**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**Vi Thế Anh**

**PHÁT TRIỂN CÔNG CỤ HỖ TRỢ QUẢN LÝ DỰ ÁN VÀ CÔNG VIỆC NỘI BỘ**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Công nghệ thông tin**

**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**VI THẾ ANH**

**PHÁT TRIỂN CÔNG CỤ HỖ TRỢ QUẢN LÝ DỰ ÁN VÀ CÔNG VIỆC NỘI BỘ**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành:** **Công nghệ thông tin**

**Cán bộ hướng dẫn:** **TS Dương Lê Minh**

**HÀ NỘI - 2021**

# LỜI CẢM ƠN

Lời đầu tiên, tôi xin gửi lời cảm ơn chân thành đến thầy giáo TS Dương Lê Minh là người đã chỉ bảo và giúp đỡ tôi tận tình trong quá trình định hướng nghiên cứu, chọn đề tài và quá trình thực hiện khóa luận này. Tôi xin gửi lời cảm ơn tới các thầy cô Trường Đại học Công Nghệ - Đại học Quốc Gia Hà Nội – những người đã dạy bảo và truyền đạt cho tôi rất nhiều kiến thức, giúp tôi có một nền tảng vững chắc sau 4 năm học tập tại Đại học Công Nghệ. Đồng thời, tôi xin cảm ơn công ty Teneocto, là bên khách hàng trực tiếp đưa tới các yêu cầu. Trong quá trình làm việc, bên phía công ty Teneocto đã có các trao đổi, góp ý cũng như chỉnh sửa những điều tôi còn hiểu chưa đúng, giúp tôi hoàn thiện các chức năng của hệ thống và cho tôi những trải nghiệm quý báu trong quá trình làm việc với khách hàng

Tôi cũng xin gửi lời cảm ơn tới các anh chị và bạn bè cùng học tập và nghiên cứu tại trường Đại học Công Nghệ đã hỗ trợ tôi rất nhiều trong quá trình học tập cũng như thực hiện khóa luận tốt nghiệp này.

Tôi xin chân thành cảm ơn!

# LỜI CAM ĐOAN

Tôi xin cam đoan toàn bộ khóa luận “Phát triển công nghệ hỗ trợ quản lý dự án và công việc nội bộ” bao gồm nội dung và ứng dụng thực tiễn là hoàn toàn được xây dựng, thực hiện một cách trung thực, chưa từng xuất hiện trên tài liệu nào tương tự. Mọi nguồn tham khảo đều được trích dẫn rõ ràng và hợp pháp.

Tôi xin chịu hoàn toàn trách nghiệm cho lời cam đoan của mình.

|  |  |
| --- | --- |
|  | Hà Nội, ngày …. tháng 05 năm 2021  Sinhviên  **Vi Thế Anh** |

# TÓM TẮT

Trong những năm trở lại đây, nhu cầu về quản lý khối lượng công việc của các cá nhân và tổ chức ngày càng lớn. Xu hướng sử dụng các phần mềm trực tuyến để quản lý công việc nhằm tiết kiệm thời gian mang lại hiệu quả cao đang ngày càng được chú trọng. Nắm bắt được nhu cầu đó, đã có nhiều trang web ra đời như: Trello, Jira, Backlog,.. đem lại lợi thế về tốc độ, năng xuất vượt trội cho các doanh nghiệp, và là trợ thủ đắc lực cho các nhà quản lý hiện đại. Nhưng để các doanh nghiệp lựa chọn được một giải pháp hữu ích phù hợp thì còn phải cân nhắc nhiều yếu tố. Các doanh nghiệp sẽ rất vất vả khi phải quản lý dữ liệu ngày càng mở rộng được tạo ra từ nhân sự, khách hàng và nhà cung cấp, nhất là với những doanh nghiệp có tốc độ phát triển nhanh. Đối với các cá nhân, nhiệm vụ chính của họ vẫn là tham gia hoàn thành công việc được giao nên việc dành thời gian quá nhiều tại các công cụ quản lý là không cần thiết. Vậy câu hỏi đặt ra là làm sao để có một hệ thống tích hợp giúp cho doanh nghiệp và cá nhân có một quy trình liền mạch từ đó giảm thiểu thời gian và đem lại hiệu quả cao. Nắm bắt được những vấn đề này, khoá luận của tôi hướng đến việc tích hợp các hệ thống quản lý công việc và dự án lại với nhau đem lại hiệu quả cao, đồng thời giảm sự gián đoạn gây ra bởi việc quản lý dữ liệu ở các nơi khác nhau. Hệ thống được xây dựng và phát triển trên các nền tảng hỗ trợ phát triển websites phổ biến bậc nhất hiện nay đó là React và NodeJS đảm bảo tính dễ tiếp cận và mở rộng đối với đội ngũ phát triển. Nội dung của khóa luận sẽ tập trung trình bày về cách xây dựng từng hệ thống, những công nghệ sử dụng cũng như những vấn đề liên quan tới việc xây dựng và phát triển hệ thống.

***Từ khóa:*** *React, NodeJS, quản lý công việc, quản lý dự án.*

MỤC LỤC

[LỜI CẢM ƠN i](#_30j0zll)

[LỜI CAM ĐOAN ii](#_1fob9te)

[TÓM TẮT iii](#_3znysh7)

[MỤC LỤC 1](#_2et92p0)

[Chương 1.](#_3dy6vkm) GIỚI THIỆU 6

[1.](#_1t3h5sf) Đặt vấn đề 6

[2.](#_4d34og8) Mục tiêu của khóa luận 7

[**3.**](#_2s8eyo1) **Cấu trúc khóa luận** 7

[Chương 2.](#_17dp8vu) CƠ SỞ LÝ THUYẾT 9

[2.1.](#_3rdcrjn) Giới thiệu ReactJS 9

[2.2.](#_26in1rg) Giới thiệu NodeJS 10

[2.3.](#_lnxbz9) Giới thiệu Firebase 12

[2.3.1.](#_1ksv4uv) Firebase Authentication 13

[2.3.2. Firebase Cloud Firestore 14](#_2jxsxqh)

[2.4.](#_z337ya) Giới thiệu Jira software RestFul API 16

[2.5.](#_3j2qqm3) Giới thiệu Google services API 16

[2.5.1.](#_1y810tw) Google Oauth 2.0 16

[2.5.2. Google Drive API 19](#_2xcytpi)

[2.5.3.](#_3whwml4) Google Calendar API 20

[Chương 3.](#_2bn6wsx) PHÂN TÍCH VÀ ĐẶC TẢ YÊU CẦU 21

[3.1.](#_qsh70q) Xác định nhóm các đối tượng sử dụng trong hệ thống 21

[3.2.](#_3as4poj) Phân tích yêu cầu 21

[3.2.1.](#_1pxezwc) Yêu cầu chức năng 21

[3.2.2.](#_49x2ik5) Yêu cầu phi chức năng 22

[3.3.](#_2p2csry) Xây dựng biểu đồ sơ lược ca sử dụng 23

[3.3.1.](#_147n2zr) Xác định tác nhân và các ca sử dụng 23

[3.3.2. Biểu đồ ca sử dụng 25](#_23ckvvd)

[3.4. Đặc tả chi tiết từng ca sử dụng 30](#_2grqrue)

[3.4.1. Ca sử dụng “Đăng nhập” 30](#_vx1227)

[3.4.2. Ca sử dụng “Thay đổi thông tin người dùng” 31](#_3fwokq0)

[3.4.3.](#_1v1yuxt) Ca sử dụng “Xóa người dùng” 32

[3.4.4.](#_4f1mdlm) Ca sử dụng “Thêm mới người dùng” 32

[3.4.5.](#_2u6wntf) Ca sử dụng “Thêm mới báo cáo” 33

[3.4.6.](#_19c6y18) Ca sử dụng “Xem danh sách báo cáo” 34

[3.4.7.](#_3tbugp1) Ca sử dụng “Sửa báo cáo” 35

[3.4.8.](#_28h4qwu) Ca sử dụng “Đánh giá công việc” 36

[3.4.9.](#_nmf14n) Ca sử dụng “Export báo cáo” 37

[3.4.10. Ca sử dụng “Quản lý báo cáo của nhân viên” 38](#_37m2jsg)

[3.4.11. Ca sử dụng “Tạo đơn xin nghỉ” 39](#_1mrcu09)

[3.4.12. Ca sử dụng “Phản hồi đơn xin nghỉ” 40](#_46r0co2)

[3.4.13. Ca sử dụng “Xem báo cáo từ trang Jira software” 40](#_2lwamvv)

[3.4.14. Ca sử dụng “Chỉ định báo cáo từ Jira về hệ thống" 42](#_111kx3o)

[3.4.15. Ca sử dụng “Thêm mới dự án” 43](#_3l18frh)

[3.4.16. Ca sử dụng “Đóng/kích hoạt dự án” 43](#_206ipza)

[3.4.17. Ca sử dụng “Xem thống kê báo cáo” 44](#_4k668n3)

[3.4.18. Ca sử dụng “Quản lý file của người dùng” 45](#_2zbgiuw)

[3.5.](#_1egqt2p) Xây dựng biểu đồ tuần tự 46

[3.5.1. Biểu đồ tuần tự ca sử dụng “Đăng nhập” 47](#_3ygebqi)

[3.5.2. Biểu đồ tuần tự ca sử dụng “Thay đổi thông tin người dùng” 48](#_sqyw64)

[3.5.3.](#_1rvwp1q) Biểu đồ tuần tự ca sử dụng “Thêm tài khoản” 49

[3.5.4.](#_2r0uhxc) Biểu đồ tuần tự ca sử dụng “Thêm mới báo cáo” 50

[3.5.5.](#_3q5sasy) Biểu đồ tuần tự ca sử dụng “Xem danh sách báo cáo” 51

[3.5.6.](#_kgcv8k) Biểu đồ tuần tự ca sử dụng “Sửa báo cáo” 52

[3.5.7.](#_1jlao46) Biểu đồ tuần tự ca sử dụng “Đánh giá công việc” 53

[3.5.8.](#_2iq8gzs) Biểu đồ tuần tự ca sử dụng “Quản lý báo cáo của nhân viên” 54

[3.5.9.](#_3hv69ve) Biểu đồ tuần tự ca sử dụng “Tạo đơn xin nghỉ” 55

[3.5.10.  Biểu đồ tuần tự ca sử dụng “Phản hồi đơn xin nghỉ” 56](#_4h042r0)

[3.5.11.  Biều đồ tuần tự ca sử dụng “Xem báo cáo từ Jira software” 57](#_1baon6m)

[3.5.12.  Biểu đồ tuần tự ca sử dụng “Chỉ định báo cáo từ Jira về hệ thống” 58](#_2afmg28)

[3.5.13.  Biểu đồ tuần tự ca sử dụng “Đóng/kích hoạt dự án” 59](#_39kk8xu)

[3.5.14.  Biểu đồ tuần tự ca sử dụng “Quản lý file drive” 60](#_48pi1tg)

[Chương 4.](#_1302m92) THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG 62

[4.1.](#_3mzq4wv) Thiết kế kiến trúc 62

[4.2.](#_haapch) Thiết kế cơ sở dữ liệu 63

[4.3. Cài đặt hệ thống 68](#_4du1wux)

[4.3.1. Xây dựng cấu trúc hệ thống 69](#_2szc72q)

[Chương 5.](#_3s49zyc) TRIỂN KHAI THỰC TẾ 72

[5.1. Môi trường triển khai 72](#_279ka65)

[5.2 Kết quả triển khai 72](#_36ei31r)

[CHƯƠNG 6. KẾT LUẬN 79](#_2y3w247)

[6.1. Kết quả đạt được 79](#_1d96cc0)

[6.2. Định hướng phát triển 79](#_2ce457m)

**Danh mục hình ảnh**

[Hình 2.1 Mô hình Firebase RESTFUL API 13](#_35nkun2)

[Hình 2.2 Các ứng dụng phổ biến firebase hỗ trợ đăng nhập 13](#_44sinio)

[Hình 2.3 Sơ đồ sử dụng Oauth 2.0 cho máy chủ Web 18](#_4i7ojhp)

[Hình 2.4 Sơ đồ sử dụng Google Drive API 19](#_1ci93xb)

[Hình 3.1 Biểu đồ ca sử dụng “Develop” 26](#_ihv636)

[Hình 3.2 Biểu đồ ca sử dụng “Admin” 27](#_32hioqz)

[Hình 3.3 Biểu đồ ca sử dụng “Leader” 28](#_1hmsyys)

[Hình 3.4. Biểu đồ ca sử dụng “Manager” 29](#_41mghml)

[Hình 3.5 Sơ đồ tuần tự ca sử dụng “Đăng nhập” 47](#_2dlolyb)

[Hình 3.6 Sơ đồ tuần tự ca sử dụng “Thay đổi thông tin người dùng” 48](#_3cqmetx)

[Hình 3.7 Sơ đồ tuần tự ca sử dụng “Thêm tài khoản” 49](#_4bvk7pj)

[Hình 3.8 Sơ đồ tuần tự ca sử dụng “Thêm báo cáo mới” 50](#_1664s55)

[Hình 3.9 Sơ đồ tuần tự ca sử dụng “Xem danh sách báo cáo” 51](#_25b2l0r)

[Hình 3.10 Sơ đồ tuần tự ca sử dụng “Sửa báo cáo” 52](#_34g0dwd)

[Hình 3.11 Sơ đồ tuần tự ca sử dụng “Đánh giá công việc” 53](#_43ky6rz)

[Hình 3.12 Sơ đồ tuần tự ca sử dụng “Quản lý báo cáo của nhân viên” 54](#_xvir7l)

[Hình 3.13 Sơ đồ tuần tự ca sử dụng “Tạo đơn xin nghỉ” 55](#_1x0gk37)

[Hình 3.14 Sơ đồ tuần tự ca sử dụng “Phản hồi đơn xin nghỉ” 56](#_2w5ecyt)

[Hình 3.15 Sơ đồ tuần tự ca sử dụng “Xem báo cáo từ Jira software” 57](#_3vac5uf)

[Hình 3.16 Sơ đồ tuần tự ca sử dụng “Chỉ định báo cáo từ Jira software về hệ thống” 58](#_pkwqa1)

[Hình 3.17 Sơ đồ tuần tự ca sử dụng “Đóng/kích hoạt dự án” 59](#_1opuj5n)

[Hình 3.18 Sơ đồ tuần tự ca sử dụng “Quản lý file Drive” 60](#_2nusc19)

[Hình 4.1 Mô tả thiết kế hệ thống 62](#_2250f4o)

[Hình 4.2 Cấu trúc cơ sở dữ liệu 63](#_319y80a)

[Hình 4.3 Cấu trúc thư mục ứng dụng 69](#_184mhaj)

[Hình 5.1 Giao diện màn hình đăng nhập 72](#_1ljsd9k)

[Hình 5.2 Giao diện danh sách báo cáo công việc 73](#_45jfvxd)

[Hình 5.3 Giao diện thêm mới báo cáo công việc 74](#_2koq656)

[Hình 5.4 Giao diện quản lý người dùng 75](#_zu0gcz)

[Hình 5.5 Giao diện quản lý file Drive 76](#_3jtnz0s)

[Hình 5.6 Giao diện danh sách dự án từ Jira 77](#_1yyy98l)

[Hình 5.7 Giao diện danh sách báo cáo từ Jira 78](#_4iylrwe)

**Danh mục bảng biểu**

[Bảng 3.1 Bảng chức tác nhân và ca sử dụng 25](#_3o7alnk)

[Bảng 4.1 Danh sách các bảng trong cơ sở dữ liệu 64](#_1gf8i83)

[Bảng 4.2 Bảng cơ sở dữ liệu người dùng 65](#_40ew0vw)

[Bảng 4.3 Bảng cơ sở dữ liệu dự án 65](#_2fk6b3p)

[Bảng 4.4 Bảng cơ sở dữ liệu báo cáo công việc 67](#_upglbi)

[Bảng 4.5 Bảng cơ sở dữ liệu đánh giá nhân viên 67](#_3ep43zb)

[Bảng 4.6 Bảng cơ sở dữ liệu xin nghỉ 68](#_1tuee74)

# GIỚI THIỆU

## Đặt vấn đề

Cách quản lý công việc của một cá nhân, một tổ chức là một trong những yếu tố quyết định sự thành công hay thất bại của cá nhân, tổ chức đó. Đối với một cá nhân cụ thể, họ thường không thể sắp xếp được thời gian hoàn thành mục tiêu đã đề ra và bị mắc kẹt trong hàng “tấn” công việc nếu không lên kế hoạch một cách tỉ mỉ, có khoa học và tuân theo kỷ luật. Cũng tương tự đối với nhiều công ty, doanh nghiệp công tác quản lý công việc còn nhiều hạn chế, chưa phù hợp nhất là trong thời đại công nghệ thông tin ngày càng phổ biến và mạnh mẽ ngày nay, làm kìm hãm chính sự phát triển của doanh nghiệp đó. Đối với những con người hay tập thể có phương pháp quản lý công việc tốt, luôn kiểm soát được tiến độ và kế hoạch đã đề ra, nắm bắt nhanh chóng những thay đổi bất ngờ để từ đó đưa ra giải pháp kịp thời thì hiệu suất làm việc sẽ luôn hiệu quả hơn, cơ hội thành công cũng cao hơn. Ngược lại, nếu công việc không được quản lý chặt chẽ và tuân theo một cách có trách nhiệm, đây sẽ là vấn đề lớn cản trở đến năng suất làm việc, dần dần trở thành gánh nặng dẫn đến dậm chân tại chỗ thậm chí là thất bại.

Trước kia, khi phát sinh nhu cầu công việc mọi người thường trao đổi miệng với nhau, mở những cuộc họp nhanh để có thể giải quyết vấn đề ngay tức khắc. Tuy nhiên đến một khi nào đó, khối lượng công việc trở lên quá đồ sộ, người làm phải ưu tiên công việc này hay việc khác, khiến họ khó để chia sẻ với nhau, làm việc này quên việc khác, không thể quản lý 100% công việc. Điều đó khiến cho năng suất và hiệu quả không được như kỳ vọng. Doanh nghiệp gặp khó khăn trong việc quản lý công việc như khó khăn khi giao việc cho nhân viên, khó khăn trong việc kiểm soát tiến độ công việc, hoàn thành công việc và chất lượng công việc, thậm chí khó khăn ngay cả trong việc quản lý nguồn nhân sự của mình và đặc biệt là vấn đề năng suất công việc không ổn định. Về phía nhân viên, do doanh nghiệp gặp khó khăn trong việc quản lý, nhân viên cũng từ đó bị thụ động trong chính cách làm việc của tổ chức mình, không thể đối mặt với khối lượng công việc lớn vì không có cách quản lý phù hợp, từ đó ảnh hưởng đến chất lượng công việc của cá nhân nói riêng và cả doanh nghiệp nói chung. Có thể thấy rất nhiều vấn đề  đã được đặt ra khi đề cập đến việc quản lý công việc.  Vì vậy, muốn giải quyết vấn đề này một cách triệt để, việc sử dụng các phần mềm quản lý tiến độ dự án và công việc sẽ là phương pháp cực kỳ tuyệt vời.

Làm sao để quản lý công việc một cách hiệu quả và hợp lý đang trở thành những vấn đề mà tất cả các nhà lãnh đạo hiện nay quan tâm và tìm phương án xử lý. Với thời đại công nghệ, kỹ thuật lên ngôi như hiện nay, để có thể đáp ứng được yêu cầu của xã hội, mỗi cá nhân, tổ chức phải nâng cao hiệu xuất của bản thân mình. Phần mềm **“Hỗ trợ quản lý dự án và công việc nội bộ”** ra đời đem lại lợi thế tốc độ, năng xuất vượt trội cho doanh nghiệp và là trợ thủ đắc lực cho các nhà quản trị hiện đại. Nó sẽ là một bước ngoặt lớn và có ý nghĩa vô cùng quan trọng, thiết thực cho các tổ chức, doanh nghiệp trong tình thế ấy. Nếu khả năng quản lý, điều khiển hoạt động công việc cũng như dự án ngày càng tốt, chắc hẳn lợi nhuận công ty cũng được gia tăng vượt bậc. Các tính năng của phần mềm sẽ giải quyết các khâu trong quản lý như minh bạch hóa quá trình liên hệ, giao- nhận việc giữa nhân viên và người quản lý, lên kế hoạch cho tổ chức, công ty và nhân viên của họ.

Cần bổ sung thêm 1 mục về khảo sát các phần mềm tương tự. Với mỗi phần mềm, nên ưu/nhược điểm của nó.

Trong mục này, đoạn cuối cùng em cần nêu bật được yêu cầu cụ thể của công ty em và/hoặc các vấn đề mà em thấy là có thể xảy ra trong các công ty khác nữa.

## Mục tiêu của khóa luận

Bản chất của mục này là nêu các idea về sản phẩm sao cho trả lời được các yêu cầu em nêu ở mục trên nhé.

Mục tiêu của đề tài là thiết kế xây dựng trang web hỗ trợ quản lý dự án và công việc nội bộ. Nó kết hợp các tính năng của các công cụ mà doanh nghiệp của bạn đang cần sử dụng hiện nay. Điều này giúp cung cấp cho bạn một cái nhìn tổng quát toàn diện về công việc và nhân viên công ty của bạn, giữ mọi người trên cùng một trang và tiết kiệm thời gian, chi phí đăng ký khác nhau trong suốt quá trình.

