TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

Đồ Án 3:Định Hướng Công Nghệ

Phát triển ứng dụng đọc mã vạch QRCode

trên nền tảng android

Sinh viên thực hiện: **Đỗ Khắc Hùng**

Lớp CN-CNTT02

Giáo viên hướng dẫn: ThS. **Nguyễn Đức Tiến**

HÀ NỘI 05 -2018

# Phiếu giao nhiệm vụ đồ án

## 1.Thông tin sinh viên

Họ và tên :Đỗ Khắc Hùng

Điện thoại liên lạc:0963806075 Email:khachungbg97@gmail.com

Lớp:CN-CNTT02 Hệ đào tạo:chính quy

Thời gian thực hiện:Từ ngày 12 tháng 1 đến ngày 1 tháng 5 năm 2018

## 2.Mục đích nội dung của đồ án:

-Tìm hiểu về mã QRCode

-Tìm hiểu việc lập trình ứng dụng trên hệ điều hành android

-Xây dựng ứng dụng cho phép đọc mã QRCode của sản phẩm và lưu trữ thông tin trên firebase.

## 3.Các nhiệm vụ cụ thể:

-Tìm hiểu việc lập trình ứng dụng cho thiết bị hệ điều hành android

-Tìm hiểu thư viện mã nguồn mở Zxing

-Đăng nhập ứng dụng bằng tài khoản google và lưu trữ cơ sở dữ liệu trên firebase.

-Xuất dữ liệu ra file cvs.

-Ứng dụng cho phép log hoạt động của người dùng(sử dụng thư viện log4j)

## 4.Lời cam kết của sinh viên

-Tôi-Đỗ Khắc Hùng –cam kết đồ án là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *Ths Nguyễn Đức Tiến* .

-Các kết quả nêu trong đồ án là trung thực và sao chép toàn văn của bất cứ công trình nghiên cứu nào.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 23 tháng 05 năm 2018*  Tác giả ĐA  *Đỗ Khắc Hùng* |

## 5.Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành và cho phép bảo vệ

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 23 tháng 05 năm 2018*  Giáo viên hướng dẫn  *ThS Nguyễn Đức Tiến* |

# 

# LỜI CẢM ƠN

Để có thể hoàn thành đồ án tốt nghiệp này, em xin gửi lời cảm ơn chân thành nhất tới tập thể các thầy giáo, cô giáo trường Đại học Bách Khoa Hà Nội nói chung, viện Công nghệ thông tin và truyền thông, bộ môn Kỹ thuật máy tính, đã đào tạo cho em những kiến thức, những kinh nghiệm quý báu trong suốt thời gian học tập và rèn luyện.

Em xin gửi lời cảm ơn tới thầy giáo, ThS. Nguyễn Đức Tiến - Giảng viên bộ môn Kỹ thuật máy tính, viện Công nghệ thông tin và truyền thông, trường Đại học Bách Khoa Hà Nội đã tận tình hướng dẫn em trong quá trình làm đồ án tốt nghiệp.

Cuối cùng là lời cảm ơn chân thành nhất tới gia đình, bạn bè đã luôn ở bên động viên, góp ý và tạo mọi điều kiện tốt nhất để tôi có thể hoàn thành đồ án tốt nghiệp này.

Tuy nhiên, do thời gian và kiến thức có hạn nên đồ án này chắc chắn không tránh khỏi những thiếu sót, em rất mong được sự đóng góp ý kiến của các thầy, các cô và toàn thể các bạn. Em xin chân thành cảm ơn.

Hà Nội ngày 23 tháng 5 năm 2018

Sinh viên: **Đỗ Khắc Hùng**

# TÓM TẮT NỘI DUNG ĐỒ ÁN

Mục tiêu của đồ án là xây dựng ứng dụng trên nền tảng android có thể quét mã QRCode của các sản phẩm trong các xưởng sản xuất.

Nội dung chính của đồ án gồm các phần:

**Chương 1:Công nghệ và nền tảng phát triển ứng dụng**

Chương này tâp trung đề cập đến các nền tảng và công nghệ phát triển ứng dụng của đồ án,gồm các phần chính:

-Tìm hiểu công nghệ lập trình ứng dụng cho android.

-Tìm hiểu các loại mã vạch UPC,Code 39,Code 128,QRCode.

-Tìm hiểu về thư viện mã nguồn mở nhận dạng Zxing trên android

-Tìm hiểu FireBase và sử dụng firebase trong ứng dụng android.

-Tìm hiểu về file cvs.

**Chương 2:Phân tích thiết kế hệ thống.**

Chương này tập trung vào việc khảo sát ,phân tích và thiết kế ứng dụng đọc mã vạch QRCode trên android:

-Phần đầu giới thiệu,khảo sát hệ thống.

-Phần giữa tập trung vào việc phân tích ,đặc tả các chức năng của hệ thông.

-Phần cuối tập trung vào thiết kế hệ thống bao gồm:kiến trúc hệ thống,thiết kế lớp,thiết kế cơ sở dữ liệu và thiết kế mã nguồn chương trình

**Chương 3:Xây dựng và phát triển hệ thống.**

* + Phần đầu của chương này sẽ tập trung mô tả việc xây dựng hệ thống trên môi trường nào, bằng công cụ gì…
  + Phần tiếp theo của chương sẽ mô tả giao diện của chương trình, kết quả kiểm thử và cách thức triển khai, phân phối ứng dụng đến tay người dùng.

**Chương 4:Tổng kết và đánh giá.**

* Chương này tập trung vào việc đánh giá kết quả đạt được, đưa ra phương hướng phát triển cho hệ thống sau khi kết thúc đồ án và phần kết luận.

# LỜI NÓI ĐẦU:

Ngày nay, việc ứng dụng mã vạch đã trở nên phổ biến đối với mọi người kể cả trên thế giới và Việt Nam. Mỗi khi đi siêu thị chúng ta vẫn thấy người thu ngân dùng một thiết bị quét mã vạch in trên sản phẩm giúp cho việc thanh toán dễ dàng hơn. Đấy mới chỉ là một ứng dụng nhỏ của việc sử dụng mã vạch vào đời sống thường ngày.

