**苹果应用程序发布流程**

**如何将应用上传到 App Store 的详细流程介绍**

http://www.59186618.com

2011年3月

目录

[一、 发布前的测试 3](#_Toc289329843)

[二、 发布的具体步骤 4](#_Toc289329844)

[三、 发布工具-如何使用iTunes Connect 7](#_Toc289329845)

[1) 销售趋势报告（Sales/Trend Reports） 8](#_Toc289329846)

[2) 合同、税费及银行信息（Contracts， Tax， & Banking Information） 8](#_Toc289329847)

[3) 财务报告 （Financial Reports） 10](#_Toc289329848)

[4) 用户管理（Manage Users） 10](#_Toc289329849)

[5) 请求用于推销的代码（Request Promotional Codes） 11](#_Toc289329850)

[6) 管理自己的In App Purchases（Manage Your In App Purchase） 11](#_Toc289329851)

[7) 管理自己的应用程序（Manage Your Applications） 11](#_Toc289329852)

[a) 概要数据 14](#_Toc289329853)

[b) 对程序的评级 16](#_Toc289329854)

[c) 上传数据 16](#_Toc289329855)

[d) 定价 17](#_Toc289329856)

[e) 地域选择 17](#_Toc289329857)

[f) 检查页面 17](#_Toc289329858)

网上已经有一些文档介绍 IPhone程序开发了，但是很少有文档介绍应用程序发布流程的。从今天开始，我就开始为大家介绍怎样将开发完成的应用程序打包发布到苹果的在线商店上去。

# 发布前的测试

程序的开发过程，我在这篇文章中就不细说了，但是，我想简单提一下应用程序上线前的发布测试。因为，程序的作者要保证所提交的程序是健壮、稳定的，如果把程序的测试寄希望于 Apple Store的工作人员，那么将会大大浪费我们的时间，一般而言，一个程序提交到 Apple Store后，大约有一个星期的时间才能从 Apple那里得到明确的结论： 允许或者拒绝上线，当然，拒绝的话会给出比较具体的解释，如果靠 Apple的反馈来修正自己的程序，那么这个周期就太长了，毕竟我们申请的证书有效期是一年而已。我们希望在这一年里，能够充分的利用这个证书，通过一个证书提交尽量多的程序。

好了，现在我们说说发布测试的事情。我想，多数程序员开发时是在模拟器上进行调试的，但是模拟器毕竟是运行在 PC上的一个应用程序，即便它能模拟真机的多数行为，但对于特定的操作（如加速计的使用，多点触摸等），内存的模拟等情况，它还是无法和真机相比！所谓的发布测试，就是在提交之前，一定要让自己开发的程序真正的运行在真机中进行测试，这样会发现很多你之前可能没有注意到的潜在问题，比如：内存泄露、图片格式问题（有些图片格式在模拟器上显示好好的，但是放在了真机上就是无法显示）等等。

那么怎样做发布测试呢？答案就在 iPhone Dev Center里面，用你自己的 Apple ID登录，然后进入到 iPhone Developer Program Portal页面后，按照如下步骤完成相关的操作（本文假设你对开发过程已经熟悉了）

1 进入页面上的 Certificates页，一般我们在开发时，只是申请 Development开发者证书就够了，但是如果我们想要把应用程序放到真机上运行的话，就必须在 Distribution发布页上，在此页上申请创建一个证书，用于发布，具体过程本文不做具体描述了，可以参考该页上的帮助，写的还是很清晰的。

2 在 App IDs再申请一个 ID用于发布，这一步是可选的，如果你在开发阶段就已经考虑到了发布的事情，那么当初申请的那个用户开发的 ID也是可以用于发布的。此处建议，申请一个最后字段为“ \*”的 ID，这样可以用同样的 ID，提交一套应用程序，而无需为每个应用都申请单独的 ID。 例如： 3452KHK343.com.youapptype.\*， 这样当你写应用程序时，第一个应用程序的 ID就可以是 3452KHK343.com.youapptype.app1，而第二个应用的 ID就可以是 3452KHK343.com.youapptype.app2，依次类推。

