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

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CNTT**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CNTT**

**CHUYÊN NGÀNH: CNPM**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG QUẢN LÍ, GIÁM SÁT TIẾN ĐỘ CÔNG VIỆC VÀ DỰ ÁN TẠI CÔNG TY TNHH SIOUX**

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 CNTT**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CNTT**

**CHUYÊN NGÀNH: CNPM**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG QUẢN LÍ, GIÁM SÁT TIẾN ĐỘ CÔNG VIỆC VÀ DỰ ÁN TẠI CÔNG TY TNHH SIOUX**

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: ……………………………………………………………………………….

Sinh viên thực hiện: ………………………………….. …………………………………

Số thẻ SV: ……………………… Lớp: …………………………………………………

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

………………………………………………………………………………………… ..

|  |  |
| --- | --- |
| 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 TNHH Sioux

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 tháng năm 2018* |
|  | **Người hướng dẫn** |

|  |  |
| --- | --- |
| 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 TNHH Sioux

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:* …………………………………..……………………
2. *Ngày giao nhiệm vụ đồ án:*  *……../……./201…..*
3. *Ngày hoàn thành đồ án: ……../……./201…..*

|  |  |
| --- | --- |
|  | *Đà Nẵng, ngày tháng năm 201* |
| **Trưởng Bộ môn** …………………….. | **Người hướng dẫn** |

**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 Tiến Sĩ 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.

Chúng em xin chân thành cảm ơn.

*Đà Nẵng, 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 chúng 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, chúng em xin chịu hoàn toàn trách nhiệm.

Sinh viên thực hiện

**Trần Ngọc Quốc**

**MỤC LỤC**

[**TÓM TẮT 3**](#_Toc26385254)

[**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 5**](#_Toc26385255)

[**LỜI NÓI ĐẦU i**](#_Toc26385256)

[**CAM ĐOAN ii**](#_Toc26385257)

[**MỤC LỤC iii**](#_Toc26385258)

[**DANH SÁCH CÁC BẢNG, HÌNH VẼ v**](#_Toc26385259)

[**DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT vi**](#_Toc26385260)

[**MỞ ĐẦU 1**](#_Toc26385261)

[**1. Tổng quan về đề tài 1**](#_Toc26385262)

[**2. Mục đích và nhiệm vụ chính của đề tài 2**](#_Toc26385263)

[**2.1. Mục đích đề tài: 2**](#_Toc26385264)

[**2.2. Nhiệm vụ chính của đề tài: 2**](#_Toc26385265)

[**3. Công nghệ sử dụng 2**](#_Toc26385266)

[**4. Công cụ hỗ trợ 2**](#_Toc26385267)

[**5. Cấu trúc đồ án 2**](#_Toc26385268)

[**Chương 1: CƠ SỞ LÝ THUYẾT 4**](#_Toc26385269)

[**1.1. React Js 4**](#_Toc26385270)

[**1.1.1. Giới thiệu 4**](#_Toc26385271)

[**1.1.2. Virtual DOM 4**](#_Toc26385272)

[**1.1.3. One-way data binding 5**](#_Toc26385273)

[**1.1.4. Giới thiệu về JSX 5**](#_Toc26385274)

[**1.1.5. Giới thiệu về Components 5**](#_Toc26385275)

[**1.1.6. Props và State 6**](#_Toc26385276)

[**1.2. React Native 6**](#_Toc26385277)

[**1.2.1. Giới thiệu 6**](#_Toc26385278)

[**1.2.2. Ưu điểm 6**](#_Toc26385279)

[**1.3. Redux-Saga 7**](#_Toc26385280)

[**1.3.1. Giới thiệu 7**](#_Toc26385281)

[**1.3.2. Side effect 7**](#_Toc26385282)

[**1.3.3. Generator function 7**](#_Toc26385283)

[**1.3.4. Cách hoạt động 7**](#_Toc26385284)

[**1.4. WEB API 8**](#_Toc26385285)

[**1.4.1. ASP.NET CORE 8**](#_Toc26385286)

[**1.4.2. Phương thức HTTP 8**](#_Toc26385287)

[**Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 10**](#_Toc26385288)

[**2.1. Cách chức năng chính 10**](#_Toc26385289)

[**2.1.1. Quản lí thông tin nhân viên 10**](#_Toc26385290)

[**2.1.2. Quản lí dự án 10**](#_Toc26385291)

[**2.1.3. Quản lí thời gian làm việc 11**](#_Toc26385292)

[**2.2. Luồng dữ liệu 11**](#_Toc26385293)

[**2.3. Sơ đồ quan hệ 12**](#_Toc26385294)

[**2.4. Thiết kế cơ sở dữ liệu 13**](#_Toc26385295)

**DANH SÁCH CÁC HÌNH VẼ**

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

|  |  |
| --- | --- |
| API | Application Programming Interface |
| BSD | Berkeley Software Distribution |
| CUDA | Compute Unified Device Architecture |
| DHCP | Dynamic Host Configuration Protocol |
| DNN | Deep Neural Networks |
| FTP | File Transfer Protocol |
| FCM | Firebase Cloud Messaging |
| GPU | Graphics Processing Unit |
| HOG | Histogram of Oriented Gradients |
| HTTP | Hypertext Transfer Protocol |
| IEEE | Institute of Electrical and Electronics Engineers |
| IP | Internet Protocol |
| ISP | Internet Service Provider |
| JPEG | Joint Photographic Experts Group |
| LAN | *Local Area Network* |
| MHI | Motion History Image |
| MMX | Multimedia Extensions |
| MOG | Mixture Of Gaussians |
| MPEG | Moving Picture Experts Group |
| NAT | *Network Address Translation* |
| NIC | Network Information Center |
| PAT | *Port Address Translation* |

