**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**

**Dự án : DULY - Let’s travel with me !**

**(Ứng dụng hỗ trợ du lịch )**

**Sinh viên:** Trương Thị Mỹ Duyên

Lâm Toàn

Bùi Thị Hà Tiên

Cao Minh Trí

Lê Đình Thắng

**Lớp :** 19TCLC-DT4

**GVHD :** Lê Mỹ Hạnh

*Đà Nẵng , 18/12/2019*

**LỜI CẢM ƠN**

Lời đầu tiên chúng em xin bày tỏ lòng biết ơn chân thành đến quý Thầy Cô trong trường và đặc biệt là quý Thầy Cô trong khoa Công Nghệ Thông Tin , những người đã hết lòng dạy dỗ , truyền đạt những kiến thức khoa học và kinh nghiệm quý báu trong thời gian chúng em học tập ở trường .

Chúng em xin chân thành gửi lời cảm ơn đến cô Lê Mỹ Hạnh người đã hết lòng giúp đỡ và tận tình hướng dẫn chúng em trong quá trình làm đồ án này .

Mặc dù chúng em đã cố gắng nổ lực thực hiện đề tài này , song luận văn chắc chắn khó tránh khỏi những thiếu sót . Do đó , chúng em rất mong nhận được sự thông cảm và chỉ bảo tận tình của quý thầy cô .

Một lần nữa chúng em xin chân thành cảm ơn !

Nhóm sinh viên thực hiện :

Trương Thị Mỹ Duyên

Lâm Toàn

Bùi Thị Hà Tiên

Cao Minh Trí

Lê Đình Thắng

**NHẬN XÉT CỦA HỘI ĐỒNG DUYỆT ĐỒ ÁN**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

Đà Nẵng , ngày tháng năm 2020

*Hội đồng duyệt đồ án*

**NHẬN XÉT CỦA HỘI ĐỒNG DUYỆT ĐỒ ÁN**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

**……………………………………………………………………………………………**

Đà Nẵng , ngày tháng năm 2020

*Hội đồng duyệt đồ án*

**MỤC LỤC**

**Chương 1 : Lý do chọn đề tài**

1.1.Bối cảnh chọn đề tài 7

1.2.Tên dự án 7

**Chương 2 : Mục tiêu, phạm vi**

2.1. Mục tiêu dài hạn 8

2.2. Mục tiêu cụ thể phù hợp với mục tiêu dài hạn 8

**Chương 3 : Cơ sở lý thuyết** 10

3.1.Giới thiệu Android 10

3.2. Áp dụng API – SMS Retriever API vào xác nhận thông tin người dùng. 10

3.2.1. Lấy số điện thoại của người dùng 13

3.2.2. Bắt đầu truy xuất SMS 15

3.2.3. Nhận tin nhắn xác minh 17 3.3 . Cloud Storage 19

3.4. Kiến trúc Android 23

3.4..1 Android librari 24

3.4.2.Application Framework 24

3.4.3. Tầng ứng dụng (Application) 25

3.5 Các thành phần của một ứng dụng Android 25

**Chương 4. Thiết kế và xây dựng chương trình**

4.1. Phân tích và xây dựng chương trình 31

4.1.1 Đối tượng mục tiêu 33

4.1.2. Mức độ phổ biến 34

4.2. Phân tích các tính năng 34

**Chương 5 : Triển khai và đánh giá kết quả**

5.1.Cách thức triển khai 34

5.2 . Đánh giá kết quả . 34

**KẾT LUẬN HƯỚNG PHÁT TRIỂN**

**TÀI LIỆU TRÍCH DẪN** 35

**DANH MỤC HÌNH ẢNH** 36

**Chương 1 : LÝ DO CHỌN ĐỀ TÀI**

**1.1. Bối cảnh chọn đề tài**

Từ lúc thế giới được hình thành , con người bắt đầu xuất hiện thì những chuyến phiêu lưu tìm tòi những vùng đất mới cũng chớm nở . Ban đầu là những chuyến di cư lẻ tẻ từ châu Phi sang Thái Bình Dương của lớp người tối cổ , sau đấy là sang châu Á . Đỉnh điểm với cuộc khai hoang của Colombo khi tìm ra châu Mỹ . Sau này , là các cuộc du hành xuyên các sa mạc của những du mục để bán hàng , vượt đại dương cũ những người thủy thủ đi tìm vùng đất lạ. Như thế có thể thấy , từ xa xưa con người đã có ham muốn tìm kiếm , khám phá những miền đất mới . Cho đến khi hiện nay , vào thế kỉ XXI , người sống chật chội , những vùng đã được khai hoang hết , con người ta vẫn muốn đi không nhằm mục đích khai hóa mà là để tìm hiểu , để trải nghiệm . Những chuyến du lịch đến những nơi xa xôi , lạ lẫm để nhìn nhận và hiểu biết ở nơi đến , gặp những con người xa lạ , thử những thứ văn hóa chưa từng nghe đến là những trải nghiệm mà ai cũng mong muốn trong đời . Ngày nay với sự phát triển của xã hội , con người ngày càng quan tâm đến đời sống tinh thần hơn , du lịch là một trong những thiết yếu mà họ sẽ nhắm đến để giải stress sau những giờ làm việc căng thẳng , cùng nhau gắn kết tình bạn hay một mối quan hệ nào đó , tự thưởng cho bản thân sau cuối tuần hay kết thúc một năm. Thế nhưng với nhiều sự lựa chọn địa điểm đến , cũng như lịch trình sẽ phải đi đâu , sẽ phải làm gì , .. vì có rất nhiều như thế khiến bạn phân vân hay lần đầu đi du lịch bạn không biết nên làm sao . Vì vậy dự án – Duly , Let’s travel with me ! – được ra đời để giúp bạn có một chuyến du lịch đầy trải nghiệm .

**1.2. Tên dự án**

**DULY** một tên gọi mà chúng tôi đặt ra để gợi đến dự án của mình , nó được hiểu như “ du lịch ” . Với sologan “Let’s travel with me ” . Chúng tôi đảm bảo cho bạn có được một chuyến đi cực kì trải nghiệm với nhiều kỷ niệm trọn vẹn .

