**TRƯỜNG ĐẠI HỌC ĐIỆN LỰC**

Khoa Công Nghệ Thông Tin

-----🙞🙜🕮🙞🙜-----



**ĐỒ ÁN MÔN HỌC**

**LẬP TRÌNH HỆ THỐNG**

**Đề Tài: Ứng Dụng Chặn Cuộc Gọi**

Giảng viên hướng dẫn : Ngô Hoàng Huy

Sinh viên thực hiện : Đinh Văn Đông

Trần Thị Diệu Ninh

***Hà Nội, tháng 06 năm 2018***

**MỤC LỤC**

[**LỜI CẢM ƠN** 2](#_Toc516678751)

[**LỜI NÓI ĐẦU** 3](#_Toc516678752)

[**CHƯƠNG I: GIỚI THIỆU** 4](#_Toc516678753)

[**CHƯƠNG II: XÂY DỰNG ỨNG DỤNG CHẶN CUỘC GỌI** 9](#_Toc516678754)

[**2.1 Mục tiêu** 9](#_Toc516678755)

[**2.2 Xây dựng** 9](#_Toc516678756)

[**2.2.1 Xây dựng cơ sở dữ liệu (CSDL)** 9](#_Toc516678757)

[**2.2.2 Xây dựng giao diện** 10](#_Toc516678758)

[**a. Logo của ứng dụng** 10](#_Toc516678759)

[**b. Màn hình chào** 10](#_Toc516678760)

[**c. Dialog xin cấp quyền người dùng trước khi vào màn hình chính** 11](#_Toc516678761)

[**d. Màn hình chính** 12](#_Toc516678762)

[**e. Dialog thêm số bị chặn** 13](#_Toc516678763)

[**f. Màn hình thêm số bị chặn từ lịch sử cuộc gọi** 13](#_Toc516678764)

[**g. Màn hình thông báo** 14](#_Toc516678765)

[**CHƯƠNG III CÀI ĐẶT – THỬ NGHIỆM** 15](#_Toc516678766)

[**3.1 Cài đặt chương trình** 15](#_Toc516678767)

[**3.2 Hướng dẫn sử dụng** 15](#_Toc516678768)

[**3.3 Kết luận** 15](#_Toc516678769)

[**3.3.1 Về kiến thức** 15](#_Toc516678770)

[**3.3.2 Về chương trình** 15](#_Toc516678771)

[**3.3.3 Hạn chế** 16](#_Toc516678772)

[**3.3.4 Hướng phát triển** 16](#_Toc516678773)

[**PHỤ LỤC** 16](#_Toc516678774)

[**Tài liệu tham khảo** 16](#_Toc516678775)

# **LỜI CẢM ƠN**

Qua thời gian học tập ở lớp, chúng em đã đạt được những kiến thức cơ bản về bộ môn lập trình hệ thống và hoàn thành tiến độ dự kiến xây dựng ứng dụng Android chặn cuộc gọi. Để đạt được kết quả này, chúng em đã nỗ lực thực hiện và đồng thời cũng nhận được rất nhiều sự giúp đỡ, quan tâm, ủng hộ của các thầy cô, bạn bè và gia đình.

Chúng em xin chân thành cảm ơn thầy giáo Ngô Hoàng Huy đã tận tình hướng dẫn, giúp đỡ chúng em trong quá trình học tập.

Vì thời gian có hạn nên không thể tránh khỏi những thiếu sót, chúng em rất mong được sự đóng góp ý kiến từ thầy cô và các bạn.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

Đinh Văn Đông

Trần Thị Diệu Ninh

**LỜI NÓI ĐẦU**

Hiện nay, mạng điện thoại di động đã trở nên phổ biến trong đồi sống sinh hoat của con người. Điều này đã đáp ứng được nhu cầu liên lạc và trao đổi thông tin thông qua mạng di động của con người. Đi kèm theo sự phát triển mạng di động là sự xuất hiện các dịch vụ quảng cáo trên mobile như quảng cáo qua tin nhắn (tin nhắn rác) hay quảng cáo qua cuộc gọi (cuộc gọi rác),... gây phiền toái cho người sử dụng. Chính vì vậy, nhóm chúng em đã chọn đề tài xây dựng ứng dụng Chặn cuộc gọi trên thiết bị di động nền tảng Android để hạn chế phần nào những phiền toái đó.

Báo cáo được chia thành các chương sau:

* Chương 1: Giới thiệu
* Chương 2: Xây dựng ứng dụng chặn cuộc gọi
* Chương 3: Cài đặt – thử nghiệm ứng dụng

Do điều kiện thời gian và khả năng có hạn nên báo cáo này không thể tránh khỏi nhầm lẫn và thiếu sót, chúng em rất mong sự đóng góp ý kiến của thầy để báo cáo ngày càng hoàn thiện hơn.

**CHƯƠNG I: GIỚI THIỆU**

* 1. **Giới thiệu về hệ điều hành Android**
     1. **Mở đầu**

Như chúng ta biết, hiện tại đã có hơn nửa nhân loại sử dụng máy di động để thoại và giao tiếp qua các mạng không day. Con số 3 tỉ người này sẽ con tăng lên và máy di động ngày càng “thông minh” với nhiều chắc năng và dịch vụ hấp dẫn, cho nên thị trường máy di động thông minh sẽ vượt xa máy vi tính trong một tương lại rất gần… Vì thế, việc lập trình trên thiết bị di động ngày càng phổ biến và phát triển rất mạnh mẽ. Từ nền tảng mã nguồn mở, Google đã cho ra mắt Android chạy trên các thiết bị di động. Android có rất nhiều công cụ và dụng cụ miễn phí để nghiên cứu và phát triển phần mềm trên nền tảng của nó. Chương này sẽ giúp chúng ta tìm hiểu sơ bộ về Android và giới thiệu cho ta biết một ứng dụng Android đơn giản được tạo ra như thế nào.