**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 Sioux 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 2 nhiệm vụ chính:

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

## **Công nghệ sử dụng**

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

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

## **Công cụ hỗ trợ**

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

## **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..

**Chương 1: CƠ SỞ LÝ THUYẾT**

## **React Js**

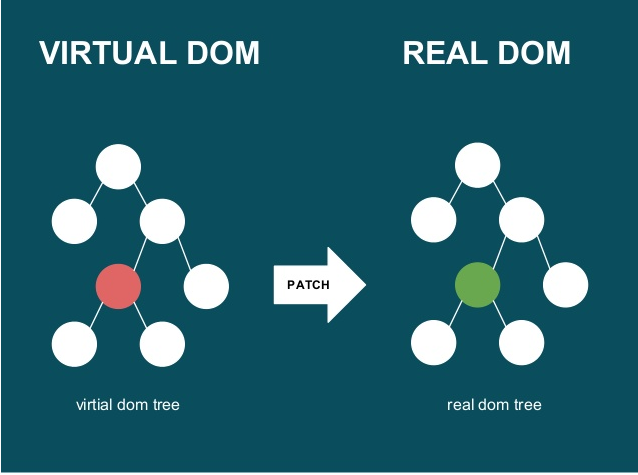
### ***Giới thiệu***

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 www.instagram.com 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***

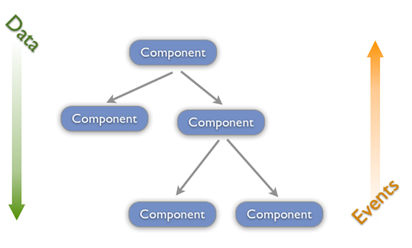
Công nghệ DOM ảo giúp tăng hiệu năng cho ứng dụng. Việc chỉ node gốc mới có trạng thái và khi nó thay đổi sẽ tái cấu trúc lại toàn bộ, đồng nghĩa với việc DOM tree cũng sẽ phải thay đổi một phần, điều này sẽ ảnh hưởng đến tốc độ xử lý. React JS sử dụng Virtual DOM (DOM ảo) để cải thiện vấn đề này.Virtual 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 . 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 . 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.

## **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***

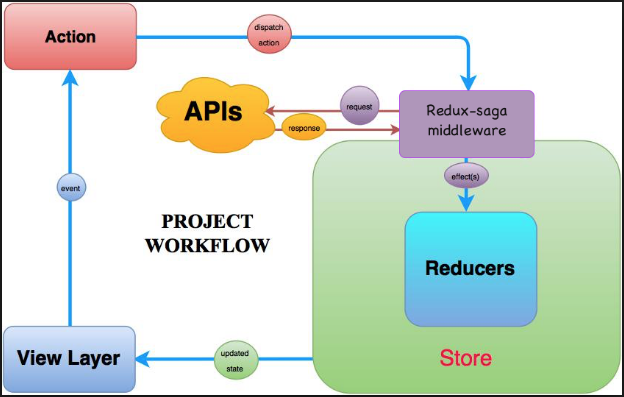
Ta đã biết tất cả những xử lý ở REDUCER đều phải là synchronous và pure tức chỉ là xử lý đồng bộ. Nhưng trong ứng dụng thực tế thì cần nhiều hơn vậy ví dụ như asynchronous (thực hiện một số việc như gọi một hàm AJAX để fetch dữ liệu về nhưng cần đợi kết quả chứ kết quả không trả về ngay được) hoặc là 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 . 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)

### ***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.

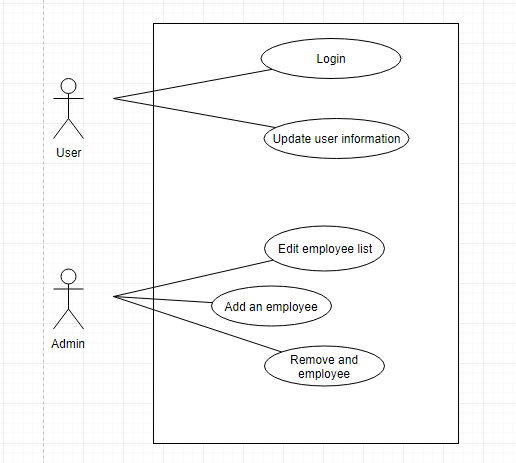
**Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

## **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

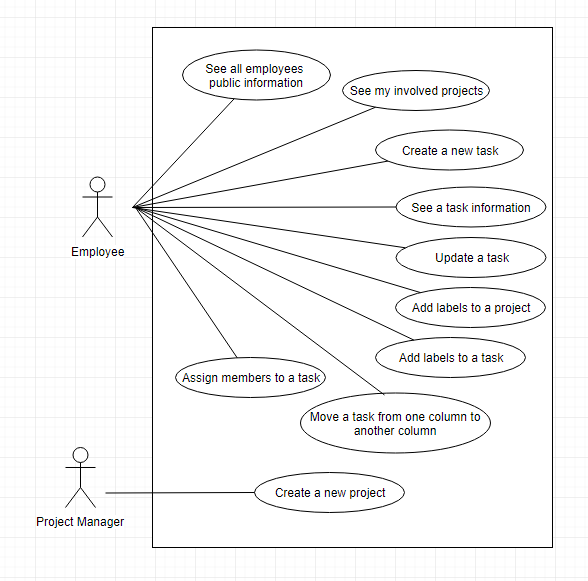


*Hình 2.1: Sơ đồ use-case cho module quản lí nhân viên*

### ***Quản lí dự án***

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

* Trưởng nhóm có thể tạo mới và quản lí một quy trình làm việc.
* Trưởng nhóm có thể thêm các thành viên khác vào quy trình làm việc
* 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 ước lượng thời gian hoàn thành một công việc/ nhiệm vụ.
* Thành viên có thể ghi lại thời gian thực tế đã dành cho công việc được giao.
* 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.
* Giám đốc / trưởng bộ phận có thể quản lí, giám sát toàn bộ dự án của công ty
* Nhân viên có thể nhận các thông báo về công việc đã được giao qua smart phone
* Nhân viên có thể cập nhật tiến độ công việc được giao qua app.

