**KHÓA LUẬN TỐT NGHIỆP**

MÔ TẢ API

**HỆ THỐNG LOYALTY POINT**

**Nguyễn Tấn Đô – 1112074**

**Nguyễn Trương Trung Tín - 1112330**

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Khoa Công nghệ thông tin

Đại học Khoa học tự nhiên TP HCM

**MỤC LỤC**

[1 NGUYÊN TẮC HOẠT ĐỘNG 3](#_Toc420688953)

[2 CÁC API CHÍNH TRONG ỨNG DỤNG 4](#_Toc420688954)

[Lấy Danh Sách Cards 4](#_Toc420688955)

[Lấy Danh Sách Shops 4](#_Toc420688956)

[Lấy Danh Sách Customers 5](#_Toc420688957)

[Lấy Danh Sách Events 5](#_Toc420688958)

[Lấy Danh Sách Awards 6](#_Toc420688959)

[Tính Điểm Tích Lũy 6](#_Toc420688960)

[Cập Nhật Điểm Tích Lũy 7](#_Toc420688961)

# NGUYÊN TẮC HOẠT ĐỘNG

**Cách chứng thực:** mỗi khi gọi api, thư viện sẽ tự động kiểm tra xem ứng dụng manager đã được cài đặt chưa.

* **Nếu chưa cài đặt:** thì mở Google Play cho người dùng cài đặt (vì app chưa đưa lên Google Play Store nên chỉ mở ra một trang bất kỳ để demo).
* **Nếu đã cài đặt:** kiểm tra xem người dùng đã đăng nhập vào ứng dụng manager hay chưa?
* **Nếu chưa đăng nhập:** mở ứng dụng manager để người dùng đăng nhập vào, lưu lại token. Sau đó quay trở lại ứng dụng của người dùng và chạy tiếp phần còn lại của api.
* **Nếu đã đăng nhập:** chạy tiếp phần còn lại của api.

**Nguyên lí hoạt động:** thư viện chạy theo dạng callback, nghĩa là khi người dùng gọi một api bất kì, người dùng sẽ phải cài đặt 2 phương thức **onSuccess** và **onError** của một callback được cung cấp sẵn trong library, có các thông số input và output phù hợp với lời gọi api hiện tại. Khi api chạy xong sẽ gọi một trong 2 phương thức trên, người dùng nhận kết quả và xử lí tiếp.

**Chú ý:**

* Tất cả các model trong thư viện (AchievedEvent, Award, Card, Customer, Event, Product, Shop) đều được implements Parcelable, có thể put vào Bundle và lấy ra khỏi Bundle dễ dàng.
* Các api đều là static method, vì vậy chỉ cần gọi thông qua tên class:

LoyaltyPointAPI.api(...)

* Muốn sử dụng thư viện phải Add dependencies đến thư viện GSON:

Đối với Android Studio, thêm dòng này vào file build.gradle:

compile **'com.google.code.gson:gson:2.2.4'**

# CÁC API CHÍNH TRONG ỨNG DỤNG

# Lấy Danh Sách Cards

*// Test get list cards*LoyaltyPointAPI.getListCards(**yourContext**, **new** LoyaltyPointAPI.OnGetListCardsResult() {  
 @Override  
 **public void** onSuccess(ArrayList<Card> listCards) {  
 Log.e(**"getListCards"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.e(**"getListCards"**, error);  
 }  
});

# Lấy Danh Sách Shops

*// Test get list shops*LoyaltyPointAPI.getListShops(**yourContext**, **new** LoyaltyPointAPI.OnGetListShopsResult() {  
 @Override  
 **public void** onSuccess(ArrayList<Shop> listShops) {  
 Log.e(**"getListShops"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.e(**"getListShops"**, error);  
 }  
});

# Lấy Danh Sách Customers

*// Test get list customers*LoyaltyPointAPI.getListCustomers(**yourContext**, **shopID**, **new** LoyaltyPointAPI.OnGetListCustomersResult() {  
 @Override  
 **public void** onSuccess(ArrayList<Customer> listCustomers) {  
 Log.e(**"getListCustomers"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.e(**"getListCustomers"**, error);  
 }  
});

Trong đó:

**shopID** là id của shop cần lấy danh sách customers.

# Lấy Danh Sách Events

*// Test get list events*LoyaltyPointAPI.getListEvents(**yourContext**, **shopID**, **new** LoyaltyPointAPI.OnGetListEventsResult() {  
 @Override  
 **public void** onSuccess(ArrayList<Event> listEvents) {  
 Log.e(**"getListEvents"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.e(**"getListEvents"**, error);  
 }  
});

Trong đó:

**shopID** là id của shop cần lấy danh sách events.

# Lấy Danh Sách Awards

*// Test get list awards*LoyaltyPointAPI.getListAwards(**yourContext**, **shopID**, **new** LoyaltyPointAPI.OnGetListAwardsResult() {  
 @Override  
 **public void** onSuccess(ArrayList<Award> listAwards) {  
 Log.e(**"getListAwards"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.e(**"getListAwards"**, error);  
 }  
});

Trong đó:

**shopID** là id của shop cần lấy danh sách awards.

# Tính Điểm Tích Lũy

LoyaltyPointAPI.calculatePoint(**yourContext**, **shopID**, **cardQRCode**, **listProducts**, **totalMoney**, **new** LoyaltyPointAPI.OnCalculatePointResult() {  
 @Override  
 **public void** onSuccess(**final** ArrayList<AchievedEvent> listAchievedEvents, **int** pointFromMoney, **final int** totalPoint) {  
 // Successfully  
 }  
  
 @Override  
 **public void** onError(**final** String error) {  
 // Error  
 }  
});

Trong đó:

Input:

**shopID** là id của shop cần tính điểm tích lũy.

**cardQRCode** là chuỗi String quét ra từ QRCode trên thẻ của customer.

**listProducts** là danh sách các sản phẩm mà customer vừa mua (trên hóa đơn)

**totalMoney** là tổng số tiền trên hóa đơn

Output:

**listAchievedEvents** là danh sách những events mà người dùng đạt được (có kèm số lượng)

**pointFromMoney** là số điểm người dùng nhận được từ tổng số tiền trên hóa đơn

**totalPoint** là tổng số điểm mà người dùng đạt được (gồm điểm quy ra từ tiền trên hóa đơn và điểm đạt được từ các events mua hàng).

# Cập Nhật Điểm Tích Lũy

1. Đầu tiên gọi api tính điểm tích lũy, nhận về kết quả (danh sách events đạt được, điểm từ tiền trên hóa đơn, tổng số điểm).
2. Dùng chính kết quả đó gọi tiếp api cập nhật điểm tích lũy.

LoyaltyPointAPI.calculatePoint(**yourContext**, **shopID**, **cardQRCode**, **listProducts**, **totalMoney**, **new** LoyaltyPointAPI.OnCalculatePointResult() {  
 @Override  
 **public void** onSuccess(**final** ArrayList<AchievedEvent> listAchievedEvents, **int** pointFromMoney, **final int** totalPoint) {  
 Log.*e*(**"calculatePoint"**, **"successfully"**);  
  
 LoyaltyPointAPI.updatePoint(**yourContext**, **shopID**, **cardQRCode**, listAchievedEvents, totalPoint, **new** LoyaltyPointAPI.OnUpdatePointResult() {  
 @Override  
 **public void** onSuccess() {  
 Log.*e*(**"updatePoint"**, **"successfully"**);  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.*e*(**"updatePoint"**, error);  
 }  
 });  
 }  
  
 @Override  
 **public void** onError(String error) {  
 Log.*e*(**"calculatePoint"**, error);  
 }  
});

Trong đó:

Input:

**shopID** là id của shop cần tính điểm tích lũy.

**cardQRCode** là chuỗi String quét ra từ QRCode trên thẻ của customer.

**listProducts** là danh sách các sản phẩm mà customer vừa mua (trên hóa đơn)

**totalMoney** là tổng số tiền trên hóa đơn

Output:

**listAchievedEvents** là danh sách những events mà người dùng đạt được (có kèm số lượng)

**pointFromMoney** là số điểm người dùng nhận được từ tổng số tiền trên hóa đơn

**totalPoint** là tổng số điểm mà người dùng đạt được (gồm điểm quy ra từ tiền trên hóa đơn và điểm đạt được từ các events mua hàng).