* + 1. **Khái niệm về Android**

Trước hết, Android là 1 nền tảng phần mềm dựa trên mã nguồn mở Linux OS (Kernel 2.6) cho máy di động, máy tính bảng và những phần mềm trung gian (middleware) để hỗ trợ các ứng dụng mà người sử dụng cần đến. Một cách định nghĩa không quá chuyên môn thì có thể coi Android là tên một nền tảng mở cho thiết bị di động của Google (gồm hệ điều hành, middleware và một số ứng dụng cơ bản). Android sẽ đương đầu với một số hệ điều hành (viết tắt là HDH) dành cho thiết bị di động khác đang hâm nóng thị trường như Windows Mobile, Symbian và dĩ nhiên là cả OSX (iPhone).

Có thể nói một cách nôm na rằng Android là một HDH chạy trên thiết bị di động, cũng giống như Windows, Linux hay Mac chạy trên máy vi tính vậy.

* + - 1. **Android khác với các hệ điều chạy trên các thiết bị di động khác**

Android đã thu hút được sự chú ý của giới công nghệ khắp toàn cầu khi đứa con của Google sử dụng giấy phép mã nguồn mở. Đó là một sản phẩm kết tinh từ ý tưởng của Khối Liên minh thiết bị cầm tay mở do Google dẫn đầu, gồm 34 thành viên với các công ty hàng đầu về công nghệ và di động toàn cầu như Qualcomn, Intel, Motorola, Texas Instruments và LG Electronucs, các nhà mạng như T-Mobile, Sprint Nextel, NTT Docomo và China Mobile.

Các nhà phát triển có thể sử dụng miễn phí bộ Kit Android Software Development (SDK) để xây dựng các ứng dụng của mình.

* + - 1. **Đặc tính mở của Android**

Android được xây dựng để cho phép các nhà phát triển tạo ra các ứng dụng di động hấp dẫn, tận dụng tất cả tính năng một chiếc điện thoại đã cung cấp. Nó được xây dựng để được thực sự mở. Ví dụ, một ứng dụng có thể gọi bất kỳ chức năng lõi của điện thoại như thực hiện cuộc gọi, gửi tin nhắn văn bản hoặc bằng cách sử dụng máy ảnh, cho phép các nhà phát triển tạo ra nhiều ứng dụng phong phú hơn cho người dùng (điều này hiện chưa có trên Windows Phone7 của Microsoft). Android được xây dựng trên mã nguồn mở Linux Kernel. Hơn nữa, nó sử dụng một máy ảo tùy chỉnh được thiết kế để tối ưu hóa bộ nhớ và tài nguyên phần cứng trong một môi trường di động.

Android không phân biệt giữa các ứng dụng lõi của điện thoại và các ứng dụng của bên thứ 3. Tất cả có thể được xây dựng để có thể truy cập bằng khả năng của một thiết bị di động cung cấp cho người sử dụng với một dải rộng các ứng dụng và dịch vụ. Với các thiết bị xây dựng trên Android, người dùng có thể hoàn toàn thích ứng với điện thoại đến lợi ích của họ. Với Android, một nhà phát triển có thể xây dựng một ứng dụng cho phép người dùng xem vị trí của bạn bè của họ và được cảnh báo khi họ đang có trong vùng phụ cận cho họ một cơ hội để kết nối.

Android cung cấp truy cập đến một loạt các thư viện công cụ hữu ích và có thể được sử dụng để xây dựng các ứng dụng phong phú. Ví dụ, Android cho phép các thiệt bị giao tiếp với nhau tạo điều kiện cho đồng đẳng rich-to-peer trong xã hội. Ngoài ra, Android bao gồm một tập hợp đầy đủ công cụ đã được xây dựng công phu, với việc cung cấp nền tảng phát triển, với năng suất cao và cái nhìn sâu vào các ứng dụng.

* + 1. **Kiến trúc của Android**

Đôi khi không cần phải hiểu rõ kiến trúc của Android ta vẫn có thể viết được ứng dụng chạy trên nền HDH này. Đây cũng chính là điều mà nhà sản xuất muốn khi họ phát hành bộ SDK đi kèm với framework của họ. Và điều này có cả mặt tốt lẫn xấu. Framework nằm ở tầng cao cấp dành cho lập trình viên, giúp họ tạo ra các ứng dụng phổ biến nhưng rất khó dùng nó để tạo ra các ứng dụng đi sâu vào hệ thống của HDH. Tuy rằng đề tài nhóm chỉ dùng framework được cung cấp sẵn để viết ứng dụng nhưng chúng ta cũng nên tìm hiểu về cấu trúc của android để có cái nhìn toàn diện hơn về HDH này.

**1.1.3.1 Android Platform**

Bao gồm HDH Android đầy đủ tính năng, các ứng dụng và các tầng trung gian để developer có thể mở rộng, tùy chỉnh hoặc thêm vào các component của họ.

Có 4 tầng cơ bản trong HDH Android: Application Framework, Android Runtime, Native Libraries, Linux Kernel… Mỗi tầng làm việc đều nhờ sự giúp đỡ của tầng bên dưới.