**Chương 2 : MỤC TIÊU , PHẠM VI**

**2.1 Mục tiêu dài hạn .**

**DULY – Let’s travel with me .** – được thành lập với mong muốn sẽ đem đến cho bạn những trải nghiệm tuyệt vời nhất về chuyến du lịch của mình . Chúng tôi cung cấp cho bạn ban đầu là những gợi ý về những địa điểm du lịch , sau khi được bạn chọn địa điểm , chúng tôi sẽ phân tích những địa điểm du lịch mà tại đó bạn có thể đi với các mốc thời gian cụ thể , bạn có thể hoặc lựa chọn nơi bạn muốn đến có thể ghi chú lại . Ứng dụng có google map mặc định offline nên bạn có thể truy cập địa điểm đi thoải mái ngay cả khi bạn không có internet . Những quán ăn , nhà hàng sẽ được thông báo trên bản đồ đó và sẽ gợi ý cho bạn những địa điểm gần nơi bạn đứng nhất , những bình luận của những người đã sử dụng nhà hàng hay món ăn đó sẽ giúp bạn có nhận xét về giá cả cũng như chất lượng hơn , từ đó lựa chọn quán ăn thích hợp nhất . Ứng dụng cũng cho phép bạn đặt khách sạn , homestay với các mốc so sánh giá cả mà chúng tôi đã tìm hiểu để bạn có thể so sánh và lựa chọn . Điều đặc biệt mà **DULY** giành cho bạn đó là bạn có thể đăng ký ở cùng nhà của người bản xứ . Giống như ở nhà của một người quen , bạn được tiếp đón , ăn cùng gia đình , ở lại phòng mà gia đình chuẩn bị sẵn , bạn cũng có thể được tham gia sinh hoạt chung với mọi người trong nhà . Một blog riêng dành cho những **DULYer** – những tay phượt thủ chuyên nghiệp – sẽ đánh giá về những địa điểm , những quán ăn , nhà hàng , khách sạn , hay một gia đình mà bạn ghé đến . Bạn cũng có thể chia sẽ câu chuyện về hành trình trải nghiệm của mình , những bài học mà bạn nhận được sau chuyến đi, những hình ảnh mà bạn có được sau khi dùng **DULY** .

**2.2 . Mục tiêu cụ thể phù hợp với mục tiêu dài hạn**

**2.2.1. Gợi ý về những địa điểm du lịch**

Duly được tích hợp, cài đặt sẵn những hình ảnh về những địa điểm du lịch nổi tiếng mà mọi người thường xuyên đến nhất . Đó là hình ảnh về những thành phố lớn , những nơi mà mọi người hay đi du lịch nhiều nhất . Trong đó là hình ảnh của từng điểm tham quan mà người dùng có thể ghé thăm khi đến thành phố . Nơi bạn lựa chọn đến có thể có những mốc thời gian nó mở cửa hay kiểm tra nó có mở cửa hay không , thời gian được khuyên nên đến thăm , hay lịch trình cụ thể về chuyến đi với các mốc thời gian và địa điểm. Nếu bạn chưa có thời gian đến nơi đấy , hoặc có thể do một lí do nào đó khiến bạn bị trục trặc bạn có thể ghi chú nó lại và sau đấy nó sẽ nhắc nhở bạn. Trong quá trình bạn đi, bạn có thể sẽ không kết nối được mạng ở những vùng núi hay những nơi không thể bắt sóng điện thoại , thì hãy yên tâm bởi DULY được cài đặt sẵn google map offline để bạn dễ dàng đến với địa điểm đã định sẵn hơn.

**2.2.2. Gợi ý về nhà hàng , quán ăn.**

DULY cũng được mặc định những địa điểm quán ăn , nhà hàng . Bạn có thể tìm kiếm món ăn , so sánh giá cả giữa các nhà hàng , tìm kiếm địa điểm của chúng , những địa điểm phù hợp nhất với bạn cũng sẽ được gợi ý . Bạn cũng có thể tìm hiểu về nhà hàng đó thông qua những lời đánh giá nhận xét của những người từng dùng .

**2.2.3 . Thuê homestay hay ở cùng hộ gia đình**

DULY với những gợi ý về những khách sạn , homestay giá rẻ , những hình ảnh phòng , đội ngũ nhân viên phục vụ , chất lượng cũng như giá cả sẽ được đặt ra , bạn có thể xem xét và lựa chọn . Nếu bạn là một người yêu thích sự hòa nhập thì hãy lựa chọn ở hạng mục khác , với tính năng giúp bạn có thể lựa chọn một ngôi nhà ngay tại địa phương , tại đó bạn có thể sống chung như một thành viên trong gia đình , được phép tham gia những hoạt động tổ chức trong nhà đó . Với chức năng sống cùng người lạ ở một nơi xa , chúng tôi đảm bảo cho bạn những trải nghiệm thật mới mẻ , đặc sắc và an toàn dành cho bạn.

**2.2.4. Viết blog – DULYer**

DULY cho phép bạn sau mỗi chuyến đi sẽ viết những trải nghiệm mà mình có được sau khi dùng app . Những đánh giá nhận xét trong mỗi phần của DULY . Bạn có thể chia sẽ những hình ảnh của mình hoặc với gia đình mà bạn đến ở cùng .

**Chương 3 : CƠ SỞ LÝ THUYẾT**

**3.1 Giới thiệu về hệ điều hành điện và ứng dụng Android**

Ứng dụng Android được viết bằng ngôn ngữ lập trình Java. Bộ công cụ SDK Android sẽ biên dịch mã cùng với tệp dữ liệu tài nguyên kèm theo, và cho vào một APK (Tệp lưu trữ có hậu tố .apk chứa tất cả nội dung ứng dụng và các thiết bị trên nền tảng Android có thể cài đặt.).

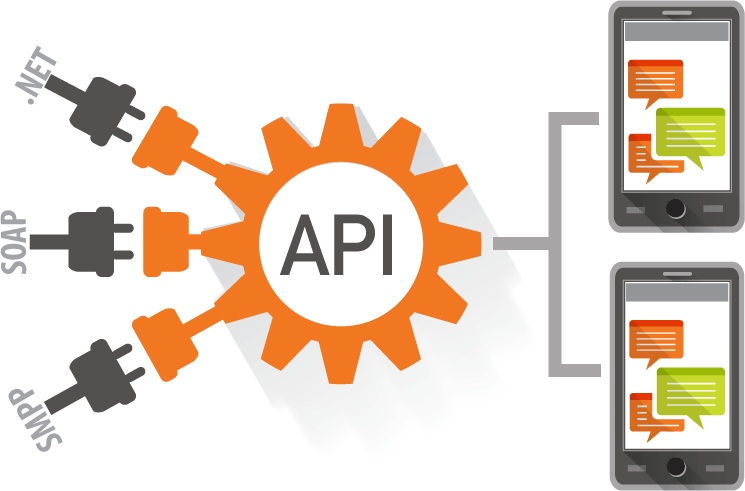
Sau khi một ứng dụng được cài đặt vào một thiết bị, nó sẽ ở bên trong hộp cát bảo mật (Sandbox) của chính nó:

* Hệ thống gán cho từng ứng dụng một ID người dùng Linux duy nhất (ID chỉ được sử dụng bởi hệ thống và không xác định đối với ứng dụng). Hệ thống sẽ đặt quyền cho tất cả tệp trong một ứng dụng sao cho chỉ ID người dùng được gán cho ứng dụng đó mới có thể truy cập.
* Mỗi tiến trình có máy ảo (VM) riêng của mình, vì thế mã của một ứng dụng sẽ chạy độc lập với các ứng dụng khác.
* Theo mặc định, mọi ứng dụng chạy trong tiến trình Linux của chính nó . Android khởi động tiến trình khi bất kỳ thành phần nào của ứng dụng cần được thực thi, sau đó tắt tiến trình khi không còn cần nữa haowcj khi hệ thống phải khôi phục bộ nhớ cho các ứng dụng khác.