***Đối với đội ngũ nhân viên:***

* Phần mềm quản lý giúp họ bao quát được tất cả công việc mình cần thực hiện trong từng ngày, tuần, tháng hoặc thậm chí là dài hạn hơn.
* Giúp họ nhận biết được đâu là những công việc quan trọng, cần phải ưu tiên hoàn thành trước.
* Là cơ sở để nhân viên có thể lên kế hoạch thời gian một cách cụ thể, chi tiết cho công việc của mình.
* Theo dõi lịch công việc được giao hàng ngày và yêu cầu xin nghỉ phép của cá nhân

***Đối với người quản lý:***

* Giúp người quản lý sắp xếp được các công việc, dự án của từng cá nhân, bộ phận trong công ty, doanh nghiệp.
* Có thể kịp thời ra quyết định, xử lý các vấn đề, sự cố xảy ra trong công việc.
* Là công cụ để phân chia nguồn lực thời gian, con người để hoàn thành các công việc đúng thời hạn, chất lượng.
* Dễ dàng quản lý, theo dõi ngày phép của nhân viên trong công ty một cách tập trung. Theo dõi trạng thái phê duyệt đơn và số ngày công trong tháng, số ngày nghỉ phép còn lại.

1. **Cấu trúc khóa luận**

Các chương còn lại của khóa luận sẽ được trình bày với các phần như sau:

* **Chương 2: Cơ sở lý thuyết** – trình bày các kiến thức nền tảng để xây dựng hệ thống phía máy chủ, máy khách, cơ sở dữ liệu và các công nghệ tích hợp
* **Chương 3: Phân tích và đặc tả yêu cầu** – Phần này sẽ đi sâu vào phân tích yêu cầu của bài toán đặt ra, đưa ra các ca sử dụng rồi sau đó minh họa chúng qua các bảng và biểu đồ trực quan.
* **Chương 4: Thiết kế và xây dựng hệ thống** – Trong chương này sẽ trình bày quá trình áp dụng lý thuyết vào thực tế, đưa các công nghệ vào để giải quyết bài toán đặt ra và xây dựng hệ thống.
* **Chương 5: Triển khai thực tế** – Phần này sẽ trình bày về môi trường phát triển và triển khai hệ thống, cùng với đó là kết quả thực nghiệm.
* **Chương 6: Kết luận –** Tổng kết quả đã đạt được so với mục tiêu khóa luận đề ra và định hướng phát triển tiếp theo.

# CƠ SỞ LÝ THUYẾT

Hệ thống phát triển công cụ quản lý dự án và công việc nội bộ được xây dựng trên nền tảng web. Ở chương này khoá luận trình bày những công nghệ đã nghiên cứu được áp dụng cho việc xử lý logic nghiệp vụ, tương tác giữa các thành phần trong hệ thống và với cơ sở dữ liệu.

## 2.1. Giới thiệu ReactJS

React được biết đến như là một Framework JavaScript phổ biến nhất thế giới hiện nay được sử dụng nhằm đơn giản hóa việc phát triển giao diện người dùng. Được phát triển bởi Facebook và bản release đầu tiên trên thế giới vào năm 2013 và được sử dụng để viết ra Facebook, instagram và được sự ủng hộ sử dụng của rất nhiều công ty khác trên thế giới. Mục tiêu của React chính là đơn giản để phát triển. Tất cả trạng thái đều được tập trung tại một thời điểm, bằng cách chia giao diện người dùng thành tập hợp các thành phần (components).

Kết hợp với Redux, một state management lib rất phổ biến hiện nay, React/Redux trở thành một bộ “framework” kinh điển để phát triển Single Page App – một trong những mối quan tâm hàng đầu của những nhà phát triển, đặc biệt là trong giới lập trình web.

Ngày nay, ReactJS đang ngày một phổ biến hơn vì tính đơn giản và linh hoạt. Nhiều người còn cho rằng, đây sẽ là tương lai của ngành web development. Người ta ước tính có khoảng hơn 1.300 developer và hơn 94.000 website đang sử dụng ReactJS.

***Một số khái niệm trong React***

* **JSX:** JSX = Javascript + XML được biết đến là một phần mở rộng của Javascript cho phép bạn tạo các thành phần giao diện người dùng tùy chỉnh một khai báo. Việc sử dụng cú pháp JSX thay thế có các loại cú pháp khác cũng không phải là một trong những trở ngại quá lớn cho các nhà phát triển hoặc thiết kế giao diện bởi cách viết cú pháp của JSX khá giống với cách viết của các thẻ HTML. Chính điều này cũng sẽ giúp cho việc viết code và sửa code một cách dễ dàng.
* **React components:** Component cho phép chúng ta chia nhỏ các thành phần UI độc lập mục đích để dễ quản lý và tái sử dụng nó. Trong mỗi React App đều có thể chứa rất nhiều components, mỗi components trong đó thường nhận về các props và trả về React elements từ đó hiển thị ra cho UI. Components trong React thường được viết theo 2 loại chính đó là functional component và class components.
* **Props và State**
  + **Props:** Props là viết tắt của properties. Chúng là các giá trị đơn hoặc các đối tượng có chứa một tập hợp các giá trị được chuyển đến Component. Nó sử dụng quy ước đặt tên tương tự như các thuộc tính của thẻ HTML. Props là cách để các component giao tiếp với nhau và được truyền từ component cha.
  + **State:** Khác với props là bất biến thì state có thể thay đổi. State được quản lý chỉ bởi duy nhất một component, nó cũng không thể truyền xuống cho component con.
* **Lifecycle:** Trong React Component, một chu kì cũng xuất hiện, components được khởi tạo (hiển thị ra DOM), update, và kết thúc (unmount),..đó được gọi là một component life cycle. React cho phép chúng ta tham gia vào các giai đoạn của mỗi component bằng cách sử dụng các phương thức được xây dựng sẵn trong mỗi giai đoạn đó
* **Redux trong React:** Redux là một thư viện giúp chúng ta quản lý các state một cách tốt hơn. Thay vì phải truyền state qua từng Component thì Redux sẽ tạo ra một store duy nhất dùng để thay đổi dữ liệu.

## 2.2. Giới thiệu NodeJS

NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript , giúp xây dựng các ứng dụng web một cách đơn giản và dễ mở rộng. Theo [khảo sát của Stack Overflow](https://insights.stackoverflow.com/survey/2019) về các framework, nền tảng được sử dụng nhiều nhất năm 2019, NodeJS đã giành vị trí số 1 với số lượng người dùng lên đến gần 50%. NodeJS có thể chạy trên nhiều nền tảng khác nhau như: Windows, Linux, Unix, Mac OS,…

NodeJS cung cấp một kiến trúc event-driven architecture và là một non-blocking I/O API băng thông của một ứng dụng và có khả năng mở rộng. NodeJS cho phép tạo ra những máy chủ web và các công cụ mang sử dụng ngôn ngữ JavaScript và các module. Các module của NodeJS đã giúp việc viết code đơn giản hơn, giảm đi độ phức tạp cho viết ứng dụng trên server. Đặc biệt NodeJS được chọn làm nền tảng phát triển của các phần mềm lớn như Microsoft, Yahoo, Walmart, Groupon, SAP, LinkedIn, Rakuten, Paypal, Voxer và GoDaddy…

***Một số đặc tính nổi bật của NodeJS***

* **Không đồng bộ**: Tất cả các API của NodeJS đều không đồng bộ (none-blocking), nó chủ yếu dựa trên nền của NodeJS Server và chờ đợi Server trả dữ liệu về. Việc di chuyển máy chủ đến các API tiếp theo sau khi gọi và cơ chế thông báo các sự kiện của Node.js giúp máy chủ để có được một phản ứng từ các cuộc gọi API trước (Realtime).
* **Khả năng mở rộng**: Node.js sử dụng một mô hình luồng duy nhất với sự kiện lặp. cơ chế tổ chức sự kiện giúp các máy chủ để đáp ứng một cách không ngăn chặn và làm cho máy chủ cao khả năng mở rộng như trái ngược với các máy chủ truyền thống mà tạo đề hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.
* **Tất Cả Đều Viết Bằng Javascript:** NodeJS là Javascript, tức là với NodeJS thì bây giờ lập trình viên full stack chỉ cần làm việc với một ngôn ngữ. Hơn nữa, tính ổn định của Javascript qua các phiên bản ES5, ES6, rồi 7, 8 gì đó ngày càng hoàn thiện, ổn định và mang lại nhiều tính năng hơn cho lập trình viên
* **Giao tiếp thời gian thực:** Công nghệ đang phát triển chóng mặt. Các công nghệ theo thời gian thực đang nảy mầm phát triển. Các công nghệ như WebSocket đã làm cho ứng dụng web gần như không có thời gian trễ. Node Js có thể xử lý một lượng kết nối khổng lồ như trên.
* **Hiệu Năng:** Nodejs sử dụng engine V8 của Google, một thứ nhanh kinh khủng, cùng với non-blocking IO thì việc tạo một trang web chậm chạp như bây giờ gần như không thể
* **Tận Dụng Tối Đa Phần Cứng**: Là một lập trình web, chẳng ai không biết sự tốn kém chính là host (phần cứng). Nhưng giờ Node Js đã cải thiện được điều đó. Hơn thế nữa là việc Nodejs có thể phát triển theo chiều ngang chỉ cần cho một Load Balancer đứng phía trước là được rồi.

***Expressjs library***

Express.js là một khuôn khổ của Node.js. Điều đó có nghĩa là hầu hết các mã code đã được viết sẵn cho các lập trình viên làm việc. Bạn có thể tạo các ứng dụng 1 web,  nhiều web hoặc kết hợp bằng cách sử dụng Express.js. Expressjs nhẹ, nó giúp tổ chức các ứng dụng web ở phía máy chủ thành một kiến ​​trúc MVC có tổ chức hơn.

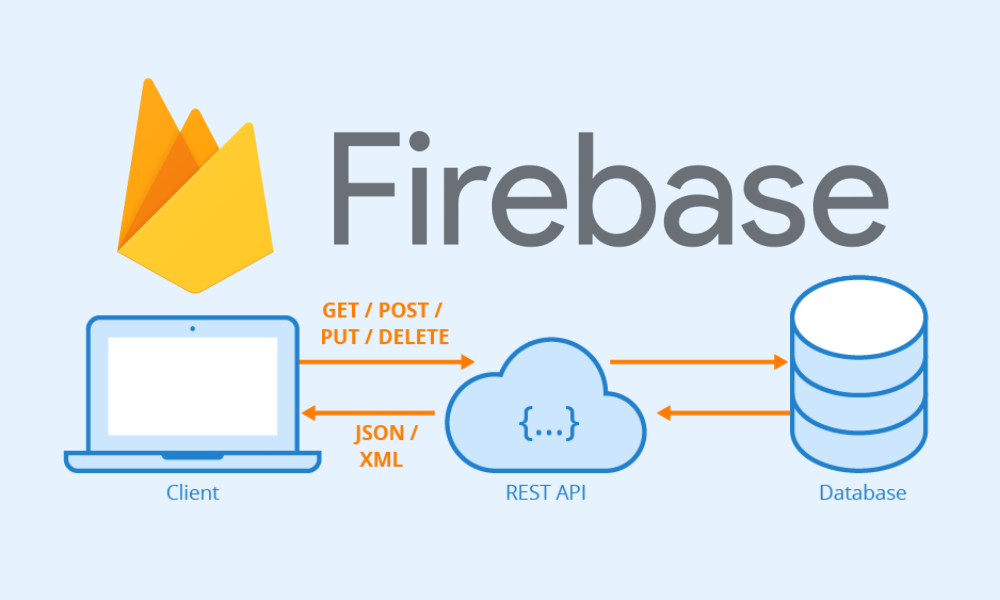
Thư viện JavaScript của Expressjs giúp các lập trình viên xây dựng các ứng dụng web hiệu quả và nhanh chóng. Express.js nâng cao chức năng của Node.js. Trên thực tế, nếu bạn không sử dụng Express.js, thì bạn phải thực hiện rất nhiều lập trình phức tạp để xây dựng một API hiệu quả. Nó đã làm cho việc lập trình trong Node.js trở nên dễ dàng và có nhiều tính năng bổ sung. Ngay cả những người mới bắt đầu tham gia vào lĩnh vực phát triển ứng dụng web cũng có thể sử dụng Express.js vì nó hỗ trợ JavaScript. Javascript rất dễ học đối với bất kỳ ai ngay cả khi họ không có bất kỳ kiến ​​thức nào về bất kỳ ngôn ngữ nào khác. Vì vậy, Express.js cho phép các tài năng trẻ tham gia vào lĩnh vực phát triển ứng dụng web và đạt được thành công.

Express.js có thể giảm một nửa thời gian viết mã mà vẫn giúp chúng tôi xây dựng các ứng dụng web hiệu quả. Nó không chỉ giảm thời gian mà còn giảm nỗ lực cần thiết để xây dựng các ứng dụng web với sự trợ giúp của các tính năng khác nhau của nó. Express.js cung cấp một định tuyến đơn giản cho các yêu cầu của khách hàng. Nó cũng cung cấp một phần mềm trung gian chịu trách nhiệm đưa ra quyết định để đưa ra các phản hồi chính xác cho các yêu cầu của khách hàng. Nếu không có Express.js, bạn phải viết mã code của riêng mình để xây dựng thành phần định tuyến. Đó là một công việc tốn thời gian và tẻ nhạt. Express.js giúp công việc này đơn giản, linh hoạt, hiệu quả, tối giản cùng với khả năng mở rộng cho các lập trình viên. Nó cũng có lợi thế về hiệu suất mạnh mẽ vì nó là một framework của Node.js.

## 2.3. Giới thiệu Firebase

Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud hỗ trợ phát triển trang web và ứng dụng di động do Google phát triển giúp chúng ta xây dựng, cải thiện và mở rộng ứng dụng của mình mà không cần backend hay server. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu. Hay nói cách khác Firebase là một bộ công cụ giúp xây dựng ứng dụng và các công cụ mà nó cung cấp cho bao gồm một phần lớn các dịch vụ các nhà phát triển thường phải tự xây dựng nhưng không thật sự muốn, nhằm tiết kiệm thời gian hoặc đơn giản là tập trung hơn vào phần trải nghiệm của ứng dụng

Firebase giúp các lập trình viên rút ngắn thời gian phát triển và triển khai ứng dụng. Chức năng chính là giúp nhà phát triển đơn giản hóa các thao tác với cơ sở dữ liệu và server. Ngoài ra Firebase còn là dịch vụ đa nền tảng, đa năng và bảo mật rất tốt. Dưới đây ta sẽ đi sâu hơn về các dịch vụ của Firebase áp dụng vào khóa luận này



*Hình 2.1 Mô hình Firebase RESTFUL API*

### 2.3.1. Firebase Authentication

Hoạt động nổi bật của Firebase là xây dựng các bước xác thực người dùng bằng Email, Facebook, Twitter, GitHub, Google,... Đồng thời cũng xác thực nặc danh cho các ứng dụng. Hoạt động xác thực có thể giúp thông tin cá nhân của người sử dụng được an toàn và đảm bảo không bị đánh cắp tài khoản.



*Hình 2.2 Các ứng dụng phổ biến firebase hỗ trợ đăng nhập*

Việc xác thực người dùng là một chức năng vô cùng quan trọng trong việc phát triển ứng dụng. Tuy nhiên, khi bạn muốn xác thực với nhiều phương thức khác nhau như email, số điện thoại, google, facebook sẽ tốn nhiều thời gian và công sức. Firebase Authentication giúp thực hiện việc đó một cách dễ dàng, giúp người dùng nhanh chóng tiếp cận sản phẩm hơn. Vì thế, nó là một chức năng vô cùng hữu ích của firebase.

***Cách thức hoạt động***

Để đăng nhập một người dùng vào ứng dụng của bạn, trước tiên bạn nhận được thông tin xác thực từ người dùng đó. Các thông tin xác thực này có thể là địa chỉ email và mật khẩu của người dùng hoặc mã thông báo OAuth từ nhà cung cấp danh tính được liên kết. Sau đó, bạn chuyển các thông tin xác thực này cho SDK xác thực Firebase. Các dịch vụ phụ trợ của chúng tôi sau đó sẽ xác minh các thông tin đăng nhập đó và trả lại phản hồi cho khách hàng.

Sau khi đăng nhập thành công, có thể truy cập thông tin hồ sơ cơ bản của người dùng và bạn có thể kiểm soát quyền truy cập của người dùng vào dữ liệu được lưu trữ trong các sản phẩm Firebase khác. Bạn cũng có thể sử dụng mã xác thực được cung cấp để xác minh danh tính của người dùng trong các dịch vụ phụ trợ của riêng bạn.

### 2.3.2.Firebase Cloud Firestore

Cloud Firestore là một Database linh hoạt và dễ mở rộng cho mobile, web và server được phát triển từ Firebase and Google Cloud Platform. Cũng giống như Firebase realtime database Cloud Firestore giúp cho việc đồng bộ dữ liệu giữa các ứng dụng phía client một các nhanh chóng (Realtime) và hỗ trợ lưu offline data trong ứng dụng của bạn.

Cloud Firestore là một cloud-hosted, NoSQL database mà các ứng dụng phía client có thể trực tiếp truy cập thông qua native SDKs. Nó lưu dữ liệu theo mô hình dữ liệu NoSQL. Dữ liệu được lưu trữ trong các file tài liệu chứa các trường được ánh xạ vào các giá trị. Các file tài liệu này được lưu trữ trong các tập hợp chúng có thể sử dụng nó để tổ chức dữ liệu và truy vấn dữ liệu. Cloud Firestore hỗ trợ rất nhiều kiểu dữ liệu từ đơn giản như String, Integer hay những kiểu dữ liệu phức tạp như các nested object

***Các tính năng chính***

* **Tính linh hoạt:** Cloud Firestore hỗ trợ các cấu trúc dữ liệu linh hoạt, phân cấp dữ liệu. Lưu trữ dữ liệu của bạn trong các document , được tổ chức thành các collection. Các document có thể chứa các đối tượng phức tạp.
* **Truy vấn tượng trưng:** Bạn có thể sử dụng các truy vấn để truy xuất các document riêng lẻ hoặc để truy xuất tất cả các document trong collection khớp với các tham số truy vấn của bạn. Các truy vấn của bạn có thể bao gồm nhiều bộ lọc, kết hợp giữa bộ lọc và sắp xếp.
* **Cập nhật thời gian thực:** Cloud Firestore sử dụng đồng bộ hóa dữ liệu để cập nhật dữ liệu trên mọi thiết bị được kết nối. Nó cũng được thiết kế để thực hiện các truy vấn tìm nạp một lần .
* **Hỗ trợ offline:** Cloud Firestore lưu trữ dữ liệu tại local, vì vậy ứng dụng có thể viết, đọc, nghe và truy vấn dữ liệu ngay cả khi thiết bị ngoại tuyến. Khi thiết bị trở lại trực tuyến, Cloud Firestore sẽ đồng bộ hóa mọi thay đổi cục bộ lên Cloud Firestore.
* **Khả năng mở rộng:** Mang đến khả năng từ Google Cloud Platform thiết kế để sử dụng cơ sở dữ liệu khó khăn nhất từ các ứng dụng lớn nhất thế giới

Cách thức hoạt động

Cloud Firestore là cơ sở dữ liệu NoSQL, được lưu trữ trên đám mây mà các ứng dụng web iOS, Android và web của bạn có thể truy cập trực tiếp thông qua SDK gốc. Cloud Firestore cũng có sẵn trong các SDK Node.js, Java, Python, Unity, C ++ và Go, ngoài các API REST và RPC.

Theo mô hình dữ liệu NoSQL của Cloud Firestore, bạn lưu trữ dữ liệu trong các tài liệu có chứa các trường ánh xạ tới các giá trị. Các tài liệu này được lưu trữ trong các bộ sưu tập, là các vùng chứa tài liệu của bạn mà bạn có thể sử dụng để sắp xếp dữ liệu và xây dựng các truy vấn. Tài liệu hỗ trợ nhiều kiểu [dữ liệu khác nhau](https://firebase.google.com/docs/firestore/manage-data/data-types) , từ chuỗi và số đơn giản, đến các đối tượng lồng nhau, phức tạp. Bạn cũng có thể tạo các bộ sưu tập con trong tài liệu và xây dựng cấu trúc dữ liệu phân cấp mở rộng quy mô khi cơ sở dữ liệu của bạn phát triển. [Mô hình dữ liệu](https://firebase.google.com/docs/firestore/data-model) Cloud Firestore hỗ trợ bất kỳ cấu trúc dữ liệu nào hoạt động tốt nhất cho ứng dụng của bạn.

Ngoài ra, truy vấn trong Cloud Firestore rất rõ ràng, hiệu quả và linh hoạt. Tạo các truy vấn nông để truy xuất dữ liệu ở cấp độ tài liệu mà không cần truy xuất toàn bộ bộ sưu tập hoặc bất kỳ bộ sưu tập con lồng nhau nào. Thêm sắp xếp, lọc và giới hạn cho các truy vấn hoặc con trỏ của bạn để phân trang kết quả của bạn. Để giữ cho dữ liệu trong các ứng dụng của bạn luôn cập nhật mà không cần truy xuất toàn bộ cơ sở dữ liệu mỗi khi cập nhật, hãy thêm trình nghe thời gian thực. Việc thêm trình nghe thời gian thực vào ứng dụng của bạn sẽ thông báo cho bạn bằng ảnh chụp nhanh dữ liệu bất cứ khi nào dữ liệu mà ứng dụng khách của bạn đang lắng nghe các thay đổi, chỉ truy xuất các thay đổi mới.

Bảo vệ quyền truy cập vào dữ liệu của bạn trong Cloud Firestore với Xác thực Firebase và Quy tắc bảo mật Cloud Firestore cho Android, iOS và JavaScript hoặc Quản lý danh tính và truy cập (IAM) cho các ngôn ngữ phía máy chủ.