Cùng với sự phát triển của công nghệ, mã vạch đã cải tiến không chỉ còn đơn giản là lưu mã của sản phẩm mà còn có khả năng lưu được nhiều thông tin của sản phẩm hơn với việc sử dụng mã vạch 2 chiều.

Một lợi thế của ứng dụng mã vạch là khả năng tiện dụng, chúng ta có thể sử dụng tại bất kỳ đâu, bất kỳ khi nào bằng việc sử dụng một thiết bị thông minh có hỗ trợ máy ảnh (ví dụ: điện thoại di động) có khả năng đọc mã vạch và giải mã mã vạch đó.

Đặt trong hoàn cảnh giữa hai thiết bị Android không có sự bất kỳ kết nối nào (không Bluetooth, không Wi-Fi) mà chúng lại muốn chia sẽ dữ liệu cho nhau. Đồ án sẽ đưa ra một giải pháp cho việc lưu trữ, chia sẻ thông tin này dựa trên mã vạch 2 chiều QR kết hợp với nền tảng Google Android − một nền tảng di động mới đầy hứa hẹn và là đối thủ xứng tầm của iPhone.

Đồ án sẽ tập trung nghiên cứu về các nền tảng di động, các đặc điểm của chúng, đi sâu nghiên cứu nền tảng FireBase và công nghệ mã vạch hiện nay, cuối cùng là ứng dụng vào giải quyết bài toán phát triển ứng dụng lưu trữ thông tin bằng mã vạch

# Mục Lục:

[Phiếu giao nhiệm vụ đồ án 1](#_Toc512810795)

[1.Thông tin sinh viên 1](#_Toc512810796)

[2.Mục đích nội dung của đồ án: 1](#_Toc512810797)

[3.Các nhiệm vụ cụ thể: 1](#_Toc512810798)

[4.Lời cam kết của sinh viên 1](#_Toc512810799)

[5.Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành và cho phép bảo vệ 1](#_Toc512810800)

[LỜI CẢM ƠN 2](#_Toc512810801)

[TÓM TẮT NỘI DUNG ĐỒ ÁN 3](#_Toc512810802)

[LỜI NÓI ĐẦU: 4](#_Toc512810803)

[Mục Lục: 5](#_Toc512810804)

[Danh Mục Hình Vẽ 7](#_Toc512810805)

[Danh Mục Các Từ Viết Tắt Và Thuật Ngữ 8](#_Toc512810806)

[Phần 1:Nền Tảng Và Công nghệ Phát Triển Ứng Dụng 9](#_Toc512810807)

[I.1Tổng quan 9](#_Toc512810808)

[I.1.1Các vấn đề ,khó khăn hiện tại 9](#_Toc512810809)

[I.1.2 Mục tiêu cần đạt được 9](#_Toc512810810)

[I.1.3 Lựa chọn và định hướng thiết kế . 9](#_Toc512810811)

[Phần 2:Phân tích và thiết kế hệ thống 12](#_Toc512810812)

[II.1 Mô hình tổng thể của các thiết bị trong hệ thống 12](#_Toc512810813)

[II.2 Mô hình triển khai phần mềm 12](#_Toc512810814)

[II.2.1 Sử dụng thư viện Zxing quét mã vạch trên android 12](#_Toc512810815)

[II.2.2 Tích hợp Firebase vào ứng dụng để lưu trữ dữ liệu và trợ giúp đăng nhập tài khoản google và facebook. 13](#_Toc512810816)

[II.2.3 Cách tích hợp log4j để ghi lại hoạt động người dùng. 13](#_Toc512810817)

[II.3 Phân tích 13](#_Toc512810818)

[II.3.1 Các chức năng của hệ thống 13](#_Toc512810819)

[II.3.2 Biểu đồ use case 14](#_Toc512810820)

[II.3.3 Biểu đồ tuần tự 15](#_Toc512810821)

[II.4 Thiết kế 15](#_Toc512810822)

[Phần 3:Cài đặt và triển khai giải pháp 18](#_Toc512810823)

[III.1 Áp dụng thư viện Zxing quét mã vạch trên hệ thống. 18](#_Toc512810824)

[III.2 Kết nối ứng dụng với firebase 20](#_Toc512810825)

[III.3 Sử dụng Log4j trong android 20](#_Toc512810826)

[III.4 Giao diện kết quả của ứng dụng 22](#_Toc512810827)

[Tài liệu tham khảo 24](#_Toc512810828)

# 

# Danh Mục Hình Vẽ

Hình 0 :Các loại mã vạch(9)

Hình 1: Mo hình các giao diện chính và kiến trúc hệ thống(11)

Hình 2: Use case của hệ thống(13)

Hình 3: Biểu đồ trình tự của hệ thống(14)

Hình 4 :Biểu đồ lớp của hệ thống(15)

Hình 5 :Màn hình đăng nhập(21)

Hình 6:Popup thông báo (21)

Hình 7 :Màn hình của Tab Scan code(22)

Hình 8 :Màn hình Tab hiển thị sản phẩm(22)

Hình 9:Màn hình Tab Settings.(22)

# Danh Mục Các Từ Viết Tắt Và Thuật Ngữ

|  |  |  |
| --- | --- | --- |
| **Ký hiệu** | **Giải thích** | |
| **Tiếng Anh** | **Tiếng Việt** |
| **API** | Application Programming Interface | Giao diện lập trình ứng dụng |
| **QR code** | Quick response code | Mã phản hồi nhanh |
| **Zxing** | Zebra crossing | Thư viện để quét mã vạch 1 và 2 chiều. |
|  |  |  |

# Phần 1:Nền Tảng Và Công nghệ Phát Triển Ứng Dụng

Trong chương này sẽ trình bày các nền tảng và công nghệ trong việc xây dựng và phát triển ứng dụng.Các vấn đề này sẽ được trình bày gồm:công nghệ lập trình ứng dụng trên android.

## I.1Tổng quan

### I.1.1Các vấn đề ,khó khăn hiện tại

Trong các nhà máy ,xưởng sản xuất các loại mặt hàng hoặc nhiều loại linh kiện hay đồ khác nhau nằm ở các vị trí khác nhau rất khó để có thể nhớ các thông số kĩ thuật của chúng như tên,khối lượng,vị trí,…...Vì vậy cần có thứ gì đó để xác định thông tin của mỗi linh kiện đó,theo đó người ta sử dụng mã QRCode trên mỗi sản phâm,linh kiện .