Bằng cách này, mỗi ứng dụng chỉ có thể truy cập vào các thành phần mà nó cần thực hiện công việc và không hơn. Điều này tạo ra một môi trường bảo mật mà trong đó ứng dụng không thể truy cập các bộ phận khác của hệ thống mà nó chưa có quyền.

* 1. **Áp dụng API – SMS Retriever API vào xác nhận thông tin người dùng**

API là một phần mềm giao tiếp khi sử dụng các ứng dụng khác . API được ví von nó như là bàn phím để người dùng tương tác khi dùng máy tính.API được tạo ra một cách riêng biệt cho những ứng dụng khác nhau.API trên từng ngôn ngữ lập trình khác nhau sẽ có bộ API tương ứng.



* **Có 2 dòng chính sách đối với việc công bố các APIs:**

1. Bảo mật các APIs: Chẳng hạn như Sony thường chỉ cung cấp API chính thức của PlayStation 2 cho các nhà phát triển có đăng ký vì muốn giới hạn những người có thể viết trò chơi trên PlayStation 2 , và muốn thu lợi nhuận từ những người này càng nhiều càng tốt.
2. Cung cấp APIs miễn phí : Chẳng hạn Microsoft công bố hầu như hoàn toàn thông tin về các API, để cho các phần mềm có thể được viết chạy trên nền Windows.



* **SMS API là gì ?**

1. Là giao thức kết nối để sử các dịch vụ SMS thông qua API giúp cho việc lập trình các chức năng SMS một cách dễ dàng như ( kiểm tra các kết nối đến hệ thống SMS, kiêm tra tài khoản , lấy trạng thái tin nhắn đã gửi , gửi tin nhắn đến hàng loạt số điện thoại,…)
2. Với SMS API do Abenla phát triển , tất cả các nhà lập trình đều có thể sử dụng để tích hợp vào hệ thống , chương trình ,… của mình một cách dễ dàng.

* **Mô hình và cơ chế hoạt động:**

1. **Mô hình hoạt động :**

- Kết nối thông qua Web Services:POST/GET,SOAP

- Giao thức hỗ trợ : SMTP,SMPP ( tích hợp vào các ứng dụng chuẩn theo giao thức này )



* Ứng dụng của SMS API :

- Tích hợp vào hệ thống ngân hàng , chứng khoáng , bảo hiểm , tài chính để cảnh báo , thông báo , xác thực ,…

- Tích hợp vào các hệ thống: CRM , ERP,HRM,…

- Tích hợp vào các hệ thống cảnh báo sever , thiết bị,…

- Tích hợp vào tất cả các website,…

**3.2.1. Lấy số điện thoại của người dùng:**

Bạn có thể lấy số điện thoại của người dùng theo bất kì cách nào phù hợp với ứng dụng của bạn.Thông thường , đó là trải nghiệm người dùng tốt nhất để sử dụng trình chọn gợi ý để nhắc người dùng chọn từ các số điện thoại được lưu trên thiết bị và do đó tránh phải nhập số điện thoại theo cách thủ công.Để sử dụng công cụ chọn gợi ý:

//Construct a request for phone numbers and show the picker

Private void requestHint(){

HintRequest hintRequest = new HintRequest.Builder()

.setPhoneNumberIdentifierSupported(true)

.build();

PendingIntent intent = Auth.CredentialsApi.getHintPickerIntent(apiClient,hintRequest);

startIntentSenderForResult(intent.getIntentSender(),RESOLVE\_HINT,null,0,0,0);

}

//Obtain the phone number from the result

@Override

Public void onActivityResult(int requestCode,int resultCode, Intent data) {

super.onActivityResult (requestCode,resultCode , data);

if (requestCode == RESOLVE\_HINT) {

If (resultCode == RESULT\_OK) {

Credential credential = data.getParcelableExtra(Credential.EXTRA\_KEY);

//credential.getId(); <-- will need to process phone number string

}

}

}

**3.2.2. Bắt đầu truy xuất SMS:**

Khi bạn đã sẵn sàng để xác minh số điện thoại của người dùng, sử đụng instance của **SmsRetrieverClient** object, thực hiện gọi hàm **startSmsRetriever** và lắng nghe callback onSuccess/onFailure của Task.

//Get an instance of SmsRetrieverClient , used to start listening for a matching

// SMS message.

SmsRetrieverClient client = SmsRetriever.getClient(this /\* context \*/);

// Starts SmsRetriever, which waits for ONE matching SMS message until time out.

// (5 minutes). The matching SMS message will be sent via a Broadcast Intent with

// action SmsRetriever#SMS\_RETRIEVED\_ACTION.

Task<Void>task = client.startSmsRetriever();

//Listen for success/failure of the start Task.If in a background thread,this can be made blocking using Tasks.await(task,[timeout]);

task.addOnSuccessLisener(new OnSuccessListener<Void>() {

@Override

public void onSuccess(Void aVoid){

//Successfully started retriever,expect broadcast intent

//…

}

}

});

task.addOnFailureListener(new OnFailureListener () {

@Override

public void onFailure(@NonNull Exception e){

//Failed to start retriever, inspect Exception for more details

//…

}

});

Tác vụ truy xuất SMS được bắt đầu khi SMS cần được lắng nghe.SmsRetrieverClient đợi trong 5 phút cho một tin nhắn sms phù hợp.SMSRetrieverClient không thể được hủy đăng ký vì nó tự động hết thời gian sau 5 phút, vì vậy cần phải cẩn thận nếu activity finish trước 5 phút và trước khi nhận được SMS.

**3.2.3. Nhận tin nhắn xác minh:**

Khi bạn đang nghe tin nhắn SMS đến , bạn có thể yêu cầu hệ thống xác minh của bạn gửi mã xác minh đến số điện thoại người dùng mà bạn đã nhận được trong bước đầu tiên.Trong năm phút tiếp theo,Khi thiết bị nhận được tin nhắn SMS chứa mã OTP này , Play services sẽ phát đến ứng dụng của bạn nhằm nhắc nhở người dùng cho phép đọc tin nhắn.Một thông báo kích hoạt Broadcast chỉ khi nó đáp ứng các tiêu chí sau:

- Tin nhắn có chứa 4-10 ký tự gồm chữ và số và có ít nhất 1 số.

- Tin nhắn được gửi bởi một số điện thoại không có trong danh bạ của người dùng.

- Nếu bạn đã chỉ định số điện thoại của người gửi, tin nhắn đã được gửi bởi số đó.

Sử dụng BroadcastReceiver để nhận thông báo xác minh.Khi SMS nhận được trên thiết bị của người dùng,Play services sẽ bắn tới ứng dụng 1 Intent SmsRetriever.SMS\_RETRIEVED\_ACTION, trong đó có chứa văn bản của tin nhắn.

