|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document History** | | | | |
| **Version** | **Date** | **Author** | **Section** | **Changes** |
| 0.1 | 25-03-2016 | Indrajit Kumar | All | Initial draft |

InApp Purchase Integration

**DOCUMENT HISTORY**

|  |  |
| --- | --- |
| **Author** | Indrajit Kumar |
| **Approved by** |  |
| **Email Id** | Indrajit.kumar@philips.com |

# Introduction

This document provides an overview of integration E-Commerce App feature in Android mobile application.

Source Path: <http://161.85.28.146:9000/scm/git/hor-inapppurchase-android>

# Prerequisites

I. Vertical project is configured for Android Studio

II. Setting->Developer Options->Don’t Keep Activities should be unchecked. [ If Developer

Mode is on]

III. Android API version should on 10[Gingerbread]



# Library Integration

## Maven repository Integration

The easiest and preferred way to use these components is using maven. Only we need to add maven repositories in app build gradle:

allprojects {  
 repositories {  
 maven { url 'http://maartens-mini.ddns.htc.nl.philips.com:8081/artifactory/jcenter' }  
 maven {  
 url 'http://maartens-mini.ddns.htc.nl.philips.com:8081/artifactory/ext-release-local'  
 }  
 maven {  
 url 'http://maartens-mini.ddns.htc.nl.philips.com:8081/artifactory/libs-release-local-android'  
 }  
 maven {  
 url 'http://maartens-mini.ddns.htc.nl.philips.com:8081/artifactory/libs-stage-local-android'  
 }  
 flatDir {  
 dirs 'libs'  
 }  
 }  
}

## Library Integration

1. Check out the code from above path wherein you can find sample app which depends on InAppPurchase care library’s aar file : <http://161.85.28.146:9000/scm/git/hor-inapppurchase-android>
2. InAppPurchase needs other libraries to build which are as below
3. registerationApi

Please make sure all these are added to application with the latest versions along with InAppPurchase library aar file under libs folder.

1. Refer InAppPurchase dependencies along with .aar files in build.gradle as mentioned below :



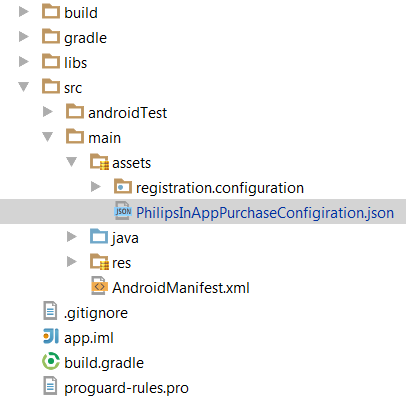
*Note: Since InAppPurchase Android was built with* ***'com.android.support:appcompat-v7:23+. version***

***so vertical app support libs should be greater or equal to mentioned versions***

dependencies {  
 *compile(name: 'localeMatch-v1.1.0', ext: 'aar')*  
 compile 'com.android.support:appcompat-v7:23+'  
 compile 'com.android.support:cardview-v7:+'  
 compile(group: 'com.philips.cdp', name: 'registrationApi', version: "5.0.0-rc.2", ext: 'aar') {  
 exclude group: 'com.android.support'  
 transitive = true  
 }  
}

1. Paste InAppPurchase related assets in assets folder from reference app:

**Note : Don’t change or rename folder structure**



Inside the PhilipsInAppPurchaseConfiguration.json, app has to configure their host and site such as:

{  
 "hostport": "acc.occ.shop.philips.com",  
 "site": "US\_Tuscany",  
}

1. Clean project



# INITIALIZATION

The library can be invoked as activity. We have IAPHandler class, which having all useful API for E-Commerce.

* 1. Create instance of IAPHandler.class.
  2. UI for InAppPurchase is theme able from vertical UIKit component.
  3. On click of Shoppingcart icon, user has to initialize the launchIAP(Context, themeIndex)
  4. In onResume() of Activity/Fragment, we should call below method for getting cart count:

mIapHandler.getProductCartCount(**this**, **mProductCountListener**);

* 1. We have two more API for “buy now” and “add to cart” i.e:

mIapHandler.addProductToCart(**this**, ctnNumber, **mAddToCartListener**);

and

mIapHandler.buyProduct(**this**, ctnNumber, **mBuyProductListener**, **DEFAULT\_THEME**);

* 1. If any error scenario is there with respect to network or InAppPurcahse component, we need to update the onFailure() of IAPHandlerListner. Eg.

**private void** showToast(**int** errorCode) {  
 String errorText = **"Unkown error"**;  
 **if** (IAPConstant.***IAP\_ERROR\_NO\_CONNECTION*** == errorCode) {  
 errorText = **"No connection"**;  
 } **else if** (IAPConstant.***IAP\_ERROR\_CONNECTION\_TIME\_OUT*** == errorCode) {  
 errorText = **"Connection time out"**;  
 } **else if** (IAPConstant.***IAP\_ERROR\_AUTHENTICATION\_FAILURE*** == errorCode) {  
 errorText = **"Authentication failure"**;  
 } **else if** (IAPConstant.***IAP\_ERROR\_INSUFFICIENT\_STOCK\_ERROR*** == errorCode) {  
 errorText = **"Product out of stock"**;  
 }  
  
 Toast toast = Toast.*makeText*(**this**, errorText, Toast.***LENGTH\_SHORT***);  
 toast.setGravity(Gravity.***CENTER***, 0, 0);  
 toast.show();  
}

* 1. Implement the IAPHandlerListner

**private** IAPHandlerListener **mAddToCartListener** = **new** IAPHandlerListener() {  
 @Override  
 **public void** onSuccess(**final int** count) {  
 **mIapHandler**.getProductCartCount(DemoAppActivity.**this**, **mProductCountListener**);  
 }  
  
 @Override  
 **public void** onFailure(**final int** errorCode) {  
//Dialog dismiss

showToast(errorCode);  
 }  
};  
  
**private** IAPHandlerListener **mProductCountListener** = **new** IAPHandlerListener() {  
 @Override  
 **public void** onSuccess(**final int** count) {  
 **if** (count > 0) {  
 **mCountText**.setText(String.*valueOf*(count));  
 **mCountText**.setVisibility(View.***VISIBLE***);  
 } **else** {  
 **mCountText**.setVisibility(View.***GONE***);  
 }  
 //Dialog dismiss   
 }  
  
 @Override  
 **public void** onFailure(**final int** errorCode) {  
 //Dialog dismiss   
 showToast(errorCode);  
 }  
};  
  
**private** IAPHandlerListener **mBuyProductListener** = **new** IAPHandlerListener() {  
 @Override  
 **public void** onSuccess(**final int** count) {  
 //Dialog dismiss   
 }  
  
 @Override  
 **public void** onFailure(**final int** errorCode) {  
 //Dialog dismiss   
 showToast(errorCode);  
 }  
};

# Reference APP link

<http://161.85.28.146:9000/scm/git/hor-inapppurchase-android>

# Notes

1. Registration is developed as separate library project. App is expected to invoke registration library on click of registration button inside consumer care support page.
2. Library should be initialized as per document or sample app otherwise library would throw runtime exception.
3. Please refer sample application for more details
4. All dependencies can be referred as it is.
5. Configuration can be followed [all sample configurations are provided in sample]