**1.1.3.2 Tầng Linux Kernel**

Đây là nhân của HDH Android, mọi xử lý của hệ thống đều phải thông qua tầng này. Linux Kernel cung cấp các trình điều khiển thiết bị phần cứng (driver) như: camera, USB, wifi, Bluetooth, Display, Power Management...Android dựa trên Linux phiên bản 2.6 lựa chọn các tính năng cốt lõi như bảo mật, quản lý bộ nhớ, quản lý tiến trình, mạng stack và các tiến trình điều khiển phần cứng. Kernel hoạt động như một lớp trừu tượng giữa phần cứng và phần mềm còn lại của hệ thống.

**1.1.3.3 Native Libraries**

* **System C library** – có nguồn gốc từ hệ thống thư viện chuẩn C (lbc), điều chỉnh các thiết bị nhúng trên Linux.
* **Media Libraries** – mở rộng từ PacketVideo’s OpenCORE; thư viện hỗ trợ playback và recording của nhiều định dạng video và image phổ biến: MPEG4, H264, MP3, AAC, AMR, JPG và PNG.
* **Surface Manager** – quả lý việc hiển thị và kết hợp đồ họa 2D và 3D.
* **LibWebCore** – Android dùng lại webkit engine cho việc render trình duyệt mặc định của HDH Android browser và cho dạng web nhúng ( như HTML nhúng).
* **SGL** – 2D engine.
* **3D libraries** – Thư viện 3D dựa trên OpenGL ES 1.0 API, có nâng cấp tăng tốc “hardware 3D acceleration”.
* **FreeType** – render bitmap và vector font.
* **SQLite** – quản lý database của ứng dụng.

**1.1.3.4 Tầng Runtime**

Mỗi ứng dụng Android chạy trên một process riêng của Dalvik VM (máy ảo). Dalvik được viết để chạy nhiều máy ảo cùng một lúc một cách hiệu quả trên cùng một thiết bị. Máy ảo Dalvik thực thi các file mang định dạng .dex (Dalvik Excutable), định dạng này là định dạng đã được tối ưu hóa để chỉ chiếm một vùng nhớ vừa đủ xài và nhỏ nhất có thể. VM chạy các class (đã được compile trước đó bởi 1 trình biên dịch ngôn ngữ Java), sở dĩ VM chạy được các class này là nhờ chương trình DX tool đã convert các class sang định dạng .dex.

**1.1.3.5 Tầng Application Framework**

Đây là tầng mà Google xây dựng cho các developer để phát triển các ứng dụng của họ trên Android, chỉ bằng cách gọi API có sẵn mà Google đã viết để sử dụng các tính năng của phần cứng mà không cần hiểu cấu trúc bên dưới.

Bằng cách cung cấp một nền tảng phát triển mở, Android cho các nhà phát triển khả năng xây dựng các ứng dụng cực kỳ phong phú và sáng tạo. Nhà phát triển được tự do tận dụng các thiết bị phần cứng, thông tin địa điểm truy cập, các dịch vụ chạy nền, thiết lập hệ thống báo thức, thêm các thông báo đến các thanh trạng thái, và nhiều, nhiều hơn nữa.

Tất cả các ứng dụng thường gồm một bộ các dịch vụ và hế thống cơ bản sau:

**View UI** dùng để xây dựng layout của ứng dụng bao gồm: listview, text field, button, dialog, form…

**Content Providers** cho phép các ứng dụng có thể truy cập dữ liệu từ các ứng dụng khác ( như ứng dụng của ta có thể lấy thông tin Contacts của điện thoại Android) hoặc để chia sẻ dữ liệu của riêng ứng dụng.

**Resource Manager** cung cấp cách thức truy cập đến non-code resources như các asset, graphic, image, music, video…

**Notification Manager** cho phép tất cả các ứng dụng hiển thị thông báo của mình trên HDH.

**Activity Manager** quản lý vòng đời của các ứng dụng.

Ở góc nhìn của người dùng ta có thêm tầng application (là ứng dụng do chúng ta viết), sau đây là sơ đồ tổng quát:



*Hình 1: Tổng quan kiến trúc Android*

* + 1. **Máy ảo Dalvik**

Một trong những thành phần quan trọng của Android là máy ảo Dalvik. Thay vì sử dụng máy ảo Java như trước kia Android sử dụng máy ảo của riêng nó được thiết kế để đảm bảo rằng đa ứng dụng có thể chạy mượt mà trên một thiết bị di động.

Máy ảo Dalvik sử dụng lõi Linux để xử lý các chức năng ở mức thấp bao gồm bảo mật, các tiến trình, các luồng và quản lý vùng nhớ. Nó cũng có thể viết bằng ứng dụng C/C++ để chạy trực tiếp lên lõi Linux bên dưới. Giữa phần cứng và các dịch vụ hệ thống được quản lý bởi máy ảo Dalvik, nó là một thành phần ở giữa. Bằng cách sử dụng máy ảo này để chạy ứng dụng, các nhà phát triển hoàn toàn không phải bận tâm gì về các phần cứng bên dưới. Khi máy ảo này chạy nó sẽ tạo ra tập tin có đuôi là .dex, tập tin này được SDK tạo ra bởi sự chuyển đổi từ các lớp biên dịch ngôn ngữ Java.

* 1. **Cài đặt môi trường cho lập trình Android**
     1. **Cài đặt**

Trước khi xây dựng được ứng dụng Chặn cuộc gọi (Block Call), chúng ta cần phải hoàn thành các cài đặt sau:

+Android Studio (https://developer.android.com/studio/index.html)

+Java Development Kit (JDK) (http://www.oracle.com/technetwork/java/javase/downloads/index.html)

+Android SDK

Hiện nay, Android SDK đã được tích hợp sẵn trong Android Studio nên chúng ta không cần cài riêng rẽ.