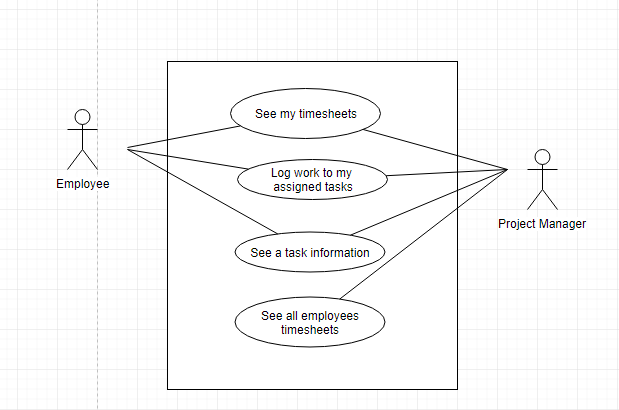


*Hình 2.2: Sơ đồ use-case cho module quản lí dự án*

### ***Quản lí thời gian làm việc***

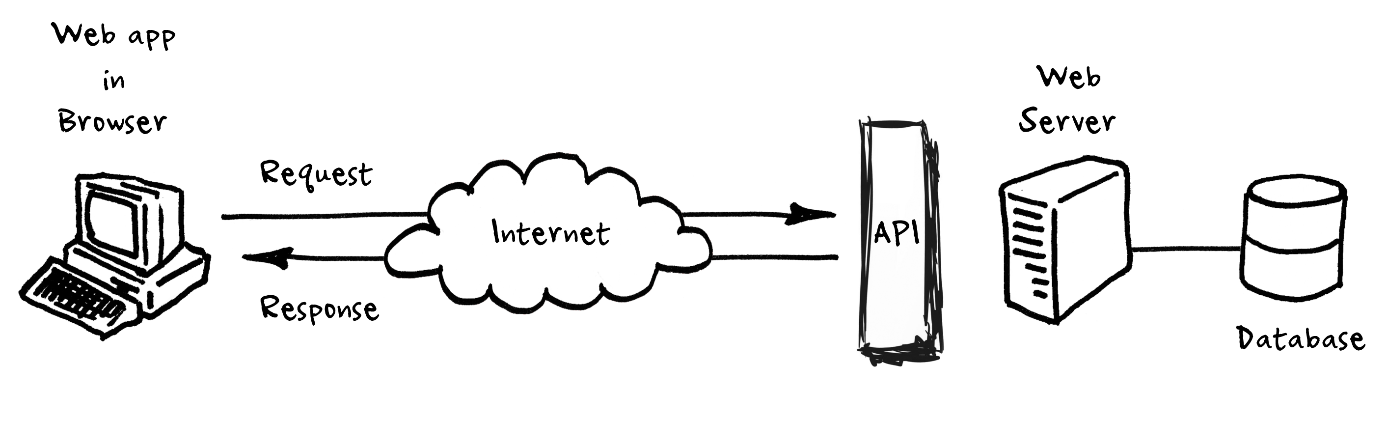
Để thuận lợi hơn trong việc đánh giá hiệu suất làm việc của nhân viên, hệ thống sẽ cho phép:

* 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.
* Giám đốc / trưởng bộ phận có thể quan sát thời gian làm việc của nhân viên.



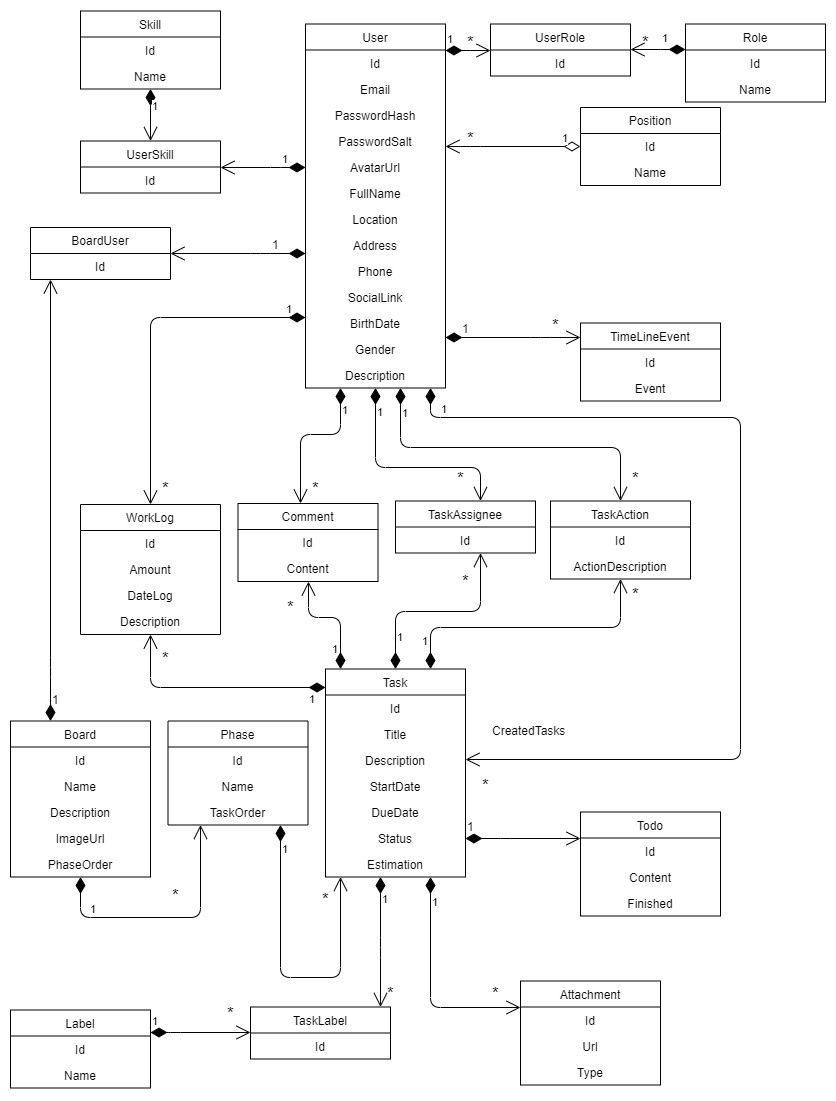
*Hình 2.3: Sơ đồ use-case module quản lí thời gian làm việc*

