

# Fastlane 自动化 App 发布或部署

程文锋

个人博客: <http://www.jianshu.com/u/641ade6037d6>

# 1.Fastlane介绍

# 什么是Fastlane?

fastlane是一种配置iOS和Android自动化部署和发布的最简单的方法之一。它可以简化一些乏味、单调、重复的工作，像代码签名、截图以及应用发布等操作。你只需要在终端执行一段代码就能自动生成ipa文件并上传到itunes Connect，还可以自动进行应用截图并上传，只需要点击一个按钮就可以将 App 发布到 App Store。

Fastlane的优点：

- 使用脚本完成大部分非开发操作
- 自动化App开发配置和部署，一键操作
- 操作简单，耗时少，缩短项目发布周期

官方文档地址：<https://docs.fastlane.tools/>

# Fastlane的主要工具



Upload screenshots, metadata and your app to the App Store using a single command.



Automate taking localized screenshots of your iOS app on every device



Quickly put your screenshots into the right device frames



Automatically generate and renew your push notification profiles



Because you would rather spend your time building stuff than fighting provisioning



Create new iOS apps on iTunes Connect and Developer Portal using the command line



Automatically create and maintain iOS code signing certificates



The easiest way to run tests of your iOS and Mac app



Building your iOS app has never been easier



Easily sync your certificates and profiles across your team using Git

# 传统iOS开发应用程序和设备配置流程

- Mac: 从证书颁发机构请求证书
- Developer官网: Registering an APP ID
- Developer官网: Certificates (Development, Production)
- Developer官网: Devices (UDID)
- Developer官网: Provisioning Profiles (Development, Distribution)
- Mac: 安装相应的Certificates, Provisioning Profiles

## 2.Xamarin With Fastlane



# 安装Fastlane

- 安装Xcode command line tools

在终端输入命令: `xcode-select --install`

- 安装fastlane tools
  - 下载解压fastlane tools 文件
  - Rubygems 命令行安装

安装教程:

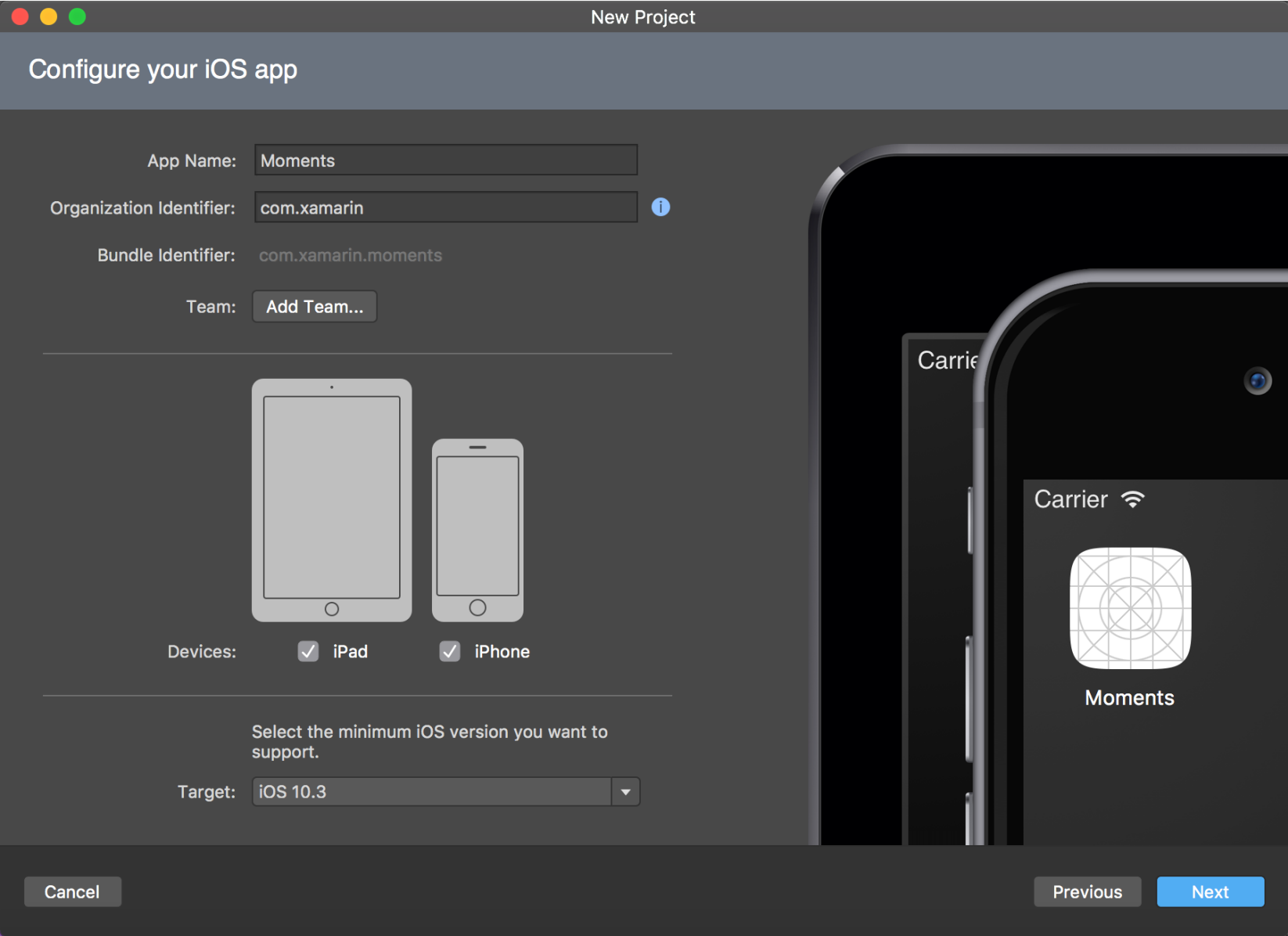
[https://developer.xamarin.com/guides/ios/deployment,\\_testing,\\_and\\_metrics/provisioning/fastlane/#Installation](https://developer.xamarin.com/guides/ios/deployment,_testing,_and_metrics/provisioning/fastlane/#Installation)

