khung4khung3khung1

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC VĂN HIẾN**

**KHOA KỸ THUẬT – CÔNG NGHỆ**

**🙟🕮🙝**



**TIỂU LUẬN MÔN LẬP TRÌNH ANDROID**

**ĐỀ TÀI**

**Xây Dựng Ứng Dụng Quản lý Phòng Ban và Nhân Viên phòng ban**

Giáo viên hướng dẫn : Nguyễn Hữu Vĩnh

Sinh viên thực hiện :Lê Thành Công

MSSV : 161A010130

khung2...........**, tháng** ... **năm** ........

MỤC LỤC

[Chương 1. Giới Thiệu Phần Mềm 3](#_Toc3907429)

[Chương 2. Giới thiệu công nghệ 4](#_Toc3907430)

[2.1 Giới thiệu về android 4](#_Toc3907431)

[2.1.1 Giao diện trong Android 5](#_Toc3907432)

[2.1.2 Quản lý bộ nhớ 7](#_Toc3907433)

[2.1.3 Cộng đồng mã nguồn mở 8](#_Toc3907434)

[2.1.4 Tỷ lệ sử dụng các phiên bản Android 9](#_Toc3907435)

[2.2 Giới Thiệu Java 11](#_Toc3907436)

[2.2.1 Phương châm 11](#_Toc3907437)

[2.2.2 Sự phổ biến 12](#_Toc3907438)

[2.3 Giới Thiệu Android Studio 12](#_Toc3907439)

[2.3.1 Project và cấu trúc tệp 13](#_Toc3907440)

[2.3.2 Trình quản lý thiết bị ảo của Android (AVD) 14](#_Toc3907441)

[2.3.4 Truy cập file dữ liệu 15](#_Toc3907442)

[Chương 3. Phân tích thiết kế hệ thống 17](#_Toc3907443)

[3.1 Nhu cầu 17](#_Toc3907444)

[3.2 Mô hình EER 18](#_Toc3907445)

[3.3 Ý nghĩa bảng dữ liệu các trường 18](#_Toc3907446)

[Chương 4. Cài đặt và kiểm thử 20](#_Toc3907447)

[4.1 Màn Hình Home 20](#_Toc3907448)

[4.2 Các form chức năng 21](#_Toc3907449)

[4.2.1 Form thêm phòng ban 21](#_Toc3907450)

[4.2.2 Form thêm nhân viên 22](#_Toc3907451)

[4.2.3 Form Danh Sách Nhân Viên 23](#_Toc3907452)

[4.2.4 Form Sửa nhân viên 23](#_Toc3907453)

[4.2.5 Form Chuyển phòng ban 24](#_Toc3907454)

[4.2.6 Form Xóa nhân viên 25](#_Toc3907455)

[4.2.7 Form Lập trưởng Phòng/ Phó Phòng 26](#_Toc3907456)

[4.2.8 Form Xóa phòng ban 27](#_Toc3907457)

[KẾT LUẬN 29](#_Toc3907458)

[1. Đánh giá kết quả 29](#_Toc3907459)

[2. Những hạn chế của hệ thống 29](#_Toc3907460)

[3. Hướng phát triển trong tương lai 30](#_Toc3907461)

**Chương 1. Giới Thiệu Phần Mềm**

Đây là phần mềm đơn giản để quản lý các phòng ban trong công ty.

Phần mềm có thể quản lý:

* 1. Thông tin nhân viên: họ và tên nhân viên, mã số nhân viên, giới tính.
  2. Thông tin phòng ban: tên phòng ban, số nhân viên có trong phòng ban, trưởng phòng, phó phòng.

Có thể:

1. Thêm nhân viên với cho phòng ban có sẵn.
2. Xem danh sách nhân viên có trong phòng.
3. Lập trưởng phòng, phó phòng.
4. Xóa phòng ban.

**Chương 2. Giới thiệu công nghệ**

**2.1 Giới thiệu về android**

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng.

Ban đầu, Android được phát triển bởi Tổng công ty Android, với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache.

Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do.

Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Vào tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, vượt qua Symbian OS vào quý 4 năm 2010, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh, và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu.Kết quả là mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, Android đã xuất hiện trên TV, máy chơi game và các thiết bị điện tử khác.

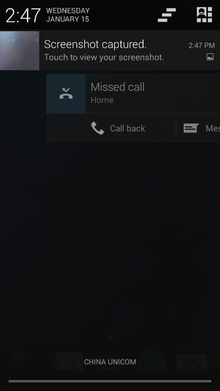
Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý. Những dự án này bổ sung các tính năng cao cấp cho những người dùng thích tìm tòi hoặc đưa Android vào các thiết bị ban đầu chạy hệ điều hành khác.

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày.Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "cuộc chiến điện thoại thông minh" giữa các công ty công nghệ.

**2.1.1 Giao diện trong Android**

Giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo giãn và thu lại để xử lý các đối tượng trên màn hình.Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng.

