**Product Registration Android Integration**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document History** | | | | |
| **Version** | **Date** | **Author** | **Section** | **Changes** |
| 1.0 | 22-03-2016 | Yogesh HS | All | Initial draft |

|  |  |
| --- | --- |
| **Author** | Yogesh HS |
| **Approved By** |  |
| **Email Id** | Yogesh.hs@philips.com |

CONTENTS

1. Introduction 3

2. Integration 3

2.1.1 Referring from Artifactory 3

2.1.2 Why use Artifactory? 4

2.1.3 Changes at top level build.gradle 4

2.2 Referring from Local libs 7

2.2.1 Dependencies 7

3. Demo apps 7

**1. Introduction**

This document provides an overview of integration procedure for Product Registration library in android mobile applications.

Source Path:<ssh://git@atlas.natlab.research.philips.com:7999/pr/hor-productregistration-android.git>

**2. Integration**

Integration can be done in following ways.

**2.1.1 Referring from Artifactory**

Artifactory is software that The SOA and Internet development teams have been using for a while. If you have never heard of it, Artifactory is a binary repository manager. We use source control for our source code, Artifactory is version control (and more) for your binary artifacts (jar/war files, etc). Artifactory is also a place where you can put a shared library so that it is easily accessible in other projects across the enterprise.

For someone who has never used a binary repository and wants to understand the benefits of using a tool like Artifactory here is a great (short) presentation: Artifactory – Sharing Binaries the Smart Way!

**2.1.2 Why use Artifactory?**

Rebuilding from source introduces points of failure and change. By having an artifact in a binary repository you have a versioned copy of the tested artifact.

Since the binaries are version it is easy to have multiple versions of libraries so that all projects do not have to be on the same version.

Artifactory can be integrated with a CI server such as Jenkins so that clean builds are automatically published with Meta data about the build.

Artifactory has “virtual repositories” that are proxies to all of the “well known” maven repositories on the internet. This give you access to most of the public shared libraries such as Apache Commons, Spring,

Hibernate, etc..; through one conduit.

It is expensive and inefficient for every developer inside the same organization to go and retrieve remote artifacts that are shared by nature

<https://www.youtube.com/watch?v=aa4YBDUDWy0>

So we are now with Artifactory support for Production Registration Module

Steps to consume Production registration following things needs to be done from App side

**2.1.3 Changes at top level build.gradle**



Need to mention the maven repo links and dependent url link in top build.gradle Root build.gardle must contain following dependent links.

buildscript {  
 repositories {  
 maven {  
 url **'http://maartens-mini.ddns.htc.nl.philips.com:8081/artifactory/jcenter'** }  
 dependencies {  
 classpath **'com.android.tools.build:gradle:1.5.0'** classpath **'org.jfrog.buildinfo:build-info-extractor-gradle:3.1.2'** classpath **'com.github.dcendents:android-maven-gradle-plugin:1.3'** }  
  
 *// NOTE: Do not place your application dependencies here; they belong  
 // in the individual module build.gradle files* }  
}  
  
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'** }  
 jcenter()  
 }  
}

Need to mention artifactory dependences for registration in dependencies {} block

dependencies {  
 compile fileTree(dir: **'libs'**, include: [**'\*.jar'**])  
 compile **'com.android.support:appcompat-v7:23.1.1'** compile project(**':prx'**)  
  
 compile(group: **'com.philips.cdp'**, name: **'registrationApi'**, version: **'5.0.0-rc.1'**)  
 compile(group: **'com.philips.cdp'**, name: **'registrationApi'**, version: **"5.0.0-rc.1"**, ext:  
 **'aar'**) {  
 exclude group: **'com.android.support'** transitive = **true** }  
 compile **'com.philips.cdp:registrationApi:5.0.0-rc.1'** compile **'com.philips.cdp:dicommClientLib:1.0.7'**}

apply plugin: **'com.android.application'**

packagingOptions {  
 exclude **'META-INF/DEPENDENCIES.txt'** exclude **'META-INF/LICENSE.txt'** exclude **'META-INF/NOTICE.txt'** exclude **'META-INF/NOTICE'** exclude **'META-INF/LICENSE'** exclude **'META-INF/DEPENDENCIES'** exclude **'META-INF/notice.txt'** exclude **'META-INF/license.txt'** exclude **'META-INF/dependencies.txt'** exclude **'META-INF/LGPL2.1'**}

android {  
 compileSdkVersion 23  
 buildToolsVersion **"23.0.2"** defaultConfig {  
 applicationId **"com.philips.cdp.demo"** minSdkVersion 18  
 targetSdkVersion 23  
 versionCode 2  
 versionName **"0.0.4"** testApplicationId **"com.philips.cdp.demo.androidTest"** testInstrumentationRunner **"android.test.InstrumentationTestRunner"** }  
 buildTypes {  
 release {  
 minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android.txt'**), **'proguard-rules.pro'** }  
 debug {  
 debuggable **true** minifyEnabled **false** shrinkResources **true** testCoverageEnabled **true** }  
 }  
  
 packagingOptions {  
 exclude **'META-INF/DEPENDENCIES.txt'** exclude **'META-INF/LICENSE.txt'** exclude **'META-INF/NOTICE.txt'** exclude **'META-INF/NOTICE'** exclude **'META-INF/LICENSE'** exclude **'META-INF/DEPENDENCIES'** exclude **'META-INF/notice.txt'** exclude **'META-INF/license.txt'** exclude **'META-INF/dependencies.txt'** exclude **'META-INF/LGPL2.1'** }  
 lintOptions {  
 abortOnError **false** }

***Note: Demo app is available for reference along with this package***

**2.2 Referring from Local libs**

**2.2.1 Dependencies**

**1 .Philips Local Match.**

**2 .UserRegistionAPi**

**3. Philips PRX Clint**

**3.** **Demo apps**

Please find demo app in: hor-productregistration-android/Demo App/DemoProductRegistrationClient