## 2.4. Giới thiệu Jira software RestFul API

Jira software là một ứng dụng theo dõi và quản lý lỗi / vấn đề trong dự án, được phát triển bởi công ty phần mềm Atlassian của Australia. Cách thức hoạt động của JIRA dựa vào trọng tâm là kết quả công việc, có thể sử dụng ngay và linh hoạt khi sử dụng. Jira cung cấp các API REST để tạo tiện ích bổ xung cho ứng dụng của bạn. Sử dụng các API này để tương tác với các tập lệnh Jira hoặc phát triển với bất kì tích hợp nào khác

**Authentication:** Các lệnh gọi API được xác thực qua JWT (Mã thông báo web JSON). Điều này được tích hợp vào các thư viện Atlassian Connect được hỗ trợ. Ở cấp độ cao, xác thực hoạt động bằng cách tiện ích bổ sung trao đổi bối cảnh bảo mật với ứng dụng. Ngữ cảnh này được sử dụng để tạo và xác thực mã thông báo JWT, được nhúng trong các lệnh gọi API. Nếu bạn đang tích hợp một ứng dụng tại chỗ với các API Jira REST, các lệnh gọi API được xác thực thông qua mã thông báo OAuth. Để lấy mã thông báo, hãy tạo một bộ thông tin xác thực OAuth có quyền cho các API mà ứng dụng cần truy cập. Sử dụng thông tin đăng nhập để yêu cầu mã thông báo tại trang <https://api.atlassian.com/oauth/token>

**URI structure:** Các API REST của Jira cung cấp quyền truy cập vào các tài nguyên (thực thể dữ liệu) thông qua các đường dẫn URI. Để sử dụng REST API, ứng dụng của bạn sẽ thực hiện một yêu cầu HTTP và phân tích cú pháp phản hồi. API Jira Agile REST sử dụng [JSON](http://en.wikipedia.org/wiki/JSON) như định dạng của nó truyền thông, và các phương pháp HTTP tiêu chuẩn như GET, PUT, POST và DELETE (xem giới thiệu API dưới đây mà các phương pháp có sẵn cho mỗi tài nguyên).

## 2.5. Giới thiệu Google services API

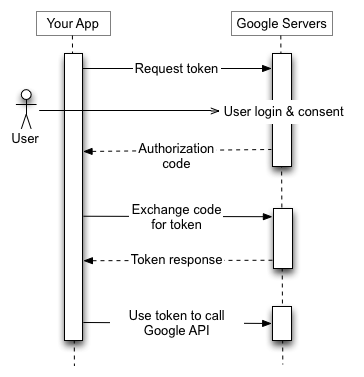
### 2.5.1. Google Oauth 2.0

API của Google sử dụng [giao thức OAuth 2.0](https://tools.ietf.org/html/rfc6749) để xác thực và ủy quyền. Google hỗ trợ các nền tảng OAuth 2.0 phổ biến, chẳng hạn như các nền tảng dành cho máy chủ web, phía máy khách, các ứng dụng thiết bị đầu vào được cài đặt và giới hạn.

Để bắt đầu, hãy lấy bằng chứng xác thực ứng dụng khách OAuth 2.0 từ [bảng điều khiển API của Google](https://console.developers.google.com/) . Sau đó, ứng dụng khách của bạn yêu cầu mã thông báo truy cập từ máy chủ ủy quyền của Google, trích xuất mã thông báo từ phản hồi và gửi mã thông báo đến API Google mà bạn muốn truy cập. Để có một minh chứng tương tác về việc sử dụng OAuth 2.0 với Google (bao gồm tùy chọn sử dụng thông tin đăng nhập khách hàng của riêng bạn)

***Các bước cơ bản***

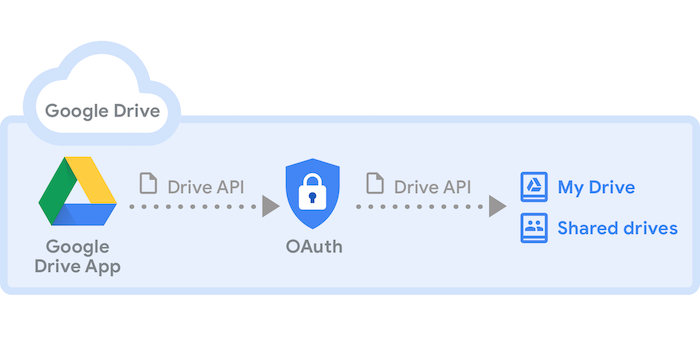
* **Bước 1. Nhận thông tin đăng nhập OAuth 2.0 từ Bảng điều khiển API của Google:** Truy cập [bảng điều khiển API của Google](https://console.developers.google.com/) để nhận thông tin đăng nhập OAuth 2.0 chẳng hạn như client ID và client secret mà cả Google và ứng dụng của bạn đều biết. Tập hợp các giá trị khác nhau dựa trên loại ứng dụng bạn đang xây dựng
* **Bước 2. Lấy mã access token từ Máy chủ ủy quyền của Google.**
  + Trước khi ứng dụng của bạn có thể truy cập dữ liệu riêng tư bằng API Google, ứng dụng phải có được access token cấp quyền truy cập vào API đó. Một mã thông báo truy cập duy nhất có thể cấp các mức độ truy cập khác nhau cho nhiều API. Một tham số biến được gọi là scope kiểm soát tập hợp tài nguyên và hoạt động mà access token cho phép. Trong quá trình yêu cầu mã thông báo truy cập, ứng dụng của bạn sẽ gửi một hoặc nhiều giá trị trong scope tham số.
  + Có một số cách để thực hiện yêu cầu này và chúng khác nhau tùy thuộc vào loại ứng dụng bạn đang xây dựng. Ví dụ: một ứng dụng JavaScript có thể yêu cầu access token bằng cách sử dụng trình duyệt chuyển hướng đến Google, trong khi một ứng dụng được cài đặt trên thiết bị không có trình duyệt sử dụng các yêu cầu dịch vụ web.
  + Một số yêu cầu yêu cầu bước xác thực mà người dùng đăng nhập bằng tài khoản Google của họ. Sau khi đăng nhập, người dùng được hỏi liệu họ có sẵn sàng cấp một hoặc nhiều quyền mà ứng dụng của bạn đang yêu cầu hay không. Quá trình này được gọi là sự đồng ý của người dùng .
  + Nếu người dùng cấp ít nhất một quyền, Máy chủ ủy quyền của Google sẽ gửi cho ứng dụng của bạn access token (hoặc authorization code mà ứng dụng của bạn có thể sử dụng để lấy mã thông báo truy cập) và danh sách các phạm vi truy cập được cấp bởi access token đó). Nếu người dùng không cấp quyền, máy chủ sẽ trả về lỗi.
* **Bước 3. Kiểm tra các phạm vi quyền truy cập được cấp bởi người dùng:** So sánh các phạm vi được bao gồm access token để truy cập với các phạm vi được yêu cầu để truy cập các tính năng và chức năng của ứng dụng của bạn phụ thuộc vào quyền truy cập vào API Google có liên quan. Tắt bất kỳ tính năng nào của ứng dụng không thể hoạt động nếu không có quyền truy cập vào API liên quan.
* **Bước 4. Gửi mã access token tới một API:** Sau khi ứng dụng có được access token truy cập, ứng dụng sẽ gửi mã thông báo tới API Google trong [tiêu đề yêu cầu Ủy quyền HTTP](https://developer.mozilla.org/docs/Web/HTTP/Headers/Authorization) . Có thể gửi mã thông báo dưới dạng tham số chuỗi truy vấn URI, nhưng Google không khuyến khích việc này, vì các tham số URI có thể kết thúc trong các tệp nhật ký không hoàn toàn an toàn. Ngoài ra, thực hành REST tốt là tránh tạo các tên tham số URI không cần thiết
* **Bước 5. Làm mới mã thông báo truy cập (nếu cần):** Access token truy cập có vòng đời giới hạn (1 giờ). Nếu ứng dụng của bạn cần quyền truy cập vào API Google ngoài thời gian tồn tại của một access token duy nhất, ứng dụng có thể nhận được refresh token. Refresh token cho phép ứng dụng của bạn nhận được access token mới.



*Hình 2.3* Sơ đồ sử dụng Oauth 2.0 cho máy chủ Web

### 2.5.2. Google Drive API

API Google Drive cho phép bạn tạo các ứng dụng tận dụng bộ nhớ đám mây của Google Drive. Bạn có thể phát triển các ứng dụng tích hợp với Google Drive và tạo chức năng mạnh mẽ trong ứng dụng của mình bằng API Google Drive.



*Hình 2.4* Sơ đồ sử dụng Google Drive API

***Những thao tác với Google Drive API***

* Download và Upload file lên Google Drive
* Tìm kiếm file, thư mục trên Google Drive.
* User có thể chia sẻ file, thư mục hợp tác về nội dung trên Google Drive.
* Kết hợp với API Google Picker để tìm kiếm tất cả các tệp trong Google Drive, sau đó trả lại tên tệp, URL, ngày sửa đổi cuối cùng và người dùng.
* Tạo các phím tắt là các liên kết bên ngoài đến dữ liệu được lưu trữ bên ngoài Drive, trong một kho lưu trữ dữ liệu hoặc hệ thống lưu trữ đám mây khác.
* Tạo thư mục Drive chuyên dụng để lưu trữ dữ liệu của ứng dụng để ứng dụng không thể truy cập tất cả nội dung của người dùng được lưu trữ trong Google Drive. Xem Lưu trữ dữ liệu dành riêng cho ứng dụng.
* Tích hợp với Giao diện người dùng Google Drive, là giao diện người dùng web tiêu chuẩn của Google mà bạn có thể sử dụng để tương tác với các tệp Drive.

### 2.5.3. Google Calendar API

Google Calendar là một ứng dụng khá tuyệt vời giúp hàng triệu người dùng làm việc tốt hơn. Nhiều người không sử dụng thường xuyên, chỉ sử dụng cho các sự kiện quan trọng.

Calendar Google có một số công cụ ẩn và tính năng có thể giúp bạn dễ dàng hơn trong việc lập lịch trình cho các cuộc họp, và tự động hóa phần lịch. Hữu ích nhất trong số này được liệt kê dưới đây. Kết hợp với một số phần mở rộng trong Chrome, bạn sẽ sử dụng đầy đủ chức năng mà Google Calendar hiện có.

Google Calendar Api cho phép bạn hiển thị, tạo và sửa đổi các sự kiện lịch cũng như làm việc với nhiều đối tượng khác liên quan đến lịch, chẳng hạn như lịch hoặc điều khiển truy cập. Tài liệu này mô tả cách sử dụng lệnh gọi RESTful và thư viện máy khách cho các ngôn ngữ lập trình khác nhau (Java, PHP, .NET, JavaScript, NodeJs, Ruby, Python, Go, Android, iOS).

# PHÂN TÍCH VÀ ĐẶC TẢ YÊU CẦU

Xuất phát từ việc khảo sát yêu cầu thực tế, tôi đã đánh giá và xác định các yêu cầu phù hợp với bối cảnh bài toán được đưa ra. Chương này sẽ đi vào phân tích và đặc tả yêu cầu của hệ thống, cùng với đó là thiết kế các biểu đồ ca sử dụng và biểu đồ tuần tự.

## 3.1. Xác định nhóm các đối tượng sử dụng trong hệ thống

Hệ thống gồm những nhóm người dùng chính như sau:

* Admin: là những người quản lý hệ thống và có quyền cao nhất. Có thể quản lý các tài khoản khác và tham gia vào hệ thống để phụ trách những chức năng cụ thể
* Leader: là những người đứng đầu một team hoặc một dự án, truy cập hệ thống để quản lý dự án và quản lý công việc của các thành viên trong team
* Manager: bộ phận phụ trách việc giám sát, thống kê và theo dõi tiến độ báo cáo công việc hàng ngày của người dùng trong hệ thống
* Develop: những thành viên còn lại trong công ty tham gia hệ thống để báo cáo công tiến độ công việc hàng ngày và sử dụng một số chức năng cụ thể khác

## 3.2. Phân tích yêu cầu

### 3.2.1. Yêu cầu chức năng

***Chức năng quản lý người dùng***

Chức năng cho phép những tài khoản có quyền truy cập xem danh sách người dùng trong hệ thống và thực hiện các chức năng cụ thể như xoá tài khoản người dùng khỏi hệ thống, sửa một số thông tin hoặc thêm người dùng đó vào dự án

***Chức năng xin nghỉ***

Chức năng cho phép hiển thị danh sách xin nghỉ ứng với quyền truy cập của tài khoản. Người dùng có thể gửi yêu cầu xin nghỉ của mình lên hệ thống, admin/leader vào xem danh sách các yêu cầu sau đó phản hồi lại cho người dùng. Sau khi đơn xin nghỉ được phê duyệt, gửi mail kèm thông một số thông tin trong đơn xin nghỉ tới tất cả gmail của người dùng trong hệ thống

***Chức năng quản lý báo cáo***

Đây là chức năng cho phép tất cả thành viên trong hệ thống tạo báo cáo công việc hiện tại mình đang làm, nhưng tương ứng với mỗi loại tài khoản việc quản lý báo cáo sẽ có những chức năng khác nhau. Đối với người có quyền develop thì chức năng chính vẫn tạo, sửa, xóa báo cáo trong hệ thống và ngoài ra có thể lọc xem danh sách báo cáo đã tạo dưới dạng bảng phân trang theo một số trường của báo cáo. Manager theo dõi công việc và export danh sách báo cáo ra file excel. Với người dùng có quyền admin/leader sẽ có thêm các quyền chỉ định báo cáo cho người khác và thực hiện việc đánh giá việc hoàn thành công việc báo cáo đó trong thực tế

***Chức năng thống kê***

Cung cấp cho người dùng cái nhìn tổng quát về hiệu suất đánh giá công việc của admin/leader, biểu đồ tổng số báo báo theo độ khó trong các khoản thời gian

***Chức năng quản lý jira***

Đối với những thành viên trong hệ thống tham gia dự án và được giao việc trên trang jira software, hệ thống kết nối với jira software và người dùng có thể sử dụng chức năng này để xem danh sách các báo cáo theo dự án trên jira software và chỉ định những báo cáo này về hệ thống quản lý báo cáo của mình

***Chức năng tích hợp google calendar***

Hiển thị dữ liệu lịch hiện tại từ google calendar theo các khoảng thời gian. Với chức năng này bạn có thể theo dõi các hoạt động hàng ngày như cuộc họp, sự kiện, lịch nghỉ,...

### 3.2.2. Yêu cầu phi chức năng

Ngoài các yêu cầu chức năng đã nêu ở trên hệ thống cần đáp ứng các yêu cầu phi chức năng chỉ ra những tiêu chí đánh giá hoạt động của hệ thống để đem lại trải nghiệm tốt nhất cho người dùng

* **Hiệu năng hoạt động**: hệ thống phản hồi nhanh chóng khi có yêu cầu gửi lên nhất định, đảm bảo việc đáp ứng tốt kể cả khi có nhiều người dùng truy cập cùng một lúc
* **Tính khả dụng**: giao diện người dùng của hệ thống phải dễ dàng sử dụng, thân thiện và sẵn sàng. Người dùng có thể sử dụng thành thạo các chức năng sau khi được hướng dẫn
* **Tính tương thích**: hệ thống hoạt động ổn định trên hầu hết các nền tảng web và phiên bản khác nhau
* **Tính bảo mật**: người dùng xác thực thông qua đăng nhập Google, thông tin người dùng và dữ liệu của hệ thống được đảm bảo về bảo mật và toàn vẹn dữ liệu, tránh rò rỉ gây mất thông tin
* **Tính bảo trì**: đảm bảo việc dễ dàng trong việc sửa chữa và nâng cấp đáp ứng với các yêu cầu hoặc môi trường thay đổi
* **Tính mở rộng**: hệ thống có thể mở rộng thông qua việc bổ sung các tính năng mới hoặc cải thiện các chức năng hiện có mà vẫn đảm bảo không làm suy giảm các chức năng hiện có

## 3.3. Xây dựng biểu đồ sơ lược ca sử dụng

### 3.3.1. Xác định tác nhân và các ca sử dụng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ca sử dụng | Tác nhân | | | | Mô tả sơ bộ ca sử dụng |
| Admin | Leader | Manager | Develop |
| Đăng ký tài khoản | v | v | v | v | Đăng ký vào hệ thống qua Gmail được xác thực |
| Đăng nhập | v | v | v | v | Đăng nhập vào hệ thống qua tài khoản đã được đăng ký trước đó |
| Thay đổi thông tin người dùng | v |  |  |  | Chỉnh sửa thông tin của tài khoản |
| Xoá người dùng | v |  |  |  | Xoá tài khoản của người dùng khỏi hệ thống |
| Thêm mới người dùng | v |  |  |  | Thêm mới tài khoản đăng nhập bằng email/password. Tài khoản sau khi được admin thêm chỉ có thể mang quyền manager |
| Xem danh sách báo cáo | v | v | v | v | Xem danh sách báo cáo đã tạo của mình trong hệ thống |
| Thêm mới báo cáo | v | v | v | v | Thêm mới báo cáo công việc của mình lên hệ thống |
| Xoá báo cáo | v | v |  |  | Cho phép xoá báo cáo công việc đã chỉ định cho người dùng khác |
| Sửa báo cáo | v | v | v | v | Người tạo hoặc người được chỉ định làm sẽ có thể chỉnh sửa thông tin báo cáo của mình |
| Đánh giá công việc | v | v |  |  | Đánh giá mức độ hoàn thành tiến độ công việc được báo cáo và phản hồi lại nếu có điều gì phát sinh |
| Export báo cáo | v |  |  | v | Xuất danh sách báo cáo ra file excel |
| Chỉ định báo cáo | v | v |  |  | Chỉ định báo cáo cho thành viên khác trong hệ thống |
| Quản lý báo cáo của nhân viên | v | v |  |  | Thực hiện các tác vụ quản lý trong hệ thống báo cáo của người dùng khác |
| Xem danh sách đơn xin nghỉ | v | v | v | v | Hiển thị danh sách đơn xin nghỉ theo loại tài khoản truy cập |
| Tạo đơn xin nghỉ | v | v | v | v | Gửi yêu cầu xin nghỉ lên hệ thống và tới gmail của Admin |
| Sửa đơn xin nghỉ | v | v | v | v | Thay đổi thông tin của của đơn xin nghỉ nếu có chỉnh sửa |
| Phản hồi đơn xin nghỉ | v |  |  |  | Gửi phản hồi đơn xin nghỉ tới tất cả tài khoản trong hệ thống |
| Gửi mail xin nghỉ | v |  |  |  | Gửi gmail khi người dùng tạo đơn xin nghỉ hoặc admin phản hồi đơn xin nghỉ của người dùng |
| Xem báo cáo từ Jira | v | v | v |  | Hiển thị danh sách những báo cáo trong các dự án có quyền truy cập của tài khoản đăng nhập từ trang Jira software |
| Chỉ định báo cáo từ Jira về hệ thống | v | v | v |  | Lấy các báo cáo từ trang Jira software về hệ thống báo cáo cá nhân |
| Xem danh sách dự án | v | v |  |  | Xem danh sách thông tin các dự án có trong hệ thống |
| Thêm mới dự án | v |  |  |  | Thêm mới dự án vào hệ thống |
| Sửa thông tin dự án | v |  |  |  | Chỉnh sửa thông tin của dự án |
| Khoá/kích hoạt dự án | v | v |  |  | Khoá khi dự án đã đóng trong thực tế hoặc mở lại khi có chỉnh sửa |
| Xem thống kê báo cáo | v | v |  | v | Xem thống kê tổng quát các báo cáo của người dùng trong hệ thống |
| Xem lịch họp | v | v | v | v | Hiển thị lịch họp được cập nhật theo google calendar |
| Quản lý file drive | v | v | v | v | Người dùng có thể tương tác với các file dữ liệu cũng như file đính kèm theo báo cáo thông qua google drive |

*Bảng 3.1 Bảng chức tác nhân và ca sử dụng*

### 3.3.2. Biểu đồ ca sử dụng

Hệ thống có bốn tác nhân Admin, leader, develop và manager và dưới đây là biểu đồ ca sử dụng tương ứng với từng tác nhân.

***Ca sử dụng của Develop***



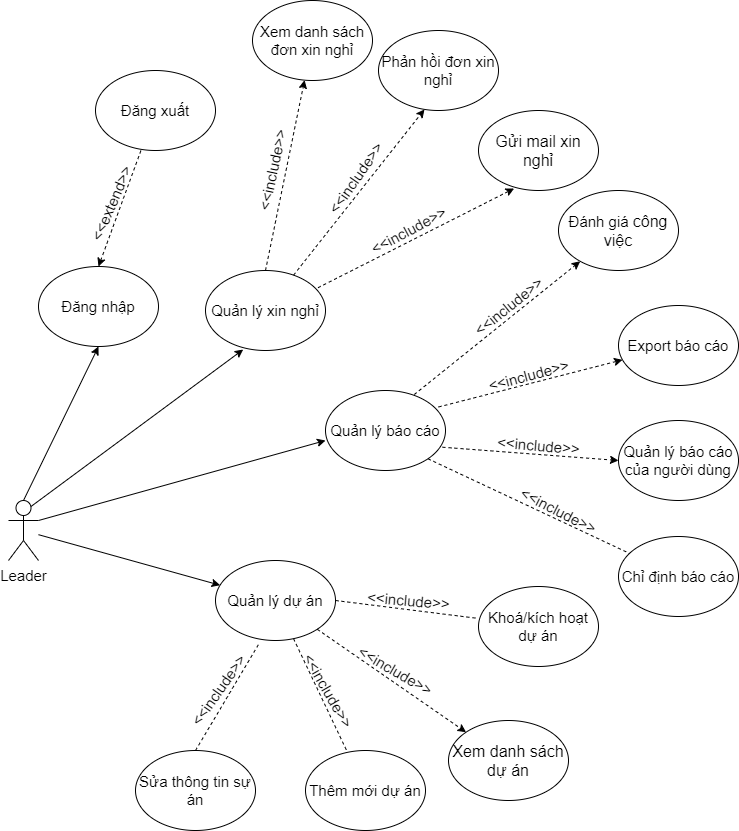
*Hình 3.1* Biểu đồ ca sử dụng “Develop”

***Ca sử dụng của Admin***



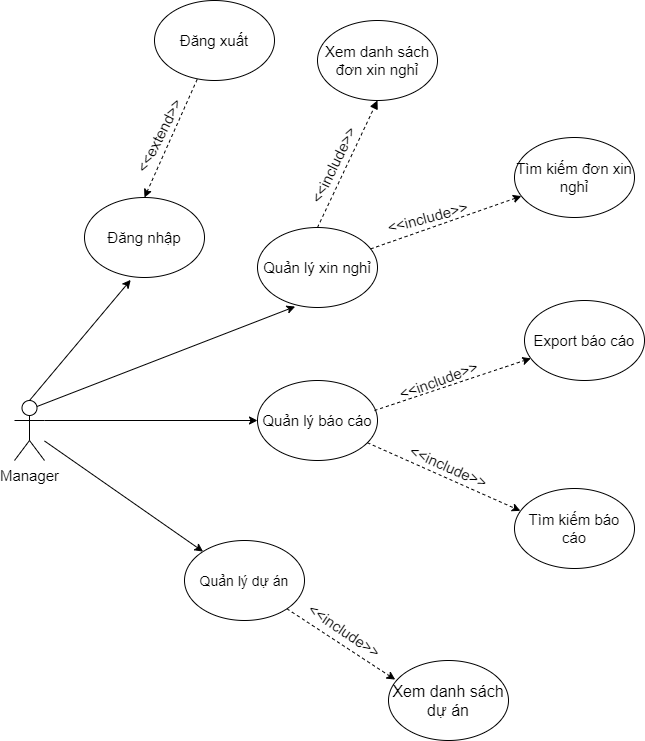
*Hình 3.2* Biểu đồ ca sử dụng “Admin”

***Ca sử dụng của Leader***



*Hình 3.3* Biểu đồ ca sử dụng “Leader”

***Ca sử dụng của Manager***



*Hình 3.4.* Biểu đồ ca sử dụng “Manager”

Trên đây là mô hình hóa ca sử dụng của các tác nhân trong hệ thống. Các chức năng này đảm bảo đáp ứng đầy đủ các yêu cầu của người dùng. Để có thể sử dụng các chức năng trên người dùng cần bắt buộc đăng nhập vào hệ thống là bước đầu tiên.