## **Luồng dữ liệu**

*Hình 2.4 Mô tả cách thức hoạt động của Redux - Saga*

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 2.5: Sơ đồ lớp cho dự án*

## **Thiết kế cơ sở dữ liệu**

****

*Hình 2.6: Sơ đồ quan hệ, thực thể dữ liệu*

**CHƯƠNG 3: 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

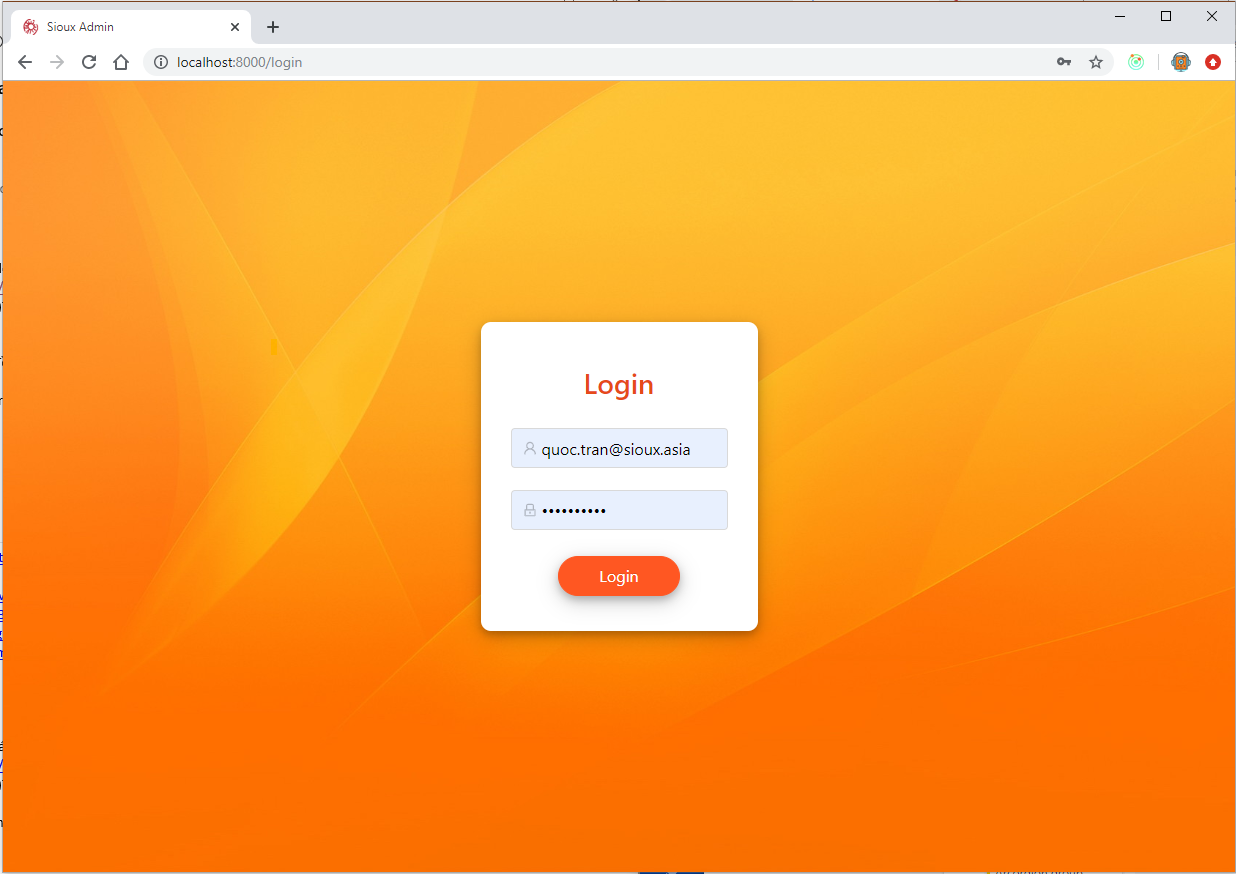
*Web Client:*

* 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, …

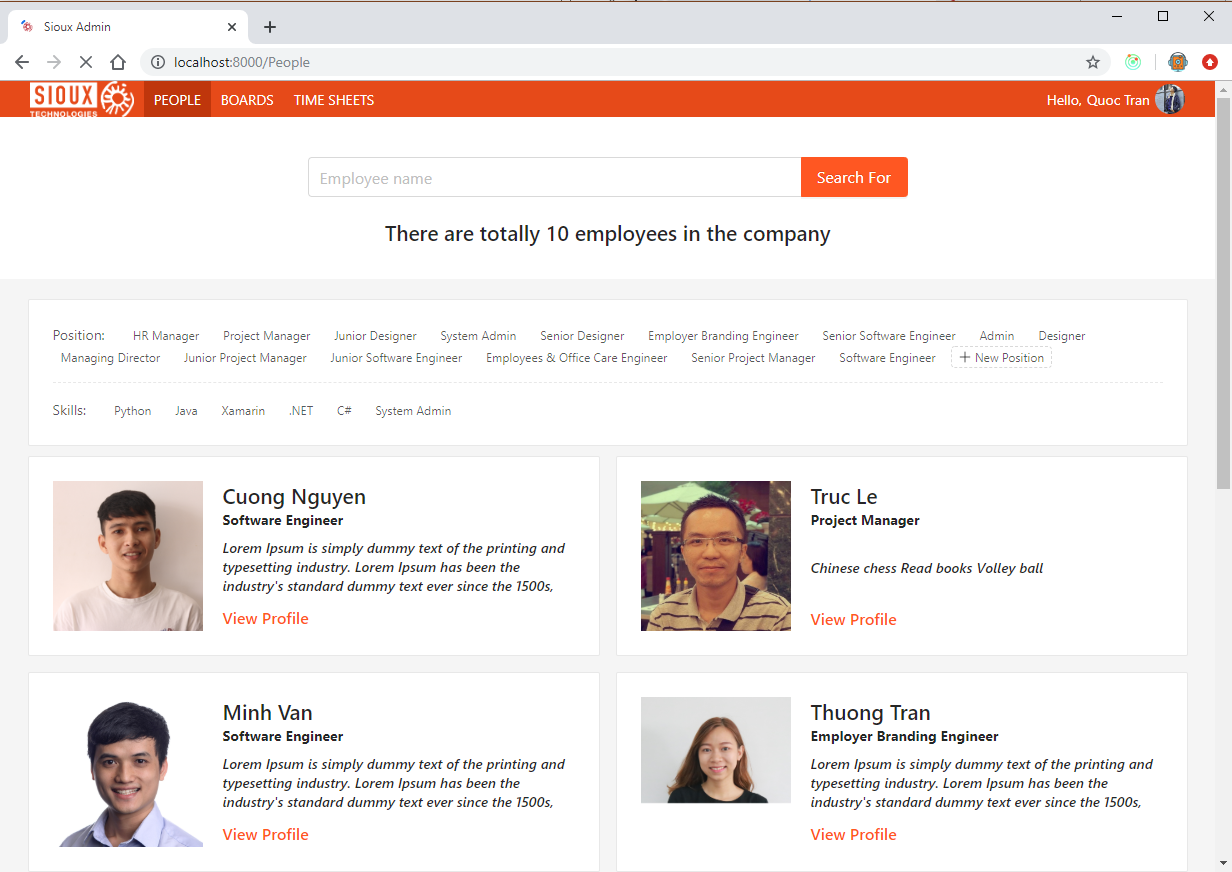
*Môi trường phát triển:*

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