### I.1.2 Mục tiêu cần đạt được

-Ứng dụng sau khi hoàn thành sẽ đáp ứng sẽ giúp cho:

+ việc quản lý các chi tiết một cách dễ dàng và thuận tiện.

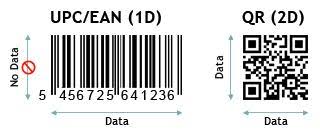
+Tăng thời gian xử lý đọc thông tin của từng sản phẩm.

+Dữ liệu được lưu giữ dễ dàng mà độ bảo mật thông tin cao.

+Giao diện thân tiện ,dễ sử dụng.

### I.1.3 Lựa chọn và định hướng thiết kế .

Ứng dụng được phát triển trên nảng tảng android có thể quét được các loại mã vạch như UPC, Code39, Code 128,QR Code sử dụng thư viện Zxing.



Hình 0 :Các loại mã vạch

QR Code, viết tắt của Quick response code (tạm dịch "Mã phản hồi nhanh") hay còn gọi là mã vạch ma trận (matrix-barcode) là dạng mã vạch hai chiều (2D) có thể được đọc bởi một máy đọc mã vạch hay smartphone (điện thoại thông minh) có chức năng chụp ảnh (camera) với ứng dụng chuyên biệt để quét mã vạch

* Barcode theo định nghĩa là phương pháp lưu trữ và truyền tải thông tin bằng một lọai ký hiệu gọi là ký mã vạch (Barcode symbology). Ký mã vạch hay gọi tắt cũng là mã vạch, là 1 ký hiệu tổ hợp các khoảng trắng và vạch thẳng để biểu diễn các mẫu tự, ký hiệu và các con số. Sự thay đổi trong độ rộng của vạch và khoảng trắng biểu diễn thông tin số hay chữ số dưới dạng mà máy có thể đọc được.
* Zxing (viết tắt của “zebra crossing”) là một thư viện mã nguồn mở, xử lý nhiều định dạng mã vạch 1 chiều và 2 chiều, được cài đặt bằng Java. Mục đích của thư viện này là sử dụng máy ảnh trên điện thoại di động để chụp và giải mã các mã vạch trên thiết bị, không cần phải kết nối với máy chủ. hiện tại thư viện hỗ trợ các định dạng mã vạch sau: UPC, Code39, Code 128,QR Code,…

Việc lưu trữ dữ liệu cần được sử dụng phổ biến trên các công ty,đam bảo tính an toàn và dễ dàng quản lý đối với quản trị viên nên sử dụng trên SQL Server vì nó được phát triển bởi microsoft.Có thể phát triển thêm hệ thống để dữ liệu có thể lưu trữ trên firebase,tại sao lại vậy?

**Firebase** là một dịch vụ hệ thống backend được Google cung cấp sẵn cho ứng dụng Mobile của bạn, với Firebase bạn có thể rút ngắn thời gian phát triển, triển khai và thời gian mở rộng quy mô của ứng dụng mobile mình đang phát triển. Hỗ trợ cả 2 nền tảng Android và IOS, Firebase mạnh mẽ, đa năng, bảo mật và là dịch vụ cần thiết đầu tiên để xây dưng ứng dụng với hàng triệu người sử dụng. Sử dụng Firebase bạn sẽ có được hưởng các lợi ích sau:

* Xây dựng ứng dụng nhanh chóng mà không tốn thời gian, nhân lực để quản lý hệ thống và cơ sơ sở hạ tầng phía sau: Firebase cung cấp cho bạn chức năng như phân tích, cơ sở dữ liệu, báo cáo hoạt động và báo cáo các sự cố lỗi để bạn có thể dễ dàng phát triển, định hướng ứng dụng của mình vào người sử dụng nhằm đem lại các trải nghiệm tốt nhất cho họ.
* Uy tín chất lượng đảm bảo từ Google: Firebase được google hỗ trợ và cung cấp trên nền tảng phần cứng với quy mô rộng khắp thế giới, được các tập đoàn lớn và các ưng dụng với triệu lượt sử dụng từ người dùng.
* Quản lý cấu hình và trải nghiệm các ứng dụng của Firebase tập trung trong một giao diện website đơn giản, các ứng dụng này hoạt động độc lập nhưng liên kết dữ liệu phân tích chặt chẽ.

Một trong những tính năng của ứng dụng nó có thể giúp người dùng có thể sử dụng tài khoản google và facebook để đăng nhập vào ứng dụng khi không muốn đăng nhập bằng tài khoản riêng.Việc tích hợp 2 loại tài khoản này vào hệ thống giúp nó thân thiện với người dùng và là 1 giải pháp khi người dùng quên tài khoản nào đó trong 3 loại tài khoản.Để tích hợp 2 loại tài khoản này vào hệ thống thì firebase đã hỗ trợ.

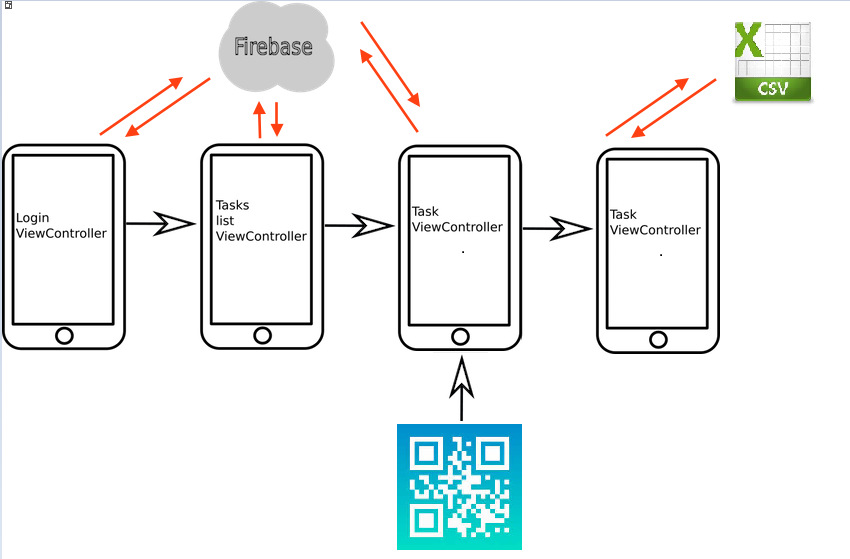
Ngoài ra để đảm bảo tính an toàn hơn nữa hệ thống có hỗ trợ giúp việc xuất dữ liệu ra file csv,file csv có thể sẽ là báo cáo giúp các bên dễ trao đổi .

