**Shape, rectangle

Description automatically generated**



**LỜI CẢM ƠN**

Trước hết, chúng em xin gửi lời cảm ơn chân thành nhất đến Trường Đại học Công nghệ Thông tin - Đại học Quốc gia Thành phố Hồ Chí Minh và Khoa Hệ thống Thông tin đã tạo điều kiện giúp đỡ, hỗ trợ chúng em trong suốt quá trình học tập và thực hiện đồ án môn học này.

Chúng em cũng xin gửi lời cảm ơn sâu sắc đến cô Trần Hồng Nghi, giảng viên bộ môn Phân tích thiết kế hệ thống thông tin, đã tận tâm hướng dẫn, truyền đạt kiến thức quý báu, giải đáp thắc mắc, định hướng và cung cấp tài liệu cần thiết cho chúng em trong suốt quá trình thực hiện đề tài. Nhờ sự giúp đỡ của cô, chúng em đã có thêm nhiều kiến thức và kinh nghiệm quý báu, giúp chúng em hoàn thành đề án một cách tốt nhất.

Cuối cùng, chúng em xin cảm ơn tất cả các bạn trong nhóm đã cùng nhau nỗ lực, cố gắng, chia sẻ công việc và hoàn thành tốt trách nhiệm của cá nhân trong suốt quá trình thực hiện đề án. Sự phối hợp nhịp nhàng của các bạn chính là yếu tố quan trọng giúp chúng em hoàn thành đề án đúng thời hạn và đạt được kết quả tốt.

Mặc dù đã cố gắng hết sức, nhưng chắc chắn trong đề án vẫn còn nhiều thiếu sót. Chúng em mong nhận được sự thông cảm và góp ý của cô và các bạn. Chúng em sẽ tiếp thu những góp ý để hoàn thiện đề án trong tương lai.

Một lần nữa, chúng em xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, tháng 12 năm 2022

Nhóm sinh viên thực hiện

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

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

**MỤC LỤC**

[**MỤC LỤC** 4](#_heading=h.gjdgxs)

[**1.**](#_heading=h.30j0zll) **CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN ĐỀ TÀI 6**

[**1.1. Đặt vấn đề 6**](#_heading=h.1fob9te)

[**1.2. Mục tiêu 6**](#_heading=h.3znysh7)

[**1.3. Phạm vi 6**](#_heading=h.2et92p0)

[**1.4. Khảo sát hiện trạng 7**](#_heading=h.3dy6vkm)

[**2.**](#_heading=h.1t3h5sf) **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 8**

[**2.1.**](#_heading=h.4d34og8) **Khái niệm cơ bản 8**

[2.1.1.](#_heading=h.2s8eyo1) Khái niệm ứng dụng di động 8

[2.1.2.](#_heading=h.3rdcrjn) Hệ điều hành di động 8

[2.1.3.](#_heading=h.lnxbz9) Các loại ứng dụng 9

[**2.2.**](#_heading=h.2jxsxqh) **Các kỹ thuật lập trình cơ bản 11**

[2.2.1.](#_heading=h.z337ya) Ngôn ngữ lập trình 11

[2.2.2.](#_heading=h.3j2qqm3) Công cụ phát triển di động 11

[2.2.3.](#_heading=h.4i7ojhp) Kiến trúc hệ điều hành Android 12

[2.2.4.](#_heading=h.1ci93xb) React Native 13

[2.2.5.](#_heading=h.3whwml4) Kỹ thuật Front-end 14

[2.2.6.](#_heading=h.2bn6wsx) Kỹ thuật Back-end 17

[**2.3.**](#_heading=h.qsh70q) **Thiết kế giao diện người dùng 18**

[2.3.1.](#_heading=h.3as4poj) View 18

[2.3.2.](#_heading=h.1pxezwc) Layout 19

[**2.4.**](#_heading=h.ihv636) **Cơ sở dữ liệu MySQL 21**

[2.4.1.](#_heading=h.32hioqz) Khái niệm 21

[2.4.2.](#_heading=h.41mghml) Tính năng 22

[2.4.3.](#_heading=h.2grqrue) Ứng dụng 22

[**2.5.**](#_heading=h.vx1227) **Công nghệ sử dụng 23**

[**2.6.**](#_heading=h.3fwokq0) **Kiểm thử và triển khai ứng dụng 23**

[**3.**](#_heading=h.1v1yuxt) **CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 24**

[**3.1.**](#_heading=h.4f1mdlm) **Sơ đồ Use-case 24**

[3.1.1.](#_heading=h.2u6wntf) Danh sách các Actor 24

[3.1.2.](#_heading=h.3tbugp1) Danh sách các Use-case 24

[**3.2.**](#_heading=h.nmf14n) **Đặc tả Use-case và sơ đồ hoạt động (Activity Diagram) 25**

[3.2.1.](#_heading=h.1mrcu09) Use-case Đăng nhập 25

[3.2.2.](#_heading=h.2lwamvv) Use-case Nhắn tin 26

[3.2.3.](#_heading=h.3l18frh) Use-case Đổi mật khẩu 27

[3.2.4.](#_heading=h.206ipza) Use-case Quản lý lịch làm việc 28

[3.2.5.](#_heading=h.4k668n3) Use-case Quản lý thông báo 29

[3.2.6.](#_heading=h.2zbgiuw) Use-case Quản lý lương 30

[3.2.7.](#_heading=h.1egqt2p) Use-case Quản lý nhân viên 31

[3.2.8.](#_heading=h.3ygebqi) Use-case Quản lý phòng ban 32

[3.2.9. Use-case Phân nhân viên vào phòng ban 33](#_heading=h.2dlolyb)

[**3.3. Sơ đồ tuần tự (SequenceDiagram) 34**](#_heading=h.sqyw64)

[3.3.1.](#_heading=h.3cqmetx) Use-case Đăng nhập 34

[3.3.2.](#_heading=h.4bvk7pj) Use-case Nhắn tin 35

[3.3.3.](#_heading=h.1664s55) Use-case Đổi mật khẩu 36

[3.3.4.](#_heading=h.25b2l0r) Use-case Quản lý lịch làm việc 37

[3.3.5.](#_heading=h.2iq8gzs) Use-case Quản lý thông báo 41

[3.3.6.](#_heading=h.3hv69ve) Use-case Quản lý lương 42

[3.3.7.](#_heading=h.4h042r0) Use-case Quản lý nhân viên 45

[3.3.8.](#_heading=h.1baon6m) Use-case Quản lý phòng ban 48

[3.3.9.](#_heading=h.2afmg28) Use- case Phân nhân viên vào phòng ban 51

[**3.4.**](#_heading=h.39kk8xu) **Ánh xạ từ sơ đồ lớp thành mô hình quan hệ 52**

[3.4.1.](#_heading=h.1opuj5n) Mô hình quan hệ 52

[3.4.2.](#_heading=h.2nusc19) Tổ chức dữ liệu 53

[3.4.3.](#_heading=h.1302m92) Mô tả tổ chức dữ liệu 53

[**4.**](#_heading=h.2fk6b3p) **CHƯƠNG 4: HIỆN THỰC ĐỀ TÀI 56**

[**4.1.**](#_heading=h.upglbi) **Mô tả UI/UX của giao diện chung 56**

[**4.2.**](#_heading=h.3ep43zb) **Mô tả UI/UX của giao diện Quản lý 60**

[**4.3.**](#_heading=h.1tuee74) **Mô tả UI/UX của giao diện Nhân viên 68**

[**4.4.**](#_heading=h.4du1wux) **Mô tả UI/UX của giao diện Admin 70**

[**5.**](#_heading=h.2szc72q) **CHƯƠNG 5: THỰC NGHIỆM ĐỀ TÀI 76**

[**6.**](#_heading=h.3s49zyc) **CHƯƠNG 6: KẾT LUẬN 77**

[**6.1. Kết quả đạt được 77**](#_heading=h.279ka65)

[**6.2. Khó khăn, hạn chế 78**](#_heading=h.meukdy)

[**6.1.** **Hướng phát triển 78**](#_heading=h.36ei31r)

[**7.**](#_heading=h.1ljsd9k) **BẢNG PHÂN CHIA CÔNG VIỆC 80**

[**TÀI LIỆU THAM KHẢO 81**](#_heading=h.45jfvxd)

# CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN ĐỀ TÀI

## Đặt vấn đề

Trong bối cảnh nền kinh tế hội nhập và cạnh tranh ngày càng gay gắt, việc quản lý nhân viên hiệu quả là một yêu cầu quan trọng đối với các doanh nghiệp, đặc biệt là các doanh nghiệp nhỏ và vừa. Tuy nhiên, việc quản lý nhân viên thủ công bằng các phương pháp truyền thống thường gặp nhiều khó khăn, tốn kém thời gian và công sức.

Ứng dụng quản lý nhân viên là một giải pháp công nghệ hiện đại giúp các doanh nghiệp nhỏ và vừa giải quyết những khó khăn trong việc quản lý nhân viên. Ứng dụng này có thể giúp doanh nghiệp thực hiện các nghiệp vụ quản lý nhân viên một cách tự động, nhanh chóng và hiệu quả, từ đó nâng cao hiệu suất hoạt động của doanh nghiệp.

## Mục tiêu

Ứng dụng quản lý nhân viên được phát triển nhằm mục đích giúp các doanh nghiệp nhỏ và vừa thực hiện một số công việc như: quản lý thông tin nhân viên một cách chính xác và đầy đủ; quản lý chấm công, tính lương một cách minh bạch và công bằng; quản lý hồ sơ nhân viên một cách khoa học và hiệu quả; quản lý công việc một cách khoa học và chặt chẽ; quản lý nhân sự một cách chuyên nghiệp và hiệu quả.

## Phạm vi

Trong thời gian thực hiện và hoàn thành đồ án là 3 tháng, việc xây dựng ứng dụng quản lý nhân viên sẽ tập trung vào 3 đối tượng chính là: admin, Quản lý và nhân viên với các chức năng như sau:

| Đối tượng sử dụng | Chức năng |
| --- | --- |
| Admin | * Quản lý phòng ban * Quản lý thông tin cá nhân * Phân công |
| Quản lý | * Đăng nhập, đổi mật khẩu * Quản lý nhân viên * Quản lý lịch làm việc * Quản lý thông báo * Quản lý lương * Nhắn tin * Xem thông tin cá nhân |
| Nhân viên | * Đăng nhập, đổi mật khẩu * Nhắn tin * Xem lịch làm việc * Xem lương * Xem thông báo * Xem thông tin cá nhân |

*Bảng 1.3.1: Bảng phạm vi đề tài.*

## Khảo sát hiện trạng

Hiện nay, trên thị trường đã xuất hiện nhiều ứng dụng quản lý nhân viên như ứng dụng HRMS của Bizfly, Hronline, Talentsoft, SAP SuccessFactors, Oracle HCM Cloud, … Tuy nhiên, một số nhược điểm của các ứng dụng này là về mặt chi phí, khả năng tùy chỉnh, tính bảo mật, hỗ trợ khách hàng.

