**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**



**VIỆN CÔNG NGHỆ THÔNG TIN – TRUYỀN THÔNG**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

****

**BÁO CÁO MÔN HỌC  
Lập trình Android**

**ĐỀ TÀI**

**Xây dựng phần mềm danh bạ trên Android**

Sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| **Họ và tên:** | **MSSV** | **Lớp** |
| **Nguyễn Tất Hòa** | **20131536** | **CNTT2.01** |
| **Nguyễn Ngọc Đôn** | **20130941** | **CNTT2.03** |

**Giảng viên hướng dẫn: TS. Trần Hải Anh**Hà Nội, tháng 12 năm 2015

**Mục Lục**

[**LỜI MỞ ĐẦU** 3](#_Toc437137305)

[**Phần 1. Phân tích và thiết kế hệ thống.** 4](#_Toc437137306)

[**1.** **Đặc tả bài toán.** 4](#_Toc437137307)

[**2.** **Đặc tả chức năng.** 4](#_Toc437137308)

[2.1. Upload phone book 4](#_Toc437137309)

[2.2. Restore phone book. 5](#_Toc437137310)

[2.3. Check duplicate phone book 5](#_Toc437137311)

[2.4. Export contact tới SD Card 5](#_Toc437137312)

[2.5. Import contact từ SD Card 5](#_Toc437137313)

[**3.** **Biểu đồ Activity.** 5](#_Toc437137314)

[3.1. Export contact tới SD Card 5](#_Toc437137315)

[3.2. Upload phone book 5](#_Toc437137316)

[3.3. Restore phone book 6](#_Toc437137320)

[3.4. Import contact từ SD Card 6](#_Toc437137321)

[**Phần 2: Thiết kế chương trình.** 7](#_Toc437137322)

[**1.** **Giao diện.** 7](#_Toc437137323)

[1.1. Activity. 7](#_Toc437137324)

[1.2. Thư mục Drawable. 7](#_Toc437137325)

[1.3. Thư mục values. 7](#_Toc437137326)

[**2.** **Code** 8](#_Toc437137327)

[2.1. Các file java. 8](#_Toc437137328)

[2.2. File build.gradle. 8](#_Toc437137329)

[2.3. File manifests.xml. 8](#_Toc437137330)

[**Phần 3. Một số hình ảnh Demo** 8](#_Toc437137331)

[**Phần 4. Tài liệu tham khảo** 9](#_Toc437137333)

# **LỜI MỞ ĐẦU**

Ngày nay, công nghệ thông tin ngày càng phát triển và đã thâm nhập vào mọi lĩnh vực của đời sống kinh tế, văn hóa, xã hội. Điện thoại thông minh là một trong những thiết bị thiết yếu của mỗi người. Đó như là một thiết bị để mọi người có thể liên lạc với nhau mỗi khi có khoảng cách về địa lý. Mỗi một người không phải chỉ có một người bạn cần liên lạc, họ có bạn bè, người thân, khách hàng,… Do đó mà danh bạ liên lạc của họ có thể có chứa hàng nghìn số, và nếu một ngày nào đó không may điện thoại của họ bị hỏng mà không thể sửa vậy thì làm sao mà họ có thể lấy lại được những danh bạ trước đó của họ.  
 Nắm bắt được điều đó nhóm chúng em quyết định xây dựng một phần mềm để có thể giải quyết được điều đó, phần mềm sẽ giúp sao lưu lại danh bạ và đồng bộ chúng lên các dịch vụ đám mây.

Và theo thống kê thì trên thế giới có khoảng 80% thiết bị chạy hệ điều hành Android trong tổng số các thiết bị di động thông minh. Vì thế nền tảng Android sẽ là lựa chọn của chúng em để xây dựng phần mềm này.

Chúng em xin trân thành cảm ơn thầy giáo: Trần Hải Anh đã giảng giải và giúp đỡ chúng em củng cố kiến thức trong quá trình xây dựng phần mềm này.