Ứng dụng cho phép ghi lại các hoạt động người dùng như thời gian,hoạt động bằng việc tích hợp log4j.

# Phần 2:Phân tích và thiết kế hệ thống

## II.1 Mô hình tổng thể của các thiết bị trong hệ thống

--



Hình 1 Mo hình các giao diện chính và kiến trúc hệ thống

Ứng dụng quét thông tin của sản phẩm thông qua mã vạch hiển thị cho người dùng,thông tin sẽ lưu trên FireBase,ngoài ra để thuận tiện ,người dùng có thể lưu vào file csv.

## II.2 Mô hình triển khai phần mềm

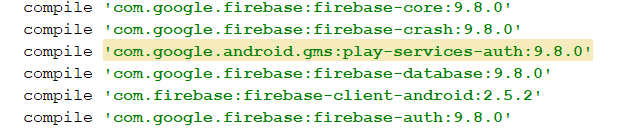
### II.2.1 Sử dụng thư viện Zxing quét mã vạch trên android

Mã vạch của sản phẩm của sản phẩm được định dạng theo các thuộc tính như thông tin người thực hiện quét ,mã sản phẩm để định danh mang tính duy nhất của sản phẩm,vị trí đặt của sản phẩm.

Để phân tách các thuộc tính đó thường sử dụng dấu cách hoặc dấu phẩy .

### II.2.2 Tích hợp Firebase vào ứng dụng để lưu trữ dữ liệu và trợ giúp đăng nhập tài khoản google và facebook.

-Thêm vào ứng dụng các thư viện sau để phục vụ đăng nhập và tích hợp realtime database của firebase.



### II.2.3 Cách tích hợp log4j để ghi lại hoạt động người dùng.

Khi người dùng thực hiện đăng nhập thành công thì ngay trong onCreate() ta gọi lại phương thức ghi lại thời gian (giờ,phút,giây,ngày,tháng,năm) và các hành động của người dùng vào file trong bộ nhớ của thiết bị .

Thư viện hỗ trợ ghi log trong android gồm :



## II.3 Phân tích

### II.3.1 Các chức năng của hệ thống

-Đăng nhập

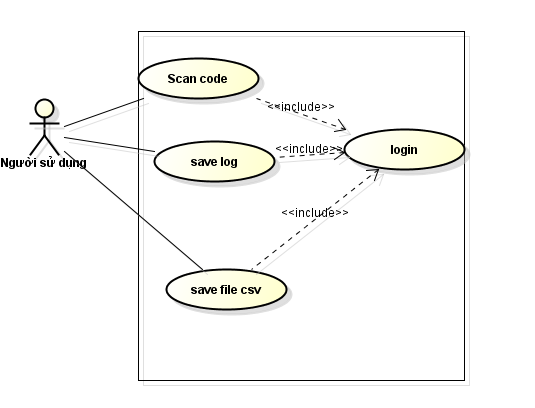
-Quét mã vạch (lưu vào cơ sở dữ liệu,lấy dữ liệu về ứng dụng,..)

-Ghi ra file csv

-Ghi log hoạt động người dùng.

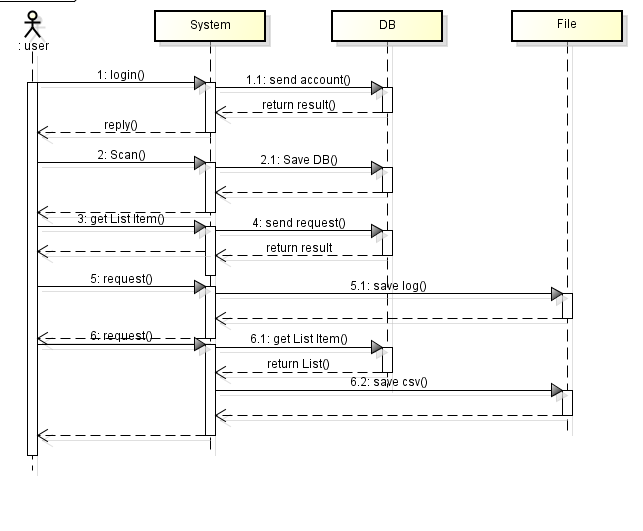
-…

### II.3.2 Biểu đồ use case



Hình 2 Use case của hệ thống

### II.3.3 Biểu đồ tuần tự

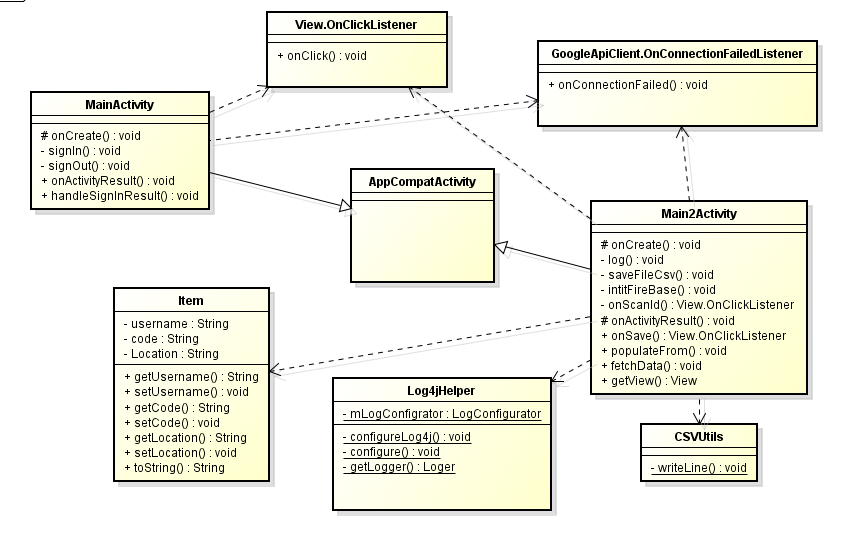


Hình 3: Biểu đồ trình tự của hệ thống

## II.4 Thiết kế

Dựa trên các phân tích trên từng ca sử dụng phần trên, ta có có thể thiết kế các lớp của hệ thống như sau:

