**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



**TO CONG TUYEN**

**ỨNG DỤNG QUẢN LÝ THỜI GIAN VÀ CÔNG VIỆC TRÊN THIẾT BỊ DI ĐỘNG**

**Giảng viên hướng dẫn: ThS. Châu Thị Bảo Hà**

HO CHI MINH CITY, 2020

**INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY**

**FACULTY OF INFORMATION TECHNOLOGY**

**TO CONG TUYEN**

**TASK AND TIME MANAGEMENT APPLICATION ON MOBILE DEVICE**

Major: Software engineer

**Supervisor: Ms. Chau Thi Bao Ha**

HO CHI MINH CITY, 2020

**Title:** TASK AND TIME MANAGEMENT APPLICATION ON MOBILE DEVICE

**Abstract:**

* **Reason for writing:**

In the context of information technology is developing with the demand for using smart mobile devices is increasing rapidly. A problem poses the same workload but shortens the time. If not arranged work scientifically, the delay, delay is indispensable. So, to optimize work performance, manage time and complete schedules on time is something everyone wants to work, but not everyone knows how to organize and divide properly. The solution to managing your time and your work on paper may not be suitable anymore, due to the time-consuming paper preparation. Therefore, using the time and task management application will be more intuitive and quick.

* **Problem:**

+ Users can register an account, login.

+ Users can plan daily, weekly and monthly tasks.

+ Users can view tasks by day, by week.

+ Statistics of tasks done or not done weekly, monthly.

* **Methods:**

Using technology such as React Native, Firebase Service, Authentication, Cloud Firestore.

* **Results:**

+ Login, register an account.

+ View tasks by day.

+ View tasks by week.

+ View the remaining task today

+ View task statistics.

+ Manage (add, edit, delete) tasks, task labels.

* **Conclusion:**

+ The team understood how to make a real mobile application.

+ Implement the basic management functions required for an application.

# LỜI CẢM ƠN

Được sự tin tưởng và tạo điều kiện đăng ký đề tài của quý thầy cô khoa Công Nghệ Thông Tin, bộ môn Kỹ Thuật Phần Mềm trường Đại học Công Nghiệp TP.HCM, sau một học kỳ học hỏi và nghiên cứu, nhóm em đã hoàn thành Khoá luận tốt nghiệp đề tài “ỨNG DỤNG QUẢN LÝ THỜI GIAN VÀ CÔNG VIỆC TRÊN THIẾT BỊ DI ĐỘNG”.

Để hoàn thành đề tài tốt nghiệp này, lời đầu tiên nhóm em xin chân thành cảm ơn cô Châu Thị Bảo Hà – giảng viên bộ môn Kỹ Thuật Phần Mềm thuộc khoa Công Nghệ Thông Tin trường Đại học Công Nghiệp Thành phồ Hồ Chí Minh đã tận tình giúp đỡ và hướng dẫn nhóm em trong suốt quá trình nghiên cứu và thực hiện đề tài này.

Em xin chân thành cảm ơn các Thầy/Cô trong Khoa Công nghệ thông tin đã truyền đạt những kiến thức quý báu trong suốt bốn năm chúng em theo học tại trường qua đó làm tiền đề sự nghiệp tương lai của chúng em.

Dù đã cố gắng học tập nghiên cứu nhưng vì kiến thức và kinh nghiệm thực tế còn hạn chế nên nội dung của báo cáo khoá luận tốt nghiệp vẫn còn những thiếu xót và hạn chế.

Nhóm em xin gửi tới quý thầy cô lời cảm ơn chân thành nhất. Mong thầy cô luôn nhiều sức khoẻ.

**NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN**

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

*TP.HCM, ngày….tháng….năm 2020*

**Giảng viên hướng dẫn**

**NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN**

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

……………………………………………………………………………………..…..

*TP.HCM, ngày….tháng….năm 2020*

**Giảng viên phản biện 1 Giảng viên phản biện 2**

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc830968694)

[MỤC LỤC 6](#_Toc1004683617)

[DANH MỤC CÁC HÌNH VẼ 8](#_Toc53634558)

[DANH MỤC CÁC BẢNG BIỂU 12](#_Toc1640368213)

[DANH SÁCH CÁC TỪ VIẾT TẮT 14](#_Toc273495705)

[LỜI MỞ ĐẦU 15](#_Toc1027309355)

[CHƯƠNG 1 : GIỚI THIỆU 17](#_Toc219807605)

[1.1 Tổng quan 17](#_Toc634544395)

[1.2 Mục tiêu đề tài 19](#_Toc383855763)

[1.3 Phạm vi đề tài 19](#_Toc422933153)

[1.4 Mô tả yêu cầu chức năng 19](#_Toc66630901)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 21](#_Toc1026573020)

[2.1 React Native 21](#_Toc729127142)

*[2.1.1](#_Toc898185812)* [React Native là gì? 21](#_Toc898185812)

*[2.1.2](#_Toc1146387521)* [Cách hoạt động của React Native. 21](#_Toc1146387521)

*[2.1.3](#_Toc111784563)* [Hướng dẫn cài đặt môi trường React Native. 21](#_Toc111784563)

*[2.1.4](#_Toc1862442863)* [Hướng dẫn code React Native 24](#_Toc1862442863)

[2.2 Firebase 45](#_Toc355559769)

*[2.2.1](#_Toc1593531629)* [Firebase là gì? 45](#_Toc1593531629)

*[2.2.2](#_Toc1217526866)* [Cấu hình Firebase trong project app 45](#_Toc1217526866)

*[2.2.3](#_Toc1749848246)* [Cơ sở dữ liệu Cloud Firestore 49](#_Toc1749848246)

[CHƯƠNG 3 : PHÂN TÍCH 52](#_Toc2058408504)

[3.1 Phân tích yêu cầu bằng UML 52](#_Toc1857657205)

*[3.1.1](#_Toc1527384349)* [Use case tổng quát 52](#_Toc1527384349)

*[3.1.2](#_Toc1876721052)* [Danh sách tác nhân và mô tả 52](#_Toc1876721052)

*[3.1.3](#_Toc1958397475)* [Danh sách các tình huống hoạt động (use case) 53](#_Toc1958397475)

*[3.1.4](#_Toc304504756)* [Mô tả tình huống hoạt động 53](#_Toc304504756)

[CHƯƠNG 4 : THIẾT KẾ VÀ HIỆN THỰC 95](#_Toc357903291)

[4.1 Class diagram 95](#_Toc178916590)

[4.2 Deployment diagram 95](#_Toc574022330)

[4.3 Hiện thực 96](#_Toc1096757986)

*[4.3.1](#_Toc1359328501)* [Cấu hình phần cứng thiết bị và môi trường chạy 96](#_Toc1359328501)

*[4.3.2](#_Toc1303079521)* [Giao diện của ứng dụng 96](#_Toc1303079521)

*[4.3.3](#_Toc819277341)* [Test case 107](#_Toc819277341)

[CHƯƠNG 5 : KẾT LUẬN 116](#_Toc2076609270)

[5.1 Kết quả đạt được 116](#_Toc667769846)

[5.2 Hạn chế của đồ án 116](#_Toc458262500)

[5.3 Hướng phát triển 116](#_Toc1141479358)

[TÀI LIỆU THAM KHẢO 116](#_Toc1372151255)

[PHỤ LỤC 117](#_Toc2066741299)

# DANH MỤC CÁC HÌNH VẼ

[Hình 2-1 Cách hoạt động React Native 22](#_Toc514351900)

[Hình 2-2 Vòng đời của component 25](#_Toc543818083)

[Hình 2-3 Ví dụ constructor 26](#_Toc1696676998)

[Hình 2-4 Ví dụ getDerivedStateFromProps 27](#_Toc1095876569)

[Hình 2-5 Ví dụ componentDidUpdate 28](#_Toc2070564868)

[Hình 2-6 Mẫu cơ bản của một component 30](#_Toc2026509398)

[Hình 2-7 Ví dụ sử dụng props 31](#_Toc1855982841)

[Hình 2-8 Thiết kế style 33](#_Toc1582286990)

[Hình 2-9 Ví dụ Text 34](#_Toc1873267600)

[Hình 2-10 Ví dụ TextInput 34](#_Toc1512381428)

[Hình 2-11 Ví dụ Button 34](#_Toc1863850943)

[Hình 2-12 Ví dụ TouchableOpacity 35](#_Toc1010980797)

[Hình 2-13 Ví dụ Flatlist 35](#_Toc1916833934)

[Hình 2-14 Ví dụ Modal 36](#_Toc1967361101)

[Hình 2-15 Ví dụ ScrollView 37](#_Toc1512782580)

[Hình 2-16 Ví dụ Switch 37](#_Toc1105336542)

[Hình 2-17 Ví dụ ActivityIndicator 37](#_Toc1908030577)

[Hình 2-18 Ví dụ Alert 38](#_Toc1735401386)

[Hình 2-19 Ví vụ react-native-color-picker 38](#_Toc1836152341)

[Hình 2-20 Ví dụ react-native-gesture-handler 39](#_Toc1350846110)

[Hình 2-21 Ví dụ react-native-modal-datetime-picker 40](#_Toc925995467)

[Hình 2-22 Ví dụ react-native-picker-select 41](#_Toc1202114718)

[Hình 2-23 Ví dụ react-native-vector-icons 42](#_Toc214890165)

[Hình 2-24 Ví dụ @react-native-community/netinfo 42](#_Toc2142802213)

[Hình 2-25 Ví dụ react-native-charts-wrapper 43](#_Toc1130224476)

[Hình 2-26 Điều hướng đến một màn hình mới 44](#_Toc1456657655)

[Hình 2-27 Quay trở lại 44](#_Toc44047773)

[Hình 2-28 Tạo project trong Firebase 46](#_Toc1359469415)

[Hình 2-29 Đặt tên cho project 47](#_Toc1927117782)

[Hình 2-30 Chọn nên tảng 47](#_Toc1150287648)

[Hình 2-31 Đặt tên cho project 48](#_Toc2032217428)

[Hình 2-32 Cấu hình Firebase 48](#_Toc1084724477)

[Hình 2-33 Cấu hình đăng nhập 49](#_Toc964934847)

[Hình 2-34 Kích hoạt đăng nhập bằng email, password 49](#_Toc1023586338)

[Hình 2-35 Ví dụ insert data 50](#_Toc1877079708)

[Hình 2-36 Ví dụ update data 51](#_Toc2058879228)

[Hình 2-37 Ví dụ remote data 51](#_Toc1514039679)

[Hình 2-38 Ví dụ realtime changes 51](#_Toc1152445861)

[Hình 2-39 Ví dụ query data 52](#_Toc1096691576)

[Hình 3-1 Use case diagram 53](#_Toc391449976)

[Hình 3-2 Activity diagram đăng nhập 56](#_Toc25201816)

[Hình 3-3 Sequence diagram đăng nhập 57](#_Toc287400007)

[Hình 3-4 Đặc tả use case đăng ký tài khoản 58](#_Toc500600976)

[Hình 3-5 Activity diagram đăng ký tài khoản 59](#_Toc269187609)

[Hình 3-6 Sequence diagram đăng ký tài khoản 60](#_Toc1467606367)

[Hình 3-7 Activity diagram đăng xuất 62](#_Toc816060718)

[Hình 3-8 Sequence diagram đăng xuất 63](#_Toc1795820595)

[Hình 3-9 Activity diagram xem công việc theo ngày 64](#_Toc2079289759)

[Hình 3-10 Sequence diagram xem công việc theo ngày 65](#_Toc1734301766)

[Hình 3-11 Activity diagram xem chi tiết công việc 66](#_Toc585032688)

[Hình 3-12 Sequence diagram xem chi tiết công việc 67](#_Toc1655647777)

[Hình 3-13 Activity diagram xem công việc theo tuần 68](#_Toc1139734311)

[Hình 3-14 Sequence diagram xem công việc theo tuần 69](#_Toc729748329)

[Hình 3-15 Activity diagram xóa công việc 70](#_Toc210091714)

[Hình 3-16 Sequence diagram xóa công việc 71](#_Toc787625433)

[Hình 3-17 Activity diagram xem công việc tuần trước 72](#_Toc734704788)

[Hình 3-18 Sequence diagram xem công việc tuần trước 73](#_Toc1614293238)

[Hình 3-19 Activity diagram xem công việc tuần hiện tại 74](#_Toc2020760295)

[Hình 3-20 Sequence diagram xem công việc tuần hiện tại 74](#_Toc1228000122)

[Hình 3-21 Activity diagram xem công việc tuần sau 75](#_Toc1905716285)

[Hình 3-22 Sequence diagram xem công việc tuần sau 76](#_Toc2055811477)

[Hình 3-23 Activity diagram cập nhật trạng thái công việc 78](#_Toc1125100355)

[Hình 3-24 Sequence diagram cập nhật trạng thái công việc 79](#_Toc291609231)

[Hình 3-25 Activity diagram xem thống kê công việc 80](#_Toc532295968)

[Hình 3-26 Sequence diagram xem thống kê công việc 81](#_Toc576053034)

[Hình 3-27 Activity diagram thêm công việc 84](#_Toc2030489039)

[Hình 3-28 Sequence diagram thêm công việc 85](#_Toc83758772)

[Hình 3-29 Activity diagram thêm nhãn công việc 87](#_Toc1302379607)

[Hình 3-30 Sequence diagram thêm nhãn công việc 88](#_Toc1777141739)

[Hình 3-31 Activity diagram xem công việc hôm nay 89](#_Toc1213242440)

[Hình 3-32 Sequence diagram xem công việc hôm nay 90](#_Toc83790691)

[Hình 3-33 Activity diagram quản lý nhãn công việc 91](#_Toc346018685)

[Hình 3-34 Sequence diagram quản lý nhãn công việc 91](#_Toc1033169361)

[Hình 3-35 Activity diagram xóa nhãn công việc 93](#_Toc1123624525)

[Hình 3-36 Sequence diagram xóa nhãn công việc 94](#_Toc1550302377)

[Hình 3-37 Activity diagram xem chi tiết nhãn công việc 95](#_Toc1339327658)

[Hình 3-38 Sequence diagram xem chi tiết nhãn công việc 96](#_Toc1572574383)

[Hình 4-1 Sơ đồ class diagram 97](#_Toc65335835)

[Hình 4-2 Sơ đồ Deployment của ứng dụng 97](#_Toc477572158)

[Hình 4-3 Màn hình đăng nhập 99](#_Toc734608447)

[Hình 4-4 Màn hình đăng ký 100](#_Toc579104113)

[Hình 4-5 Màn hình thông báo công việc còn lại hôm nay 101](#_Toc276231259)

[Hình 4-6 Màn hình xem công việc theo ngày 102](#_Toc1771377835)

[Hình 4-7 Màn hình xem công việc theo tuần 103](#_Toc819759981)

[Hình 4-8 Màn hình thêm mới công việc 104](#_Toc1522617411)

[Hình 4-9 Màn hình chi tiết công việc 105](#_Toc926025617)

[Hình 4-10 Màn hình thêm màu nhãn 106](#_Toc910728066)

[Hình 4-11 Màn hình xem chi tiết màu nhãn 107](#_Toc862308149)

[Hình 4-12 Màn hình thống kê công việc 108](#_Toc1325615519)

# DANH MỤC CÁC BẢNG BIỂU

[Bảng 1 Danh sách các từ viết tắt 14](#_Toc1924159694)

[Bảng 3-1 Danh sách tác nhân và mô tả 53](#_Toc392971436)

[Bảng 3-2 Danh sách các tình huống hoạt động 53](#_Toc1062754893)

[Bảng 3-3 Đặt tả use case đăng nhập 54](#_Toc2069202088)

[Bảng 3-4 Đặc tả use case đăng xuất 60](#_Toc1882787457)

[Bảng 3-5 Đặc tả use case xem công việc theo ngày 63](#_Toc1368970834)

[Bảng 3-6 Đặc tả use case xem chi tiết công việc 65](#_Toc1152910101)

[Bảng 3-7 Đặc tả use case xem công việc theo tuần 67](#_Toc1317383281)

[Bảng 3-8 Đặc tả use case xóa công việc 69](#_Toc1974669715)

[Bảng 3-9 Đặc tả use case xem công việc tuần trước 71](#_Toc1340881193)

[Bảng 3-10 Đặc tả use case xem công việc tuần hiện tại 72](#_Toc622748795)

[Bảng 3-11 Đặc tả use case xem công việc tuần sau 74](#_Toc135553602)

[Bảng 3-12 Đặc tả use case cập nhật trạng thái công việc 76](#_Toc2108655158)

[Bảng 3-13 Đặc tả use case thống kê công việc 79](#_Toc944890097)

[Bảng 3-14 Đặc tả use case thêm công việc 82](#_Toc854186962)

[Bảng 3-15 Đặc tả use case thêm nhãn công việc 85](#_Toc1379137177)

[Bảng 3-16 Đặc tả use case xem công việc hôm nay 88](#_Toc1736434582)

[Bảng 3-17 Đặc tả use case quản lý nhãn công việc 89](#_Toc368472610)

[Bảng 3-18 Đặc tả use case xóa nhãn công việc 91](#_Toc237811306)

[Bảng 3-19 Đặc tả use case xem chi tiết nhãn công việc 94](#_Toc400887263)

[Bảng 4-1 Bảng test case 116](#_Toc1867855006)

[Bảng 22 . Môi trường kiểm thử 119](#_Toc131813642)

[Bảng 23 . Danh sách các chức năng kiểm thử 120](#_Toc1966612180)

[Bảng 24 . Kiểm thử chức năng 121](#_Toc628375868)

[Bảng 25 . Kiểm thử giao diện ứng dụng 122](#_Toc1716402495)

[Bảng 26 . Kế hoạch nhân sự 123](#_Toc1622879066)

# **DANH SÁCH CÁC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **Từ viết tắt/ký hiệu** | **Tên đầy đủ** |
| API | Application Programming Interface |
| CSDL | Cơ sở dữ liệu |
| DHKTPM12A | Đại học kỹ thuật phần mềm 12A |
| IDE | Integrated Development Environment |
| JS | JavaScript |
| SQL | Structure Query Language |
| UML | Unifiel Modeling Language |

Bảng 1 Danh sách các từ viết tắt

# LỜI MỞ ĐẦU

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

Trong xu thế mới của thế giới hiện nay, thời gian đang trở thành thứ tài nguyên vô giá. Tuy nhiên, trong thực tế không phải ai cũng coi trọng và sử dụng thời gian một cách khoa học. Vì vậy quản lý thời gian là một trong những kỹ năng quan trọng nhất mà con người cần rèn luyện càng sớm càng tốt. Quản lý thời gian là một chủ đề không mới nhưng lại có nhiều quan điểm khác nhau về lĩnh vực này qua từng năm tháng. Nếu ở thời đại cách mạng công nghiệp, quản lý thời gian đồng nghĩa với việc làm nhiều hơn và liên tục hơn trong một khoảng thời gian cụ thể hay đến thời đại Internet bắt đầu bùng nổ thì quản lý thời gian nghĩa là có một danh sách việc phải làm trong một ngày và đến bây giờ thì quản lý thời gian đồng nghĩa với việc ưu tiên cho những điều quan trọng và mối quan hệ quan trọng nhất của mình.

Để giải quyết những vấn đề trên hiện nay đã có những ứng dụng quản lý thời gian và công việc trên kho ứng dụng Google Play và App Store tiêu biểu là Google Calendar, Microsoft ToDo. Tuy nhiên hai ứng dụng trên còn một số hạn chế như ở ứng dụng Google Calendar không phân loại công việc theo màu sắc, giao diện xem công việc theo tuần không thấy rõ nội dung công việc, trạng thái công việc. Với ứng dụng Microsoft ToDo thì không có giao diện xem công việc theo tuần, cả hai ứng dụng đều không có thống kê công việc. Nhận thấy những hạn chế của hai ứng dụng trên, từ đó thu thập yêu cầu người dùng để viết lại ứng dụng trên. Như vậy với đề tài này sẽ khắc phục những hạn chế của hai ứng dụng trên để giúp người dùng lập kế hoạch theo dõi quản lý công việc một cách hiệu quả.