3 进入 Provisioning页面，在 Distribution标签页下面，我们同样需要为发布而创建新的 Profile，这个 profile在编译应用程序时是需要指定的。具体的步骤可以参考该页的帮助，只提示一点，每个 profile对应于一个 App ID，也就是说，刚才创建的 App ID在此处应该生成一个相对应的 Profile。

4 最后就是发布 Distribution了，在这一页中，有 Prepare app， App Store和 Ad hoc三个主要的标签页。 上述所讲的内容，其实也都包含在了 Prepare App页中的内容了，只是我把发布测试中比较关键的几步提取了出来而已，这里还是建议开发人员耐心的读完该页上的介绍，这样会对加深对编译创建程序的理解。而 Ad Hoc页介绍的是分布式发布的情况，比如应用程序做好了，不想放到 App Store上，或者需要多个设备进行测试，那么就需要用到 Ad Hoc这种发布方式了，具体过程请参考该页的帮助。本文主要介绍 App Store页上的内容，也就是主要介绍怎样把应用程序发布到苹果的在线商店中，毕竟如果向通过自己的劳动挣钱的话，还是要上架销售啊：）。

# 发布的具体步骤

首先在这里需要明确声明一下，本文所涉及的具体内容，都是基于苹果的官方文档《 iTC\_Dev\_Guide\_5.0.pdf》，有英文好的，或者觉得我说的不够明白的地方，请自行参阅该英文文档。

现在言归正传，将应用提交到 App Store需要通过 iTunes Connect这个基于 Web的工具。 用户可以通过自己的 Apple ID登录到 iPhone Developer Program后，通过页面右侧的连接可以进入到 iTunes Connect界面。在这里，用户可以管理自己的银行帐号，应用的版本维护等等。下面我们会具体地加以介绍。

在进入 iTunes Connect之前，要做好以下几点准备工作：

1 首先需要保证我们的应用程序已经为发布准备好了编译成功到 app文件。

这可以通过 XCode提供的 Build Result来检查，如下图 1所示：

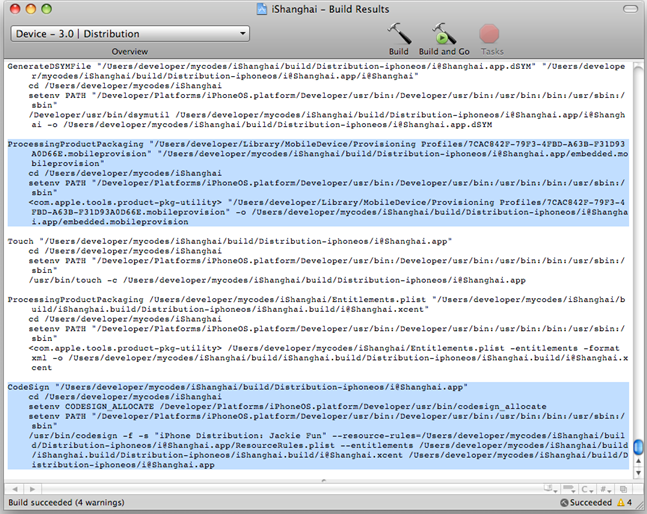


图 1 Build Result

在这个窗口中，用户必须找到“ embedded.mobileprovision”和“ CodeSign”到字样，并且要确保这两行中没有错误，尤其需要注意，这两行中所显示地目标路径是否正确，一般情况下，为编译发布的 app会放在一个 …/Distribution-iphone/目录下。开发人员一定要注意这一点，这和 WM、 Smbian等不同，不是用的 release目录下的文件！

此处，我还想提醒一句的就是，在编译最终 app的时候，要保证把程序的 icon图标也编译进去了，而且此图标必须是 57\*57的 png图片，其他格式的图片是不被接受的，即便编译成功了，也会在真机显示时出问题！

2 收集应用程序相关的信息

这部分内容之所以需要提前准备，是因为在通过 iTunes Connect提交应用时，要求必须填写的信息，主要包括以下的几方面内容：

应用程序的描述

适用的设备种类（ iPhone或者 iPod touch）

应用的类别，比如游戏、旅游、教育等等

版本

版权

支持 URL

支持 Email

等等。

关于如何填写这些内容，我们会在下面的章节结合 iTunes Connect具体介绍。

3 准备一个大图标

这个图标要求 512\*512的 JPG或者 TIFF格式的图片， png格式是不被接受的。它主要是用于在 App Store里面显示你的应用程序。当然，图片的内容要和那个小图标一样，但要注意，不要简单的把小图标简单的放大，那样的话，显示效果会很差，从而会影响到应用的销量：）。