## 3.4. Đặc tả chi tiết từng ca sử dụng

### 3.4.1. Ca sử dụng “Đăng nhập”

**Mô tả:** Người dùng sử dụng tài khoản đã được cấp để đăng nhập và sử dụng các chức năng của hệ thống

**Tác nhân:** Tất cả các tác nhân

**Luồng sự kiện chính:**

* Ca sử dụng được bắt đầu khi người dùng muốn đăng nhập để tiếp cận với hệ thống bên trong
* Giao diện đăng nhập được hiển thị đầu tiên khi lần đầu người dùng truy cập vào hệ thống
* Manager nhập email/password mà hệ thống cung cấp và ấn đăng nhập
* Người dùng khác ấn đăng nhập bằng google và tiến hành xác thực bằng google
* Đăng nhập thành công hệ thống đưa người dùng vào giao diện sẵn sàng sử dụng

**Luồng thay thế:**

* Khi người dùng nhập sai thông tin, hiển thị thông báo yêu cầu người dùng nhập lại
* Hiển thị giao diện báo lỗi khi đăng nhập bằng gmail không có quyền truy cập vào hệ thống

**Yêu cầu đặc biệt**:

* Hệ thống được kết nối internet
* Đăng nhập bằng google công ty cung cấp

**Điều kiện trước:**

* Tài khoản đăng nhập đã được đăng ký thành công trong hệ thống cơ sở dữ liệu

**Điều kiện sau:**

* Nếu thành công:  Người dùng được điều hướng đến trang sử dụng chính của hệ thống
* Thất bại: Thông báo lỗi và giữ nguyên ở màn hình đăng nhập

### 3.4.2. Ca sử dụng “Thay đổi thông tin người dùng”

**Mô tả:** Cho phép admin thay đổi thông tin tài khoản của người dùng truy cập vào hệ thống

**Tác nhân:** Admin

**Luồng sự kiện chính:**

* Ca sử dụng được bắt đầu khi admin thực hiện thay đổi một thông tin của người dùng
* Hệ thống hiển thị danh sách người dùng dưới dạng bảng phân trang
* Admin tiến hành chọn tài khoản mà mình cần thay đổi thông tin để chỉnh sửa
* Tiến hành sửa các thông tin cần thiết và xác nhận để hệ thống cập nhật
* Thay đổi quyền người dùng: tài khoản sau khi được thay đổi quyền có thể truy cập các chức năng tương ứng với quyền hiện tại của tài khoản đó
* Thêm mới vào dự án: Tài khoản khi được thêm mới vào dự án có thể tạo báo cáo trong dự án đó
* Số ngày nghỉ: Khi số ngày nghỉ phép có sự thay đổi admin có thể vào cập nhật lại

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống được kết nối internet

**Điều kiện trước:**

* Người dùng truy cập hệ thống với quyền admin và tiến hành điền các trường bắt buộc

**Điều kiện sau**

Nếu thành công: Sau khi cập nhật thông tin, người dùng có thể truy cập các chức năng với thông tin tương ứng

Nếu thất bại: Thông tin hiện tại của tài khoản người dùng đó vẫn được giữ nguyên và hệ thống trả về lỗi cho người dùng

### 3.4.3. Ca sử dụng “Xóa người dùng”

**Mô tả:** Người dùng xóa những tài khoản đã không có quyền truy cập khỏi hệ thống

**Tác nhân:** Admin

**Luồng sự kiện chính:**

* Ca sử dụng được bắt đầu khi admin muốn xóa một tài khoản người dùng không còn hoạt động trong công ty khỏi hệ thống quản lý công việc
* Giao diện quản lý người dùng chứa bảng danh sách các tài khoản trong hệ thống
* Với mỗi tài khoản hệ thống hiển thị một button xóa tương ứng trong bảng danh sách
* Người dùng thực hiện xóa, một message sẽ được hiển thị để xác nhận lại hành động này

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet

**Điều kiện trước:**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin
* Tài khoản người dùng muốn xoá đã tồn tại trong hệ thống

**Điều kiện sau**

* Nếu thành công: Tài khoản sau khi bị xóa và không thể truy cập hệ thống được nữa
* Nếu thất bại: Tài khoản vẫn được giữ nguyên và hệ thống hiển thị lỗi cho người dùng

### 3.4.4. Ca sử dụng “Thêm mới người dùng”

**Mô tả:** Hệ thống cung cấp tài khoản để người dùng truy cập

**Tác nhân:** Admin

**Luồng sự kiện chính**

* Ca sử dụng được bắt đầu khi admin muốn cung cấp một tài khoản cho thành viên mới để truy cập hệ thống
* Tại giao diện quản lý người dùng admin ấn button “Add user”
* Hệ thống sẽ trả về một modal để chứa các thông tin cần thiết của một tài khoản
* Admin nhập các thông tin và xác nhận để tạo một tài khoản mới trong hệ thống
* Tài khoản khi được admin tạo bằng email & password chỉ có thể mang quyền Manager

**Luồng thay thế:**

* Tài khoản tạo với các quyền khác “manager” hệ thống sẽ thông báo lỗi
* Muốn một tài khoản có quyền khác để truy cập các chức năng tích hợp với google mail người dùng cần yêu cầu quản lý cấp cho một gmail công ty và dùng gmail đó để đăng nhập

**Yêu cầu đặc biệt**

* Hệ thống có kết nối internet

**Điều kiện trước**

* Người dùng phải đăng nhập với tài khoản có quyền admin và điền đầy đủ thông tin các trường bắt buộc

**Điều kiện sau**

* Nếu thành công: Thông tin tài khoản thêm vào database và người dùng có thể dùng tài khoản vừa tạo đó để truy cập vào hệ thống
* Nếu thất bại: Hệ thống hủy tác vụ thêm mới tài khoản và trả về thông báo lỗi cho người dùng

### 3.4.5. Ca sử dụng “Thêm mới báo cáo”

**Mô tả:** Người dùng thêm một báo cáo về công việc mình đang làm hiện tại vào hệ thống quản lý

**Tác nhân:** Tất cả tác nhân

**Luồng sự kiện chính:**

* Khi người dùng ấn mục “Report” tại thanh menu hệ thống trả về giao diện quản lý báo cáo
* Người dùng ấn button “Add report” để thêm một báo cáo mới vào hệ thống
* Giao diện hiển thị một modal để người dùng nhập thông tin cho công việc mà mình báo cáo
* Người dùng nhập các trường thông tin cần thiết
* Người dùng có thể đính kèm file vào báo cáo và những file này sẽ được lưu vào google drive
* Tiến hành ấn “Submit” để thêm mới báo cáo

**Luồng thay thế:**

* Muốn đính kèm file khi tạo báo cáo người dùng phải đăng nhập tài khoản gmail
* Người dùng phải đăng nhập với quyền admin/leader nếu muốn thực hiện chỉ định báo cáo cho người dùng khác
* Báo cáo sẽ được thêm mới ở giao diện quản lý báo cáo tương ứng của người được chỉ định

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet
* Chấp nhận uỷ quyền các thao tác với google drive khi đăng nhập nếu muốn đính kèm file vào báo cáo

**Điều kiện trước:** Không có

**Điều kiện sau:**

* Nếu thành công: Báo cáo được thêm vào cơ sở dữ liệu và hiển thị ở bản danh sách trong giao diện
* Nếu thất bại: Đóng modal thêm mới và hệ thống trả về lỗi cho người dùng

### 3.4.6. Ca sử dụng “Xem danh sách báo cáo”

**Mô tả:** Người dùng có thể xem lại danh sách những báo cáo mình đã thêm vào hệ thống trước đó

**Tác nhận:** Tất cả tác nhân

**Luồng sự kiện chính:**

* Khi người dùng ấn mục “Report” tại thanh menu hệ thống trả về giao diện quản lý báo cáo. Mặc định những báo này sắp xếp theo ngày đăng
* Các báo cáo đã thêm được hiển thị dưới dạng bảng phân trang. Bảng chứa thông tin của từng báo cáo, số báo cáo mặc định trong mỗi bảng là 10 và có thể chỉnh sửa
* Người dùng có thể lọc báo cáo nếu muốn xem danh sách báo cáo theo status, time, name, difficult

**Luồng thay thế:**

* Hiển thị bảng “No data” nếu chưa có báo cáo trong hệ thống của người dùng

**Yêu cầu đặc biệt:**

* Hệ thống có kết  nối internet

**Điều kiện trước:**

* Người dùng đã thêm báo cáo vào hệ thống trước đó

**Điều kiện sau:** Không có

### 3.4.7. Ca sử dụng “Sửa báo cáo”

**Mô tả:** Người dùng chỉnh sửa thông tin báo cáo

**Tác nhận**: Tất cả các tác nhân

**Luồng sự kiện chính:**

* Ca sử dụng bắt đầu khi người dùng muốn sửa thông tin một cáo báo đã thêm trước đó
* Trong bảng danh sách chọn báo cáo mà mình muốn sửa để xem chi tiết thông tin
* Hệ thống trả lại một modal hiển thị chi tiết thông tin và tiến hành chỉnh sửa các trường thay đổi của báo cáo này
* Có thể ấn xem lịch sử để biết những thay đổi trước đó của báo cáo này
* Tiến hành ấn “submit” để hoàn tất việc chỉnh sửa

**Luồng thay thế:**

* Khi đã thêm ngày thực tế hoàn thành công việc trong báo cáo sẽ không thể chỉnh sửa trường “Actual date to finish task”

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet
* Chỉ có admin/leader mới có quyền chỉnh sửa deadline sau khi người dùng đã thêm deadline khi tạo báo cáo

**Điều kiện trước:**

* Người dùng đã thêm báo cáo công việc vào hệ thống trước đó

**Điều kiện sau:**

* Nếu thành công: Đóng modal, hiển thị message thông báo và quay trở lại bảng danh sách báo cáo
* Nếu thất bại: Báo cáo sẽ không được chỉnh sửa và giữa nguyên các thông tin trước đó, hệ thống hiển thị thông báo lỗi

### 3.4.8. Ca sử dụng “Đánh giá công việc”

**Mô tả:** Người dùng có quyền sử dụng truy cập để đánh giá tiến độ các báo cáo của người dùng trong hệ thống

**Tác nhân:** Admin, leader

**Luồng sự kiện chính**

* Ca sử dụng bắt đầu khi admin/leader tiến hành đánh giá công việc đã làm của nhân viên
* Tại giao diện bảng danh sách báo cáo mỗi báo cáo sẽ có một button để đánh giá tương ứng
* Hệ thống sẽ hiển thị một modal chứa các thông tin mà admin/leader cần nhập để đánh giá một công việc
* Điểm độ khó: dựa theo độ khó của công việc đang làm : easy(1), medium(2), hard(3), very hard(4)
* Điểm đánh giá: dựa theo tiến độ hoàn thành và kết quả của công việc: insufficient(1), fair(2), good(3), great(4), excellent(5)
* Score: Điểm trung bình của một công việc được tính theo công thức

Score = điểm độ khó\*0.3 + điểm đánh giá\*0.7

* Tiến hành ấn “submit” để hoàn tất việc đánh giá một báo cáo

**Luồng thay thế:**

* Ấn button expand tại bảng danh sách báo cáo để đánh giá cũng như sửa lại thông tin đã đánh giá đó

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet

**Điều kiện trước:**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin/leader

**Điều kiện sau:**

* Nếu thành công: Đóng modal đánh giá và quay trở lại bảng danh sách báo cáo với trường "Evaluation status” của báo cáo đó chuyển thành “Evaluated”
* Nếu thất bại: Thông tin đánh giá báo cáo sẽ không được thêm và hệ thống thông báo lỗi cho người dùng

### 3.4.9. Ca sử dụng “Export báo cáo”

**Mô tả:** Hệ thống cho phép xuất bảng thông tin báo cáo từ bảng danh sách ra file excel

**Tác nhân:** Admin, manager

**Luồng sự kiện chính**

* Ca sử dụng được thực hiện khi xuất danh sách báo cáo ra file excel để có thể dễ dàng trong việc download, in, lưu trữ dữ liệu, ...
* Tại bảng danh sách báo cáo người dùng tiến hành ấn button “export” để xuất ra excel
* Thông tin báo cáo được xuất ra file excel chứa tất cả thông tin của báo cáo
* Người dùng chọn xuất báo cáo theo loại thời gian chọn khoảng thời gian tương ứng
* Báo cáo được xuất ra file với tên có dạng “Name\_Type time\_Time”

**Luồng thay thế:**

* Hiển thị một message thông báo nếu chưa chọn thời gian xuất danh sách báo báo

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet
* Dung lượng lưu trữ trong máy tính muốn xuất file còn đủ dung lượng còn trống

**Điều kiện trước:**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin/manager
* Chọn khoảng thời gian muốn xuất danh sách báo cáo ra file excel

**Điều kiện sau:**

* Nếu thành công: Bảng danh sách được xuất ra file excel
* Nếu thất bại: Hệ thống thông báo lỗi cho người dùng

### 3.4.10. Ca sử dụng “Quản lý báo cáo của nhân viên”

**Mô tả:** Có thể thực hiện các thao tác quản lý đối với hệ thống báo cáo của người dùng khác

**Tác nhân:** Admin, leader

**Luồng sự kiện chính**

* Có hai cách có thể truy cập và quản lý báo cáo của người dùng khác
  + Truy cập từ báo cáo tương ứng với của mỗi nhân viên tại giao diện quản lý người dùng
  + Tại giao diện bảng danh sách báo cáo nhập tên người dùng mình muốn truy cập hệ thống báo cáo của họ tại phần tìm kiếm
* Tại giao diện quản lý báo cáo của người dùng khác người dùng có thể thực hiện các tác vụ như thêm, sửa, xóa báo cáo
  + Admin: Có thể thực hiện các tác vụ quản lý đối với tất cả báo cáo của toàn bộ người dùng trong hệ thống
  + Leader: Chỉ thực hiện tác vụ xóa đối với báo cáo đã chỉ định cho nhân viên

**Luồng thay thế:**

* Hiển thị bảng “No data” nếu chưa có báo cáo trong hệ thống của người dùng

**Yêu cầu đặc biệt:** Không có

**Điều kiện trước:**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin/manager
* Nhân viên đã thêm báo cáo trong hệ thống của họ

**Điều kiện sau:**

* Nếu thành công: Hệ thống thông báo thành công khi thực hiện các tác vụ trên báo cáo của nhân viên
* Nếu thất bại: Hệ thống thông báo lỗi tương ứng với các tác vụ thực hiện

### 3.4.11. Ca sử dụng “Tạo đơn xin nghỉ”

**Mô tả:** Khi người dùng muốn xin nghỉ phép, họ có thể tạo một đơn xin nghỉ rồi gửi lên hệ thống và chờ admin xác nhận lại

**Tác nhân:** Tất cả các tác nhân

**Luồng sự kiện chính**

* Ca sử dụng được thực hiện khi người dùng gửi một đơn xin nghỉ lên hệ thống
* Tại giao diện “Leave”, người dùng nhấn button “Apply leave” để tạo một đơn xin nghỉ
* Hệ thống trả về một modal chứa các thông tin cơ bản của một đơn xin nghỉ
* Người dùng tiến hành điền các thông tin này và ấn button “submit” để gửi đơn xin nghỉ lên hệ thống
* Hệ thống sẽ gửi email chứa toàn bộ thông tin của đơn xin nghỉ tới những người dùng có quyền admin trong hệ thống
* Người dùng sẽ chờ xác nhận và phản hồi lại đơn xin nghỉ này từ phía admin

**Luồng thay thế:**

* Với loại nghỉ phép thường niên(annual leave), mỗi nhân viên chỉ có tối đa 12 ngày nghỉ trong 1 năm. Số ngày nghỉ sẽ bị trừ đi khi admin/leader chấp nhận yêu cầu. Nếu nghỉ quá số ngày cho loại này trong 1 năm sẽ không thể gửi yêu cầu lên hệ thống nữa

**Yêu cầu đặc biệt**

* Hệ thống có kết nối internet
* Hệ thống có kết nối với bên thứ 3 là gmail

**Điều kiện trước:** Không có

**Điều kiện sau**

* Nếu thành công: Hệ thống sẽ người email cho admin để chờ họ xác nhận yêu cầu
* Nếu thất bại:  Hệ thống thông báo lỗi cho người dùng

### 3.4.12. Ca sử dụng “Phản hồi đơn xin nghỉ”

**Mô tả:** Khi nhận được yêu cầu xin nghỉ của nhân viên được gửi tới gmail, quản lý xem xét và xác nhận lại yêu cầu xin nghỉ đó

**Tác nhân:** Admin, leader

**Luồng sự kiện chính**

* Ca sử dụng thực hiện khi admin/leader muốn xác nhận lại đơn xin nghỉ của nhân viên
* Tại giao diện xin nghỉ khi truy cập với quyền admin/leader sẽ hiển thị tất cả danh sách xin nghỉ đã được gửi lên hệ thống
* Khi người dùng ấn vào đơn sẽ hiển thị một drawer bên phải màn hình hiển thị chi tiết thông tin của đơn
* Tại drawer người dùng cho thể chuyển trạng thái của đơn xin nghỉ và gửi lại lời phản hồi(nếu có) và ấn “submit” để xác nhận

**Luồng thay thế**

* Người dùng không thể tự phản hồi đơn xin nghỉ của chính mình. Khi phản hồi đơn xin nghỉ của mình sẽ hiển thị giao diện chỉnh sửa

**Yêu cầu đặc biệt**

* Hệ thống có kết nối internet
* Hệ thống có kết nối với bên thứ 3 là gmail

**Điều kiện trước**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin/leader

**Điều kiện sau**

* Nếu thành công: Hệ thống sẽ gửi email chứa thông tin của đơn xin nghỉ (không bao gồm lý do) đến toàn bộ gmail người dùng trong hệ thống
* Nếu thất bại:  Hệ thống thông báo lỗi cho người dùng

### 3.4.13. Ca sử dụng “Xem báo cáo từ trang Jira software”

**Mô tả:** Chức năng hiển thị những báo cáo công việc được giao trên Jira software

**Tác nhân:** Tất cả các tác nhân

**Luồng sự kiện chính**:

* Tại giao diện “Jira” trong hệ thống khi truy cập lần đầu cần một token. Thực hiện ấn button “Get token” hệ thống sẽ mở giao diện tạo token để xác thực khi sử dụng API của Jira software. Tạo token mới bằng một dãy ký tự bất kỳ
* Quay lại hệ thống điền token vừa tạo được và chọn công ty chứa các dự án mà bạn có quyền truy cập
* Sau khi truy cập thành công, một giao diện hiển thị danh sách các dự án mà bạn có quyền xem
* Khi xem chi tiết thông tin báo cáo trong một dự án, hệ thống sẽ trả về một bảng danh sách các báo với trạng thái ban đầu chứa các báo cáo tại sprint hiện tại của dự án đó
* Người dùng có thể xem thêm thông tin báo cáo này khi click để xem chi tiết
* Người dùng có lọc nếu muốn xem danh sách báo cáo theo: sprint, trạng thái báo cáo hiện tại

**Luồng thay thế**

* Hệ thống hiển thị giao diện yêu cầu quyền truy cập nếu người dùng chưa nhập token hoặc không có quyền truy cập các dự án từ jira software
* Nếu dự án chưa có sprint, hiển thị một thông báo lỗi cho người dùng
* Khi chọn công ty mà tài khoản gmail của người dùng không có quyền truy cập hệ thống sẽ hiển thị một message thông báo lỗi

**Yêu cầu đặc biệt**

* Hệ thống có kết nối internet
* Tài khoản gmail của người dùng là tài khoản đăng nhập trên jira software
* Đã được thêm vào dự án trên jira software

**Điều kiện trước**

* Người dùng nhập token để truy cập hệ thống nếu muốn xem danh sách báo cáo cũng như dự án
* Hệ thống có kết nối với bên thứ ba là Jira software

**Điều kiện sau**

* Nếu thành công: Hệ thống sẽ hiển thị giao diện chứa dự án và danh sách báo cáo từ jira software
* Nếu thất bại:  Hệ thống thông báo lỗi cho người dùng và trở về giao diện yêu cầu quyền truy cập

