĐỒ ÁN TỐT NGHIỆP**

1. **Thông tin chung:**
2. Họ và tên sinh viên: Trần Ngọc Quốc
3. Lớp: 15T2 Số thẻ SV: 102150127
4. Tên đề tài: Xây dựng hệ thống quản lí, giám sát tiến độ công việc và dự án tại công ty
5. Người phản biện: ..………………………….………… Học hàm/ học vị: ………….
6. **Nhận xét, đánh giá đồ án tốt nghiệp:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Các tiêu chí đánh giá** | **Điểm**  **tối đa** | **Điểm**  **đánh giá** |
| **1** | **Sinh viên có phương pháp nghiên cứu phù hợp, giải quyết đủ nhiệm vụ đồ án được giao** | **80** |  |
| 1a | * Tính mới (nội dung chính của ĐATN có những phần mới so với các ĐATN trước đây). * Đề tài có giá trị khoa học, công nghệ; có thể ứng dụng thực tiễn. | 15 |  |
| 1b | * Kỹ năng giải quyết vấn đề; hiểu, vận dụng được kiến thức cơ bản, cơ sở, chuyên ngành trong vấn đề nghiên cứu. * Chất lượng nội dung ĐATN (thuyết minh, bản vẽ, chương trình, mô hình,…). | 50 |  |
| 1c | * Có kỹ năng vận dụng thành thạo các phần mềm ứng dụng trong vấn đề nghiên cứu; * Có kỹ năng đọc, hiểu tài liệu bằng tiếng nước ngoài ứng dụng trong vấn đề nghiên cứu; * Có kỹ năng làm việc nhóm; | 15 |  |
| **2** | **Kỹ năng viết:** | **20** |  |
| 2a | * Bố cục hợp lý, lập luận rõ ràng, chặt chẽ, lời văn súc tích | 15 |  |
| 2b | * Thuyết minh đồ án không có lỗi chính tả, in ấn, định dạng | 5 |  |
| **3** | **Tổng điểm đánh giá theo thang 100:** |  |  |
|  | **Quy về thang 10 (lấy đến 1 số lẻ)** |  |  |

1. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa: ……………………………………..

……………………………………………………………………………………….....

1. Câu hỏi đề nghị sinh viên trả lời trong buổi bảo vệ: ………………………………

……………………………………………………………………………………….....

……………………………………………………………………………………….....

1. Đề nghị:  Được bảo vệ đồ án  Bổ sung để bảo vệ  Không được bảo vệ

*Đà Nẵng, ngày 30 tháng 12 năm 2019*

**Người phản biện**

|  |  |
| --- | --- |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | **CỘNG HÒA XÃ HÔI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc |

# **NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

Họ tên sinh viên: Trần Ngọc Quốc Số thẻ sinh viên: 102150127

Lớp:15T2 Khoa:CNTT Ngành: CNPM

1. *Tên đề tài đồ án:*

Xây dựng hệ thống quản lí, giám sát tiến độ công việc và dự án tại công ty

1. *Đề tài thuộc diện:*  *Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện*
2. *Các số liệu và dữ liệu ban đầu:*

……………………………………..……………………………………………..……......……………………………………………………………………………………………..………………………………….…..………………………..………………………

1. *Nội dung các phần thuyết minh và tính toán:*

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

1. *Các bản vẽ, đồ thị ( ghi rõ các loại và kích thước bản vẽ ):*

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

…...………………………………………………………………………………………

1. *Họ tên người hướng dẫn:* PGS.TS. Nguyễn Tấn Khôi
2. *Ngày giao nhiệm vụ đồ án:*  *05/09/2019.*
3. *Ngày hoàn thành đồ án: 20/12/2019.*

|  |  |
| --- | --- |
|  | *Đà Nẵng, ngày 20 tháng 12 năm 2019* |
| **Trưởng Bộ môn:** Lê Thị Mỹ Hạnh | **Người hướng dẫn**  PGS.TS. Nguyễn Tấn Khôi |

**LỜI NÓI ĐẦU**

Để đạt được kết quả tốt cho đồ án lần này, chúng em đã nhân được sự giúp đỡ nhiệt tình từ rất nhiều người. Với tình cảm sâu sắc và sự chân thành, chúng em muốn bày tỏ lòng biết ơn đối với tất cả. những cá nhân, cơ quan đã giúp đỡ chúng em trong quá trình học tập và nghiên cứu.

Đầu tiên chúng em xin chân thành cảm ơn các thầy cô của trường Đại học Bách Khoa Đà Nẵng nói chung và khoa Công Nghệ Thông Tin nói riêng đã truyền đạt những kiến thức quý báu thông qua các giờ học trên giảng đường trong suốt những năm đại học, tạo tiền đề cho chúng em thực hiện tốt đồ án tốt nghiệp lần này.

Đặc biệt, chúng em muốn gởi lời cảm ơn sâu sắc thầy Nguyễn Tấn Khôi – giảng viên ngành Công Nghệ Thông Tin, đã nhiệt tình hướng dẫn, chia sẻ các kinh nghiệm quý báu, và tạo điều kiện về nhiều mặt để chúng em có thể hoàn thành tốt được đồ án lần này.

Chúng em cũng xin chân thành cảm ơn gia đình, bạn bè và nhiều người xung quanh cũng tạo điều kiện, hỗ trợ về vật chất và tinh thần trong thời gian làm đồ án.

Với kinh nghiệm giới hạn của sinh viên, bài báo cáo có thể sẽ xảy ra những sai sót không muốn, em rất mong sự góp ý của quý thầy cô và mọi người để em rút kinh nghiệm.

Em xin chân thành cảm ơn.

*Đà Nẵng, ngày 20 tháng 12 năm 2019*

Sinh viên thực hiện

**Trần Ngọc Quốc**

**CAM ĐOAN**

Chúng em xin cam đoan:

1. Nội dung trong đồ án này là do em thực hiện dưới sự hướng dẫn trực tiếp của thầy Nguyễn Tấn Khôi.

2. Các tham khảo dùng trong đồ án đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.

3. Mọi hình ảnh, video demo trong báo cáo đều do bản thân nhóm tự làm, không sao chép toàn bộ từ bất cứ nguồn nào. Nếu có những sao chép không hợp lệ, vi phạm, em xin chịu hoàn toàn trách nhiệm.

*Đà Nẵng, ngày 20 tháng 12 năm 2019*

Sinh viên thực hiện

**Trần Ngọc Quốc**

**MỤC LỤC**

[**TÓM TẮT**](#_Toc27573930)

[**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**](#_Toc27573931)

[**LỜI NÓI ĐẦU**](#_Toc27573932)

[**CAM ĐOAN**](#_Toc27573933)

[**MỤC LỤC**](#_Toc27573934)

[**DANH SÁCH CÁC HÌNH VẼ**](#_Toc27573935)

[**DANH SÁCH CÁC BẢNG**](#_Toc27573936)

[**DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT** 1](#_Toc27573937)

[**MỞ ĐẦU** 2](#_Toc27573938)

[**1.** **Tổng quan về đề tài** 2](#_Toc27573939)

[**2.** **Mục đích và nhiệm vụ chính của đề tài** 3](#_Toc27573940)

[*2.1.* *Mục đích đề tài:* 3](#_Toc27573941)

[*2.2.* *Nhiệm vụ chính của đề tài:* 3](#_Toc27573942)

[**3.** **Công nghệ sử dụng** 3](#_Toc27573943)

[**4.** **Công cụ hỗ trợ** 3](#_Toc27573944)

[**5.** **Cấu trúc đồ án** 3](#_Toc27573945)

[**CHƯƠNG 1.** **CƠ SỞ LÝ THUYẾT** 5](#_Toc27573946)

[**1.1.** **React Js** 5](#_Toc27573947)

[*1.1.1.* *Giới thiệu* 5](#_Toc27573948)

[*1.1.2.* *Virtual DOM* 5](#_Toc27573949)

[*1.1.3.* *One-way data binding* 6](#_Toc27573950)

[*1.1.4.* *Giới thiệu về JSX* 6](#_Toc27573951)

[*1.1.5.* *Giới thiệu về Components* 6](#_Toc27573952)

[*1.1.6.* *Props và State* 7](#_Toc27573953)

[*1.1.7.* *Lifecycle* 7](#_Toc27573954)

[**1.2.** **Redux-Saga** 8](#_Toc27573955)

[*1.2.1.* *Giới thiệu* 8](#_Toc27573956)

[*1.2.2.* *Side effect* 8](#_Toc27573957)

[*1.2.3.* *Generator function* 8](#_Toc27573958)

[*1.2.4.* *Cách hoạt động* 8](#_Toc27573959)

[**1.3.** **WEB API** 9](#_Toc27573960)

[*1.3.1.* *ASP.NET CORE* 9](#_Toc27573961)

[*1.3.2.* *Entity Framework* 9](#_Toc27573962)

[*1.3.3.* *Phương thức HTTP* 10](#_Toc27573963)

[**1.4.** **Docker** 10](#_Toc27573964)

[*1.4.1.* *Khái niệm* 10](#_Toc27573965)

[*1.4.2.* *Lợi ích của Docker* 11](#_Toc27573966)

[**CHƯƠNG 2.** **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG** 12](#_Toc27573967)

[**2.1.** **Cách chức năng chính** 12](#_Toc27573968)

[*2.1.1.* *Quản lí thông tin nhân viên* 12](#_Toc27573969)

[*2.1.2.* *Quản lí dự án* 13](#_Toc27573970)

[*2.1.3.* *Quản lí công việc* 14](#_Toc27573971)

[*2.1.4.* *Quản lí thời gian làm việc* 15](#_Toc27573972)

[**2.2.** **Sơ đồ hoạt động** 17](#_Toc27573973)

[**2.3.** **Luồng dữ liệu** 21](#_Toc27573974)

[**2.4.** **Sơ đồ quan hệ** 22](#_Toc27573975)

[**2.5.** **Thiết kế cơ sở dữ liệu** 23](#_Toc27573976)

[*2.5.1.* *Sơ đồ quan hệ, thực thể dữ liệu* 23](#_Toc27573977)

[*2.5.2.* *Đặc tả các bảng dữ liệu* 24](#_Toc27573978)

[**CHƯƠNG 3.** **TRIỂN KHAI VÀ KẾT QUẢ** 32](#_Toc27573979)

[**3.1.**  **Môi trường triển khai** 32](#_Toc27573980)

[**3.2.**  **Kết quả thực hiện** 33](#_Toc27573981)

[**KẾT LUẬN** 40](#_Toc27573982)

[**TÀI LIỆU THAM KHẢO** 41](#_Toc27573983)

**DANH SÁCH CÁC HÌNH VẼ**

[*Hình 1.1: Đặc trưng Virtual DOM 5*](#_Toc27570434)

[*Hình 1.2 Các luồng dữ liệu và sự kiện trong React 6*](#_Toc27570435)

[*Hình 1.3 Các luồng dữ liệu và sự kiện trong React 7*](#_Toc27570436)

[*Hình 1.4 Mô tả cách thức hoạt động của Redux - Saga 9*](#_Toc27570437)

[*Hình 1.5 Mô tả cách thức hoạt động của Entity Framework 10*](#_Toc27570438)

[*Hình 1.6 Mô tả cách thức hoạt động của Docker 11*](#_Toc27570439)

[*Hình 2.1 Sơ đồ use-case cho module quản lí nhân viên 12*](#_Toc27570440)

[*Hình 2.2 Sơ đồ use-case cho module quản lí dự án 13*](#_Toc27570441)

[*Hình 2.3 Sơ đồ use-case cho module quản lí tiến độ công việc trong dự án 14*](#_Toc27570442)

[*Hình 2.4 Sơ đồ use-case module quản lí thời gian làm việc 15*](#_Toc27570443)

[*Hình 2.5 Sơ đồ hoạt động của module quản lí nhân viên 17*](#_Toc27570444)

[*Hình 2.6 Sơ đồ hoạt động của module quản lí dự án 18*](#_Toc27570445)