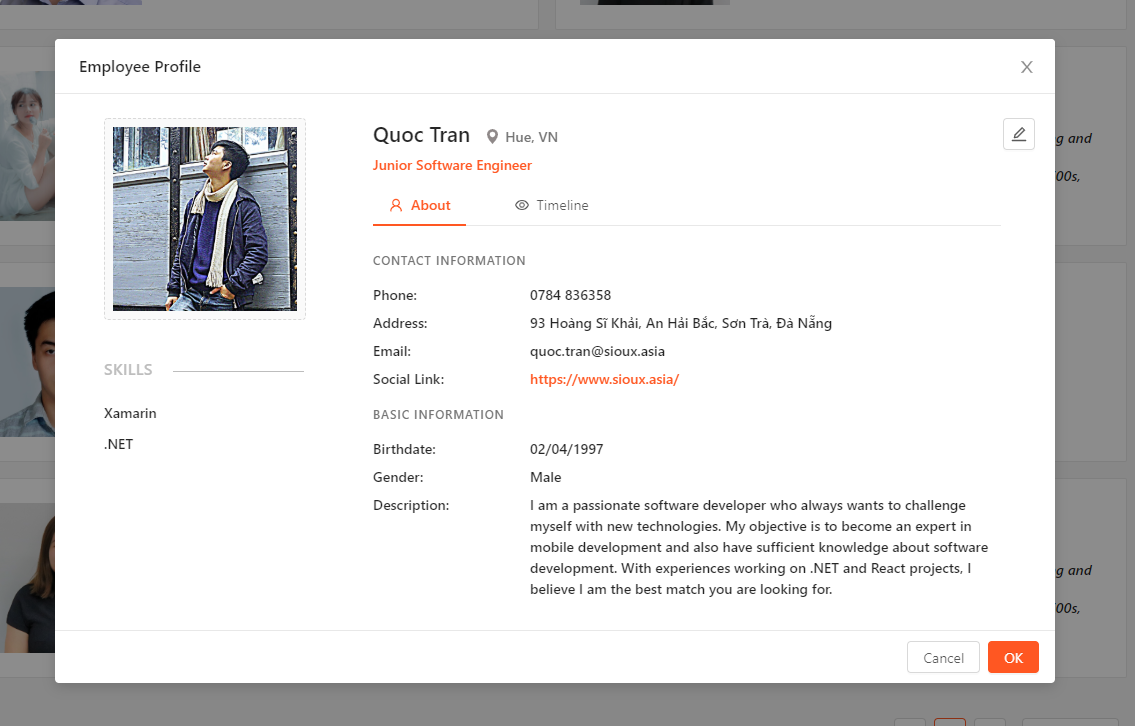
## **3.2. Kết quả thực hiện**



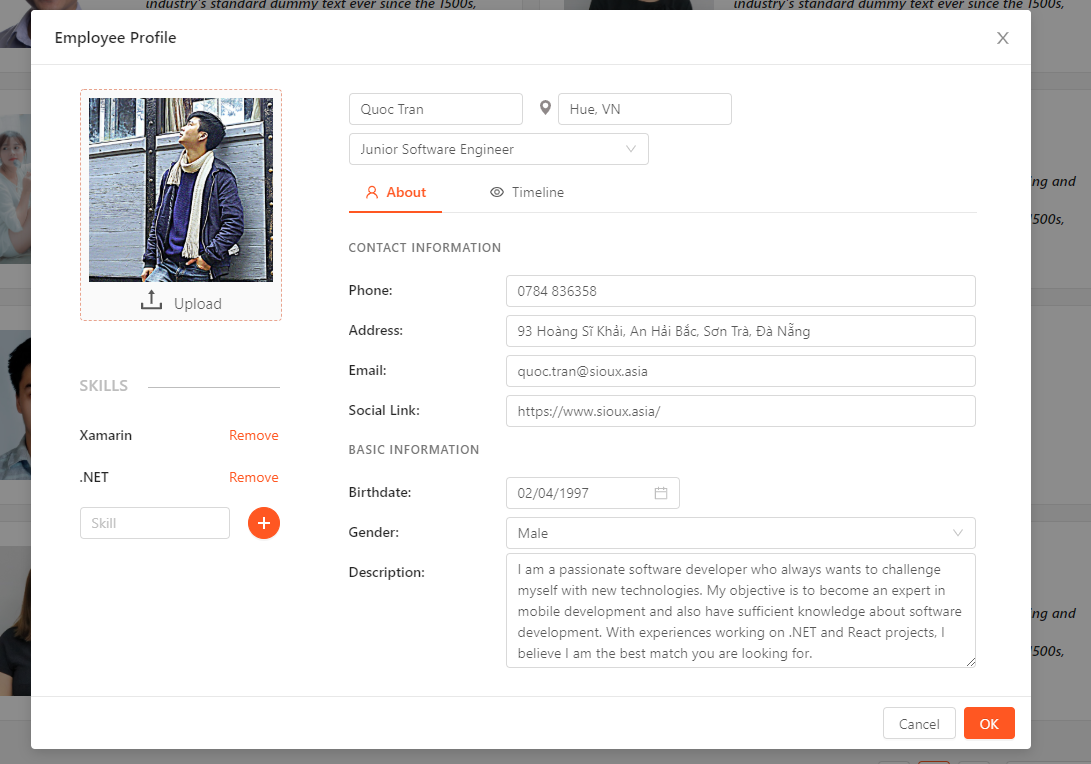
*Hình 3.1: Màn hình Login*



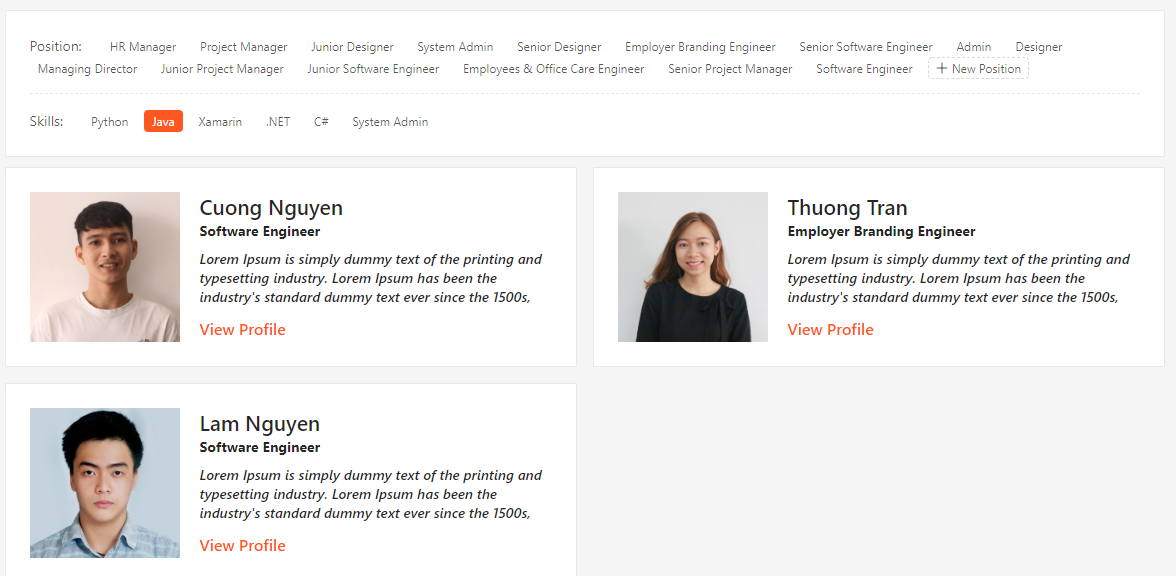
*Hình 3.2 Quản lí nhân viên*

****

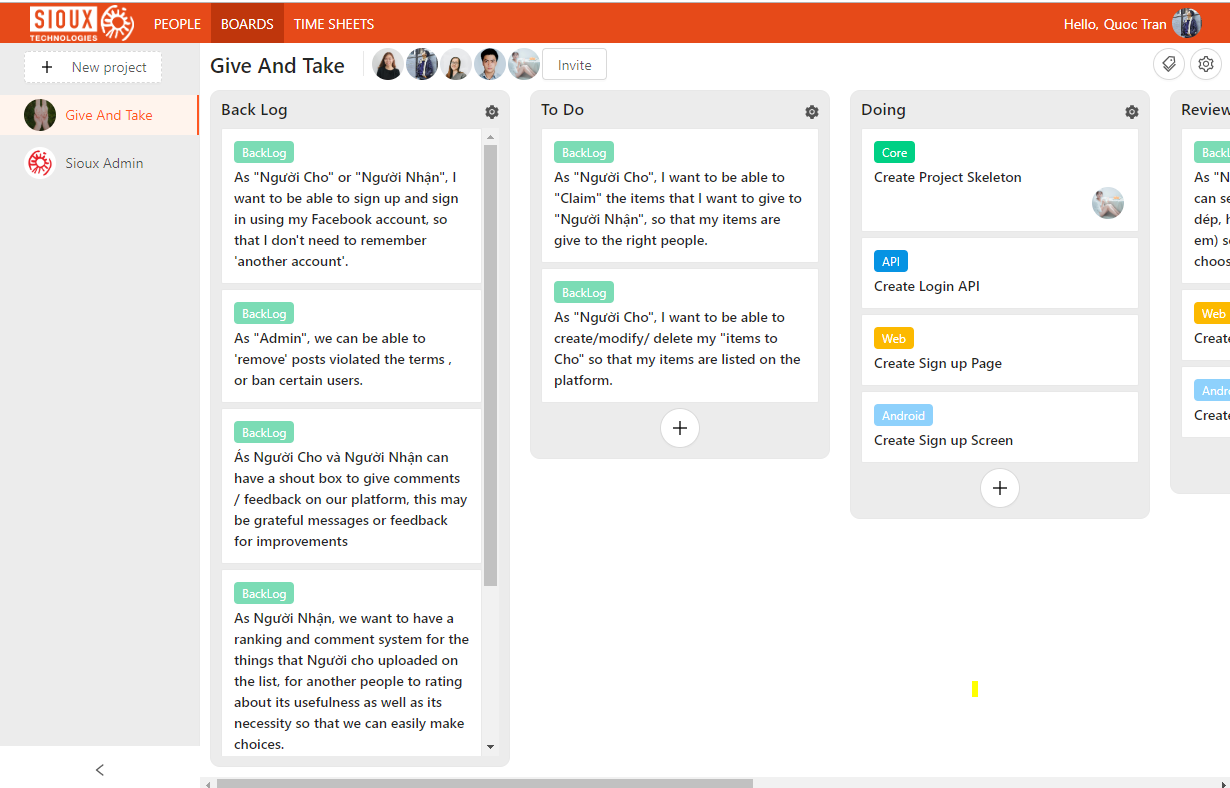