* + 1. **Ngôn ngữ lập trình**

Để xây dựng ứng dụng Android, nhà phát triển có thể sử dụng nhiều ngôn ngữ lập trình khác nhau như: C/C++, Kotlin, Java… Trong đồ án này, chúng em đã lựa chọn ngôn ngữ lập trình Java để phát triển ứng dụng.

**CHƯƠNG II: XÂY DỰNG ỨNG DỤNG CHẶN CUỘC GỌI**

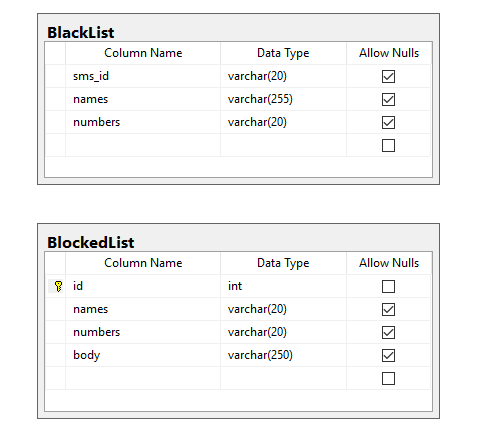
**2.1 Mục tiêu**

- Xây dựng được ứng dụng chặn cuộc gọi với chức năng chặn các cuộc gọi không mong muốn, gây phiền toái cho người dùng.

- Phương thức hoạt động: Các số điện thoại được đưa vào danh sách chặn, khi gọi đến, điện thoại sẽ không đổ chuông hay có bất kì một thông báo gì tới người dùng. Thay vào đó, thông tin về cuộc gọi đó sẽ được lưu vào ứng dụng. Nếu người dùng muốn kiểm tra xem liệu có cuộc gọi nào bị chặn thì sẽ xem trong ứng dụng này.

**2.2 Xây dựng**

**2.2.1 Xây dựng cơ sở dữ liệu (CSDL)**

****

CSDL gồm 2 bảng

BlackList chứa danh sách số điện thoại bị chặn.

**"create table IF NOT EXISTS "** + ***TABLE\_BLACK\_LIST*** + **"(sms\_id varchar(20),"** + **" names varchar(255),"** + **" numbers varchar(20) UNIQUE)"**

BlockedList chứa lịch sử gọi của số điện thoại bị chặn.

**"create table IF NOT EXISTS "** + ***TABLE\_BLOCKED\_LIST*** + **"(id integer primary key autoincrement,"** + **" names varchar(20),"** + **" numbers varchar(20),"** + **" body varchar(250))"**

**2.2.2 Xây dựng giao diện**

**a. Logo của ứng dụng**

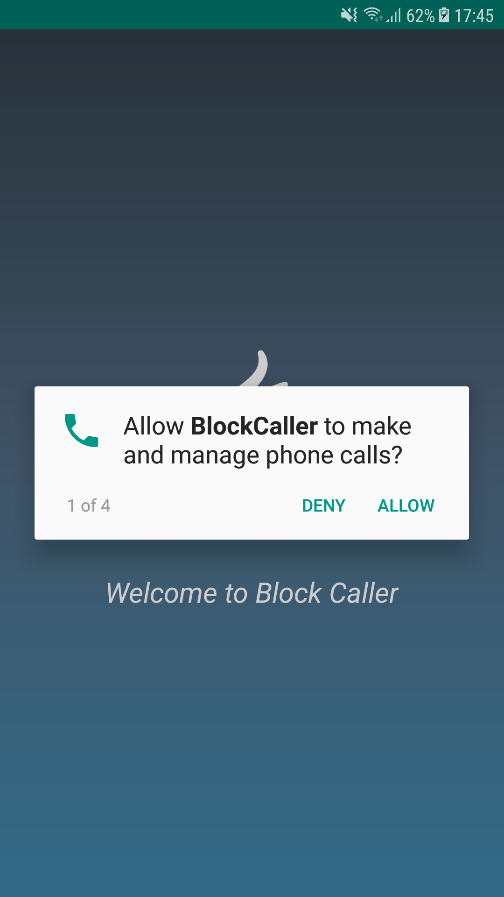
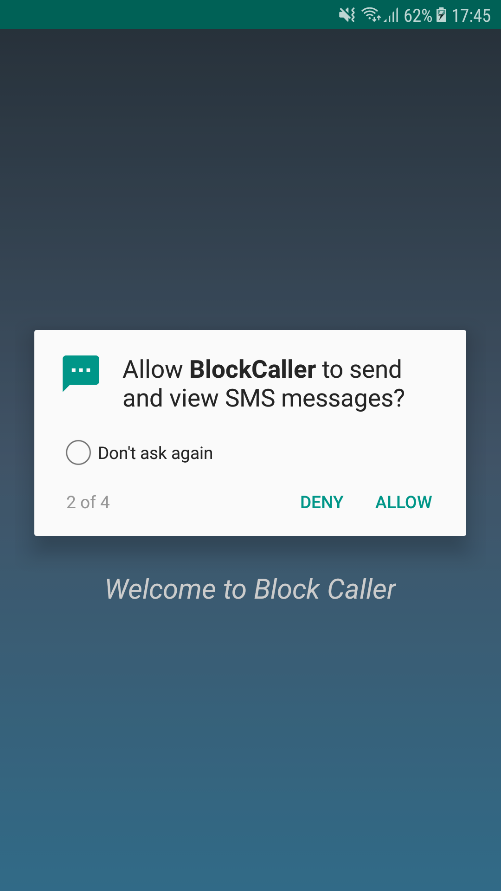
****