[*Hình 2.7 Sơ đồ hoạt động của module quản lí tiến độ công việc 19*](#_Toc27570446)

[*Hình 2.8 Sơ đồ hoạt động của module quản lí thời gian biểu, chấm công 20*](#_Toc27570447)

[*Hình 2.9 Mô tả luồng đi của dữ liệu 21*](#_Toc27570448)

[*Hình 2.10 Sơ đồ lớp cho dự án 22*](#_Toc27570449)

[*Hình 2.11 Sơ đồ quan hệ, thực thể dữ liệu 23*](#_Toc27570450)

[*Hình 3.1: Màn hình Login 33*](#_Toc27570451)

[*Hình 3.2 Quản lí nhân viên 33*](#_Toc27570452)

[*Hình 3.3 Xem thông tin cá nhân của nhân viên 34*](#_Toc27570453)

[*Hình 3.4 Chỉnh sửa thông tin cá nhân 34*](#_Toc27570454)

[*Hình 3.5 Lọc và tìm kiếm nhân viên dựa trên kĩ năng, vị trí, tên, … 35*](#_Toc27570455)

[*Hình 3.6 Màn hình quản lí các dự án 35*](#_Toc27570456)

[*Hình 3.7 Tạo một dự án mới 36*](#_Toc27570457)

[*Hình 3.8 Thông tin chi tiết của một công việc 36*](#_Toc27570458)

[*Hình 3.9 Tạo một công việc mới 37*](#_Toc27570459)

[*Hình 3.10 Giao việc cho thành viên trong dự án 37*](#_Toc27570460)

[*Hình 3.11 Thêm nhãn cho công việc để dễ quản lí 38*](#_Toc27570461)

[*Hình 3.12 Xóa một công việc 38*](#_Toc27570462)

[*Hình 3.13 Các tùy chọn cho một bản 38*](#_Toc27570463)

[*Hình 3.14 Quản lí thời gian 39*](#_Toc27570464)

[*Hình 3.15 Ghi lại thời gian đã dành cho một công việc 39*](#_Toc27570465)

**DANH SÁCH CÁC BẢNG**

[*Bảng 2.1 Bảng User 24*](#_Toc27569623)

[*Bảng 2.3 Bảng Position 25*](#_Toc27569624)

[*Bảng 2.4 Bảng Skill 25*](#_Toc27569625)

[*Bảng 2.5 Bảng Skill 25*](#_Toc27569626)

[*Bảng 2.6 Bảng Role 26*](#_Toc27569627)

[*Bảng 2.7 Bảng UserRole 26*](#_Toc27569628)

[*Bảng 2.8 Bảng TimeLineEvent 26*](#_Toc27569629)

[*Bảng 2.9 Bảng Board 27*](#_Toc27569630)

[*Bảng 2.10 Bảng Phase 27*](#_Toc27569631)

[*Bảng 2.11 Bảng BoardUser 28*](#_Toc27569632)

[*Bảng 2.12 Bảng Task 28*](#_Toc27569633)

[*Bảng 2.13 Bảng Attachment 29*](#_Toc27569634)

[*Bảng 2.14 Bảng Todo 29*](#_Toc27569635)

[*Bảng 2.15 Bảng Label 29*](#_Toc27569636)

[*Bảng 2.16 Bảng TaskLabel 30*](#_Toc27569637)

[*Bảng 2.17 Bảng WorkLog 30*](#_Toc27569638)

[*Bảng 2.18 Bảng Comment 30*](#_Toc27569639)

[*Bảng 2.19 Bảng TaskAssignee 31*](#_Toc27569640)

[*Bảng 2.20 Bảng TaskAction 31*](#_Toc27569641)

**DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT**

|  |  |
| --- | --- |
| API | Application Programming Interface |
| JS | Javascript |
| SQL | Structured Query Language |
| UI | User Interface |
| DOM | Document Object Model |
| ASP | Active Server Pages |
| OS | Operating System |
| HTTP | HyperText Transer Protocol |
| WWW | World Wide Web |
| App | Application |
| URL | Uniform Resource Locator |
| ID | Identifier |
| .NET | Dotnet |
| ORM | Object Relational Mapper |
| ES | ECMA Script |
| AJAX | Asynchromous Javascript and XML |
| IoT | Internet of Things |
| TCP | Transmission Control Protocol |
| IP | Internet Protocol |

**MỞ ĐẦU**

## **Tổng quan về đề tài**

Bài toán của các nhà quản trị là làm sao kiểm soát những công việc và dự án đang diễn ra, tối ưu nguồn lực có hạn và nâng cao năng suất doanh nghiệp. Để giải được bài toán này, các phần mềm quản lý công việc giải quyết 2 việc:

* Thứ nhất, minh bạch hóa quá trình giao việc - nhận việc giữa nhà quản lý và nhân viên, từ đó minh bạch hóa trách nhiệm;
* Thứ hai, giúp nhà quản lý lên kế hoạch và giúp nhân viên cộng tác, làm việc tập trung trên một nền tảng duy nhất.

Tóm lại, lợi ích của việc sử dụng một phần mềm quản lý công việc có thể quan sát ở 2 góc độ. Đối với nhân viên, phần mềm giúp họ:

* Nhìn thấy được tất cả công việc của mình, không sót việc.
* Biết đâu là những việc cần ưu tiên.
* Tính toán và sắp xếp được thời gian làm việc hiệu quả.
* Cộng tác với đồng đội để hoàn thành công việc với chất lượng tốt nhất.

Đối với nhà quản lý thì một phần mềm quản lý công việc đem lại những lợi ích như sau:

* Thấy được tổng quan công việc và dự án của tất cả bộ phận
* Ra quyết định và xử lý các vấn đề phát sinh một cách kịp thời
* Sắp xếp, phân bổ nguồn lực (nhân sự + thời gian) một cách hiệu quả
* Tất cả công việc, dự án đều được hoàn thành đúng thời hạn với chất lượng tốt nhất

Hiện nay, trên thị trường có khá nhiều cái tên nổi bật như Trello, Asana, Wrike, Jira,… Tuy nhiên, phần lớn những ứng dụng trên đều tồn tại một số vấn đề bất cập như:

* Không phù hợp với qui trình làm việc của công ty
* Khuyết thiếu một số chức năng cần thiết
* Chi phí cao.

Do đó, cần thiết xây dựng một hệ thống để quản lí, giám sát tiến độ công việc, dự án phù hợp cho quy trình làm việc của công ty, dễ tùy chỉnh, đồng thời tối thiếu chi phí phát sinh cho công ty.

## **Mục đích và nhiệm vụ chính của đề tài**

### ***Mục đích đề tài:***

Xây dựng một hệ thống giúp cho công ty có thể dễ dàng thực hiện các công việc sau:

* Tìm kiếm và truy xuất thông tin nhân viên của công ty.
* Quản lí, giao việc và giám sát tiến độ công việc cho các dự án.
* Quản lí thời gian làm việc của nhân viên.

### ***Nhiệm vụ chính của đề tài:***

Đề tài có nhiệm vụ chính:

* Xây dựng website để admin có thể quản lí thông tin nhân viên, dự án và thời gian làm việc của toàn bộ nhân viên
* Xây dựng website để nhân viên có thể cập nhật thông tin tiến độ của công việc được giao, quản lí thời gian làm việc của mình và quản lí thông tin cá nhân.

## **Công nghệ sử dụng**

Hệ thống sử dụng những công nghệ mới và phổ biến trên thị trường:

* React JS
* Redux & Saga
* ASP .NET CORE Web API
* SQL Server
* Azure services
* Docker
* Nginx

## **Công cụ hỗ trợ**

* Visual Studio 2019
* Visual Studio Code
* SQL Server Management System
* Postman
* Git Kraken
* Filezilla

## **Cấu trúc đồ án**

**MỞ ĐẦU-** Giới thiệu tổng quan về đồ án, lý do chọn đề tài. Trình bày về mục đích, nhiệm vụ và cách thức tiến hành để đạt được kết quả cao. Tóm tắt nội dung của những phần tiếp theo.

**Chương 1: LÝ THUYẾT VÀ CÔNG NGHỆ-** Chương này giới thiệu về tất cả cơ sở lý thuyết và công nghệ sử dụng trong đồ án.

**Chương 2: PHÂN TÍCH VÀ THIẾT KẾ-** Chương này trình bày về tổng quan hệ thống, việc phân tích, đặc tả yêu cầu, và thiết kế hệ thống. Trình bày về phương pháp phát hiện té ngã được lựa chọn.

**Chương 3: TRIỂN KHAI VÀ KẾT QUẢ-** Chương này trình bày về kết quả của việc lập trình của hệ thống, các hình ảnh thực tế và các bước cài đặt để sử dụng hệ thống.

**KẾT LUẬN-** Nêu lên kết luận của nhóm về đồ án, các bài học trong quá trình thực hiện, nên lên nhận xét về các vấn đề còn tồn tại và các đề xuất.

**TÀI LIỆU THAM KHẢO-** Trình bày về các thông tin cần thiết, liên quan đến các nguồn trích dẫn trong đồ án..

# **CƠ SỞ LÝ THUYẾT**

## **React Js**

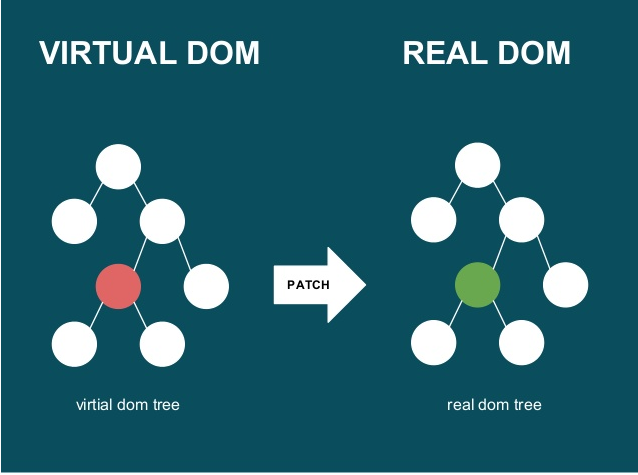
### ***Giới thiệu***

React là một thư viện UI phát triển tại Facebook để hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. React được sử dụng tại Facebook trong production, và đồng thời ứng dụng instagram cũng được viết hoàn toàn trên React.

Một trong những điểm hấp dẫn của React là thư viện này không chỉ hoạt động trên phía client, mà còn được render trên server và có thể kết nối với nhau. React so sánh sự thay đổi giữa các giá trị của lần render này với lần render trước và cập nhật ít thay đổi nhất trên **DOM**. Trươc khi đến cài đặt và cấu hình, chúng ta sẽ đi đến một số khái niệm cơ bản:

### ***Virtual DOM***

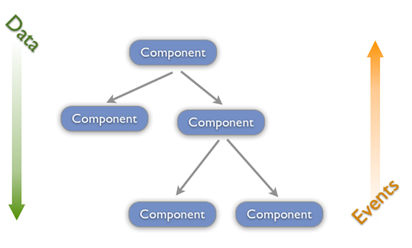
Công nghệ DOM ảo giúp tăng hiệu năng cho ứng dụng. Việc chỉ node gốc mới có trạng thái và khi nó thay đổi sẽ tái cấu trúc lại toàn bộ, đồng nghĩa với việc DOM tree cũng sẽ phải thay đổi một phần, điều này sẽ ảnh hưởng đến tốc độ xử lý. React JS sử dụng Virtual DOM (DOM ảo) để cải thiện vấn đề này.Virtual DOM là một object Javascript, mỗi object chứa đầy đủ thông tin cần thiết để tạo ra một DOM, khi dữ liệu thay đổi nó sẽ tính toán sự thay đổi giữa object và tree thật, điều này sẽ giúp tối ưu hoá việc re-render DOM tree thật.