**Nhóm sinh viên thực hiện:**

Nguyễn Tất Hòa

Nguyễn Ngọc Đôn

## **Phần 1. Phân tích và thiết kế hệ thống.**

1. **Đặc tả bài toán.**

Đối với mọi chiếc điện thoại đi động Contact là ứng dụng không thể thiếu, dùng để quản lý danh bạ điện thoại, thông tin cá nhân cần liên lạc trong danh bạ như Email, địa chỉ, sinh nhật,…Trong Android phiên bản mới nhất hiện nay cũng đã có hỗ trợ ứng dụng như Contact với nhiều chức năng cơ bản giống các ứng dụng thông thường khác trên các điện thoại di động. Ngoài ra, còn có một số chức năng nổi trội như:

* Hỗ trợ kết nối với ứng dụng Map có thể đinh vị người cần liên lạc thông qua địa chỉ được lưu trong Contact.
* Cho phép lưu trữ nhiều số điện thoại, email, address của người dùng trong Contact.

Mặc dù vậy, ứng dụng contact của Android đã cung cấp nhiều chức năng nhưng ứng dụng này vẫn còn nhiều hạn chế, đặc biết thiếu chức năng sao lưu, khôi phục và kiểm tra các danh bạ trùng lặp trên chính những danh bạ của người dùng. Ngoài ra, với một số người sử dụng thì thông tin lưu trong contact cũng rất quan trọng. Đề phòng mất điện thoại hay tránh để người khác truy cập những thông tin khác được lưu trong điện thoại hoặc khi bị mất các contact do tác động ngoại cảnh hoặc khi thay đổi số thuê bao. Do đó, ứng dụng contact mà chúng em phát triển dựa trên nền tảng Android này sẽ giúp cải tiến một số hạn chế của ứng dụng contact gốc trong Android nhằm giúp người dùng có thể yên tâm contact của họ được bảo vệ an toàn.

1. **Đặc tả chức năng.**
   1. **Upload phone book** 
      * **Mô tả chức năng**Chức năng upload phone book cho phép người dùng lưu trữ toàn bộ sổ contact của họ vào tài khoản online. Tuy nhiên một số thông tin ít sử dụng sẽ được lược bỏ nhằm giảm kích thước của nội dung XML khi đóng gói
      * **Mô tả hoạt động**
        + Đọc tất cả các thông tin trong phone book và lưu trữ vào một list các contacts.
        + Phân tích list các contact đó thành nội dung XML dựa vào kĩ thuật Xstream
        + Kết nối các tới webservice và gửi nội dung XML kèm theo nội dung tài khoản người dùng cần thiết (Dropbox, Google drive)
        + Webservice lưu nội dung XML vào đúng tài khoản người dùng
        + Trích xuất thông báo đã lưu khi đã xong hoặc không được lưu.
   2. **Restore phone book.**
      * **Mô tả chức năng:**

Cho phép người dùng lấy lại các danh bạ đã sao lưu lại trước đó từ (Dropbox, Google drive) tệp XML.

* + - **Mô tả hoạt động:**
      * Người dùng lựa chọn tài khoản đã upload phone book trước đó.
      * Ứng dụng gửi đi thông tin và tải về tệp XML đã upload
      * Ứng dụng nhận dữ liệu đã tải về từ Dropbox hoặc Google drive và phân tích nội dung XML thành một list contacts.
      * Cảnh báo người dùng có thể thay thế các contact đang có trong điện thoại.
      * Ghi các dữ liệu trong phiên bản lấy về vào điện thoại.
  1. **Check duplicate phone book**
     + **Mô tả chức năng:**

Kiểm tra trùng và xóa các danh bạ trùng lặp.

* + - **Mô tả hoạt động:**
      * Người dùng nhấn nút đọc danh bạ sẵn có trong máy.
      * Ứng dụng đọc và kiểm tra các danh bạ trùng lặp số điện thoại và xóa đi nếu có.
      * Hiển thị lại tất cả danh bạ một cách chi tiết.
  1. **Export** **contact tới SD Card**
     + **Mô tả chức năng:**

Chứcnăng export to an SD card cho phép người dùng chia sẻ hoặc backup của họ vào thẻ nhớ dưới dạng XML. Khi người dùng cần chia sẻ họ có thể trao đổi thông qua thẻ nhớ hoặc có thể gửi file XML này đi bất kì đâu và người nhận có thể phân tích file này thành các danh bạ như cũ

* 1. **Import contact từ SD Card**
     + **Mô tả chức năng:**

Ngược lại với chức năng export to SD Card thì chức năng này đưỡ dùng để phân tích ngược lại từ một file XML thành các contact.