Vì vậy, cần có một ứng dụng quản lý nhân viên tốt hơn, giúp người dùng tiết kiệm thời gian, chi phí, đảm bảo khả năng tùy chỉnh, hỗ trợ khách hàng đồng thời đảm bảo tính an toàn và bảo mật thông tin.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Khái niệm cơ bản

### Khái niệm ứng dụng di động

Ứng dụng di động [1] là một phần mềm ứng dụng trên thiết bị di động, còn được gọi tắt là ứng dụng di động, hoặc chỉ ứng dụng, (tiếng Anh: mobile application hoặc mobile app hoặc app) là phần mềm ứng dụng được thiết kế để chạy trên điện thoại thông minh, máy tính bảng và các thiết bị di động khác.

A picture containing text, electronics, black, cellphone

Description automatically generated

*Hình 2.1.1: Các biểu tượng (icon) của các phần mềm ứng dụng*

*trên một điện thoại dùng hệ điều hành Android [1].*

### Hệ điều hành di động

Hệ điều hành di động [2] (tiếng Anh "mobile operating system") là một hệ điều hành dành cho các thiết bị điện thoại di động, máy tính bảng, đồng hồ thông minh, laptop 2 trong 1 (laptop có thể chuyển giữa chế độ máy tính và máy tính bảng), hoặc các thiết bị di động khác. Ngày nay, sự phân biệt giữa hệ điều hành cho máy tính để bàn và hệ điều hành di động càng trở nên mờ dần, khi một số hệ điều hành mới hoặc các phiên bản mới cho phép hỗ trợ cả hai nền tảng di động và cố định.

IOS và Android là hai hệ điều hành di động phổ biến nhất hiện nay. iOS là hệ điều hành của Apple, được cài đặt trên các thiết bị iPhone, iPad và iPod Touch. Trong khi đó, Android là hệ điều hành do Google phát triển, chạy trên nhiều loại thiết bị di động khác nhau từ các nhà sản xuất khác nhau.

Icon

Description automatically generatedShape

Description automatically generated with medium confidence

*Hình 2.1.2: Hệ điều hành trên di động phổ biến hiện nay [2].*

### Các loại ứng dụng

Ứng dụng mobile là các ứng dụng di động cho phép người dùng sử dụng để truy cập vào các nội dung mà họ mong muốn trên các thiết bị như điện thoại di động. Có một số loại ứng dụng [3] sau:

* **Native app**

Được phát triển cho một nền tảng cụ thể, bao gồm iOS, Android, Windows Phone,... Các ứng dụng này được viết bằng các ngôn ngữ lập trình được hỗ trợ bởi nền tảng đó, ví dụ như Swift hoặc Objective-C cho iOS và Java hoặc Kotlin cho Android. Native apps có thể tận dụng tối đa sức mạnh của nền tảng, đồng thời cung cấp trải nghiệm tốt nhất cho người dùng.

Timeline

Description automatically generated

*Hình 2.1.3: Native app.*

* **Web app**

Chạy trên trình duyệt web và được phát triển bằng HTML, CSS và JavaScript. Web apps không cần phải được tải về và cài đặt, và thường được cập nhật nhanh chóng hơn so với native apps. Tuy nhiên, chất lượng trải nghiệm người dùng và khả năng tương tác có thể bị giới hạn so với native apps.

Graphical user interface, application

Description automatically generated

*Hình 2.1.4: Minh họa Web app.*

* **Hybrid app**

Kết hợp các công nghệ web và native. Hybrid apps được phát triển bằng HTML, CSS và JavaScript, sau đó được đóng gói vào một ứng dụng native để chạy trên điện thoại. Hybrid apps thường nhanh hơn và dễ phát triển hơn so với native apps, tuy nhiên, chất lượng trải nghiệm người dùng có thể bị giảm do hiệu năng kém hơn so với native apps.

Graphical user interface

Description automatically generated with medium confidence

*Hình 2.1.5: Ứng dụng Hybrid app.*

## Các kỹ thuật lập trình cơ bản

### Ngôn ngữ lập trình

Ngôn ngữ lập trình và công cụ phát triển ứng dụng di động là một chủ đề rất quan trọng trong lĩnh vực phát triển ứng dụng di động. Hiện nay, có nhiều ngôn ngữ lập trình và công cụ phát triển ứng dụng di động được sử dụng phổ biến, trong đó có:

* **Java**: là một ngôn ngữ lập trình phổ biến và được sử dụng rộng rãi trong việc phát triển ứng dụng di động Android. Java được phát triển bởi Sun Microsystems và hiện nay được quản lý bởi Oracle Corporation. Java được thiết kế để có tính đa nền tảng, có thể chạy trên nhiều hệ điều hành khác nhau mà không cần phải biên dịch lại. Java cũng có các tính năng an toàn và bảo mật, giúp bảo vệ ứng dụng khỏi các cuộc tấn công mạng.
* **Kotlin**: là một ngôn ngữ lập trình mới hơn, được Google giới thiệu vào năm 2017 để thay thế cho Java trong phát triển ứng dụng di động Android. Kotlin là một ngôn ngữ lập trình đa năng, đa nền tảng và hoàn toàn tương thích với Java. Kotlin chủ yếu được sử dụng để phát triển ứng dụng di động trên nền tảng Android, tuy nhiên nó cũng có thể được sử dụng để phát triển ứng dụng máy tính và máy chủ.

### Công cụ phát triển di động

Công cụ phát triển ứng dụng di động (Mobile development tools) là những phần mềm hỗ trợ cho việc phát triển ứng dụng di động trên các nền tảng khác nhau như Android, iOS hay Windows Phone. Trong đó, Android Studio là một công cụ phát triển ứng dụng di động miễn phí, được cung cấp bởi Google, được sử dụng rộng rãi bởi các nhà phát triển trên toàn thế giới để tạo ra các ứng dụng di động chất lượng cao cho nền tảng Android. Nó cung cấp cho người dùng một giao diện thân thiện, hỗ trợ đầy đủ các tính năng phát triển, kiểm tra và triển khai ứng dụng di động. Điều này giúp cho việc phát triển ứng dụng di động trở nên dễ dàng và hiệu quả hơn.

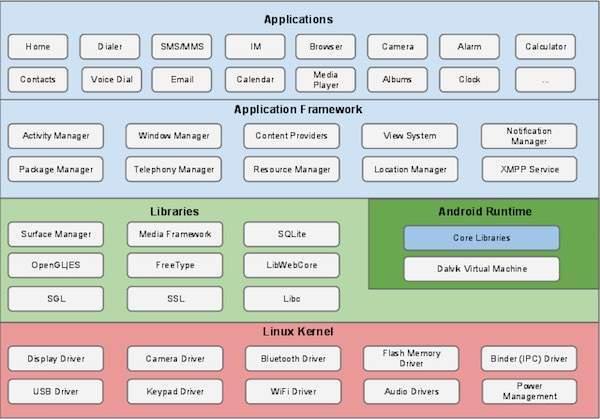
A picture containing graphical user interface

Description automatically generated

*Hình 2.2.1: Hình ảnh minh họa Android Studio.*

### Kiến trúc hệ điều hành Android

Hệ điều hành Android là 1 ngăn xếp các thành phần phần mềm, được chia thành 5 phần và 4 lớp chính như trong hình bên dưới.



*Hình 2.2.2: Kiến trúc của hệ điều hành Android [4].*

* **Linux Kernel**

Dưới cùng là lớp Linux - Linux 3.6 cùng với khoảng 115 bản vá. Lớp này cung cấp 1 cấp độ trừu tượng giữa phần cứng của thiết bị và các thành trình điều khiển phần cứng thiết yếu như máy ảnh, bàn phím, màn hình hiển thị... Đồng thời, hạt nhân (kernel) còn xử lý tất cả các thứ mà Linux có thể làm tốt như mạng kết nối và 1 chuỗi các trình điều khiển thiết bị, giúp cho giao tiếp với các thiết bị ngoại vi dễ dàng hơn.

* **Các thư viện**

Ở trên lớp nhân Linux là tập các thư viện bao gồm WebKit - trình duyệt Web mã nguồn mở, được biết đến như thư viện libc, cơ sở dữ liệu SQLite - hữu dụng cho việc lưu trữ và chia sẻ dữ liệu ứng dụng, các thư viện chơi và ghi âm audio, video, hay các thư viện SSL chịu trách nhiệm bảo mật Internet...

* android.app - Cung cấp quyền truy cập vào ứng dụng và là nền tảng của tất cả ứng dụng Android.
* android.content - Cung cấp quyền truy cập nội dung (content), truyền tải thông điệp giữa các ứng dụng hay các thành phần của ứng dụng.
* android.database - Được sử dụng để truy cập dữ liệu của content provider và cơ sở dữ liệu SQLite
* android.opengl - giao diện các phương thức Java để sử dụng OpenGL ES
* android.os - Cung cấp các ứng dụng với quyền truy cập vào các dịch vụ của hệ điều hành bao gồm thông điệp, các dịch vụ hệ thống và các giao tiếp nội tại
* android.text - Được sử dụng để hiển thị và điều chỉnh chữ trên màn hình thiết bị
* android.view - Các thành phần cơ bản trong việc xây dựng giao diện người dùng của ứng dụng.
* android.widget - Tập các thành phần giao diện người dùng đã được xây dựng sẵn như các nút, các nhãn (label), list view,....
* android.webkit - Tập các lớp cho phép xây dựng khả năng duyệt web.
* **Android Runtime**

Đây là phần thứ 3 của kiến trúc và nằm ở lớp thứ 2 từ dưới lên. Phần này cung cấp 1 bộ phận quan trọng là Dalvik Virtual Machine - là 1 loại Java Virtual Machine được thiết kế đặc biệt để tối ưu cho Android.

* **Application Framework**

Lớp Android Framework cung cấp các dịch vụ cấp độ cao hơn cho các ứng dụng dưới dạng các lớp Java. Các nhà phát triển ứng dụng được phép sử dụng các dịch vụ này trong ứng dụng của họ.

### React Native

React Native là framework xây dựng ứng dụng di động native sử dụng Javascript do Facebook phát hành, cho phép các dev sử dụng JavaScript để làm mobile apps trên cả Android và iOS với cảm nhận và giao diện native. React Native giải quyết được vấn đề về hiệu suất trên Hybrid và tối ưu được chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động.