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

Ý nghĩa khoa học:

Qua đề tài tìm hiểu được các kỹ thuật, quy trình công nghệ để phục vụ cho việc phát triển ứng dụng trên thiết bị di động.

Ý nghĩa thực tiễn:

Ứng dụng này giúp người dùng giảm áp lực hơn trong công việc, giúp người dùng quản lý công việc một cách hiệu quả, tiết kiệm thời gian, giúp đơn giản hoá công việc, tạo môi trường làm việc chuyên nghiệp.

# : GIỚI THIỆU

## Tổng quan

Trong bối cảnh công nghệ thông tin ngày càng phát triển cùng với nhu cầu sử dụng thiết bị di động thông minh gia tăng nhanh chóng. Một vấn đề đặt ra cùng một khối lượng công việc nhưng thời gian rút ngắn lại. Nếu không sắp xếp công việc một cách khoa học thì công việc chậm trễ, trì hoãn là điều tất yếu. Vậy để tối ưu hiệu suất công việc, quản lý thời gian và hoàn thành kế hoạch đúng hạn là điều tất cả mọi người đều muốn hướng tới, tuy nhiên không phải ai cũng biết cách sắp xếp và phân chia sao cho hợp lý. Nếu chúng ta đang sống trong áp lực vì khối lượng công việc nhiều không đếm xuể và thường xuyên bỏ quên công việc quan trọng hoặc tập trung quá nhiều vào một công việc nào đó thì lúc này sử dụng phần mềm quản lý công việc sẽ chính là giải pháp hiệu quả. Để quản lý công việc cần có sự can thiệp của công nghệ để nâng cao hiệu quả và tiết kiệm thời gian. Do đó việc phát triển một ứng dụng quản lý thời gian và công việc là nhu cầu cần thiết, với các tính năng lập kế hoạch, nhắc việc, theo dõi tiến độ công việc...

Hiện nay có hai nền tảng trên thiết bị di động được người dùng sử dụng nhiều nhất là Android và iOS. Để phát triển ứng dụng cho Android thì cần ngôn ngữ Java/Kottlin và công cụ Android Studio, phát triển ứng dụng cho iOS thì cần ngôn ngữ Objective-C/Swift và công cụ Xcode. Như vậy để phát triển ứng dụng cho Android và iOS thì phải biết tối thiểu hai ngôn ngữ và phải viết riêng cho từng nền tảng, như thế sẽ rất tốn thời gian và chi phí cho người phát triển. Vì thế lập trình đa nền tảng ra đời để giải quyết vấn đề trên, chỉ cần viết một lần chạy được cho cả hai nền tảng, đổi lại về mặt hiệu năng của ứng dụng sẽ thấp so với phát triển đơn nền tảng. Với đề tài ứng dụng quản lý thời gian và công việc thì nhận thấy rằng yêu cầu hiệu năng phần cứng của ứng dụng không quá cao, vì vậy lựa chọn phát triển đa nền tảng là một giải pháp hiệu quả. Hiện nay có ba Framework phổ biến nhất để phát tiển ứng dụng đa nền tảng là Flutter, Xamarin, React Native.

**Flutter** là một khung phát triển ứng dụng di động đa nền tảng nguồn mở do Google tạo ra vào năm 2017.

Ưu điểm Flutter: mạnh về animation, performance app rất cao

Nhược điểm Flutter: còn khá mới, chưa hỗ trợ nhiều thư viện, cộng đồng phát triển còn nhỏ, chưa ổn định, học thêm ngôn ngữ DART.

**Xamarin** được xây dựng vào tháng 5 năm 2011 bởi các kỹ sư đã tạo ra Mono, Mono cho Android và MonoTouch, mục đích là triển khai chạy ứng dụng trên nhiều nền tảng của Common Language Infrastructure.

Ưu điểm Xamarin: performance cao, nhiều thư viện hỗ trợ làm ứng dụng cực nhanh, ngôn ngữ C# thông dụng.

Khuyết điểm Xamarin: ngốn nhiều dung lượng trên thiết bị, cộng đồng Xaramin còn hạn chế, bị giới hạn truy cập vào các thư viện mã nguồn mở.

**React Native** là một Framework do Facebook phát triển hướng đến tối ưu hóa hiệu năng Hybrid và tối giản số lượng ngôn ngữ Native di động.

Ưu điểm React Native: hiện tại đã rất nhiều thư viện, gần như đã rất đầy đủ cho các nhu cầu app thông dụng, ổn định, ngôn ngữ JavaScript thông dụng.

Nhược điểm React Native: khó bảo trì nâng cấp, hiệu năng animation là điểm yếu, không thích hợp cho các app cần năng lực tính toán cao.

Chọn React Native để phát triển: vì hiện là Framework phổ biến nhất, nhiều tài liệu, nhiều code mẫu, nhiều thư viện, cộng đồng đông, ngôn ngữ javascript là một lợi thế.

Giải pháp cho cơ sở dữ liệu:

* Database cục bộ: sqlite, realm,...

Ưu điểm: đơn giản, phát triển nhanh, ứng dụng không cần internet.

Khuyết điểm mất dữ liệu khi gỡ ứng dụng, không đồng bộ dữ liệu.

* Máy chủ web

Ưu điểm: gỡ ứng dụng không mất dữ liệu, đồng bộ dữ liệu.

Khuyết điểm: lưu trữ dữ liệu phía server, khó làm, tốn kém chi phí, ứng dụng cần kết nối internet.

* Firebase

Ưu điểm: đơn giản, nhanh, đồng bộ dữ liệu, gỡ ứng dụng không mất dữ liệu.

Khuyết điểm: ứng dụng cần kết nối internet.

Chọn Firebase vì đơn giản, nhanh, bảo mật, ổn định, đồng bộ dữ liệu, gỡ ứng dụng không mất dữ liệu.

## Mục tiêu đề tài

Tìm hiểu cách thức lập trình đa nền tảng trên thiết bị di động.

Xây dựng ứng dụng quản lý thời gian và công việc trên thiết bị di động.

## Phạm vi đề tài

Framework React Native, ngôn ngữ javascript, công cụ phát triển visual code, môi trường nodejs 14, dịch vụ Firebase, cơ sở dữ liệu Cloud Firestore.

Ứng dụng dùng cho hệ điều hành iOS phiên bản 9.0 trở lên và Android 4.4 trở lên.

## Mô tả yêu cầu chức năng

Ứng dụng quản lý thời gian và công việc cung cấp các tính năng cơ bản cho phép người dùng dễ dàng quản lý thời gian biểu. Để có thể sử dụng các tính năng của hệ thống người phải đăng nhập vào hệ thống. Sau khi đăng nhập thành công người dùng có thể sử dụng các chức năng như:

* Đăng ký tài khoản, xem thông tin tài khoản và đăng xuất, thông tin tài khoản gồm email, password. Khi đăng ký tài khoản, kiểm tra email trùng.
* Quản lý công việc gồm thêm mới công việc, cập nhật công việc, xoá công việc, xem chi tiết công việc. Thông tin công việc gồm: tên công việc, thời gian bắt đầu (không được trước thời gian hiện tại), thời gian nhắc nhở (tuỳ chọn nhắn nhở mặc định không nhắn nhở) thời gian nhắn nhở không được sau thời gian bắt đầu làm, ghi chú, màu sắc, trạng thái công việc (hoàn thành hay không hoàn thành), công việc lặp lại (tuỳ chọn mặc định không lặp, hằng ngày, cách ngày, hằng tuần, cách tuần...), ngày kết thúc công việc lặp (không được trước ngày bắt đầu). Không được cập nhật trạng thái công việc khi chưa đến thời gian bắt đầu.
* Quản lý màu nhãn công việc gồm thêm, xoá, sửa màu. Thông tin màu nhãn gồm tên nhãn, màu sắc, cho phép chọn màu tuỳ ý. Xoá màu có ràng buộc dữ liệu, các công việc đang có màu đã chọn thì không thể xoá
* Xem công việc theo ngày: chức năng chọn ngày để xem các công việc theo ngày, mặc định là ngày hiện tại, thông tin hiển thị của mỗi công việc gồm thời gian bắt đầu, tên công việc, ghi chú, trạng thái công việc, màu sắc (nhẵn). Cho phép chuyển sang các tuần trước, tuần sau. Chọn vào từng công việc để xem chi tiết của công việc.
* Xem công việc theo tuần: hiển thị bảy ngày trong tuần cho phép chuyển giữa các tuần trước, tuần hiện tại, tuần sau, thông tin hiển thị công việc gồm tên công việc, thời gian bắt đầu, trạng thái, màu sắc. Nhấp vào từng công việc để xem chi tiết của công việc. Cho phép cập nhật nhanh trang thái công việc khi vuốt sang phải, vuốt sang trái để xoá.
* Thống kê công việc: thống kê công việc đã hoàn thành, chưa hoàn thành theo tuần, tháng.

# : CƠ SỞ LÝ THUYẾT

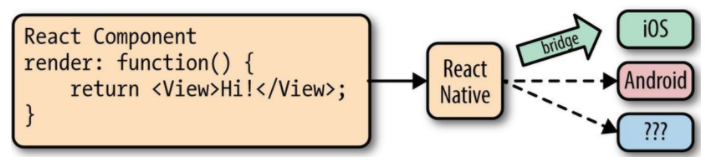
## React Native

### React Native là gì?

React Native là một Framework do Facebook phát triển hướng đến tối ưu hóa hiệu năng Hybrid và tối giản số lượng ngôn ngữ Native di động.

### Cách hoạt động của React Native.

Bằng cách tích hợp hai Thread là Main Thread và JS Thread cho ứng dụng mobile. Với Main Thread sẽ đảm nhận vai trò cập nhật giao diện người dùng(UI). Sau đó sẽ xử lý tương tác người dùng. Trong khi đó, JS Thread sẽ thực thi và xử lý code JavaScript. Hai luồng này hoạt động độc lập với nhau.



Hình 2-1 Cách hoạt động React Native

Để tương tác được với nhau hai Thread sẽ sử dụng một Bridge (cầu nối). Cho phép chúng giao tiếp mà không phụ thuộc lẫn nhau, chuyển đổi dữ liệu từ Thread này sang Thread khác. Dữ liệu từ hai Thread được vận hành khi tiếp nối dữ liệu cho nhau.

### Hướng dẫn cài đặt môi trường React Native.

#### Hướng dẫn cài đặt môi trường React Native trên hệ điều hành Windows.

Bước 1: Cài đặt Chocolatey từ <https://chocolatey.org>

Bước 2: Cài đặt Nodejs Java và Python 2 thông qua Chocolatey sử dụng dòng lệnh sau (dùng cmd để chạy lệnh này):

choco install -y nodejs.install python2 jdk8

Bước 3: Cài đặt Android Studio

Bước 4: Cài đặt SDK: Nên cài SDK Platform 23 vì mặc định React Native hiện tại sử dụng Android SDK Platform 23 để build ứng dụng.

#### Hướng dẫn cài đặt môi trường React Native trên hệ điều hành macOS.

Bước 1: Cài đặt Brew

{{/usr/bin/ruby -e "$(curl –fsSL https://raw.githubusercontent.com/ Homebrew/install/master/install)"}}

Bước 2: Cài đặt Nodejs: brew install node

Bước 3: Cài đặt Watchman: brew install watchman

Bước 4: Cài đặt react-native: npm install -g react-native-cli

Bước 5: Cài đặt Xcode: truy cập App Store trên macOS để cài đặt Xcode.

#### Các IDE thường dùng.

Code: sử dụng Visual Studio Code, Sublime Text, Atom,...

Build ứng dụng:

iOS: sử dụng Xcode

Android: sử dụng Android Studio

#### Khởi tạo dự án đầu tiên.

**Bước 1:** Khởi tạo dự án: mở Terminal (cd vào thư mục muốn tạo dự án trước)

react-native init ProjectName

**Bước 2:** Truy cập vào dự án vừa tạo.

cd ProjectName

**Bước 3:** Chạy ứng dụng trên hệ điều hành:

iOS: react-native run-ios

Android: react-native run-android

Hiển thị menu điều khiển:

Command + D (hoặc lắc điện thoại iOS) để hiển thị menu điều khiển khi run debug ứng dụng trên macOS.

ctrl + D hoặc phím menu để hiển thị menu điều khiển khi run debug ứng dụng trên Windown.

Command + R để reload lại source code máy ảo iOS.

ctrl + R để reload lại source code máy ảo Android.

#### Các thành phần cơ bản của dự án.

Thư mục Android: chứa toàn bộ source build ứng dụng Android. Chúng ta có thể mở thư mục Android bằng Android Studio và chạy ứng dụng thay vì sử dụng dòng lệnh react-native run-android nhưng có thể ứng dụng sẽ không build mã JavaScript được và sẽ xuất hiện màn hình trắng trên điện thoại Android.

Thư mục iOS: chứa toàn bộ source build ứng dụng iOS. Chúng ta có thể mở file ProjectName.xcodeproj bằng Xcode để run ứng dụng iOS thay vì sử dụng dòng lệnh react-native run-ios . Lần đầu có thể chạy hơi lâu nhưng những lần tiếp theo sẽ nhanh hơn việc build bằng dòng lệnh.

Thư mục node modules: chứa toàn bộ các package (thư viện) cần để chạy một ứng dụng react-native.

File package.js: file quản lý các package nodejs đi kèm với dự án. Nếu tải các dự án demo về cần dử dụng dòng lệnh npm install để tải toàn bộ thư viện yêu cầu của dự án về.

File package-lock.js file được general sau khi chạy cài đặt npm install

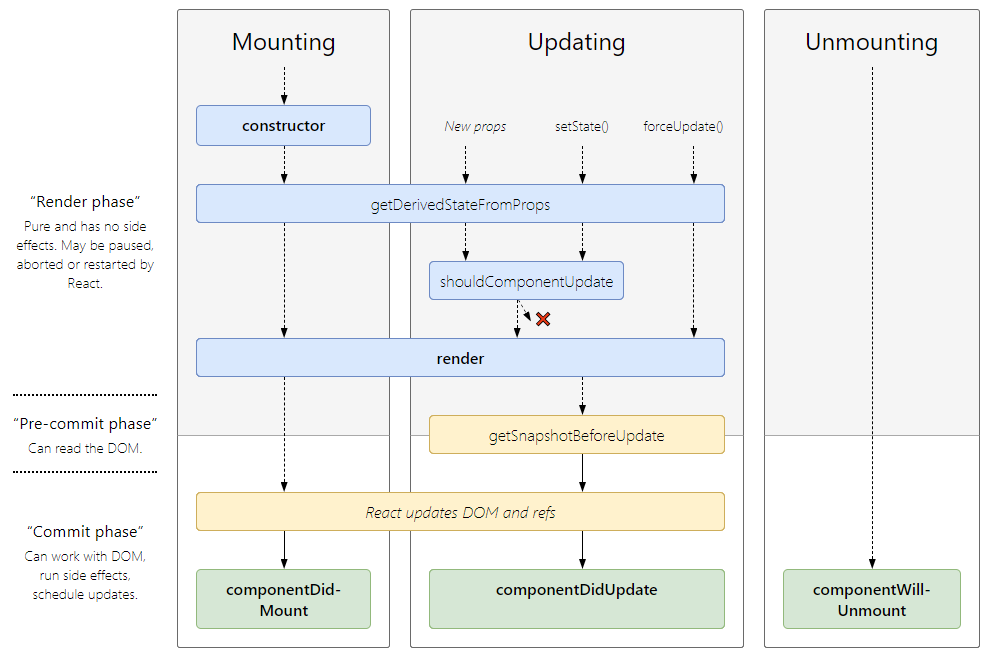
File index.js: file đầu tiên được binding khi chạy ứng dụng. File này sẽ đăng ký một component, component này sẽ được load lên đầu tiên khi chạy, mặc định ứng dụng sẽ đăng ký component trong App.js

File app.json: file config tên ứng dụng và tên hiển thị.

File App.js là một component mặc định có sử dụng một số Component khác như Text, View...

### Hướng dẫn code React Native

#### Vòng đời của component



Hình 2-2 Vòng đời của component

Mỗi thành phần React đi kèm với một số phương thức cho phép các nhà phát triển cập nhật trạng thái ứng dụng và phản ánh sự thay đổi thành UI. Có ba giai đoạn chính của một thành phần bao gồm Mount, Update và Unmount.

**Mounting** là các phương thức sẽ được gọi khi một instance của thành phần React được tạo và gắn vào DOM.

**constructor()**

Phương thức này được gọi trước khi một thành phần React được mount. Để gọi được super(props) trước bất kỳ statement trong các constructor đều bắt buộc phải có phương thức này. Vì nó sẽ cho phép chúng ta gọi hàm constructor của lớp cha và tự khởi tạo trong trường hợp lớp của chúng ta mở rộng bất kỳ lớp nào khác có chính nó. Nếu không, this.props sẽ không xác định được thành phần có props hay không.

Constructor dùng để khởi tạo trạng thái hoặc liên kết các trình xử lý sự kiện với thể hiện của lớp rất tốt.

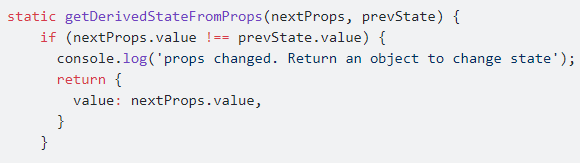


Hình 2-3 Ví dụ constructor

**getDerivedStateFromProps(nextProps, prevState)**

Chức năng này được gọi khi thành phần được mount cũng như nhận props mới cho dù chúng có bị thay đổi hay không. Nó cũng được gọi nếu thành phần cha được render lại, vì vậy nếu bạn chỉ muốn cập nhật thay đổi giá trị, phải có sự so sánh giá trị mới và trước đó.

Hàm này sẽ trả về một đối tượng để cập nhật trạng thái hoặc null vì hàm static sẽ không có bất kỳ xác nhận nào tới this. Nếu state của component được khởi tạo từ các props nhận được từ component cha. Chức năng này là nơi thích hợp để đồng bộ hóa props và state. Để làm như vậy, khi bạn cần cập nhật state, hãy nhớ trả về đối tượng chứa thuộc tính cần cập nhật.



Hình 2-4 Ví dụ getDerivedStateFromProps

**render()**

Đây là hàm cần thiết duy nhất để render ra view của một component.

Hàm này sẽ không được gọi lại nếu hàm shouldComponentUpdate() return false.

Không được setState ở hàm này vì sẽ gặp ra vòng lặp vô hạn.

**componentDidMount()**

Hàm này gọi ngay sau khi component được hiển thị.

Hàm này thường được dùng để đăng kí (Tuy nhiên, nếu đăng kí thì trong hàm componentWillUnmount thì phải huỷ đăng kí) , fetch data...

Ngoài ra, ta có thể setState, tuy nhiên, một số state khởi tạo thì hãy viết ở constructor,ngoài một số trường hợp ngoại lệ để tránh ảnh hưởng về performace.

Hàm này sẽ không được gọi nếu hàm shouldComponentUpdate() return false.

**Updating** là quá trình cập nhật lại component khi state hoặc prop thay đổi

**shouldComponentUpdate (nextProps, nextState)**

Phương pháp này sẽ được gọi trước khi render khi nhận được props hoặc state mới. Theo mặc định, khi có thay đổi state hoặc props, các thành phần sẽ được render lại. Khi bạn không muốn các thành phần có nhiều render vô dụng, thì phương pháp này là nơi tuyệt vời để xác minh tất cả các thay đổi bạn cần trước khi render lại. Nếu bạn không muốn render lại các thành phần, hãy trả lại false , nếu không nó sẽ quay trở lại true.