/\*\*

\* BroadcastReceiver to wait for SMS messages.This can be registered either

\* in the AndroidManifest or at runtime.Should filter Intents on

\*SmsRetriever.SMS\_RETRIEVED\_ACTION

\*/

public class MySMSBroadcastReceiver extends BroadcastReceiver {

@Override

public void onReceive(Context context,Intent intent) {

if (SmsRetriever.SMS\_RETRIEVED\_ACTION.equals(intent.getAction())) {

Bundle extras=intent.getExtras();

Status status = (Status) extras.get(SmsRetriever.EXTRA\_STATUS);

switch(status.getStatusCode()) {

case CommomStatusCodes.SUCESS:

// Get SMS message contents

String message =(String) extras.get(SmsRetriever.EXTRA\_SMS\_MESSAGE);

// Extract one-time code from the message and complete verification

// by sending the code back to your server.

break;

case CommonStatusCodes.TIMEOUT:

// Waiting for SMS timed out ( 5 minutes)

// Handle the error …

}

}

}

}

Bạn cần đăng ký BroadcastReceiver này trong Manifest

<receiver android:name=”.SMSBroadcastReceiver”android:exported=”true”>

<intent-filter>

<action android:name=”com.google.gms.auth.api.phone.SMS\_RETRIEVED”/>

<intent-filter>

</receiver>

Broadcast Receiver cũng có thể register tại activity hoặc fragment.

**3.3. Cloud Storage**

**Công nghệ Cloud Storage là gì ?**

- Cloud Storage là tên một giải pháp lưu trữ toàn diện được xây dựng dựa trên nền tảng điện toán đám mây thông minh , với dung lượng lưu trữ không giưới hạn , tính an toàn bảo mật cao và chi phí thấp.

- Cloud Storage sẽ là không gian lữu trữ an toàn cho dữ liệu của bạn và giúp bạn giải quyết mọi vấn đề liên quan đến dữ liệu, việc hợp tác hay làm việc nhóm.

- Một số hình thức sử dụng dịch vụ của nhà cung cấp cho phép người dùng là lưu trữ, quản lý , chia sẽ và backup dữ liệu của họ từ xa. Dịch vụ này cho phép người dùng có thể truy cập tập tin ấy bất cứ lúc nào , miễn là có truy cập internet. Dữ liệu của bạn sẽ được lưu trên server của nhà cung cấp đó và bạn sẽ sử dụng một ứng dụng desktop hoặc ứng dụng web online của họ để truy xuất dữ liệu

- Một số dịch vu lưu trữ đám mây hiện nay : AWS, Google Drive , Google Storage , Droxbox, OneDrive ,…

**Cloud Storage có gì ?**

- Dịch vụ sao lưu dự phòng dữ liệu : Cloud Storage sao lưu toàn bộ các loại dữ liệu , tự động lập lịch quay vòng . Cloud Storage hỗ trợ đa giao thức kết nối như FTP,SMB,S3… giúp bạn có thể thoải mái kết nối , và Cloud Storage tích hợp phần mềm chuyên nghiệp, cho phép bạn làm việc mọi nơi.

- Dịch vụ phân phối , chia sẽ dữ liệu : Dữ liệu của bạn có thể được truy cập qua nhiều dạng thức như FTP,S3,HTTP/HTTPS. Cloud Storage hỗ trợ quản lý phiên bản, phân quyền , mã hóa, duy trì chính sách bảo mật mãi mãi, đảm bảo dữ liệu của bạn sẽ luôn nằm trong vòng kiểm soát của bạn và những người bạn chia sẻ mà thôi.

- Storebox - Đồng bộ , chia sẻ dữ liệu : Đây là giải pháp lưu trữ , chia sẻ dữ liệu nội bộ, phù hợp với cá nhân, doanh nghiệp có nhu cầu chia sẻ, làm việc nhóm ,… Storebox được xây dựng đa nhiệm , đa thiết bị , đa nền tảng . Với tốc độ đồng bộ cực nhanh, chức năng phân quyền , đặt hiệu lực thười gian chia sẻ thông minh hay chia sẽ trực tiếp với những người cùng sử dụng Storebox chỉ bằng thao tác gõ tên… Storebox xứng đáng là một công cụ hữu ích cho công việc của bạn và có thể đánh bại những sản phẩm khác.

- Dịch vụ tư vấn giải pháp , cung cấp thiết bị lữu trữ : Tư vấn , thiết kế các giải pháp lưu trữ , sao lưu , dự phòng dữ liệu phù hợp với nhu cầu thức tế và mong muốn của bạn. Chúng tôi xây dựng phần mềm , dịch vụ lưu trữ theo yêu cầu , và cung cấp thiết bị lưu trữ theo nhiều hình thức như : bán , cho thuê,… với chi phí hoàn toàn hợp lý với nhu cầu của bạn.

**Các trường hợp sử dụng lưu trữ đám mây :**

- Backup and Recovery: Sao lưu và phục hồi luôn là một phần quan trọng để đảm bảo dữ liệu luôn được bảo vệ và lấy lại khi cần thiết. Với yêu cầu ngày một tăng như hiện nay đó là cả một thách thức . Tuy nhiên lưu trữ đám mây mang lại chi phí thấp , độ bền cao và quy mô lớn là giải pháp thích hợp cho sao lưu , phục hồi .



- Kiểm thử và phát triển phần mềm : Môi trường kiểm tra và triển khai phần mềm , ứng dụng luôn đòi hỏi phải tách biệt , độc lập và mỗi môi trường cũng phải giống nhau để xây dựng , quản lý . Có một số công tu lớn đã tạo ra các ứng dụng có tốc độ nhanh bởi tận dụng được tính linh hoạt, hiệu suất cao và chi phí thấp của lưu trữ đám mây. Hiện nay , các developer trên thế giưới cũng đang theo xu hướng sang các lựa chọn lưu trữ đám mây thay vì phải nhức đầu quản lý ở môi trường cục bộ.