React Native hoạt động được nhờ tích hợp 2 thread là Main Thread và JS Thread cho ứng dụng mobile. Trong đó:

* Main Thread: cập nhật giao diện người dùng(UI) và xử lý tương tác người dùng.
* JS Thread: thực thi và xử lý code Javascript.

Hai luồng Main Thread và JS Thread này hoạt động độc lập với nhau. Hai Thread sẽ tương tác với nhau nhờ một Bridge (cầu nối). Chiếc cầu này sẽ chuyển đổi dữ liệu qua lại giữa các Thread.

### Kỹ thuật Front-end

* **Axios**

Axios là một HTTP client dựa được phát triển trên đối tượng Javascript Promise, nó có thể sử dụng trong các ứng dụng font-end Vue.js, React, Angular... Sử dụng Axios dễ dàng gửi đi các request HTTP bất đồng bộ đến các REST endpoint và thực hiện các chức năng CRUD. CRUD viết tắt của Create, Read, Update, Delete là một thuật ngữ lập trình nói đến 4 phương thức quen thuộc khi làm việc với kho dữ liệu. Axios là một "trình duyệt" trong Javascript giúp thực hiện thao tác với các website hoặc API giúp xây dựng những ứng dụng front-end linh hoạt mạnh mẽ hơn.

Thư viện axios có thể cài đặt thông qua công cụ npm:



Hoặc chèn đường dẫn file Javascript trên các CDN:

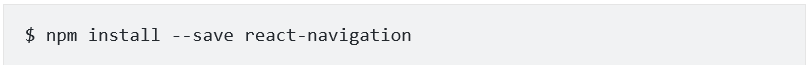
<script src="https://unpkg.com/axios/dist/axios.min.js"></script>

* **NavigationContainer**

NavigationContainer: để chứa toàn bộ các Component khác trong gói Navigation dùng cho việc quản lý chuyển đổi màn hình như tạo tabs, navigation drawer … và để có thể sử dụng được các react-navigation.

React Navigation là một thư viện dựa trên JavaScript để điều hướng màn hình. Nó chính thức được giới thiệu bởi cả Facebook và React Native Document như là giải pháp tối ưu để điều hướng màn hình.

Đầu tiên, cài đặt react-navigation:



Tiếp theo, tạo một file có tên AppNavigator.js:

Graphical user interface, text, application

Description automatically generated

Nội dung hiển thị của ứng dụng bắt đầu từ App.js. Nếu đưa StackNavigator của mình vào App.js, ứng dụng của sẽ được module hóa và clean hơn nhiều.

Sửa file App.js thành như sau:

Graphical user interface, application

Description automatically generated

* **React-navigation/bottom-tabs**

Tạo ra thanh tab ở phía dưới để tương tác qua lại giữa các screen chính. Tùy chọn cơ bản nhất để cung cấp điều hướng theo tab, điều này tạo ra “Một thanh tab đơn giản ở cuối màn hình cho phép chuyển đổi giữa các tuyến đường khác nhau.”.

Sử dụng thư viện này rất đơn giản: đảm bảo rằng đã cài đặt

@react-navigation/native, sau đó cài đặt [@react-navigation/bottom-tabs]

* **React-navigation/material-top-tabs react-native-tab-view**

Gần tương tự như Bottom tab nhưng nó có được thiết kế theo chuẩn Material Design của Google. Muốn sử dụng chúng ta cài thêm thư viện react-navigation-material-bottom-tabs. Nó được sử dụng để tạo ra thanh tab ở phía trên để tương tác qua lại giữa các thể loại ở màn hình bảng xếp hạng sách.



* **React-native-vector-icon**

Để sử dụng bộ icon từ nhiều nguồn khác nhau hỗ trợ cho react-native.

* **React-native-bouncy-checkbox**

Để sử dụng checkbox do checkbox cơ bản của react native đã bị ngừng hỗ trợ.

* **React-native-gesture-handler**

Để sử dụng TextInput và các hiệu ứng chạm như TouchableOpacity, TouchableHightlight...

* **React-native-countdown-component**

Hỗ trợ countdown đếm ngược thời gian ở phần FlashSale

* **React-native/stack**

Công cụ điều trợ tạo điều hướng Ngăn xếp sẽ điều hướng từ màn hình này sang màn hình khác theo kiểu phân cấp

* **React-native/native**

Hỗ trợ tạo điều hướng ứng dụng, chuyển các màn hình

* **React-number-format**

Hỗ trợ chuyển định dạng số sang định dạng tiền tệ ở các sản phẩm

### Kỹ thuật Back-end

* **Express.js framework**
* **Express js** là một Framework nhỏ, nhưng linh hoạt được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile
* **Về các package hỗ trợ**: Expressjs có vô số các package hỗ trợ nên các bạn không phải lo lắng khi làm việc với Framework này.
* **Về performance**: Express cung cấp thêm về các tính năng (feature) để dev lập trình tốt hơn. Chứ không làm giảm tốc độ của NodeJS.
* Và hơn hết, các Framework nổi tiếng của NodeJS hiện nay đều sử dụng ExpressJS như một core function, chẳng hạn: SailsJS, MEAN....
* Được xây dựng theo mô hình MVC (Model – View – Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm.
* Cấu trúc của Express.js:
* Root:
  + app.js chứa các thông tin về cấu hình, khai báo, các định nghĩa,...
  + package.json chứa các package cho ứng dụng chạy.
* Folder routes: chứa các route có trong ứng dụng
* Folder view: chứa view/template cho ứng dụng
* Folder public chứa các file css, js, images,...cho ứng dụng

## Thiết kế giao diện người dùng

### View

* **TextView**

TextView là View dùng để hiển thị văn bản (text) lên màn hình. TextView được định nghĩa bởi thẻ <TextView> trong XML.

A picture containing shape

Description automatically generated

* **Button**

Button là View dùng để nhận 1 các cú chạm (nhấn nút) và thực hiện các chức năng trong phương thức onClick(). Button được định nghĩa bởi thẻ <Button> trong XML.

A picture containing shape

Description automatically generated

* **CheckBox**

Checkbox thường được sử dụng trong các ứng dụng Android để cho phép người dùng thực hiện các tác vụ như chọn hoặc bỏ chọn các tùy chọn.

A picture containing shape

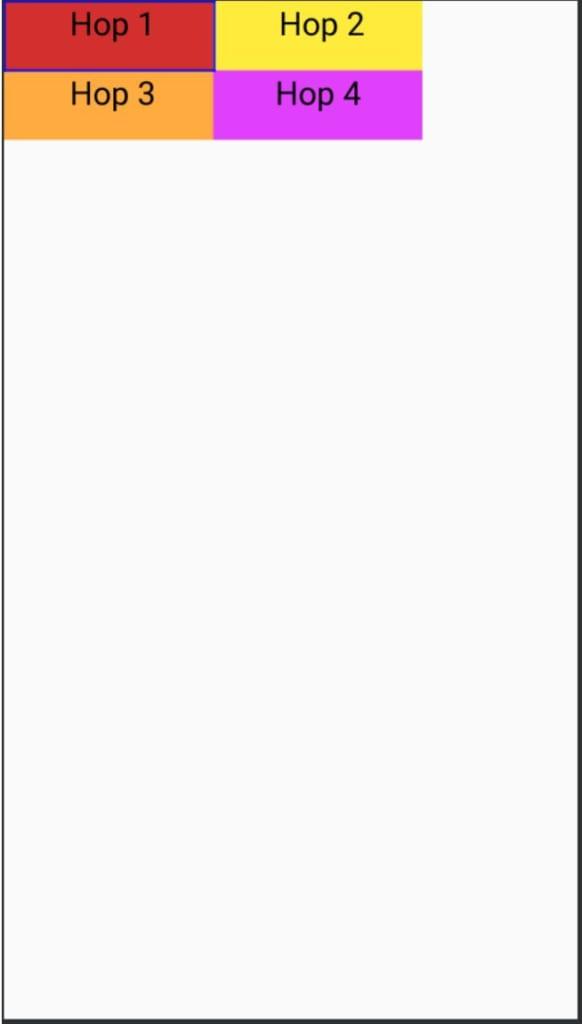
Description automatically generated

### Layout

Layout là thành phần định nghĩa cấu trúc giao diện người dùng hay nói cách khác là thành phần quyết định đến giao diện của một màn hình trong ứng dụng Android. Layout hỗ trợ việc căn chỉnh các widget (Ví dụ: TextView, Button, hay EditText…) như chúng ta thấy trong các ứng dụng Android.

* **RelativeLayout**

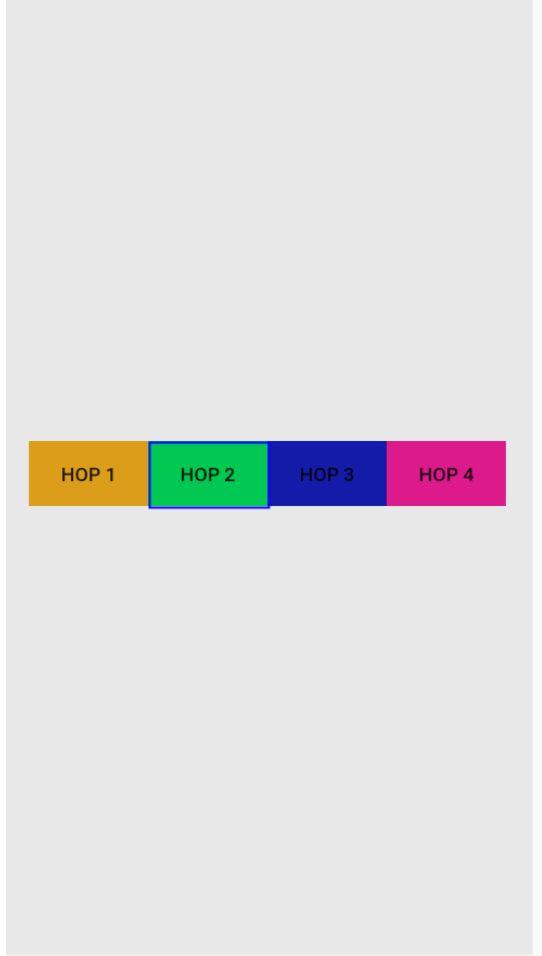
RelativeLayout là một ViewGroup có hiển thị các View con ở các vị trí tương đối. Vị trí của mỗi View có thể được quy định liên quan đến các View anh em (như bên trái của hoặc bên dưới một View khác) hoặc ở các vị trí tương đối với khu vực cha RelativeLayout (chẳng hạn như sắp xếp ngay phía dưới, bên trái hoặc trung tâm).