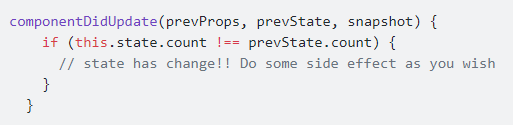
Khi chức năng này trở lại false, render và componentDidUpdate sẽ không được gọi.

**getSnapshotBeforeUpdate()**

Phương thức này được gọi ngay trước khi virtual DOM sắp thực hiện thay đổi đối với DOM, cho phép các thành phần của chúng ta nắm bắt các giá trị hiện tại. Đây là một chu kỳ mới đã được thêm vào kể từ React 16.3 và trả về một giá trị. Mọi giá trị được trả về sẽ được chuyển qua như một tham số thứ ba componentDidUpdate.

**componentDidUpdate(prevProps, prevState, snapshot)**

Phương pháp này sẽ được gọi sau mỗi lần render. Vì phương pháp này chỉ được gọi một lần sau khi cập nhật, đây là nơi thích hợp để thực hiện bất kỳ hoạt động hiệu ứng phụ nào. Tuy nhiên, đừng quên thực hiện so sánh giữa các props trước đây và hiện tại.



Hình 2-5 Ví dụ componentDidUpdate

**Unmounting** làquá trình component bị remove khỏi DOM, hay nói một cách khác là hàm componentWillUnmount sẽ được gọi khi render ra không có component nào hoặc người dùng chuyển hướng

**componentWillUnmount()**

Khi một thành phần không được unmounted hoặc destroyed, phương thức này sẽ được gọi. Đây là một nơi để clean lại:

Bộ hẹn giờ vô hiệu hóa

Hủy bỏ bất kỳ yêu cầu mạng

Xóa trình xử lý sự kiện

Xóa mọi đăng ký

Gọi setState ở đây là vô ích vì sẽ không có render lại trên component.

#### Các thành phần cơ bản của component

Sau đây là chương trình mẫu cơ bản của một Component



Hình 2-6 Mẫu cơ bản của một component

**State** - là biến điều khiển trạng thái nội bộ của một component. State có thể thay đổi bằng cách gọi hàm this.setState({...}).

Mỗi lần thay đổi state hàm render sẽ được gọi lại ngay sau đó (hàm render chỉ thay đổi những thành phần có liên quan đến những giá trị trong state bị thay đổi).

Chúng ta nên bỏ các biến có liên quan đến UI vào trong state này, để khi state thay đổi, UI màn hình sẽ được vẽ lại và thay đổi theo.

**Lưu ý:** Không được thay đổi state trực tiếp bằng cách gọi this.state = {...} nếu sử dụng thay đổi state trực tiếp toàn bộ component này sẽ không còn hoạt động đúng như mong muốn nữa.

**Props** - là các thuộc tính được sử dụng truyền thằng vào. Đây là các thông số được truyền vào để tùy chỉnh theo ý muốn của người xây dựng Component. Khác với state chúng ta không được thay đổi props ở trong chính bản thân của nó. Chúng ta chỉ nên đọc các thuộc tính được truyền vào để sử dụng mà thôi. Ví dụ sử dụng props



Hình 2-7 Ví dụ sử dụng props

#### Một số hàm đặc biệt

**Hàm this.setState()** - Hàm dùng để thay đổi state của component. Đây là phương thức chính để cập nhật giao diện người dùng. Khi hàm này thực thi xong thì hàm render() sẽ được tự động gọi lại. Những giá trị nào của state thay đổi thì chỉ những thành phần có sử dụng biến state tương ứng đó được gọi để vẽ lại UI.

Lưu ý: hàm này chạy bất đồng bộ nên chúng ta không nên đọc giá trị sau khi gọi hàm này.

**Hàm forceUpdate()** - Mặc định hàm render() sẽ được gọi khi props hoặc state thay đổi. Nhưng nếu một vài thành phần UI sử dụng một số dữ liệu khác state hoặc prop muốn thay đổi, thì chúng ta cần thông báo cho React biết để vẽ lại toàn bộ bằng cách gọi hàm forceUpdate().

#### Cấu trúc chương trình

Toàn bộ source code của chương trình sẽ được đặt trong thư mục app:

**assets** là thư mục chứa resource của mình bao gồm các resource như custom font (fonts), hình ảnh (images), ngôn ngữ (languages)

**configs** là thư mục chứa các cấu hình của ứng dụng: bao gồm các cấu hình server, link, màu sắc cơ bản.

**libs** là thư mục chứa các thư viện cơ bản của mình để xử lý một số vấn đề nội bộ như:

Database (xử lý lưu trữ dữ liệu bằng database)

Storage (xử lý lưu trữ dữ liệu bằng storage)

Language (Cấu hình xử lý đa ngôn ngữ trong ứng dụng)

RESTClient (Cấu hình, danh sách các api truy cập hệ thống server)

SoundPlayer (Điều khiển âm thanh)

Inapp (Một vài cấu hình, xử lý thanh toán mua bán với store)

Ads (Cấu hình hiển thị quảng cáo từ bên thứ ba)

**models** là thư mục chứa các model do mình định nghĩa, có thể là định nghĩa các đối tượng hoặc các loại của đối tượng

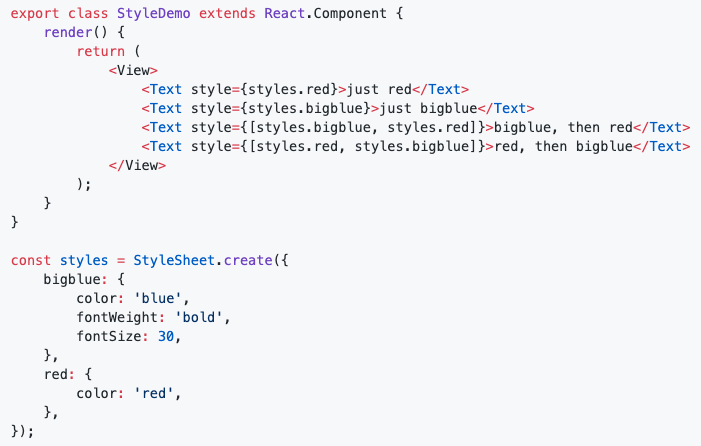
**modules** là thư mục chứa các module do mình định nghĩa hoặc tùy biến lại. Trong đó bao gồm:

screens - module chứa toàn bộ xử lý màn hình của ứng dụng

views - module chứa toàn bộ view đã được custom.

Và một số module mình muốn chỉnh sửa từ thư viện, thì có thể thêm vào đây để tùy biến.

#### Thiết kế View (Style).



Hình 2-8 Thiết kế style

Một vài lưu ý: Không nên quá rườm rà code ngắn nhưng đạt được yêu cầu là tốt nhất.

#### Các Component thường sử dụng.

##### View

Là một component cũng thường xuyên được sử dụng. Thường được sử dụng với mục đích chia các view con theo hàng dọc hoặc hàng ngang dựa vào thuộc tính flexDirection trong style là 'column/row' (dọc / ngang), hoặc sử dụng để chứa nhiều view con hoặc khi cần in ra màn hình một view không hiển thị gì hết.

flex: 1 ở style sẽ giúp kéo view rộng hết khung chứa có thể.

##### Text

Dùng để hiển thị 1 message lên màn hình. Có thể sử dụng text cố định hoặc in nội dung của một biến lên màn hình.



Hình 2-9 Ví dụ Text

##### TextInput



Hình 2-10 Ví dụ TextInput

onChangeText: bắt sự kiện khi thay đổi text

##### Button

Cách sử dụng một Button



Hình 2-11 Ví dụ Button

onPress: nhận vào một hàm call back, là sự kiện khi người dùng nhấn nút

##### TouchableOpacity

Thông thường thay thế việc sử dụng Button bằng TouchableOpacity để việc định dạng style giống nhau cho cả Android và iOS.



Hình 2-12 Ví dụ TouchableOpacity

onPress: nhận vào một hàm call back, là sự kiện khi người dùng nhấn nút

##### Flatlist

Đây là Component thường được sử dụng để hiển thị 1 danh sách lên màn hình.



Hình 2-13 Ví dụ Flatlist

Data: nhận vào là một mảng

renderItem: Hiện giao diện của từng item trong data

keyExtrator: key của từng item để phân biệt từng item

##### Modal



Hình 2-14 Ví dụ Modal

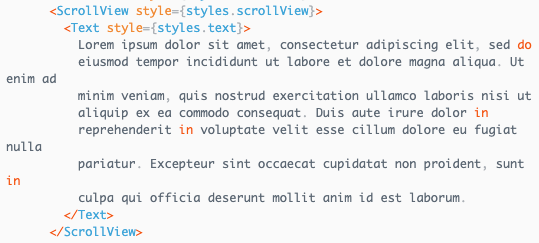
animationType: là hiện ứng khi mở Modal

transparent: trông suốt

visible: trạng thái đóng mở (true/false)

onRequestClose: nhận vào sự kiện khi đóng Modal

##### ScrollView



Hình 2-15 Ví dụ ScrollView

##### Switch

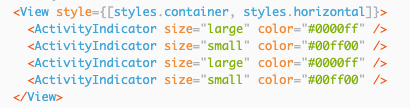


Hình 2-16 Ví dụ Switch

onValueChange: nhận sự kiện khi thay đổi trạng thái của Switch

value: giá trị của switch (true/false)

##### ActivityIndicator



Hình 2-17 Ví dụ ActivityIndicator

##### Alert



Hình 2-18 Ví dụ Alert

#### Các thư viện bên ngoài thường dùng

##### Thư viện chọn màu

react-native-color-picker



Hình 2-19 Ví vụ react-native-color-picker

##### Thư viện thao tác vuốt chạm trên màn hình

react-native-gesture-handler



Hình 2-20 Ví dụ react-native-gesture-handler

##### Thư viện chọn ngày, giờ

react-native-modal-datetime-picker



Hình 2-21 Ví dụ react-native-modal-datetime-picker

##### Thư viện lịch chọn ngày trong tuần

react-native-calendar-strip

##### Thư viện chọn lịch chọn ngày trong tháng

react-native-calendars

##### Thư viện picker-select

react-native-picker-select



Hình 2-22 Ví dụ react-native-picker-select

##### ***Thư viện Icon***

react-native-vector-icons



Hình 2-23 Ví dụ react-native-vector-icons

##### Thư viện kiểm tra kết nối mạng

@react-native-community/netinfo



Hình 2-24 Ví dụ @react-native-community/netinfo

##### Thư viện biểu đồ

react-native-charts-wrapper



Hình 2-25 Ví dụ react-native-charts-wrapper

#### Chuyển đổi giữa các màn hình.

Điều hướng đến một màn hình mới



Hình 2-26 Điều hướng đến một màn hình mới

Quay trở lại



Hình 2-27 Quay trở lại

Tóm tắt:

navigation.navigate('RouteName') đẩy một tuyến mới đến bộ điều hướng ngăn xếp nếu nó chưa có trong ngăn xếp, nếu không nó sẽ nhảy tới màn hình đó.

Có thể gọi navigation.push('RouteName')bao nhiêu lần tùy thích và nó sẽ tiếp tục đẩy các tuyến.

Thanh tiêu đề sẽ tự động hiển thị nút quay lại, nhưng có thể lập trình quay lại bằng cách gọi navigation.goBack(). Trên Android, nút quay lại phần cứng chỉ hoạt động như mong đợi.

Có thể quay lại màn hình hiện có trong ngăn xếp với navigation.navigate('RouteName') và có thể quay lại màn hình đầu tiên trong ngăn xếp với navigation.popToTop().

Các navigationprop hiện có sẵn cho tất cả các thành phần màn hình (thành phần định nghĩa là màn hình trong cấu hình tuyến đường và hiển thị bởi Phản ứng Navigation như một tuyến đường).

#### Kỹ thuật Debug.

Xây dựng ứng dụng React Native khác với ứng dụng native là không thể đặt break point rồi chạy và chờ chương trình nhảy vào vị trí mà đợi và xem trạng thái hay biến lúc đó bằng bao nhiêu đang như thế nào. Thay vì vậy chương trình

React Native cho phép in giá trị tại thời điểm đó và xuất ra màn hình console.

Sử dụng lệnh console.log(variable) để in giá trị của biến bất kì (xem ví dụ phía trên để biết việc in giá trị của biến message trong state).

Xcode và Android studio mặc định khi run debug sẽ xuất các log này ra trong phần All Output (Xcode), Logcat (Android Studio).

Bên cạnh đó có thể sử dụng chức năng Debug JS Remotely (xem phần hiển thị menu điều khiển trong mục 4) để thấy các log này trong phần console của trình duyệt web.

Ngoài ra có thể sử dụng terminal (cmd) để xem log iOS hoặc Android bằng cách gõ lệnh:

react-native log-ios //or

react-native log-android

## Firebase

### Firebase là gì?

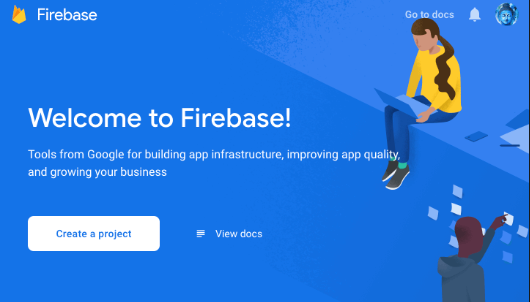
Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu. Cụ thể là những giao diện lập trình ứng dụng API đơn giản. Mục đích nhằm tăng số lượng người dùng và thu lại nhiều lợi nhuận hơn.

Ưu điểm: Triển khai ứng dụng nhanh chóng, bảo mật, ổn định

Nhược điểm: Cơ sở dữ liệu của được tổ chức theo kiểu trees, parent-children, không phải là kiểu Table nên đã quen với SQL có thể sẽ gặp khó khăn khi mới tiếp cận.

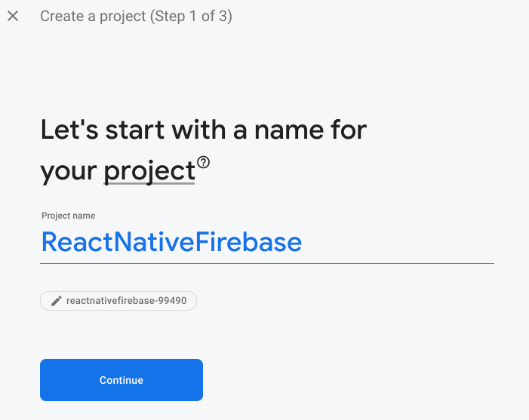
### Cấu hình Firebase trong project app

Truy cập trang: console.firebase.google.com

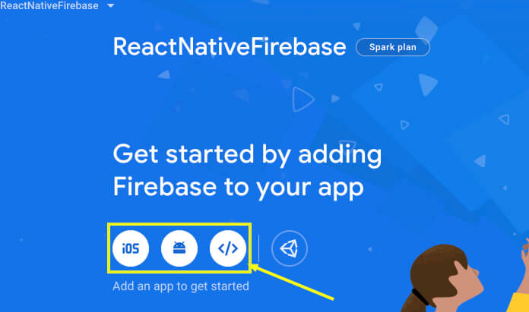


Hình 2-28 Tạo project trong Firebase

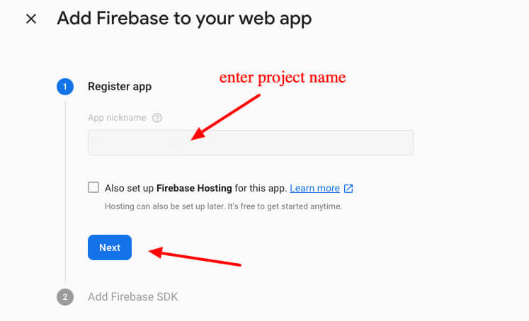
Tạo tài khoản Firebase từ bảng điều khiển Firebase. Nhấp vào trên Tạo Tạo một dự án Nút nút và tạo một dự án xác thực Firebase hoàn toàn mới.



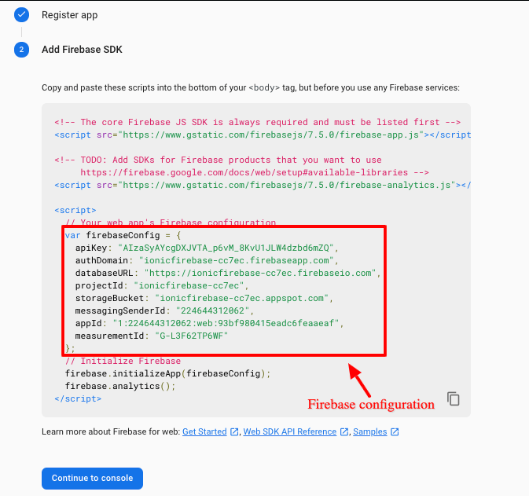
Hình 2-29 Đặt tên cho project



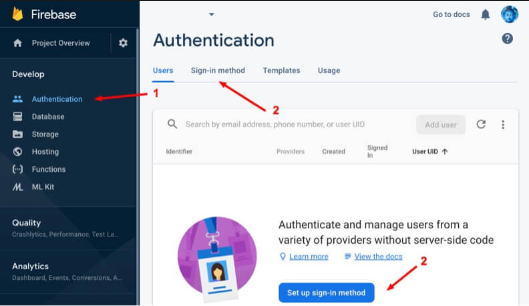
Hình 2-30 Chọn nên tảng



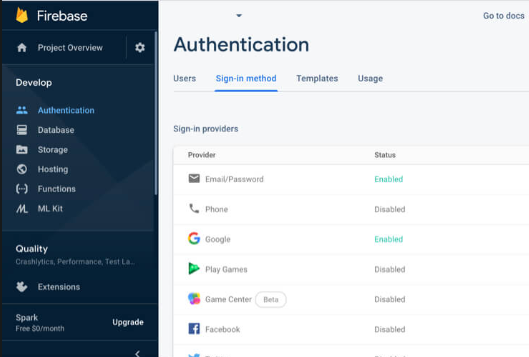
Hình 2-31 Đặt tên cho project



Hình 2-32 Cấu hình Firebase



Hình 2-33 Cấu hình đăng nhập



Hình 2-34 Kích hoạt đăng nhập bằng email, password

### Cơ sở dữ liệu Cloud Firestore

#### Cloud Firestore là gì

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.