# Visual Studio for Mac With Fastlane

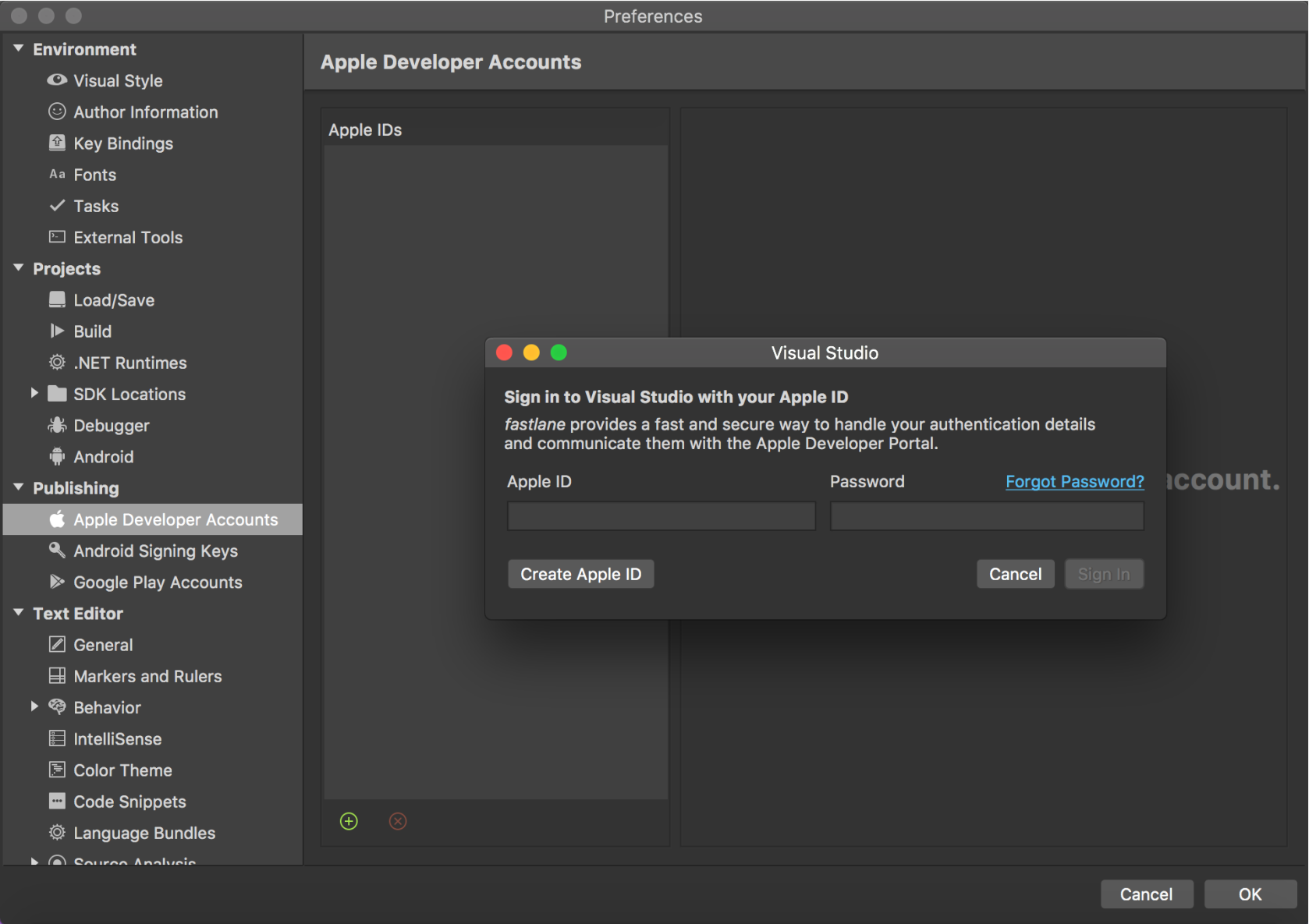
- 新建项目
- 添加开发者账号并启用fastlane和开发者账户的集成
- 创建开发者证书(Development Certificates)
- 创建配置文件(Provisioning Profile)
- 添加调试设备(Device)
- 创建发布证书(Production Certificates)