*Hình 2.2.3: Minh họa RelativeLayout [5].*

* **LinearLayout**

LinearLayout là loại layout sẽ sắp xếp các view theo chiều dọc hoặc ngang theo thứ tự của các view. Đây là ViewGroup sẽ giúp người dùng sắp xếp các view con chứa bên trong theo dạng hàng ngang hoặc hàng dọc với nhau.



*Hình 2.2.4: Minh họa Linear Layout [5].*

* **Frame Layout**

Framelayout là dạng layout cơ bản nhất khi gắn các view lên layout này thì nó sẽ luôn giữ các view này ở phía góc trái màn hình và không cho chúng ta thay đổi vị trí của chúng, các view đưa vào sau sẽ đè lên view ở trước trừ khi người dùng thiết lập transparent cho view sau đó.

A picture containing chart

Description automatically generated

*Hình 2.2.5: Minh họa Frame Layout [6].*

* **TableLayout**

Với TableLayout nó sẽ sắp xếp các View con bên trong thành dạng bảng. Mỗi hàng là một đối tượng view TableRow bên trong TableRow chứa các View con, mỗi View con này nằm ở vị trí một ô bảng (cell). Cột / hàng trong bảng bắt đầu từ số 0.

Graphical user interface

Description automatically generated with medium confidence

*Hình 2.2.6: Minh họa Table Layout [6].*

* **GridLayout**

GridLayout của Androidđơn giản chỉ là 1 layout dạng lưới và chúng ta có thể chia các cột và dòng cho cái lưới đó, các view sẽ được đặt vào các ô trong cái lưới này.

Chart

Description automatically generated

*Hình 2.2.7: Minh họa Grid Layout [6].*

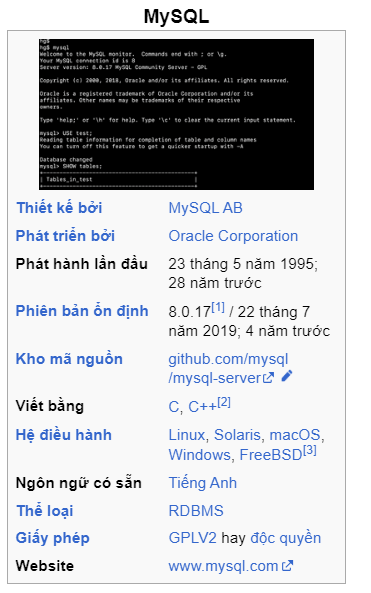
## Cơ sở dữ liệu MYSQL

### Khái niệm

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở, được phát triển bởi Oracle Corporation. MySQL là một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới, được sử dụng bởi các doanh nghiệp, tổ chức và cá nhân trên khắp thế giới.

MySQL lưu trữ dữ liệu trong các bảng, mỗi bảng bao gồm các hàng và cột. Các hàng đại diện cho các bản ghi dữ liệu, trong khi các cột đại diện cho các trường dữ liệu. Các bảng được liên kết với nhau bằng các mối quan hệ, giúp tổ chức dữ liệu một cách hiệu quả.

MySQL cung cấp một bộ các lệnh SQL để truy cập và quản lý dữ liệu trong cơ sở dữ liệu. SQL là một ngôn ngữ chuẩn được sử dụng bởi hầu hết các hệ quản trị cơ sở dữ liệu quan hệ.



*Hình 2.2.8: Giới thiệu MYSQL.*

### Tính năng

* MySQL hỗ trợ giao dịch theo nguyên tắc ACID
* MySQL cung cấp các lệnh SQL để quản lý giao dịch, chẳng hạn như:
  + START TRANSACTION: Bắt đầu một giao dịch.
  + COMMIT: Xác nhận một giao dịch.
  + ROLLBACK: Hủy bỏ một giao dịch.
* MySQL là một hệ quản trị cơ sở dữ liệu sẵn sàng sử dụng. Nó không cần bất kỳ cấu hình nào để bắt đầu sử dụng.
* MySQL cung cấp một tập hợp các tính năng đầy đủ, bao gồm:
  + Hỗ trợ cho các bảng, cột, chỉ mục, ràng buộc, v.v.
  + Hỗ trợ cho các truy vấn SQL, bao gồm các truy vấn SELECT, INSERT, UPDATE và DELETE.
  + Hỗ trợ cho các giao dịch ACID.
  + Hỗ trợ cho các tính năng bảo mật, chẳng hạn như xác thực và mã hóa.
* Một cơ sở dữ liệu MySQL hoàn chỉnh được lưu trữ trong một tệp duy nhất. Tệp cơ sở dữ liệu này có thể được lưu trữ trên bất kỳ hệ thống tệp nào được hỗ trợ bởi MySQL.
* MySQL hỗ trợ các cơ sở dữ liệu có kích thước lên tới hàng terabyte. Nó cũng hỗ trợ các chuỗi có kích thước lên tới hàng gigabyte.
* MySQL cung cấp một API C/C++ đầy đủ chức năng. API này cho phép các nhà phát triển ứng dụng truy cập và quản lý dữ liệu trong cơ sở dữ liệu MySQL.
* MySQL cũng cung cấp các bindings cho nhiều ngôn ngữ lập trình khác, bao gồm Java, Python, PHP, v.v.
* MySQL được thiết kế để cung cấp hiệu suất cao. Trong một số trường hợp, MySQL có thể nhanh hơn hệ thống tệp tin trực tiếp I/O.
* MySQL được viết bằng C/C++. Bindings cho nhiều ngôn ngữ lập trình khác có sẵn một cách riêng biệt.
* MySQL là một dự án mã nguồn mở. Mã nguồn của MySQL có sẵn để tải xuống và sử dụng miễn phí.
* MySQL có sẵn trên nhiều nền tảng, bao gồm Windows, Linux, macOS, Solaris, v.v.
  + 1. **Ứng dụng**
* **Lưu trữ dữ liệu cho các trang web:** MySQL là một lựa chọn phổ biến để lưu trữ dữ liệu cho các trang web. Nó có thể được sử dụng để lưu trữ dữ liệu người dùng, dữ liệu sản phẩm và dữ liệu khác.
* **Lưu trữ dữ liệu cho các ứng dụng doanh nghiệp:** MySQL cũng được sử dụng rộng rãi để lưu trữ dữ liệu cho các ứng dụng doanh nghiệp. Nó có thể được sử dụng để lưu trữ dữ liệu khách hàng, dữ liệu bán hàng và dữ liệu khác.
* **Lưu trữ dữ liệu cho các ứng dụng khoa học:** MySQL cũng có thể được sử dụng để lưu trữ dữ liệu cho các ứng dụng khoa học. Nó có thể được sử dụng để lưu trữ dữ liệu nghiên cứu, dữ liệu thí nghiệm và dữ liệu khác.
  1. **Công nghệ sử dụng**

Ngoài Android Studio và MySQL, ứng dụng còn sử dụng các công nghệ khác như:

**2.5.1. GitHub**

Github là một nền tảng trực tuyến mạnh mẽ để lưu trữ và hợp tác phát triển các dự án phần mềm. Chủ yếu, nó được sử dụng để kiểm soát phiên bản, nghĩa là theo dõi các thay đổi đối với mã theo thời gian, nhưng nó cũng cung cấp một loạt các tính năng khác khiến nó trở thành một công cụ thiết yếu cho các nhà phát triển.

Đây là một số lợi ích khác của việc sử dụng GitHub:

* Chất lượng mã tốt hơn: GitHub khuyến khích đánh giá mã và hợp tác, có thể giúp cải thiện chất lượng mã.
* Năng suất tăng: GitHub có thể giúp tăng năng suất bằng cách làm cho việc hợp tác trên các dự án và theo dõi tiến độ trở nên dễ dàng hơn.
* Khả năng hiển thị tốt hơn: GitHub giúp dễ dàng chia sẻ mã với người khác và nhận phản hồi về công việc của bạn.
* Cộng đồng mạnh mẽ: GitHub có một cộng đồng lớn và tích cực các nhà phát triển sẵn sàng giúp đỡ và chia sẻ kiến thức.

**2.5.2. REST APIs with .NET and C#**

REST APIs là gì?

* REST (REpresentational State Transfer) là một kiến trúc phong cách thiết kế phần mềm cho phép các hệ thống giao tiếp qua mạng một cách linh hoạt và hiệu quả.
* REST APIs là các giao diện lập trình ứng dụng (API) tuân thủ các nguyên tắc REST, cung cấp khả năng truy cập và thao tác dữ liệu thông qua các yêu cầu HTTP đơn giản.

.NET và C# trong REST APIs

* .NET là một nền tảng phát triển phần mềm được phát triển bởi Microsoft, cung cấp nhiều ngôn ngữ lập trình, thư viện và công cụ để xây dựng các ứng dụng web, desktop và di động.
* C# là một ngôn ngữ lập trình hướng đối tượng hiện đại và mạnh mẽ được sử dụng rộng rãi trong nền tảng .NET.
* .NET cung cấp các framework và thư viện hỗ trợ mạnh mẽ cho việc xây dựng REST APIs, bao gồm ASP.NET Core, Web API và WebAPI.

Lợi ích của việc sử dụng .NET và C# cho REST APIs:

* Hỗ trợ mạnh mẽ: .NET cung cấp các framework và thư viện phong phú hỗ trợ xây dựng REST APIs.
* Hiệu suất cao: .NET và C# có hiệu suất cao, giúp tạo ra các API nhanh và mạnh mẽ.
* Khả năng mở rộng: REST APIs có thể được mở rộng dễ dàng để đáp ứng nhu cầu phát triển.
* Dễ sử dụng: .NET và C# dễ học và sử dụng, giúp các nhà phát triển nhanh chóng xây dựng các API chất lượng.
* Khả năng tương tác: REST APIs có thể được sử dụng bởi nhiều nền tảng và ngôn ngữ lập trình khác nhau.

**2.5.3. NGINX Reverse Proxy**

Trình proxy ngược là gì?

* Vị trí: Trình proxy ngược là một máy chủ nằm trước một hoặc nhiều máy chủ web, chặn các yêu cầu của khách hàng đến và chuyển tiếp chúng đến máy chủ backend thích hợp.
* Mục đích: Nó hoạt động như một trung gian, xử lý các tác vụ như cân bằng tải, bộ nhớ đệm, bảo mật và kết thúc SSL/TLS.