#### Adding documents



Hình 2-35 Ví dụ insert data

#### Update documents



Hình 2-36 Ví dụ update data

#### Remove documents



Hình 2-37 Ví dụ remote data

#### Realtime changes



Hình 2-38 Ví dụ realtime changes

#### Query data

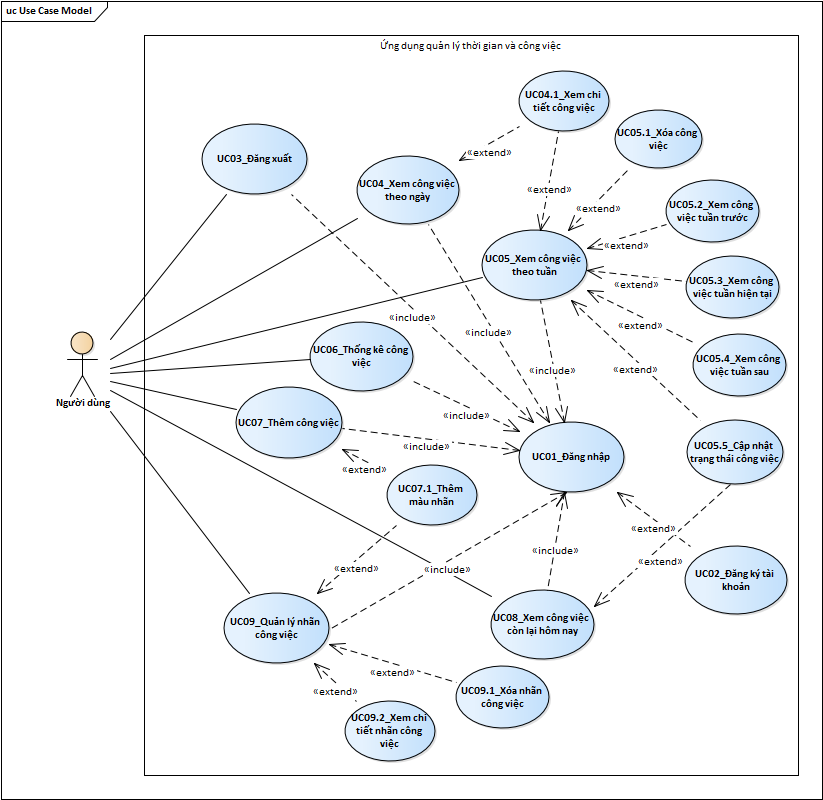


Hình 2-39 Ví dụ query data

# : PHÂN TÍCH

## Phân tích yêu cầu bằng UML

### Use case tổng quát

  
Hình 3-1 Use case diagram

### Danh sách tác nhân và mô tả

|  |  |
| --- | --- |
| Tác nhân | Mô tả tác nhân |
| Người dùng | Là tác nhân tương tác trực tiếp với ứng dụng, thực hiện các chức năng của ứng dụng |

Bảng 3-1 Danh sách tác nhân và mô tả

### Danh sách các tình huống hoạt động (use case)

|  |  |
| --- | --- |
| ID | Tên use case |
| UC01 | Đăng nhập |
| UC02 | Đăng ký tài khoản |
| UC03 | Đăng xuất |
| UC04 | Xem công việc theo ngày |
| UC04.1 | Xem chi tiết công việc |
| UC05 | Xem công việc theo tuần |
| UC05.1 | Xoá công việc |
| UC05.2 | Xem công việc tuần trước |
| UC05.3 | Xem công việc tuần hiện tại |
| UC05.4 | Xem công việc tuần sau |
| UC05.5 | Cập nhật trạng thái công việc |
| UC06 | Thống kê công việc |
| UC07 | Thêm công việc |
| UC07.1 | Thêm nhãn công việc |
| UC08 | Xem công việc còn lại hôm nay |
| UC08.1 | Cập nhật trạng thái công việc |
| UC09 | Quản lý nhãn công việc |
| UC09.1 | Xóa nhãn công việc |
| UC09.2 | Xem chi tiết nhãn công việc |

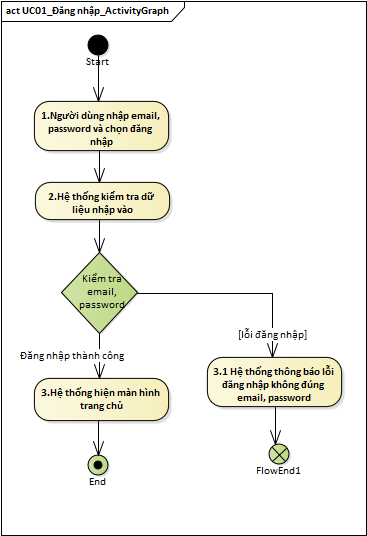
Bảng 3-2 Danh sách các tình huống hoạt động

### Mô tả tình huống hoạt động

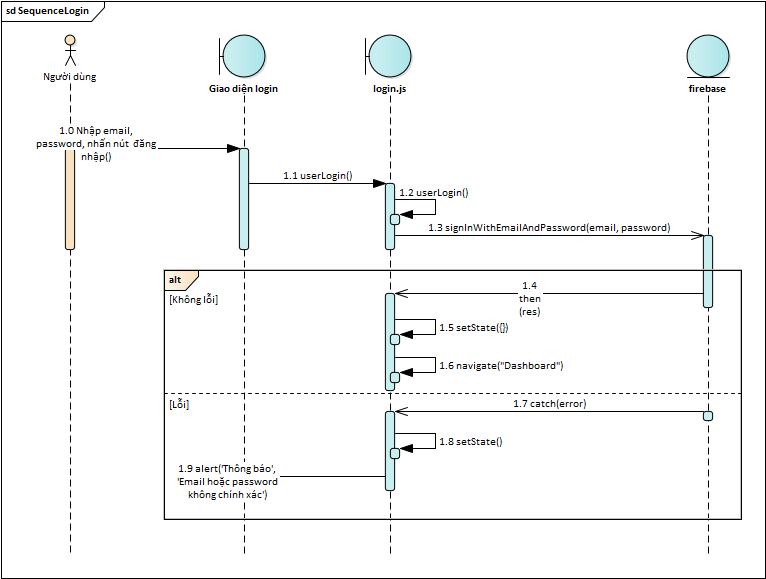
#### Đặc tả use case đăng nhập

|  |  |  |
| --- | --- | --- |
| ID | UC01 | |
| Tên use case | Đăng nhập | |
| Mô tả | Người dùng nhập email, password đăng nhập vào ứng dụng | |
| Tiền điều kiện | Chưa đăng nhập vào hệ thống | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhập email, password và chọn đăng nhập |  |
|  | 1. Hệ thống kiểm tra dữ liệu nhập vào. |
|  | 1. Hệ thống hiện màn hình trang chủ |
| Luồng sự kiện phụ |  | 1. Hệ thống thông báo lỗi đăng nhập không đúng email, password |
| Luồng ngoại lệ |  | |

Bảng 3-3 Đặt tả use case đăng nhập



Hình 3-2 Activity diagram đăng nhập

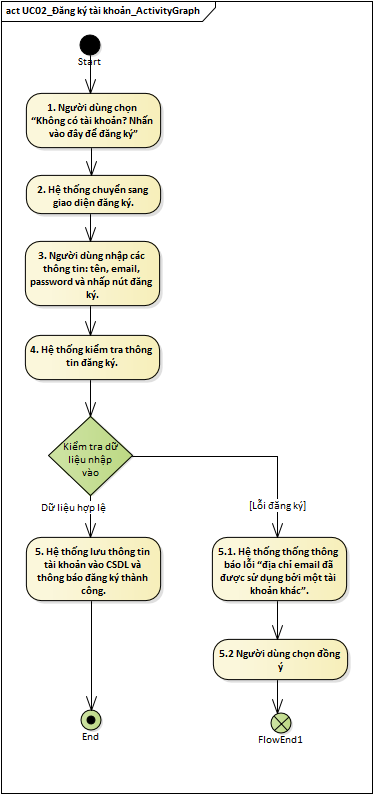


Hình 3-3 Sequence diagram đăng nhập

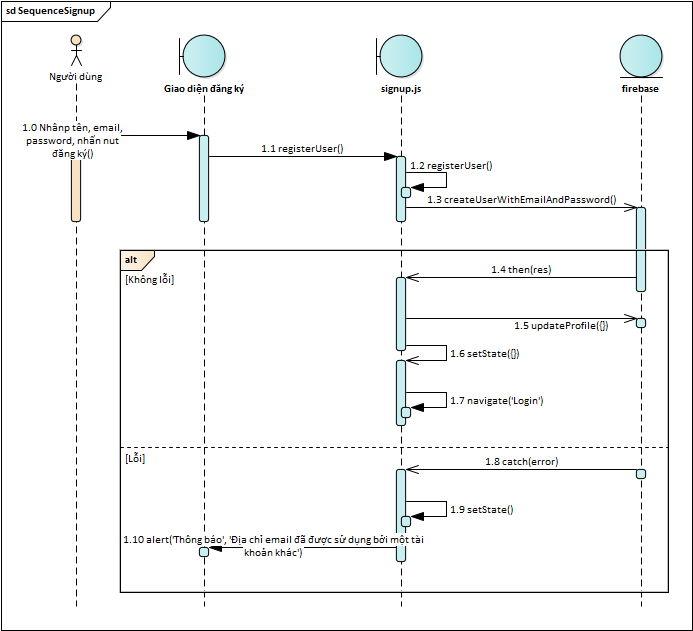
#### Đặc tả use case đăng ký tài khoản

|  |  |  |
| --- | --- | --- |
| ID | UC02 | |
| Tên use case | Đăng ký tài khoản | |
| Mô tả | Người dùng chưa có tài khoản, nhập tên, email, password để đăng ký tài khoản. | |
| Tiền điều kiện | Người dùng chưa đăng nhập vào ứng dụng và đang ở giao diện đăng nhập | |
| Hậu điều kiện | Tài khoản được đăng ký thành công và lưu vào CSDL | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn “Không có tài khoản? Nhấn vào đây để đăng ký” |  |
|  | 1. Hệ thống chuyển sang giao diện đăng ký |
| 1. Người dùng nhập các thông tin: tên, email, password và nhấp nút đăng ký |  |
|  | 1. Hệ thống kiểm tra thông tin đăng ký |
|  | 1. Hệ thống lưu thông tin tài khoản vào CSDL và thông báo đăng ký thành công |
| Luồng sự kiện phụ |  | 1. Hệ thống thống thông báo lỗi “địa chỉ email đã được sử dụng bởi một tài khoản khác” |
| 1. Người dùng chọn đồng ý |  |

Hình 3-4 Đặc tả use case đăng ký tài khoản



Hình 3-5 Activity diagram đăng ký tài khoản

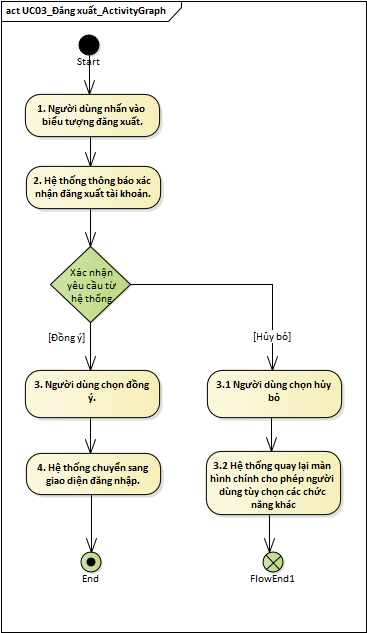


Hình 3-6 Sequence diagram đăng ký tài khoản

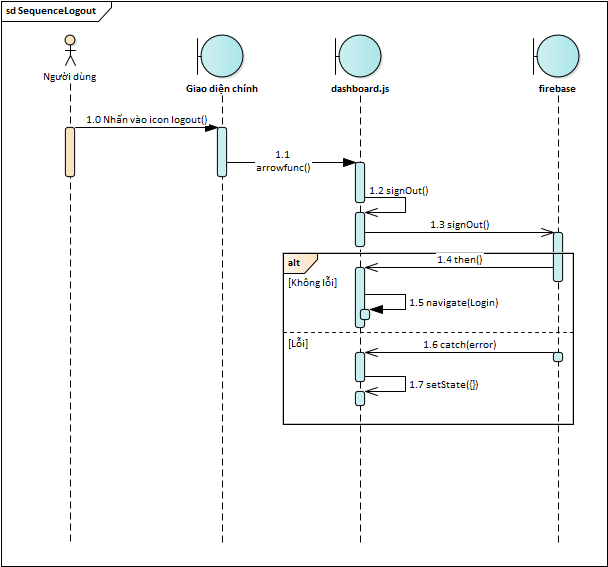
#### Đặc tả use case đăng xuất.

|  |  |  |
| --- | --- | --- |
| ID | UC03 | |
| Tên use case | Đăng xuất | |
| Mô tả | Cho phép người dùng đăng xuất thông tin tài khoản khỏi ứng dụng. | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, đang ở màn hình chính | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhấn vào biểu tượng đăng xuất |  |
|  | 1. Hệ thống thông báo xác nhận đăng xuất tài khoản |
| 1. Người dùng chọn đồng ý |  |
|  | 1. Hệ thống chuyển sang giao diện đăng nhập |
| Luồng sự kiện phụ | 1. Người dùng chọn hủy bỏ | 1. Hệ thống quay lại màn hình chính cho phép người dùng tùy chọn các chức năng |

Bảng 3-4 Đặc tả use case đăng xuất



Hình 3-7 Activity diagram đăng xuất



Hình 3-8 Sequence diagram đăng xuất

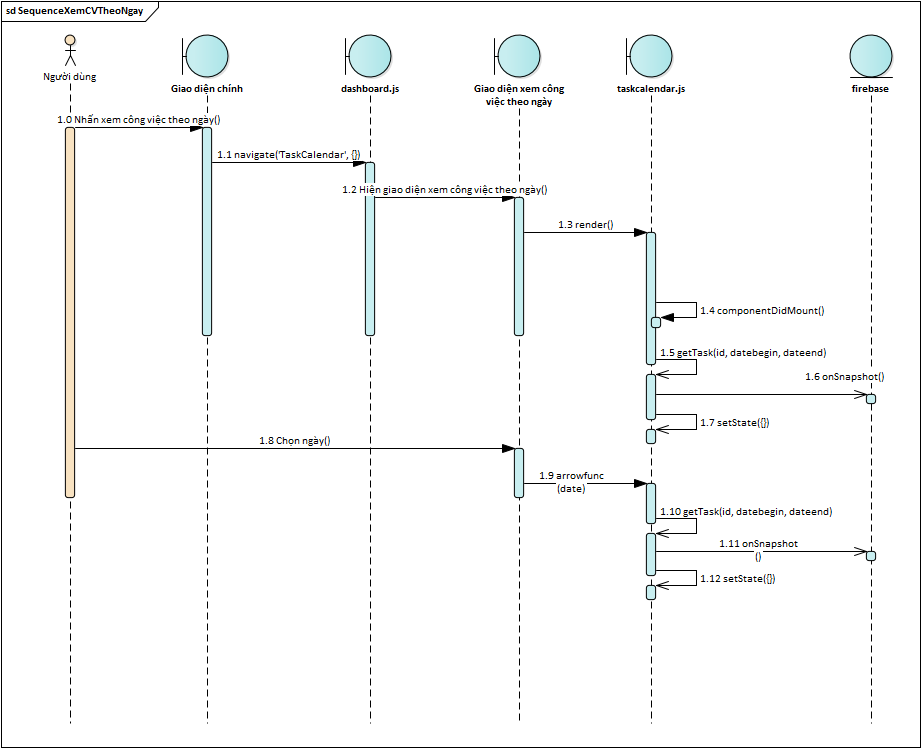
#### Đặc tả use case xem công việc theo ngày

|  |  |  |
| --- | --- | --- |
| ID | UC04 | |
| Tên use case | Xem công việc theo ngày | |
| Mô tả | Cho phép người dùng chọn ngày để xem công việc, mặc định là ngày hiện tại, cho phép chuyển qua lại giữa các tuần, thông tin hiển thị của mỗi công việc gồm: thời gian bắt đầu, tên công việc, ghi chú, màu sắc, trạng thái công việc | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, đang ở giao diện chính | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn xem công việc theo ngày |  |
|  | 1. Hệ thống hiện giao diện xem công việc theo ngày |
| 1. Người dùng chọn ngày muốn xem |  |
|  | 1. Hệ thống hiện các công việc theo ngày người dùng đã chọn |
| Luồng sự kiện phụ | Không | |

Bảng 3-5 Đặc tả use case xem công việc theo ngày



Hình 3-9 Activity diagram xem công việc theo ngày

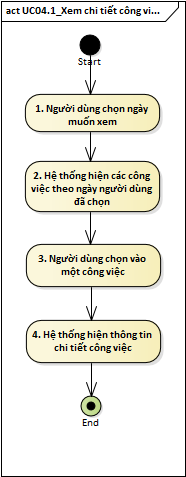


Hình 3-10 Sequence diagram xem công việc theo ngày

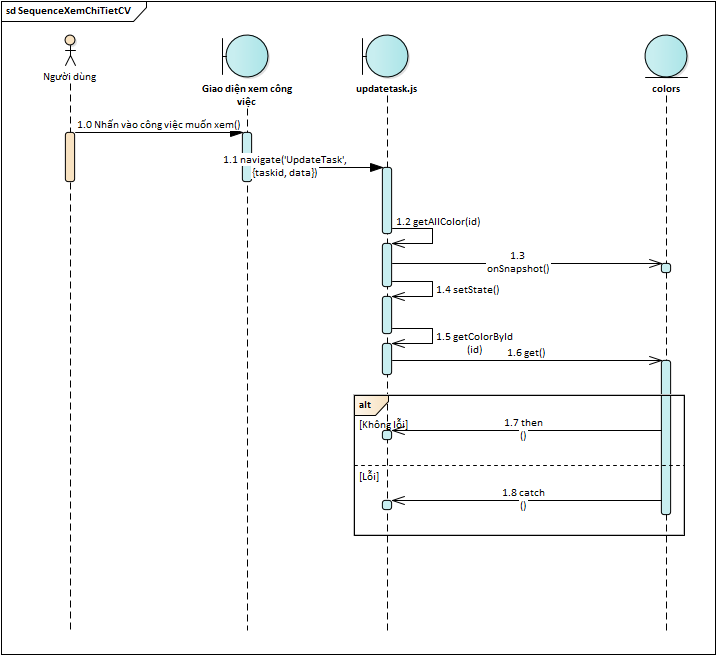
#### Đặc tả use case xem chi tiết công việc

|  |  |  |
| --- | --- | --- |
| ID | UC04.1 | |
| Tên use case | Xem chi tiết công việc | |
| Mô tả | Cho phép người dùng xem đầy đủ thông tin công việc gồm: thời gian bắt đầu, tên công việc, ghi chú, nhắn nhở, trạng thái, màu sắc, và tùy chọn các chức năng cập nhật, xóa công việc. | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, và đang ở giao diện xem công việc theo ngày. | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn ngày muốn xem |  |
|  | 1. Hệ thống hiện các công việc theo ngày người dùng đã chọn |
| 1. Người dùng chọn vào một công việc |  |
|  | 1. Hệ thống hiện thông tin chi tiết công việc |
| Luồng sự kiện phụ |  | |

Bảng 3-6 Đặc tả use case xem chi tiết công việc



Hình 3-11 Activity diagram xem chi tiết công việc



Hình 3-12 Sequence diagram xem chi tiết công việc

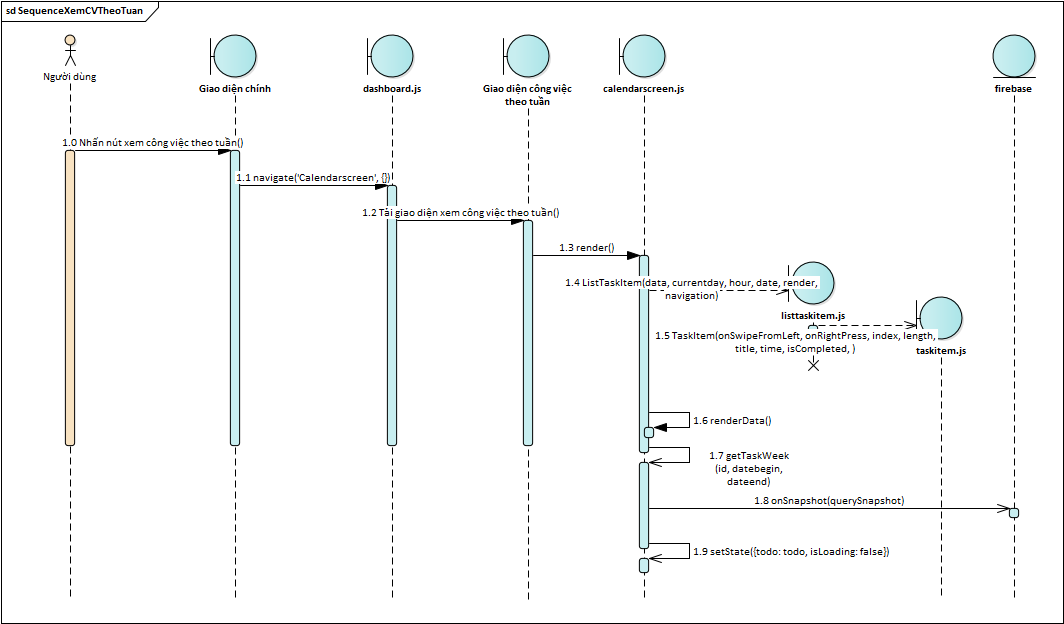
#### Đặc tả use case xem công việc theo tuần

|  |  |  |
| --- | --- | --- |
| ID | UC05 | |
| Tên use case | Xem công việc theo tuần | |
| Mô tả | Sau khi đăng nhập vào giao diện chính, người dùng chọn chức năng xem công việc theo tuần, thông tin hiển thị của màn hình gồm: bảy cột tương ứng bảy ngày trong tuần, cột đầu tiên bên trái là các mốc thời gian từ 6-24h. Cho phép chuyển qua lại giữa các tuần, thông tin hiển thị của mỗi công việc gồm: thời gian bắt đầu, tên công việc, màu sắc, trạng thái công việc | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, đang ở giao diện chính | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn xem công việc theo tuần |  |
|  | 1. Hệ thống hiện giao diện xem công việc theo tuần |
|  | 1. Hệ thống cho phép người dùng tùy chọn các chức năng khác |
| Luồng sự kiện phụ |  | |