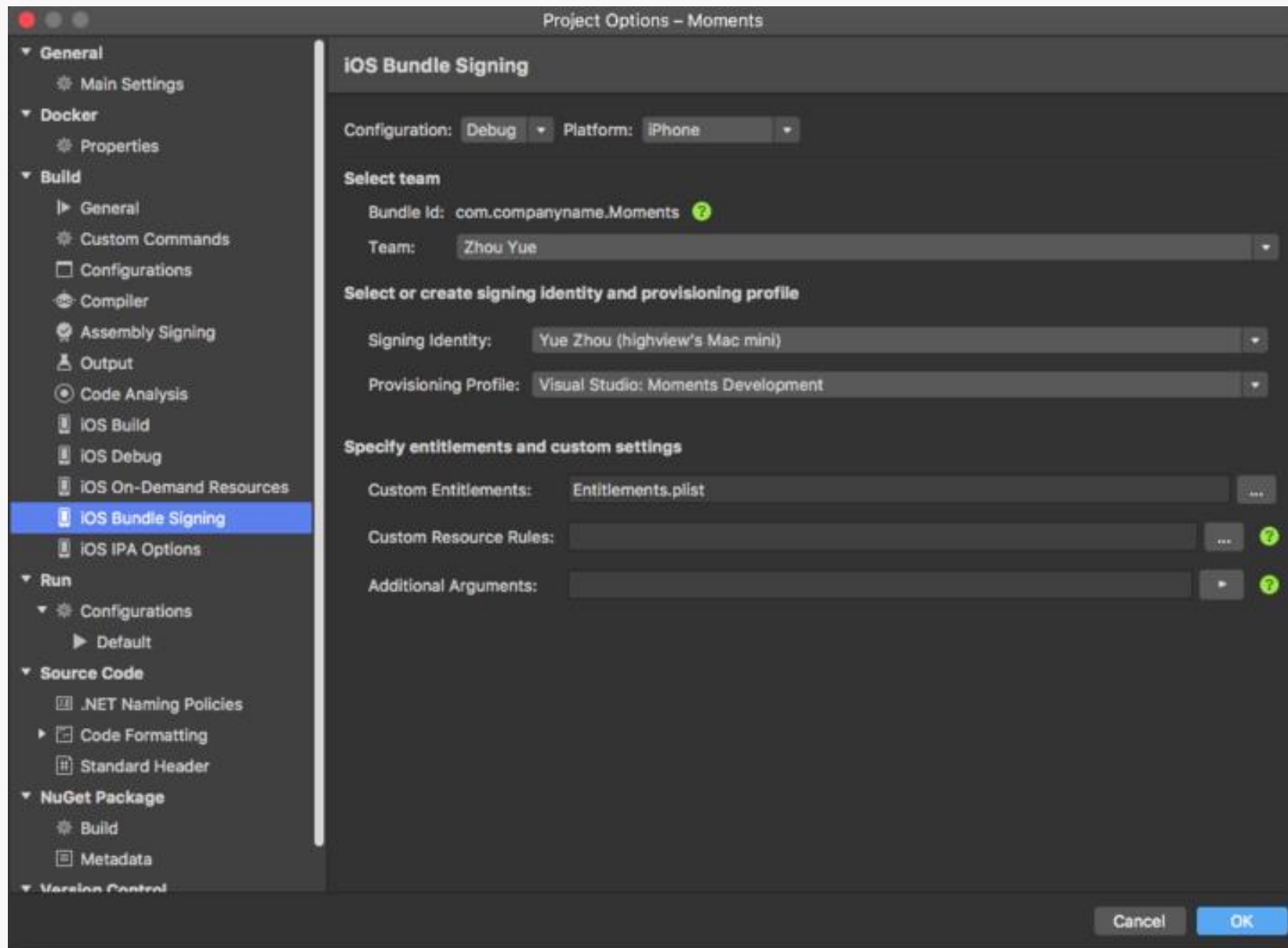
点击Add Team



# 选择Apple Developer Accounts，添加开发者账号



选择Signing Identity->Create signing identity



# Fastlane组件介绍

- 测试
  - scan=>自动运行测试工具，并且可以生成漂亮的HTML报告。
- 证书，配置文件
  - cert => 自动创建管理iOS代码签名证书
  - sigh => 创建、更新、下载、修复Provisioning Profile的工具。
  - pem => 自动生成、更新推送配置文件
  - match => 一个新的证书和配置文件管理工具。
- 截图
  - snapshot => 用Xcode7推出的UI test功能实现自动化截图
  - frameit => 可以把截的图片自动套上一层外边框

- 编译
  - gem=>fastlane家族的自动化编译工具，和其他工具配合的非常默契。
- 发布
  - produce =>如果你的产品还没在iTunes Connect或者Apple Developer Center建立，produce可以自动帮你完成这些工作
  - deliver =>自动上传截图，APP的元数据，二进制(ipa)文件到iTunes Connect。
- TestFlight管理
  - pilot =>管理TestFlight的测试用户，上传二进制文件
  - boarding =>建立一个添加测试用户界面，发给测试者，可自行添加邮件地址，并同步到iTunes Connect
- Android
  - supply =>自动上传到Google Play工具
  - screengrab =>Android的自动截图工具