此外，我们在看 iPhone上已经安装的那些应用时，会发现桌面的图标样子是经过渲染的（比如：图标变成了圆角、上方有高亮显示），请注意，这个渲染的效果 iPhone手机完成的，在做图片时无需考虑这部分内容。

 4 准备一些应用的截图

这其实是很关键的内容，多数情况下，当消费者在 iTunes上搜索到某个应用时，他最感兴趣的就是这些截图了，从截图中，他可以初步了解该程序的主要功能，从而会对其是否会购买这个商品起到很大的影响！

基于上述的原因，我们应该尽量设计漂亮、吸引人的程序界面，并且在提交前，一定要把程序最吸引人的截图提交上去，以期获得较高的下载量。

对于截图的要求，也很简单，根据程序的风格，用户可以提供水平及垂直两种格式的图片。

对于垂直的图片，要求其大小为：

最小 320\*460， 最大 320\*480

这是因为在全屏幕显示的时候，高 20的状态条会被隐藏。

而对于水平的图片，其大小要求为：

最小 480\*300， 最大 480\*320

同样也是因为状态条的因素，有 20个高度的差异。

另外，对于图片的格式也是要求为 jpg或者 TIFF格式， png格式是不行的！

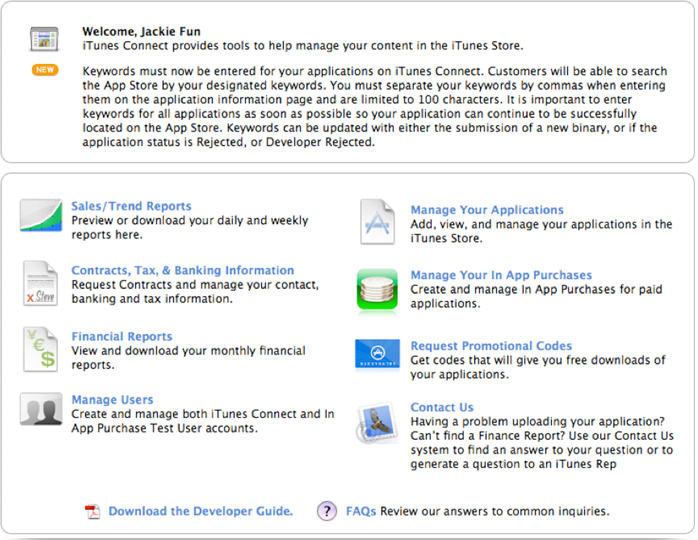
提示：在手机上截屏的方法是：

按住 Home键的同时，按下 Power键，就会将当前显示存放在 iPhone的照片库中，用户可以通过 Mac的 iPhoto工具，将手机上的截图导入到 PC上。

好了，此篇到这里，我们基本上就把需要提前准备的东西都已经准备完成了。后面的文件会开始介绍 iTunes Connect的使用。

# 发布工具-如何使用iTunes Connect

使用Apple ID登录iTunes后，会进入到其Home页面，其中包含了以下几个主要功能项：



下面我们来一项一项的介绍其具体的功能。

## 销售趋势报告（Sales/Trend Reports）

在这个栏目下，用户可以查看自己的应用程序每天、每周的销售报告。 当然用户可以将结果下载，但其格式是text的。

由于此栏目中的数据具有实时性，因此不能作为财务报表使用。另外，只有团队管理员或财务人员才能查看这项内容。

对普通开发者而言，这一项内容此时不必关心，毕竟我们的程序还没有提交呢，哪里有销售数据呢。 而且，即便是我们的程序成功的提交了，如果没有人下载，那也是没有什么数据可供查看的：）