Hình 1.1 Đặc trưng Virtual DOM

### ***One-way data binding***

React sử dụng cơ chế one-way data binding – luồng dữ liệu 1 chiều. Dữ liệu được truyền từ parent đến child thông qua props. Luồng dữ liệu đơn giản giúp chúng ta dễ dàng kiểm soát cũng như sửa lỗi.



Hình 1.2 Các luồng dữ liệu và sự kiện trong React

Với các đặc điểm ở trên, React dùng để xây dựng các ứng dụng lớn mà dữ liệu của chúng thay đổi liên tục theo thời gian. Dữ liệu thay đổi thì hầu hết kèm theo sự thay đổi về giao diện. Ví dụ như Facebook: trên Newsfeed của bạn cùng lúc sẽ có các status khác nhau và mỗi status lại có số like, share, comment liên tục thay đổi. Khi đó React sẽ rất hữu ích để sử dụng.

### ***Giới thiệu về JSX***

JSX là một dạng ngôn ngữ cho phép viết các mã HTML trong Javascript. Đặc điểm: Faster: Nhanh hơn. JSX thực hiện tối ưu hóa trong khi biên dịch sang mã Javacsript. Các mã này cho thời gian thực hiện nhanh hơn nhiều so với một mã tương đương viết trực tiếp bằng Javascript. Safer: an toàn hơn. Ngược với Javascript, JSX là kiểu statically-typed, nghĩa là nó được biên dịch trước khi chạy, giống như Java, C++. Vì thế các lỗi sẽ được phát hiện ngay trong quá trình biên dịch. Ngoài ra, nó cũng cung cấp tính năng gỡ lỗi khi biên dịch rất tốt. Easier: Dễ dàng hơn. JSX kế thừa dựa trên Javascript, vì vậy rất dễ dàng để cho các lập trình viên Javascripts có thể sử dụng

### ***Giới thiệu về Components***

React được xây dựng xung quanh các component, chứ không dùng template như các framework khác. Trong React, chúng ta xây dựng trang web sử dụng những thành phần (component) nhỏ. Chúng ta có thể tái sử dụng một component ở nhiều nơi, với các trạng thái hoặc các thuộc tính khác nhau, trong một component lại có thể chứa thành phần khác. Mỗi component trong React có một trạng thái riêng, có thể thay đổi, và React sẽ thực hiện cập nhật component dựa trên những thay đổi của trạng thái. Mọi thứ React đều là component. Chúng giúp bảo trì mã code khi làm việc với các dự án lớn. Một react component đơn giản chỉ cần một method render. Có rất nhiều methods khả dụng khác, nhưng render là method chủ đạo.

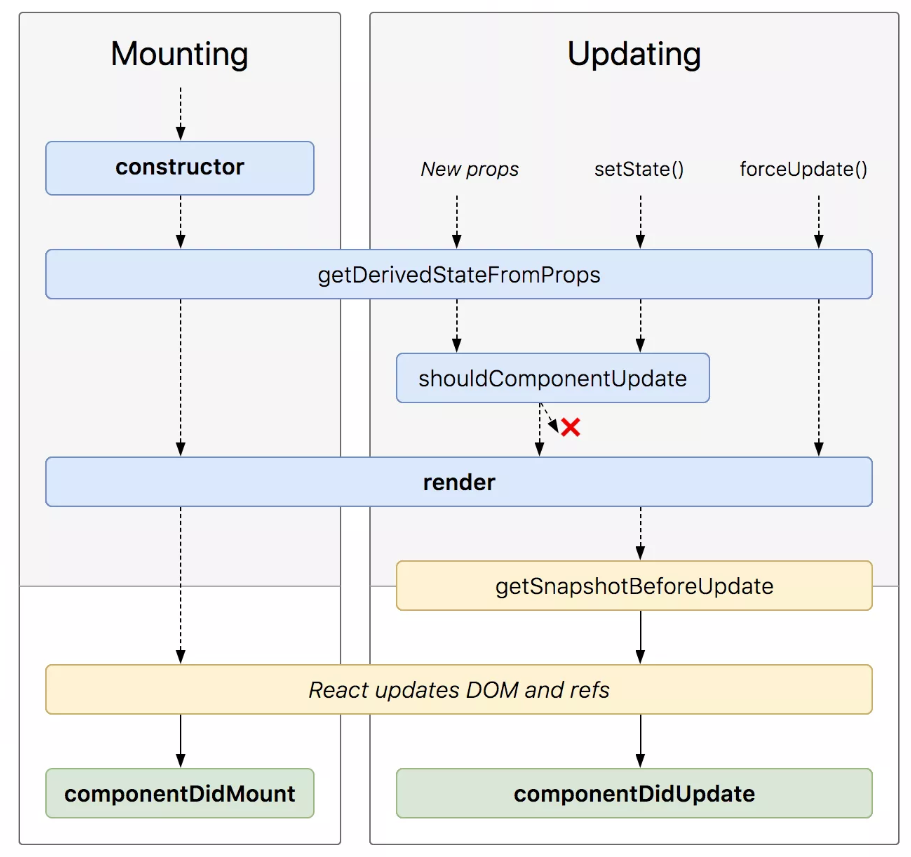
### ***Props và State***

Props: giúp các component tương tác với nhau, component nhận input gọi là props, và trả thuộc tính mô tả những gì component con sẽ render. Prop là bất biến.

State: thể hiện trạng thái của ứng dụng, khi state thay đồi thì component đồng thời render lại để cập nhật UI.

### ***Lifecycle***

*Lifecycle methods* là những method được gọi tại một thời điểm nào đó trong vòng đời của một component. Chúng ta có thể viết một *lifecycle methods* được gọi trước khi component được render trong lần đầu tiên. Hoặc được gọi sau khi component được render trong những lần sau.



Hình 1.3 Các luồng dữ liệu và sự kiện trong React

## **Redux-Saga**

### ***Giới thiệu***

Redux-Saga là một thư viện redux middleware, giúp quản lý những side effect trong ứng dụng redux trở nên đơn giản hơn. Bằng việc sử dụng tối đa tính năng Generators (function\*) của ES6, nó cho phép ta viết async code nhìn giống như là synchronos.

### ***Side effect***

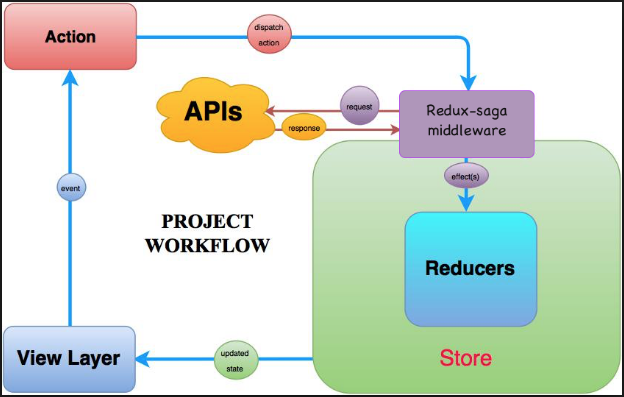
Ta đã biết tất cả những xử lý ở REDUCER đều phải là synchronous và pure tức chỉ là xử lý đồng bộ. Nhưng trong ứng dụng thực tế thì cần nhiều hơn vậy ví dụ như asynchronous (thực hiện một số việc như gọi một hàm AJAX để fetch dữ liệu về nhưng cần đợi kết quả chứ kết quả không trả về ngay được) hoặc là impure (thực hiện lưu, đọc dữ liệu ra bên ngoài như lưu dữ liệu ra ổ cứng hay đọc cookie từ trình duyệt… đều cần đợi kết quả). Những việc như thế trong lập trình hàm gọi nó là side effects.

### ***Generator function***

Khác với function bình thường là thực thi và trả về kết quả, thì Generator function có thể thực thi, tạm dừng trả về kết quả và thực thi bằng tiếp. Từ khóa để làm được việc đấy là “YIELD”. Generator được đưa ra cách đây mấy chục năm nhưng đến ES2015 mới được bổ sung, các ngôn ngữ khác đã được bổ sung tính năng này như C#, PHP, Ruby, C++, R….

### ***Cách hoạt động***

Đối với logic của saga, ta cung cấp một hàm cho saga, chính hàm này là hàm đứng ra xem xét các action trước khi vào store, nếu là action quan tâm thì nó sẽ thực thi hàm sẽ được thực thi, nếu bạn biết khái niệm hook thì hàm cung cấp cho saga chính là hàm hook. Điều đặc biệt của hàm hook này nó là một generator function, trong generator function này có yield và mỗi khi yield ta sẽ trả về một plain object. Object trả về đó được gọi Effect object. effect object này đơn giản chỉ là một object bình thường nhưng chứa thông tin đặc biệt dùng để chỉ dẫn middleware của Redux thực thi các hoạt động khác ví dụ như gọi một hàm async khác hay put một action tới store. Để tạo ra effect object đề cập ở trên thì ta gọi hàm từ thư viện của saga là redux-saga/effects.



Hình 1.4 Mô tả cách thức hoạt động của Redux - Saga

## **WEB API**

### ***ASP.NET CORE***

ASP.NET Core là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile. Do là một framework đa nền tảng nên .NET hỗ trợ hầu hết các hệ điều hành như Windows, Linux, macOS.

Ứng dụng ASP.NET Core có thể chạy trên phiên bản đầy đủ của .NET Framework. Nó được thiết kế để cung cấp và tối ưu development framework cho những dụng cái mà được triển khai trên đám mây (clound) hoặc chạy on-promise.

DotNET Core framework bao gồm .NET Core, ASP.NET Core, Entity Framework Core. Các công nghệ này khác với .NET Framework ở chỗ chúng chạy CoreCLR runtime(used in Universal Windows Platform)

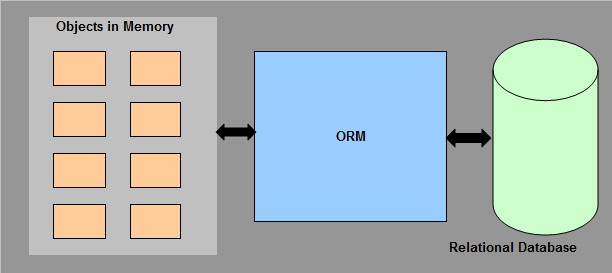
### ***Entity Framework***

Entity Framework được ra mắt lần đầu tiên vào năm 2008, giúp cho việc truy vấn cơ sỡ dữ liệu quan hệ trở lên đơn giản, dễ học, giúp giảm bớt thời gian phát triển dự án xuống đáng kể.

Entity Framework là một Object Relational Mapper(ORM) một công cụ mapping giữa các objects trong ứng dụng với các tables và columns của cơ sở dữ liệu quan hệ:

* Entity Framework là một Open source ORM Framework.
* Một ORM có thể tạo kết nối đến một Relational database, thực thi cậu lệnh và trả về kết quả là các objects trong ứng dụng.
* Một ORM sẽ theo dõi sự thay đổi dữ liệu,trạng thái của các objects trả về ở trên, mỗi khi có sự thay đổi dữ liệu từ những objects này ORM sẽ gửi những thay đổi đó đến Relational database.

Hiểu một cách đơn giản thì ORM là công cụ tự động tạo ra các Objects trong ứng dựng từ tables, views, store procedures của một Relational Database nào đó..



Hình 1.5 Mô tả cách thức hoạt động của Entity Framework

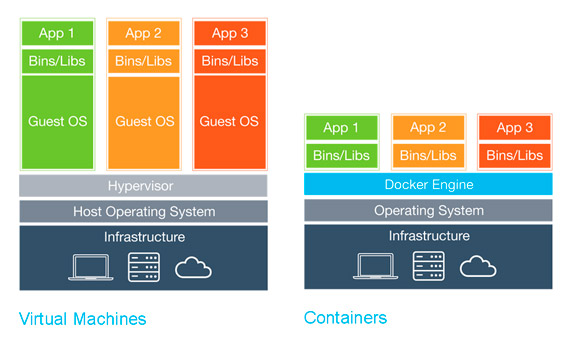
### ***Phương thức HTTP***

**HTTP** (Tiếng Anh: **HyperText Transfer Protocol** - Giao thức truyền tải siêu văn bản) là một trong năm giao thức chuẩn của mạng Internet, được dùng để liên hệ thông tin giữa Máy cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client) trong mô hình Client/Server dùng cho World Wide Web-WWW, **HTTP** là một giao thức thuộc tầng ứng dụng, nằm trên cặp giao thức tầng giao vận & tầng mạng là TCP/IP.