*Các lớp MainAcivity(màn hình login), Main2Acitity, Item ,CSVUtils,Log4jHelper và lớp dẫn xuất của lớp cơ sở AppcompatActivity.*



Hình 4 :Biểu đồ lớp của hệ thống

**Lớp Main2Acivity**: tạo giao diện chính của chương trình liên kết với các chức năng khác của ứng dụng như quét mã vạch,hiển thị danh sách thông tin các đối tượng ,ghi vào file csv,ghi lại hoạt động của người dùng.

**Phương thức:**

**onCreate()** là phương thức Overide của lớp *AppcompatActivity*, được gọi khi lớp này khởi tạo.

**Log()** là phương thức được gọi trong onCreate(),trong lớp này gọi đến lớp Log4jHelper() để thực hiện ghi lại hoạt động của người dùng vào filelog.txt

**saveFileCsv()**:là phương thức gọi khi người dùng click vào button SaveCSV ,ghi lại hoạt động gồm :thời gian như giờ ,phút, giây,ngày,tháng,năm,..

**onScanId():**bắt sự kiện của một nút và gọi tới Intent của Barcode Scanner để quét mã QR.

**onActivityResult():** bắt sự kiện trả về kết quả khi Barcode Scanner quét xong. Nếu quét thành công mã resultCode sẽ là RESULT\_OK, nếu Intent bị hủy thì mã resultCode sẽ là RESULT\_CANCELED.Sau đó tách mã kết quả theo các trường đã xác định theo dấu “,” rồi hiển thị ra EditText.

**intitFireBase()** là phương thức khởi tạo kết nối của app với FireBase để phục vụ việc lưu trữ,và trong onCreate gọi hàm này để mỗi khi mở app dữ liệu có thể hiển thị trên listView.

**onSave():** bắt sự khi người dùng nhấn vào button Save dữ liệu trên các dòng EditText lưu vào cơ sở dữ liệu đồng thời update bên Tab hiển thị listview.

**Lớp MainAcivity**: tạo giao diện chính đăng nhập của chương trình có 2 button login và logout.Thực hiện kiểm tra đăng nhập dựa trên FireBase.

**Phương thức:**

**onCreate()** là phương thức Overide của lớp *AppcompatActivity*, được gọi khi lớp này khởi tạo.

**signIn()** là được gọi khi người nhấn vào login để mở 1 popup hiển thị danh sách các tài khoản đã đăng nhập hoặc thêm mới 1 tài khoản ,sử dung GoogleSignInApi để phục vụ việc này.

**signOut()**  giúp thoát tài khoản google của người dùng ra khỏi app.

**onActivityResult()**  thực hiện bắt sự kiện trả về khi người dùng gọi hàm signIn() hoặc signOut().Nó thực hiện mã requestCode trả về có bằng RC\_SIGN\_IN

,nếu đúng sẽ gọi startActivity() để chuyển sang Main2Activity

**Lớp Log4jHelper**:để cấu hình định dạng của file log.

# Phần 3:Cài đặt và triển khai giải pháp

## III.1 Áp dụng thư viện Zxing quét mã vạch trên hệ thống.

Android đã hỗ trợ thư viện Zxing rất tốt ta chỉ cần thêm thư viện vào project.

**private** View.OnClickListener **onScanName** = **new** View.OnClickListener() {  
 **public void** onClick(View v) {  
 **if** (v.getId() == R.id.***name***) {  
 IntentIntegrator integrator = **new** IntentIntegrator(Main2Activity.**this**);  
 integrator.setDesiredBarcodeFormats(IntentIntegrator.***QR\_CODE\_TYPES***);  
 integrator.setPrompt(**"Scan"**);  
 integrator.setCameraId(0);  
 integrator.setBeepEnabled(**false**);  
 integrator.setBarcodeImageEnabled(**false**);  
 **check** = 1;  
 integrator.initiateScan();  
  
 }  
 }  
};

Đoạn mã trên bắt sự kiện của một nút và gọi tới Intent của Barcode Scanner để quét mã QR.

**public** **void** onActivityResult(**int** requestCode, **int**  resultCode, Intent intent) {

**if** (requestCode == 0) {

**if** (resultCode == *RESULT\_OK*) {

String contents =

intent.getStringExtra("SCAN\_RESULT");

String format =

intent.getStringExtra("SCAN\_RESULT\_FORMAT");

// Handle successful scan

} **else** **if**(resultCode==*RESULT\_CANCELED*)

{

// Handle cancel

}

}

}

}

Đoạn mã trên bắt sự kiện trả về kết quả khi Barcode Scanner quét xong. Nếu quét thành công mã resultCode sẽ là RESULT\_OK, nếu Intent bị hủy thì mã resultCode sẽ là RESULT\_CANCELED.

**Sử dụng lớp** **IntentIntegrator**

Lớp IntentIntegrator gọi Barcode Scanner rất đơn giản thông qua hàm sau:

IntentIntegrator.*initiateScan*(Activity);

Để lấy giá trị trả về, lớp IntentIntegrator có hàm parseActivityResult

**protected** **void** onActivityResult (**int** requestCode,

**int** resultCode, Intent data) {

**switch**(requestCode) {

**case** IntentIntegrator.REQUEST\_CODE:

**if** (resultCode == RESULT\_OK) {

IntentResult scanResult = IntentIntegrator.parseActivityResult(

requestCode,resultCode, data);

**if** (scanResult != **null**) {

String strContent=scanResult.getContents();

}

}

}

## III.2 Kết nối ứng dụng với firebase

Sau khi thêm thư viện ,ta khai báo trong Main2Activity

FirebaseDatabase **firebaseDatabase**;  
DatabaseReference **data**;  
FirebaseHelper **helper**;

Trong hàm onCreate() ta gọi phương thức sau để thưc hiện việc kết nối :

