

xctool 更换为 xcodebuild

因打包机器的 xcode 升级为 xcode8.2, xctool 将不再支持 xcode8 + , 故之前自动化构建脚本将需要改变

主要的更改文件 `auto-build.sh`

搜索到大概 44 行左右, 可见到起始

```
echo ===== 打包 STEP2.生成 archive 文件 =====
```

将 xctool 的命令进行替换

```
echo ===== 打包 STEP2.生成 archive 文件 =====
# # 更新子模块代码
# if [ "$TARGET_NAME" == "ZhongXinJianTou_Phone" ]; then
#     # 以下为中信建投需要加入的 SDK
#     # cp -r /Users/kuan/vendors/BonreeAgent ${CURRENT_WORK_PA
TH}/KDS_Phone/Vendors/
#     cp -r /Users/kuan/vendors/ZXJTOpenAccount ${CURRENT_WORK_
PATH}/KDS_Phone/Vendors/
# fi

archiveFilePath=./build/$TARGET_NAME.xcarchive
logFilePath=~/Desktop/iOS_log_$(date +%G%m%d%k%M)
# -reporter pretty:${logFilePath}
# 需要输出完整的编译 log, 注释掉 -reporter phabricator \ 这行
xcodebuild clean -workspace $XCODE_WORKSPACE_NAME.xcworkspace \
    -scheme $TARGET_NAME \
    -configuration ${buildConfig}

xcodebuild archive -workspace $XCODE_WORKSPACE_NAME.xcworkspace \
    -scheme $TARGET_NAME \
    -configuration ${buildConfig} \
    -archivePath ${archiveFilePath} | xcpretty -c | tee logFile
Path

if [ "$?" -ne "0" ]; then
    echo "archive error 停止自动构建"
```

```

    rm -rf ./DailyBuild
    exit 1
fi

echo ===== 打包 STEP3.生成ipa文件 =====
ipaExportPath="${IPA_FILE_PATH}/${TARGET_NAME}.ipa"          # ipa
    输出路径
xcodebuild -exportArchive -exportFormat IPA \
    -archivePath ${archiveFilePath} \
    -exportPath ${ipaExportPath} \
    -exportProvisioningProfile $PROFILE_NAME | xcpretty -c

```

结束行为

```

if [ "$?" -ne "0" ]; then
    echo "export ipa error 停止自动构建"
    rm -rf ./DailyBuild
    exit 1
fi

```

修改后的完整的 auto-build.sh 脚本如下

```

# 以下为 Xcode 自动打包脚本
# 1. 使用 facebook 开源的 xctool 打包, 需要打包的 mac 机安装 xctool
# 安装方式: brew install xctool (https://github.com/facebook/xctool)
# 2. 传入的 打包证书的描述文件的文件名 Mobile_Provision_File_Name, 必须此证书在 打包的 Mac 机上安装过一次
# 3. 测试此文件: sh ./autobuild.sh KDS_Phone PackedFileName DailyBuild ZLDevelopment0705

# target Name: 券商对应的 target 名 (传入)
TARGET_NAME=$1
# 注: IPA_FILE_PATH 是打出的包的存放目录 (传入)
IPA_FILE_PATH=$2
# 打包方式, 默认为DailyBuild
# 注: 可选 Release, DailyBuild, Debug
buildConfig=$3
# 描述文件名称 (需要打包的电脑上有安装)
PROFILE_NAME=$4

if [ -z "$buildConfig" ]; then
    echo "未指定打包方式, 默认打 DailyBuild 包"

```

```

    buildConfig="DailyBuild"
fi

if [ -z "$PROFILE_NAME" ]; then
    echo "打包证书未指定, 默认使用 ZLDevelopment0705"
    Mobile_Provision_File_Name="ZLDevelopment0705"
fi

# 每次打包前先删除旧的
rm -rf "$IPA_FILE_PATH"
mkdir "$IPA_FILE_PATH"

XCODE_WORKSPACE_NAME="KDS_Phone" # 项目的 workSpace name (目前是固定的)
keychainPw="123" # 打包 Mac 的密码 (非必须, 仅第一次需要)

echo ===== 打包 STEP1.更新第三方pod依赖 =====
sh ./auto-make-podfile.sh $TARGET_NAME
pod install --verbose --no-repo-update
# pod update

# echo "=====解锁钥匙串===== "
security list-keychain
security unlock-keychain -p $keychainPw ${HOME}/Library/Keychains/login.keychain

echo ===== 打包 STEP2.生成 archive 文件 =====
# # 更新子模块代码
# if [ "$TARGET_NAME" == "ZhongXinJianTou_Phone" ]; then
#     # 以下为中信建投需要加入的 SDK
#     # cp -r /Users/kuan/vendors/BonreeAgent ${CURRENT_WORK_PATH}/KDS_Phone/Vendors/
#     cp -r /Users/kuan/vendors/ZXJTOpenAccount ${CURRENT_WORK_PATH}/KDS_Phone/Vendors/
# fi

archiveFilePath=./build/$TARGET_NAME.xcarchive
logFilePath=~/Desktop/iOS_log_$(date +%G%m%d%k%M)
# -reporter pretty:${logFilePath}
# 需要输出完整的编译 log, 注释掉 -reporter phabricator \ 这行
xcodebuild clean -workspace $XCODE_WORKSPACE_NAME.xcworkspace \
    -scheme $TARGET_NAME \
    -configuration ${buildConfig}

xcodebuild archive -workspace $XCODE_WORKSPACE_NAME.xcworkspace \

```

```

-scheme $TARGET_NAME \
-configuration ${buildConfig} \
-archivePath ${archiveFilePath} | xcpretty -c | tee logFile
Path

if [ "$?" -ne "0" ]; then
    echo "archive error 停止自动构建"
    rm -rf ./DailyBuild
    exit 1
fi

echo ===== 打包 STEP3.生成ipa文件 =====
ipaExportPath="${IPA_FILE_PATH}/${TARGET_NAME}.ipa"          # ipa
输出路径
xcodebuild -exportArchive -exportFormat IPA \
    -archivePath ${archiveFilePath} \
    -exportPath ${ipaExportPath} \
    -exportProvisioningProfile $PROFILE_NAME | xcpretty -c

if [ "$?" -ne "0" ]; then
    echo "export ipa error 停止自动构建"
    rm -rf ./DailyBuild
    exit 1
fi

# 移除临时文件
rm -rf ./DailyBuild

```

可参考 [脚本变化](#)