## **Docker**

### ***Khái niệm***

Docker - đây là một công cụ tạo môi trường được packaging (Container) trên máy tính độc lập mà không làm tác động tới môi máy, môi trường trong Docker sẽ chạy độc lập. Mỗi containers chứa ứng dụng gồm chứa các thư viện riêng, nhưng kernel được chia sẻ với các containers khác. Mỗi kernel được chạy trong các môi trường độc lập với nhau. Docker infrastructure chạy được trên mọi máy tính, đám mây hay nền tảng nào. Vì vậy bạn có thể vận chuyển ứng dụng đến bất cứ đâu và không cần phải quan tâm đến môi trường phát triển, thiếu thư viện, ... Tựu chung lại thì Docker rất tốt tại việc xây dựng và chia sẻ Disk Image qua hệ thống Docker Index, là một phần mềm quản lý cơ sở hạ tầng, làm việc tốt vời với các công cụ quản lý file config (vd: Chef, Puppet), sử dụng btrfs để giảm sát các file hệ thống và có thể được chia sẻ với user khác. (Như cách hoạt động của Git) và có một bộ kho trung tâm của các Disk Images (có thể được public hoặc private), điều này cho phép bạn dễ dàng chạy trên nhiều hệ điều hành khác nhau (Ubuntu, Centos, Fedora, Gentoo).



Hình 1.6 Mô tả cách thức hoạt động của Docker

### ***Lợi ích của Docker***

* Thời gian khởi động nhanh
* Gói gọn mọi environment cần thiết trong một container, có thể chạy container trên mọi môi trường win, linux, macos, …
* Tốc độ build nhanh hơn.
* Dễ dàng thiết lập môi trường làm việc khi có vấn đề, sự cố với server.
* Giữ cho workspace sạch sẽ hơn
* Các container có thể chạy độc lập
* Có thể chạy nhiều container trên cùng một máy

# **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

## **Đặt vấn đề**

Hệ thống cần giải quyết được bài toán tối ưu quá trình quản lí dự án của công ty. Hiện tại công ty đang vận hành các dự án theo mô hình scrum-agile nên hệ thống cần đáp ứng việc tạo mới các dự án, quản lí các dự án theo mô hình này bằng các cột tiến trình. Các tiến trình có thể tùy chỉnh và thay đổi để phù hợp với từng dự án. Đồng thời việc tạo mới, quản lí các công việc phải hợp lí, logic

Module quản lí nhân viên

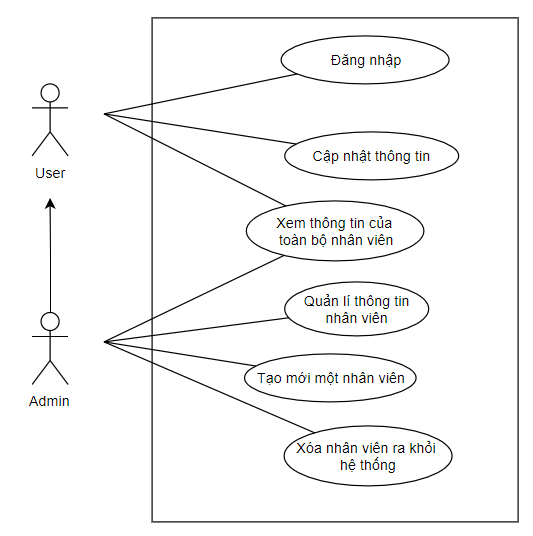
Module quản lí thời gian biểu

## **Cách chức năng chính**

### ***Quản lí thông tin nhân viên***

Để thuận lợi cho việc truy xuất cũng như quản lí thông tin các nhân viên hiện thời của công ty, hệ thống sẽ cho phép:

* Nhân viên quản lí, cập nhật thông tin cá nhân của mình.
* Xem thông tin của những nhân viên hiện có trong công ty.
* Tìm kiếm nhân viên theo tên, chức vụ hiện thời, hoặc bằng các kỹ năng hiện có của nhân viên.
* Admin có thể tạo mới tài khoản cho nhân viên mới, chỉnh sửa, khóa hoặc xóa tài khoản của nhân viên ra khỏi hệ thống

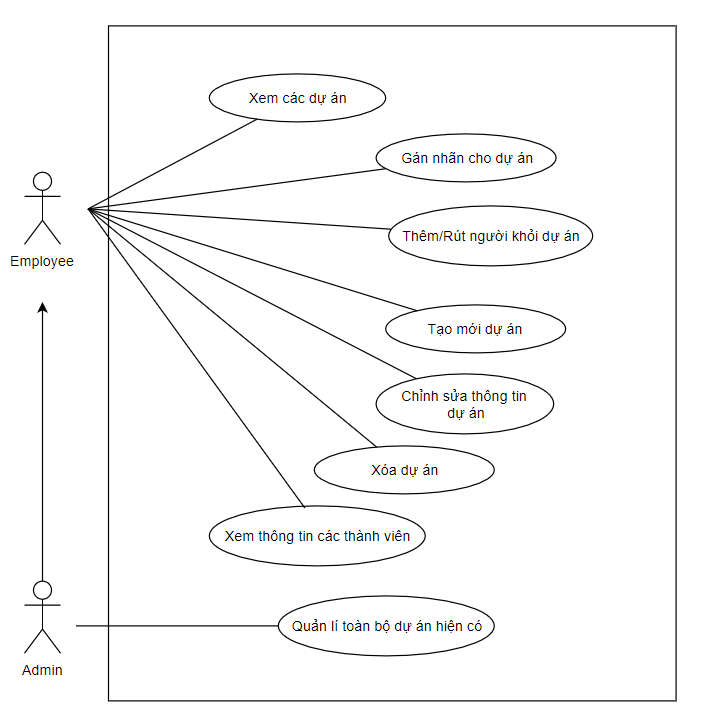


*Hình 2.1 Sơ đồ use-case cho module quản lí nhân viên*

### ***Quản lí dự án***

Hệ thống sẽ cung cấp công cụ để quản lý các dự án của công ty:

* Nhân viên có thể xem các dự án mà mình đang tham gia
* Nhân viên có thể tạo mới một dự án
* Nhân viên có thể thêm người vào dự án
* Nhân viên có thể thêm các nhãn màu cho dự án để dễ phân biệt
* Nhân viên có thể chỉnh sửa thông tin của dự án
* Nhân viên có thể xem hồ sơ của các thành viên trong dự án
* Giám đốc / trưởng bộ phận (Admin) có thể quản lí, giám sát toàn bộ dự án của công ty

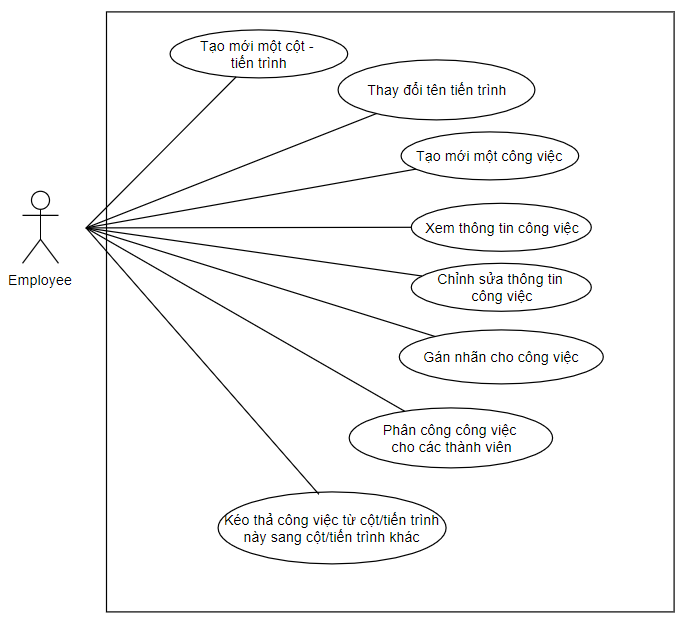


Hình 2.2 Sơ đồ use-case cho module quản lí dự án

### ***Quản lí công việc***

Hệ thống sẽ cung cấp công cụ để quản lý tiến độ công việc các dự án của công ty theo dạng scrum board:

* Mọi thành viên trong nhóm có quản lí tiến độ công việc của dự án bằng cách thêm các luồng thực hiện công việc, thêm mới và quản lí các nhiệm vụ, công việc.
* Thành viên có thể giao, nhận các công việc.
* Thành viên có thể quan sát tiến độ của dự án thông qua tổng thời gian ước lượng của toàn bộ công việc và lượng thời gian thực tế đã tiêu tốn cho những công việc trong dự án.

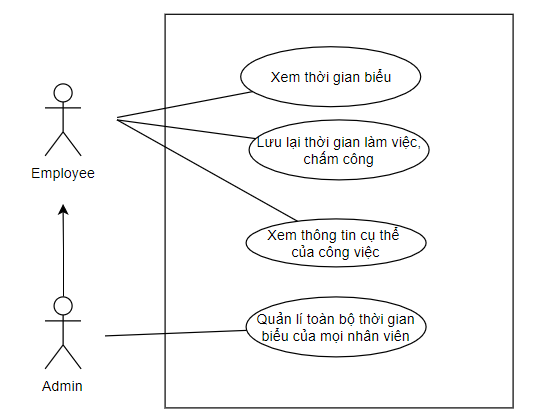


*Hình 2.3 Sơ đồ use-case cho module quản lí tiến độ công việc trong dự án*

### ***Quản lí thời gian làm việc***

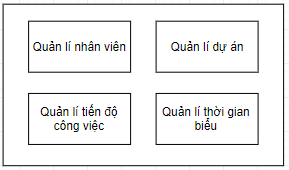
Để thuận lợi hơn trong việc đánh giá hiệu suất làm việc của nhân viên, hệ thống sẽ cho phép:

* Nhân viên có thể quản lí, kiểm soát thời gian làm việc của mình thông qua việc lưu lại thời gian đã tiêu tốn cho từng công việc được giao.
* Nhân viên có thể xem toàn bộ thời gian làm việc của mình
* Nhân viên có thể xem thông tin của công việc đã làm ngay tại thời gian biểu
* Giám đốc / trưởng bộ phận có thể quan sát thời gian làm việc của mọi nhân viên.