### 3.4.14. Ca sử dụng “Chỉ định báo cáo từ Jira về hệ thống"

**Mô tả:** Lấy những báo cáo từ jira software về hệ thống báo cáo công việc nội bộ

**Tác nhận:** Tất cả các tác nhân

**Luồng sự kiện chính:**

* Ca sử dụng được thực hiện khi người dùng muốn chỉ định báo cáo này về hệ thống báo cáo công của mình
* Tại bảng danh sách báo cáo trong dự án từ Jira người dùng ấn button “Assign” ứng với báo cáo muốn chỉ định,
* Hệ thống hiển thị một modal chứa thông tin của một báo cáo và người dùng chỉnh sửa thông tin các trường và tạo tương tự như báo cáo thông thường khác
* Disabled button “Assign” khi báo cáo được chỉ định thành công
* Báo cáo sau khi được chỉ định sẽ hiển thị tại bảng danh sách báo cáo với type là “jira”

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt**

* Hệ thống có kết nối internet
* Người dùng đã có quyền truy cập hệ thống xem báo cáo từ jira software
* Hệ thống có kết nối với bên thứ ba là Jira software

**Điều kiện trước:** Không có

**Điều kiện sau**

* Nếu thành công: Hệ thống hiển thị một message thông báo thành công và báo cáo được chỉ định sẽ được thêm vào bảng danh sách báo cáo cá nhân của người chỉ định
* Nếu thất bại:  Đóng modal thêm mới và hệ thống trả về lỗi cho người dùng

### 3.4.15. Ca sử dụng “Thêm mới dự án”

**Mô tả:** Khi công ty có một dự án mới vào người dùng muốn thêm báo cáo công việc khi làm dự án đó

**Tác nhân:** Admin

**Luồng sự kiện chính**

* Ca sử dụng được thực hiện khi admin muốn thêm thông tin một dự án mới vào hệ thống
* Tại giao diện quản lý dự án. Các dự án đã được thêm thông tin trước đó sẽ được hiển thị theo dạng bảng
* Admin ấn button “Add project” để tạo một dự án mới
* Hệ thống trả về một modal chứa các thông tin cơ bản của một dự án
* Admin tiến hành điền các trường thông tin cần thiết và ấn “Submit” để thêm dự án mới vào hệ thống

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet

**Điều kiện trước**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin

**Điều kiện sau**

* Nếu thành công: Dự án được thêm vào cơ sở dữ liệu và người dùng có thể tạo báo cáo công việc trong dự án này
* Nếu thất bại: Đóng modal thêm mới và hệ thống trả về lỗi cho người dùng

### 3.4.16. Ca sử dụng “Đóng/kích hoạt dự án”

**Mô tả:** Khi báo một dự án trong công ty đã hoàn thành và đóng lại hoặc muốn kích hoạt lại dự án đã đóng trước đó

**Tác nhân:** Admin

**Luồng sự kiện chính**

* Ca sử dụng được thực hiện khi người dùng muốn thay đổi trạng thái hoạt động hiện tại của dự án
* Tại giao diện bảng danh sách các dự án hiện tại trên hệ thống, một modal chứa thông tin của dự án sẽ được show khi người dùng ấn vào xem để chi tiết
* Khi click vào trạng thái kích hoạt/đóng dự án, sẽ hiển thị một alert để xác nhận lại hành động của người dùng

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet

**Điều kiện trước**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin

**Điều kiện sau**

* Nếu thành công: Khi đóng một dự án, người dùng sẽ không thể tạo được báo cáo công việc trong dự án đó nữa và ngược lại
* Nếu thất bại: Hệ thống hiển thị thông báo lỗi cho người dùng

### 3.4.17. Ca sử dụng “Xem thống kê báo cáo”

**Mô tả:** Hiển thị thống kê báo cáo công việc đã tạo trong hệ thống giúp những người quản lý có thể nắm bắt tình trạng công việc đã báo cáo lên hệ thống

**Tác nhân:** Admin, leader, manager

**Luồng sự kiện chính**

* Những người có quyền truy cập có thể xem thống kê báo cáo của thành viên khác trong hệ thống thông qua việc tìm kiếm theo tên người dùng
* Bảng thống kê hiệu suất sẽ được hiển thị theo loại đánh giá bao gồm
* Tổng điểm đánh giá báo cáo được phân theo loại mà admin/leader đánh giá theo các khoảng thời gian
* Tổng số báo cáo đã được đánh giá
* Biểu đồ tròn chứa số lượng báo cáo được phân dựa vào độ khó của chúng
* Người dùng có thể ấn chọn thời gian để xem bảng hiệu suất và biểu đồ tròn tại các khoảng thời gian khác như tuần, tháng, năm
* Người truy cập có thể xem danh sách những báo cáo công việc chưa được đánh giá xếp theo thời gian sớm nhất. Ấn vào báo cáo để thực hiện đánh giá báo cáo đó trực tiếp trong giao diện thống kê

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet

**Điều kiện trước**

* Người dùng đăng nhập vào hệ thống với tài khoản mang quyền admin/leader/manager

**Điều kiện sau**: không có

### 3.4.18. Ca sử dụng “Quản lý file của người dùng”

**Mô tả:** Người dùng muốn xem lại hoặc thực hiện các thao tác vớ những file trong drive cá nhân của mình và những file đã đính kèm theo báo cáo khi tạo

**Tác nhân:** Admin, leader, develop

**Luồng sự kiện chính**

* Tại giao diện quản lý file drive hệ thống sẽ hiển thị danh sách file được phân loại
* Người dùng muốn xem chi tiết thư mục hoặc file hệ thống sẽ link tới giao diện hiển thị file của google drive
* Tại thư mục “Task Management” trong giao diện quản lý file chứa tất cả file người dùng đã kèm theo khi tạo báo cáo và được phân theo thư mục với tên chính là “main task” của báo cáo
* Người dùng có thể thêm, xóa hoặc sửa file trực tiếp trong thư mục báo cáo trên drive, khi đó file mà đính kèm theo trong báo cáo này cũng sẽ tương ứng được cập nhật theo
* Hệ thống hiển thị dung lượng hiện tại người dùng có và dung lượng đã sử dụng trên google drive

**Luồng thay thế:** Không có

**Yêu cầu đặc biệt:**

* Hệ thống có kết nối internet
* Hệ thống có kết nối với bên thứ ba là Google Drive
* Google drive còn dung lượng trống nếu muốn thêm update thêm file mới

**Điều kiện trước**

* Đã đính kèm file vào báo cáo nếu muốn thực hiện xem hoặc thực hiện các tác vụ khác đối với file trong báo cáo đó

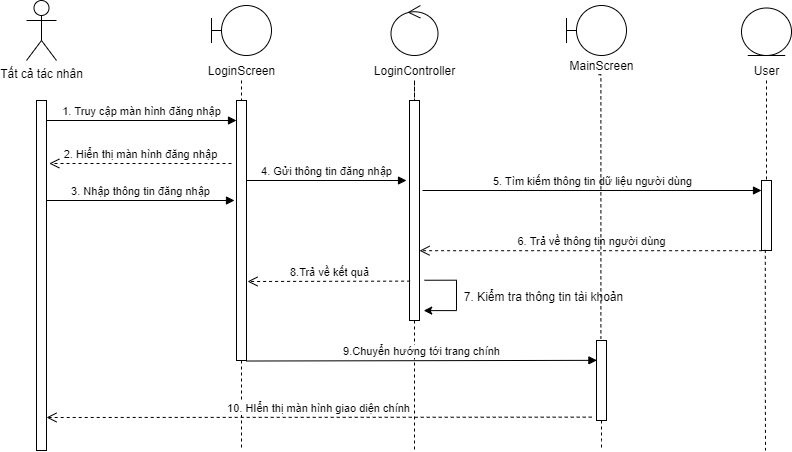
**Điều kiện sau**

* Nếu thành công: File trong google drive sẽ được cập nhật tương ứng với các thao tác trên hệ thống của người dùng
* Nếu thất bại: Danh sách file vẫn được dữ nguyên và hệ thống hiển thị thông báo lỗi cho người dùng

## 3.5. Xây dựng biểu đồ tuần tự

Trong phần này, tôi sẽ trình bày biểu đồ tuần tự để xác định các trình tự diễn ra sự kiện đã được nêu trong phần “Đặc tả ca sử dụng”. Mỗi ca sử dụng sẽ được mô tả bằng một biểu đồ tuần tự tương ứng và dưới đây là biểu đồ tuần tự một số ca sử dụng của hệ thống.

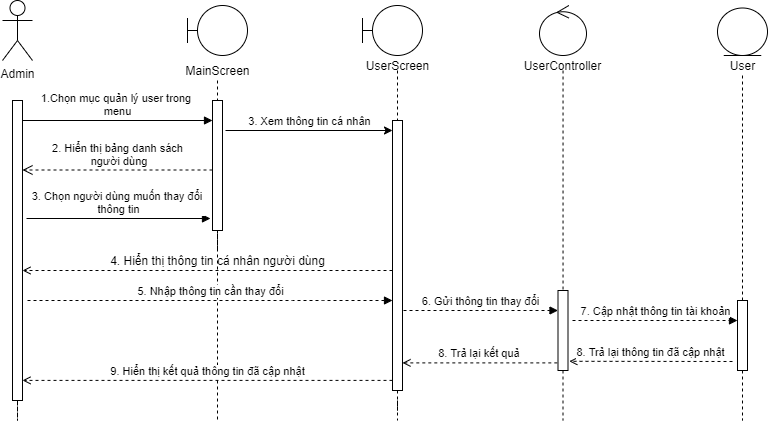
### 3.5.1. Biểu đồ tuần tự ca sử dụng “Đăng nhập”



*Hình 3.5* Sơ đồ tuần tự ca sử dụng “Đăng nhập”

Sau khi truy cập vào trang web, hệ thống trả về màn hình đăng nhập cho người dùng. Người dùng chọn “Google” trang web được chuyển hướng tới màn hình đăng nhập bằng google, người nhập gmail truy cập của mình và chấp nhận các yêu cầu xác thực từ google. Một yêu cầu sẽ được gửi tới máy chủ, máy chủ kiểm tra các thông tin tài khoản sau kết quả sẽ được trả lại phía máy khách. Khi đăng nhập thành công, mà hình sẽ được chuyển hướng tới giao diện chính của hệ thống.

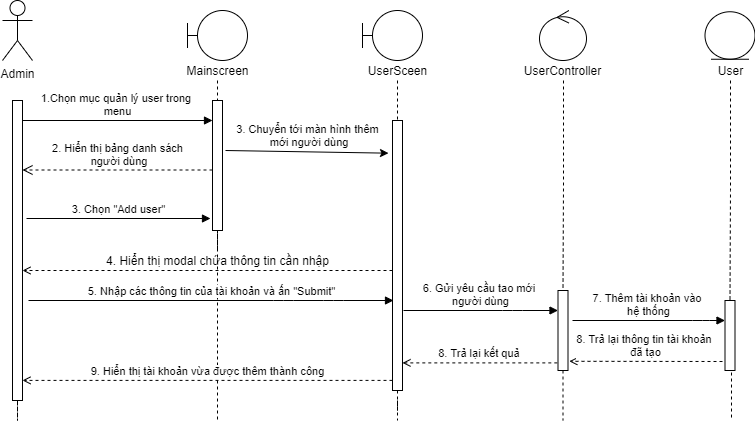
### 3.5.2. Biểu đồ tuần tự ca sử dụng “Thay đổi thông tin người dùng”



*Hình 3.6* Sơ đồ tuần tự ca sử dụng “Thay đổi thông tin người dùng”

Sau khi truy cập vào hệ thống, admin chọn phần quản lý người dùng trên danh mục thanh menu bên phải. Chọn “User management” để hiển thị màn hình danh sách người dùng. Admin chọn người dùng mà mình muốn cập nhật thông tin. Màn hình thông tin cá nhân của người dùng đó sẽ được hiển thị, admin tiếng hành điền các thông tin cập nhận sau đó ấn “Submit”. Lúc này, một yêu cầu thay đổi thông tin chứa thông tin các trường thay đổi sẽ được gửi tới phía máy chủ. Máy chủ thực hiện quá trình sử lý thay đổi và cập nhật thông tin vào cơ sở dữ liệu sau đó gửi lại thông tin đã cập nhật cho máy khách. Thông tin người dùng được cập nhật tại bảng danh sách và hiển thị một thông báo thành công trên màn hình.

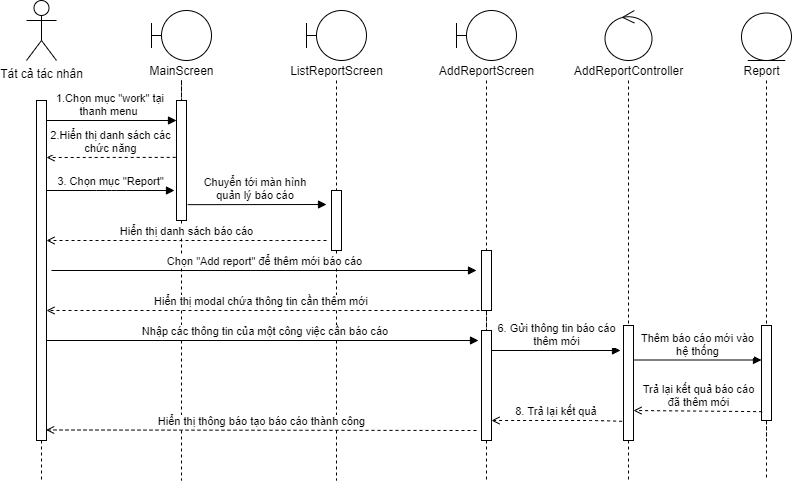
### 3.5.3. Biểu đồ tuần tự ca sử dụng “Thêm tài khoản”



*Hình 3.7* Sơ đồ tuần tự ca sử dụng “Thêm tài khoản”

Sau khi truy cập vào hệ thống, admin chọn phần quản lý người dùng trên danh mục thanh menu bên phải. Chọn “User management” để hiển thị màn hình danh sách người dùng. Admin chọn button “Add user” để hiển thị màn hình thêm mới thông tin một tài khoản, điền các trường thông tin cần thiết và sau đó ấn “Submit”. Khi đó, một request chứa thông tin tài khoản thêm mới sẽ được gửi đến phía máy chủ. Máy chủ thêm mới tài khoản này vào Firebase Admin và cơ sở dữ liệu danh sách người dùng sau đó gửi lại thông tin tài khoản đã thêm mới thành công cho máy khách. Thông tin người dùng được cập nhật tại bảng danh sách và hiển thị một thông báo nếu ca sử dụng này được thực hiện thành công.

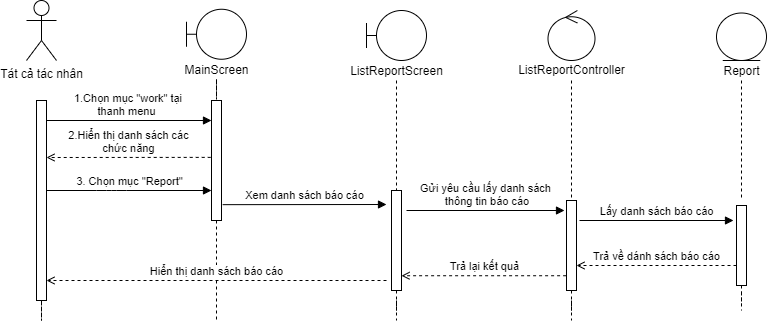
### 3.5.4. Biểu đồ tuần tự ca sử dụng “Thêm mới báo cáo”



*Hình 3.8* Sơ đồ tuần tự ca sử dụng “Thêm báo cáo mới”

Người dùng chọn mục “Work” trên thanh menu và truy cập phần “Report”, hệ thống sẽ trả về bảng danh sách báo cáo mà người dùng đã thêm trước đó. Tại màn hình danh sách báo cáo, người dùng chọn button “Add report” để nhập thông tin của một công việc cần báo cáo lên hệ thống.  Sau khi điền đầy đủ các thông tin của báo cáo người dùng tiến hành ấn “Submit” để gửi yêu cầu tạo mới báo cáo. Máy chủ nhận các trường thông tin này của báo cáo và lưu vào cơ sở dữ liệu, phần file đính kèm theo báo cáo sẽ được lưu vào google drive. Sau đó máy chủ trả về báo cáo này,  thêm mới vào màn hình danh sách và hiển thị một thông báo cho người dùng biết báo cáo đã được thêm mới thành công.

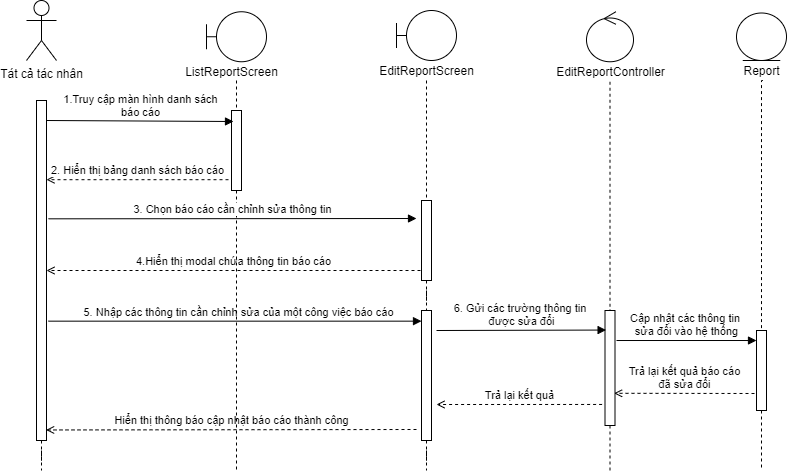
### 3.5.5. Biểu đồ tuần tự ca sử dụng “Xem danh sách báo cáo”



*Hình 3.9* Sơ đồ tuần tự ca sử dụng “Xem danh sách báo cáo”

Ca sử dụng này bắt đầu khi người dùng truy cập vào màn hình “Report” hiển thị danh sách báo cáo tại phần “Work” trên thanh menu.  Một yêu cầu lấy danh sách báo cáo của người dùng sẽ được gửi tới phía máy chủ. Máy chủ xử lý yêu cầu và trả về danh sách báo cáo của người dùng yêu cầu. Trong giao diện bảng danh sách báo cáo, người dùng có thể lọc để xem theo các trường thông tin của báo cáo, khoảng thời gian hiển thị, số lượng báo cáo trong một bảng. Khi đó yêu cầu lọc theo nhu cầu xem sẽ được gửi tới hệ thống và trả về phía máy khách bảng danh sách báo cáo tương ứng.

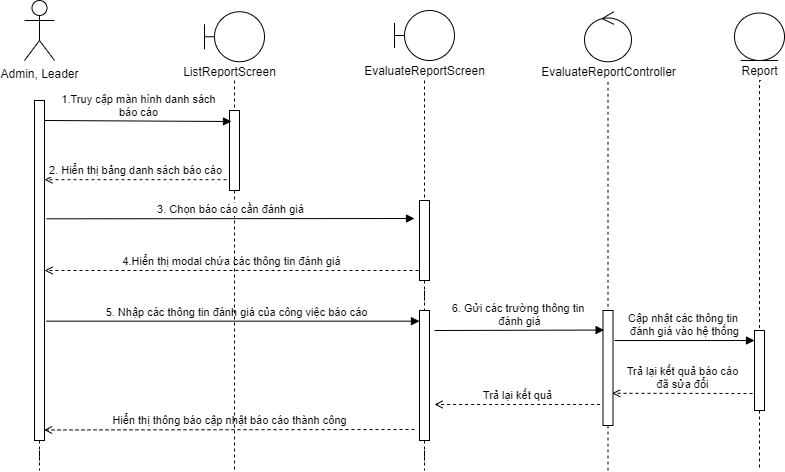
### 3.5.6. Biểu đồ tuần tự ca sử dụng “Sửa báo cáo”



*Hình 3.10* Sơ đồ tuần tự ca sử dụng “Sửa báo cáo”

Sau khi người dùng cập nhật một báo cáo lên hệ thống, ca sử dụng này cho phép người đó hoặc những người có quyền khác chỉnh sửa thông tin báo cáo này. Sau khi truy cập phần quản lý báo cáo. Hệ thống sẽ trả về danh sách báo cáo công việc mà người dùng đã thêm trước đó. Chọn báo cáo cần cập nhật thông tin, hệ thống sẽ hiển thị màn hình chi tiết thông tin báo cáo đó. Cập nhật các trường thông tin cần thiết và ấn “Submit”, lúc này một yêu cầu chỉnh sửa báo cáo sẽ được gửi tới máy chủ, yêu cầu sau khi được xử lý sẽ trả về một thông báo thành công và cập nhật những thay đổi tại bảng danh sách báo cáo.

