公司放置android组件svn地址

<https://192.168.10.15:8443/svn/platform/trunk/android/>

code文件夹：统一为放置组件源码

doc文件夹：统一放置文档，设计文档请跟你的组件名一致。

命名规定：

gruoupid=“com.yuexunit.android.library”

artifactid=“library-你的组件名”

设计文档名=“library-你的组件名”

1. 安装JDK

Android SDK

Ecplise : ADT插件、Maven插件、Svn插件、Android for Maven Ecplise

Maven3.2+

下载svn - doc目录下的maven-android-sdk-deployer-master，解压安装你所需的android系统版本

（建议安装Ecplise j2ee版本，自带了maven插件，自行配置下Manven3.0+路径即可）具体参考书籍：Maven实战\_基于Maven3(许晓斌)

1. 配置环境变量
2. 创建环境变量

JAVA\_HOME

ANDROID\_HOME

MAVEN\_HOME

1. 配置path

%ANDROID\_HOME%\platform-tools;

%ANDROID\_HOME%\tools;

%JAVA\_HOME%\bin;

%MAVEN\_HOME%\bin;

1. 配置
2. 在局部setting.xml配置发布构件的仓库所需的认证信息（公司内部私服）

<servers>

<server>

<id>nexus-release</id>

<username>admin</username>

<password>yuexun90</password>

</server>

<server>

<id>nexus-snapshots</id>

<username>admin</username>

<password>yuexun90</password>

</server>

</servers>

1. 在你开发的项目构件pom.xml中配置发布构件的仓库（公司内部私服），目前还未对发布版本及快照版本进行明确管理，所以发布到同一个仓库

<distributionManagement>

<repository>

<id>nexus-release</id>

<name>Nexus Release Repository</name> <url>http://192.168.10.90:8081/repository/snapshots</url>

</repository>

<snapshotRepository>

<id>nexus-snapshots</id>

<name>Nexus Snapshots Repository</name> <url>http://192.168.10.90:8081/repository/snapshots</url>

</snapshotRepository>

</distributionManagement>

1. 在局部setting.xml配置依赖下载仓库也可以在你开发项目的pom中配置（公司内部私服）

<mirrors>

<mirror>

<id>nexus</id>

<mirrorOf>\*</mirrorOf>

<name>所有仓库的镜像</name>

<url>http://192.168.10.90:8081/repository/public</url>

</mirrors>

1. 在局部sertting.xml 中配置，开启对中央仓库的快照版本的支持

<profiles>

<profile>

<id>central</id>

<repositories>

<repository>

<id>central</id>

<name>Central Repository</name>

<url>https://repo.maven.apache.org/maven2</url>

<layout>default</layout>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>central</id>

<name>Central Repository</name>

<url>https://repo.maven.apache.org/maven2</url>

<layout>default</layout>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</pluginRepository>

</pluginRepositories>

</profile>

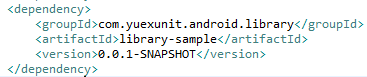
</profiles>

<activeProfiles>

<activeProfile>central</activeProfile>

</activeProfiles>

1. 创建无res资源引用的构件
2. 创建maven-archetype-quickstart原型的项目，删除test目录，修改pom.xml即可，最后记得执行deploy命令。具体可参考svn – code下的模板sample-library
3. 创建android-quickstart原型项目，引用输出的构件。只需在你的项目中加入依赖配置，具体可参考svn – code下的项目simple-library



1. 创建带res资源引用的构件（创建于引用较为复杂无法完全自动化）参考：http://xiaoyaozjl.iteye.com/blog/2007900
2. 创建android-quickstart原型项目，修改pom配置packaging必须是apklib，最后记得执行deploy命令。具体可参考svn – code 下的模板library-res。
3. 创建android-quickstart原型项目，引用带res的构件，需要做一些手动性的操作

* 手动安装apklib到本地仓库



* 在ADT工作空间下，新建groupId同名目录
* 将apklib的内容解压到该目录；将改pom文件也拷贝到该目录，名为pom.xml
* 在命令行运行 :android update project -t target -p path
* 在Ecplise导入这个maven项目，如果导入后报错，就按上文配置一下pom.xml并Update Project
* 在App项目的pom内依赖这个apklib项目，如果看到R文件生成，说明成功了