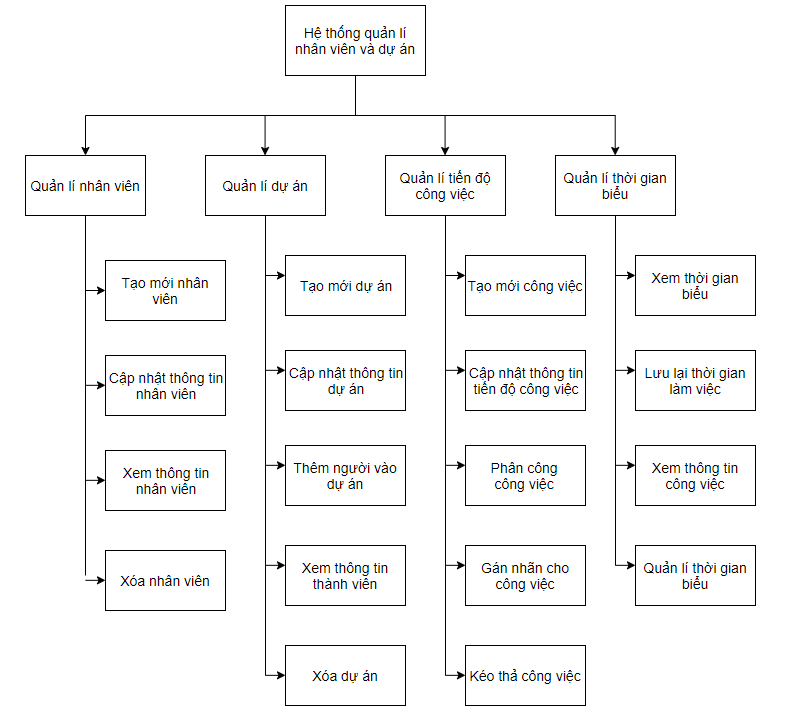
Hình 2.4 Sơ đồ use-case module quản lí thời gian làm việc

## **Các module chính của hệ thống**



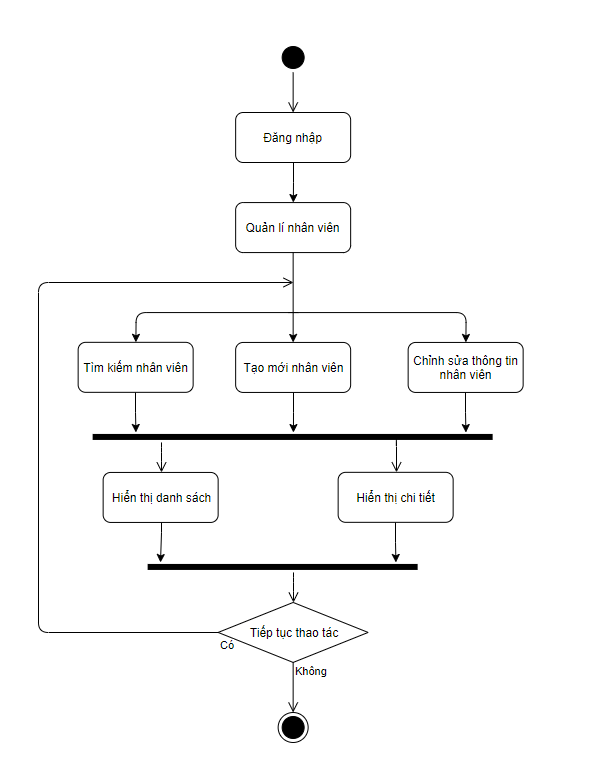
Hình 2.5 Các module chính của dự án

## **Sơ đồ phân rã chức năng**

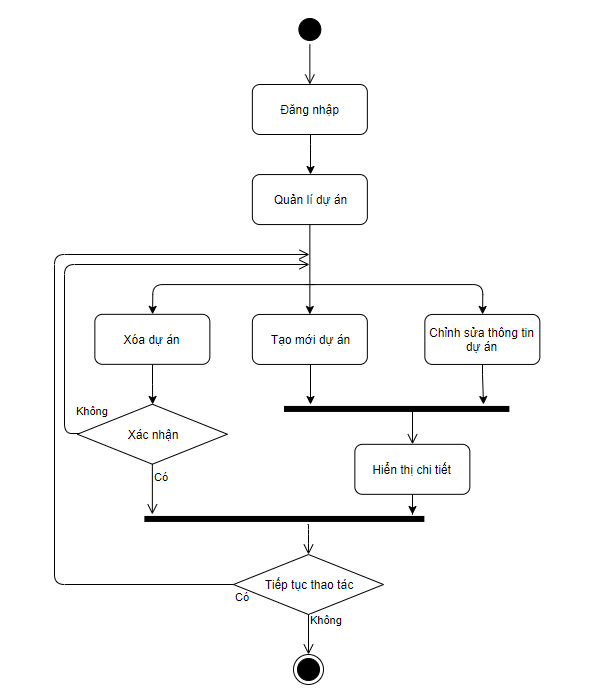


Hình 2.6 Sơ đồ phân rã chức năng

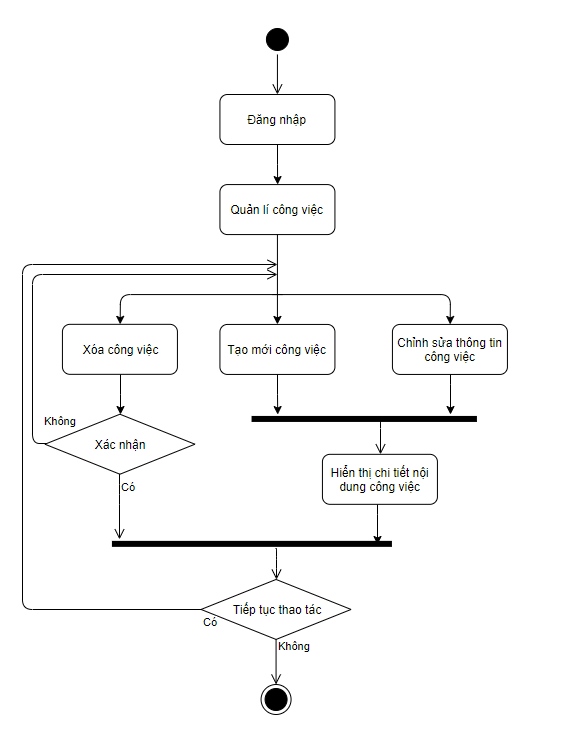
## **Sơ đồ hoạt động**



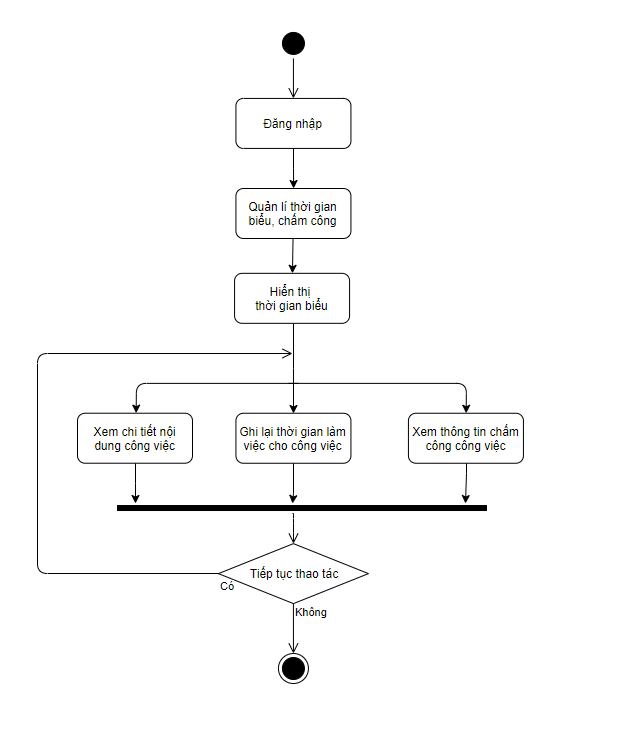
Hình 2.7 Sơ đồ hoạt động của module quản lí nhân viên



*Hình 2.8 Sơ đồ hoạt động của module quản lí dự án*

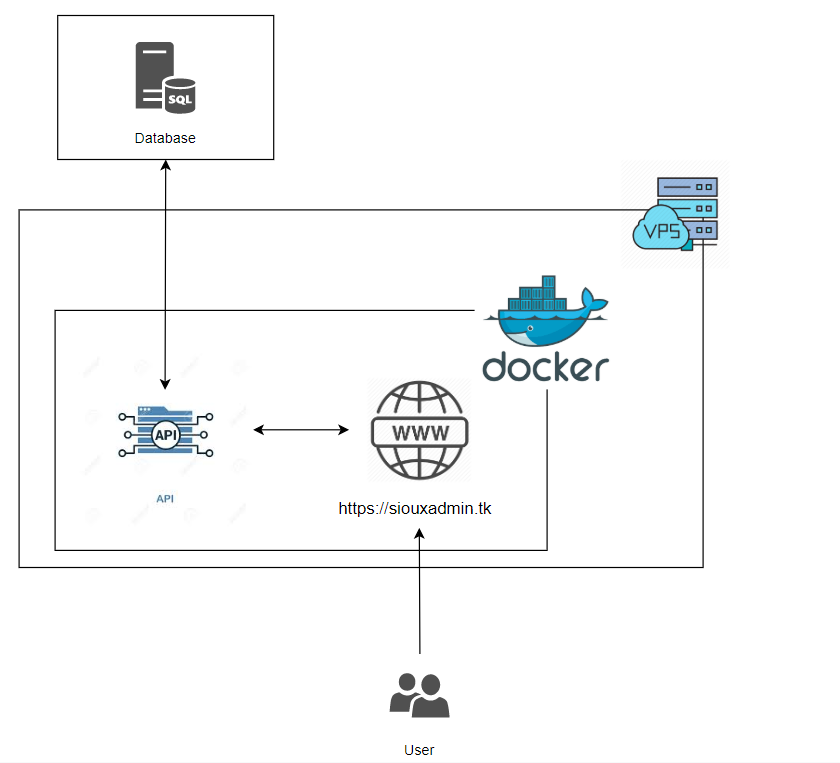


*Hình 2.9 Sơ đồ hoạt động của module quản lí tiến độ công việc*



Hình 2.10 Sơ đồ hoạt động của module quản lí thời gian biểu, chấm công

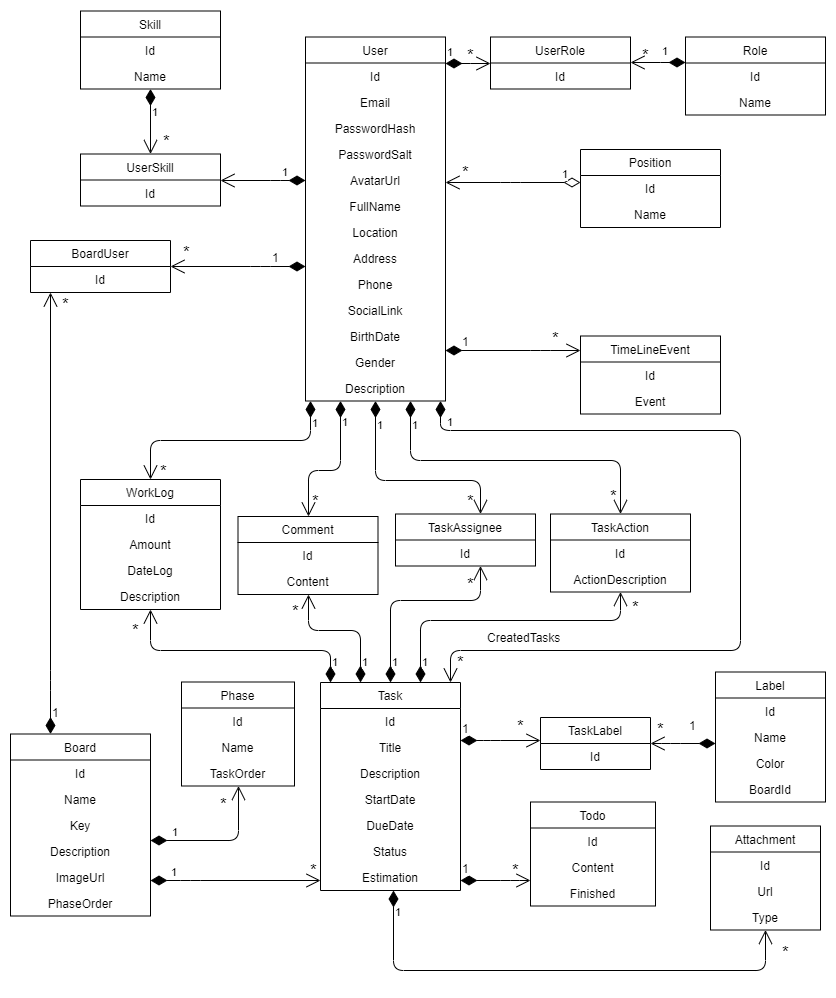
## **Luồng dữ liệu**



*Hình 2.11 Mô tả luồng đi của dữ liệu*

Thông thường, nhân viên sẽ đăng nhập vào hệ thống quản lí qua website. Các thao tác trên website sẽ thông qua API để truy cập đến cơ sở dữ liệu. Sau khi lấy dữ liệu từ cơ sở dữ liệu, API server sẽ trả tín hiệu về cho trình duyệt để hiển thị.