Bảng 3-7 Đặc tả use case xem công việc theo tuần



Hình 3-13 Activity diagram xem công việc theo tuần



Hình 3-14 Sequence diagram xem công việc theo tuần

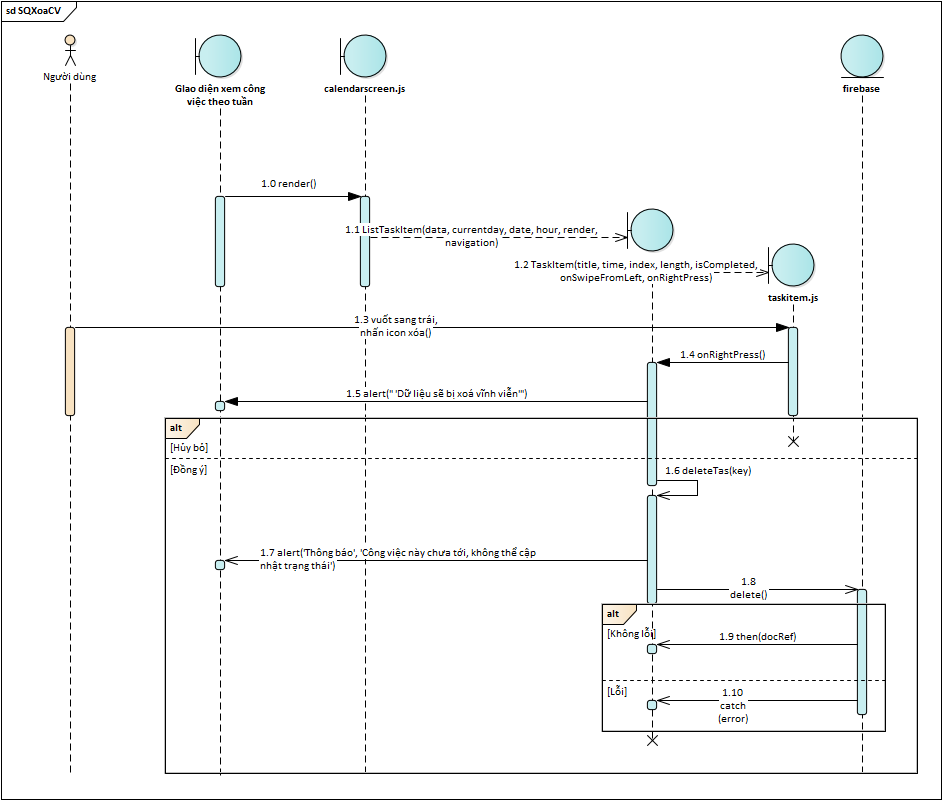
#### Đặc tả use case xóa công việc

|  |  |  |
| --- | --- | --- |
| ID | UC05.1 | |
| Tên use case | Xóa công việc | |
| Mô tả | Người dùng vuốt từ phải sang trái của công việc muốn xóa và nhấn biểu tượng xóa, công việc sẽ được xóa ra khỏi hệ thống | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện xem công việc theo tuần. | |
| Hậu điều kiện | Công việc được xoá ra khỏi CSDL | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng vuốt từ phải sang trái ở công việc muốn xóa và nhấn vào biểu tượng xóa |  |
|  | 1. Hệ thống hiện thông báo xác nhận xóa công việc |
| 1. Người dùng nhấn nút đồng ý |  |
|  | 1. Hệ thống xóa công việc ra khỏi CSDL |
| Luồng sự kiện phụ |  | |

Bảng 3-8 Đặc tả use case xóa công việc



Hình 3-15 Activity diagram xóa công việc

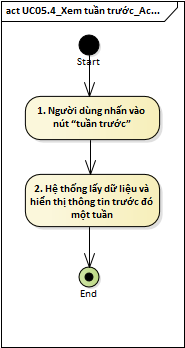


Hình 3-16 Sequence diagram xóa công việc

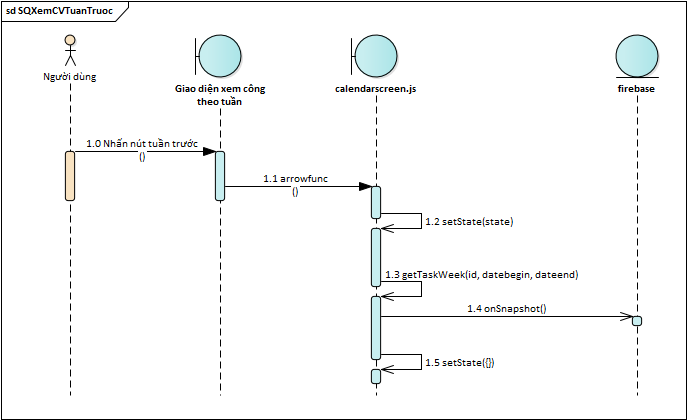
#### Đặc tả use case xem công việc tuần trước

|  |  |  |
| --- | --- | --- |
| ID | UC05.2 | |
| Tên use case | Xem công việc tuần trước | |
| Mô tả | Khi người dùng nhấn vào nút “tuần trước”, dữ liệu sẽ được tải lại thời gian trước đó một tuần | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện xem công việc theo tuần. | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhấn vào nút “tuần trước” |  |
|  | 1. Hệ thống lấy dữ liệu và hiển thị thông tin trước đó một tuần |
| Luồng sự kiện phụ |  | |

Bảng 3-9 Đặc tả use case xem công việc tuần trước



Hình 3-17 Activity diagram xem công việc tuần trước

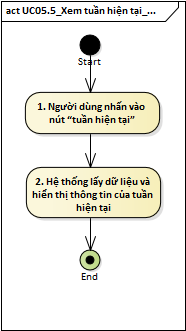


Hình 3-18 Sequence diagram xem công việc tuần trước

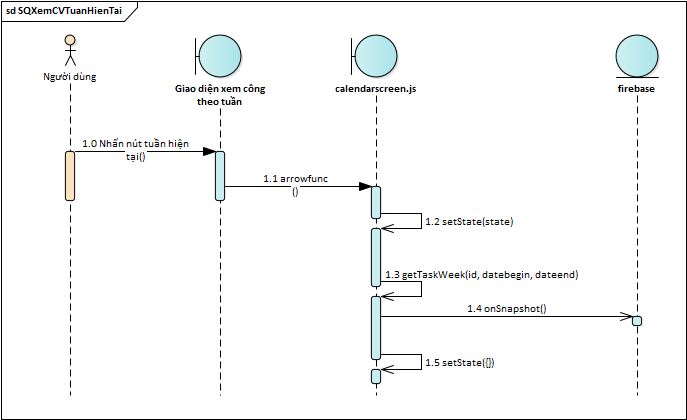
#### Đặc tả use case xem công việc tuần hiện tại

|  |  |  |
| --- | --- | --- |
| ID | UC05.3 | |
| Tên use case | Xem công việc tuần hiện tại | |
| Mô tả | Khi người dùng nhấn vào nút “tuần hiện tại”, dữ liệu sẽ được tải lại của tuần hiện tại. | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện xem công việc theo tuần. | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhấn vào nút “tuần hiện tại” |  |
|  | 1. Hệ thống lấy dữ liệu và hiển thị thông tin của tuần hiện tại |
| Luồng sự kiện phụ |  | |

Bảng 3-10 Đặc tả use case xem công việc tuần hiện tại



Hình 3-19 Activity diagram xem công việc tuần hiện tại

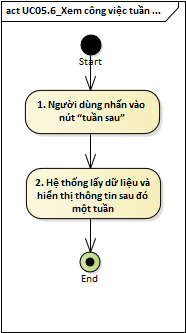


Hình 3-20 Sequence diagram xem công việc tuần hiện tại

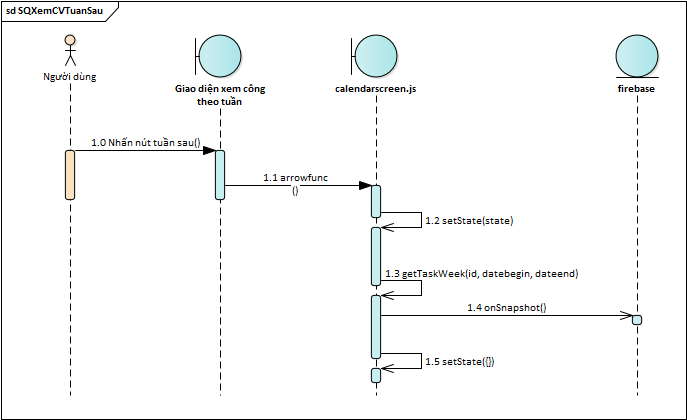
#### Đặc tả use case xem công việc tuần sau

|  |  |  |
| --- | --- | --- |
| ID | UC05.4 | |
| Tên use case | Xem công việc tuần sau | |
| Mô tả | Khi người dùng nhấn vào nút “tuần sau”, dữ liệu sẽ được tải lại thời gian sau đó một tuần | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện xem công việc theo tuần | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhấn vào nút “tuần sau” |  |
|  | 1. Hệ thống lấy dữ liệu và hiển thị thông tin sau đó một tuần |
| Luồng sự kiện phụ |  | |

Bảng 3-11 Đặc tả use case xem công việc tuần sau



Hình 3-21 Activity diagram xem công việc tuần sau

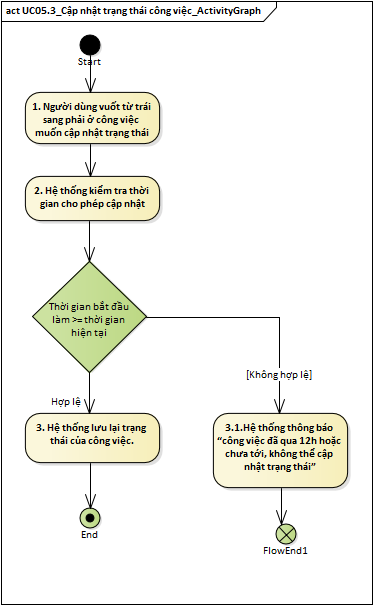


Hình 3-22 Sequence diagram xem công việc tuần sau

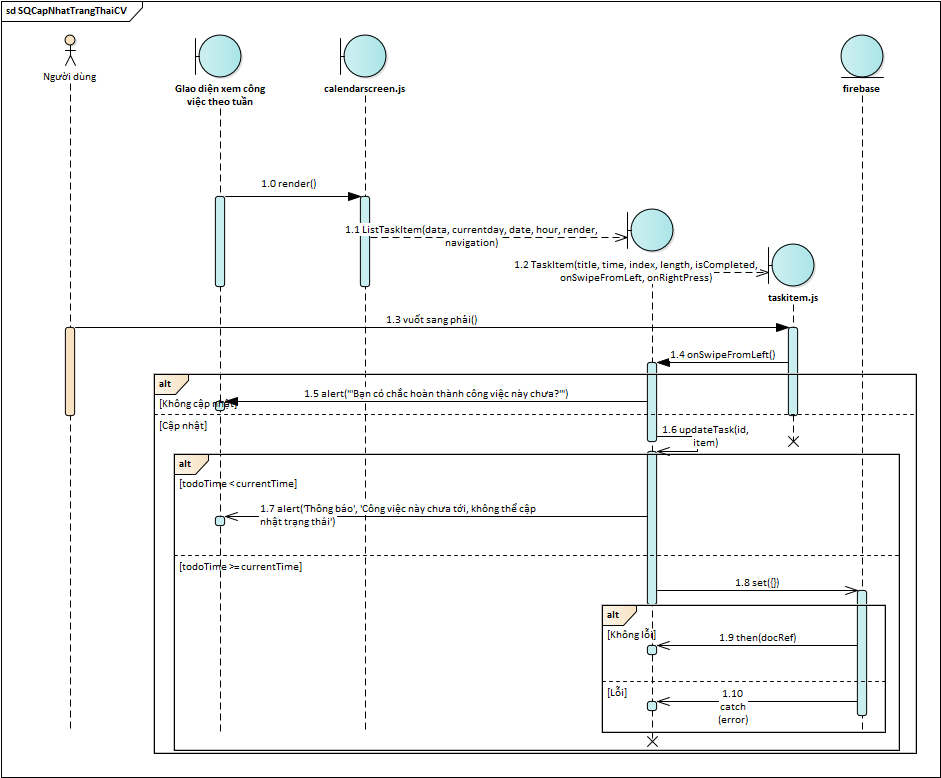
#### Đặc tả use case cập nhật trạng thái công việc

|  |  |  |
| --- | --- | --- |
| ID | UC05.5 | |
| Tên use case | Cập nhật trạng thái công việc | |
| Mô tả | Ở giao diện xem công việc còn lại hôm nay người dùng vuốt từ trái sang phải ở công việc muốn cập nhật trạng thái. Thời gian cho phép cập nhật trạng thái công việc không quá 12h kể từ khi bắt đầu | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện xem công việc theo tuần. | |
| Hậu điều kiện | Trạng thái công việc được cập nhật lại | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng vuốt từ trái sang phải ở công việc muốn cập nhật trạng thái |  |
|  | 1. Hệ thống kiểm tra thời gian cho phép cập nhật |
|  | 1. Hệ thống lưu lại trạng thái của công việc |
| Luồng sự kiện phụ |  | 1. Hệ thống thông báo “công việc chưa tới, không thể cập nhật trạng thái” |

Bảng 3-12 Đặc tả use case cập nhật trạng thái công việc



Hình 3-23 Activity diagram cập nhật trạng thái công việc

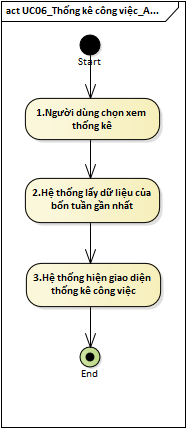


Hình 3-24 Sequence diagram cập nhật trạng thái công việc

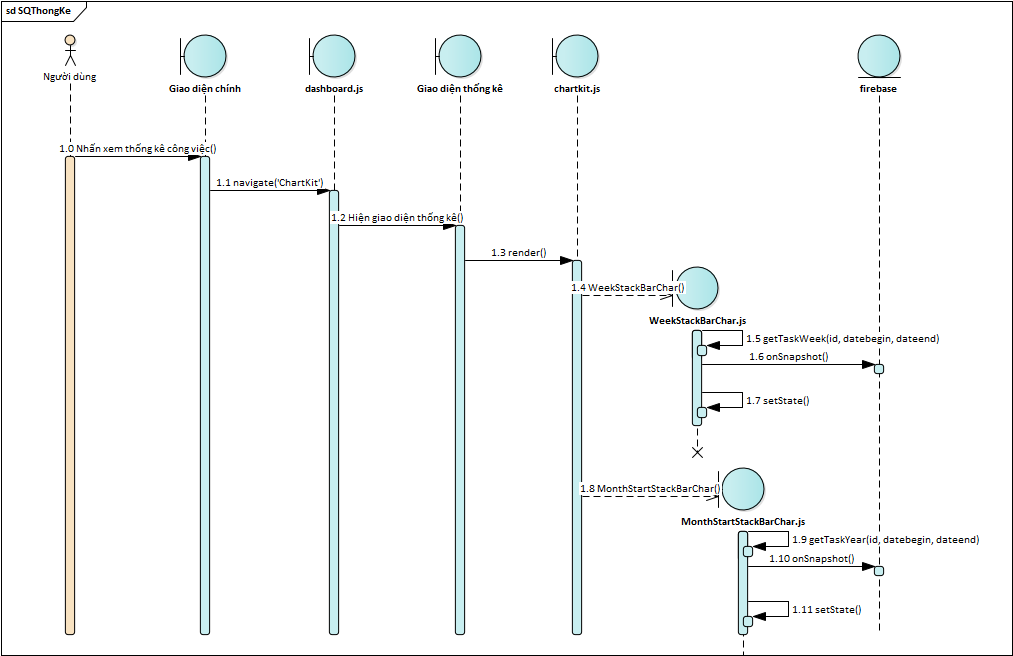
#### Đặc tả use case thống kê công việc

|  |  |  |
| --- | --- | --- |
| ID | UC06 | |
| Tên use case | Thống kê công việc | |
| Mô tả | Khi người dùng chọn chức năng xem thống kê, hệ thống hiện dữ liệu thống kê hằng tuần | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện chính | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn xem thống kê |  |
|  | 1. Hệ thống lấy dữ liệu của bốn tuần gần nhất |
|  | 1. Hệ thống hiện giao diện thống kê công việc |
| Luồng sự kiện phụ |  | |