Những thiết bị phần cứng bên trong như gia tốc kế, con quay hồi chuyển và cảm biến khoảng cách được một số ứng dụng sử dụng để phản hồi một số hành động khác của người dùng, ví dụ như điều chỉnh màn hình từ chế độ hiển thị dọc sang chế độ hiển thị ngang tùy theo vị trí của thiết bị, hoặc cho phép người dùng lái xe đua bằng xoay thiết bị, giống như đang điều khiển vô-lăng.

**

*Thanh thông báo được kích hoạt khi trượt từ cạnh trên màn hình; các thông báo có thể được xoá đi bằng cách trượt sang 2 bên và có thể thêm một vài chức năng khác (như thông báo "Cuộc gọi nhỡ" (Missed call) ở trên)*

Các thiết bị Android sau khi khởi động sẽ hiển thị màn hình chính, điểm khởi đầu với các thông tin chính trên thiết bị, tương tự như khái niệm desktop (bàn làm việc) trên máy tính để bàn. Màn hính chính Android thường gồm nhiều biểu tượng (icon) và tiện ích (widget); biểu tượng ứng dụng sẽ mở ứng dụng tương ứng, còn tiện ích hiển thị những nội dung sống động, cập nhật tự động như dự báo thời tiết, hộp thư của người dùng, hoặc những mẩu tin thời sự ngay trên màn hình chính.

Màn hình chính có thể gồm nhiều trang xem được bằng cách vuốt ra trước hoặc sau, mặc dù giao diện màn hình chính của Android có thể tùy chỉnh ở mức cao, cho phép người dùng tự do sắp đặt hình dáng cũng như hành vi của thiết bị theo sở thích.

Những ứng dụng do các hãng thứ ba có trên Google Play và các kho ứng dụng khác còn cho phép người dùng thay đổi "chủ đề" của màn hình chính, thậm chí bắt chước hình dáng của hệ điều hành khác như Windows Phone chẳng hạn. Phần lớn những nhà sản xuất, và một số nhà mạng, thực hiện thay đổi hình dáng và hành vi của các thiết bị Android của họ để phân biệt với các hãng cạnh tranh.

Ở phía trên cùng màn hình là thanh trạng thái, hiển thị thông tin về thiết bị và tình trạng kết nối. Thanh trạng thái này có thể "kéo" xuống để xem màn hình thông báo gồm thông tin quan trọng hoặc cập nhật của các ứng dụng, như email hay tin nhắn SMS mới nhận, mà không làm gián đoạn hoặc khiến người dùng cảm thấy bất tiện.

Trong các phiên bản đời đầu, người dùng có thể nhấn vào thông báo để mở ra ứng dụng tương ứng, về sau này các thông tin cập nhật được bổ sung thêm tính năng, như có khả năng lập tức gọi ngược lại khi có cuộc gọi nhỡ mà không cần phải mở ứng dụng gọi điện ra. Thông báo sẽ luôn nằm đó cho đến khi người dùng đã đọc hoặc xóa nó đi.

### 2.1.2 Quản lý bộ nhớ

Vì các thiết bị Android chủ yếu chạy bằng pin, nên Android được thiết kế để quản lý bộ nhớ (RAM) để giảm tối đa tiêu thụ điện năng, trái với hệ điều hành máy tính để bàn luôn cho rằng máy tính sẽ có nguồn điện không giới hạn.

Khi một ứng dụng Android không còn được sử dụng, hệ thống sẽ tự động ngưng nó trong bộ nhớ - trong khi ứng dụng về mặt kỹ thuật vẫn "mở", những ứng dụng này sẽ không tiêu thụ bất cứ tài nguyên nào (như năng lượng pin hay năng lượng xử lý) và nằm đó cho đến khi nó được cần đến. Cách làm như vậy có lợi kép là vừa làm tăng khả năng phản hồi nói chung của thiết bị Android, vì ứng dụng không nhất phải đóng rồi mở lại từ đầu, vừa đảm bảo các ứng dụng nền không làm tiêu hao năng lượng một cách không cần thiết.

Android quản lý các ứng dụng trong bộ nhớ một cách tự động: khi bộ nhớ thấp, hệ thống sẽ bắt đầu diệt ứng dụng và tiến trình không hoạt động được một thời gian, sắp theo thời điểm cuối mà chúng được sử dụng (tức là cũ nhất sẽ bị tắt trước). Tiến trình này được thiết kế ẩn đi với người dùng, để người dùng không cần phải quản lý bộ nhớ hoặc tự tay tắt các ứng dụng. Tuy nhiên, sự che giấu này của hệ thống quản lý bộ nhớ Android đã dẫn đến sự thịnh hành của các ứng dụng tắt chương trình của bên thứ ba trên cửa hàng Google Play; những ứng dụng kiểu như vậy được cho là có hại nhiều hơn có lợi.

### 2.1.3 Cộng đồng mã nguồn mở

Android có một cộng đồng các lập trình viên và những người đam mê rất năng động. Họ sử dụng mã nguồn Android để phát triển và phân phối những phiên bản chỉnh sửa của hệ điều hành.