## **Sơ đồ quan hệ**



Hình 2.12 Sơ đồ lớp cho dự án

## **Thiết kế cơ sở dữ liệu**

### ***Sơ đồ quan hệ, thực thể dữ liệu***

****

Hình 2.13 Sơ đồ quan hệ, thực thể dữ liệu

### ***Đặc tả các bảng dữ liệu***

Bảng 2.1 Bảng User

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên** | **Mô tả** | **Data Type** | **Note** |
| **Id** | Id của bảng User | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Email | Địa chỉ Email của nhân viên | Text |  |
| PasswordHash | Mã hóa hash của password | Text |  |
| PasswordSalt | Mã hóa salt của password | Text |  |
| AvatarUrl | Link ảnh Avatar của nhân viên | Text |  |
| FullName | Tên đầy đủ của nhân viên | Text |  |
| Location | Tỉnh/thành phố hiện tại của nhân viên | Text |  |
| Address | Địa chỉ nhà của nhân viên | Text |  |
| Phone | Số điện thoại của nhân viên | Text |  |
| SocialLink | Link trang cá nhân của nhân viên | Text |  |
| BirthDate | Ngày sinh của nhân viên | DateTime |  |
| Gender | Giới tính của nhân viên | Text |  |
| Description | Sơ lược giới thiệu nhân viên | Text |  |
| PositionId | Id chức vụ của nhân viên | Char | Khóa ngoại |

Bảng 2.2 Bảng Position

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Position | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên chức vụ của nhân viên | Text |  |

*Bảng 2.3 Bảng Skill*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Skill | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên kĩ năng của nhân viên | Text |  |

*Bảng 2.4 Bảng Skill*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| UserId | Id của User | Char | Khóa ngoại |
| SkillId | Id của Skill | Char | Khóa ngoại |

*Bảng 2.5 Bảng Role*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Role | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên vai trò của nhân viên | Text |  |

*Bảng 2.6 Bảng UserRole*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng UserRole | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| UserId | Id của User | Char | Khóa ngoại |
| RoleId | Id của Role | Char | Khóa ngoại |

*Bảng 2.7 Bảng TimeLineEvent*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng TimeLineEvent | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Event | Tên sự kiện trong đời của nhân viên | Text |  |
| UserId | Id của User | Char | Khóa ngoại |

*Bảng 2.8 Bảng Board*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Board | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên của dự án | Text |  |
| Key | Key của dự án | Text |  |
| Description | Giới thiệu sơ lược dự án | Text |  |
| ImageUrl | Link ảnh đại diện của dự án | Text |  |
| PhaseOrder | Thứ tự của các tiến tình trong dự án | Text |  |

*Bảng 2.9 Bảng Phase*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên của tiến trình | Text |  |
| TaskOrder | Thứ tự của các task trong cột | Text |  |
| BoardId | Id của dự án | Char | Khóa ngoại |

*Bảng 2.10 Bảng BoardUser*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| UserId | Id của User | Char | Khóa ngoại |
| BoardIdId | Id của Board | Char | Khóa ngoại |

*Bảng 2.11 Bảng Task*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Title | Title của công việc | Text |  |
| Description | Sơ lược về công việc | Text |  |
| StartDate | Ngày bắt đầu công việc | DateTime |  |
| DueDate | Ngày kết thúc công việc | DateTime |  |
| Status | Tình trạng công việc | Text |  |
| Estimation | Ước lượng thời gian làm việc | TimeSpan |  |
| UserId | Id của nhân viên tạo ra công việc | Char | Khóa ngoại |
| BoardId | Id của dự án | Char | Khóa ngoại |

*Bảng 2.12 Bảng Attachment*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Attachment | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Url | Link Attachment | Text |  |
| Type | Loại attachment | Int |  |
| TaskId | Id của công việc | Char | Khóa ngoại |

*Bảng 2.13 Bảng Todo*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Todo | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Content | Nội dung phải làm của công việc | Text |  |
| Finished | Trạng thái công việc | Bool |  |
| TaskId | Id của công việc | Char | Khóa ngoại |

*Bảng 2.14 Bảng Label*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Label | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Name | Tên của nhãn dán cho công việc | Text |  |
| Color | Màu của nhãn dán | Text |  |
| BoardId | Id của dự án | Char | Khóa ngoại |

*Bảng 2.15 Bảng TaskLabel*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng TaskLabel | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| TaskId | Id của công việc | Char | Khóa ngoại |
| LabelId | Id của nhãn dán | Char | Khóa ngoại |

*Bảng 2.16 Bảng WorkLog*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng WorkLog | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Amount | Khối lượng thời gian làm việc | TimeSpan |  |
| DateLog | Ngày ghi lại | DateTime |  |
| Description | Mô tả về worklog | Text |  |
| TaskId | Id của công việc | Char | Khóa ngoại |
| UserId | Id của nhân viên | Char | Khóa ngoại |

*Bảng 2.17 Bảng Comment*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Comment | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| Content | Nội dung comment | Text |  |
| TaskId | Id của công việc | Char | Khóa ngoại |
| UserId | Id của nhân viên | Char | Khóa ngoại |

*Bảng 2.18 Bảng TaskAssignee*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng TaskAssignee | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| TaskId | Id của công việc | Char | Khóa ngoại |
| UserId | Id của nhân viên | Char | Khóa ngoại |

*Bảng 2.19 Bảng TaskAction*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Mô tả | Data Type | Note |
| Id | Id của bảng Task Action | Char | Khóa chính |
| EntityStatus | Trạng thái của một thực thể | Integer |  |
| CreatedTime | Ngày khởi tạo | Date Time |  |
| UpdatedTime | Ngày chỉnh sửa cuối cùng | Date Time |  |
| ActionDescription | Mô tả về action | Text |  |
| TaskId | Id của công việc | Char | Khóa ngoại |
| UserId | Id của nhân viên | Char | Khóa ngoại |

# **TRIỂN KHAI VÀ KẾT QUẢ**

## **3.1. Môi trường triển khai**

*API Web Service:*

* Ngôn ngữ: C#
* Công nghệ: .NET CORE, Entity Framework
* Design Patern: Repository Patern
* Hệ quản trị cơ sở dữ liệu: SQL Server
* Hệ điều hành: Docker container trên môi trường linux

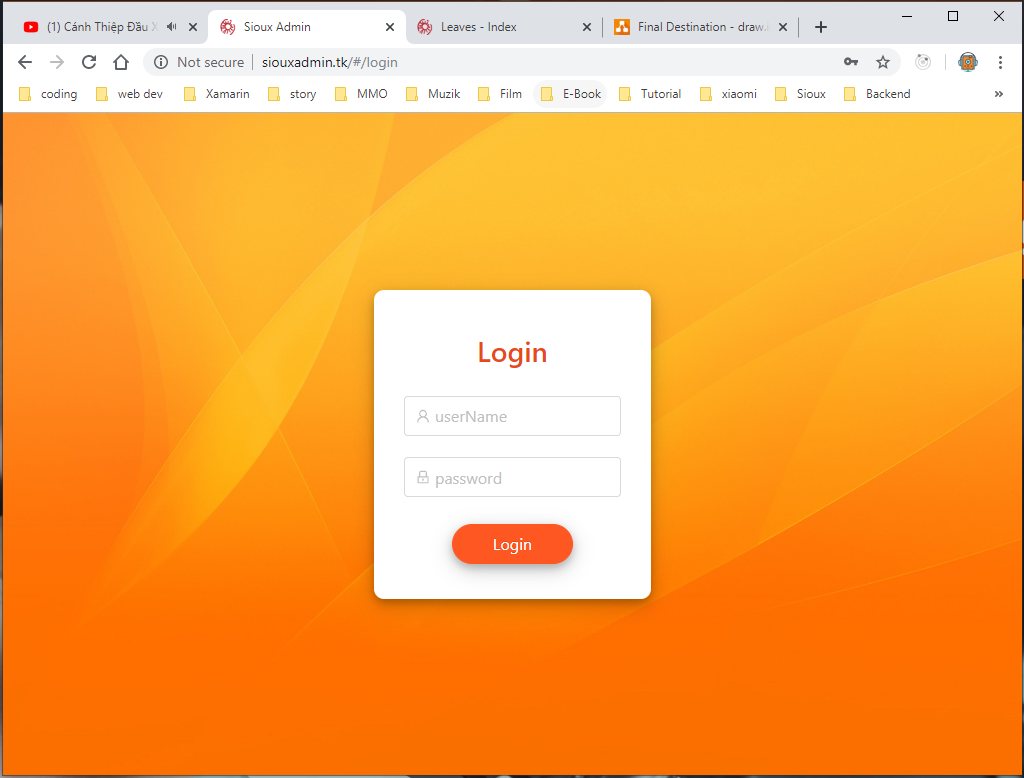
*Web Client:*

* Ngôn ngữ: HTML, CSS, JavaScript
* Công nghệ: React JS, Ant Design, Umi, Node
* Tích hợp trên hầu hết các trình duyệt Google Chrome, Firefox, IE, Edge, …
* Hệ điều hành: Docker container trên môi trường linux
* Địa chỉ website: [*http://siouxadmin.tk*](http://siouxadmin.tk)