Bảng 3-13 Đặc tả use case thống kê công việc



Hình 3-25 Activity diagram xem thống kê công việc

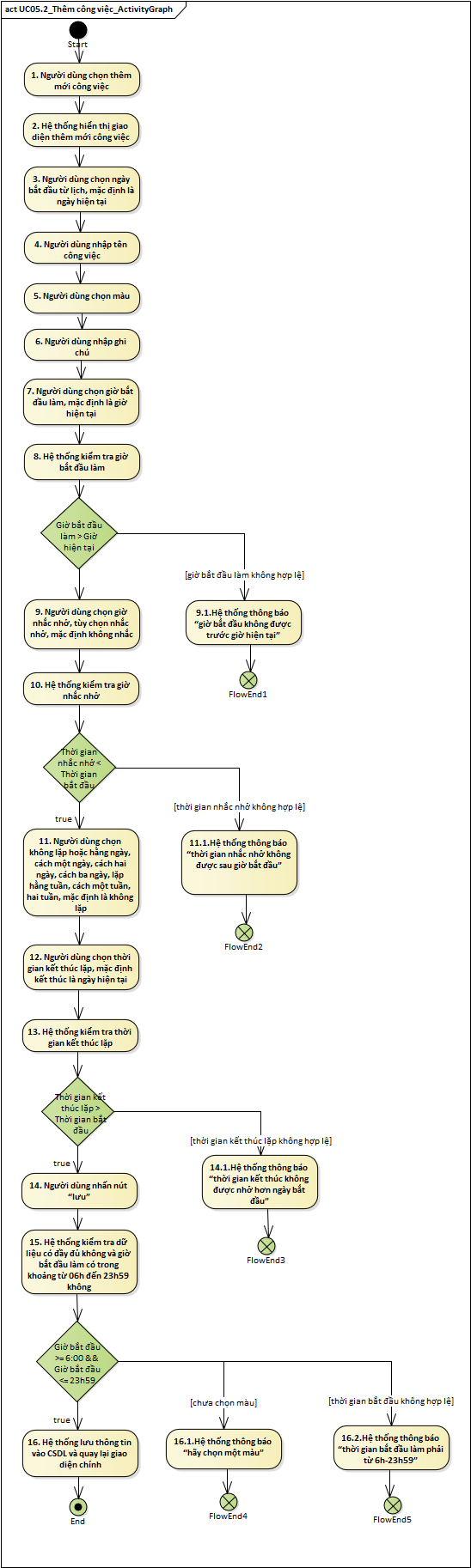


Hình 3-26 Sequence diagram xem thống kê công việc

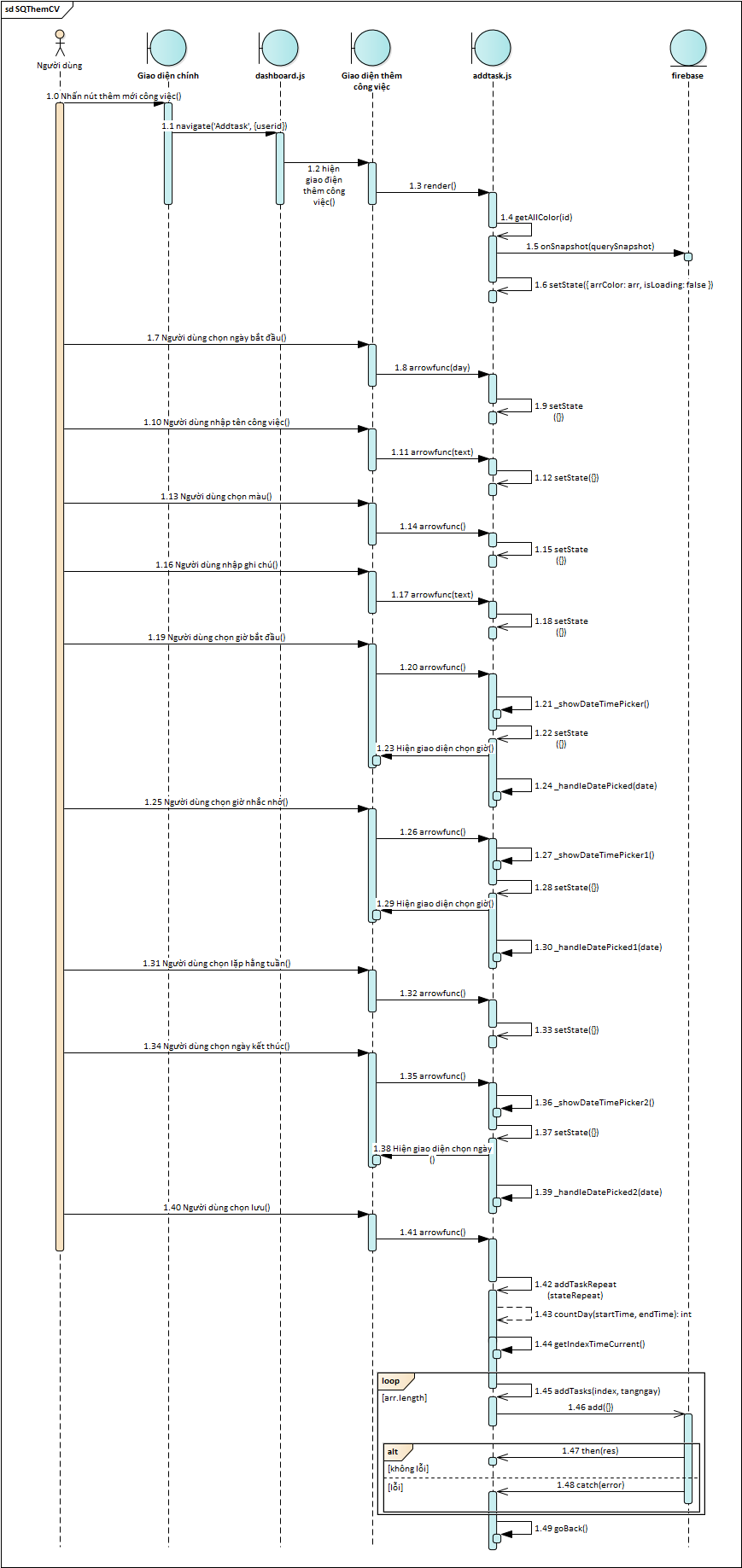
#### Đặc tả use case thêm công việc

|  |  |  |
| --- | --- | --- |
| ID | UC07 | |
| Tên use case | Thêm công việc | |
| Mô tả | Khi người dùng chọn chức năng thêm mới công việc, hệ thống hiện giao diện thêm công việc, người dùng chọn ngày bắt đầu làm từ lịch, nhập tên công việc, chọn màu, ghi chú, giờ bắt đầu, thời gian nhắc nhở, ngày kết thúc lặp lại, tùy chọn không lăp, lặp hằng ngày, lặp cách một ngày, cách hai ngày, cách ba ngày, lặp hằng tuần, cách một tuần, hai tuần. Thông tin bắt buộc nhập gồm tên công việc, màu sắc, các thông tin còn lại được thiết lặp mặc định | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện chính | |
| Hậu điều kiện | Công việc mới được lưu lại trong CSDL | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn thêm mới công việc. |  |
|  | 1. Hệ thống hiển thị giao diện thêm mới công việc. |
| 1. Người dùng chọn ngày bắt đầu từ lịch, mặc định là ngày hiện tại |  |
| 1. Người dùng nhập tên công việc. |  |
| 1. Người dùng chọn màu. |  |
| 1. Người dùng nhập ghi chú |  |
| 1. Người dùng chọn giờ bắt đầu làm, mặc định là giờ hiện tại |  |
|  | 1. Hệ thống kiểm tra giờ bắt đầu làm |
| 1. Người dùng chọn giờ nhắc nhở, tùy chọn nhắc nhở, mặc định không nhắc. |  |
|  | 1. Hệ thống kiểm tra giờ nhắc nhở |
| 1. Người dùng chọn không lặp hoặc hằng ngày, cách một ngày, cách hai ngày, cách ba ngày, lặp hằng tuần, cách một tuần, hai tuần, mặc định là không lặp |  |
| 1. Người dùng chọn thời gian kết thúc lặp, mặc định kết thúc là ngày hiện tại |  |
|  | 1. Hệ thống kiểm tra thời gian kết thúc lặp |
| 1. Người dùng nhấn nút “lưu” |  |
|  | 1. Hệ thống kiểm tra dữ liệu có đầy đủ không và giờ bắt đầu làm có trong khoảng từ 06h đến 23h59 không. |
|  | 1. Hệ thống lưu thông tin vào CSDL và quay lại giao diện chính. |
| Luồng sự kiện phụ |  | 1. Hệ thống thông báo “giờ bắt đầu không được trước giờ hiện tại” |
|  | 1. Hệ thống thông báo “thời gian nhắc nhở không được sau giờ bắt đầu” |
|  | 1. Hệ thống thông báo “thời gian kết thúc không được nhở hơn ngày bắt đầu” |
|  | 1. Hệ thống thông báo “hãy chọn một màu” |
|  | 1. Hệ thống thông báo “thời gian bắt đầu làm phải từ 6h-23h59” |

Bảng 3-14 Đặc tả use case thêm công việc



Hình 3-27 Activity diagram thêm công việc

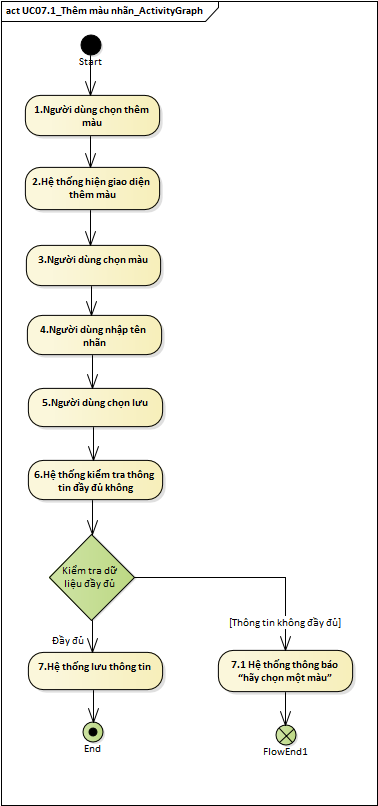


Hình 3-28 Sequence diagram thêm công việc

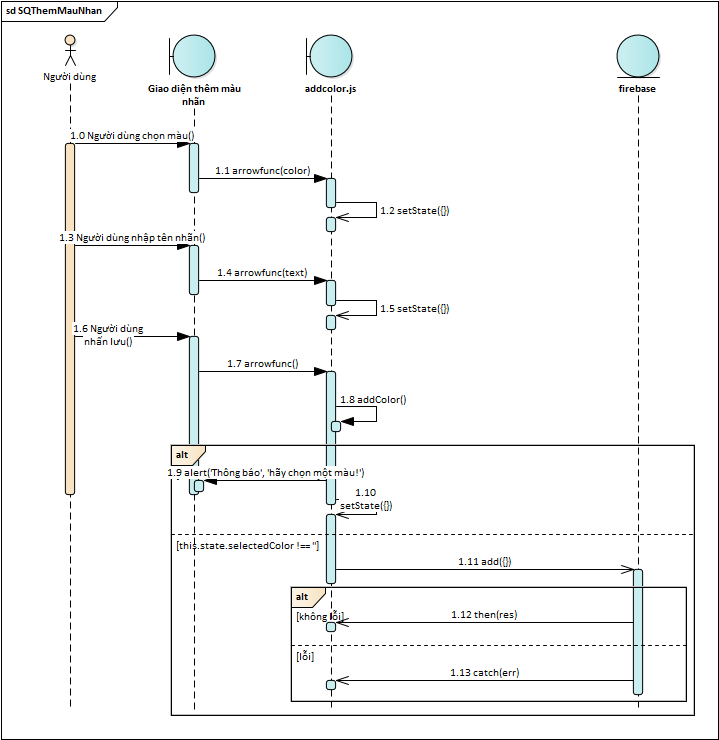
#### Đặc tả use case thêm nhãn công việc

|  |  |  |
| --- | --- | --- |
| ID | UC07.1 | |
| Tên use case | Thêm nhãn công việc | |
| Mô tả | Khi người dùng chọn thêm màu hệ thống sẽ chuyển sang giao diện thêm màu, thông tin màu nhãn gồm: tên nhãn và màu sắc, người dùng nhập tên nhãn và chọn màu và nhấn lưu dữ liệu sẽ được lưu vào CSDL | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng vào giao diện thêm công việc | |
| Hậu điều kiện | Thông tin màu nhãn được lưu vào CSDL | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn thêm màu. |  |
|  | 1. Hệ thống hiện giao diện thêm màu. |
| 1. Người dùng chọn màu. |  |
| 1. Người dùng nhập tên nhãn. |  |
| 1. Người dùng chọn lưu. |  |
|  | 1. Hệ thống kiểm tra thông tin đầy đủ không |
|  | 1. Hệ thống lưu thông tin. |
| Luồng sự kiện phụ |  | 1. Hệ thống thông báo “hãy chọn một màu” |

Bảng 3-15 Đặc tả use case thêm nhãn công việc



Hình 3-29 Activity diagram thêm nhãn công việc



Hình 3-30 Sequence diagram thêm nhãn công việc

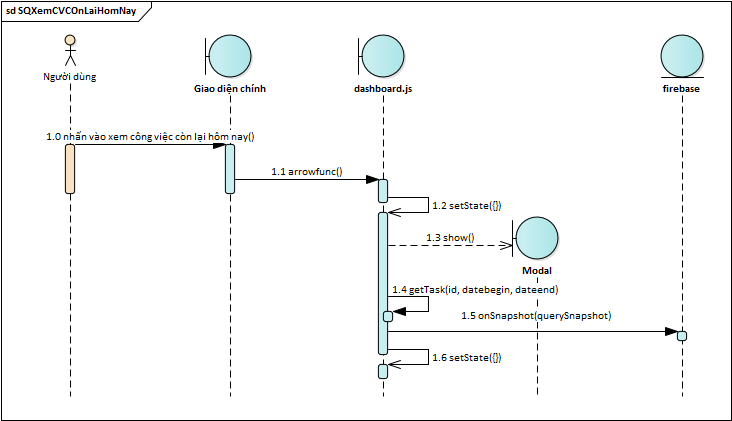
#### Đặc tả use case xem công việc còn lại hôm nay

|  |  |  |
| --- | --- | --- |
| ID | UC08 | |
| Tên use case | Xem công việc còn lại hôm nay | |
| Mô tả | Cho phép người dùng xem nhanh các công việc còn lại hôm nay, tự động thông báo sau khi người dùng đăng nhập, hoặc người dùng chọn xem công việc còn lại hôm nay | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện chính. | |
| Hậu điều kiện | Không | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn xem công việc con lại hôm nay |  |
|  | 1. Hệ thống hiện danh sách các công việc chưa hoàn thành hôm nay |
| Luồng sự kiện phụ |  | |

Bảng 3-16 Đặc tả use case xem công việc hôm nay



Hình 3-31 Activity diagram xem công việc hôm nay

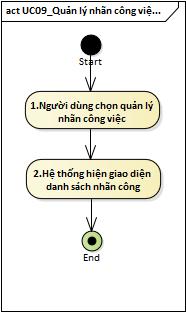


Hình 3-32 Sequence diagram xem công việc hôm nay

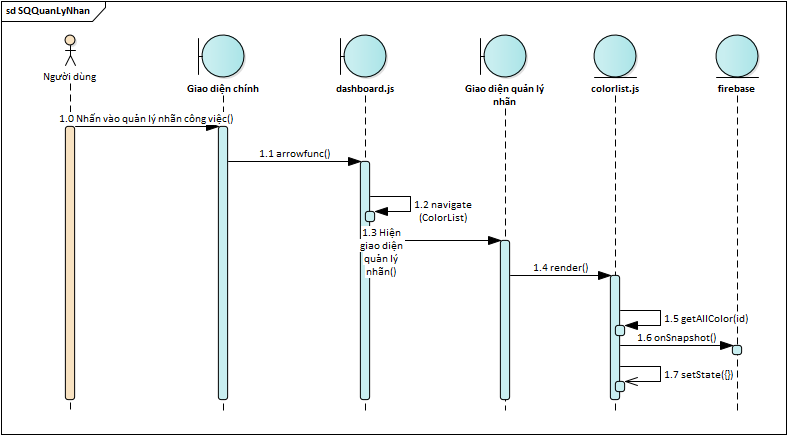
#### Đặc tả use case **quản lý nhãn công việc**

|  |  |
| --- | --- |
| ID | UC09 |
| Tên use case | Quản lý nhãn công việc |
| Mô tả | Khi người dùng chọn quản lý công việc nhãn hệ thống chuyển sang giao diện danh sách nhãn |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện chính |
| Hậu điều kiện | Không |
| Tác nhân | Người dùng |
| Luồng sự kiện chính | 1. Người dùng chọn quản lý nhãn công việc 2. Hệ thống hiện giao diện danh sách nhãn công |
| Luồng sự kiện phụ |  |

Bảng 3-17 Đặc tả use case quản lý nhãn công việc



Hình 3-33 Activity diagram quản lý nhãn công việc

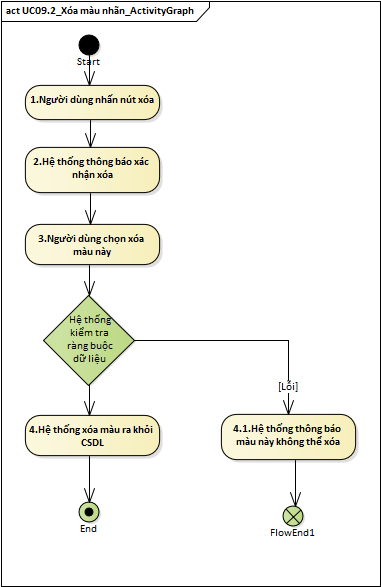


Hình 3-34 Sequence diagram quản lý nhãn công việc

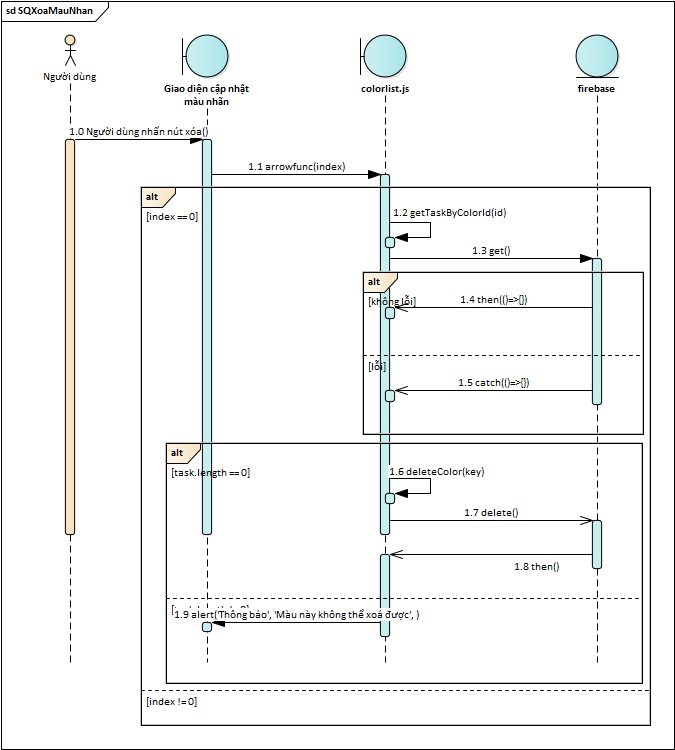
#### Đặc tả use case xóa nhãn công việc

|  |  |  |
| --- | --- | --- |
| ID | UC09.1 | |
| Tên use case | Xóa nhãn công việc | |
| Mô tả | Khi người dùng chọn màu, nhập tên nhãn đễ cập nhật lại thông tin | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện quản lý nhãn. | |
| Hậu điều kiện | Thông tin màu nhãn được xóa khỏi CSDL | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng nhấn nút xóa |  |
|  | 1. Hệ thống thông báo xác nhận xóa |
| 1. Người dùng chọn xóa màu này |  |
|  | 1. Hệ thống xóa màu ra khỏi CSDL |
| Luồng sự kiện phụ |  | 1. Hệ thống thông báo màu này không thể xóa |

Bảng 3-18 Đặc tả use case xóa nhãn công việc



Hình 3-35 Activity diagram xóa nhãn công việc

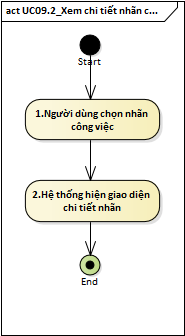


Hình 3-36 Sequence diagram xóa nhãn công việc

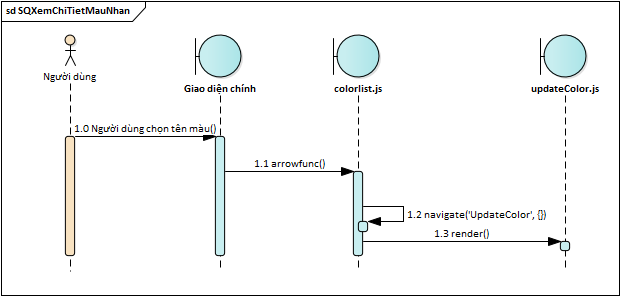
#### Đặc tả use case xem chi tiết nhãn công việc

|  |  |  |
| --- | --- | --- |
| ID | UC09.2 | |
| Tên use case | Xem chi tiết nhãn công việc | |
| Mô tả | Khi người dùng chọn nhãn công việc, hệ thống hiện thông tin chi tiết nhãn | |
| Tiền điều kiện | Người dùng đã đăng nhập vào ứng dụng, ở giao diện quản lý nhãn | |
| Hậu điều kiện | Thông tin màu nhãn được cập nhật | |
| Tác nhân | Người dùng | |
| Luồng sự kiện chính | Người dùng | Hệ thống |
| 1. Người dùng chọn nhãn công việc |  |
|  | 1. Hệ thống hiện giao diện chi tiết nhãn |
| Luồng sự kiện phụ |  | |