## 合同、税费及银行信息（Contracts， Tax， & Banking Information）

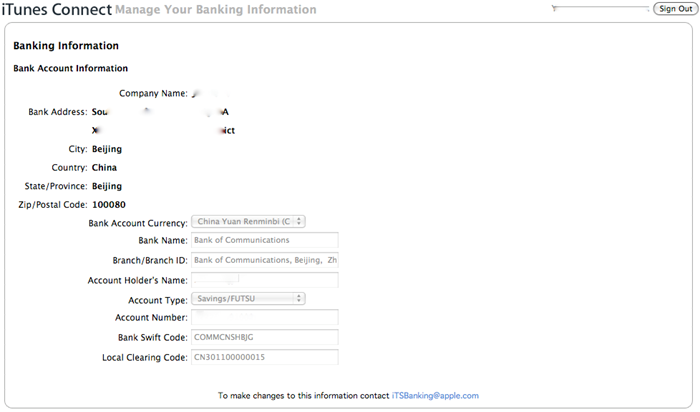
这项很明确了，如果开发人员想收钱的话，那就得认认真真到填写这部分内容了，嘿嘿。

正确提交了相关信息后，该页面应该是如下显示的。



填写过程中，会需要用户提供银行相关的信息代码，这部分内容一定要认真填写，不能有错哦，否则怎么收钱呢，呵呵。

银行的信息包含如下内容：



银行地址为开户行的地址，Branch ID为分行的名称，这都可以在银行的网站上找到。Account Holder’s Name 这一项是用户的拼音名称，需要和开户时填的一样啊！银行帐号一般就选Savings/FUTSU储蓄类型的就可以，这个根据自己的实际情况而定。

最后两个比较特殊，Bank Swift Code和Local Clearing Code这两个应该银行间识别用的，也就是说Apple需要知道这两个才能向你的帐户里转账！

这可以通过银行的热线获得，或者从网上搜索。

对于交通银行，Bank Swift Code可以从如下的网址上查的：

<http://www.bankcomm.com/BankCommSite/cn/node/beijin.jsp?id=1215050573100&type=CMS.STD&column=1131428070100&miniset_column=1131428070100>

而Local Clearing Code建议电话询问，或者自己在网上找吧。

## 财务报告 （Financial Reports）

在这一个功能下面，用户可以看到每月的财务报告，其中包含了本月下载的次数，日期等等。

注意： 只有管理员或者财务人员才能查看此页，并且只有当帐户中漫150美元后，苹果才会付款！

## 用户管理（Manage Users）

用户管理涉及到两种用户，iTunes Connect帐户，和App Purchase Test帐户，前者很好理解了，就是登录iTunes Connect时用的帐号；后者则是一种测试用户，这种用户可以测试程序的程序内购买功能。也就是说，如果开发的程序具备In App Purchase功能，那么就需要这种特殊的帐号加以测试。

那么这个In App Purchase功能什么时候使用呢？其实也很容易理解，比如开发者为了吸引消费者下载，那么可能首先提供了一个功能有限的免费版本，当消费者下载了这个免费版本后，试用完觉得很有意思，那么就可以在不退出程序的情况下，支持用户实时的购买付费版本，这样方便了用户的同时，尽量的提高收费版本的下载量。一般情况下，对于相对简单的应用程序，可以不用这个功能，开发者可以先提供一个免费的版本（功能有限），然后再提供一个功能完整的收费版本，分别提交到Apple Store，可以在免费的版本中，适当的位置加上提示信息，比如提示用户如果希望得到完整的功能体验，那么请下载另外的版本等，这种方式同样能够达到相同的目的。至于选用哪种，就由开发者自己选择吧。个人建议，如果想快速的发布，那么就简单一些，先不需要考虑这个In App Purchase功能，先把免费的版本放上去，再放一个完整的收费版本，随后，再陆续推出相应的In App Purchase版本，这样程序自身也有一个升级演进的过程，同时也给老用户一种不断的新鲜感。

至于之前提到的iTunes Coonect帐号类型，因为很明确，此处就不多少了。只简单的提一下，这种用户分为几种类型：Admin， Legal， Finance， Technical。不同种类的用户有不同的权限，这也是很好理解的。Legal这个用户用的很少，开发者不必太深究，一般而言，个人开发者不用配置其他的角色，自己作为Admin用户就可以完成全部的操作了：）。

## 请求用于推销的代码（Request Promotional Codes）

当应用程序成功的提交后，开发者可以向Apple请求一个用于推销的代码。开发者可以为每个发布的应用程序申请最多50个代码，这个代码可以用于免费下载该应用程序，以做产品的先期推广之用。