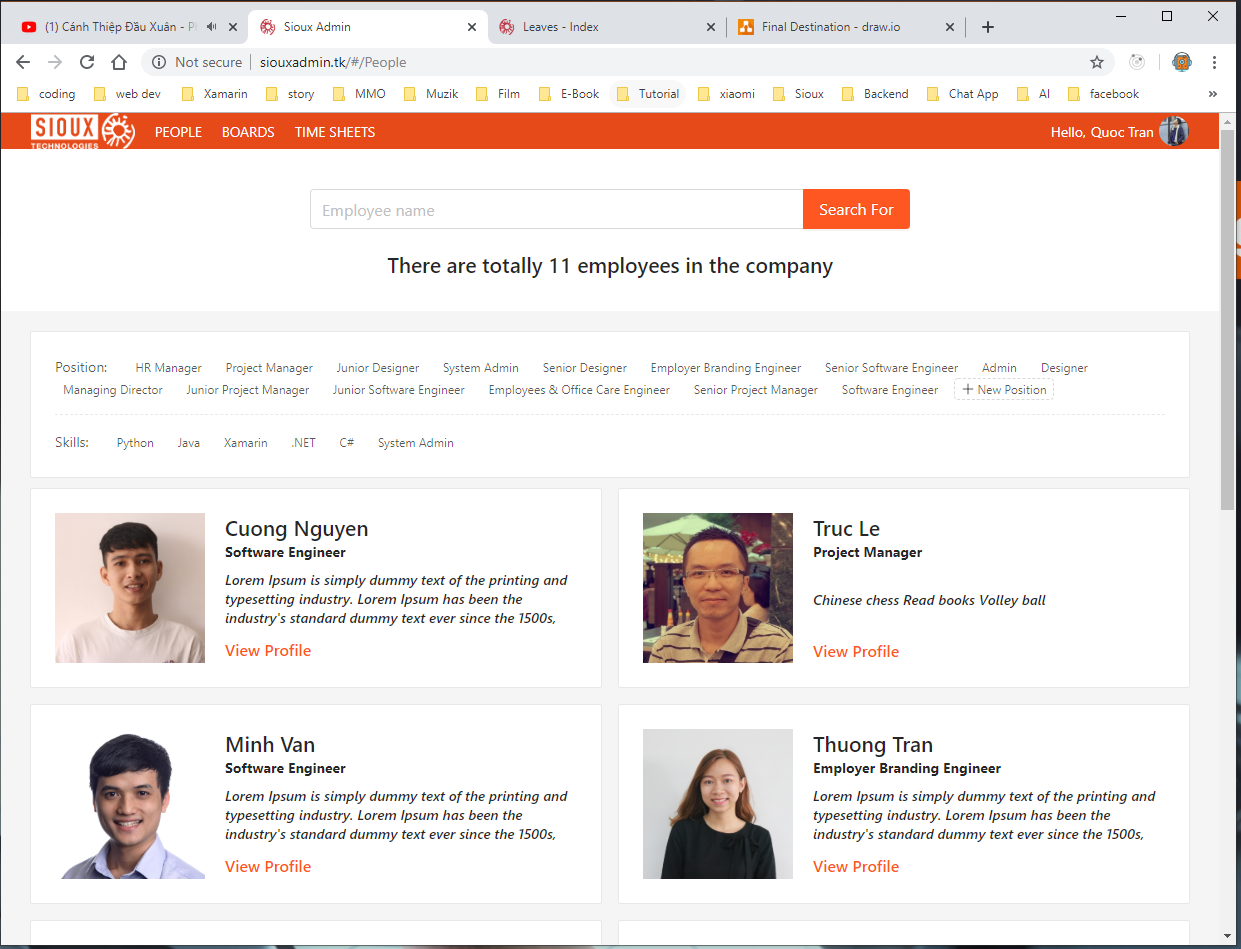
*Môi trường phát triển:*

* Visual Studio Code.
* Visual Studio 2019.
* Sql Server Management System

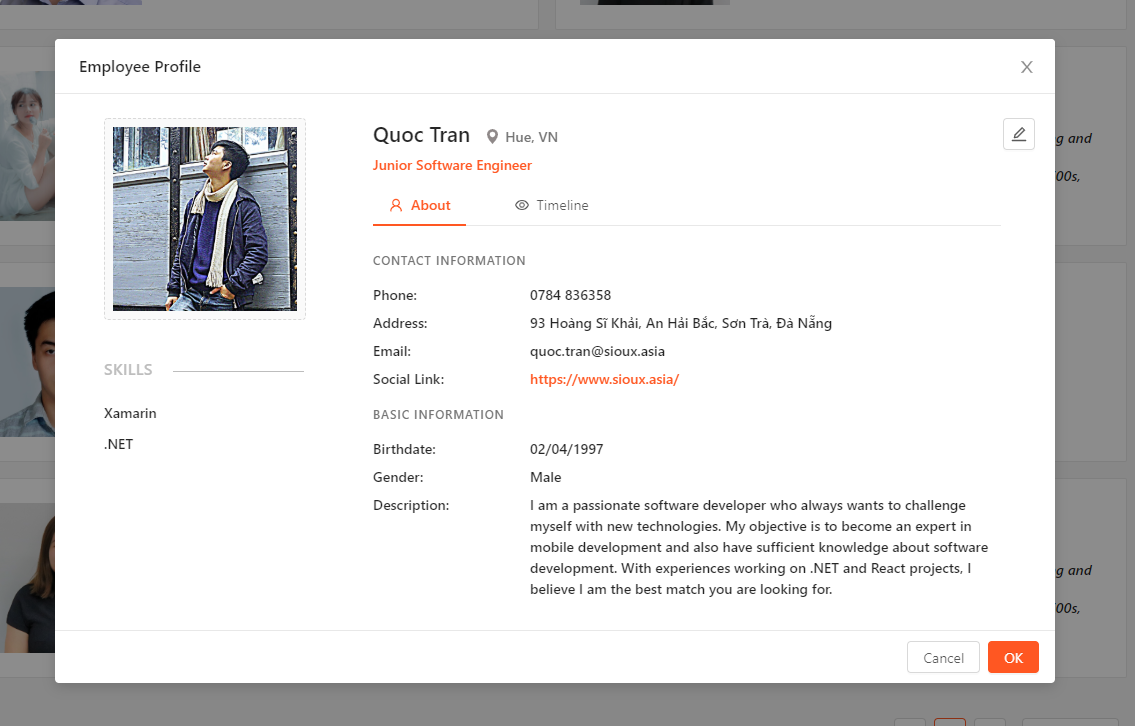
## **3.2. Kết quả thực hiện**



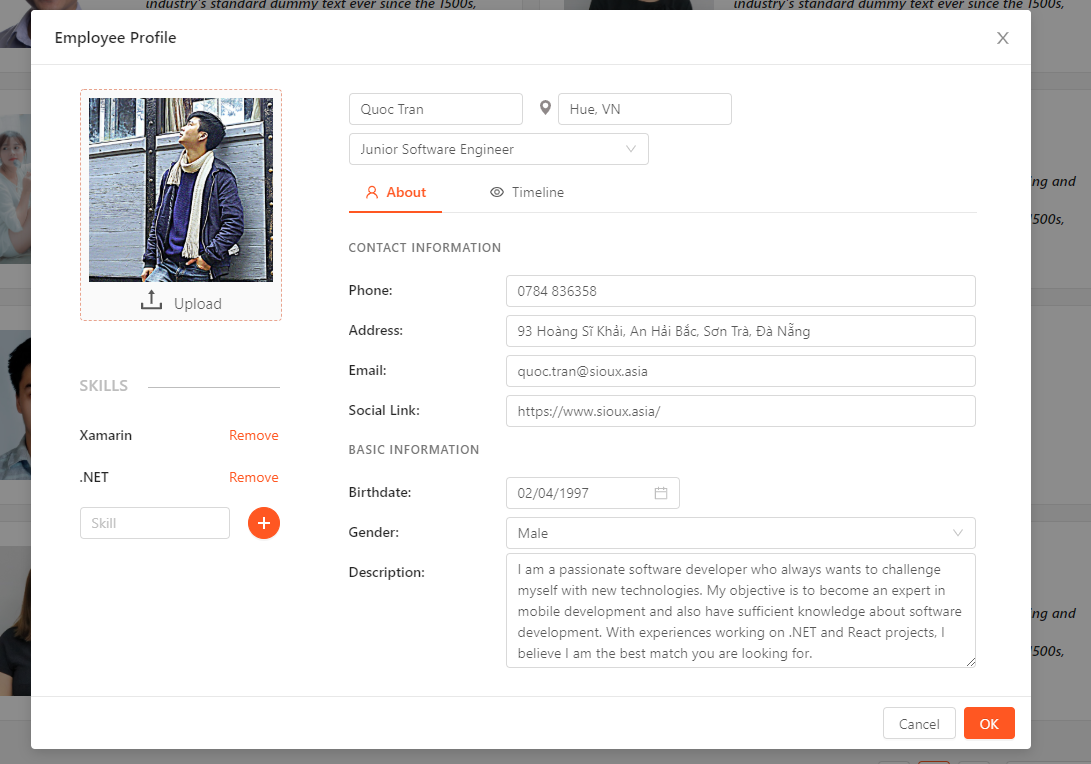
*Hình 3.1 Màn hình Login*



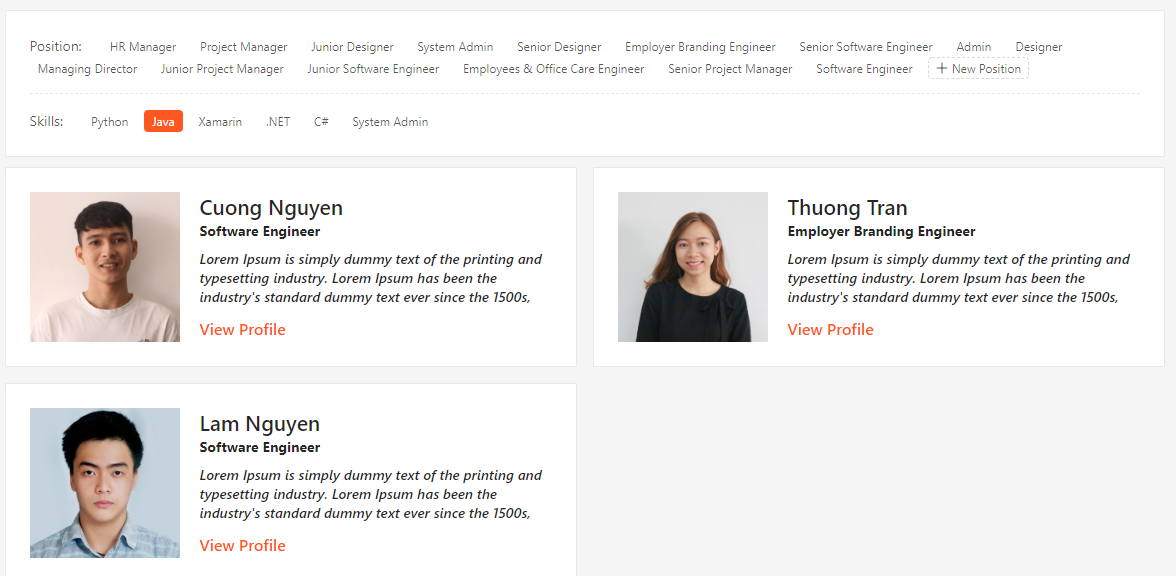
*Hình 3.2 Quản lí nhân viên*

****

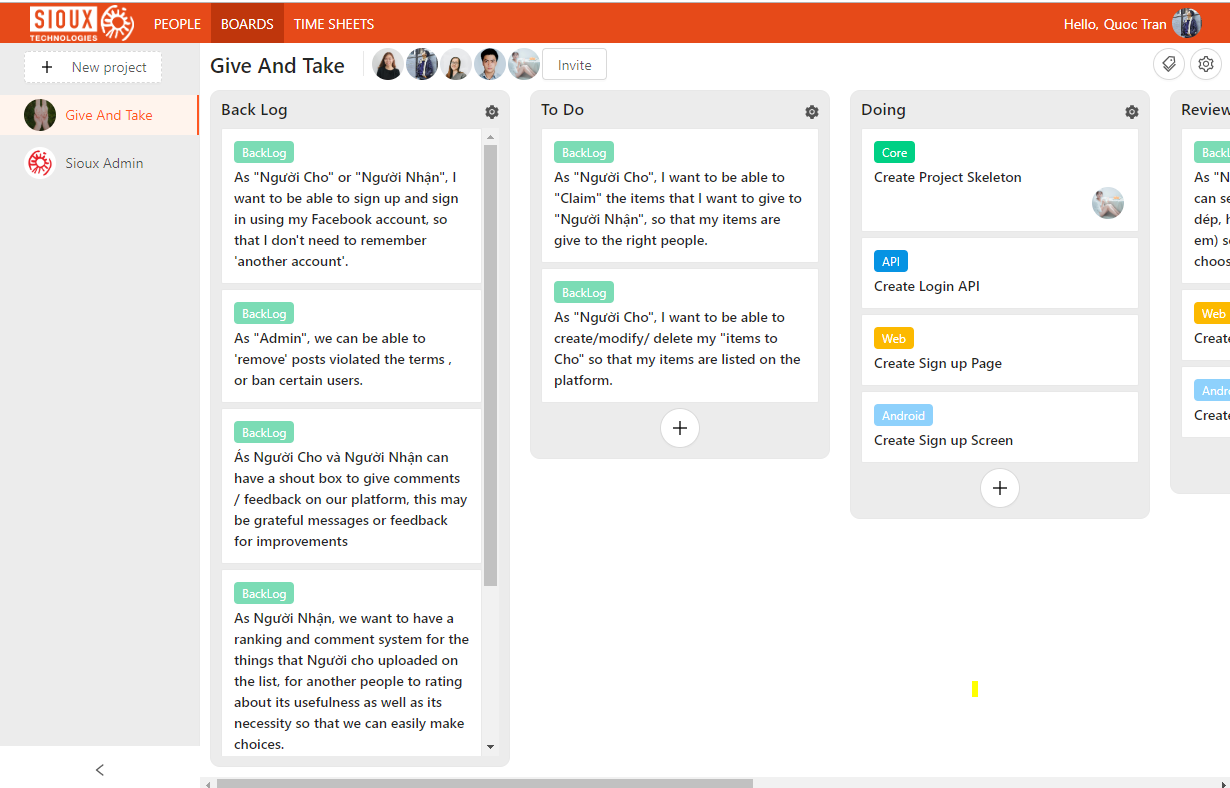
*Hình 3.3 Xem thông tin cá nhân của nhân viên*