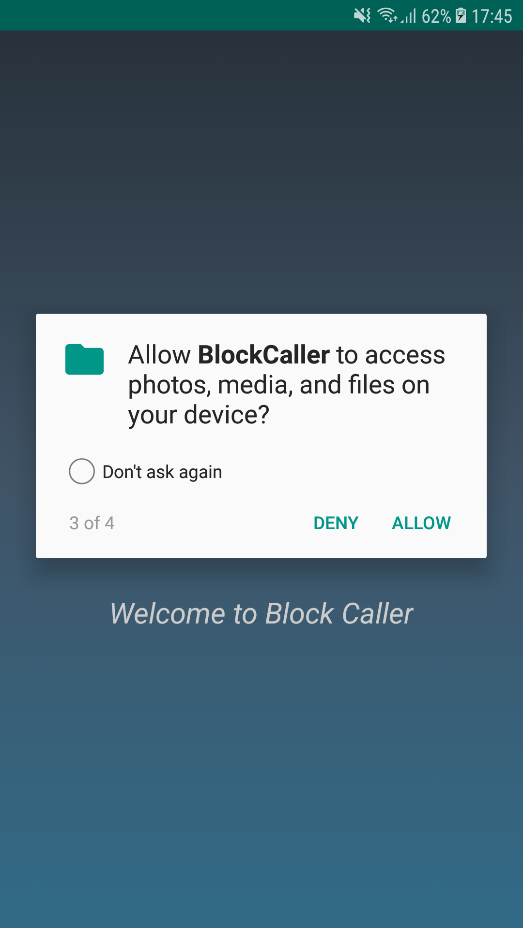
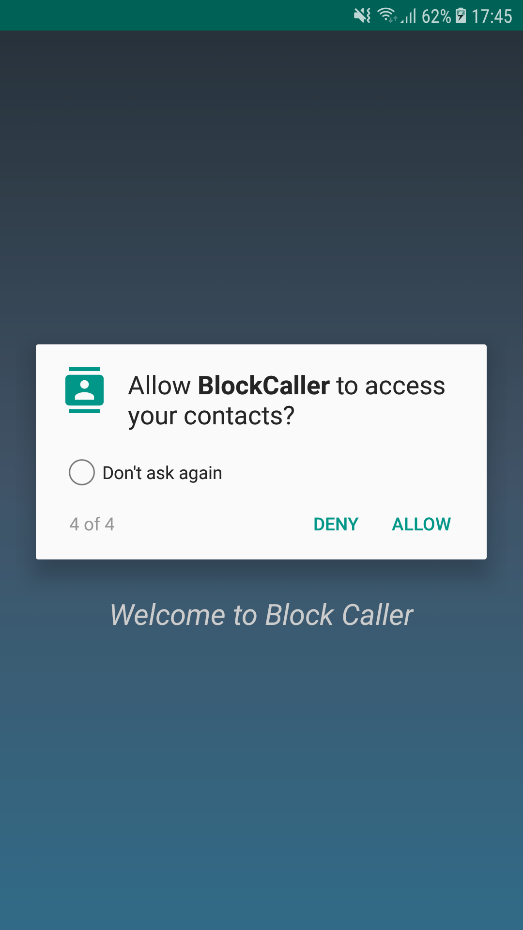
## **b. Màn hình chào**



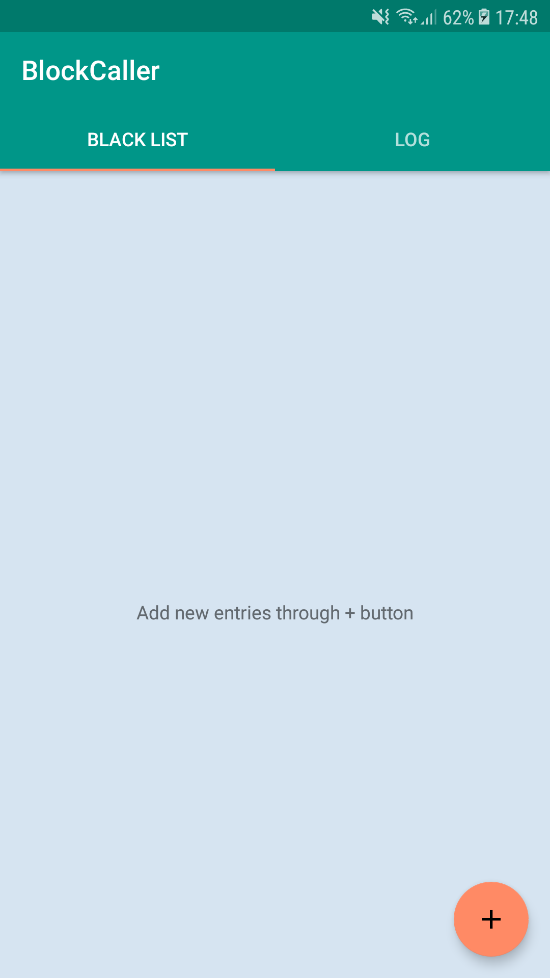
## **c. Dialog xin cấp quyền người dùng trước khi vào màn hình chính**