Bảng 3-19 Đặc tả use case xem chi tiết nhãn công việc



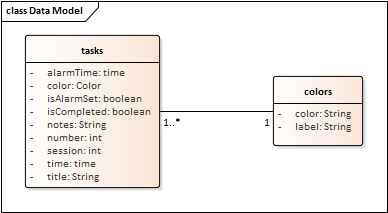
Hình 3-37 Activity diagram xem chi tiết nhãn công việc



Hình 3-38 Sequence diagram xem chi tiết nhãn công việc

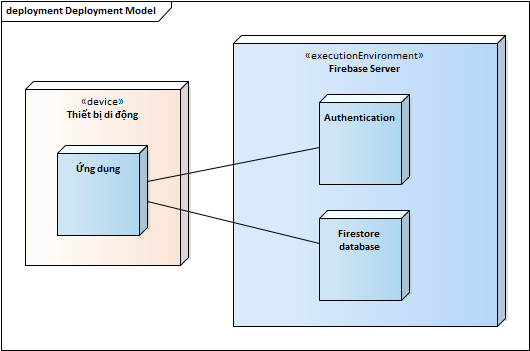
# : THIẾT KẾ VÀ HIỆN THỰC

## Class diagram



Hình 4-1 Sơ đồ class diagram

## Deployment diagram



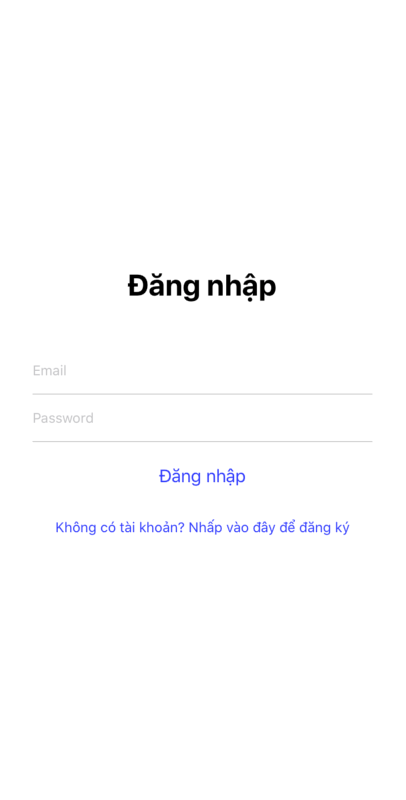
Hình 4-2 Sơ đồ Deployment của ứng dụng

## Hiện thực

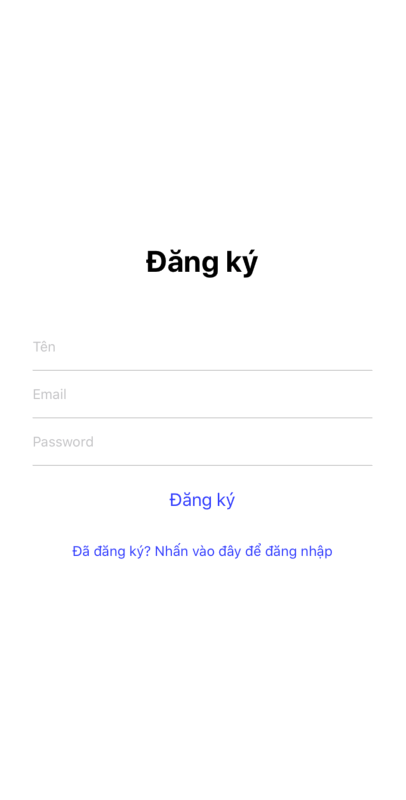
### Cấu hình phần cứng thiết bị và môi trường chạy

* Cấu hình phần cứng test:
* macOS 10.15.5
* Xcode 11.5
* Thiết bị test: iPhone SE, iOS 14
* Cấu hình yêu cầu cho mobile:
* Thiết bị là iPhone, yêu cầu iOS 9 trở lên
* Thiết bị có kết nối internet.

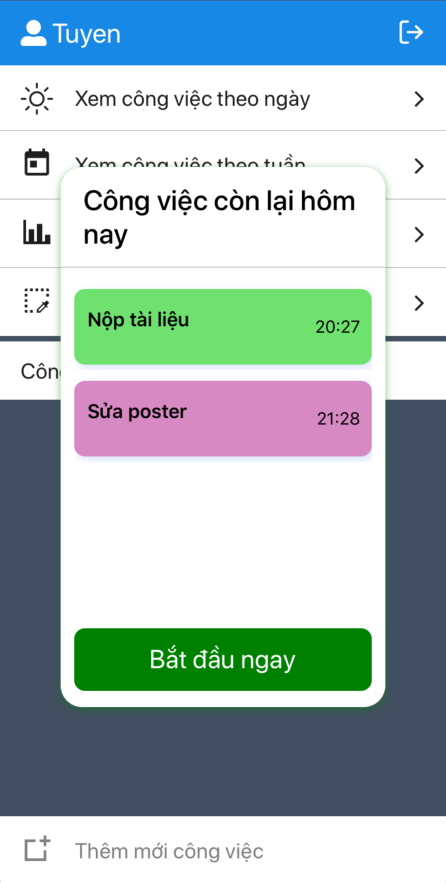
### Giao diện của ứng dụng



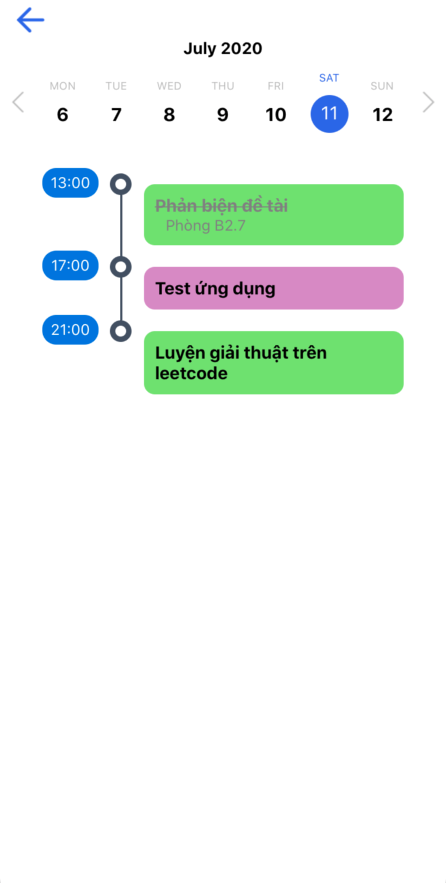
Hình 4-3 Màn hình đăng nhập



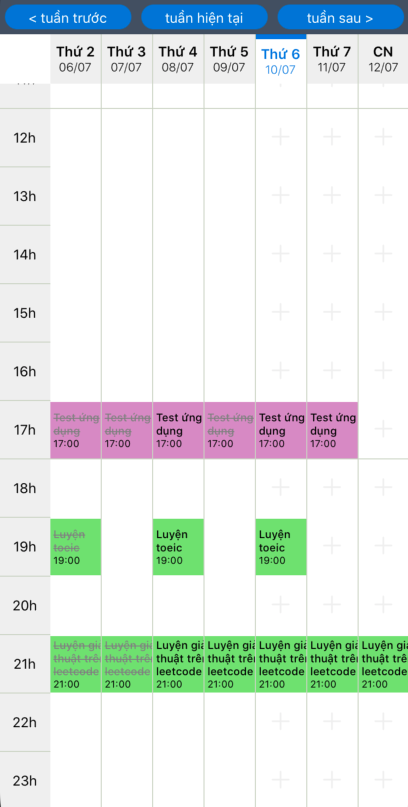
Hình 4-4 Màn hình đăng ký



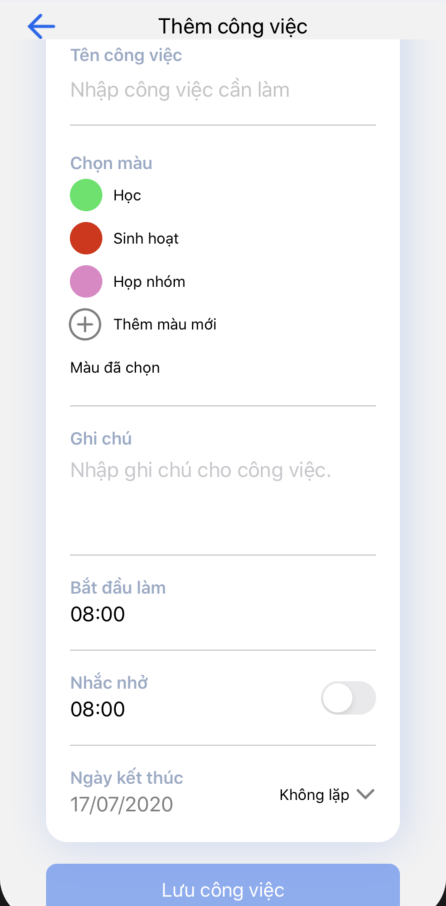
Hình 4-5 Màn hình thông báo công việc còn lại hôm nay



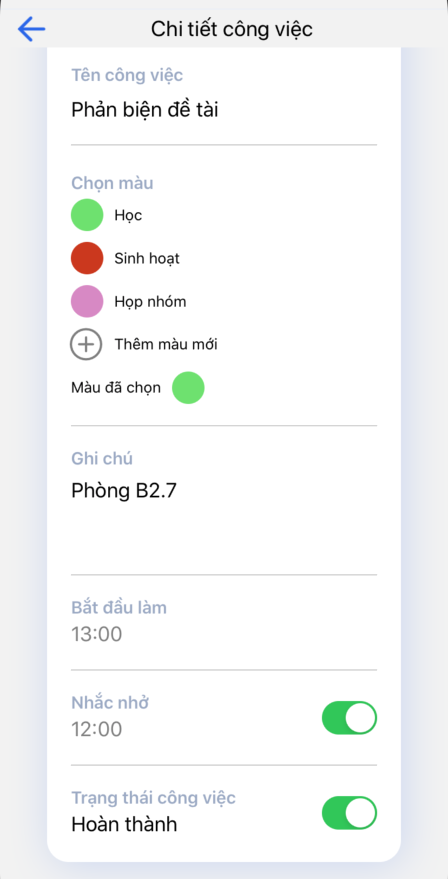
Hình 4-6 Màn hình xem công việc theo ngày



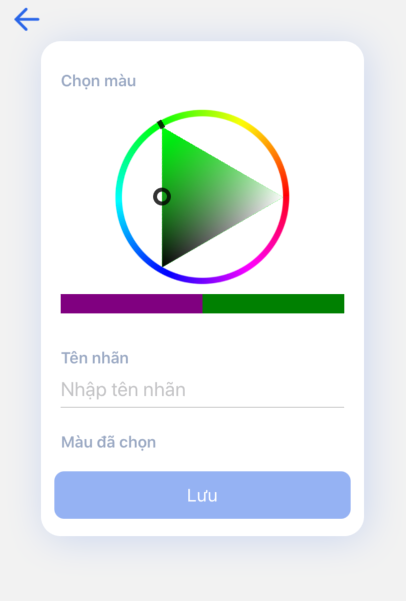
Hình 4-7 Màn hình xem công việc theo tuần



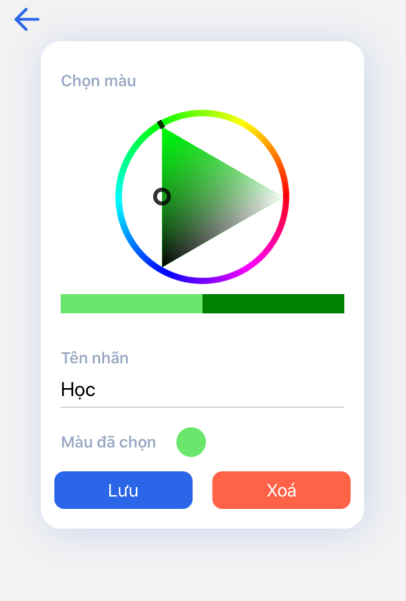
Hình 4-8 Màn hình thêm mới công việc



Hình 4-9 Màn hình chi tiết công việc



Hình 4-10 Màn hình thêm màu nhãn



Hình 4-11 Màn hình xem chi tiết màu nhãn



Hình 4-12 Màn hình thống kê công việc

### Test case

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Chức năng** | **Mô tả** | **Điều kiện trước** | **Dữ liệu Test** | | | | | | | | | | | | | | **Pass/Fail** |
| 1 | Đăng nhập | Đăng nhập vào hệ thống | Tài khoản đăng nhập phải có trong hệ thống | **Input** | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Email | | | | | | | Password | | | | |
| tocongtuyen@gmail.com | | | | | | | 123456 | | | | | Hiện giao diện trang chủ | Hiện giao diện trang chủ | pass |
| Rỗng | | | | | | | Rỗng | | | | | Thông báo “Email hoặc password không chính xác” | Thông báo “Email hoặc password không chính xác” | pass |
| abc | | | | | | | 123 | | | | | Thông báo “Email hoặc password không chính xác” | Thông báo “Email hoặc password không chính xác” | pass |
| 2 | Đăng ký | Đăng ký tài khoản |  | **Input** | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Tên | | | Email | | | | | | password | | |
| Tuyen | | | [tocongtuyen@gmail.com](mailto:tocongtuyen@gmail.com) | | | | | | 123456 | | | Hiện giao diện đăng nhập | Hiện giao diện đăng nhập | pass |
| Tuyen | | | Rỗng | | | | | | Rỗng | | | Thông báo “Nhập thông tin email, password” | Thông báo “Nhập thông tin email, password” | pass |
| Tuyen | | | [tocongtuyen@gmail.com](mailto:tocongtuyen@gmail.com) | | | | | | 123456 | | | Thông báo “Địa chỉ email đã được sử dụng bởi một tài khoản khác” | Thông báo “Địa chỉ email đã được sử dụng bởi một tài khoản khác” | pass |
| 3 | Đăng xuất |  | Đã đăng nhập vào hệ thống |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện giao diện trang đăng nhập | Hiện giao diện trang đăng nhập | pass |
| 4 | Xem công việc theo ngày | Load dữ liệu công việc theo ngày được chọn | Đã đăng nhập vào hệ thống, ở giao diện chính |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện giao diện công việc theo ngày | Hiện giao diện công việc theo ngày | pass |
| 5 | Xem chi tiết công việc | Hiện thông tin chi tiết công việc | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo ngày hoặc theo tuần |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Thông tin chi tiết công việc được hiển thị | Thông tin chi tiết công việc được hiển thị | pass |
| 6 | Xem công viêc theo tuần | Load dữ liệu công việc theo tuần | Đã đăng nhập vào hệ thống, ở giao diện chính |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Thông tin công việc theo tuần được hiển thị | Thông tin công việc theo tuần được hiển thị | pass |
| 7 | Xoá công việc | Chọn công việc cần xoá | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo tuần | Công việc: luyện toeic,  Ngày 26/6/2020 | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Thông tin công việc được xoá khỏi database | Thông tin công việc được xoá khỏi database | pass |
| 8 | Cập nhật thông tin trạng thái công việc | Vuốt sang phải ở công việc cần cập nhật trạng thái | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo tuần | Công việc: báo cáo  Ngày 4/7/2020 | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Thông tin được lưu lại trong hệ thống | Thông tin được lưu lại trong hệ thống | pass |
| 9 | Xem công việc tuần trước | Nhấn vào nút tuần trước | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo tuần |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện dữ liệu của tuần trước | Hiện dữ liệu của tuần trước | pass |
| 10 | Xem công việc tuần hiện tại | Nhấn vào nút tuần hiện tại | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo tuần |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện dữ liệu của tuần hiện tại | Hiện dữ liệu của tuần hiện tại | pass |
| 11 | Xem công việc tuần sau | Nhấn vào nút tuần sau | Đã đăng nhập vào hệ thống, ở giao diện xem công việc theo tuần |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện dữ liệu của tuần sau | Hiện dữ liệu của tuần sau | pass |
| 12 | Thống kê công việc |  | Đã đăng nhập vào hệ thống, ở giao diện chính |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện thị giao diện thống kê công việc và dữ liệu của tuần hiện tại | Hiện thị giao diện thống kê công việc và dữ liệu của tuần hiện tại | pass |
| 13 | Xem thống kê công việc tuần hiện tại |  | Đã đăng nhập vào hệ thống, ở giao diện thống kê |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện dữ liệu thống kê của tuần hiện tại | Hiện dữ liệu thống kê của tuần hiện tại | pass |
| 14 | Xem thống kê công việc bốn tuần trước |  | Đã đăng nhập vào hệ thống, ở giao diện thống kê |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện dữ liệu thống kê của bốn tuần trước | Hiện dữ liệu thống kê của bốn tuần trước | pass |
| 15 | Thêm công việc | Nhập thông tin công việc và nhấn nút lưu | Đã đăng nhập vào hệ thống, ở giao diện chính | **Input** | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Ngày bắt đầu | Tên công việc | Màu | | Ghi chú | | Giờ bắt đầu | | Giờ nhắc nhở | | Không lặp | Thời gian kết thúc |
| 3/7/2020 | báo cáo | màu hồng | | chuẩn bị tài liệu | | 9h15 | | 8h15 | | Không lặp | 3/7/2020 | Dữ liệu được lưu lại | Dữ liệu được lưu lại | pass | |
| 3/7/2020 | rỗng | màu hồng | | chuẩn bị tài liệu | | 9h15 | | 8h15 | | Không lặp | 3/7/2020 | Nút lưu ẩn | Nút lưu ẩn | pass | |
| 3/7/2020 | báo cáo |  | | chuẩn bị tài liệu | | 9h15 | | 8h15 | | Không lặp | 3/7/2020 | Thông báo “Hãy chọn một màu” | Thông báo “Hãy chọn một màu” | pass | |
| 3/7/2020 | báo cáo | màu hồng | | chuẩn bị tài liệu | | 6h | | 6h | | Không lặp | 3/7/2020 | Thông báo “Giờ không được trước giờ hiện tại” | Thông báo “Giờ không được trước giờ hiện tại” | pass | |
| 3/7/2020 | báo cáo | màu hồng | | chuẩn bị tài liệu | | 1h | | 1h | | Không lặp | 3/7/2020 | Thông báo “Thời gian bắt đầu làm phải từ 6-23h59” | Thông báo “Thời gian bắt đầu làm phải từ 6-23h59” | pass | |
| 3/7/2020 | báo cáo | màu hồng | | chuẩn bị tài liệu | | 9h15 | | 9h20 | | Không lặp | 3/7/2020 | Thông báo “Thời gian nhắc nhở không được sau giờ bắt đầu” | Thông báo “Thời gian nhắc nhở không được sau giờ bắt đầu” | pass | |
| 3/7/2020 | báo cáo | màu hồng | | chuẩn bị tài liệu | | 9h15 | | 8h15 | | Không lặp | 2/7/2020 | Thông báo “Ngày kết thúc không được nhỏ hơn ngày bắt đầu” | Thông báo “Ngày kết thúc không được nhỏ hơn ngày bắt đầu” | pass | |
| 16 | Thêm màu nhãn | Nhập thông tin màu nhãn và nhấn nút lưu | Đã đăng nhập vào hệ thống, ở giao diện thêm công việc | **Input** | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Tên nhãn màu | | | | | Chọn màu | | | | | | |
| học | | | | | màu xanh | | | | | | | Thông tin màu nhãn được lưu lại trong hệ thống | Thông tin màu nhãn được lưu lại trong hệ thống | pass |
| Rỗng | | | | | màu xanh | | | | | | | Nút lưu ẩn | Nút lưu ẩn | pass |
| học | | | | | không chọn | | | | | | | Thông báo “Hãy chọn một màu” | Thông báo “Hãy chọn một màu” | pass |
| 17 | Xem chi tiết màu nhãn | Nhấn vào tên màu để xem chi tiết màu | Đã đăng nhập vào hệ thống, ở giao diện thêm công việc |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện chi tiết màu nhãn đã chọn | Hiện chi tiết màu nhãn đã chọn | pass |
| 18 | Xem công việc còn lại hôm nay | Chọn vào chức năng xem công việc còn lại hôm nay | Đã đăng nhập vào hệ thống, ở giao diện chính |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Hiện thông tin các công việc chưa của ngày hôm nay | Hiện thông tin các công việc chưa của ngày hôm nay | pass |
| 19 | Cập nhật trạng thái công việc | Người dùng vuốt sang phải ở công việc muốn cập nhật trạng thái | Đã đăng nhập vào hệ thống, ở giao diện chính |  | | | | | | | | | | | | **Kết quả mong muốn** | **Kết quả thực tế** |  |
| Tên công việc: báo cáo | | | | | | | | | | | | Trạng thái công việc được cập nhật thành công | Trạng thái công việc được cập nhật thành công | pass |
| Tên công việc: luyện toeic | | | | | | | | | | | | Thông báo “công việc chưa tới không thể cập nhật” | Thông báo “công việc chưa tới không thể cập nhật” | pass |

Bảng 4-1 Bảng test case

# : KẾT LUẬN

## Kết quả đạt được

Đăng nhập, đăng ký tài khoản.