****

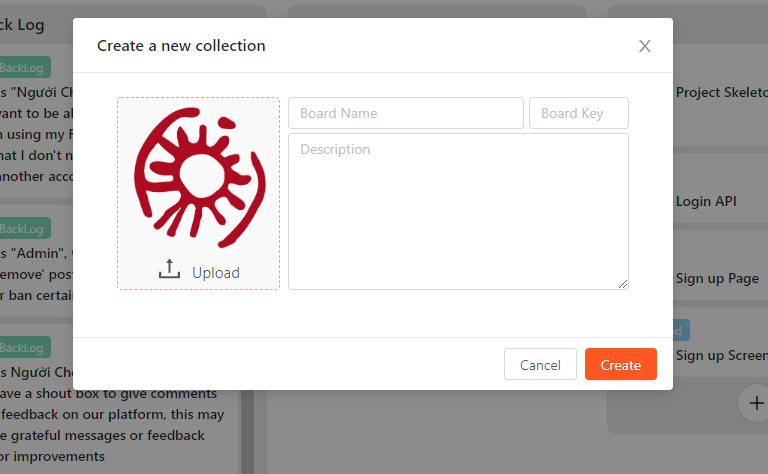
Hình 3.4 Chỉnh sửa thông tin cá nhân

****

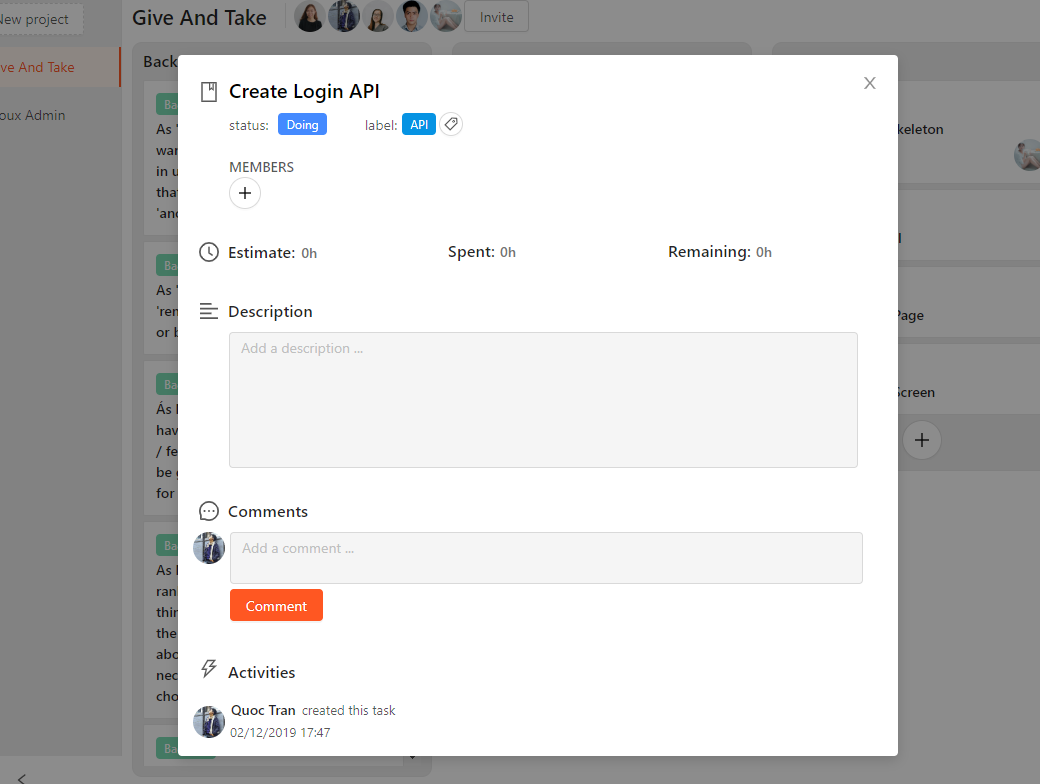
Hình 3.5 Lọc và tìm kiếm nhân viên dựa trên kĩ năng, vị trí, tên, …

****

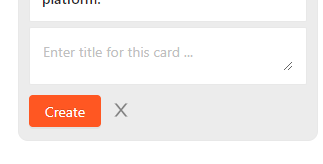
*Hình 3.6 Màn hình quản lí các dự án*

****

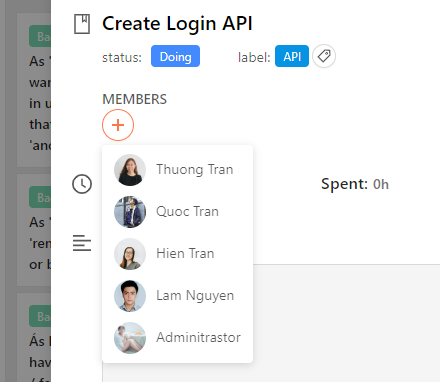
*Hình 3.7 Tạo một dự án mới*

****

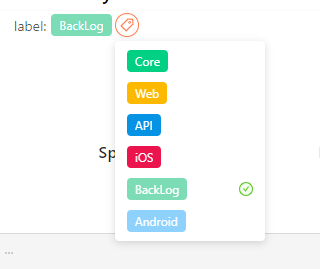
Hình 3.8 Thông tin chi tiết của một công việc

****

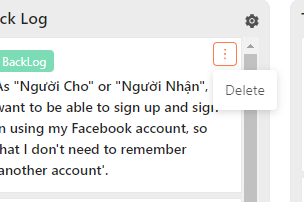
*Hình 3.9 Tạo một công việc mới*

****

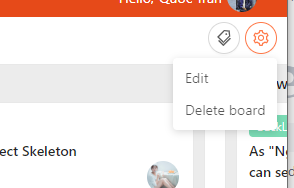
*Hình 3.10 Giao việc cho thành viên trong dự án*

****

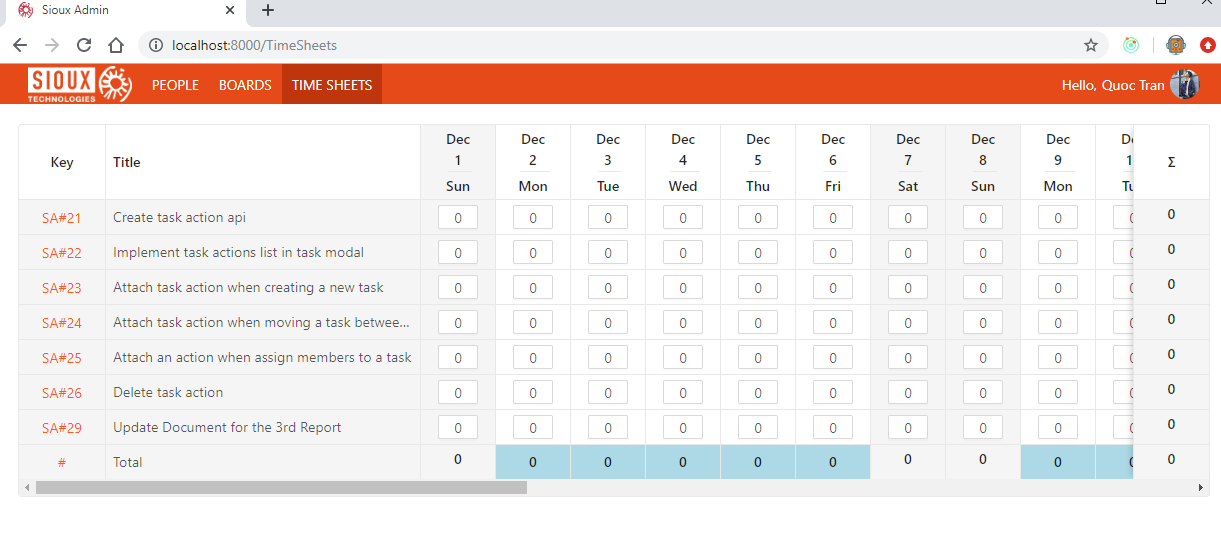
*Hình 3.11 Thêm nhãn cho công việc để dễ quản lí*

****

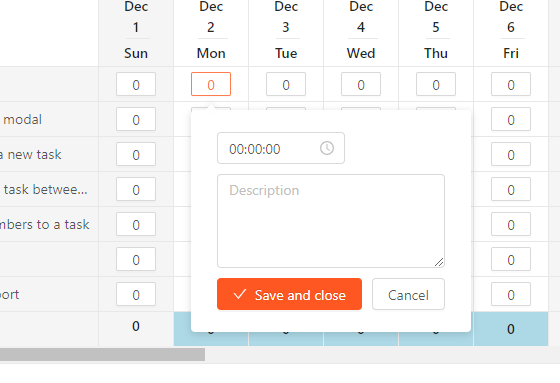
*Hình 3.12 Xóa một công việc*

****

*Hình 3.13 Các tùy chọn cho một bản*

****

*Hình 3.14 Quản lí thời gian*

****

*Hình 3.15 Ghi lại thời gian đã dành cho một công việc*

**KẾT LUẬN**

1. **Kết quả đạt được**

* Củng cố và áp dụng các kiến thức đã được tiếp thu vào thực tế
* Xây dựng website giúp công ty quản lí công việc và thời gian cho nhân viên, phục vụ cho mục đích giảm thiểu chi phí vận hành cho công ty
* Xây dựng thái độ làm việc nghiêm túc, tinh thần tự đánh gia bản thân
* Tăng thêm kiến thức và kinh nghiệm của bản thân về React, .NET và quy trình phát triển phần mềm
* Học được cách tự lập, tự xây dựng và nghiên cứu đề tài
* Hoàn thiện và phát triển các kỹ năng mềm và có cái nhìn mới về công việc

1. **Hướng phát triển**

Một số hướng nghiên cứu và phát triển của đề tài như sau:

* Cải thiện giao diện ứng dụng
* Tạo ứng dụng cho nền tảng di động để báo tiến độ công việc về điện thoại của nhân viên
* Có thể tích hợp thêm quản lí lịch nghỉ phép, báo vắng, ốm của nhân viên để hoàn thiện quy trình làm việc của nhân viên
* Tích hợp các biểu đồ trong quy trình Scrum – Agile như Burn down chart, … để tiện hơn cho việc kiểm soát tiến độ dự án

**TÀI LIỆU THAM KHẢO**

1. Tháp dân số Việt Nam 2015- http://taichinh24.com/thap-dan-so-viet-nam/
2. Phân tích dáng đi từ dữ liệu video, Nguyễn Trọng Nguyên (2014)
3. Histogram of oriented gradients, Wikipedia-

<https://en.wikipedia.org/wiki/Histogram_of_oriented_gradients>

1. Tìm hiểu về hog(histogram of oriented gradients), Viblo- <https://viblo.asia/p/tim-hieu-ve-hoghistogram-of-oriented-gradients-m68Z0wL6KkG>
2. Gaussian Mixture Models - Semantic Scholar, Douglas Reynolds-

<https://pdfs.semanticscholar.org/734b/07b53c23f74a3b004d7fe341ae4fce462fc6.pdf>

1. Motion History Image, Wikipedia-

<https://en.wikipedia.org/wiki/Motion_History_Images>

**Ghi chú:**

Sinh viên xem “Quy định về liêm chính học thuật” ban hành kèm theo Quyết định số 29/QĐ-ĐHBK ngày 09/01/2017 và “Hướng dẫn trích dẫn và lập Danh mục tài liệu tham khảo” được ban hành theo văn bản số 30/HD-ĐHBK ngày 09/01/2017 để thực hiện trích dẫn và lập Danh mục tài liệu tham khảo.