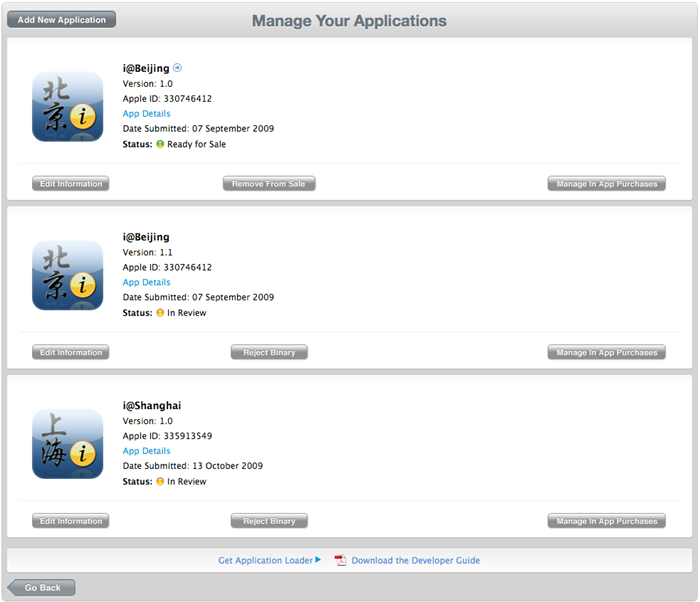
## 管理自己的In App Purchases（Manage Your In App Purchase）

一般开发者先期可以不考虑此功能，此篇文章对此不做细致的介绍，有兴趣的朋友，请自行参考《iTC\_Dev\_Guide\_5.0》.

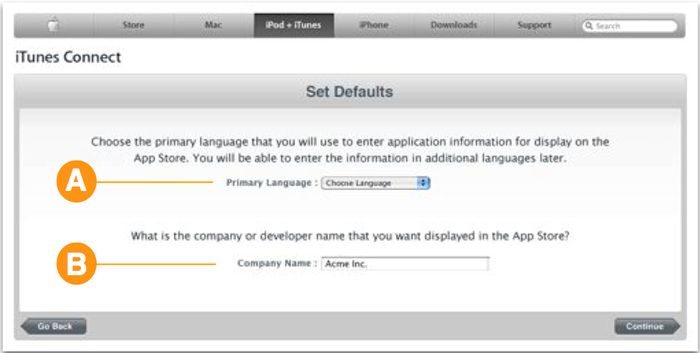
## 管理自己的应用程序（Manage Your Applications）

iTunes Connect的核心功能，应用程序的管理，其中涉及到新应用程序的发布、版本的升级等。

在iTunes Connect中，应用程序的管理位于Manage Your Applications这个连接下面。用户点击进入后，会有如下图所示的内容：（本人现在已经提交了两个用户程序i@Beijing和i@Shanghai, 其中第二个i@Beijing是一个版本升级申请。）



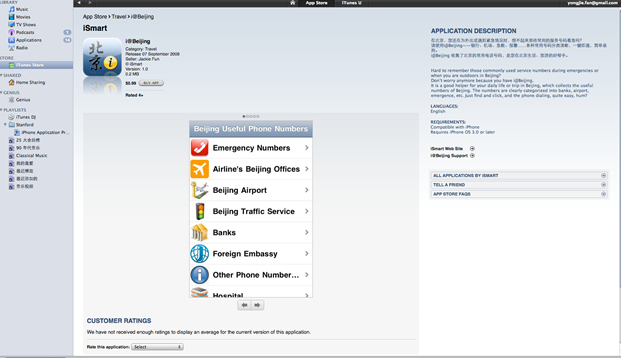
开发者想要提交新的应用程序，点击左上方的Add New Application按钮。如果是第一次提交应用程序，那么就会弹出缺省配置的设置页面，如下图所示：

**重要提醒：此页的配置将会应用到此iTunes Connect帐号下的所有应用程序，其中包括两个主要配置项：主语言和公司名称。切记：一旦这两项配置提交后，以后就不能修改了！**

图中的A是主语言配置项：即用户稍后输入程序详细数据所会使用的主要语言。

图中的B是公司的名称：对于个人开发者，建议尽量起一个有趣的，容易记的名称吧，不要懒省事的就用自己的帐号名，除非你的帐号名也很有特色：）

因为这个公司名称将会在App Store中，显示在一个非常醒目的位置，如下图所示：（图中为iTunes搜到本人的上架程序i@Beijing的信息页，左上角的黑体显示：iSmart即为本人的公司名称，是个很醒目的位置）