1. **Biểu đồ Activity.**

* 1. **Export** **contact tới SD Card**

Convert to XML

Export contact to SD Card

Write XML to SD Card

Show message

Read contact

### 

### **Upload phone book**

Upload phone book

Read contacts

Convert to XML

## ***Send to***

Dropbox

Google drive

Show message

* 1. **Restore phone book**

Restore phone book using ContentProvider

Read file in SD Card

Save file to SD Card

Google drive

Dropbox

## ***Download from***

Restore phone book

* 1. **Import contact từ SD Card**

Import contact from SD Card

Read file

***Read* *done Read* *error***

Convert to contact list

Show message

Write to phone book

## **Phần 2: Thiết kế chương trình.**

1. **Giao diện.**
   1. Activity.
      * home\_layout.xml: mô tả giao diện chính của chương trình với bốn Button: contacts, restore, back up và exit.
      * view\_contact\_layout.xml: thể hiện chi tiết danh bạ của bạn bè.
      * row.xml: Hiển thị listview đơn giản của danh bạ có trong điện thoại.
      * server\_comunicate\_layout.xml: hiển thị lựa chọn tài khoản để back up, restore danh bạ.
   2. Thư mục Drawable.
      * blue\_button.xml: Chỉnh sửa giao diện cho các button khác với mặc định.
      * left\_arrow.png: thay đổi nút trờ về thành hình ảnh ở activity view\_contact\_layout.
   3. Thư mục values.
      * Strings.xml: Bao gồm các dòng text định nghĩa sẵn để tiện cho việc sửa đổi.

<string name="btnUploadToGoogleDrive">Google Drive</string>

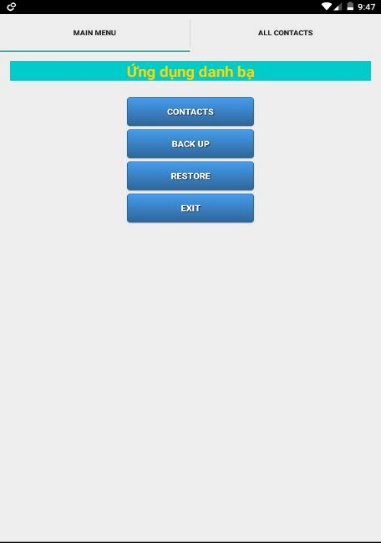
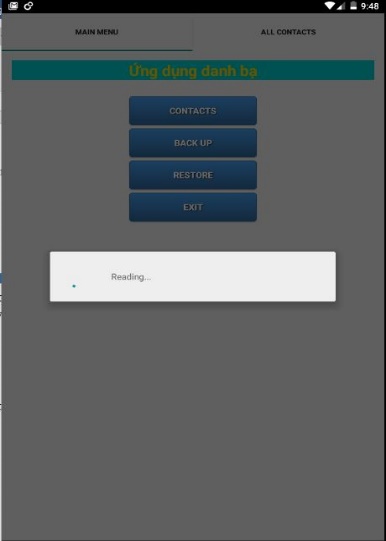
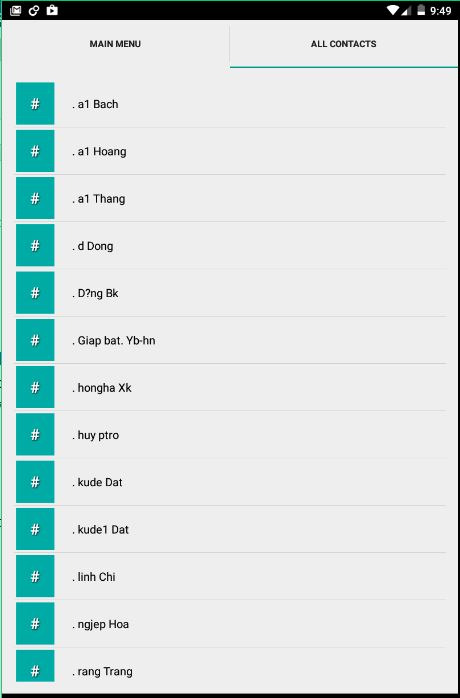
* + - colors.xml: Chứa các màu sắc cho các đoạn text, background.