*Hình 3.3 Xem thông tin cá nhân của nhân viên*

****

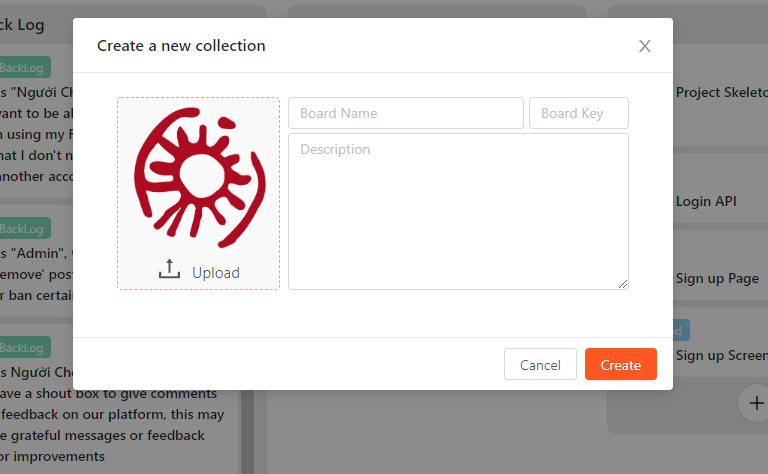
*Hình 3.4 Chỉnh sửa thông tin cá nhân*

****

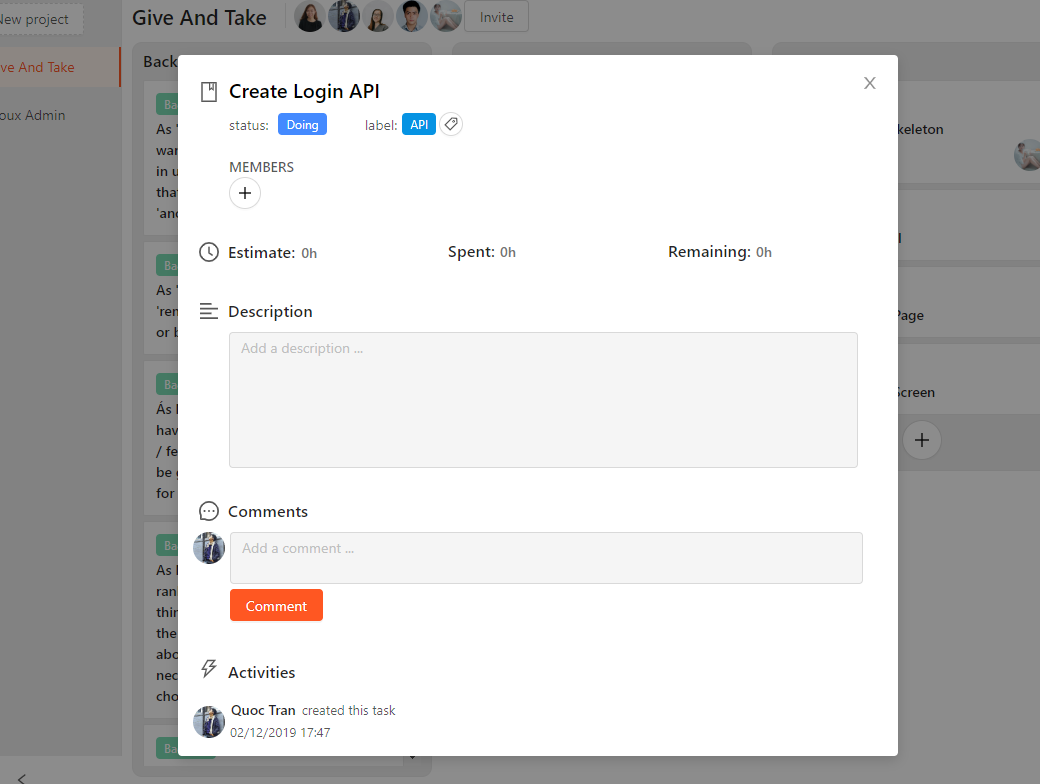
*Hình 3.5 Lọc và tìm kiếm nhân viên dựa trên kĩ năng, vị trí, tên, …*