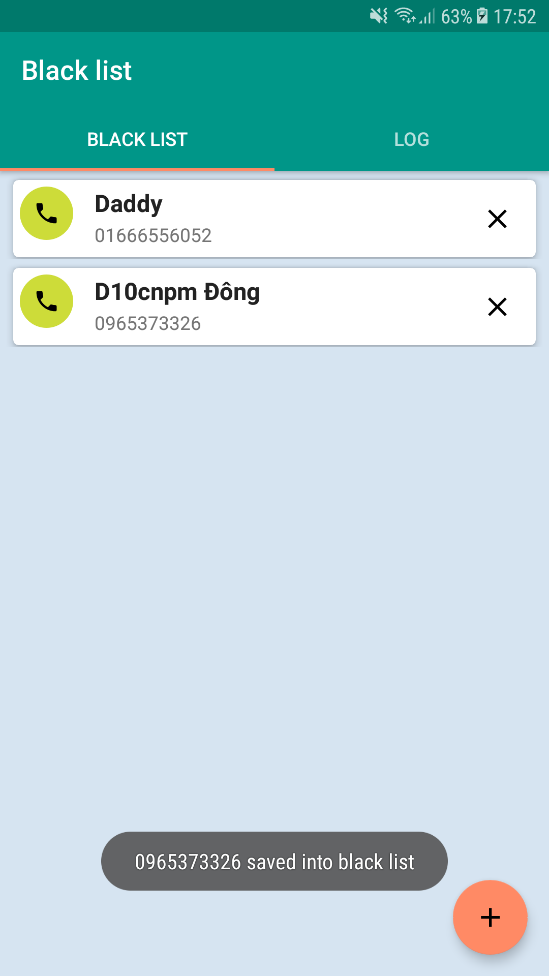
Do vấn đề bảo mật nên từ Android 6 ứng dụng phải xin người dùng cấp quyền để có thể truy cập vào tài nguyên máy tính nếu cần.

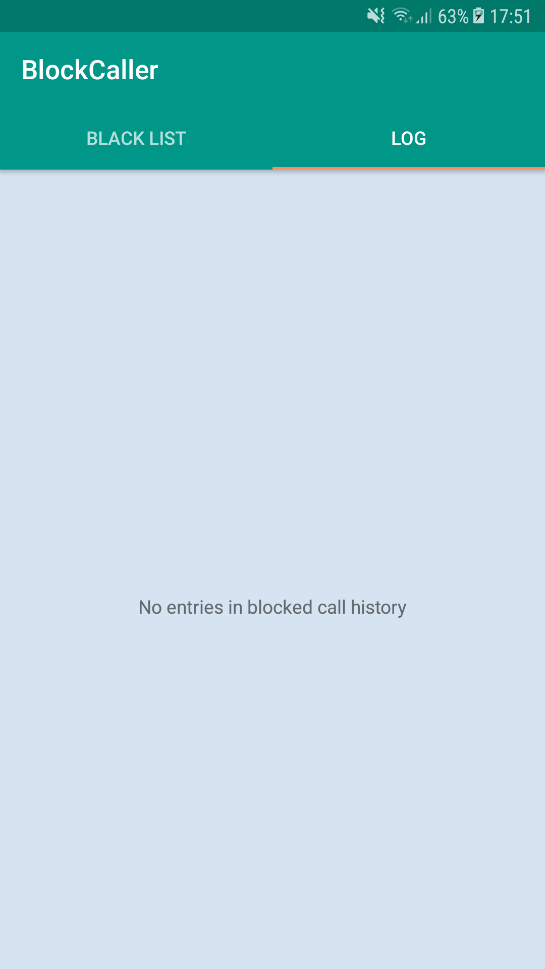
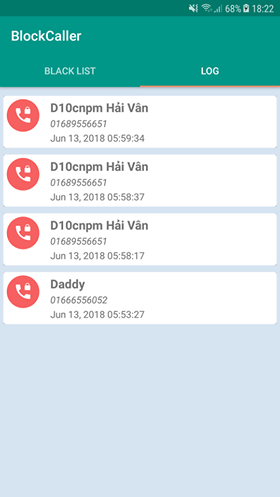