Các trường hợp sử dụng phổ biến:

* Cân bằng tải các ứng dụng web: Phân phối tải trên nhiều máy chủ web để có khả năng mở rộng và phục hồi.
* Bảo vệ các máy chủ backend: Hoạt động như một tường lửa để bảo vệ các máy chủ backend khỏi các cuộc tấn công trực tiếp.
* Kết thúc SSL/TLS: Giảm tải mã hóa/giải mã SSL/TLS khỏi các máy chủ backend để cải thiện hiệu suất.
* Bộ nhớ đệm nội dung tĩnh: Lưu trữ các tài sản tĩnh như hình ảnh, CSS và JavaScript để giảm tải máy chủ và tăng tốc tải trang.
* Mạng phân phối nội dung (CDNs): Phân phối nội dung trên nhiều vị trí địa lý để cải thiện hiệu suất toàn cầu.

**2.5.4. Cloudflare proxy**

Cloudflare Proxy là một dịch vụ proxy ngược hoạt động như một trung gian giữa khách truy cập trang web và máy chủ gốc của trang web đó. Nó cung cấp nhiều tính năng để cải thiện hiệu suất, bảo mật và khả năng mở rộng của các trang web.

Lợi ích chính của việc sử dụng Cloudflare Proxy:

* Tăng tốc độ trang web: Bằng cách lưu nội dung tĩnh trong bộ nhớ đệm và tối ưu hóa nội dung động, Cloudflare có thể giảm thời gian tải trang đáng kể.
* Bảo vệ chống lại các cuộc tấn công: Cloudflare có thể bảo vệ trang web khỏi các cuộc tấn công DDoS, bot độc hại, SQL injection và các mối đe dọa khác.
* Tăng tính khả dụng: Cloudflare có mạng lưới máy chủ toàn cầu rộng lớn, giúp đảm bảo trang web luôn hoạt động ngay cả khi máy chủ gốc gặp sự cố.
* Giảm tải máy chủ: Bằng cách lưu nội dung trong bộ nhớ đệm và lọc lưu lượng truy cập, Cloudflare có thể giảm tải trên máy chủ gốc của trang web, giúp tiết kiệm chi phí và cải thiện hiệu suất.
* Cải thiện SEO: Tốc độ trang web và thời gian hoạt động là các yếu tố xếp hạng quan trọng của công cụ tìm kiếm. Cloudflare có thể giúp cải thiện thứ hạng SEO của trang web.
  1. **Kiểm thử và triển khai ứng dụng**

Các kỹ thuật kiểm thử và triển khai ứng dụng trên di động là những quy trình và công cụ được sử dụng để đảm bảo tính ổn định và chất lượng của ứng dụng trên nhiều thiết bị di động khác nhau. Các kỹ thuật này bao gồm:

* **Kiểm thử đơn vị (Unit testing):** Đây là quy trình kiểm tra các thành phần của ứng dụng một cách riêng lẻ, đảm bảo chúng hoạt động đúng và ổn định.
* **Kiểm thử tích hợp (Integration testing):** Kiểm tra xem các thành phần của ứng dụng có tương tác với nhau một cách đúng đắn hay không.
* **Kiểm thử chấp nhận (Acceptance testing):** Kiểm tra tính năng và hiệu suất của ứng dụng bằng cách sử dụng các kịch bản thử nghiệm.
* **Kiểm thử tải (Load testing):** Kiểm tra khả năng chịu tải của ứng dụng, đảm bảo nó hoạt động ổn định khi có nhiều người dùng truy cập cùng lúc.
* **Kiểm thử giao diện người dùng (User interface testing):** Kiểm tra giao diện người dùng, đảm bảo các thành phần hiển thị đúng và hoạt động một cách trơn tru.

# CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Sơ đồ Use-case

### Danh sách các Actor

| STT | Tác nhân của hệ thống | Ý nghĩa |
| --- | --- | --- |
| 1 | Admin | Người được sử dụng tất cả các chức năng của hệ thống. |
| 2 | Quản lý | Người phụ trách các chức năng về quản lý nhân viên |
| 3 | Nhân viên | Người dung các chức năng kiểm tra thông tin |

*Bảng 3.1.1.1: Bảng danh sách các Actor trong hệ thống.*

### Danh sách các Use-case

| STT | Tên Usecase | Ý Nghĩa |
| --- | --- | --- |
| 1 | Usecase Đăng Nhập | Cho phép người dùng truy cập vào hệ thống. |
| 2 | Usecase Nhắn tin | Cho phép người dùng có thể tương tác trao đổi với Admin và quản lý. |
| 3 | Usecase Đổi mật khẩu | Cho phép người dùng có thể thay đổi mật khẩu mới để thêm tính bảo mật. |
| 4 | Usecase Quản lý lịch làm việc | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin và xóa lịch làm việc. |
| 5 | Usecase Quản lý thông báo | Cho phép người dùng thực hiện chức năng thêm thông báo. |
| 6 | Usecase Quản lý lương | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin lương nhân viên. |
| 7 | Usecase Quản lý nhân viên | Cho phép người quản lý thực hiện chức năng xem danh sách nhân viên, gọi điện và sao chép email. |
| 8 | Usecase Quản lý phòng ban | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin phòng ban. |
| 9 | Usecase Quản lý thông tin cá nhân | Cho phép người dùng có thể xem và cập nhật thông tin cá nhân của bản than và cấp dưới. |
| 10 | Usecase Phân nhân viên vào phòng ban | Cho phép admin phân công các nhân viên vào phòng ban. |

*Bảng 3.1.2.1:Bảng danh sách các Usecase.*

## Đặc tả Use-case và sơ đồ hoạt động (Activity Diagram)

A diagram of a person's work flow

Description automatically generated

*Hình 3.2.1: Usecase tổng.*

### Use-case Đăng nhập

| Tên | Usecase Đăng nhập |
| --- | --- |
| Tóm tắt | Người dùng sử dụng chức năng để đăng nhập vào ứng dụng |
| Các yêu cầu đặc biệt | Có tài khoản đã đăng ký trước đó |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình đăng nhập thành công và truy cập vào ứng dụng |

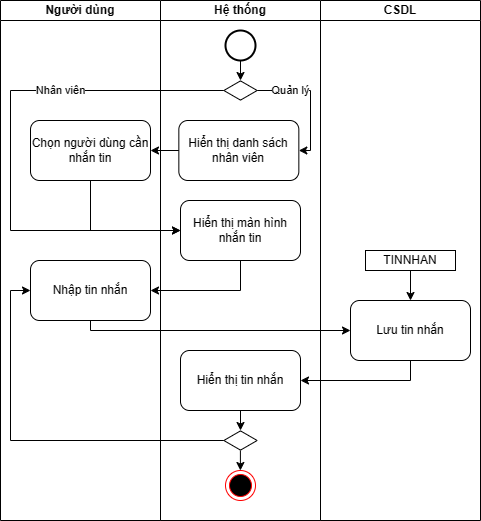
A diagram of a flowchart

Description automatically generated

*Hình 3.2.1.1:Đặc tả Usecase đăng nhập.*

### Use-case Nhắn tin

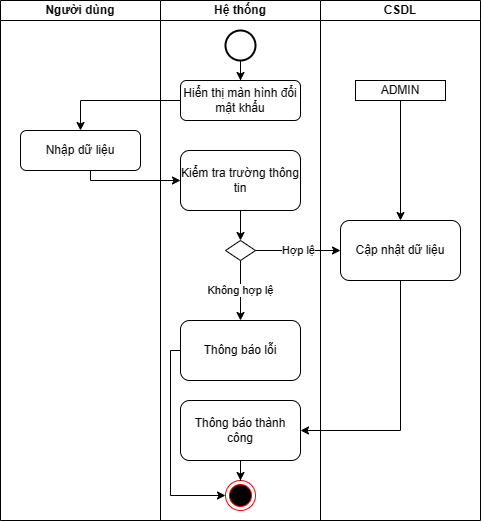
| Tên | Usecase Nhắn tin |
| --- | --- |
| Tóm tắt | Người dùng sử dụng chức năng để liên hệ với admin để trao đổi những thông tin thắc mắc |
| Các yêu cầu đặc biệt | Tài khoản đã đăng nhập |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị thông tin trao đổi |



*Hình 3.2.2.1: Đặc tả Usecase Nhắn tin.*

### Use-case Đổi mật khẩu

| Tên | Usecase Đổi mật khẩu |
| --- | --- |
| Tóm tắt | Cho phép người dùng có thể thay đổi mật khẩu mới để thêm tính bảo mật. |
| Các yêu cầu đặc biệt | Có tài khoản đã đăng ký trước đó |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình cài đặt. |



*Hình 3.2.3.1: Đặc tả Usecase Đổi mật khẩu.*

### Use-case Quản lý lịch làm việc

| Tên | Usecase Quản lý lịch làm việc |
| --- | --- |
| Tóm tắt | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin và xóa lịch làm việc. |
| Các yêu cầu đặc biệt | Đăng nhập vào tài khoản quản lý. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |

A diagram of a flowchart

Description automatically generated

*Hình 3.2.4.1: Đặc tả Usecase Quản lý lịch làm việc.*

### Use-case Quản lý thông báo

| Tên | Usecase Quản lý thông báo |
| --- | --- |
| Tóm tắt | Cho phép người dùng thực hiện chức năng thêm thông báo. |
| Các yêu cầu đặc biệt | Đăng nhập vào tài khoản quản lý. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |

A diagram of a diagram

Description automatically generated

*Hình 3.2.5.1: Đặc tả Usecase Quản lý thông báo.*

### Use-case Quản lý lương

| Tên | Usecase Quản lý lương |
| --- | --- |
| Tóm tắt | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin lương nhân viên. |
| Các yêu cầu đặc biệt | Đăng nhập vào tài khoản quản lý. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |

A diagram of a flowchart

Description automatically generated

*Hình 3.2.6.1: Đặc tả Usecase Quản lý lương.*

### Use-case Quản lý nhân viên

| Tên | Usecase Quản lý nhân viên |
| --- | --- |
| Tóm tắt | Cho phép người quản lý thực hiện chức năng xem danh sách nhân viên, gọi điện và sao chép email. |
| Các yêu cầu đặc biệt | Đăng nhập vào tài khoản quản lý. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |

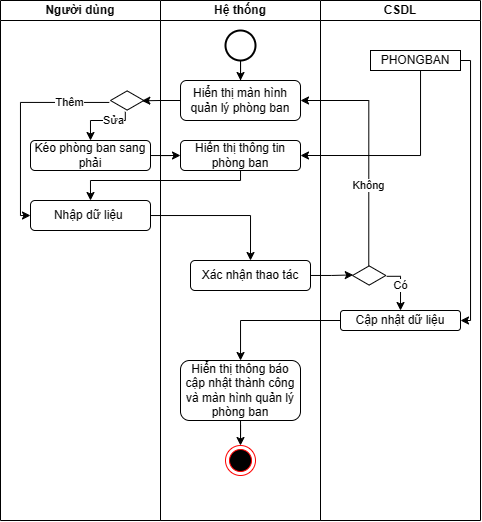
A diagram of a company

Description automatically generated

*Hình 3.2.7.1: Đặc tả Usecase Quản lý nhân viên.*

### Use-case Quản lý phòng ban

| Tên | Usecase Quản lý phòng ban |
| --- | --- |
| Tóm tắt | Cho phép người dùng quản lý các chức năng như thêm mới, cập nhập thông tin phòng ban. |
| Các yêu cầu đặc biệt | Đăng nhập vào tài khoản admin. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |



*Hình 3.2.8.1: Đặc tả Usecase Quản lý phòng ban.*

### Use-case Phân nhân viên vào phòng ban

| Tên | Usecase Quản lý tài khoản |
| --- | --- |
| Tóm tắt | Cho phép admin phân công nhân viên vào các phòng ban. |
| Các yêu cầu đặc biệt | Đăng nhập tài khoản admin. |
| Pre-Conditions | Thiết bị của người dùng phải được kết nối với internet. |
| Post-Conditions | Hiển thị màn hình chức năng nếu thực hiện thành công. |

A diagram of a company

Description automatically generated

*Hình 3.2.9.1: Đặc tả Usecase Phân nhân viên vào phòng ban.*

## Sơ đồ tuần tự

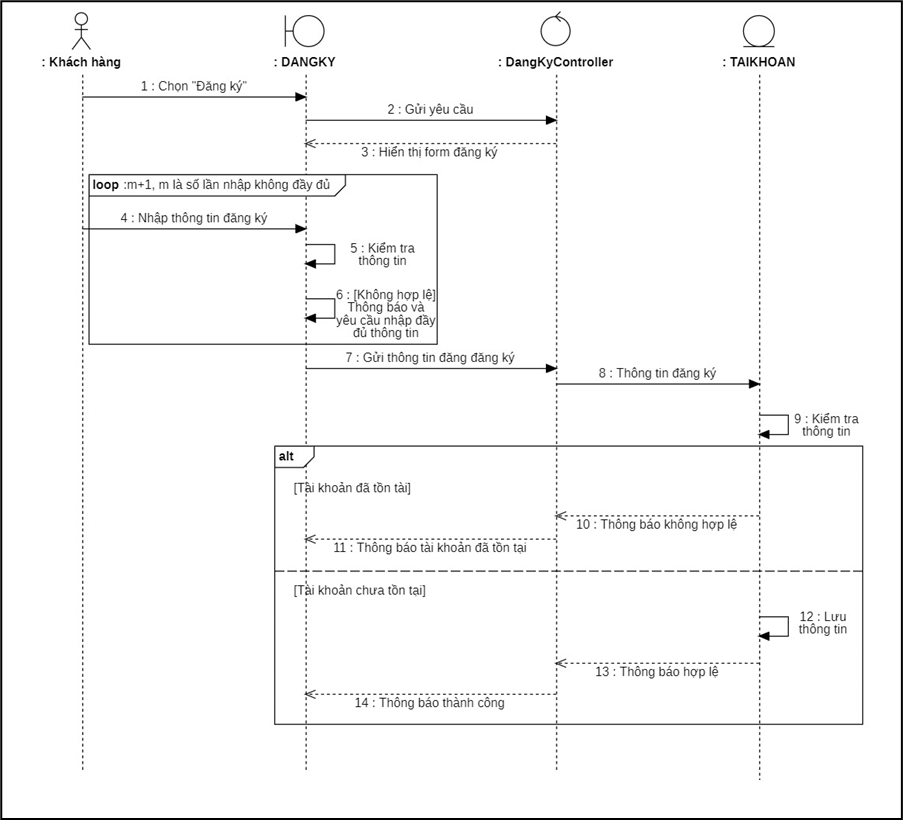
### Use-case Đăng nhập

A diagram of a diagram

Description automatically generated

*Hình 3.3.1: Sequence Đăng nhập.*

### Use-case Nhắn tin



*Hình 3.3.2: Sequence Nhắn tin.*

### Use-case Đổi mật khẩu

A diagram of a diagram

Description automatically generated with medium confidence

*Hình 3.3.3: Sequence Đổi mật khẩu.*

### Use-case Quản lý lịch làm việc

A diagram of a computer program

Description automatically generated

*Hình 3.3.4: Sequen Quản lý lịch làm việc.*

A diagram of a diagram

Description automatically generated

*Hình 3.3.5: Sequence Thêm lịch làm việc.*

A diagram of a diagram

Description automatically generated

*Hình 3.3.6: Sequence Cập nhật lich làm việc.*

A diagram of a diagram

Description automatically generated

*Hình 3.3.7: Sequence Xóa lịch làm việc.*

### Use-case Quản lý thông báo

A diagram of a project

Description automatically generated with medium confidence

*Hình 3.3.8: Sequence Thêm thông báo.*

### Use-case Quản lý lương

A diagram of a computer

Description automatically generated

*Hình 3.3.9: Sequence Quản lý lương.*

*A diagram of a diagram

Description automatically generated with medium confidence*

*Hình 3.3.10: Sequence thêm lương.*

*A diagram of a project

Description automatically generated*

*Hình 3.3.11: Sequence sửa lương.*

### Use-case Quản lý nhân viên

A diagram of a computer

Description automatically generated

*Hình 3.3.12: Sequence Quản lý nhân viên.*

*A diagram of a diagram

Description automatically generated*

*Hình 3.3.13: Sequence sao chép email nhân viên.*

*A diagram of a diagram

Description automatically generated*

*Hình 3.3.14: Sequence gọi nhân viên.*

### Use-case Quản lý phòng ban

A diagram of a computer

Description automatically generated

*Hình 3.3.15: Sequence Quản lý phòng ban.*

*A diagram of a diagram

Description automatically generated with medium confidence*

*Hình 3.3.16: Sequence them phòng ban*

*A diagram of a diagram

Description automatically generated with medium confidence*

*Hình 3.3.17: Sequence sửa phòng ban.*

### Use- case Phân nhân viên vào phòng ban

*A diagram of a project

Description automatically generated*

*Hình 3.3.18: Sequence Phân nhân viên vào phòng ban*

## **Ánh xạ từ sơ đồ lớp thành mô hình quan hệ**

### Mô hình quan hệ

A diagram of a computer

Description automatically generated

*Hình 3.3.19: Mô hình quan hệ.*

### Tổ chức dữ liệu

* HOPDONGLAODONG(MAHD, MANV, LOAIHD, TUNGAY, DENNGAY)
* THONGBAO(ID, NGGUI, TG, PHBAN, NGNHAN, ND, SONGNHAN, SONGXEM)
* VIPHAM(MANV,TGVP, NDVP, PHAT)
* NHANVIEN(MANV, MK, HOTEN, GIOITINH, NGSINH, NGVL, DC, SDT, EMAIL, CCCD, LCB, PHBAN)
* ADMIN(Id, Ten, MK)
* PHONGBAN(MAPH, TRPH, NGNC)
* NGXEMTB(MATB, MANV)
* NGNHANTB(MATB, MANV)
* THAMGIALAMVIEC(MALV, MANV)
* LUONG(MANV, TG, PC, THUONG, TGTT)
* YKIEN(NGGUI, TG, ND)
* GIAYTO(MAGT,NGGUI, TG, LOAIDON, NOIDUNG, TT, PHANHOI, NGD)
* TINNHAN(NGGUI, NGNHAN, TG, LTN, ND, TT)
* LICHLAMVIEC(MALV, TIEUDE, MOTA, NGAYBATDAU, NGAYKETTHUC, PHBAN, SoLuongNhanVien)

### Mô tả tổ chức dữ liệu

#### HOPDONGLAODONG

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MAHD | Mã hợp đồng lao động | int |
| 2 | MANV | Mã nhân viên | int |
| 3 | LOAIHD | Loại hợp đồng | varchar |
| 4 | TUNGAY | Ngày bắt đầu hợp đồng | datetime |
| 5 | DENNGAY | Ngày kết thúc hợp đồng | datetime |

*Mô tả chi tiết thuộc tính bảng HOPDONGLAODONG*

#### THONGBAO

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | ID |  | int |
| 2 | NGGUI | Người gửi thông báo | varchar |
| 3 | TG | Thời gian gửi thông báo | datetime |
| 4 | PHBAN | Phòng ban gửi thông báo | varchar |
| 5 | NGNHAN | Người nhận thông báo | varchar |
| 6 | ND | Nội dung của thông báo | varchar |
| 7 | SONGNHAN |  | int |
| 8 | SONGXEM |  | int |

#### VIPHAM

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MANV | Mã nhân viên | int |
| 2 | TGVP | Thời gian vi phạm | datetime |
| 3 | NDVP | Nội dung vi phạm | varchar |
| 4 | PHAT | Nội dung phạt | varchar |

#### NHANVIEN

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MANV | Mã nhân viên | int |
| 2 | TGVP | Thời gian vi phạm | datetime |
| 3 | NDVP | Nội dung vi phạm | varchar |
| 4 | PHAT | Nội dung phạt | varchar |

#### ADMIN

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | Id | Mã của Admin | varchar |
| 2 | Tên | Tên Admin | varchar |
| 3 | MK | Mật khẩu tài khoản admin | varchar |

#### PHONGBAN

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MAPH | Mã phòng ban | varchar |
| 2 | TGPH | Mã trưởng phòng ban | varchar |
| 3 | NGNC |  | datetime |

#### NGXEMTB

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MATB | Mã thông báo | varchar |
| 2 | MANV | Mã nhân viên đã xem thông báo | varchar |

#### NGNHANTB

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MATB | Mã thông báo | varchar |
| 2 | MANV | Mã nhân viên nhận thông báo | varchar |

#### THAMGIALAMVIEC

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MALV | Mã công việc | varchar |
| 2 | MANV | Mã nhân viên tham gia làm việc | varchar |

#### LUONG

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MANV | Mã nhân viên | varchar |
| 2 | TG | Thời gian nhận lương | datetime |
| 3 | PC |  | decimal |
| 4 | THUONG | Thưởng | decimal |
| 5 | TGTT |  | datetime |

#### YKIEN

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | NGGUI | Mã người gửi | varchar |
| 2 | TG | Thời gian gửi | datetime |
| 3 | ND | Nội dung | varchar |