Các bản Android do cộng đồng phát triển thường đem những tính năng và cập nhật mới vào nhanh hơn các kênh chính thức của nhà sản xuất/nhà mạng, tuy không được kiểm thử kỹ lưỡng cũng như không có đảm bảo chất lượng; cung cấp sự hỗ trợ liên tục cho các thiết bị cũ không còn nhận được bản cập nhật chính thức; hoặc mang Android vào những thiết bị ban đầu chạy một hệ điều hành khác, như HP Touchpad. Các bản Android của cộng đồng thường được root sẵn và có những điều chỉnh không phù hợp với những người dùng không rành rẽ, như khả năng ép xung hoặc tăng/giảm áp bộ xử lý của thiết bị. CyanogenMod là firmware của cộng đồng được sử dụng phổ biến nhất, và hoạt động như một tổ chức của số đông khác.

Trước đây, nhà sản xuất thiết bị và nhà mạng tỏ ra thiếu thiện chí với việc phát triển firmware của bên thứ ba. Những nhà sản xuất còn thể hiện lo ngại rằng các thiết bị chạy phần mềm không chính thức sẽ hoạt động không tốt và dẫn đến tốn tiền hỗ trợ. Hơn nữa, các firmware đã thay đổi như CyanogenMod đôi khi còn cung cấp những tính năng, như truyền tải mạng (tethering), mà người dùng bình thường phải trả tiền nhà mạng mới được sử dụng.

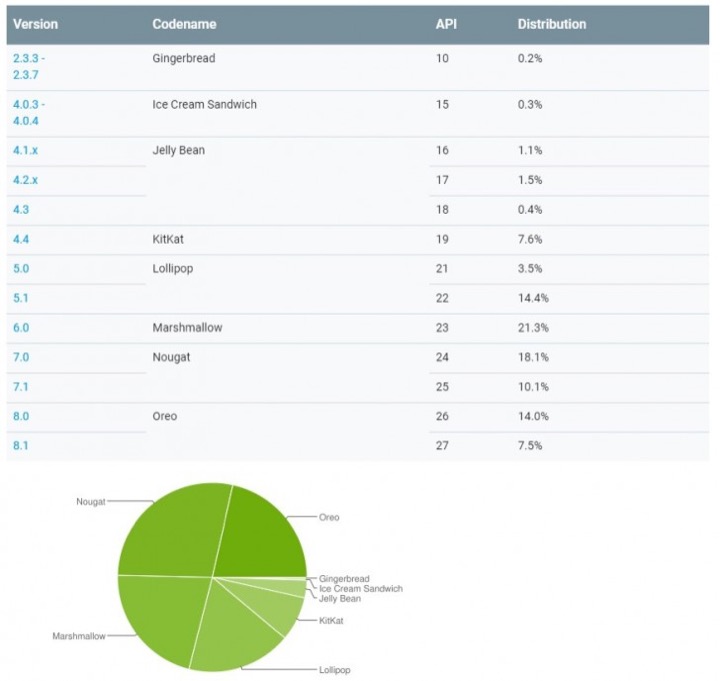
Kết quả là nhiều thiết bị bắt đầu đặt ra hàng rào kỹ thuật như khóa bootloader hay hạn chế quyền truy cập root. Tuy nhiên, khi phần mềm do cộng đồng phát triển ngày càng trở nên phổ biến, và sau một thông cáo của Thư viện Quốc hội Hoa Kỳ cho phép "jailbreak" (vượt ngục) thiết bị di động, các nhà sản xuất và nhà mạng đã tỏ ra mềm mỏng hơn với các nhà phát triển thứ ba, thậm chí một số hãng như HTC, Motorola, Samsung và Sony còn hỗ trợ và khuyến khích phát triển.

Kết quả của việc này là dần dần nhu cầu tìm ra các hạn chế phần cứng để cài đặt được firmware không chính thức đã bớt đi do ngày càng nhiều thiết bị được phát hành với bootloader đã mở khóa sẵn hoặc có thể mở khóa, tương tự như điện thoại dòng Nexus, tuy rằng thông thường họ sẽ yêu cầu người dùng từ bỏ chế độ bảo hành nếu họ làm như vậy.

Tuy nhiên, tuy được sự chấp thuận của nhà sản xuất, một số nhà mạng tại Mỹ vẫn bắt buộc điện thoại phải bị khóa.

Việc mở khóa và "hack" điện thoại thông minh và máy tính bảng vẫn còn là tác nhân gây căng thẳng giữa cộng đồng và công nghiệp. Cộng đồng luôn biện hộ rằng sự hỗ trợ không chính thức ngày càng trở nên quan trọng trước việc nền công nghiệp không cung cấp các bản cập nhật thường xuyên và/hoặc ngưng hỗ trợ cho chính các thiết bị của họ.