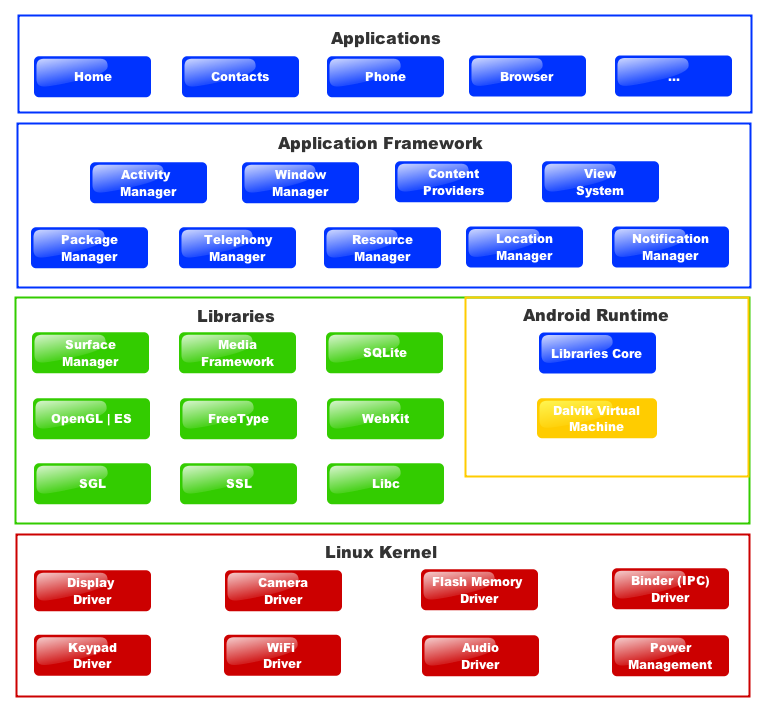
- Di chuyển dữ liệu đám mây : Tính sẵn sàng , độ ổn định và lợi thế về chi phí khiến các chủ doanh nghiệp bị hấp dẫn. Tuy nhiên với các quản trị viên quản lý việc lưu trữ, sao lưu , bảo mật lo ngại việc phải di chuyển một lương lớn dữ liệu lên đám mây hoặc ngược lại vì sợ mất thời gian , bảo mật ,… Hiện nay có một số nhà cung cấp ra mặt dịch vụ di chuyển dữ liệu đến đám mây bằng băng thông cao , giải quyết được vấn đề thời gian và bảo mật .

**Tổng kết**

Nhìn chung , Cloud Storage mang lại lợi ích nhiều. Nếu biết cách sử dụng cẩn thận và chính xác với nhu cầu của mình thì đúng là một bước tiến lớn của công nghệ trong cuộc sống và công việc .

**3.4. Kiến trúc Android**

Hệ điều hành android là một ngăn xếp của các thành phần ứng dụng (stack of software componnent), có thể chia thành sơ đồ 5 phần và 4 lớp như sơ đồ



*Sơ đồ kiến trúc Android*

**3.4.1. Android libraries**

Phía trên tầng Linux kernel là tầng libraries, đây là lớp thư viện gốc của Android. Lớp này cho phép các thiết bị xử lý các loại dữ liệu khác nhau. Các thư viện này được viết bằng ngôn ngữ C/C++:

* Surface manager: Quản lý hiển thị nội dung 2D và 3D.
* Media Framework: Bộ thư viện hỗ trợ âm thanh và hình ảnh
* SQLite: hệ cơ sở dữ liệu nhỏ gọn, tiện lợi hơn so với SQL, Oracle.. phù hợp với ứng dụng Android
* OpenGLES: dùng để tạo ra các đồ họa 3D dựa vào chuẩn OpenGLES 1.0 API
* FreeType: hỗ trợ xử lí bitmap và font vector.
* Webkit: công cụ trình duyệt được sử dụng để hiển thị nội dung HTML.
* SGL: các công cụ đồ họa 2D tiềm ẩn.
* SSL: hỗ trợ giao thức mã hóa Secure Sockets Layer trong bảo mật truyền thông Internet.
* Libc: bộ thư viện standard C được Google phát triển cho Android, có tên là bionic được điều chỉnh để tối ưu hóa cho các thiết bị chạy trên nền Linux.

**3.4.2. Application Framework**

Framework là các đoạn code đã được viết sẵn, cấu thành nên một bộ khung và các thư viện lập trình được đóng gói. Chúng cung cấp các tính năng có sẵn như mô hình, API và các yếu tố khác để tối giản cho việc phát triển các ứng dụng.

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.

Android Framework bao gồm các dịch vụ chính sau:

* Activitty Manager – Quản lí chu kì sống của ứng dụng và điều hướng các hoạt động
* Content Providers – Cho phép các ứng dụng có thể truy xuất dữ liệu từ các ứng dụng khác hoặc chia sẻ dữ liệu giữa các ứng dụng.
* Resource Manager – Hỗ trợ các thành phần không thuộc mã nguồn như các chuỗi, đồ họa được đặt bên ngoài…
* Notifications Manager – Cho phép các ứng dụng hiển thị cảnh báo và các thông báo cho người dùng.
* View System – Tập các thành phần giao diện (view) được sử dụng để tạo giao diện người dùng.

**3.4.3. Tầng ứng dụng (Application)**

Đây là lớp trên cùng của kiến trúc nền tảng Android. Android sẽ hoạt động với một bộ các ứng dụng bao gồm ứng dụng thư điện tử, gửi tin nhắn, lịch, bản đồ, trình duyệt web, danh bạ,... Tất cả các ứng dụng được viết bằng ngôn ngữ Java. Các ứng dụng này có thể được cung cấp sẵn hoặc phát triển bởi lập trình viên.

**3.5. Các thành phần của một ứng dụng Android**

****

*Các thành phần của một ứng dụng Android*

**ACTIVITY**

**Activity** là một thành phần của ứng dụng Android. Android Activity là nơi để ứng dụng tương tác trực tiếp với người dùng thông qua giao diện. Một ứng dụng có thể sẽ có nhiều màn hình và mỗi màn hình có thể là một Activity (nếu không sử dụng Fragment)..

Vòng đời của một Activity

Sơ đồ bắt đầu từ khi Activity launched, tức là khi Activity được kích hoạt, và được hệ thống đẩy vào BackStack. Sau khi kích hoạt, lần lượt các callback onCreate(), onStart(), onResume() sẽ được hệ thống gọi đến.  
Sau khi gọi đến các callback trên, thì Activity mới chính thức được xem là đang chạy (Activity running).

**Running**: khi Activity được kích hoạt, vào được hệ thống để vào BackStack, nó sẽ bước vào trạng thái active. Với trạng thái active, người dùng hoàn toàn có thể nhìn thấy và tương tác với Activity của ứng dụng

**Pause**: trạng thái này khá đặc biệt. Trạng thái tạm dừng. Như bạn đã làm quen trên kia, trạng thái này xảy ra khi mà Activity của bạn vẫn đang chạy, người dùng vẫn nhìn thấy, nhưng Activitykhi này lại bị che bởi một thành phần nào đó. Sự che Activity này ko hoàn toàn. Chính vì vậy mà Activity đó tuy được người dùng nhìn thấy nhưng không tương tác được.

**Stop**: trạng thái này khá giống với trạng thái dừng, nhưng Activity bị che khuất hoàn toàn bởi một thành phần giao diện nào đó, hoặc bởi một ứng dụng khác, lúc này người dùng không thể nhìn thấy Activity nữa.

**Deal**: nếu Activity được lấy ra khỏi BackStack, chúng sẽ bị hủy và rơi vào trạng thái này. Trường hợp này xảy ra khi user nhấn nút Back ở System Bar để thoát một Activity. Hoặc lời gọi hàm finish() từ một Activity để kết thúc chính nó. Cũng có khi ứng dụng ở trạng thái background quá lâu, hệ thống sẽ thu hồi tài nguyên bằng cách dừng hẳn các Activity trong ứng dụng, làm cho tất cả các Activity đều rơi vào trạng thái này.