**private void** intitFireBase() {  
 FirebaseApp.*initializeApp*(Main2Activity.**this**);  
 **firebaseDatabase**=FirebaseDatabase.*getInstance*();  
 **data**=**firebaseDatabase**.getReference();  
 **helper**=**new** FirebaseHelper(**data**);  
 DatabaseReference ref=**data**.child(**"Item"**);  
 ref.addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 **for** (DataSnapshot chidSnap : dataSnapshot.getChildren()) {  
 Item spacecraft=chidSnap.getValue(Item.**class**);  
 *model*.add(spacecraft);  
 }  
 }  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
  
  
}

## III.3 Sử dụng Log4j trong android

Sau khi thêm các thư viện vào mục .lib

Thêm quyền trong manifest để sử dụng external storage

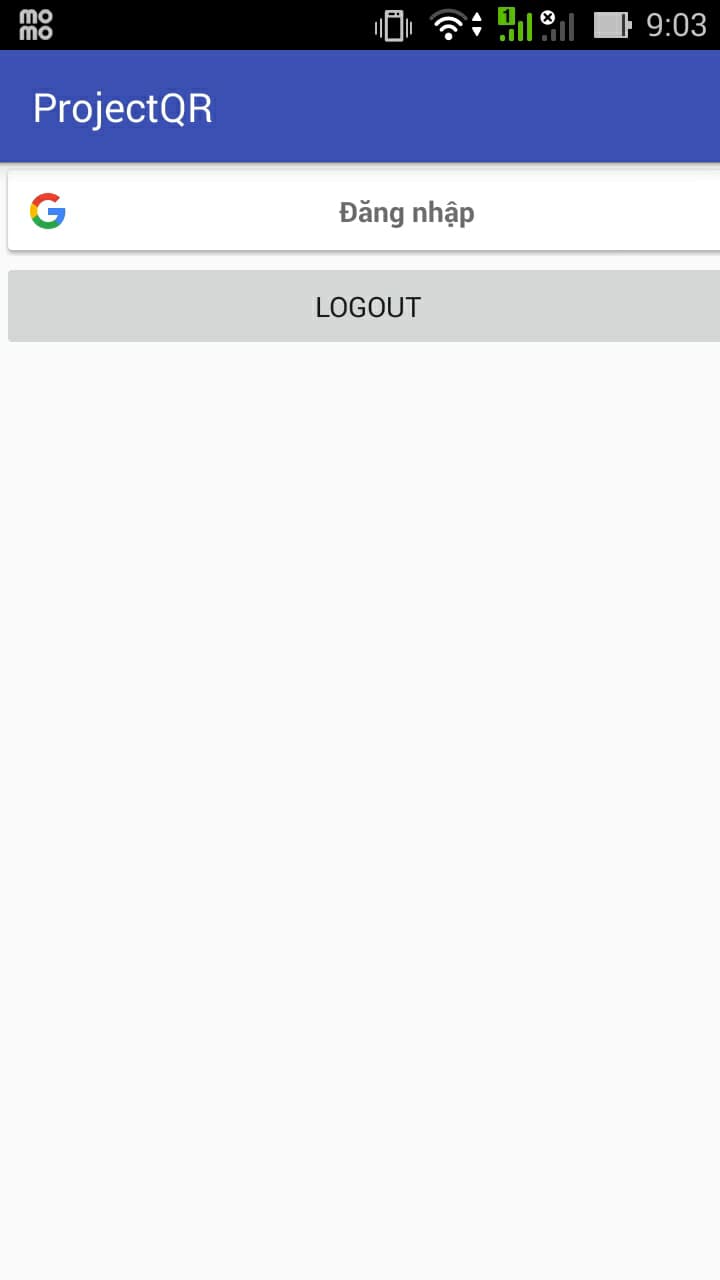
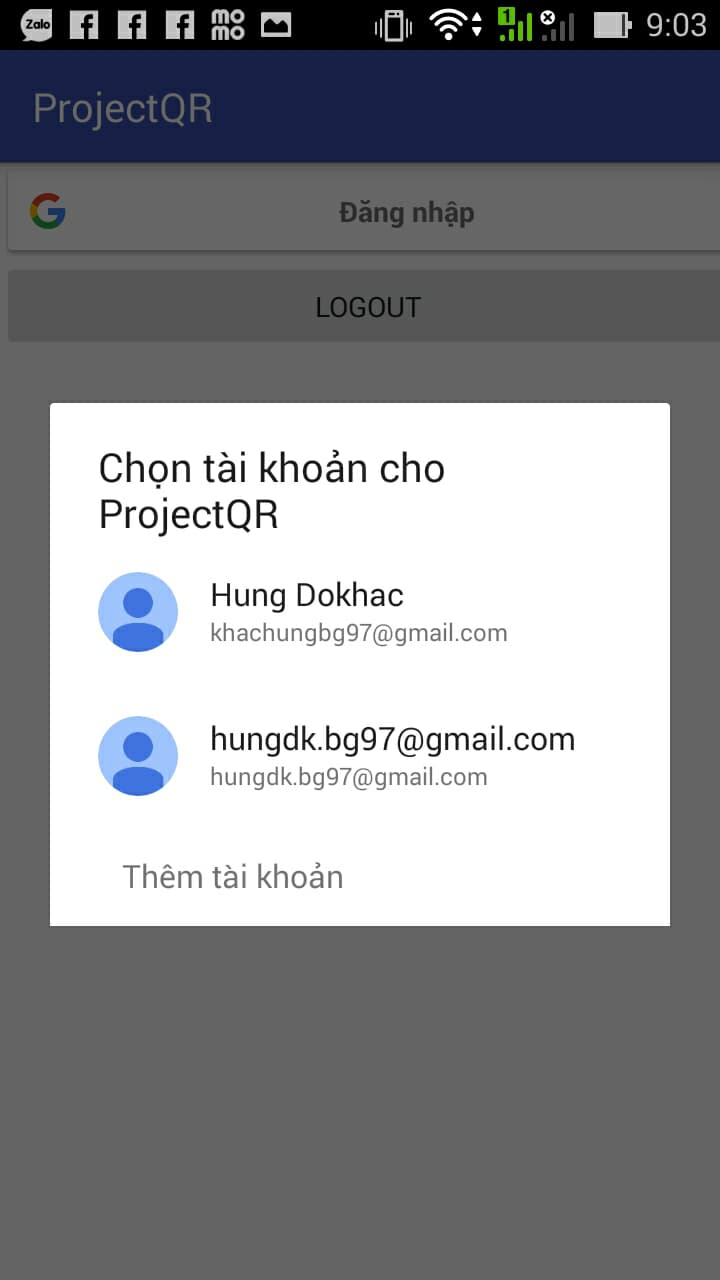
<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

