

## Mavenització d'aplicacions Android

### A qui va dirigit

Aquest Howto va dirigit a tots aquells perfils tècnics encarregats de la integració d'aplicacions Android al SIC, pel que fa a la custòdia de codi.

### Introducció

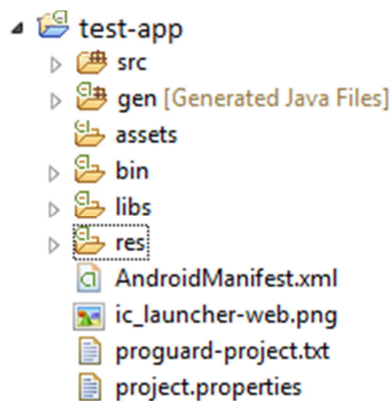
Un dels requeriments que ha de complir una aplicació per poder integrar-se completament al SIC és que estigui mavenitzada.

A més a més el SIC té com a restricció (entre altres) la no acceptació d'artefactes, és a dir, les aplicacions no poden pujar al repositori llibreries de les que depenen. Només es custòdia el seu codi font.

Actualment, les aplicacions Android que estiguin mavenitzades podran fer servir el SIC per la custòdia del codi, però està previst que en el futur es puguin fer servir també d'altres funcionalitats que ofereix el SIC integrant-les amb Jenkins i Sonar.

### Estructura aplicació Android

Normalment les aplicacions Android tenen una estructura semblant a aquesta:



- /src: conté el codi font del projecte.
- /res: conté els recursos que necessita el projecte, codi estàtic.
- /gen: conté uns fitxers que es generen automàticament quan es compila el projecte.
- /assets: conté la resta de fitxers auxiliar necessaris (configuració, dades, ...).
- /bin: conté els elements compilats i l'executable del projecte (fitxer ".apk").
- /libs: conté les llibreries auxiliars que fa servir l'aplicació (normalment en format .jar).
- AndroidManifest.xml fitxer que conté la definició en XML d'aspectes de l'aplicació.

Al empaquetar una aplicació per línia de comandament amb la SDK d'Android, per defecte, la tasca recorre la carpeta /libs incloent tots els jars al fitxer .apk resultat.

## Mavenització d'aplicacions Android

### Estructura aplicació Maven

Maven gestiona de forma automàtica les llibreries d'una aplicació resolent en temps de compilació, les llibreries i les versions que es necessiten.

Per tal de que Maven faci aquesta gestió, les aplicacions Maven tenen un fitxer **pom.xml** a l'arrel mitjançant el que es defineixen (entre altres) els següents tags:

- **ArtifactId**, **groupId** i **version**: identifiquen de forma única el projecte.
- **Repositories**: és el lloc d'on Maven extraurà les llibreries que necessita l'aplicació per construir-se.
- **Dependencies**: és on s'especifiquen les dependències o llibreries que l'aplicació necessita per construir-se.

Pel cas d'aplicacions Android es pot fer servir el Android Maven Plugin per tal de construir, desplegar i gestionar les diferents versions d'aquestes aplicacions.

Per defecte Maven accedeix al Repositori Central per obtenir els artefactes/llibreries especificades al pom.xml del projecte.

### Mavenització

A continuació s'expliquen els passos que s'han de seguir per tal de mavenitzar una aplicació Android.

1. Crear un fitxer a l'arrel del projecte anomenat pom.xml amb la dependència d'Android i amb el Android Maven Plugin:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>android.examples</groupId>
  <artifactId>test-app</artifactId>
  <version>1.0.0-SNAPSHOT</version>
  <packaging>apk</packaging>
  <name>Aplicacio de Test</name>

  <properties>
    <project.build.sourceEncoding>UTF8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <dependency>
      <groupId>com.google.android</groupId>
      <artifactId>android</artifactId>
      <version>1.6_r2</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>
```

## Mavenització d'aplicacions Android

```
<build>
  <finalName>${project.artifactId}</finalName>
  <sourceDirectory>src</sourceDirectory>
  <pluginManagement>
    <plugins>
      <plugin>

<groupId>com.jayway.maven.plugins.android.generation2</groupId>
      <artifactId>android-maven-plugin</artifactId>
      <version>3.5.3</version>
      <extensions>true</extensions>
    </plugin>
  </plugins>
</pluginManagement>

  <plugins>
    <plugin>

<groupId>com.jayway.maven.plugins.android.generation2</groupId>
    <artifactId>android-maven-plugin</artifactId>
    <configuration>
      <sdk>
        <!-- platform or api level (api level 4 = platform
1.6)-->
        <platform>4</platform>
      </sdk>
    </configuration>
  </plugin>
</plugins>
</build>
</project>
```

2. Especificar la dependència d'Android amb la versió de la plataforma amb la que ha de ser compatible.
3. A la configuració del android-maven-plugin especificar la versió de plataforma que correspongui a la versió especificada a la dependència d'Android.
4. Anar a la carpeta /libs i per cada fitxer jar identificar la llibreria amb que es correspon. Posteriorment s'hauria de cercar aquesta llibreria en algú repositori de Maven públic.

A priori, la majoria, es trobaran al Repositori Central que Maven ja fa servir per defecte.

Si calgués afegir un repositori, es farà de la següent manera:

```
<repositories>
  <id>[nom_repositori]</id>
  <url>[url_repositori]</url>
</repositories>
```

5. Per cada llibreria identificada al pas anterior, afegir al fitxer pom.xml una dependència especificant:

- ArtifactId: Nom de la llibreria
- GroupID: Nom del grup al que pertany la llibreria
- Version: Versió de la llibreria

## Mavenització d'aplicacions Android

Cada llibreria/dependència s'ha d'afegir dins del tag <dependencies> de la següent manera:

```
<dependency>
  <artifactId>[nom_artefacte]</artifactId>
  <groupId>[nom_grup]</groupId>
  <version>[versio]</version>
</dependency>
```

6. Una vegada afegides totes les dependències, construir el fitxer apk del projecte amb el comandament:

```
mvn clean install
```

7. Per desplegar el fitxer apk a un dispositiu connectat, fer servir el comandament:

```
mvn android:deploy
```

Per a poder realitzar la custòdia de codi al SIC és necessari haver executat els 5 primers punts. En un futur es preveu que el servei Jenkins disponible al SIC sigui capaç de construir aquestes aplicacions per a realització d'anàlisi de codi.