**2.1.4 Tỷ lệ sử dụng các phiên bản Android**



*Tỷ lệ các phiên bản Android tháng 10/ 2018*

**2.2 Giới Thiệu Java**

Java (phiên âm Tiếng Việt: "Gia-va") là một ngôn ngữ lập trình hướng đối tượng (OOP) và dựa trên các lớp (class). Khác với phần lớn ngôn ngữ lập trình thông thường, thay vì biên dịch mã nguồn thành mã máy hoặc thông dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã nguồn thành bytecode, bytecode sau đó sẽ được môi trường thực thi (runtime environment) chạy.

Trước đây, Java chạy chậm hơn những ngôn ngữ dịch thẳng ra mã máy như C và C++, nhưng sau này nhờ công nghệ "biên dịch tại chỗ" - Just in time compilation, khoảng cách này đã được thu hẹp, và trong một số trường hợp đặc biệt Java có thể chạy nhanh hơn. Java chạy nhanh hơn những ngôn ngữ thông dịch như Python, Perl, PHP gấp nhiều lần. Java chạy tương đương so với C#, một ngôn ngữ khá tương đồng về mặt cú pháp và quá trình dịch/chạy.

Cú pháp Java được vay mượn nhiều từ C & C++ nhưng có cú pháp hướng đối tượng đơn giản hơn và ít tính năng xử lý cấp thấp hơn. Do đó việc viết một chương trình bằng Java dễ hơn, đơn giản hơn, đỡ tốn công sửa lỗi hơn.

Trong Java, hiện tượng rò rỉ bộ nhớ hầu như không xảy ra do bộ nhớ được quản lý bởi Java Virtual Machine (JVM) bằng cách tự động "dọn dẹp rác". Người lập trình không phải quan tâm đến việc cấp phát và xóa bộ nhớ như C, C++. Tuy nhiên khi sử dụng những tài nguyên mạng, file IO, database (nằm ngoài kiểm soát của JVM) mà người lập trình không đóng (close) các streams thì rò rỉ dữ liệu vẫn có thể xảy ra.

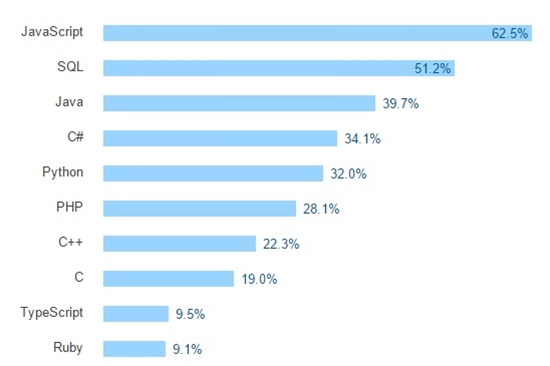
**2.2.1 Phương châm**

Có 6 mục tiêu chính trong việc xây dựng ngôn ngữ Java:

1. Đơn giản, hướng đối tượng và quen thuộc.
2. Mạnh mẽ và an toàn.
3. Kiến trúc trung lập và di động.
4. Thực thi với hiệu suất cao.
5. Dịch ra bytecode, phân luồng và năng động.
6. Dễ sử dụng cho người dùng Java

**2.2.2 Sự phổ biến**

Tính đến ngày 9 tháng 12 năm 2017, Java là ngôn ngữ lập trình phổ biến nhất thế giới với tỉ lệ 13,268%, nhưng đã giảm 4,59% so với cùng kì năm trước. Java và C vẫn luôn chiếm hai vị trí đầu tiên trong bảng các ngôn ngữ lập trình phổ biến nhất suốt 20 năm qua, nhưng Java là ngôn ngữ duy nhất giữ được tỉ lệ trên 10%, trong khi C đã tụt xuống 6,478% vào ngày 2/8/2017.



*Sự phổ biến của các ngôn ngữ lập trình (1/2017)*

**2.3 Giới Thiệu Android Studio**

Android Studio là IDE chính thức trong phát triển ứng dụng Android, dựa trên IntelliJ IDEA. Ngoài các khả năng đáng mong đợi từ IntelliJ, Android Studio còn cung cấp:

- Hệ thống Gradle-based linh hoạt

- Xây dựng các biến thể và tạo nhiều tệp APK

- Code các mẫu template để hỗ trợ các tính năng app thông thường

- Chỉnh sửa bố cục đa dạng với khả năng kéo và thả theme

- Công cụ lint giúp nắm bắt hiệu suất, khả năng sử dụng, phiên bản tương thích và các vấn đề khác.

- ProGuard và ứng dụng ký app-signing

- Hỗ trợ tích hợp trên Google Cloud Platform, cho phép dễ dàng tích hợp Google Cloud Messaging và App Engine.

- Và nhiều tính năng khác nữa.

Các tính năng cơ bản

Nếu bạn mới sử dụng Android Studio hoặc giao diện IntelliJ IDEA, sau đây là một số tính năng chính của Android Studio.