Khi vào trạng thái dead, Activity sẽ kết thúc vòng đời của nó.

Các class cơ sở Activity:

**SERVICE**

S**ervice** là một thành phần (component) có thể thực hiện các hoạt động lâu dài trong background và nó không cung cấp một giao diện người dùng. Một thành phần khác của ứng dụng có thể khởi động nó, và nó tiếp tục chạy trong background ngay cả khi người dùng chuyển sang ứng dụng khác. Ngoài ra một thành phần có thể liên kết (bind) với một Service để tương tác với Service đó, thậm chí là thực hiện truyền thông liên tiến trình IPC (interprocess communication - IPC là một hoạt động chia sẽ dữ liệu qua nhiều tiến trình, thông thường sử dụng giao thức truyền thông và nó phải có Client và Server).

Service dùng để cập nhật dữ liệu, đưa ra các cảnh báo cần thiết, chơi nhạc,…

* Phân loại service:

**Unbound Service**

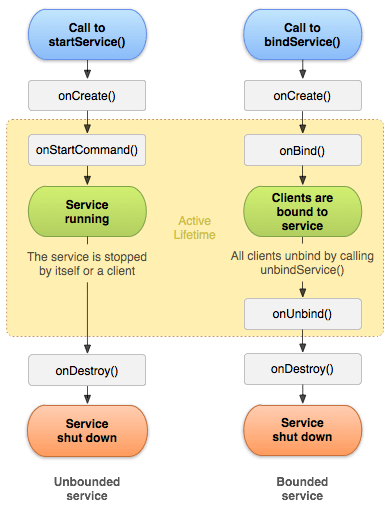
* Được sử dụng để thực hiện nhiệm vụ lâu dài, lặp đi lặp lại và có thể hoạt động mãi mãi.
* Được khởi động bởi gọi phương thức startService().
* Bị dừng lại hoặc bị hủy bởi gọi một cachs tường minh phương thức stopService().
* Hoạt động độc lập với thành phần đã khởi động nó.

**Bound service**

* Được sử dụng để thực hiện nhiệm vụ ở nền Background và ràng buộc với thành phần giao diện nào đó.
* Được khởi động bởi gọi phương thức bindService().
* Bị gỡ ràng buộc hoặc bị hủy bởi gọi unbindService()
* Phụ thuộc vào thành phần giao diện đã khởi động nó

**Intent Service**

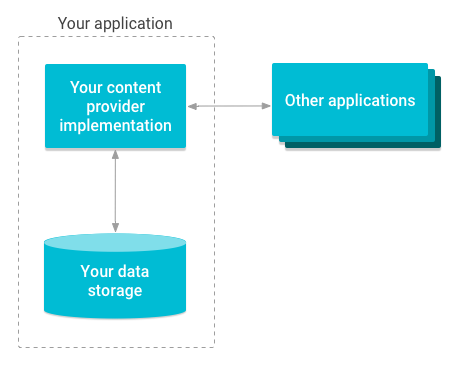
* Được sử dụng để thực hiện các nhiệm vụ một lần duy nhất, nghĩa là khi nhiệm vụ hoàn thành dịch vụ tự hủy.
* Được khởi động bởi gọi startService().
* Gọi một cách không tường minh phưng thức stopself() để hủy dịch vụ.
* Hoạt động độc lập với thành phần đã khởi động nó.



*Vòng đời của Service*

**CONTENT PROVIDER**

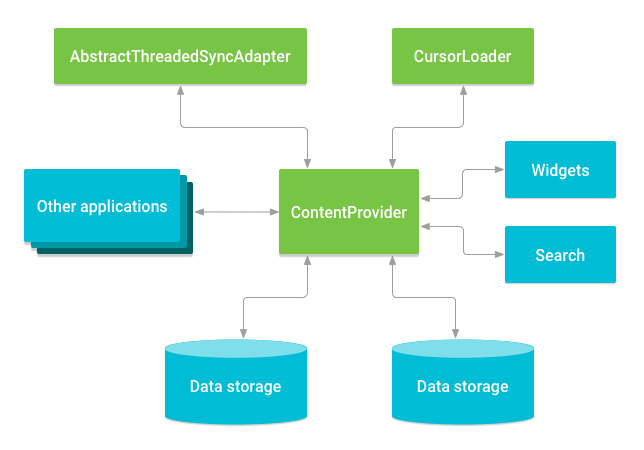
**Content provider** là một thành phần để quản lí truy cập dữ liệu, nó cung cấp các phương thức khác nhau để các ứng dụng có thể truy cập dữ liệu từ một ứng dụng khác bằng cách sử dụng ContentResolver. ContentProvider có thể giúp cho một ứng dụng quản lý quyền truy cập đến dữ liệu được lưu bởi ứng dụng đó, hoặc các ứng dụng khác, và đó là cách để ta có thể chia sẻ dữ liệu cho các ứng dụng khác nhau. Dữ liệu có thể được lưu trữ trong hệ thống file, trong một cơ sở dữ liệu SQLite, hay trong một cách khác nào đó



*Cách content providers quản lý việc truy cập tới bộ nhớ*

Content Provider điều phối việc truy cập tới bộ lưu trữ dữ liệu thông qua các API và các component như hình dưới, nó bao gồm

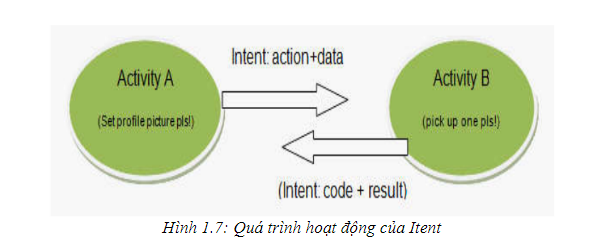
* Chia sẻ dữ liệu từ ứng dụng tới các ứng dụng khác
* Gửi dữ liệu sang widget
* Trả về một kết quả gợi ý khi search cho ứng dụng của bạn thông qua Seach Framework sử dụng [*SearchRecentSuggestionsProvider*](https://developer.android.com/reference/android/content/SearchRecentSuggestionsProvider.html)
* Đồng bộ dữ liệu của ứng dụng với server bằng cách sử dụng [*AbstractThreadedSyncAdapter*](https://developer.android.com/reference/android/content/AbstractThreadedSyncAdapter.html)
* Tải dữ liệu lên UI sử dụng [*CursorLoader*](https://developer.android.com/reference/android/content/CursorLoader.html)



Content provider hoạt động rất giống một cơ sở dữ liệu, bạn có thể truy vấn, chỉnh sửa nội dung, cũng như là thêm xóa các nội dung sử dụng phương thức: insert(), update(), delete(), query().