#### GIAYTO

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MAGT | Mã giấy tờ | int |
| 2 | NGGUI | Người gửi giấy tờ | varchar |
| 3 | TG | Thời gian gửi giấy tờ | datetime |
| 4 | LOAIDON | Loại giấy tờ được gửi | varchar |
| 5 | NOIDUNG | Nội dung giấy tờ | varchar |
| 6 | TT | Trạng thái | varchar |
| 7 | PHANHOI | Nội dung được phản hồi | varchar |
| 8 | NGD | Người duyệt | varchar |

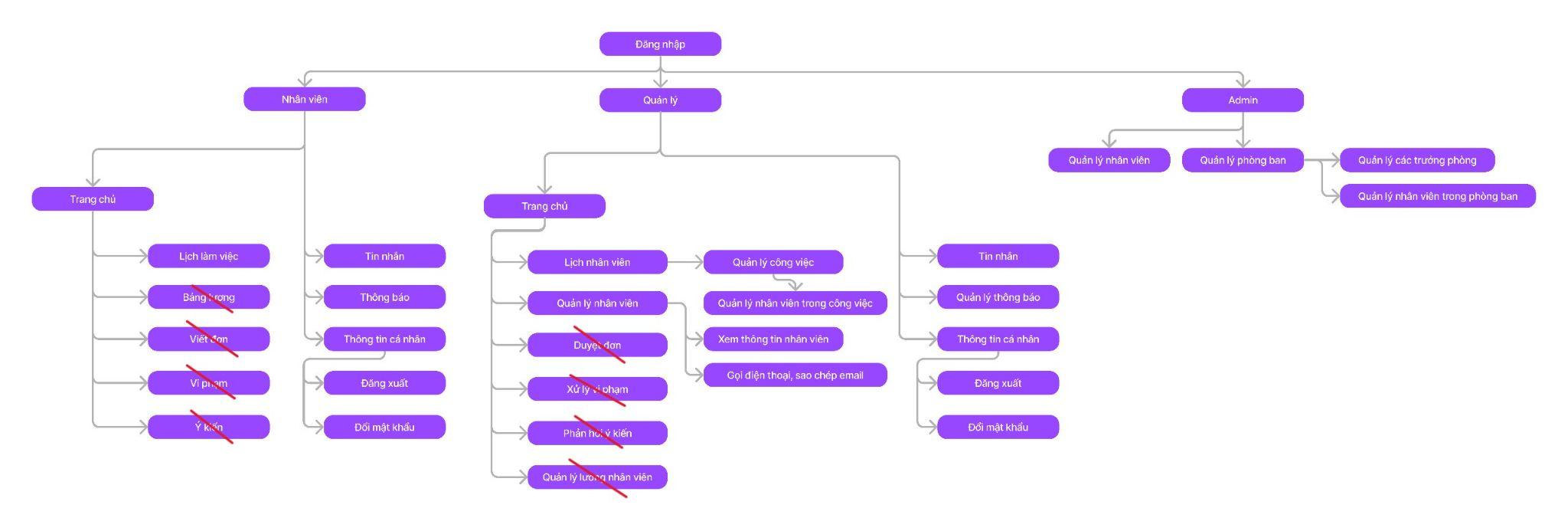
#### TINNHAN

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | NGGUI | Mã người gửi tin nhắn | varchar |
| 2 | NGNHAN | Mã người nhận thông báo | varchar |
| 3 | TG | Thời gian gửi tin nhắn | datetime |
| 4 | LTN | Loại tin nhắn | varchar |
| 5 | ND | Nội dung tin nhắn | varchar |
| 6 | TT | Trạng thái | varchar |

#### LICHLAMVIEC

| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** |
| --- | --- | --- | --- |
| 1 | MALV | Mã công việc | int |
| 2 | TIEUDE | Tên công việc | varchar |
| 3 | MOTA | Mô tả công việc | varchar |
| 4 | NGAYBATDAU | Ngày bắt đầu công việc | datetime |
| 5 | NGAYKETTHUC | Ngày kết thúc công việc | datetime |
| 6 | PHBAN | Phòng ban thực hiện công việc | varchar |
| 7 | SoLuongNhanVien | Số lượng nhân viên tham gia thực hiện công việc | int |

# 3.5. Sơ đồ phân rã chức năng



# 3.6. Các APIs

# 

# CHƯƠNG 4: HIỆN THỰC ĐỀ TÀI

## Mô tả UI/UX của giao diện chung

- Màn hình đăng nhập:

A screen shot of a phone

Description automatically generated

*Hình 4.1.1.: Giao diện đăng nhập*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextInputLayout | til\_username | Dùng để nhập tên người dùng (Username) |
| 2 | TextInputEditText | tie\_username | Trường nhập liệu cho tên người dùng |
| 3 | TextInputLayout | til\_password | Dùng để nhập mật khẩu (Password) |
| 4 | TextInputEditText | tie\_password | Trường nhập liệu cho mật khẩu |
| 5 | CheckBox | chb\_saveLogin | Cung cấp tùy chọn có lưu thông tin đăng nhập hay không |
| 6 | Button | btn\_login | Dùng để đăng nhập |

- Màn hình đặt mật khẩu cho lần đăng nhập đầu tiên:

A screenshot of a black phone

Description automatically generated

*Hình 4.1.2: Giao diện đặt mật khẩu cho lần đăng nhập đầu tiên*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextInputLayout | til\_new\_password | Dùng để nhập mật khẩu mới |
| 2 | TextInputEditText | (trong til\_new\_password) | Trường nhập liệu cho mật khẩu mới |
| 3 | TextInputLayout | til\_renew\_password | Dùng để nhập lại mật khẩu mới |
| 4 | TextInputEditText | (trong til\_renew\_password) | Trường nhập liệu nhập lại mật khẩu mới |
| 5 | Button | btn\_apply | Nút áp dụng để cập nhật mật khẩu mới |

- Màn hình xem thông tin cá nhân:

A screenshot of a phone

Description automatically generated

*Hình 4.1.3.: Giao diện xem thông tin cá nhân*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | ImageView | avatar | Hình ảnh đại diện |
| 2 | TextView | name | Hiển thị tên người dùng |
| 3 | TextView | birth\_day | Hiển thị ngày sinh |
| 4 | TextView | id | Hiển thị mã nhân viên |
| 5 | ImageButton | setting | Nút cài đặt |
| 6 | LinearLayout | (Không có ID) | Chứa các thông tin khác |
| 7 | TextView | gioi\_tinh | Hiển thị giới tính |
| 8 | TextView | cccd | Hiển thị số căn cước công dân |
| 9 | TextView | dia\_chi | Hiển thị địa chỉ |
| 10 | TextView | ngay\_vao\_lam | Hiển thị ngày vào làm |
| 11 | TextView | phong\_ban | Hiển thị thông tin phòng ban |
| 12 | TextView | sdt | Hiển thị số điện thoại |
| 13 | TextView | email | Hiển thị địa chỉ email |

- Màn hình cài đặt:

A close up of a phone

Description automatically generated

*Hình 4.1.4.: Giao diện cài đặt*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | Button | btn\_changepassword | Nút "Đổi mật khẩu" |
| 2 | Button | btn\_signout | Nút "Đăng xuất" |

## Mô tả UI/UX của giao diện Quản lý

- Màn hình chính:

A screenshot of a cell phone

Description automatically generated

*Hình 4.2.1: Giao diện chính*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | Button | btn\_lich\_nhan\_vien | Nút "Lịch nhân viên" |
| 2 | Button | btn\_quan\_ly\_nhan\_vien | Nút "Quản lý nhân viên" |
| 3 | Button | btn\_duyet\_don | Nút "Duyệt đơn" |
| 4 | Button | btn\_xu\_ly\_vi\_pham | Nút "Xử lý vi phạm" |
| 5 | Button | btn\_phan\_hoi\_y\_kien | Nút "Phản hồi ý kiến" |
| 6 | Button | btn\_luong\_nhanvien | Nút "Quản lý lương nhân viên" |

- Màn hình quản lý lịch nhân viên:

A screenshot of a phone

Description automatically generated

*Hình 4.2.2.: Giao diện quản lý lịch nhân viên*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | tv\_malv | Hiển thị mã lịch làm việc |
| 2 | EditText | editTextTieuDe | Nhập tiêu đề lịch làm việc |
| 3 | EditText | editTextMoTa | Nhập mô tả công việc |
| 4 | TextView | textViewNgayBatDau | Hiển thị ngày bắt đầu công việc |
| 5 | TextView | tvTimePickerDB | Hiển thị thời gian bắt đầu |
| 6 | TextView | textViewNgayKetThuc | Hiển thị ngày kết thúc công việc |
| 7 | TextView | tvTimePickerKT | Hiển thị thời gian kết thúc |
| 8 | EditText | editTextPhBan | Nhập thông tin phòng ban |
| 9 | Button | buttonAdd | Nút "Thêm" |
| 10 | Button | buttonUpdate | Nút "Cập nhật" |
| 11 | Button | buttonCancel | Nút "Hủy" |
| 12 | RecyclerView | recyclerViewSchedule | Danh sách hiển thị lịch làm việc |

- Màn hình quản lý lịch nhân viên trong công việc:

A screenshot of a phone

Description automatically generated

*Hình 4.2.3.: Giao diện quản lý lịch nhân viên trong công việc*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | txt\_malv | Hiển thị tiêu đề "Mã làm việc" |
| 2 | TextView | txt\_manv | Hiển thị tiêu đề "Mã nhân viên" |
| 3 | AutoCompleteTextView | auto\_complete\_spinner\_manv | Cho phép tự động điền và chọn mã nhân viên |
| 4 | Button | btn\_add | Nút "Thêm" để thêm nhân viên vào công việc |
| 5 | RecyclerView | recyclerView | Danh sách nhân viên đã được thêm vào công việc này |

- Màn hình quản lý nhân viên:

A screenshot of a phone

Description automatically generated

*Hình 4.2.4.: Giao diện quản lý nhân viên*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | (Không có ID) | Hiển thị tiêu đề "Danh sách nhân viên" |
| 2 | ListView | list\_nhanvien | Hiển thị danh sách nhân viên |

- Màn hình quản lý thông báo:

A screen shot of a phone

Description automatically generated

*Hình 4.2.5.: Giao diện quản lý thông báo*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | Button | btn\_add | Nút "Thêm" để thêm thông báo mới |
| 2 | RecyclerView | recyclerView | Hiển thị danh sách thông báo |

- Màn hình thêm thông báo:

A screen shot of a cell phone

Description automatically generated

*Hình 4.2.6.: Giao diện thêm thông báo*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | tvPhongBan | Hiển thị tiêu đề "Mã phòng ban" |
| 2 | TextView | (Không có ID) | Hiển thị tiêu đề "Nội dung: " |
| 3 | EditText | etNoiDung | Nhập nội dung thông báo |
| 4 | Button | btnGui | Nút "Gửi" thông báo |

## Mô tả UI/UX của giao diện Nhân viên

- Màn hình chính:

A screen shot of a phone

Description automatically generated

*Hình 4.3.1: Giao diện chính*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | Button | btn\_lich\_nhan\_vien | Mở giao diện xem lịch làm việc |
| 2 | Button | btn\_luong\_nhanvien | Mở giao diện bảng lương |
| 3 | Button | btn\_vietdon | Mở giao diện viết đơn |
| 4 | Button | btn\_xu\_ly\_vi\_pham | Mở giao diện vi phạm |
| 5 | Button | btn\_y\_kien | Mở giao diện ý kiến |

- Màn hình lịch làm việc:

A screen shot of a phone

Description automatically generated

*Hình 4.3.2.: Giao diện xem lịch làm việc*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | tv\_title | Hiển thị tiêu đề "Lịch làm việc" |
| 2 | Button | buttonT2 | Nút chọn ngày thứ 2 |
| 3 | Button | buttonT3 | Nút chọn ngày thứ 3 |
| 4 | Button | buttonT4 | Nút chọn ngày thứ 4 |
| 5 | Button | buttonT5 | Nút chọn ngày thứ 5 |
| 6 | Button | buttonT6 | Nút chọn ngày thứ 6 |
| 7 | Button | buttonT7 | Nút chọn ngày thứ 7 |
| 8 | RecyclerView | recycler\_view | Hiển thị danh sách lịch làm việc |

- Màn hình xem thông báo:  
A black cell phone with a black screen

Description automatically generated

*Hình 4.3.3.: Giao diện xem thông báo*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | RecyclerView | recyclerView | Hiển thị danh sách thông báo |

## Mô tả UI/UX của giao diện Admin

- Màn hình chính

A screenshot of a phone

Description automatically generated

*Hình 4.4.1: Giao diện chính*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | tvTime | Hiển thị thời gian |
| 2 | Button | btnManageEmployees | Mở giao diện "Quản lý nhân viên" |
| 3 | Button | btnManageDepartments | Mở giao diện "Quản lý phòng ban" |

- Màn hình chính quản lý phòng ban

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

*Hình 4.4.2.: Giao diện quản lý phòng ban*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | txt\_maph | Hiển thị tiêu đề "MAPH:" |
| 2 | EditText | edit\_maph | Nhập mã phòng ban |
| 3 | TextView | txt\_trph | Hiển thị tiêu đề "TRPH:" |
| 4 | AutoCompleteTextView | auto\_complete\_spinner\_trph | Chọn trưởng phòng từ danh sách nhân viên |
| 5 | Button | btn\_add | Nút "Thêm" phòng ban mới |
| 6 | Button | btn\_cancel | Nút "Hủy" |
| 7 | Button | btn\_apply | Nút "Xác nhận" |
| 8 | Button | btn\_more | Nút "Chi tiết" để xem thông tin chi tiết |
| 9 | RecyclerView | recyclerView | Hiển thị danh sách phòng ban và trưởng phòng |

- Màn hình chính quản lý nhân viên trong phòng ban

A screenshot of a cell phone

Description automatically generated

*Hình 4.4.3.: Giao diện quản lý nhân viên trong phòng ban*

* Các thành phần

| STT | Kiểu | Tên | Ý nghĩa |
| --- | --- | --- | --- |
| 1 | TextView | txt\_maph | Hiển thị tiêu đề "MAPH:" |
| 2 | TextView | txt\_trph | Hiển thị tiêu đề "TRPH:" |
| 3 | TextView | txt\_manv | Hiển thị tiêu đề "MANV:" |
| 4 | AutoCompleteTextView | auto\_complete\_spinner\_manv | Chọn mã nhân viên từ danh sách |
| 5 | Button | btn\_add | Nút "Thêm" nhân viên vào phòng ban |
| 6 | RecyclerView | recyclerView | Hiển thị danh sách nhân viên trong phòng ban |

# CHƯƠNG 5: THỰC NGHIỆM ĐỀ TÀI

| TT | Màn hình | Vấn đề | Ghi chú |
| --- | --- | --- | --- |
| 1 | Giao diện quản lý nhân viên trong phòng ban | Danh sách nhân viên chưa được lấy từ server, adapter chưa hiển thị theo như mong muốn | Đã được chỉnh sửa |
| 2 | Giao diện xem thông báo | Cập nhật theo thời gian thật | Đã được chỉnh sửa |
| 3 | Giao diện profile | Chưa hiển thị đúng | Đã được chỉnh sửa |
| 4 | Giao diện quản lý thông báo | Giám sát nhận, xem thông báo của nhân viên | Đã được chỉnh sửa |
| 5 | Giao diện nhắn tin | Kết nối với server | Đang được phát triển |
| 6 | Giao diện tính lương | Kết nối với server | Đang được phát triển |

*Bảng 4.4.1: Bảng các bug và tình hình khắc phục.*

# CHƯƠNG 6: KẾT LUẬN

## 6.1. Kết quả đạt được

Ứng dụng quản lý nhân viên trên thiết bị di động đang trở thành xu hướng mới trong thời đại công nghệ 4.0. Với sự phát triển của smartphone và các thiết bị di động khác, việc sử dụng ứng dụng trên điện thoại để quản lý nhân sự trở nên thuận tiện và dễ dàng hơn bao giờ hết.

Ứng dụng quản lý nhân viên trên thiết bị di động có nhiều ưu điểm vượt trội so với các giải pháp quản lý truyền thống. Dưới đây là những chức năng cơ bản đạt được của nhóm:

* Đối với user:
* Đăng nhập, đăng xuất.
* Đổi mật khẩu
* Nhắn tin
* Xem lịch làm việc
* Xem lương
* Xem thông tin cá nhân
* Nhận thông báo
* Viết đơn
* Ý kiến
* Xem các vi phạm
* Đối với admin:
* Quản lý thêm, xóa, sửa, các phòng ban
* Đổi mật khẩu
* Bổ nhiệm quản lý, quản lý nhiên viên các phòng ban

Nhờ việc sử dụng mô hình MVP, việc quản lý mã nguồn chương trình và phân chia công việc cho các thành viên trong nhóm trở nên đơn giản hơn. Khi một thành phần nào đó bị hỏng thì chỉ cần sửa thành phần đó mà không cần cập nhật lại toàn bộ.

Ngoài ra việc thiết kế giao diện cũng được đẩy mạnh bằng việc đưa ra 2 giao diện của hệ thống (giao diện dành cho admin và giao diện dành cho user). Nhờ đó, việc sử dụng hệ thống (đối với user) và việc quản lý hệ thống (đối với các quản trị viên) trở nên tách biệt và dễ dàng hơn.

## 6.2. Khó khăn, hạn chế

Trong quá trình thực hiện đề tài nhóm đã gặp khá nhiều khó khăn như:

* Phân bố thời gian chưa hợp lý (vì các thành viên trong nhóm học nhiều môn và có nhiều đồ án khác nhau) dẫn đến 1 số tính năng chưa làm được
* Vấn đề mạng và đường truyền khiến cho việc trao đổi thông tin giữa các thành viên bị hạn chế.
* Thiết bị ảo chạy tốn ram nên hay bị lag
* Học và sử dụng những công nghệ và thư viện mới trong thời gian ngắn
* Dùng github quản lý code chưa thực sự hiệu quả dẫn đến nhiều lỗi xung đột xảy ra trong quá trình pull/push code và mất thời gian fix.

Vì thời gian nghiên cứu và hiện thực đề tài còn giới hạn nên kết quả của đề tài còn có một số hạn chế. Cụ thể phần demo có các hạn chế như sau:

* Chức năng viết/duyệt đơn
* Chức năng viết/ duyệt ý kiến
* Chức năng nhắn tin
* Chức năng quản lý vi phạm
  1. **Hướng phát triển**

Như đã trình bày ở các phần trước, nhóm tác giả nhận định còn rất nhiều việc cần phải làm để hoàn thiện Ứng dụng quản lý nhân sự. Chính vì vậy nhóm đặt ra mục tiêu để phát triển đề tài trong tương lai như sau:

* - Hoàn thiện các chức năng còn sai sót
* - Bổ sung một số tính năng mới trong tương lai
* - Thêm ngôn ngữ tiếng anh cho ứng dụng.
* - Xử lý để tốc độ load dữ liệu của app nhanh và mượt hơn

# BẢNG PHÂN CHIA CÔNG VIỆC

| Họ và tên | Github | Công việc | Đánh giá công việc |
| --- | --- | --- | --- |
| Nguyễn Hoàng Khang | Cazo (Cazo04) | Dựa trên timeline và commit trên github  <https://github.com/Cazo04/NT118>  <https://github.com/Cazo04/NT118_Server_API>  [Timeline](https://docs.google.com/spreadsheets/d/1td3YC7V9WWrcmGIMCZ5PJS9unJYxm8N8v4HXi73rJYY/edit?usp=sharing) | 30% |
| Huỳnh Quang Khải | ABSCTer | 20% |
| Đặng Gia Bảo | danggbao | 20% |
| Nguyễn Hải Đăng | ddawng-8 | 30% |

# TÀI LIỆU THAM KHẢO

| [1] | vi.wikipedia.org, "vi.wikipedia.org," [Online]. Available: https://vi.wikipedia.org/wiki/Ứng\_dụng\_trên\_thiết\_bị\_di\_động. |
| --- | --- |
| [2] | vi.wikipedia.org, "Hệ\_điều\_hành\_di\_động," [Online]. Available: https://vi.wikipedia.org/wiki/Hệ\_điều\_hành\_di\_động. |
| [3] | itzone.com.vn. [Online]. Available: https://itzone.com.vn/vi/article/phan-loai-ung-dung-di-dong/. |
| [4] | tutorialspoint.com. [Online]. Available: http://www.tutorialspoint.com/android/android\_architecture.htm. |
| [5] | topdev.vn. [Online]. Available: https://topdev.vn/blog/cac-loai-layout-trong-android-relativelayout-linearlayout/. |
| [6] | viblo.asia. [Online]. Available: https://viblo.asia/p/tim-hieu-ve-cac-loai-layout-trong-android-XL6lANjB5ek. |
| [7] | vi.wikipedia.org. [Online]. Available: https://vi.wikipedia.org/wiki/SQLite. |