**2.3.1 Project và cấu trúc tệp**

Chế độ xem project trên Android

Theo mặc định, Android Studio hiển thị các tệp project trong chế độ xem project trên Android. Chế độ xem này cho phép xem cấu trúc project theo lát cắt, cung cấp truy cập nhanh vào các tệp source chính của các project trên Android và giúp bạn làm việc với hệ thống Gradle-based. Chế độ xem dự án Android:

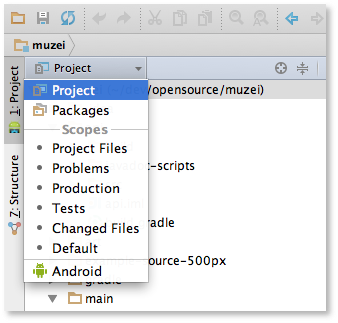
Android studio là gì? Tất cả các tính năng có trong Android studio - Ảnh 1.

- Hiển thị các thư mục gốc quan trọng nhất ở cấp cao nhất của phân cấp module.

- Nhóm các build file cho tất cả các module vào một thư mục chung.

- Nhóm tất cả các file kê khai cho từng module vào một thư mục chung.

- Hiển thị các tệp tài nguyên từ tất cả các tập nguồn nguồn Gradle.

- Nhóm các tệp tài nguyên cho các khu vực và kiểu màn hình khác nhau trong một nhóm duy nhất cho mỗi loại tài nguyên.

*Cấu trúc tập tin trên Android Studio*

Chế độ xem project trong Android hiển thị tất cả các build files ở cấp cao nhất của hệ thống phân cấp dự án theo Gradle Scripts. Mỗi module dự án xuất hiện dưới dạng một thư mục ở cấp cao nhất của hệ thống phân cấp dự án và chứa bốn thành phần sau ở cấp cao nhất:

java/ - Tệp nguồn cho mô-đun.

manifest/ - Tệp kê khai cho mô-đun.

res/ - Tài nguyên tập tin cho mô-đun.

Gradle Scripts/ - Các file Gradle thiết kế và cố định

**2.3.2 Trình quản lý thiết bị ảo của Android (AVD)**

AVD Manager đã cập nhật các link lên màn hình để giúp bạn chọn các cấu hình thiết bị phổ biến nhất, lựa chọn kích thước màn hình và độ phân giải trong chế độ xem trước.

Trình quản lý AVD có liên kết với bộ mô phỏng cho các thiết bị Nexus 6 và Nexus 9. AVD cũng hỗ trợ tạo giao diện thiết bị Android tùy chỉnh dựa trên các đặc tính mô phỏng cụ thể và gán các giao diện đó cho cấu hình phần cứng.

Android Studio cài đặt Intel® x86 Hardware Accelerated Execution Manager (HAXM) và tạo bộ mô phỏng mặc định để tạo nhanh ứng dụng mẫu.

**2.3.4 Truy cập file dữ liệu**

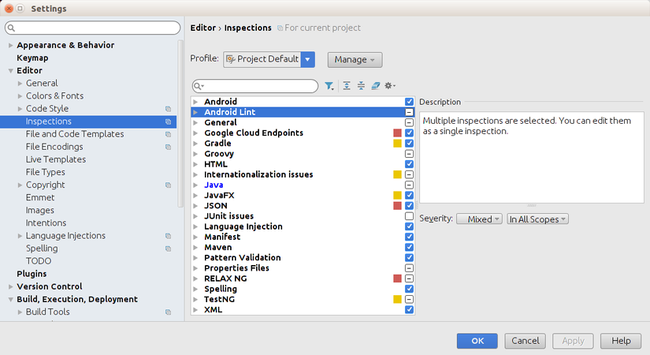
Các công cụ SDK của Android, như Systrace, logcat và Traceview, tăng hiệu suất và gỡ lỗi dữ liệu giúp phân tích ứng dụng chi tiết.

Để xem các file dữ liệu khả dụng, click Capture ở góc bên trái trên cửa sổ runtime. Trong danh sách các tệp được tạo, click đúp vào tệp muốn xem dữ liệu. Chuột phải vào bất kỳ tệp .hprof nào để chuyển đổi chúng sang định dạng tệp .hprof tiêu chuẩn.

Trong Android Studio, lint đã được cấu hình và các kiểm tra IDE khác sẽ tự động chạy bất cứ khi nào bạn thực hiện biên dịch chương trình. Ngoài các kiểm tra lint đã cấu hình, kiểm tra mã IntelliJ bổ sung và chạy xác thực chú thích để review dòng code.

Với Android Studio, bạn cũng có thể chạy kiểm tra lint cho một biến build xác định hoặc cho tất cả các biến build. Bạn có thể cấu hình kiểm tra lint bằng cách thêm thuộc tính lintOptions vào cài đặt Android trong tệp build.gradle.

Bạn cũng có thể quản lý hồ sơ kiểm tra và cấu hình kiểm tra trong Android Studio. Chọn File > Settings >, mở rộng các tùy chọn Trình Editor và chọn Inspections. Trang Inspection Configuration sẽ xuất hiện cùng với các kiểm tra được hỗ trợ.



chạy kiểm tra thủ công trong Android Studio, chọn Analyze > Inspect Code. Hộp thoại Inspections Scope sẽ xuất hiện để bạn có thể xác định cấu hình và phạm vi kiểm tra mong muốn.