cấu hình của file log được viết trong file Log4jHelper.class

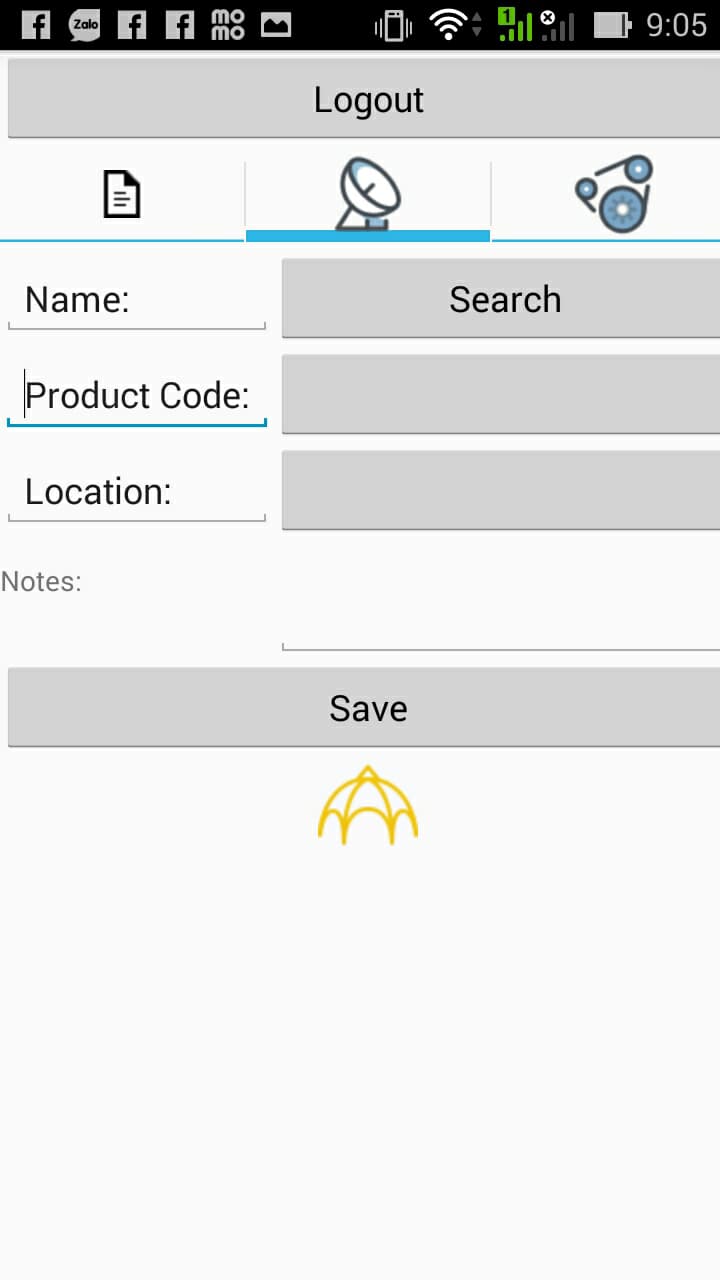
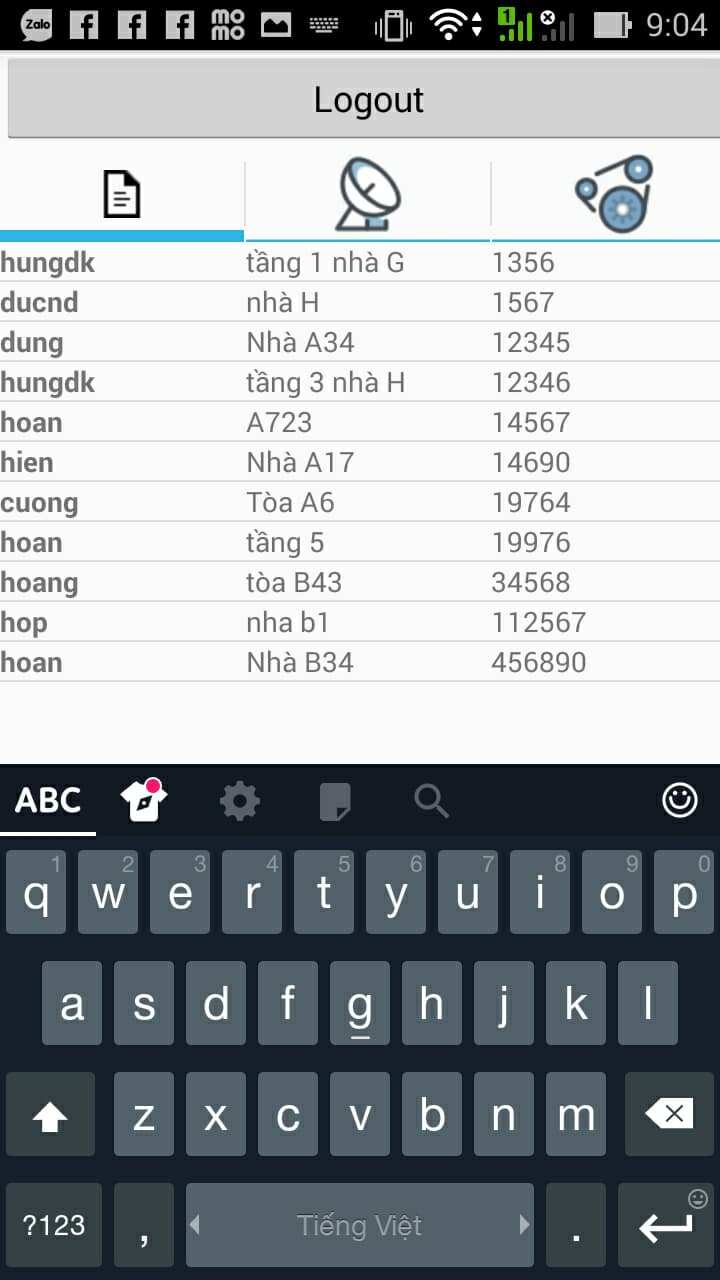
**public class** Log4jHelper {  
 **private final static** LogConfigurator ***mLogConfigrator*** = **new** LogConfigurator();  
  