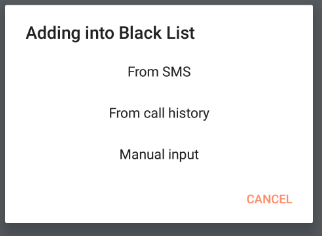
## **d. Màn hình chính**

****

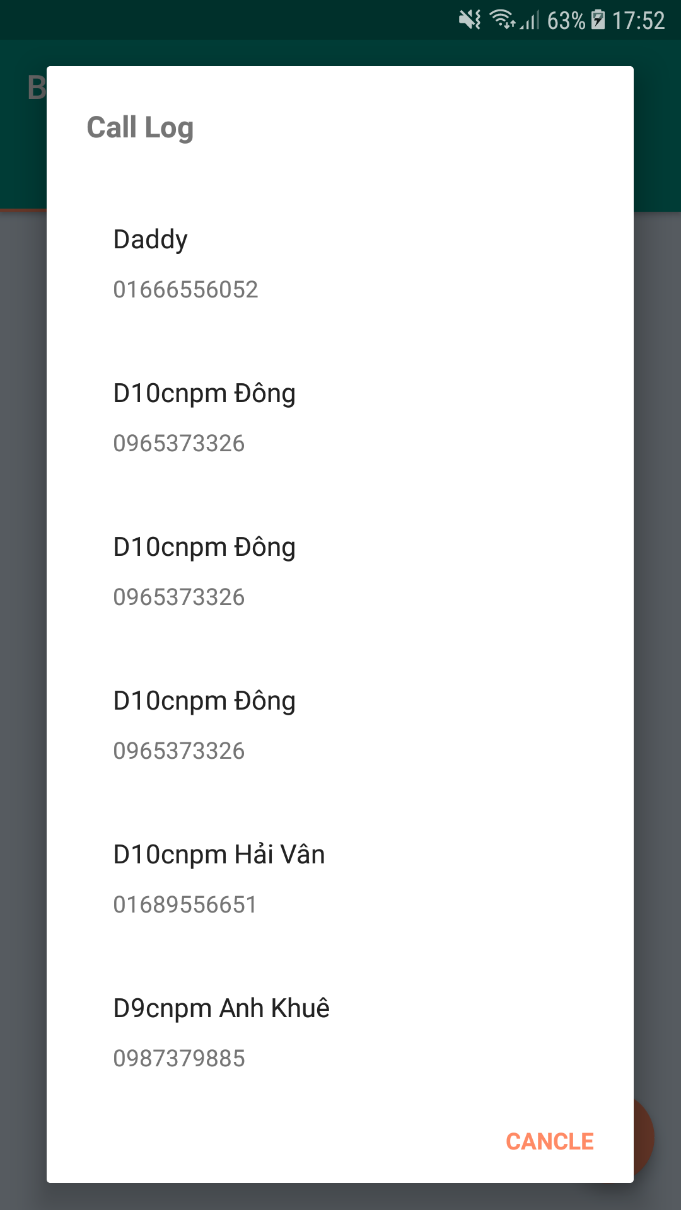
****

****

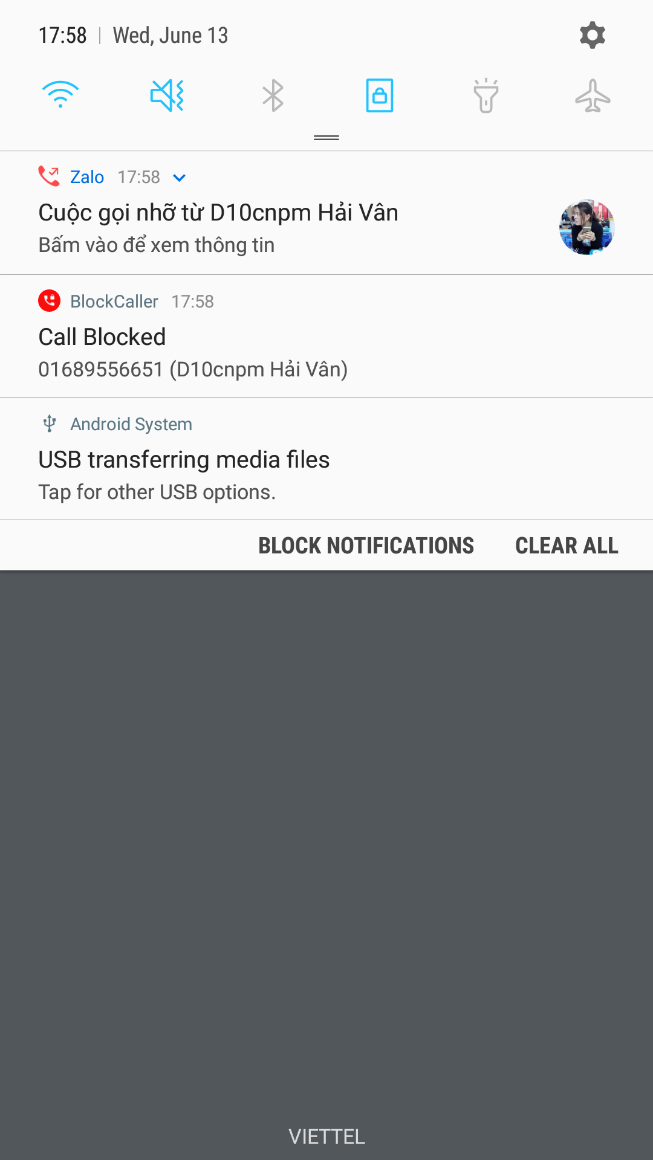
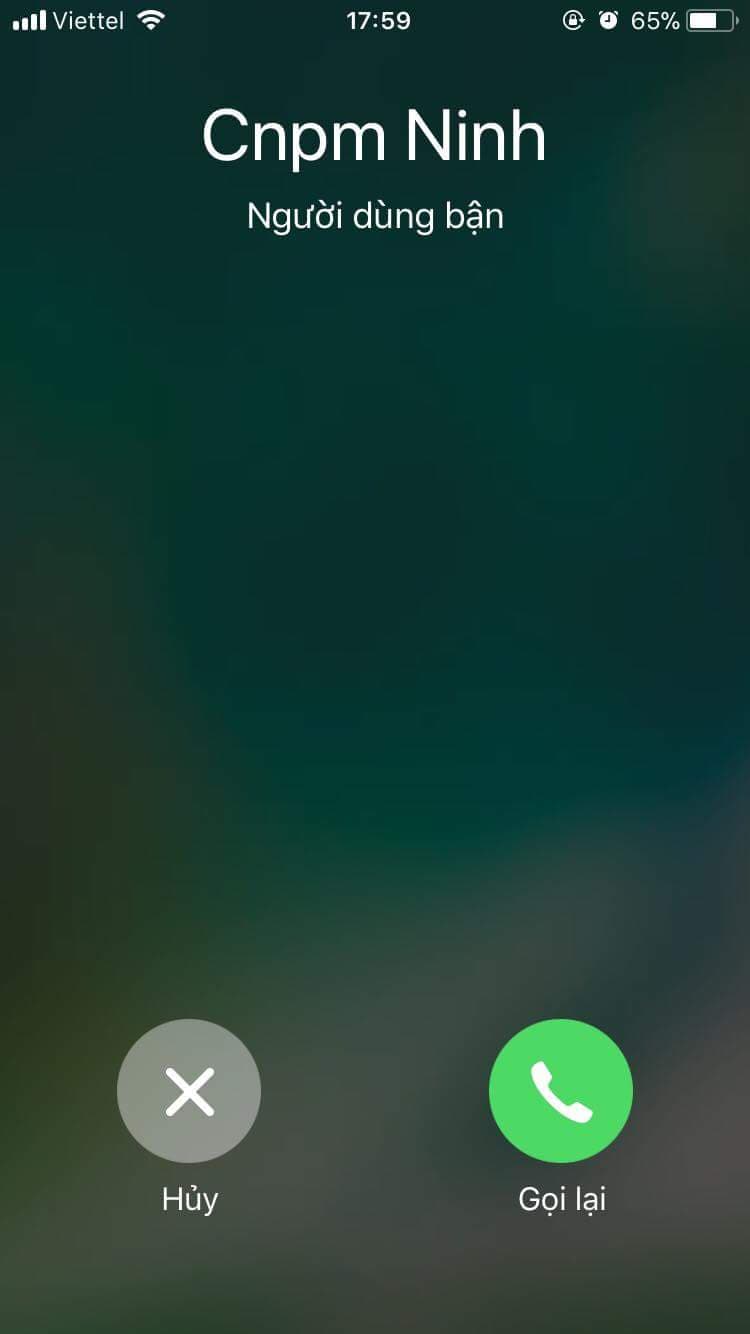
## **e. Dialog thêm số bị chặn**