**Chương 3. Phân tích thiết kế hệ thống**

**3.1 Nhu cầu**

a. Về giao diện

- Chương trình phải hoàn toàn bằng tiếng Việt

- Giao diện dễ hiểu, thân thiện với người dùng.

- Dễ sử dụng, những người không biết tin học có thể sử dụng được chương trình.

- Ít thao tác.

b/ Phân tích hệ thống về chức năng.

- Lưu trữ thông tin của nhân viên và thông tin của phòng

- Thay đổi thông tin của nhân viên và thông tin của phòng.

c/ Phân tích hệ thống về dữ liệu

\_ Mỗi Phòng ban của công ty có 1 mã phòng và tên phòng. Mã phòng và tên phòng không trùng nhau.

\_Mỗi phòng có 1 trưởng phòng và 1 hoặc nhiều phó phòng.

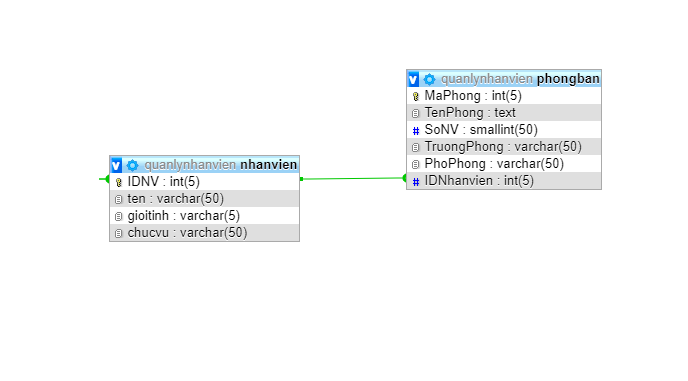
\_Mỗi phòng có nhiều nhân viên.

\_Nhân viên có thể là trưởng phòng hoặc phó phòng.

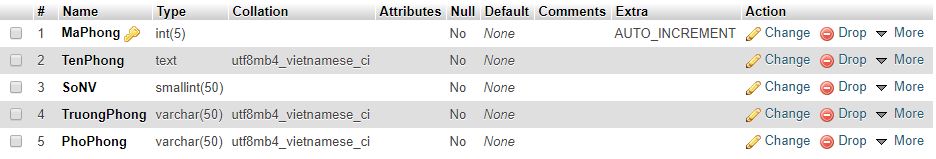
\_Thông tin của phòng: mã phòng, tên phòng, số lượng nhân viên của phòng, mã số nhân viên của trưởng phòng và phó phòng.

\_Thông tin của nhân viên gồm: Mã số nhân viên, tên nhân viên, giới tính, chức vụ.

**3.2 Mô hình EER**

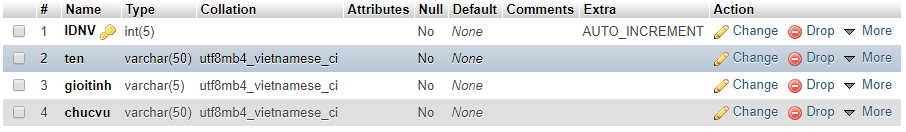


**3.3 Ý nghĩa bảng dữ liệu các trường**

**Bảng Phòng Ban**

|  |  |
| --- | --- |
| Tên trường | Nội dung |
| MaPhong | Mã số của phòng ban |
| TenPhong | Tên phòng ban |
| SoNV | Số lượng nhân viên của phòng |
| TruongPhong | Thông tin trưởng phòng |
| PhoPhong | Thông tin phó phòng |

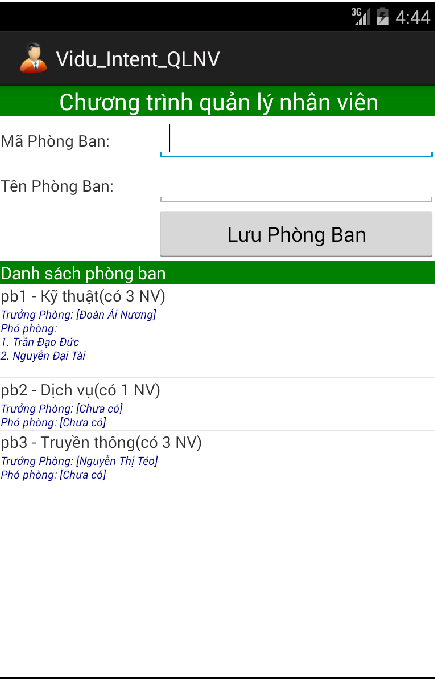
**Bảng Nhân Viên**



|  |  |
| --- | --- |
| Tên trường | Nội dung |
| IDNV | Mã số nhân viên |
| ten | Tên nhân viên |
| gioitinh | Giới tính của nhân viên |
| chucvu | Chức vụ nhân viên |

**Chương 4. Cài đặt và kiểm thử**

* 1. **Màn Hình Home**



Ý nghĩa hoạt động:

- Là trang đầu tiên được load khi người dùng truy cập đến trang ứng dụng.