# 使用方式

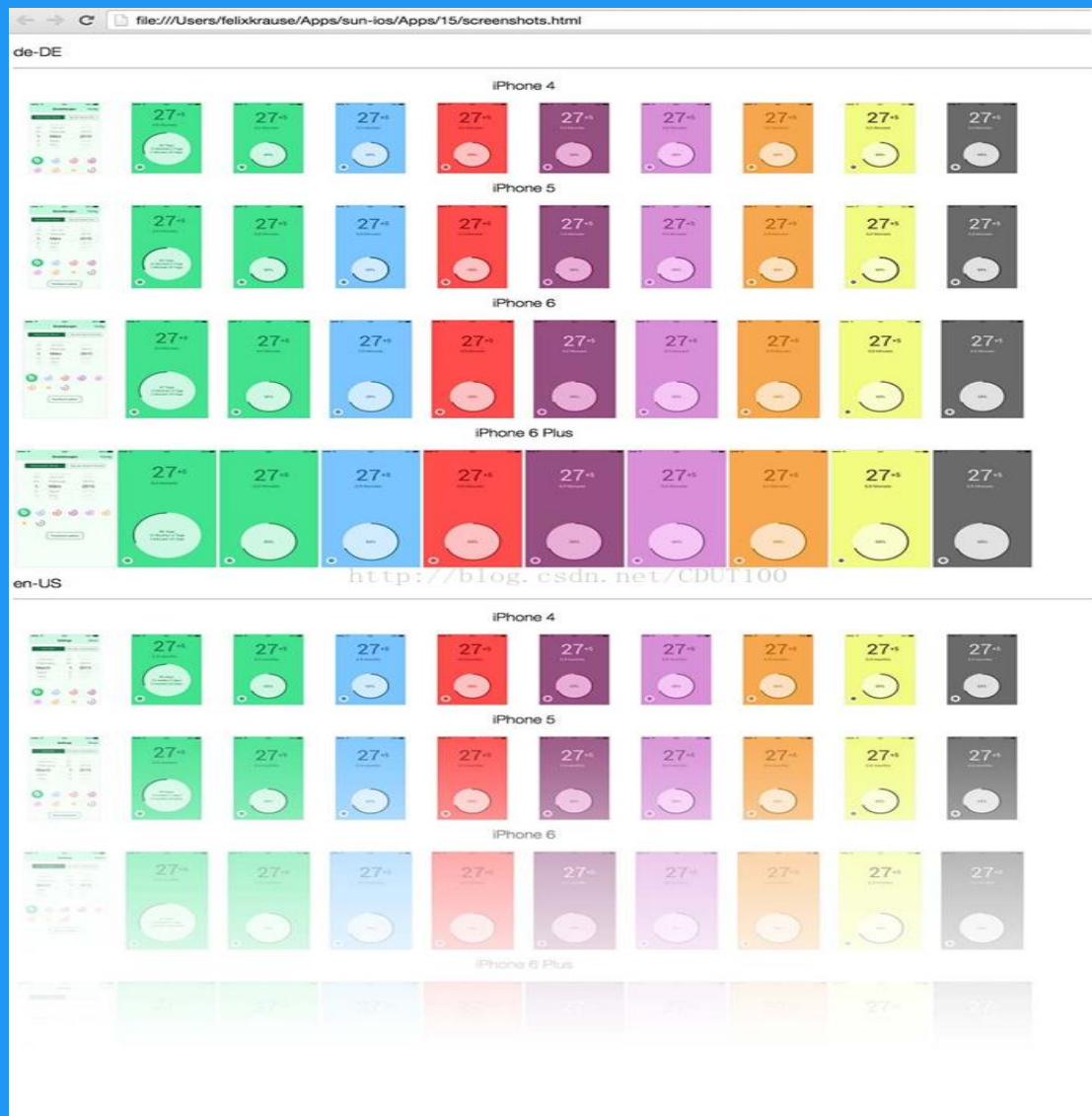
通过 fastlane 内置的工具，你可以从项目工程中创建fastLane文件开始，在fastLane文件里定义了Beta部署或App Store发布过程，例如上传app到TestFlight 和 App Store:

```
lane :beta do
  increment_build_number
  gym           # Build your app
  testflight    # Upload to TestFlight
end
```