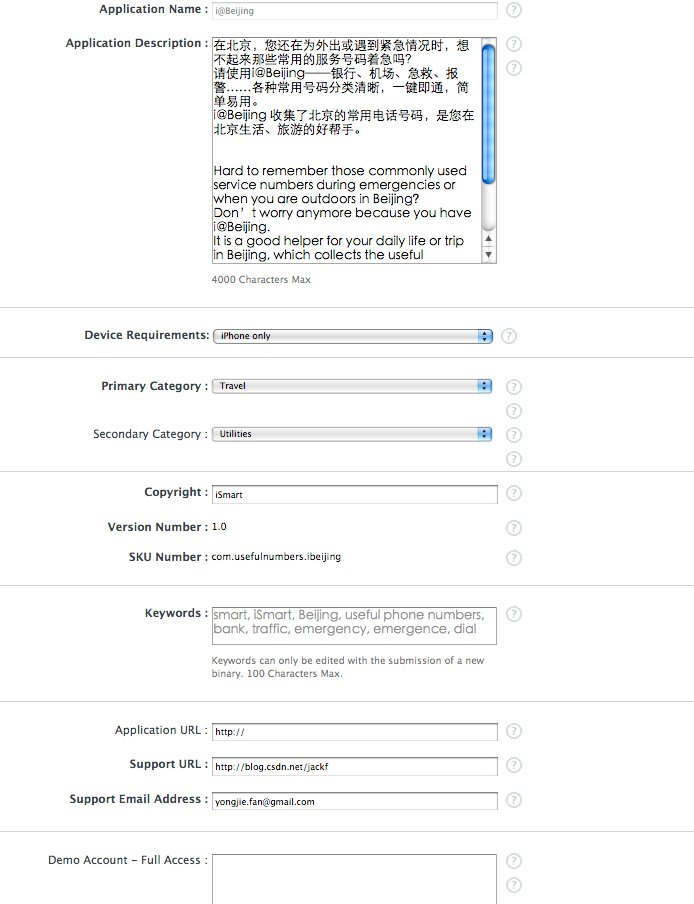
<!--[if !vml]--><!--[endif]-->

此页的两项配置完成后，选择 Continue继续下面的配置，即出口产品承诺的页相关的配置。这是由于美国法律对加密算法等的要求，如果你的应用程序没有涉及到加密啥的，可以选择No，就可以跳过了。（一般开发者而言，做点儿小工具、小游戏啥的，应该不会设计到加密，因此可以直接skip跳过，继续下面的内容）。

如果的确有涉及到加密，那么请选择Yes，而后回答两到三个更加关于用的的加密算法具体的问题，而后根据需要还要提供相关的文档，当Apple认为都OK了，那么才能继续下面的配置，这就比较麻烦了，所以如果是个人开发者，没有特殊的要求的话，就别搞啥子加密了，那样会自找麻烦哦。

### 概要数据

我们继续，下面就是提交程序时的核心数据了，即概要数据配置页面，如下图所示：



其中的必填项包括：程序名（Application Name），程序描述（Application Description）， 对设备的要求（Device Requirements），主要分类（Primary），版权（Copyright），版本信息（Version Number），唯一标示（SKU Number），关键字（Keywords），技术支持网址（Support URL），技术支持Email（Support Email Address）。

这是此页中必须填写的内容，有如下需要注意的地方：

1 程序名： 这一项中所填的内容，必须和你的程序安装在iPhone或者iPod中后显示的程序名是一致的，否则是无法通过Apple的审查。

2 程序描述：虽然我们之前选择了主语言，一般为English，但是此处，我们同样可以用两种或多种语言输入对自己程序的描述。对于此项具体的内容，那就需要开发者自己去想了，开动大脑，尽力忽悠吧，呵呵。这是消费者了解这个应用的很重要的描述，同时对其做出购买决定的影响也是很大的，所以尽量言简意赅、引人入胜吧。