****

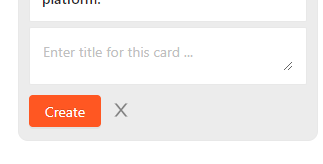
*Hình 3.6 Màn hình quản lí các dự án*

****

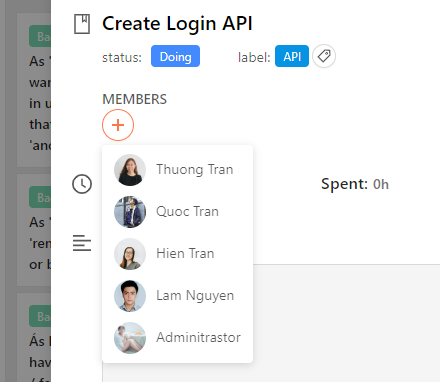
*Hình 3.7 Tạo một dự án mới*

****

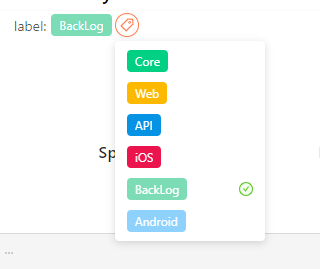
*Hình 3.8 thông tin chi tiết của một công việc*

****

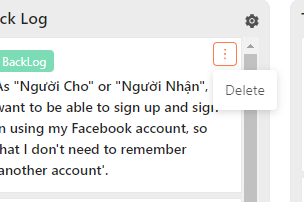
*Hình 3.9 Tạo một công việc mới*

****

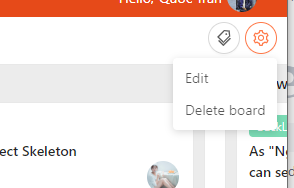
*Hình 3.10 Giao việc cho thành viên trong dự án*

****

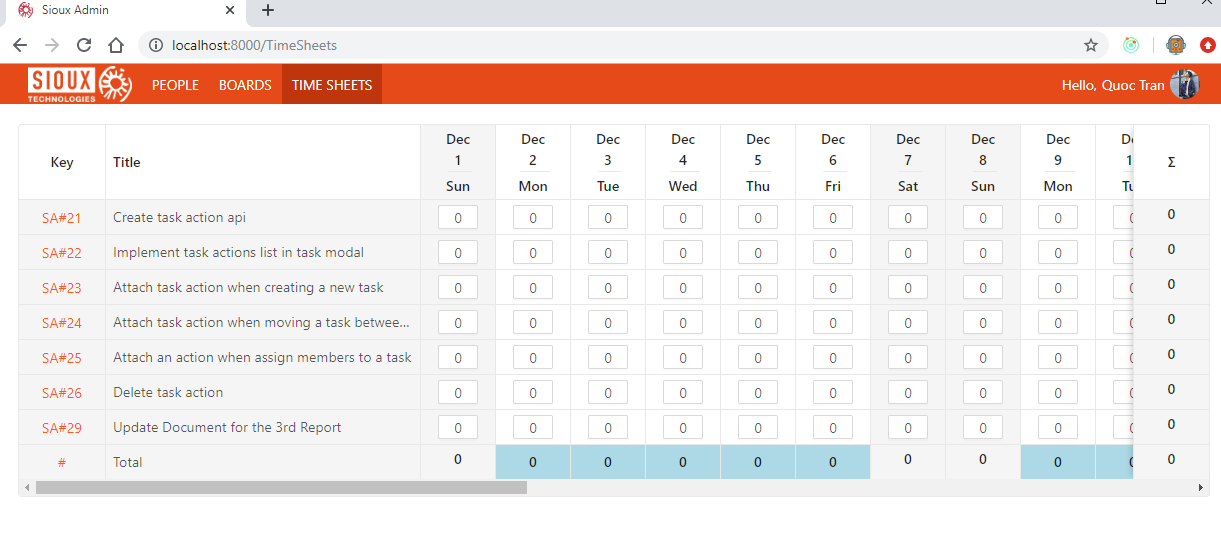
*Hình 3.11 Thêm nhãn cho công việc để dễ quản lí*

****

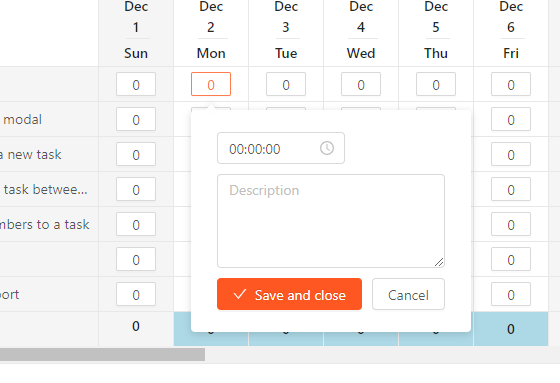
*Hình 3.12 Xóa một công việc*

****

*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 {size 14**}

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**

**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.

**PHỤ LỤC 1**

**PHỤ LỤC 2**