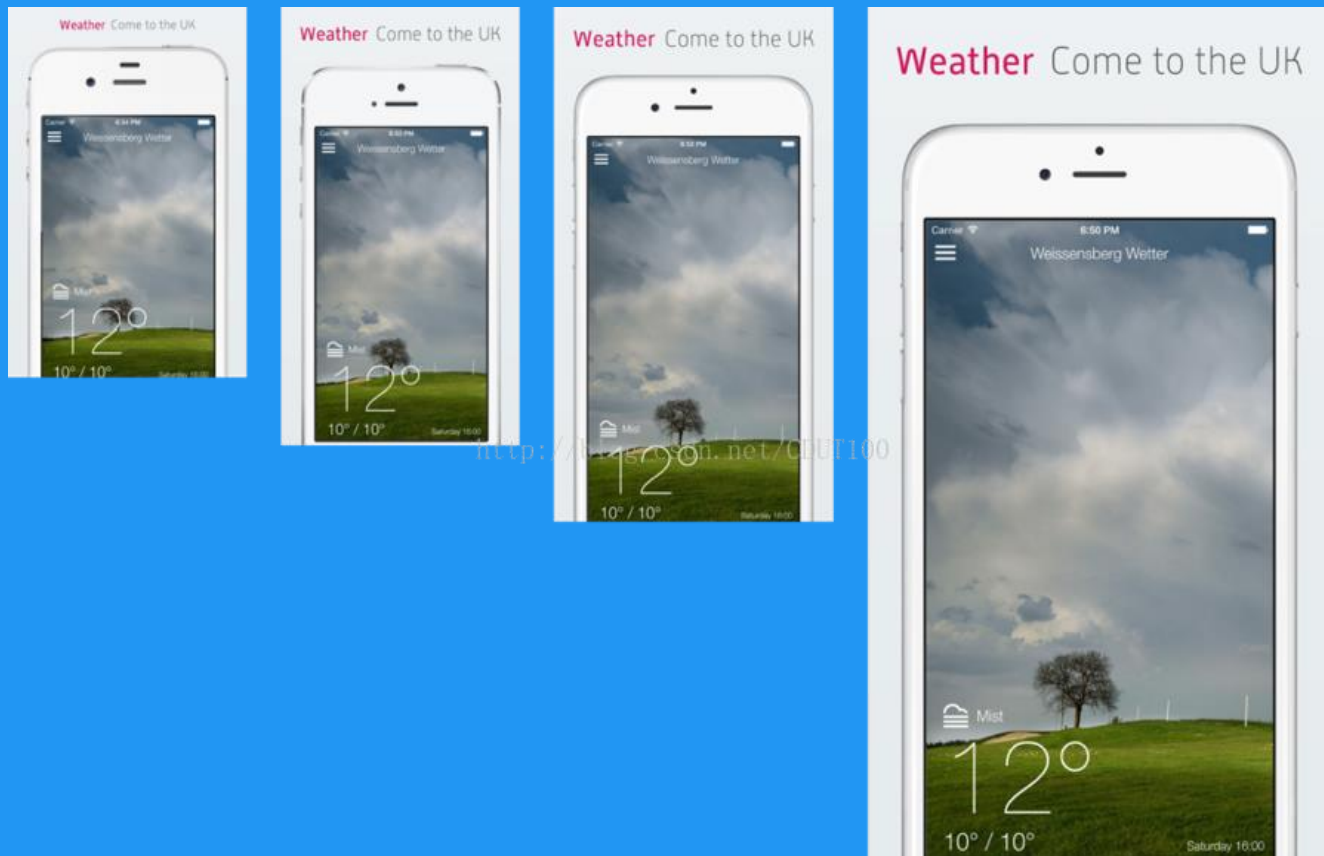
```
lane :appstore do
  snapshot      # Generate screenshots for the App Store
  gym           # Build your app
  deliver       # Upload the screenshots and the binary to iTunes
  slack         # Let your team-mates know the new version is live
end
```

<http://blog.csdn.net/CDUT100>

## 自动生成截图



## 自动生成带边框截图



# 相关博客和资源链接

- Easy iOS App Provisioning with fastlane and Visual Studio for Mac:
- [https://blog.xamarin.com/easy-ios-app-provisioning-fastlane-visual-studio-mac/?utm\\_medium=social&utm\\_campaign=blog&utm\\_source=twitter&utm\\_content=fastlane](https://blog.xamarin.com/easy-ios-app-provisioning-fastlane-visual-studio-mac/?utm_medium=social&utm_campaign=blog&utm_source=twitter&utm_content=fastlane)
- fastlane 教程: 入门: <http://blog.csdn.net/kmyhy/article/details/52441681>
- 使用fastlane实现iOS持续集成: <http://everettjf.com/2015/09/08/ios-ci-with-fastlane/>
- 使用fastlane gym/xctool编写ipa打包脚本: <http://www.jianshu.com/p/54ab07f2e63b/>
- fastlane tools 下载链接: <https://download.fastlane.tools>





感谢聆听

Thank you for hearing me!