- Vừa là trang chủ vừa đóng vài trò là 1 trang thêm phòng ban.

\* Qui tắc hoạt động:

- Được nhìn thấy và sử dụng bởi tất cả những ai truy cập vào ứng dụng.

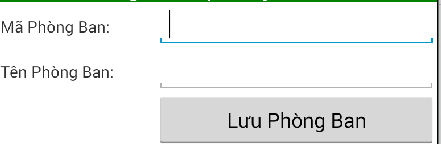
\* Các thao tác màn hình:

- Người truy cập có thể xem danh sách các phòng ban hiện có của công ty bên dưới danh sách phòng ban.

**4.2 Các form chức năng**

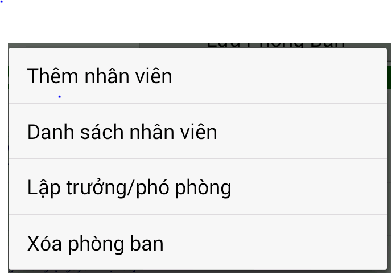
**4.2.1 Form thêm phòng ban**

- Người truy cập có thể điền đầy đủ thông tin vào các EditText và nhấn vào button “Lưu phòng ban” để thêm 1 phòng ban cho công ty.



- Input: MaPhong, TenPhong

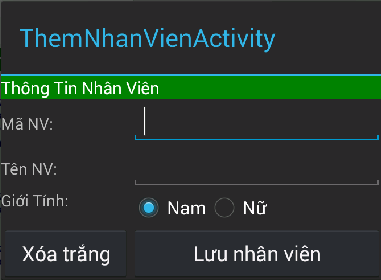
- Output:Tạo ra một phòng mới

\_ Khi longclick vào 1 phòng ban hiện có xuất hiện cửa sổ

Đây là các lựa chọn để thay đổi thông tin của phòng.

**4.2.2 Form thêm nhân viên**

\_Khi nhấn vào lựa chọn thêm nhân viên sẽ xuất hiện 1 bảng



\_Người dùng điền đầy đủ thông tin hợp lệ vào các EditText và CheckBox và nhấn vào button “Lưu nhân viên" để thêm 1 nhân mới cho phòng.

\_Người dùng nhấn vào button “Xóa trắng” để xóa hết thông tin vừa nhập vào các EditText nhưng chưa lưu.

\_Input: MaNV, ten

\_Output: Thông tin nhân viên

**4.2.3 Form Danh Sách Nhân Viên**

**\_**Khi người dùng nhấn vào lựa chọn “Danh sách nhân viên” sẽ xuất hiện trang

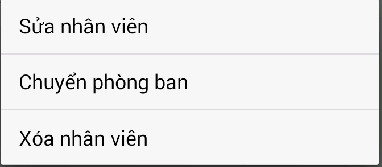


\_Đây là danh sách nhân viên của phòng hiện có

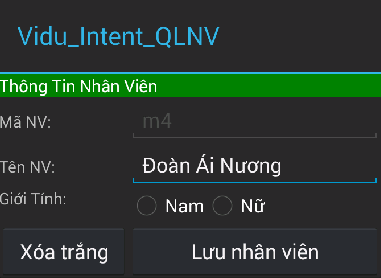
\_Nhấn vào button để quay lại màn hình Home.

**4.2.4 Form Sửa nhân viên**

\_Khi Nhấn vào một thông tin nhân viên sẽ xuất hiện các lựa chọn



\_Khi nhấn vào lựa chọn “Sửa nhân viên” sẽ có form sửa nhân viên xuất hiện

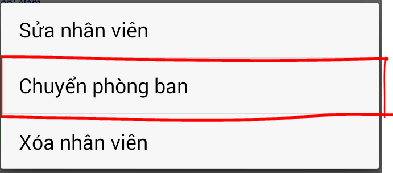


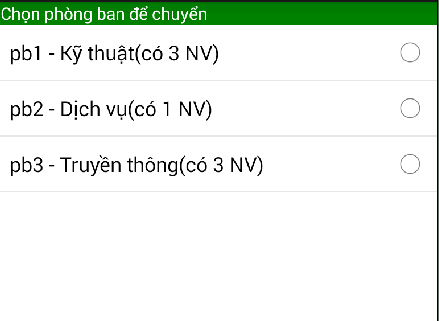
\_Form sửa nhân viên không thể sửa MaNV chỉ có thể sửa tên nhân viên và giới tính.

\_Nhấn vào button “Lưu nhân viên" để cập nhập nhân viên.