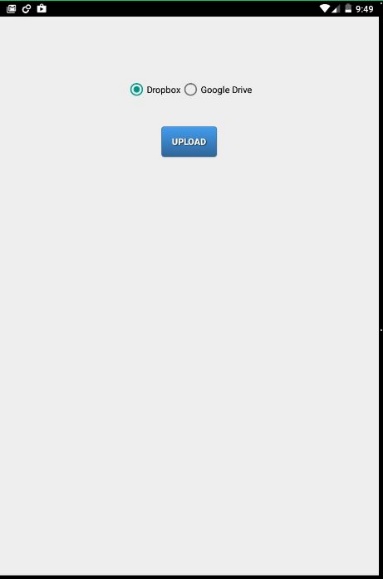
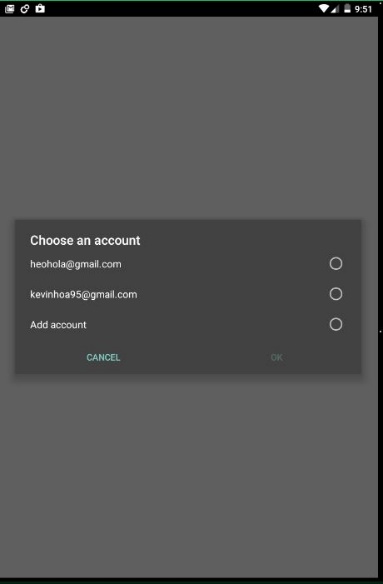
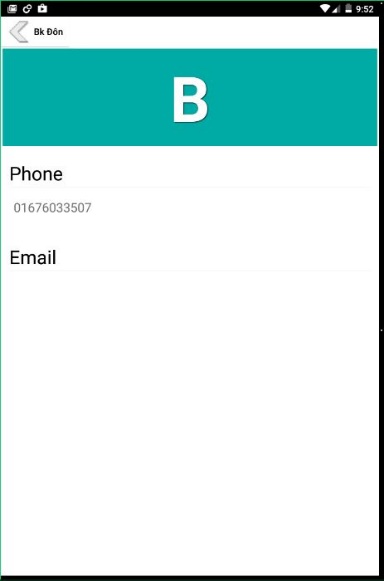
<color name="tv\_danhba">#00CCCC</color>

* + - styles.xml: Định nghĩa style cho các button.

1. **Code**
   1. Các file java.
      * **Contact.java**: Định nghĩa cấu trúc danh bạ người dùng được sử dụng trong ứng dụng. Bao gồm các phương thức getter(), setter(), toString() và đặc biết là phương thức **equals(object oc)** phục vụ cho kiểm tra trùng lặp danh bạ.
      * **DropboxActivity.java, GoogleActivity.java**: sử dụng Android Api để tạo kết nối tới tài khoản người dùng từ đó tạo thao tác download và upload file xml nhằm chức năng back up và restore danh bạ cho người dùng ứng dụng. Thông qua các phương thức cơ bản mà đã được nhà phát triển định nghĩa.
      * **Information.java**: Chứa các hằng là từ khóa để code cho thuận tiện và nhanh chóng. Ngoài ra còn chứa app\_key để có thể kết nối được tới Dropbox.
      * **ServerComunicateActivity.java**: kế thừa từ lớp Activity với chức năng để xử lý việc back up cũng như restore danh bạ từ file XML.
      * **ViewContactActivity.java:** Xử lý hiển thị chi tiết danh bạ người dùng với số điện thoại, Email và tên.
      * **MainActivity.java**: Bao gồm các hàm với chức năng đọc danh bạ trên điện thoại, kiểm tra, xử lý trùng lặp rồi cuối cùng hiển thị danh sách danh bạ lên listview.
   2. File build.gradle.
      * Thêm các thư viện java cần thiết để sử dụng kết nối tới Google drive api.
   3. File manifests.xml.
      * Cung cấp các quyền truy cập internet, quyền đọc ghi thẻ nhớ, danh bạ trên điện thoại,…
      * Khai báo các Activity sử dụng trong ứng dụng, tên của ứng dụng.

## **Phần 3. Một số hình ảnh Demo**

**  **

## **Phần 4. Tài liệu tham khảo**

Trong quá trình tìm hiểu và xây dựng ứng dụng chúng em có tham khảo slide bài giảng của thầy giáo Trần Hải Anh và một số trang web hướng dẫn như:

* <https://developers.google.com/android>
* <https://www.dropbox.com/developers-v1/sync/start/android>
* <https://myclass.vn>

cùng với một số trang web tìm kiếm khác.

* Link source code: [**https://gitlab.com/ngocdon0127/ContactsIT3660**](https://gitlab.com/ngocdon0127/ContactsIT3660)