****

## **f. Màn hình thêm số bị chặn từ lịch sử cuộc gọi**

****

## **g. Màn hình thông báo**

****

**CHƯƠNG III CÀI ĐẶT – THỬ NGHIỆM**

**3.1 Cài đặt chương trình**

Mã nguồn: https://github.com/dieuninh1997/BlockCaller

**3.2 Hướng dẫn sử dụng**

**-**Người dùng mở ứng dụng BlockCaller qua shortcut trên màn hình điện thoại.

- Ứng dụng yêu cầu người dùng cho phép một số quyền như:

* Make and manage phone calls
* Send and view SMS messages
* Access photos, media, and files on your device
* Access your contacts

Nếu người dùng không cấp đủ quyền mà ứng dụng yêu cầu thì ứng dụng sẽ hỏi lại cho đến khi được cấp đủ quyền và sau đó mới cho người dùng vào màn hình chính của ứng dụng.

- Tại tab BlackList, theo hướng dẫn trên màn hình, để thêm số điện thoại cần chăn, người dùng chạm vào nút “+” góc phải màn hình. Ứng dụng sẽ hiển thị hộp thoại “Adding into Black List”. Người dùng chọn mục 2 trong hộp thoại “From call history”. Danh sách lịch sử cuộc gọi hiện ra. Người dùng vuốt danh sách để chọn số cần chặn. Click chọn số. Số vừa chọn sẽ được lưu vào local của điện thoại.

- Khi có cuộc gọi đến, ứng dụng sẽ so khớp số điện thoại gọi đến với BlackList đã lưu. Nếu tồn tại số này trong BlackList, cuộc gọi đến sẽ bị chặn và ứng dụng hiển thị thông báo trên thanh Notifications. Lịch sử cuộc gọi bị chặn sẽ được hiển thị tường ứng bên tab Log.

- Nếu muốn hủy chăn số điện thoại, người dùng click vào số điện thoại muốn hủy trong BlackList.

- Nếu muốn xóa lịch sử cuộc gọi bị chặn, người dùng click vào dòng lịch sử muốn xóa trong Log.

**3.3 Kết luận**

**3.3.1 Về kiến thức**

Sau thời gian thực hiện đề tài môn học, chúng em đã đạt được nhiều tiến bộ cả về mặt tìm hiểu, nghiên cứu lý thuyết lẫn kỹ năng lập trình. Có thể nói, thông qua đồ án môn học, chúng em đã đạt được:

-Hiểu biết nhiều hơn về các kỹ thuật và kỹ xảo trong lập trình Java

-Nâng cao tinh thần tự học, tự nghiên cứu.

-Hiểu hơn về hệ điều hành nguồn mở Android.

**3.3.2 Về chương trình**

- Ứng dụng chặn cuộc gọi đã phần nào xây dựng và đáp ứng được một số chức năng chính cần có.

- Giao diện thân thiện, dễ sử dụng.

**3.3.3 Hạn chế**

- Do thời gian thực hiện đồ án môn học là tương đối hạn chế so với đề tài tương đối rộng nên không tránh khỏi những thiếu sót nhất định. Bên cạnh đó, chương trình còn một số chức năng chưa hoàn thiện và chính xác.

- Do kinh nghiệm và kiến thức chưa nhiều, hơn nữa đây là một ứng dụng di động nên việc tối ưu mã nguồn để tinh tế sử dụng tài nguyên bộ nhớ còn chưa tốt.

**3.3.4 Hướng phát triển**

**-** Để tiếp tục phát đề tài này và có thể áp dụng nhiều trong thực tế, chúng em nhận thấy cần phải tiếp tục thực hiện một số công việc như sau:

**+** Xử lý các lỗi chặt chẽ hơn trước khi đưa vào sử dụng.

**+** Bổ sung thêm chức năng chặn tin nhắn.

**PHỤ LỤC**

# **Tài liệu tham khảo**

[1] <http://www.androidbegin.com/tutorial/automatic-call-blocking-using-android-telephony-tutorial/>

[2] https://developer.android.com