**4.2.5 Form Chuyển phòng ban**

\_Khi nhấn vào “Chuyển phòng ban” sẽ xuất hiện form





\_Đây là các phòng hiện có trong công ty

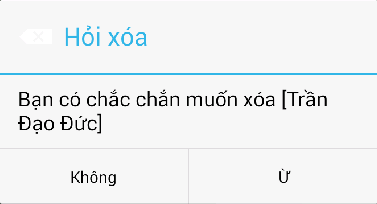
\_Người dùng có thể check vào CheckBox và nhấn vào button để lưu lại thông tin của phòng ban.

**4.2.6 Form Xóa nhân viên**

****

Người dùng nhấn vào lựa chọn “Xóa nhân viên” để xóa viên để xóa nhân viên khỏi công ty.

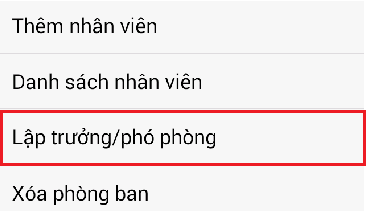
\_Khi nhấn xóa sẽ có thông báo



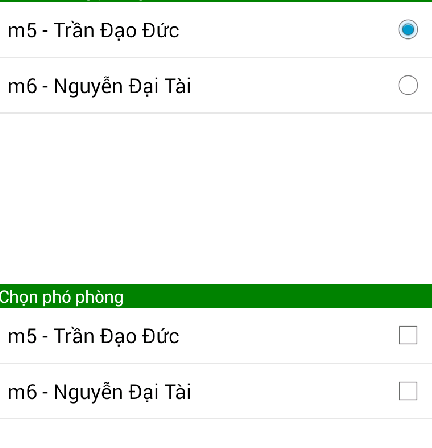
\_Nhấn “Không" để hủy hành động xóa

\_Nhấn “Ừ” để xóa

**4.2.7 Form Lập trưởng Phòng/ Phó Phòng**

****

**\_**Nhấn vào lựa chọn “Lập trưởng Phòng/ Phó Phòng” sẽ xuất hiện trang

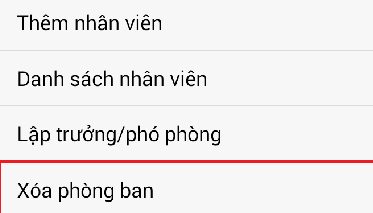


\_Người dùng check vào radio button để chọn trưởng phòng.

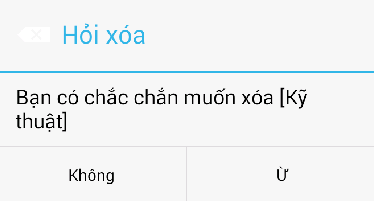
\_Người dùng check vào CheckBox để chọn phó phòng.

\_Nhấn vào để lưu thông tin trưởng Phòng/ Phó Phòng.

**4.2.8 Form Xóa phòng ban**



**\_**Khi người dùng nhấn vào lựa chọn “Xóa phòng ban” sẽ xuất hiện thông báo



\_ Chọn “ Không ” để hủy xóa

\_Chọn “ Ừ”để xóa phòng ban

**KẾT LUẬN**

**1. Đánh giá kết quả**

Trong quá trình thực hiện, em đã cố gắng hết sức để tìm hiểu và cài đặt chương trình nhưng vì thời gian có hạn nên chưa giải quyết được tất cả các vấn đề đặt ra. Em rất mong nhận được sự thông cảm của quý thầy cô. Em xin chân thành cảm ơn.

Những kết quả đạt được:

+ Về công nghệ:

- Tìm hiểu và nắm bắt được các công cụ thiết kế Android.

- Biết được cách thiết kế giao diện android cũng như cách tổ chức CSDL.

+ Về cài đặt chương trình:

- Quản lý được nhập xuất dữ liệu thực tế

- Cho phép tra cứu các thông tin thực tế

- Cho phép dễ dàng cập nhật khi có thay đổi

+ Tính năng của chương trình:

- Thông tin phòng ban và nhân viên được cập nhật kịp thời, chính xác.

- Giao diện thân thiện với người dùng.

**2. Những hạn chế của hệ thống**

- Chưa kiểm tra hết các trường hợp sảy ra dẫn đến phát sinh nhiều lỗi

- Chức năng quản lý còn đơn giản

- Chưa thực hiện được chức năng thống kê báo cáo

- Giao diện hệ thống còn đơn giản.

- Chưa có chức năng liên lạc với nhân viên

- Do điều kiện, chương trình mới chỉ chạy thử trên máy cá nhân.

**3. Hướng phát triển trong tương lai**

- Thực hiện kiểm tra hết các trường hợp có thể có để hạn chế tối đa các lỗi phát sinh từ hệ thống.

- Phát triển và hoàn thiện chức năng quản lý.

- Thực hiện chức năng thống kê báo cáo.

- Chỉnh sửa giao diện để thân thiện và bắt mắt hơn.

- Cho phép chạy chính thức trên mạng.

- Hỗ trợ nhập xuất từ file

- Bảo mật hệ thống tốt hơn.