注意：此处不要给出一些关键字的列表！为了提高搜索的可见性，开发者输入尽量多的关键字是无可厚非的，但这里绝不是输入关键字的理想的场所，毕竟这个简单描述是用来给消费者看的，本页的下面专门提供了供用户输入关键字的地方！

3 对设备的要求：这就要根据你的应用程序的类型来做出选择了，比如如果你的应用程序跟电话功能结合的很紧密，那么就请不要选择iPod，否则用户下载了，却不能用，那就不好了。还有，请切记要给出明确的说明，此应用所支持的iPhone的版本信息，目前iPhone已经到3.x了。Apple要求，新提交的应用必须支持3.0，否则审查是通不过的了。

4 主要分类： 这个很简单，根据所开发的程序类型选择合适的分类，比如：电子书（Book）、小工具（Utilities）、音乐（Music），旅游（Travel），游戏（Games）等等。

此页还提供了子分类的选择，这可以更加准确的描述程序类型，比如游戏的子类包括：纸牌（Card），运动（Sports）等等。

5 版权信息： 这就是刚才起到那个公司名称的官方名称了：）

6 版本信息： 这无需多说了，一般都从1.0开始。

7 唯一标示： 这个字符串可以是任意格式的，但是建议试用Apple ID中的定义，试用URL格式的标示串，如：com.xxx.getst。

8 关键字：开发者可以输入尽量多、准确的关键字以便提高程序被搜索的概率！但最多不能超过100个字符。所以要选择合适的，站在消费者的角度，注意消费者很可能不是技术人员！所以这里的关键字一定不要太技术了！

9 技术支持URL和Email： 这两项内容是必须提供的，Email相信都回有的，但是URL，如果没有的话，最简单的办法就是到某个有名的博客站上申请一个帐号，然后把博客的地址写进去就好了！比如：<http://blog.csdn.net/jackf>

好，到了这里，关于程序的概要数据就算填写完成了！ 下面开始要收集一些其他的数据。

### 对程序的评级

Apple提供了好几个评级的标准，开发者需要根据所开发程序的实际情况进行相应的选择，比如是否包含成人内容等，而后该页会根据用户的选择给出相应的程序评级，这个大家根据实际情况填写就好了，没有什么好坏之分：），开发者自己体会吧，我就不说那么直白了，呵呵。

### 上传数据

完成了上面的数据收集后，Apple就要求用户上传相应的程序文件以及截屏图了。关于这部分内容的准备工作，我已经在前面的文章中介绍过了，此处用户只需要根据提示，上传相应的文件即可，包括：

程序文件：是个zip包，该包中只包含编译成功的app文件。

512的图标：这是个jpg格式的大图标，用于在Apple Store中的显示。

程序截图：主截图以及最多4个其他截图。

再次强调一下，截图是重中之重！尽量给用户展示出最好的界面！

### 定价

想要收钱吗？想的话，就给自己的程序定个价，不想的话也很好啊，那就选择Free，供人免费下载，呵呵。估计大多数人还是想着挣些小钱吧：），那就选一个合适的价位吧，注意：Apple提供的这个App Store可不是免费的啊，所以它要从你的收入中提成的！Apple采用7:3分成，即开发者得到总收入的7成，Apple得到剩余的3成。不爽吗？那没办法啊，Apple就是这样强势！

### 地域选择

这个地方，可以选择另外一个输入用户应用程序数据的语言，如果你的应用程序支持多语言，那么强烈建议此处增加相应语言的描述，否则，English这个主语言就够了：）

### 检查页面

当所有的上述页面都完成后，我们可以在这个最后的页面再次检查一下我们所填入的内容是否完整、正确，毕竟这可是商品的包装啊！只有好的包装才能吸引消费者的眼前，从而才会产生潜在的收入哦：）。

点击提交后，我们推出编辑界面，回到了此篇第一副图所示的界面，程序处于In Review的状态！

一般而言，Apple会在一到二内通过Email通知你审查的结果，如果通过了，那么此处就变为了Ready for Sale状态，那么你就可以在iTunes中搜索到自己的应用程序了。如果没有通过，此处会是Rejected状态，Apple会通过邮件给出原因，开发者需要根据邮件中的描述做出修改后，再次提交申请。