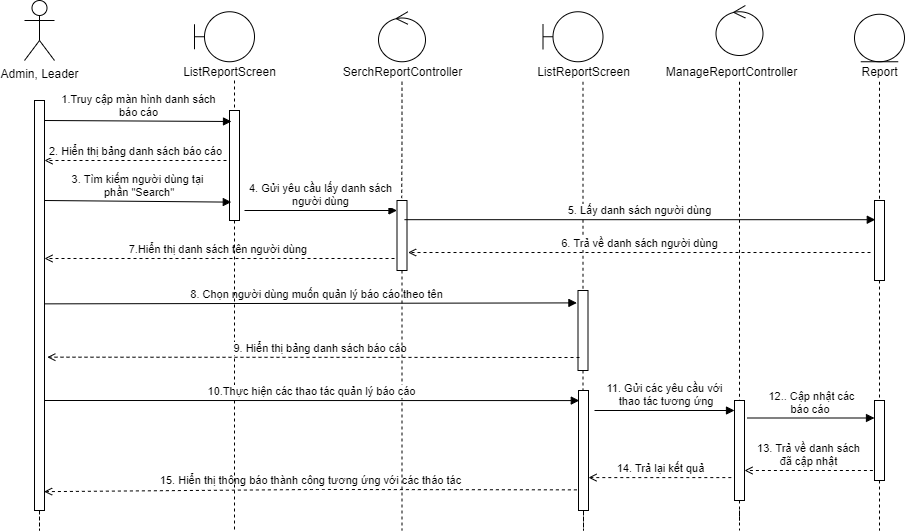
### 3.5.7. Biểu đồ tuần tự ca sử dụng “Đánh giá công việc”



*Hình 3.11* Sơ đồ tuần tự ca sử dụng “Đánh giá công việc”

Admin, leader sau khi truy cập màn hình danh sách báo cáo có thể nhập tên người dùng tại mục “Search”, máy chủ trả về bảng danh sách báo cáo theo tên người dùng. Tại bảng người dùng ấn button “Evaluate” tương ứng với mỗi báo cáo, hệ thống sẽ hiển thị các trường đánh giá. Người dùng điền các đánh giá công việc tương ứng và ấn “Submit”. Lúc này một yêu cầu cập nhật thông tin đánh giá báo cáo sẽ được gửi lên hệ thống. Máy chủ xử lý yêu cầu cập nhật đánh giá và trả thông tin về cho máy khách. Khi thành công, trạng thái “waiting” của báo cáo trong bảng danh sách sẽ được chuyển thành “evaluated”.

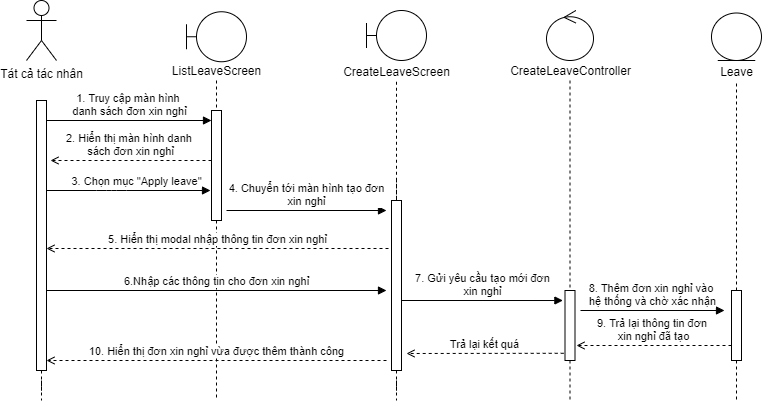
### 3.5.8. Biểu đồ tuần tự ca sử dụng “Quản lý báo cáo của nhân viên”



*Hình 3.12* Sơ đồ tuần tự ca sử dụng “Quản lý báo cáo của nhân viên”

Admin, leader sau khi truy cập màn hình danh sách báo cáo có thể nhập tên người dùng tại mục “Search”, máy chủ trả về bảng danh sách báo cáo theo tên người dùng. Tại bảng danh sách báo cáo của người dùng admin và leader có thể tạo, sửa, export báo cáo của người dùng. Tại đây nếu muốn xoá một báo cáo trong bảng danh sách báo cáo của người dùng, tài khoản bạn đăng nhập phải mang quyền admin, nếu đăng nhập với quyền leader chỉ có thể xóa được các báo cáo đã chỉ định cho người dùng này. Khi thực hiện các thao tác với danh sách báo cáo của người dùng, một yêu cầu tương ứng với thao tác thực hiện sẽ được gửi tới phía máy chủ. Khi nào máy chủ sẽ xử lý yêu cầu tương ứng và trả lại thông báo kết quả tại màn hình máy khách.

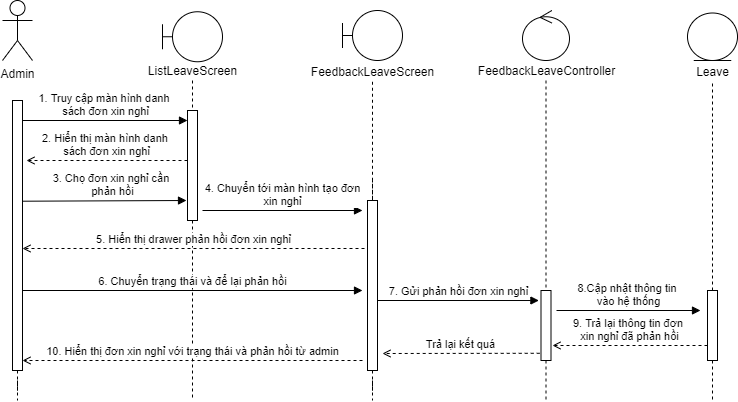
### 3.5.9. Biểu đồ tuần tự ca sử dụng “Tạo đơn xin nghỉ”



*Hình 3.13* Sơ đồ tuần tự ca sử dụng “Tạo đơn xin nghỉ”

Ca sử dụng này bắt đầu khi người dùng muốn tạo một yêu cầu xin nghỉ phép gửi lên hệ thống. Truy cập vào phần “leave” tại thanh menu, đối với admin hệ thống sẽ trả lại danh sách tất cả các yêu cầu xin nghỉ của người dùng trong hệ thống, còn với tài khoản khác chỉ hiển thị danh sách đơn xin nghỉ của cá nhân tài khoản đó. Ấn “Add leave” hệ thống sẽ hiển thị một màn hình tạo đơn xin nghỉ, người dùng nhập chính xác các thông tin và ấn “Submit” để gửi yêu cầu xin nghỉ lên hệ thống. Máy chủ sẽ nhận thông tin đơn xin nghỉ này và thêm vào cơ sở dữ liệu với trạng thái “Waiting” đồng thời gửi email tới những tài khoản mang quyền “Admin” trong hệ thống và chờ họ xác nhận yêu cầu xin nghỉ của người dùng.

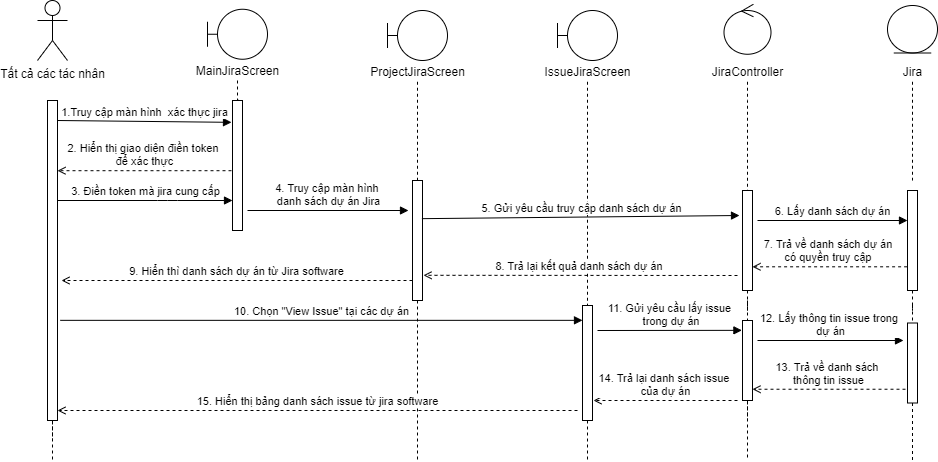
### 3.5.10.  Biểu đồ tuần tự ca sử dụng “Phản hồi đơn xin nghỉ”



*Hình 3.14 Sơ đồ tuần tự ca sử dụng “Phản hồi đơn xin nghỉ”*

Ca sử dụng này bắt đầu khi Admin muốn phản hồi lại yêu cầu xin nghỉ phép của người dùng khác. Truy cập vào phần “leave” tại thanh menu, hệ thống sẽ trả về danh sách các yêu cầu xin nghỉ mà người dùng đã gửi lên. Hoặc khi một yêu cầu xin nghỉ được gửi sẽ có một email gửi tới gmail cá nhân, ở đó có đường dẫn tới đơn xin nghỉ mà người dùng gửi lên hệ thống. Admin ấn vào đơn xin nghỉ, một màn hình yêu cầu điền “feedback” sẽ được hiển thị, admin chuyển trạng thái đơn xin nghỉ tương ứng với phản hồi của bản thân, để lại một phản hồi ngắn nếu cần thiết và ấn “Submit” để gửi phản hồi lên hệ thống. Khi đó, một yêu cầu sẽ được gửi tới máy chủ, máy chủ cập nhật đơn xin nghỉ và gửi lại thông tin cho máy khách. Nếu đơn xin nghỉ được chấp nhận, trạng thái “waiting” sẽ được chuyển thành “approved” đồng thời gửi email tới gmail của tất cả người dùng trong hệ thống.

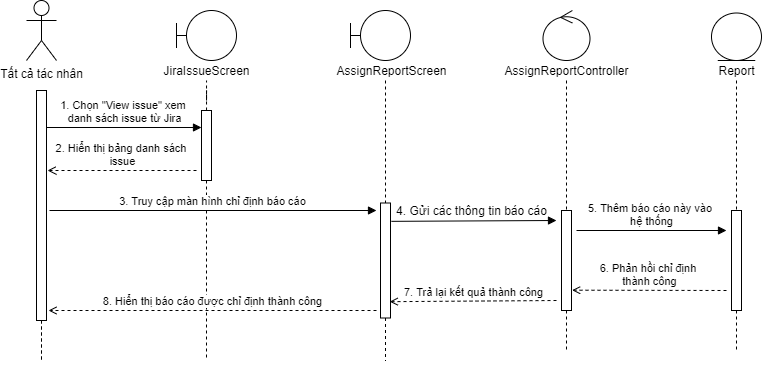
### 3.5.11.  Biều đồ tuần tự ca sử dụng “Xem báo cáo từ Jira software”



*Hình 3.15* Sơ đồ tuần tự ca sử dụng “Xem báo cáo từ Jira software”

Ca sử dụng này bắt đầu khi người dùng truy cập vào màn hình “Jira” hiển thị danh sách báo cáo từ Jira software tại phần “Work” trên thanh menu. Khi lần đầu truy cập, bạn phải nhập một “Token” để xác thực từ phía Jira software. Tại màn hình xác thực ấn button “Get token”, hệ thống sẽ chuyển tới màn hình tạo mã token của Jira, người dùng nhập dãy ký tự bất kỳ để tạo một mã token. Quay trở lại hệ thống dùng mã token vừa tạo được để điền vào “Your jira token” và chọn công ty chứa báo cáo của bạn trên Jira software sau đó ấn “Submit”. Khi đó hệ hệ thống sẽ gửi một yêu cầu tới phía Jira, Jira kiểm tra yêu cầu và trả lại màn hình hiển thị danh sách dự án mà gmail của bạn có quyền truy cập. Người dùng ấn “View issue”, một yêu cầu xem thông tin báo cáo theo dự án sẽ được gửi tới phía Jira. Khi đó sẽ chuyển tới màn hình hiển thị danh sách báo cáo mà người dùng đang được giao trên trang Jira software.

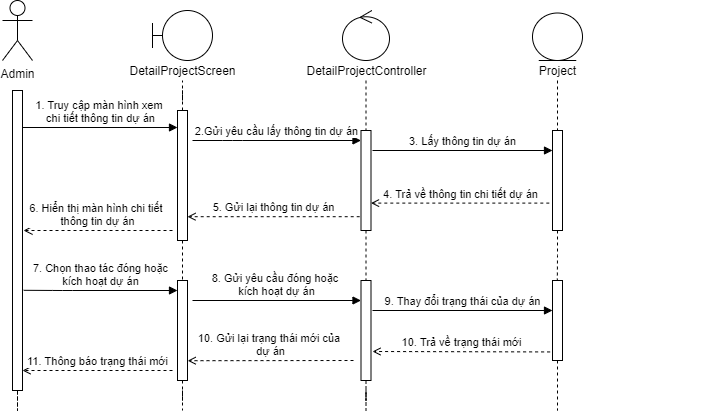
### 3.5.12.  Biểu đồ tuần tự ca sử dụng “Chỉ định báo cáo từ Jira về hệ thống”



*Hình 3.16* Sơ đồ tuần tự ca sử dụng “Chỉ định báo cáo từ Jira software về hệ thống”

Sau khi truy cập thành công để xem danh sách báo cáo từ Jira, hệ thống sẽ hiển thị màn hình danh sách các công việc người dùng đang được giao. Người dùng ấn “Assign” một màn hình hiển thị thông tin những trường tương ứng với thông tin báo cáo được tạo bình thường tại hệ thống. Người dùng điền những trường thông tin còn thiếu và ấn “Assign to report” để chỉ định báo cáo này về hệ thống báo cáo cá nhân trong hệ thống. Máy chủ nhận các trường thông tin này của báo cáo và lưu vào cơ sở dữ. Sau đó máy chủ thêm mới vào màn hình “Report” và hiển thị một thông báo cho người dùng biết báo cáo đã được chỉ định thành công. Những báo cáo khi được chỉ định về hệ thống báo cáo cá nhân, máy chủ sẽ dùng hàm để kiểm tra và chuyển button “Assign” thành “Assigned” và không để chỉ định báo cáo đó nữa.

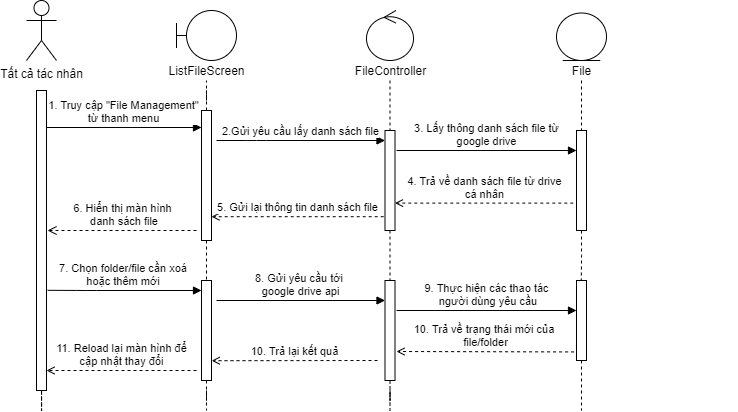
### 3.5.13.  Biểu đồ tuần tự ca sử dụng “Đóng/kích hoạt dự án”



*Hình 3.17* Sơ đồ tuần tự ca sử dụng “Đóng/kích hoạt dự án”

Ca sử dụng này bắt đầu khi Admin muốn kích hoạt hoặc đóng một dự án trong hệ thống. Sau khi truy cập màn hình danh sách dự án, một yêu cầu được gửi về phía máy chủ, máy chủ nhận được yêu cầu sẽ lấy danh sách các dự án trong cơ sở dữ liệu để trả về màn hình giao diện. Admin chọn dự án muốn thực hiện ca sử dụng này, hệ thống sẽ trả về màn hình hiển thị chi tiết thông tin dự án. Tại đây, admin chuyển trạng thái button switch trạng thái hoạt động của dự án tương ứng với yêu cầu và ấn "Submit". Sau đó một yêu cầu sẽ được gửi đến máy chủ và cập nhật trạng thái hoạt động của dự án vào cơ sở dữ liệu. Sau đó máy chủ trả về một thông báo, máy khách hiển thị thông báo này trên màn hình danh sách dự án.

### 3.5.14.  Biều đồ tuần tự ca sử dụng “Quản lý file drive”



*Hình 3.18* Sơ đồ tuần tự ca sử dụng “Quản lý file Drive”

Người dùng truy cập màn hình quản lý file. Một yêu cầu sẽ được gửi tới phía máy chủ, máy chủ dùng token máy khách gửi lên để kết nối với Google drive. Kết nối thành google drive api sẽ gửi về cho máy chủ danh sách file và thông tin dung lượng ở drive cá nhân tài khoản người dùng, sau đó máy chủ sẽ gửi danh sách file này về cho máy khách, máy khách lọc file theo loại và trạng thái và hiển thị danh sách ở giao diện quản lý file. Tại đây người dùng có thể thực hiện các thao tác xóa hoặc export các file này, người dùng cũng có thể ấn vào file để truy cập tới giao diện google drive của file đó để thực hiện thêm các thao tác với file.

Đối với những file đính kèm theo báo cáo của người dùng được hệ thống lưu trong thư mục "Task management" và chia theo thư mục tên là báo cáo của người dùng. Khi người dùng thực hiện các thao tác với này thì các file đính kèm trong các báo cáo cũng được cập nhật tương ứng.

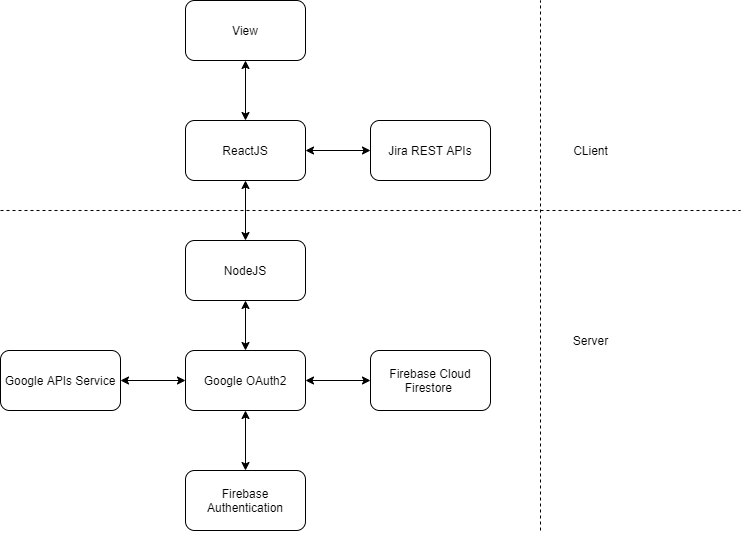
**Kết chương**

Trong chương ba này, tôi đã trình bày chi tiết về các yêu cầu chức năng và yêu cầu phi chức năng của hệ thống. Đồng thời, tôi đã xác định các đối tượng sử dụng khác nhau, nêu ra các chức năng phục vụ đầy đủ các yêu cầu được đặt ra. Khi đã có được các chức năng tôi đi đến đặc tả chi tiết rồi biểu diễn dưới dạng biểu đồ tuần tự giúp trực quan hóa và có cái nhìn rõ ràng hơn về hệ thống cũng như luồng xử lý các chức năng.

# THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG

Chương này sẽ trình bày về kiến trúc hệ thống, cũng như thiết kế cơ sở dữ liệu và miêu tả cách thức cài đặt phía máy chủ và máy khách.

## 4.1. Thiết kế kiến trúc



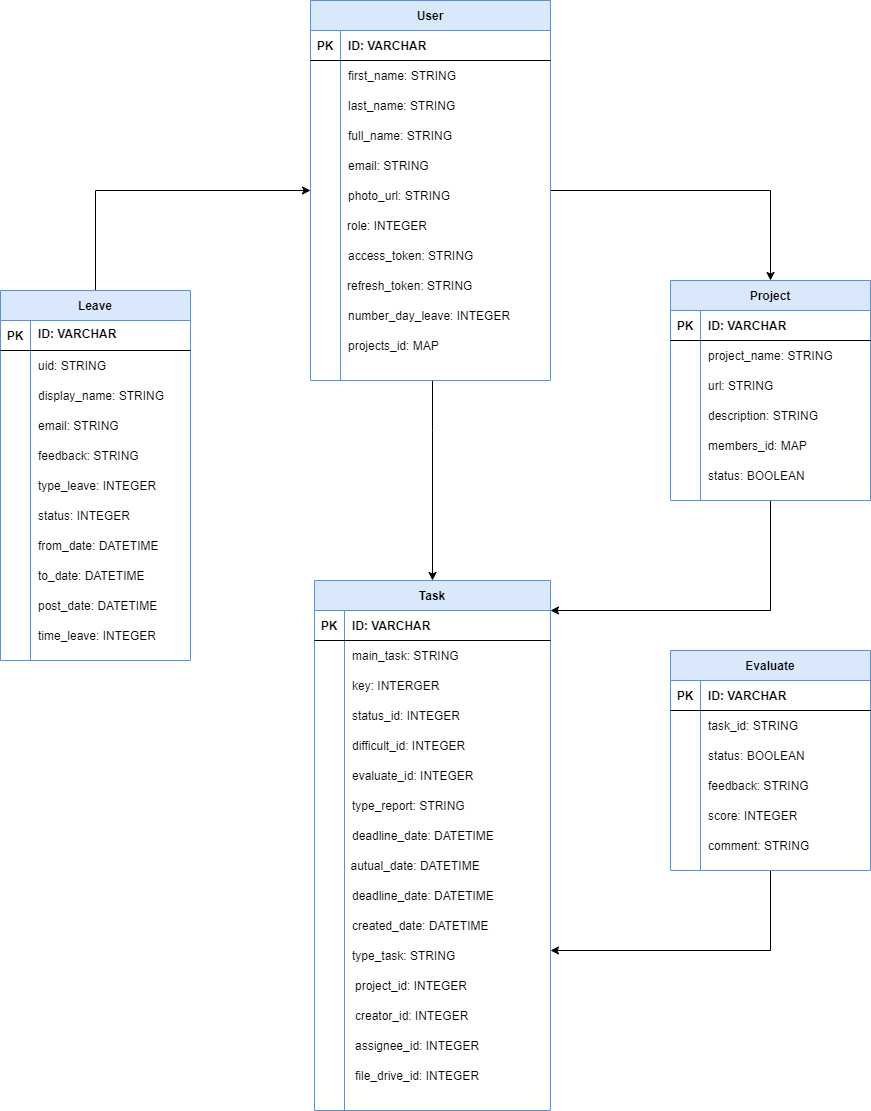
*Hình 4.1* Mô tả thiết kế hệ thống

Hệ thống hỗ trợ phát triển công cụ hỗ trợ quản lý dự án và công việc nội bộ được xây dựng dựa trên các phương pháp đã đề cập tại Chương 3. Hệ thống được xây dựng dựa trên mô hình client – server và sử dụng Firebase để lưu trữ thông tin. Phía xây dựng giao diện tương tác người dùng hệ thống được xây dựng bằng thư viện ReactJS. Máy chủ được xây dựng trên nền tảng NodeJS, thực hiện các yêu cầu từ phía client và thao tác xử lý dữ liệu với Firebase.