**INTENT**

**Intent** là một cơ cấu cho phép truyền thông điệp giữa các thành phần của một ứng dụng và giữa các ứng dụng với nhau.



Intent có hai thuộc tính quan trọng là:

* Action: Là hành động được thực hiện
* Data: Là dữ liệu được xử lý trong action, thường được diễn tả là một Uri.

**BROADCAST RECEIVE**

**Broadcast Receive** là một thành phần có nhiệm vụ nhận và đáp lại các thông báo broadcast. Nhiều broadcast khởi đầu trong mã hệ thống – ví dụ như thông báo múi giờ thay đổi, pin yếu, ảnh đã chụp, hay người dùng đã thay đổi ngôn ngữ… Các ứng dụng có thể tạo ra các broadcast, chẳng hạn để ứng dụng khác biết được một số dữ liệu đã được tải về thiết bị và sẵn sàng cho việc sử dụng chúng.

**** *Hoạt động của Broadcast Receive*

Một ứng dụng có thể có một số Broadcast Receiver để đáp lại thông báo nào mà nó cho là quan trọng . Tất cả các Receiver mở rộng từ lớp cơ sở BroadcastReceiver

Broadcast không hiển thị một giao diện người dùng. Tuy nhiên chúng có thể bắt đầu một Activity để đáp lại thông tin mà chúng nhận, hay chúng có thể sử dụng NotificationManager để cảnh báo người dùng. Notifications có thể lấy sự chú ý

**Chương 4 : THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH**

**4.1. Phân tích và xây dựng chương trình:**

* Làm rõ yêu cầu (Eliciting requirements): giao tiếp với khách hàng và người sử dụng để xác định các yêu cầu của họ.
* Xem xét yêu cầu (Analyzing requirements): xác định xem các yêu cầu được đặt ra có ở tình trạng không rõ ràng, không hoàn chỉnh, đa nghĩa, hoặc mâu thuẫn hay không, và giải quyết các vấn đề đó.
* Làm tài liệu yêu cầu (Recording requirements): các yêu cầu có thể được ghi lại theo nhiều hình thức, chẳng hạn các tài liệu ngôn ngữ tự nhiên, các tình huống sử dụng (use case), câu chuyện sử dụng (user story), hoặc các đặc tả tiến trình.

**4.1.1 Đối tượng mục tiêu:**

* **Đối tượng người dùng:**

**DULY – Let’s travel with me** là ứng dụng hướng tới đối tượng là:

* Những người trẻ đầy năng động đam mê du lịch phượt hay có thể là những người yêu thích khám phá những điều khác lạ chẳng hạn như trải nghiệm sinh hoạt cùng một người hoặc một gia đình bản địa hay cũng có thể chỉ đơn giản là bạn yêu thích cái đẹp của những danh lam thắng cảnh tại nơi mà bạn muốn đặt chân đến.
* Những gia đình muốn có một chuyến du lịch phù hợp với túi tiền và đem lại những trải nghiệm mới mẽ và đầy thú vị.
* **Đối tượng khách hàng:**

**DULY – Let’s travel with me** hướng tới những khách hàng tìm năng như:

* Các công ty cung cấp dịch vụ du lịch và muốn quảng bá dịch vụ của họ đồng thời đưa du lịch địa phương ngày càng phát triễn.
* Những địa điểm kinh doanh dịch vụ nghỉ ngơi như là nhà hàng và homestay và điểm đặc biệt mới lạ là các hộ gia đình địa phương có nhu cầu chia sẽ chỗ ở với mức giá hợp lý.
* Những địa điểm kinh doanh dịch vụ ẩm thực như:
* Nhà hàng, các quán ăn đặc sản mang tính chất địa phương
* Các quán ăn ngon, sạch đồng thời nổi tiếng được nhiều người dân bản địa biết đến.
* Những công ty chuyên cung cấp dịch vụ vận chuyển trong nội thành như:
* Những địa điểm chuyên cung cấp dịch vụ cho thuê xe máy phù hớp với những người trẻ thích thoải mái đi lại.
* Các công ty chuyên cho thuê xe vận tải từ 4 đến 16 chỗ phù hợp cho gia đình hoặc hội bạn.

**4.1.2. Mức độ phổ biến**

**DULY – Let’s travel with me** hướng tới phần lớn là những người trẻ(chiếm khoảng 70% dân số đất nước) là những người dễ dàng tiếp cận với công nghệ và có đam mê du lịch cùng với sự tiện lợi và những dịch vụ vô cùng thuận lợi đem đến nhiều lợi ích cho người dùng được tích hợp một cách chọn lọc của DULY nên có thể dễ dàng trở nên phổ biến .

**4.2. Phân tích các tính năng**

* **Tính năng DULY-Share your story**: tính năng này cho phép người dùng có thể chia sẽ những chuyến đi của mình bằng các định dạng như là hình ảnh hoặc video đi kèm với những dòng trạng thái của người dùng có thể giúp lưu lại chi tiết những trải nghiệm đó. Đồng thời nó cũng có thể được chia sẽ một cách công khai để các người dùng khác có thể đọc và tham khảo cho chuyến đi của chính họ. Tính năng này cho phép các người dùng được tương tác với nhau bằng hệ thống tin nhắn hoặc là gọi điện.
* **Tính năng DULY-Book accommodation**: tính năng này hỗ trợ người dùng tìm kiếm những chỗ nghỉ ngơi phù hợp với túi tiền và nhu cầu của người dùng. Tính năng cho người dùng biết được tình trạng nơi mình sẽ ở bằng các hình ảnh và thông tin được cung cấp từ chính các đối tác chuyên cung cấp dịch vụ nghỉ dưỡng hợp tác cùng DULY và được chính DULY xác minh.
* **Tính năng DULY-Experience**: Đây là một tính năng cũng liên quán đến chỗ ở nhưng nó lại mang một tính chất hoàn toàn khác với tất cả ứng dụng tìm kiếm chỗ nghỉ ngơi từ trước đến nay. Nếu bạn la một người trẻ đầy tự tin và yêu thích những sự trải nghiệm thì DULY-Experience chính là tính năng quá phù hợp với bạn. Bạn nghĩ sao với việc trải nghiệm cuộc sống với người bản địa, sinh hoạt như người bản địa với chi phí chỗ ở vô cùng thấp. Phía DULY sẽ cam kết về mức độ tin cây và an toàn cho người dùng với các điều khoản được rằng buộc trên hợp đồng với các chủ hộ gia đình.
* **Tính năng DULY-Transport**: tính năng này hổ trợ người dùng trong việc tìm những địa điểm chuyên cung cấp các dịch vụ như cho thuê phương tiện đi lại trong nội thành với ưu điểm là có thể đưa ra những gợi ý gần vị trí của bạn nhất.
* **Tính năng DULY-Planning**: tính năng này cho phép người dùng tạo ra kế hoạch cho chuyến đi của họ một cách chi tiết vừa phù hợp với túi tiền vừa tiết kiệm được thời gian di chuyển cho người dùng. Hãy thử nghĩ xem bạn sẽ thế nào khi du lịch tới một nơi hoàn toàn mới lạ với mình, DULY sẽ giúp bạn giải quyết các vấn đề đó.