 **static** {  
 *configureLog4j*();  
 }  
  
 **private static void** configureLog4j() {  
 String fileName = Environment.*getExternalStorageDirectory*() + **"/"** + **"log4j.log"**;  
 String filePattern = **"%d - [%c] - %p : %m%n"**;  
 **int** maxBackupSize = 10;  
 **long** maxFileSize = 1024 \* 1024;  
  
 *configure*( fileName, filePattern, maxBackupSize, maxFileSize );  
 }  
  
 **private static void** configure( String fileName, String filePattern, **int** maxBackupSize, **long** maxFileSize ) {  
 ***mLogConfigrator***.setFileName( fileName );  
 ***mLogConfigrator***.setMaxFileSize( maxFileSize );  
 ***mLogConfigrator***.setFilePattern(filePattern);  
 ***mLogConfigrator***.setMaxBackupSize(maxBackupSize);  
 ***mLogConfigrator***.setUseLogCatAppender(**true**);  
 ***mLogConfigrator***.configure();  
  
 }  
  
 **public static** org.apache.log4j.Logger getLogger( String name ) {  
 org.apache.log4j.Logger logger = org.apache.log4j.Logger.*getLogger*( name );  
 **return** logger;  
 }  
}

Sau đó tạo đối tượng Loger để ghi lại hoạt động trong Main2Activity.class

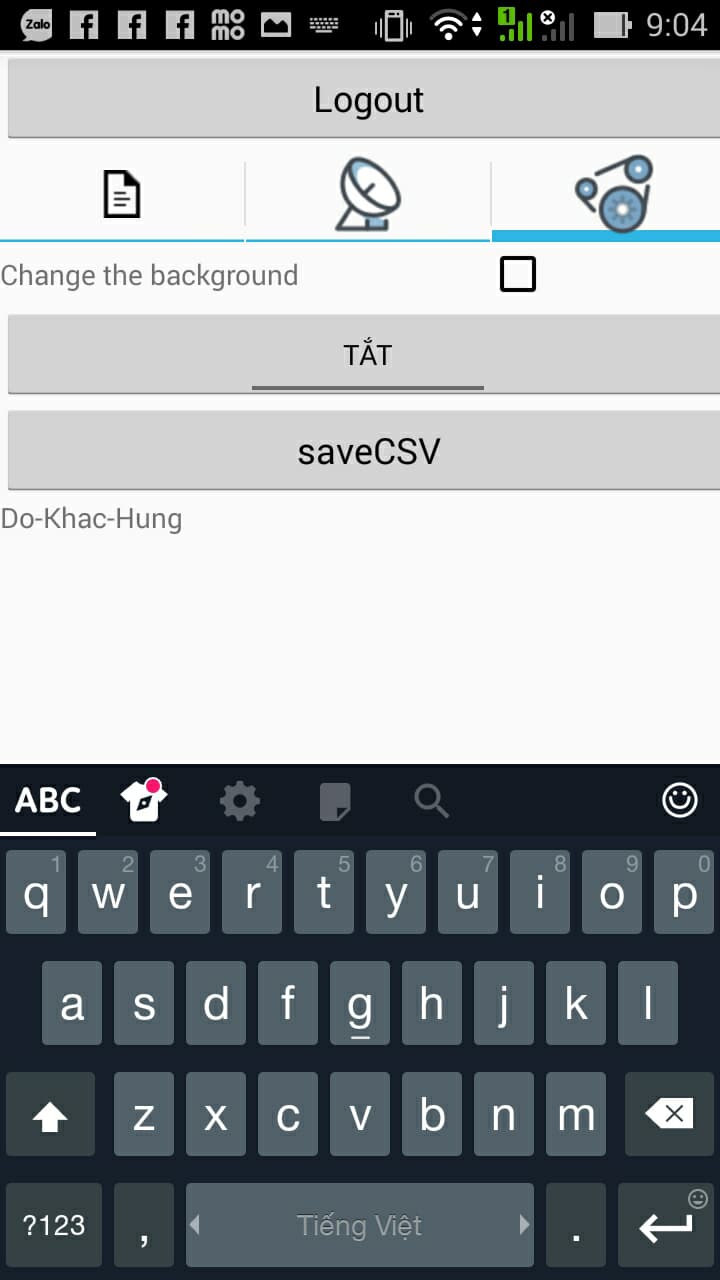
## III.4 Giao diện kết quả của ứng dụng

Hình 5 :Màn hình đăng nhập Hình 6:Popup thông báo

Hình 7 :Màn hình của Tab Scan code Hình 8 :Màn hình Tab hiển thị sản phẩm



Hình 9:Màn hình Tab Settings.

# Tài liệu tham khảo

1. *What is Android?* (n.d.). Được lấy từ trang Android Offical Website: http://www.android.com/about/
2. *Mã vạch*. (n.d.). Được lấy từ trang Wikipedia, the free encyclopedia: http://vi.wikipedia.org/wiki/Mã\_vạch
3. *Mã QR*. (n.d.). Được lấy từ trang Wikipedia, the free encyclopedia: http://vi.wikipedia.org/wiki/Mã\_QR
4. *zxing - Google Code*. (n.d.). Được lấy từ trang Google Code: http://code.google.com/p/zxing/
5. *Calling ZXing in a program*. (n.d.). Được lấy từ trang Google Code: http://code.google.com/p/zxing/wiki/DeveloperNotes

# [6] [*Android Writing Logs to text File*](https://stackoverflow.com/questions/1756296/android-writing-logs-to-text-file)*.*Được lấy từ trang stackoverflow:

https://stackoverflow.com/questions/1756296/android-writing-logs-to-text- file/2873062