Client giao tiếp với server thông qua các Restful API và server thông qua Google Oauth2 để xác thực thông tin người dùng. Client dùng token nhận được khi đăng nhập xác thực thông tin để gửi tới phía Server và dùng token này truy cập các dịch vụ API mà Google cung cấp

## 4.2. Thiết kế cơ sở dữ liệu

Phần này sẽ mô tả cấu trúc cơ sở dữ liệu cũng như trình bày mô tả chi tiết các bảng có nhiệm vụ chính của hệ thống. Hình dưới đây sẽ trình bày về cấu trúc, quan hệ các bảng trong cơ sở dữ liệu.



*Hình 4.2* Cấu trúc cơ sở dữ liệu

Hệ thống sử dụng Firebase để lưu trữ cơ sở dữ liệu, dưới đây tôi sẽ trình bày mô tả chi tiết dữ liệu các bảng trong hệ thống.

|  |  |
| --- | --- |
| **Tên bảng** | **Mô tả** |
| User | Dữ liệu người dùng trong hệ thống |
| Leave | Dữ liệu đơn xin nghỉ |
| Project | Dữ liệu dự án |
| Task | Dữ liệu báo cáo công việc |
| Evaluation | Dữ liệu đánh giá báo cáo |

*Bảng 4.1 Danh sách các bảng trong cơ sở dữ liệu*

***Mô tả bảng “User”***

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | String | Định danh người dùng |
| First\_name | String | Tên của người dùng |
| Last\_name | String | Họ của người dùng |
| Full\_name | String | Tên đầy đủ của người dùng |
| Email | String | Email truy cập của người dùng |
| Photo\_url | String | Link ảnh đại diện |
| Role | Integer | Quyền của người dùng trong hệ thống |
| Access\_token | String | Mã truy cập api của Google |
| Refresh\_token | String | Mã làm mới Access\_token |
| Number\_day\_leave | Integer | Số ngày nghỉ còn lại của người dùng |
| Project\_id | Map | Định danh những dự án người dùng tham gia |

*Bảng 4.2 Bảng cơ sở dữ liệu người dùng*

***Mô tả bảng “Project”***

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | String | Định danh dự án |
| Project\_name | String | Tên |
| Url | String | Link sản phẩm dự án |
| Description | String | Mô tả về dự án |
| Members\_id | Map | Định danh về những người dùng tham gia dự án |
| Status | Boolean | Trạng thái hoạt thộng của dự án |

*Bảng 4.3 Bảng cơ sở dữ liệu dự án*

***Mô tả bảng “Task”***

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | String | Định danh về công việc báo cáo |
| Main\_task | String | Mô tả ngắn về công việc |
| Key | Integer | Mã báo cáo công việc |
| Status\_id | Integer | Định danh về trạng thái công việc |
| Difficult\_id | Integer | Độ khó của công việc báo cáo |
| Type\_report | String | Loại báo cáo |
| Type\_task | String | Loại công việc báo cáo |
| Evaluate\_id | Integer | Định danh đánh giá báo cáo công việc |
| Deadline\_date | Datetime | Thời hạn hoàn thành công việc |
| Created\_date | Datetime | Thời gian tạo báo cáo |
| Actual\_date | Datetime | Thời gian hoàn thành công việc trong thực tế |
| Projetc\_id | Integer | Định danh dự án chứa công việc báo cáo |
| Creator\_id | Integer | Định danh người tạo báo cáo |
| Assignee\_id | Integer | Định danh người được chỉ định báo cáo |
| File\_drive\_id | String | Định danh tới Google drive |

*Bảng 4.4 Bảng cơ sở dữ liệu báo cáo công việc*

***Mô tả bảng “Evaluate”***

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Id | String | Định danh đánh giá |
| Task\_id | String | Định danh công việc được đánh giá |
| Status | Boolean | Trạng thái đánh giá của báo cáo |
| Feedback | String | Phản hổi từ người đánh giá |
| Comment | String | Bình luận từ người làm công việc báo cáo |

*Bảng 4.5 Bảng cơ sở dữ liệu đánh giá nhân viên*

***Mô tả bảng “Leave”***

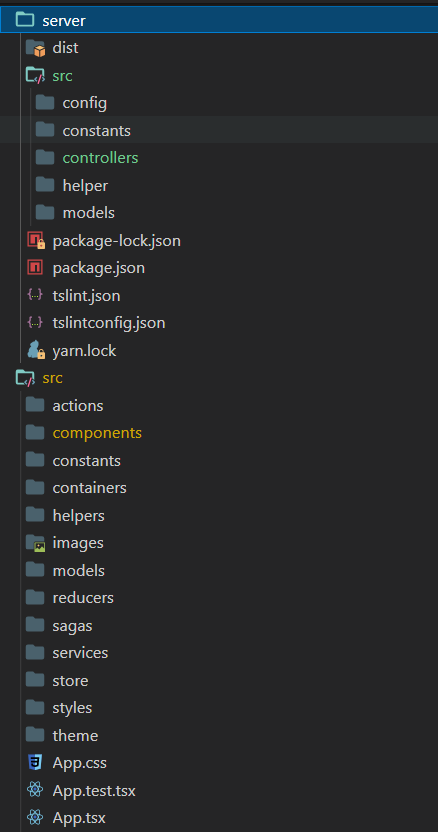
|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Mô tả** |
| Uid | String | Định danh người dùng xin nghỉ |
| Display\_name | String | Tên người dùng xin nghỉ |
| Email | String | Email người dùng xin nghỉ |
| Type\_leave | Integer | Loại đơn xin nghỉ |
| Status | Integer | Trạng thái hiện tại của đơn xin nghỉ |
| From\_date | Datetime | Thời gian bắt đầu nghỉ |
| To\_date | Datetime | Thời gian kết thúc nghỉ |
| Post\_date | Datetime | Thời gian gửi yêu cầu xin nghỉ |
| Time\_leave | Integer | Tổng thời gian nghỉ |

*Bảng 4.6 Bảng cơ sở dữ liệu xin nghỉ*

## 4.3. Cài đặt hệ thống

Dựa trên những phân tích và thiết kế hệ thống ở chương bốn, chương này sẽ giới thiệu cách triển khai và kiểm thử hệ thống phát triển công cụ hỗ trợ quản lý dự án và công việc nội bộ. Cấu trúc mã nguồn hệ thống được tích hợp cả client và server trên cùng một thư mục dự án. Điều này giúp cho việc thao tác, cũng như lập trình các thành phần chức năng của hệ thống trở lên thuận tiện hơn.

### 4.3.1. Xây dựng cấu trúc hệ thống



*Hình 4.3* Cấu trúc thư mục ứng dụng

Hệ thống được xây dựng phía máy chủ dựa trên nền tảng NodeJS, phía máy khách được dựa trên framework ReactJS được chia cấu trúc làm hai phần chính

1. ***Cấu trúc phía máy khách***

Phần xây dựng cấu trúc của máy khách được nằm trong thư mục “src”, chi tiết các mục như sau

* **actions**: Chứa các file actions của redux
* **components**: Chứa các file component của React, thư mục này thực hiện nhiệm vụ chủ yếu liên quan đến việc render HTML ra trình duyệt
* **constants**: Chứa các biến hằng số được sử dụng nhiều lần trong hệ thống
* **containers**: Thư mục này để xác định một component cha bao bọc tất cả component thành phần bên trong nó. Trong này sẽ gọi qua bên store để nhận về các thành phần hiển thị tương ứng. Việc kết nối với redux giữa state và props actions cũng được xác định tại đây, các components con chỉ việc dùng lại  từ lớp container này.
* **helpers**:Mục đích thư mục này là define sẵn một số function mà app có thể import vào và sử dụng lại
* **images**: Thư mục này chứa file để import ảnh từ mục public
* **models**: Hệ thống được viết dựa trên Typescripts, mục này chứa các file định nghĩa các interface có cấu trúc nhất định, chúng được sử dụng làm một công cụ để tạo sự ràng buộc, quy ước giữa các thành phần tương tác trong hệ thống.
* **reducers**: Ứng dụng react + redux ko thể thiếu reducers được, Vậy nên thư mục này chứa tất cả các file liên quan đến reducer của hệ thống.
* **sagas**: Vì hệ thống sử dụng redux-saga nên đây là nơi chứa các hàm watch của sagas để tiện cho việc tìm kiếm và kế thừa
* **services**: Đây là nơi chứa cái Api để phía máy khách có thể gọi để lấy dữ liệu từ máy chủ trả về
* **styles**: Chứa các file css theo các component tương ứng

1. **Cấu trúc phía máy chủ**

Phía máy chủ sử dụng NodeJS để kết nối với firebase cũng như các api services khác của google và được nằm trong thư mục “server”. Chức năng thực hiện các thư mục này tương tự giống chức năng các thư mục cùng tên phía client.

# TRIỂN KHAI THỰC TẾ

Trong chương này tôi sẽ trình bày về các thông tin liên quan đến môi trường phát triển của hệ thống và kết quả triển khai hệ thống trên thực tế.

Hệ thống được phát triển bằng công cụ lập trình là Visual Studio Code. Hệ thống sử dụng nền tảng Nodejs để kết nối với cơ sở dữ liệu Firebase

## 5.1. Môi trường triển khai

* Cài đặt NodeJs
* Cài đặt NodeJs tại đây: <https://nodejs.org/en/download/>
* Cài đặt ReactJS
* Hướng dẫn cài đặt tài đây:

* <https://reactjs.org/docs/create-a-new-react-app.html>

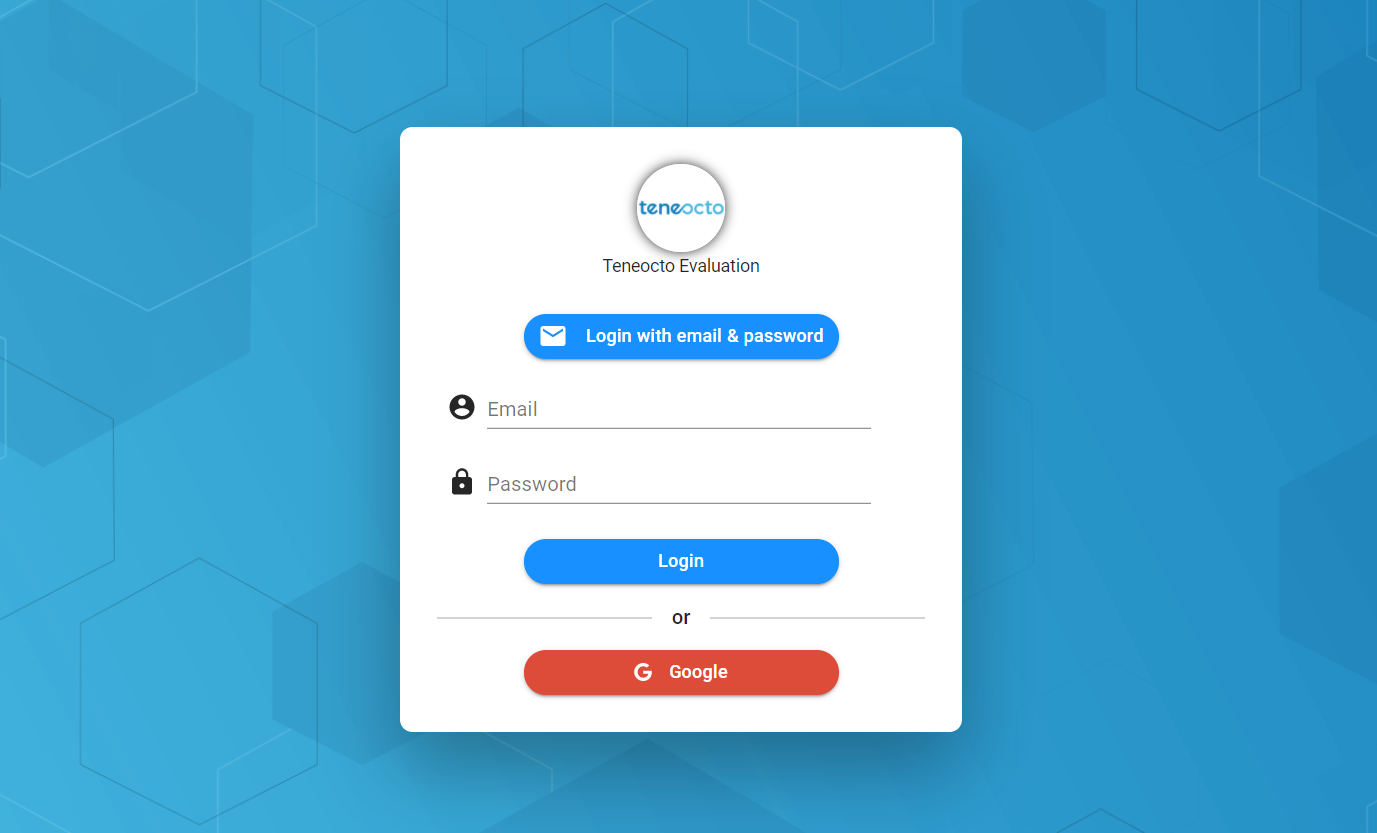
Các bước để chạy dự án

* Trỏ tới thư mục chứa dự án và mở cửa sổ terminal
* Nhập lệnh “npm install”
* Nhập lệnh “npm start”

## 5.2 Kết quả triển khai

Sau đây là màn hình chính của trang web trong môi trường phát triển thực tế

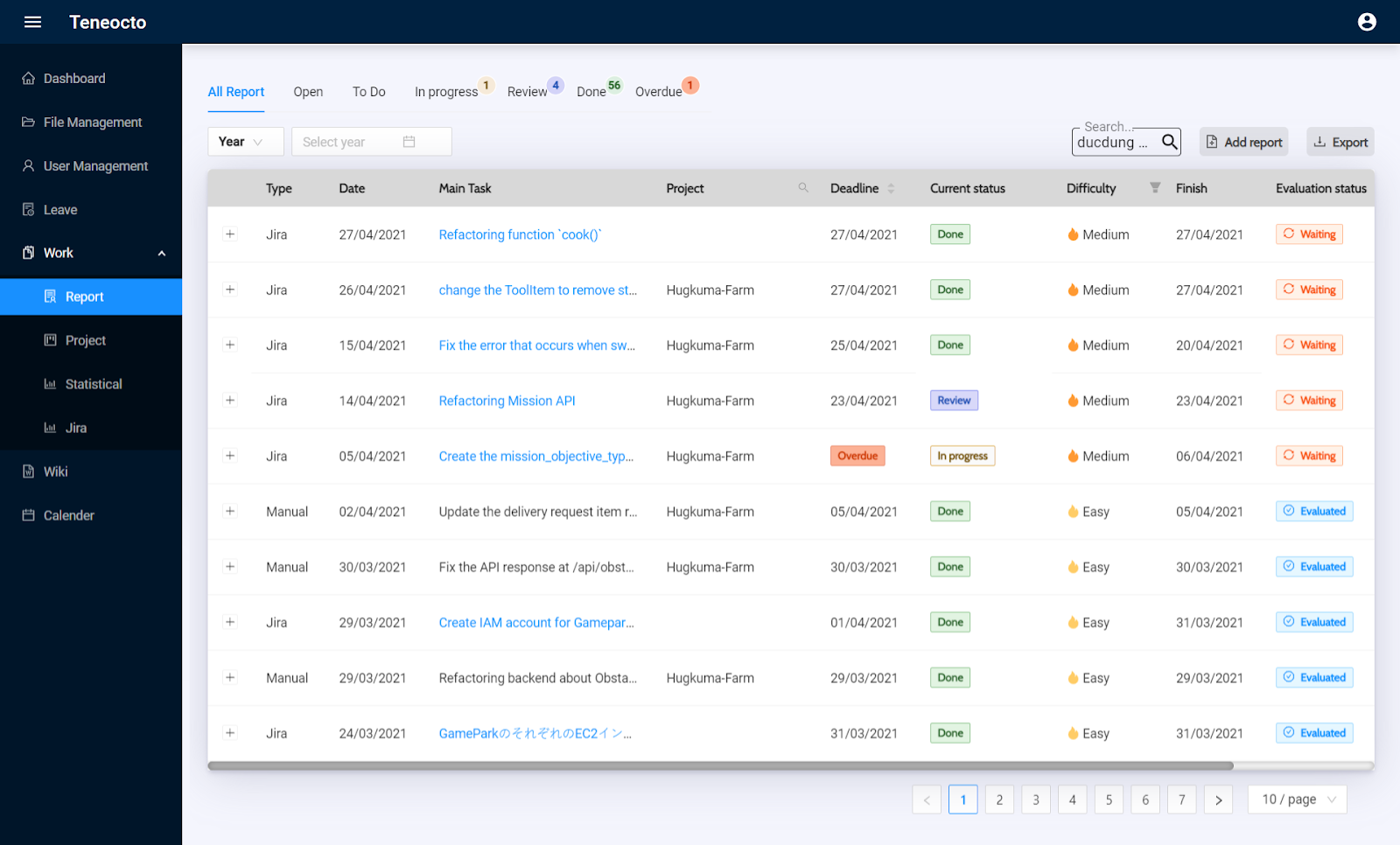
**Màn hình đăng nhập**



*Hình 5.1* Giao diện màn hình đăng nhập

Hình 5.1 mô tả màn hình chức năng đăng nhập người dùng. Màn hình này gồm form đăng nhập chưa email và password. Người dùng đăng nhập bằng sử dụng form này để đăng nhập với quyền “manager”. Ngoài ra người dùng với các quyền khác đăng nhập qua Google bằng cách ấn vào nút “Google”. Sau đó điền đầy đủ thông tin và chấp nhận các quyền để truy cập vào hệ thống

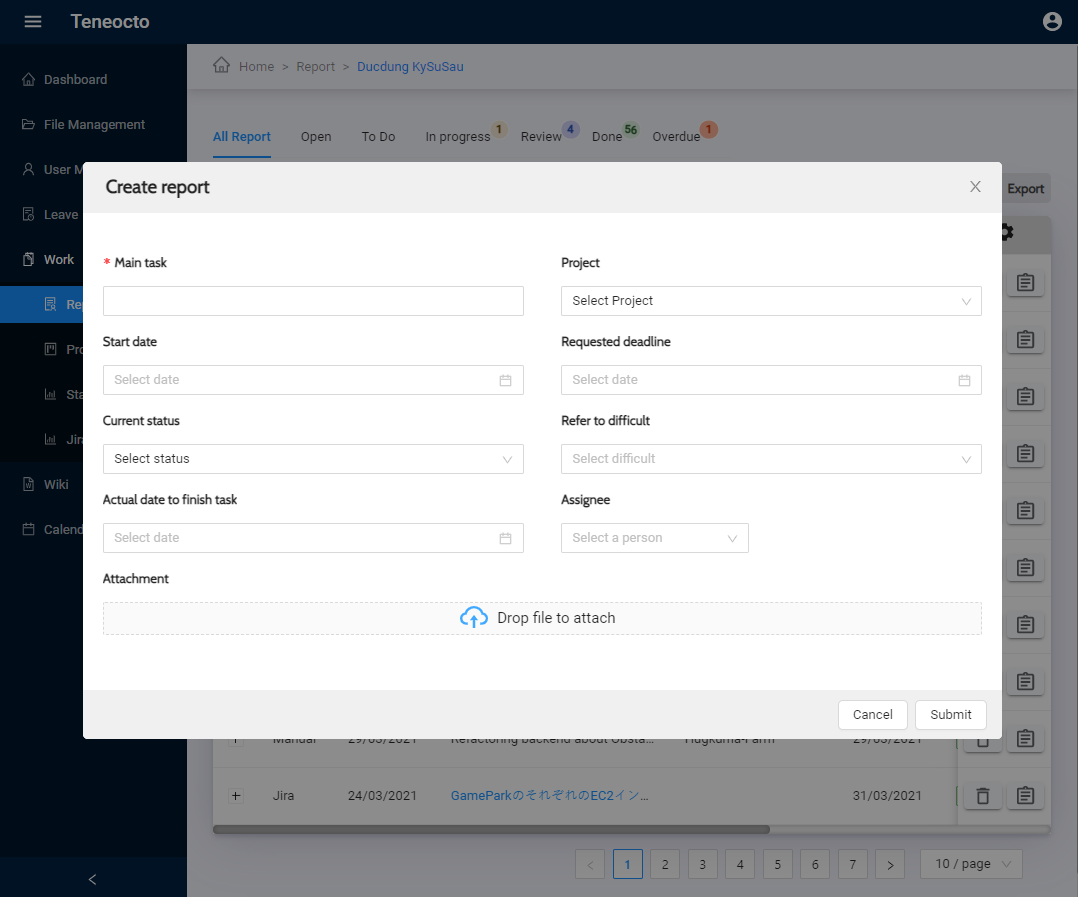
**Màn hình quản lý báo cáo công việc**



*Hình 5.2* Giao diện danh sách báo cáo công việc

Đây là màn hình danh sách công việc mà người dùng đã báo cáo lên hệ thống với quyền truy cập là Admin. Từ giao diện này người dùng có thể thực hiện với các thao tác như thêm báo cáo mới, xuất báo cáo ra file excel, sửa báo cáo, lọc danh sách báo cáo,..và với chức năng tìm kiếm, Admin có thể truy cập tới hệ thống báo cáo của các nhân viên khác trong hệ thống.