+ Đầu tiên, tính nắng DULY-planning hỗ trợ người dùng sẽ đưa ra những gợi ý về những địa điểm ăn uống, vui chơi, nghỉ dưỡng và nhiều dịch vụ du lịch khác được sắp xếp theo nhóm thuận tiện cho người dùng.

+ Tiếp theo, người dùng sẽ chọn lọc những địa điểm mà mình cảm thấy thú vị và muốn đến tùy vào sở thích của mỗi người bằng cách chọn “thêm vào kế hoạch của tôi”.

+ Sau khi hoàn tất phần chọn lựa, DULY-Planning sẽ giúp người dùng đưa ra một thời gian biểu nhất định trong từng ngày. Việc của người dùng đơn giản chính là chọn những địa điệm họ muốn đến vào những khoảng thời gian nhất định vào thời gian biểu. Đồng thời lúc này hệ thống cũng sẽ hiện ra bản đồ để người dùng có thể biết nơi mình muốn đến ở vị trí nào giúp người dùng tránh khỏi việc bị tốn quá nhiều thời gian cho việc đi lại.

+ Tiếp đến hệ thộng sẽ tự tính toán quãng đường mà người dùng phải đi nếu áp dụng kế hoạch này và đưa ra góp ý cho người dùng .

+ Sau khi hoàn tất và lưu kế hoạch thì tính năng sẽ hổ trợ người dùng trong việc thực hiện chuyến đi bằng cách thông báo đến người dùng qua điện thoại ngày giờ và địa điểm chính xác trong kế hoạch đã được lưu cũng như là địa chỉ của điểm muốn đến.

* **Tính năng DULY-Evaluate**: cho phép người dùng đánh giá từng dịch vụ mà họ được trải nghiệm tại từng địa điểm qua hình thức cho điểm. Điểm này sẽ được tính bằng cách lấy trung bình cộng của tất cả các đánh giá và sẽ được hiển thị trực tiếp lên thư mục của dịch vụ đó giúp người dùng cân nhặc về việc lựa chọn điểm đến cho mình.

**Chương 5: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

**5.1 Cách thức triển khai:**

* **Chuẩn bị dự án**: Ở thời kỳ đầu, đội ngũ của DULY sẽ trực tiếp là người đi tìm hiểu các địa điểm du lịch của địa phương để cung cấp dữ liệu cho app. Đồng thời sẽ tìm kiếm những hộ gia đình sẵn sàng đồng hành cùng DULY để phát triển du lịch địa phương. Ở giai đoạn này, DULY sẽ cố gắng đưa được các tiện ích đến được nhiều khách hàng cũng như người dùng nhất có thể. Đối với khách hàng là những doanh nghiệp kinh doanh dịch vụ du lịch nếu cộng tác với chúng tôi thì họ sẽ gián tiếp quảng bá thương hiệu của mình mà không tốn một khoảng phí nào. Về phía người dùng chúng tôi sẽ sẽ làm các video viral trên youtube trên những kênh có lượng người xem lớn kết hợp với việc cung cấp những ưu đãi cho khách hàng để đưa được ứng dụng đến nhiều khách hàng nhất. DULY sẽ hoạt động phi lợi nhuận trong giai đoạn này.
* **Xây dựng dự án**: Sau khi đã thu hút một lượng người dùng và khách hàng ổn định, DULY sẽ bắt đầu thu lợi nhuận từ phía các doanh nghiệp trên mỗi người dùng của DULY sử dụng dịch vụ của doanh nghiệp đó thông qua DULY với càng nhiều người dùng thì phần trăm lợi nhuận mà DULY được hưởng sẽ càng giảm. Ngược lại từ phía người dùng DULY càng sử dụng nhiều dịch vụ trên DULY thì họ sẽ càng được hưởng nhiều ưu đãi các chương trình khuyến mãi hoặc tích điểm trên mỗi dịch vụ mà khách hàng đã sử dụng. Giai đoạn này DULY bất đầu hoàn vốn.
* **Đẩy mạnh dự án**: DULY sẽ nhân rộng phạm vi phủ sóng của mình bằng cách đào tạo thêm các nhóm nhân viên hiểu rõ về khu vực muốn mở rộng. Cử các nhân viên theo dự án này từ những ngày đầu để trực tiếp chỉ đạo việc thực hiện dự án để đẩy mạnh thị phần và thâu tóm người dùng một cách nhanh nhất.

**5.2 Đánh giá kết quả:**

DULY đã làm một cuộc khảo sát nhỏ thì trên 1000 người tham gia khảo sát có khoảng 65% cảm thấy đây là một ứng dụng hữu ích và có khả năng cao sẽ dùng trong thời gian tới nếu có cơ hội. Khoảng 25% cảm thấy ứng dùng cần được cải tiến để thu hút họ hơn. Khoảng 10% còn lại họ cho rằng ứng dụng vẫn chưa thực sự nổi bật với những tiện ích mà DULY đem lại.

**TÀI LIỆU TRÍCH DẪN**

<https://developer.android.com/reference/android/app/Application>

<https://www.tutorialspoint.com/android/android_architecture.htm>

<https://developer.android.com/studio/projects/android-library>

<https://source.android.com/devices/tech/dalvik>

<https://source.android.com/devices/architecture/kernel>

<https://developers.google.com/identity/sms-retriever/choose-an-api>

<https://developers.google.com/identity/sms-retriever/user-consent/overview>

<https://medium.com/androiddevelopers/automatic-sms-verification-with-sms-user-consent-da8c16135e25>

<https://developers.google.com/identity/sms-retriever/overview>

<https://medium.com/@sampadasalimath14/sms-retriever-api-for-automatic-sms-verification-e029cc501b1f>

<https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)>

<https://developer.android.com/guide/components/fundamentals>

<https://gs.statcounter.com/os-market-share>

**DANH MỤC HÌNH ẢNH**

https://fibo.vn/sms/wp-content/uploads/2018/06/sms-api-tich-hop-sms-vao-he-thong-chua-bao-gio-don-gian-hon-the.jpg

<https://www.topitworks.com/blogs/wp-content/uploads/2017/12/Capture-1024x973.jpg>

<http://tinnhanthuonghieu.net/wp-content/uploads/2016/03/sms-api.png>

<https://d3i71xaburhd42.cloudfront.net/eadaebd21a7ed9dd93d05edaba961064cf1843db/28-Figure4.1-1.png>

<https://zmikisoft.com/zmikisoft/uploads/2016/12/Components-of-Android1.jpg>

<https://d3i71xaburhd42.cloudfront.net/670f33b7d1236835928ac155a601f7d48f2164af/10-Figure3.2-1.png>

<http://artimg.ishenping.com/20190905221037005_ZCCCVB.jpg>

https://media.hoclaptrinh.vn/images/content-provider-trong-android-5d1aca47594eb.png