Xem công việc theo ngày.

Xem công việc theo tuần.

Xem công việc còn lại hôm nay

Xem thống kê công việc.

Quản lý (thêm, sửa, xóa) công việc, nhãn công việc.

## Hạn chế của đồ án

Code chưa tối ưu, chức năng còn ít, chưa chạy được trên Android.

## Hướng phát triển

Tối ưu code, mở rộng thêm chức năng, ứng dụng chạy trên Android.

# TÀI LIỆU THAM KHẢO

Các tài liệu Tiếng Việt

1. Tài liệu, giáo trình các môn đã học : Phân tích thiết kế hệ thống, phát triển ứng dụng, công nghệ mới.

Các tài liệu từ Internet

1. Website: <https://reactnative.dev>
2. Website: <https://rnfirebase.io/firestore/usage>
3. Website: <https://reactnavigation.org/>

# PHỤ LỤC

**TEST PLAN**

ỨNG DỤNG QUẢN LÝ THỜI GIAN VÀ CÔNG VIỆC TRÊN THIẾT BỊ DI ĐỘNG

1. **Giới thiệu**
   1. **Mục tiêu**

* Lập kế hoạch kiểm thử, đo lường các rủi ro có thể xảy ra.
* Đảm bảo giảm thiểu lỗi ứng dụng.
* Hỗ trợ việc kiểm thử theo đúng kế hoạch, dễ dàng.
  1. **Phạm vi kiểm thử**
* Theo yêu cầu ứng dụng, đề tài chỉ tập trung vào kiểm thử tất cả các chức năng và giao diện ứng dụng.
* Không thực hiện kiểm thử mức độ hiệu suất ứng dụng, logic cơ sở dữ liệu.
  1. **Các ràng buộc về quy trình kiểm thử**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tài nguyên** | **Mô tả** |
| 1 | Các dịch vụ | Firebase Authentication, Cloud Firestore |
| 2 | Công cụ | Kiểm thử thủ công |
| 3 | Mạng | Mạng LAN hoặc Internet |
| 4 | Thiết bị test | Thiết bị có hỗ trợ internet.  Máy ảo Simulator iPhone 11 (iOS 13.5).  Máy thật iPhone SE (iOS 14) |
| 5 | Các tài nguyên khác | Tài khoản Firebase đăng nhập bằng tài khoản Google |

Bảng 22. Môi trường kiểm thử

1. **Yêu cầu kiểm thử**
   1. **Danh sách các chức năng kiểm thử**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả vắn tắt chức năng (outline)** | **Ước lượng số lượng tình huống kiểm kiểm thử (test case).** | **Ghi chú** |
| 1 | Đăng ký tài khoản | Đăng ký tài khoản thông tin gồm tên, email, password | 3 |  |
| 2 | Đăng nhập | Đăng nhập thông tin gồm email, password | 3 |  |
| 3 | Thêm công việc | Thông tin gồm ngày bắt đầu, tên công việc, màu nhãn, ghi chú, giờ bắt đầu, giờ nhắc nhở, trạng thái lặp, ngày kết thúc lặp. | 8 |  |
| 4 | Cập nhật trạng thái công việc | Vuốt sang phải để cập nhật trạng thái công việc | 2 |  |
| 5 | Thêm nhãn công việc | Thông tin gồm tên nhãn, màu. | 3 |  |

Bảng 23. Danh sách các chức năng kiểm thử

* 1. **Điều kiện chấp nhận**
* Phạm vi bao phủ của Test cần đạt 70%
* Số lượng lỗi cần < 3

1. **Kỹ thuật kiểm thử** 
   1. **Kiểm thử đơn vị**

Thành viên thực hiện đơn vị nào thì kiểm thử đơn vị đó.

* 1. **Kiểm thử Module/chức năng**

|  |  |
| --- | --- |
| **Mục tiêu** | Đảm bảo chức năng chạy đúng với mục tiêu thiết kế ban đầu |
| **Kỹ thuật** | Thực thi mỗi use case, luồng hoạt động cho use case, hoặc chức năng, dùng dữ liệu đúng, dữ liệu không đúng để xác định:  -    Kết quả mong muốn khi dữ liệu đưa vào là đúng.  -    Các thông báo lỗi, cảnh báo hiển thị khi dữ liệu không chính xác đưa vào.  -    Quy tắc nghiệp vụ áp dụng cho trường hợp test. |
| **Tiêu chí hoàn thành chức năng** | -   Tất cả các kế hoạch kiểm thử cho chức năng được thực hiện  -   Chỉnh sửa các lỗi đã phát hiện. |
| **Điều kiện đặc biệt.** | Xác định hoặc mô tả các mục hoặc vấn đề (bên trong hoặc bên ngoài) mà có ảnh hưởng đến việc triển khai và thực hiện kiểm thử chức năng. |

Bảng 24. Kiểm thử chức năng

* 1. **Kiểm thử giao diện**

|  |  |
| --- | --- |
| **Mục tiêu** | Kiểm thử các vấn đề:   * Các cử chỉ thao tác nhanh như vuốt trái, phải, lên, xuống. * Giao diện, kích cỡ, vị trí, trạng thái. |
| **Kỹ thuật** | Tạo hoặc sửa đổi các kiểm thử cho mỗi giao diện để xác minh điều hướng và trạng thái đối tượng thích hợp cho từng giao diện ứng dụng |
| **Tiêu chí hoàn thành chức năng** | Giao diện thiết kế rõ ràng và dễ sử dụng. |
| **Điều kiện đặc biệt** | Khi xảy ra lỗi, cần hiện thông báo lỗi cụ thể. |

Bảng 25. Kiểm thử giao diện ứng dụng

* 1. **Kiểm thử hệ thống**

Kiểm thử hệ thống thuộc loại Black Box Testing

* Kiểm thử hệ thống được thực hiện khi hệ thống đã được tích hợp đầy đủ các chức năng bao gồm cả các thiết bị bên ngoài, kiểm thử các thành phần tương tác với nhau và với toàn bộ hệ thống.
* Tạo kịch bản kiểm thử cuối cùng. Kiểm thử mọi đầu vào và đầu ra mong muốn.
* Kiểm thử trải nghiệm của người sử dụng với ứng dụng*.*

1. **Kế hoạch nguồn nhân lực thực hiện kiểm thử cho ứng dụng**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thành viên** | **Vai trò** | **Nhiệm vụ** | **Ước lượng thời gian thực hiện** |
| 1 | Tô Công Tuyển | Lập trình và kiểm thử | * Quản lý toàn bộ việc kiểm thử ứng dụng. Xác định tài nguyên phù hợp cho việc kiểm thử. * Code và Test các chức năng của ứng dụng | Kiểm thử trong một tuần |

Bảng 26. Kế hoạch nhân sự

**KẾ HOẠCH VÀ NHẬT KÝ THỰC HIỆN**

1. **KẾ HOẠCH THỰC HIỆN**

| **Thời gian** | **Nội dung** |
| --- | --- |
| Tuần 01 09/03/2020 – 15/03/2020 | * Tìm kiểu và xác định nội dung của môn học cũng như những kiến thức mới cần tìm hiểu để làm đề tài. * Chọn ngôn ngữ lập trình, xác định cơ sở dữ liệu. * Đặt mục tiêu chung sau khi kết thúc môn học. * Tìm kiểu và xác định nội dung của môn học cũng như những kiến thức mới cần cho dự án. * Hoàn thành menu các chương cần thực hiện của đề tài. |
| Tuần 02 16/03/2020 – 22/03/2020 | * Phân tích sơ bộ các nghiệp vụ, chức năng cần giải quyết, các ràng buộc dữ liệu bằng cách đặt các câu hỏi liên quan. * Tìm hiểu các mô hình đề tài tương tự đã được sử dụng trong thực tế để áp dụng cho đề tài của nhóm. * Phân tích sơ lược các chức năng của đề tài. * Hoàn thành nội dung cơ bản trong menu và gửi tài liệu cho thầy duyệt. |
| Tuần 03 23/03/2020 – 29/03/2020 | * Phân tích chi tiết các chức năng, nghiệp vụ cần thực hiện trong đề tài. * Lược bỏ các chức năng không cần thiết, xác định các chức năng phụ, chức năng chính của đề tài. * Chỉnh sửa lý thuyết theo sự định hướng của giáo viên hướng dẫn. * Tìm hiểu các ứng dụng tương tự. |
| Tuần 04 30/03/2020 – 05/04/2020 | * Đặc tả các chức năng cho ứng dụng. * Tìm hiểu các công nghệ, ngôn ngữ và công cụ liên quan để phục vụ cho đề tài. * Xác định danh sách các tác nhân và mô tả hoạt động của các tác nhân đó. |
| Tuần 05. 06/04/2020 – 12/04/2020 | * Vẽ sơ đồ activity diagram cho các chức năng. * Vẽ sequence diagram cho các chức năng. * Tìm hiểu các phương pháp thực hiện và triển khai đề tài. * Phân chia chức năng, công việc giữa các thành viên trong nhóm. |
| Tuần 06 13/04/2020 – 19/04/2020 | * Cập nhật định kỳ file nhật ký thực hiện để đảm bảo đúng tiến độ. * Tiếp tục tìm hiểu các công nghệ, công cụ để thực hiện đề tài. |
| Tuần 07 20/04/2020 – 26/04/2020 | * Xây dựng giao diện ứng dụng. * Thực hiện coding các chức năng cơ bản. * Sửa tài liệu, các mô hình UML cho phù hợp. * Bổ sung các phần còn thiếu. |
| Tuần 08 27/04/2020 – 03/05/2020 | * Xây dựng kế hoạch và kiểm thử hệ thống. * Tiếp tục hoàn thành giao diện ứng dụng. * Tiếp tục coding các chức năng đã hoàn thành giao diện. |
| Tuần 09 04/05/2020 – 10/05/2020 | * Hoàn thành giao diện ứng dụng. * Cập nhật tiến độ đề tài và điều chỉnh cho phù hợp. * Tiếp tục coding. * Nhìn lại các chức năng đã làm, xem xét các vấn đề liên quan đến dự án để củng cố đầu ra của dự án. |
| Tuần 10 11/05/2020 – 17/10/2020 | * Giải quyết các vấn đề phát sinh trong quá trình thực hiện đề tài. * Tiếp tục coding các chức năng. * Kiểm tra các ràng buộc dữ liệu các chức năng đã thực hiện. |
| Tuần 11 18/05/2020 – 24/05/2020 | * Tiếp tục coding và testing. * Thực hiện kiểm thử ứng dụng. * Cập nhật file nhật ký làm việc để kiểm tra tiến độ. * Khắc phục các lỗi phát sinh (nếu có). |
| Tuần 12 25/05/2020 – 31/05/2020 | * Hoàn thành các chức năng cuối cùng. * Chuẩn bị các tài liệu báo cáo (tài liệu, file nhật ký thực hiện). * Kiểm thử toàn bộ hệ thống, kiểm thử các chức năng chính, các chức năng liên quan với nhau. |
| Tuần 13 01/06/2020 – 07/06/2020 | * Hoàn tất đồ án. * Cài đặt ứng dụng lên thiết bị di động. * Kiểm tra, khắc phục các lỗi (nếu có) khi deploy lên server. |
| Tuần 14 08/06/2020 – 14/06/2020 | * Báo cáo với giáo viên hướng dẫn. * Nộp tài liệu và source code. * Tổng kết đồ án. |
| Tuần 15 15/06/2020 – 21/06/2020 | * Báo cáo đồ án. |

Bảng 7 kế hoạch thực hiện đề tài

*Tp. Hồ Chí Minh, ngày... tháng ... năm 2020*

**Giảng viên hướng dẫn**

1. **NHẬT KÝ THỰC HIỆN ĐỀ TÀI**

| **Thời gian** | **Nội dung thực hiện** | **Nhận xét của GVHD** |
| --- | --- | --- |
| Tuần 01 09/03/2020 – 15/03/2020 | * Tìm hiểu nội dung đề tài. * Liên hệ giáo viên hướng dẫn. * Trao đổi với giáo viên hướng dẫn các việc cần làm. * Xác định các công nghệ cần để thực hiện cho dự án. | Kế hoạch thực hiện đề tài (tính từ ngày 9/3/2020, sau này nhớ nhắc cô ký xác nhận)  Yêu cầu chức năng cho ứng dụng: phải ghi rõ yêu cầu, các ràng buộc... càng chi tiết càng tốt, kèm lược đồ Use-case. Các em phải tự tìm hiểu yêu cầu, tự nghĩ ra hoặc tham khảo UD tương tự đã có và mô tả lại bằng lời văn.  Ở mỗi tuần nộp bài các em cần ghi chú vào nhật ký, sau này sẽ nộp kèm đề tài. |
| Tuần 02 16/03/2020 – 22/03/2020 | * Gửi cho giáo viên hướng dẫn cố cục của tài liệu, file kế hoạch thực hiện để được duyệt trước khi làm. * Thực hiện các công việc đã đề ra ở file nhật ký làm việc. |  |
| Tuần 03 23/03/2020 – 29/03/2020 | * Chỉnh sửa các phần của tài liệu và nhật ký làm việc theo yêu cầu của giảng viên. * Viết yêu cầu chức năng ứng dụng. * Tiếp tục thực hiện các công việc đã đề ra trong file nhật ký làm việc. | Các em xem file word, cô có chú ý trong đó.  Các em chưa viết xong Lời mở đầu: Phần Tổng quan tình hình... các em tìm hiểu các nghiên cứu, các ứng dụng tương tự đề tài của các em, ghi lý do tại sao các em lại chọn công nghệ định sử dụng trong UD. Phần Ý nghĩa khoa học chưa thấy trong báo cáo.  Viết lại phần Tổng quan trong chương 1, các em ko được copy hoàn toàn từ nguồn nào đó, phải viết bằng văn của mình, có thể tinh chỉnh lại từ nguồn khác chứ ko dc copy hoàn toàn.  Phần mô tả chức năng, phải viết lại và sắp xếp ý cho hợp lý. Vẫn mắc lỗi như lần trước: không nêu ý tổng quát trước. Quá ít tính năng cho UD. |
| Tuần 04 30/03/2020 – 05/04/2020 | * Tham khảo các ứng quản lý thời gian để thống nhất lại chức năng của dự án. * Tìm hiểu các vấn đề đã xảy ra hoặc đăng phát sinh trong quá trình làm đồ án. * Tiếp tục thực hiện các công việc đã đề ra trong file nhật ký làm việc. |  |
| Tuần 05. 06/04/2020 – 12/04/2020 | * Thảo luận lại các vấn đề gây khó khăn trong quá trình làm việc. * Xác định các chức năng chính, phụ hoặc các chức năng khó, dễ thực hiện của đồ án. |  |
| Tuần 06 13/04/2020 – 19/04/2020 | * Tiếp tục thực hiện các nhiệm vụ đã đề ra trong file nhật ký làm việc. * Lập kế hoạch khi làm việc offline trong mùa dịch. |  |
| Tuần 07 20/04/2020 – 26/04/2020 | * Tiếp tục thực hiện các nhiệm vụ đã đề ra trong file nhật ký làm việc. * Chuẩn bị các công cụ cho việc coding, các công cụ để làm việc online một cách hiệu quả. | Em tiếp tục làm demo cho chương trình rồi trình bày cô xem, càng sớm càng tốt |
| Tuần 08 27/04/2020 – 03/05/2020 | * Coding các chức năng giao diện ứng dụng. * Xây dựng giao diện ứng dụng |  |
| Tuần 09 04/05/2020 – 10/05/2020 | * Báo cáo với giáo viên hướng dẫn | Nhập lại dữ liệu thật, cho ứng dụng, giao diện quá đơn giản |
| Tuần 10 11/05/2020 – 17/10/2020 | * Sửa lại dữ liệu |  |
| Tuần 11 18/05/2020 – 24/05/2020 | * Báo cáo với giáo viên hướng dẫn. | Ứng dụng chưa đạt yêu cầu |
| Tuần 12 25/05/2020 – 31/05/2020 | * Thực hiện các chức năng mà giáo viên hướng dẫn đã đề cập. * Báo cáo tiến độ, công việc đã hoàn thành, chưa hoàn thành với giáo viên hướng dẫn. | Viết lại giao diện cho ứng dụng. |
| Tuần 13 01/06/2020 – 07/06/2020 | * Thực hiện các công việc mà giáo viên hướng dẫn yêu cầu * Viết lại giao diện công việc theo tuần | Sửa lại tài liệu mô tả chức năng. |
| Tuần 14 08/06/2020 – 14/06/2020 | * Làm chức năng thêm nhãn cho công việc, * Làm chức năng xem công việc theo tuần, xem công việc theo ngày. * Làm chức năng cập nhật trạng thái công việc. | Giao diện xem công việc theo tuần khó xem, Làm chức năng thêm nhãn công việc, |
| Tuần 15 15/06/2020 – 21/06/2020 | * Sửa lại giao diện xem công việc theo tuần * Làm chức năng cập nhật trạng thái công việc | Thêm chức năng sửa xoá, công việc theo ngày.  Xoá nhanh công việc. |
| Tuần 16 22/06/2020 –  28/06/2020 | * Làm theo yêu cầu của giáo viên hướng dẫn * Báo cáo với giáo viên hướng dẫn. | Thêm chức năng xem công việc còn lại hôm nay.  Viết yêu cầu cho chức năng thống kê. |
| Tuần 17 29/06/2020 – 05/07/2020 | * Test ứng dụng * Sửa lại tài liệu. * Làm theo yêu cầu của giáo viên hướng dẫn * Báo cáo với giáo viên hướng dẫn. | Thêm chức năng thống kê công việc hoàn thành không hoàn thành, nhãn công việc, theo tuần, tháng.  Dữ liệu chưa thật. |
| Tuần 18 06/07/2020 – 12/07/2020 | * Test ứng dụng * Sửa lại tài liệu. * Báo báo phản biện. | Sửa lại tài liệu và các mô hình.  Chuẩn bị các nội dung cần báo cáo khi ra phản biện trước giáo viên phản biện.  Chuẩn bị các kiến thức liên quan đến quá trình làm dự án để trả lời các câu hỏi liên quan khi ra phản biện. |

Bảng 8 Nhật ký làm việc

*Tp. Hồ Chí Minh, ngày... tháng ... năm 2020*

**Giảng viên hướng dẫn**