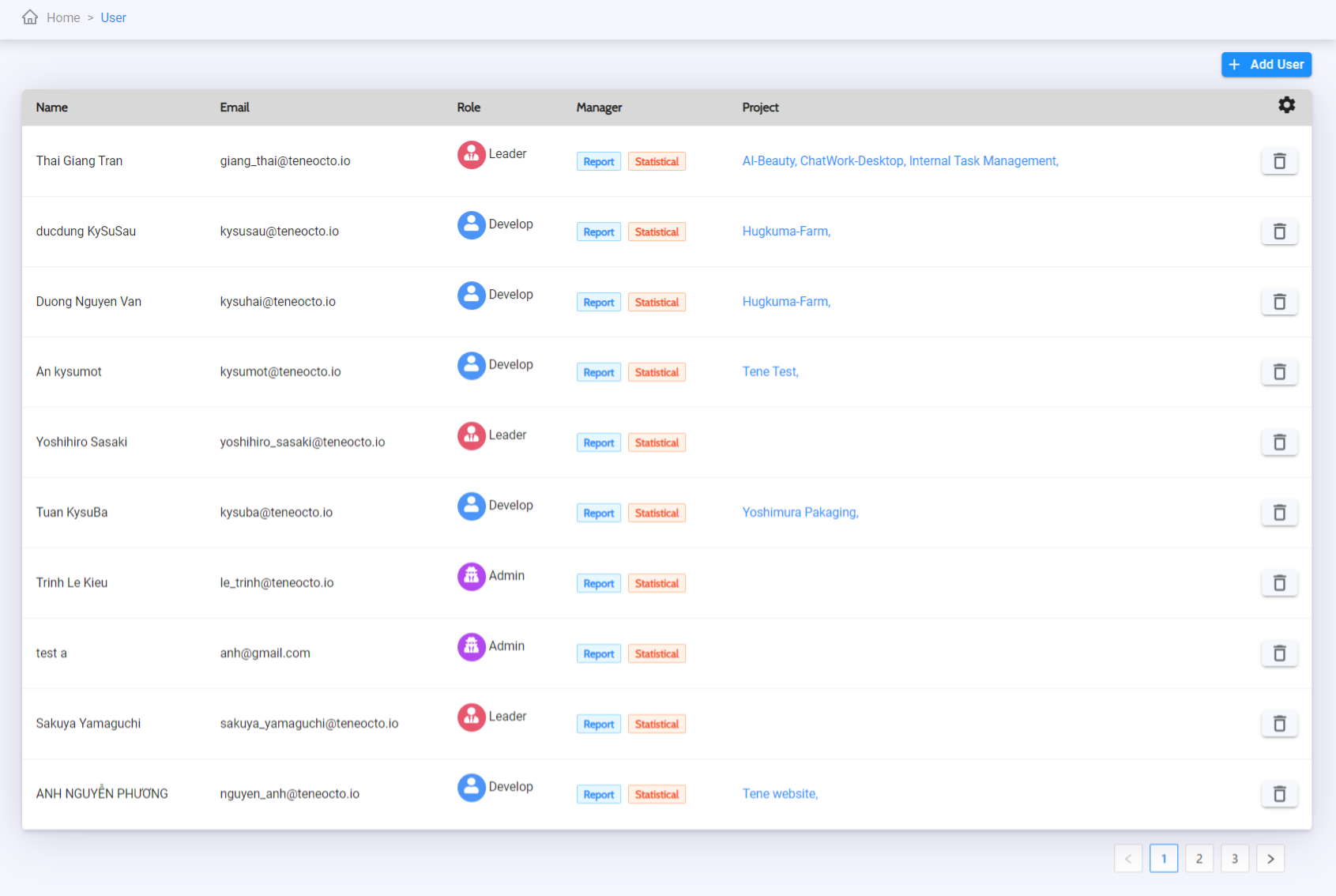
**Màn hình thêm mới báo cáo**



*Hình 5.3* Giao diện thêm mới báo cáo công việc

Hình 5.3 trên đây mô tả giao diện màn hình thêm mới một báo cáo vào hệ thống. Tại đây người dùng nhập các trường thông tin tương ứng với công việc mà mình báo cáo. Lưu ý những trường \* là những trường yêu cầu người dùng bắt buộc phải nhập. Sau khi hoàn thiện các trường thông tin người dùng ấn “Submit” để thêm mới một báo cáo vào hệ thống.

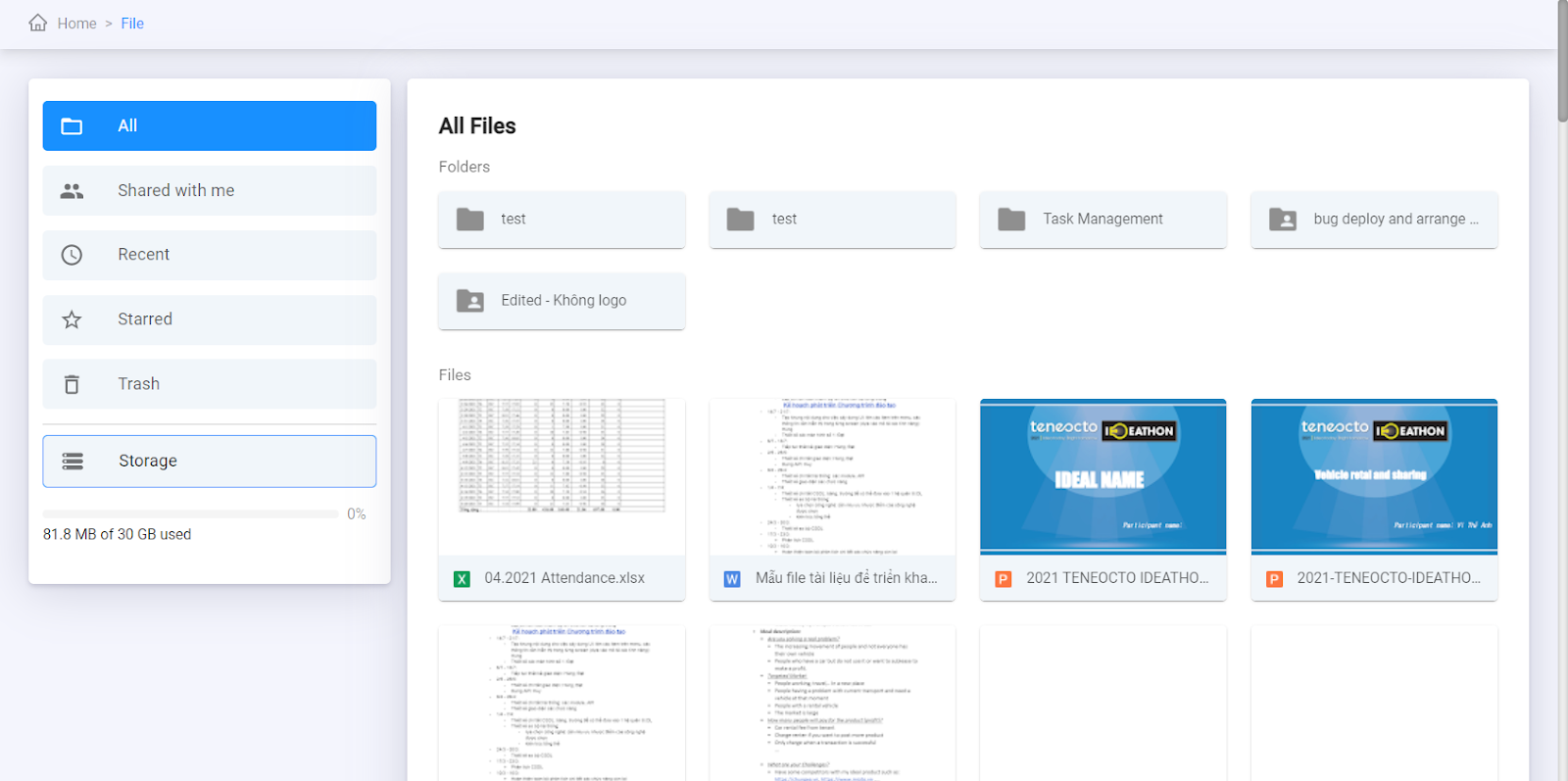
**Màn hình quản lý người dùng**



*Hình 5.4* Giao diện quản lý người dùng

Hình 5.4 hiển thị giao diện quản lý người dùng của admin. Tại giao diện này admin có thể thêm người dùng manager mới với username password bằng cách ấn vào button “Add user” hoặc thực hiện sửa các trường thông tin của user bằng cách ấn vào row chứa tên người dùng. Khi thêm người dùng vào một dự án thông tin data của người dùng đó cũng sẽ được thêm vào danh sách dự án tại mục quản lý dự án.

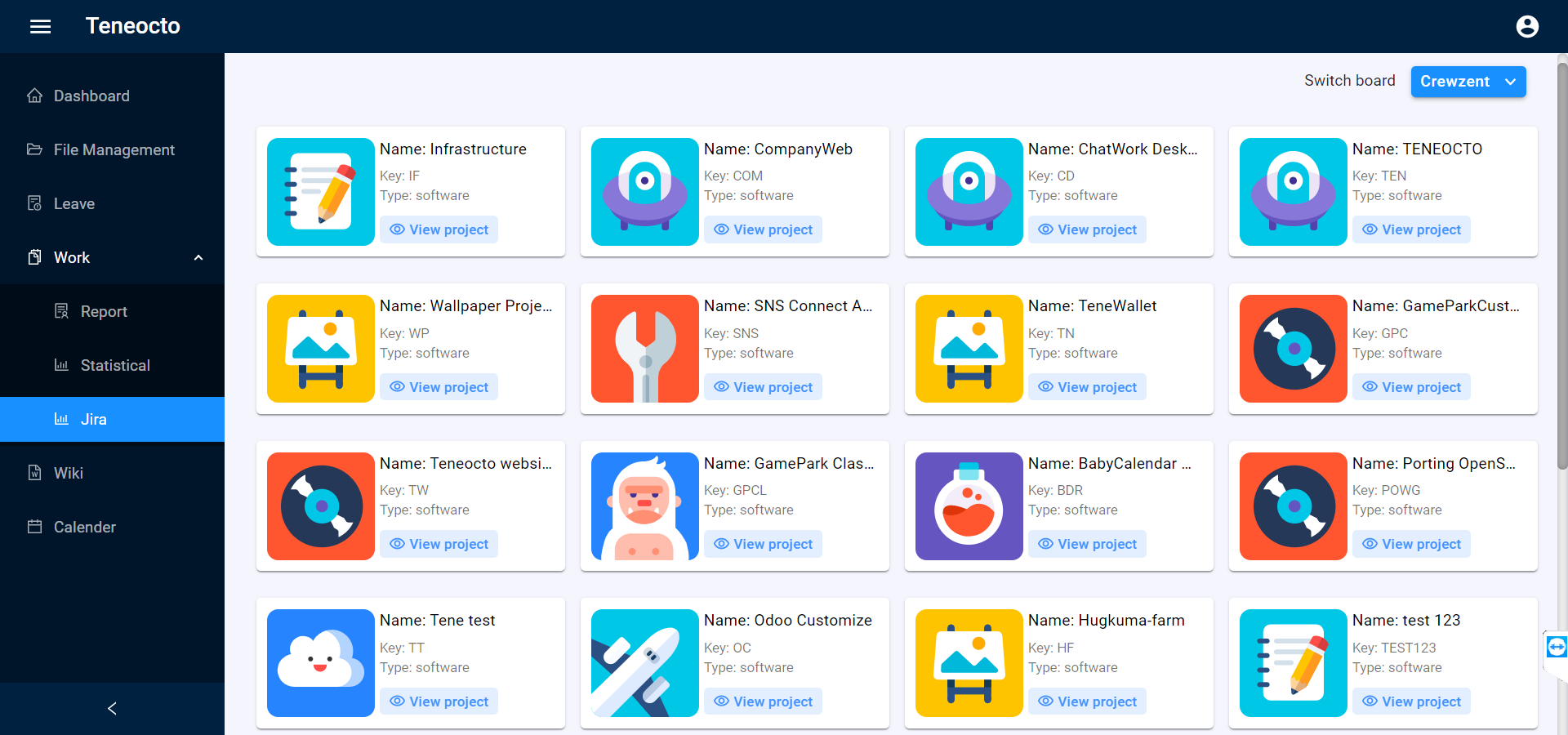
**Màn hình quản lý file Drive**

****

*Hình 5.5* Giao diện quản lý file Drive

Trên đây là giao diện quản lý file Drive của gmail đăng nhập cũng như quản lý file đính kèm theo báo cáo của người dùng. Trong giao diện này người dùng có thể thêm file mới, xóa file hoặc đi tới thư mục chi tiết của file trong Google Drive. Đồng thời có thể xem được dung lượng còn trống còn lại của Drive.

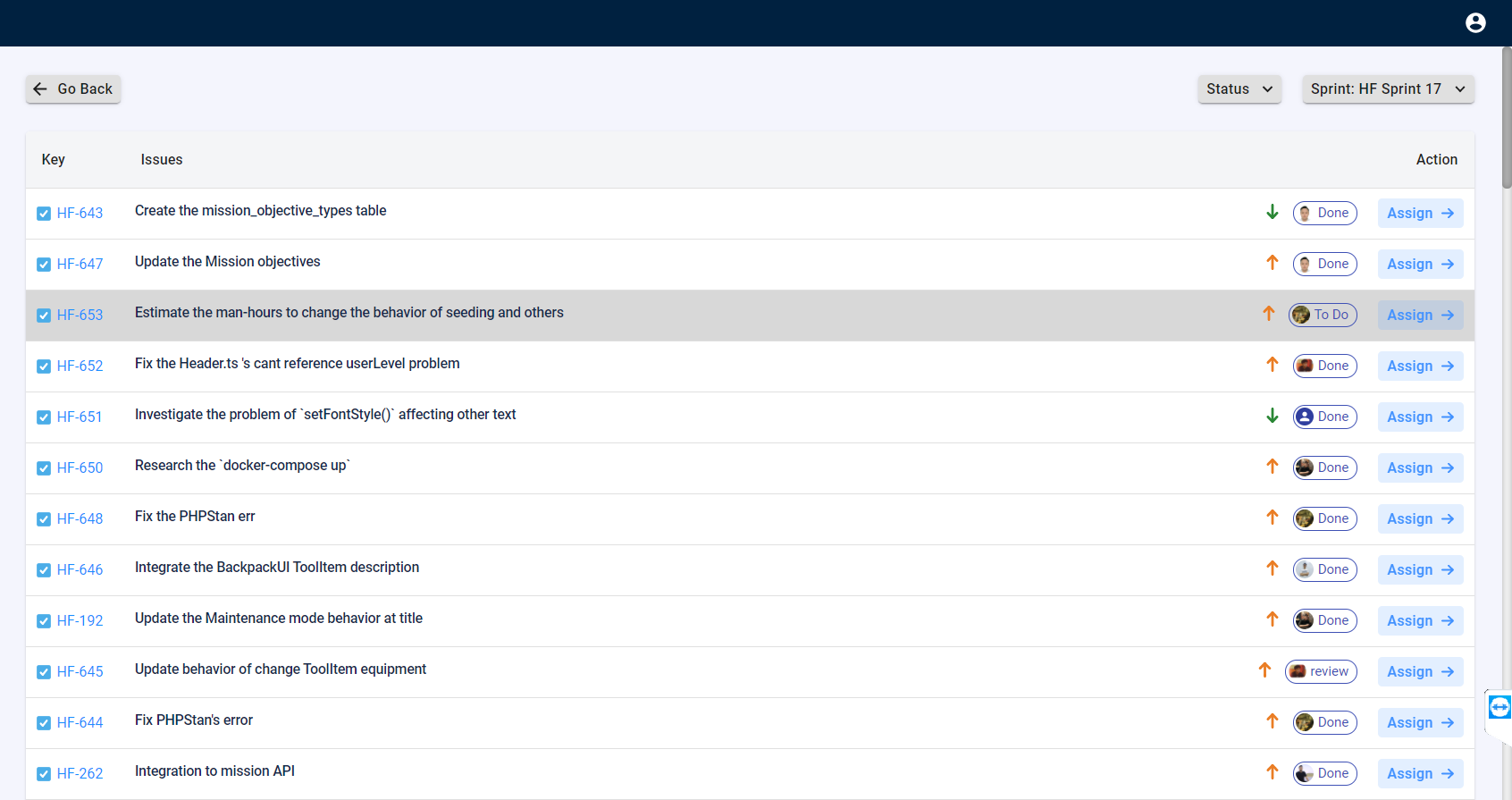
**Màn hình danh sách dự án Jira**



*Hình 5.6* Giao diện danh sách dự án từ Jira

Trên đây là giao diện màn hình hiển thị danh sách các dự án tham gia theo công ty mà gmail có quyền truy cập ở Jira software. Từ giao diện này người dùng có thể xem danh sách các dự án mình được giao công việc và ấn “View issue” để xem danh sách công việc mình được giao.

**Màn hình danh sách báo cáo được giao trên Jira**

****

*Hình 5.7* Giao diện danh sách báo cáo từ Jira

Hình 5.7 là giao diện hiển thị danh sách các báo cáo mà người dùng được giao trân trang Jira software. Tại đây người dùng có thể xem thông tin chi tiết bằng cách ấn vào báo cáo theo từng hàng, một cửa sổ nhỏ sẽ được hiện ra bên phải chứa tất cả thông tin về báo cáo công việc đó. Hơn nữa người dùng cũng cũng có thể chỉ định những báo cáo này về hệ thống báo cáo công việc cá nhân.

# CHƯƠNG 6. KẾT LUẬN

## 6.1. Kết quả đạt được

Hệ thống đã được triển khai thực tế tại địa chỉ <https://task.teneocto.io/> kể từ tháng 3/2021 và đáp ứng tất cả các yêu cầu chức năng bên phía công ty đề ra. Nhân viên và quản trị viên trong công ty có thể sử dụng hệ thống một cách dễ dàng, nhanh chóng đảm bảo các yêu cầu của bài toán được đặt ra ban đầu. Hệ thống sử dụng Google để đăng nhập và dùng các thông tin của Google gửi về để xác thực ủy quyền đảm bảo thông tin được sử dụng một cách thống nhất và toàn diện. Việc sử dụng các quy tắc bảo mật của Firebase hỗ trợ giúp giữ liệu của người dùng trong hệ thống được đảm bảo an toàn tránh khỏi các truy cập độc hại. Hơn nữa, các yêu cầu phi chức cũng được đáp ứng tốt, đảm bảo hệ thống hoạt động tốt và thuận lợi nhất cho người dùng trên nền tảng web. Bên cạnh đó hệ thống thường xuyên lắng nghe người dùng để cập nhật, nâng cấp các tính năng đáp ứng nhu cầu công việc ngày càng tăng cao.

Hệ thống có ưu điểm kết hợp nhiều công cụ, ở đó các đội ngũ công ty có thể lên kế hoạch cho dự án, phân công nhiệm vụ, theo dõi tiến trình công việc, quản lý thời gian, phân bổ tài nguyên, giao tiếp & hợp tác, và lưu trữ tài liệu, v..v. Từ đó công ty có thể theo dõi mọi khía cạnh của dự án và nhân viên trên một nền tảng duy nhất. Tuỳ nhiên không thể phủ nhận được hệ thống vẫn còn một số thiếu sót cần khắc phục như quản lý công việc trong dự án qua sprint (một vòng lặp ngắn hạn), chức năng kéo thả báo cáo, các thông báo đến email khi có thay đổi,..

Trong quá trình nghiên cứu, thực hiện triển khai khóa luận tôi đã học thêm được rất nhiều kiến bổ ích, những công nghệ mới phục vụ cho việc phát triển phần mềm như: các kiến thức về cơ sở dữ liệu, làm việc với các API của Google, thiết kế giao diện web,.. Hơn nữa trong quá trình phát triển hệ thống tôi đã có cơ hội làm việc trước tiếp với khách hàng sử dụng, đây là một trải nghiệm vô cùng quý báu đối với một sinh viên như tôi. Nó giúp tôi có trau dồi được các kỹ năng làm việc, xử lý các vấn đề trong môi trường doanh nghiệp

## 6.2. Định hướng phát triển

Trong tương lai ngoài việc khắc phục những thiếu sót còn tồn đọng, hệ thống sẽ hoàn thiện giao diện và phát triển thêm một số chức năng phục nhu cầu của người dùng như sau:

* **Giao việc trong dự án theo sprint:** một nhóm công việc trong ngắn hạn ( lý tưởng là 2-4 tuần) để đội phát triển thực hiện đầy đủ các công việc cần thiết như lập kế hoạch, phân tích yêu cầu, thiết kế, triển khai để cho ra các phần nhỏ của sản phẩm.
* **Thiết lập nhắc nhở:**  giúp bạn note lại những công việc và thời gian thực phải thực hiện và trong trường hợp bạn sợ quên đi hay bỏ lỡ. Ví dụ sau 30 phút nữa bạn có việc đi họp, ngay lập tức tính năng này sẽ hẹn thời 30 phút sau, hệ thống sẽ báo cho bạn khi đến hết thời gian.
* **Chức năng thông báo:** nhận được các thông báo hoặc chỉ tên bạn được nhắc đến đối với công việc được giao hoặc thay đổi liên quan đến người dùng đó. Điều này cực kỳ hữu ích khi bạn muốn lọc bớt thông báo và chú ý đến việc mình làm.

**Tài liệu tham khảo**

**Tiếng Anh**

[1] "React Documentation" April 2021. [Online]. Available:

<https://reactjs.org/docs/getting-started>

[2] "NodeJS Documentation" April 2021. [Online]. Available:

<https://nodejs.org/en/docs/>

[3] "Firebase Authentication" April 2021. [Online]. Available:

[https://firebase.google.com/docs/auth](https://firebase.google.com/docs/cloud-messaging)

[4] "Firebase Cloud FireStore" April 2021. [Online]. Available:

<https://firebase.google.com/docs/firestore>

[5] "Google Oauth 2.0" April 2021. [Online]. Available:

<https://developers.google.com/identity/protocols/oauth2>

[6] "Google Drive API" April 2021. [Online]. Available:

<https://developers.google.com/drive/api/v3/about-files>

[7] "Google Calendar API" April 2021. [Online]. Available:

<https://developers.google.com/calendar>

[7] "Jira software Restful API" April 2021. [Online]. Available:

<https://developer.atlassian.com/cloud/jira/software/rest/intro/>