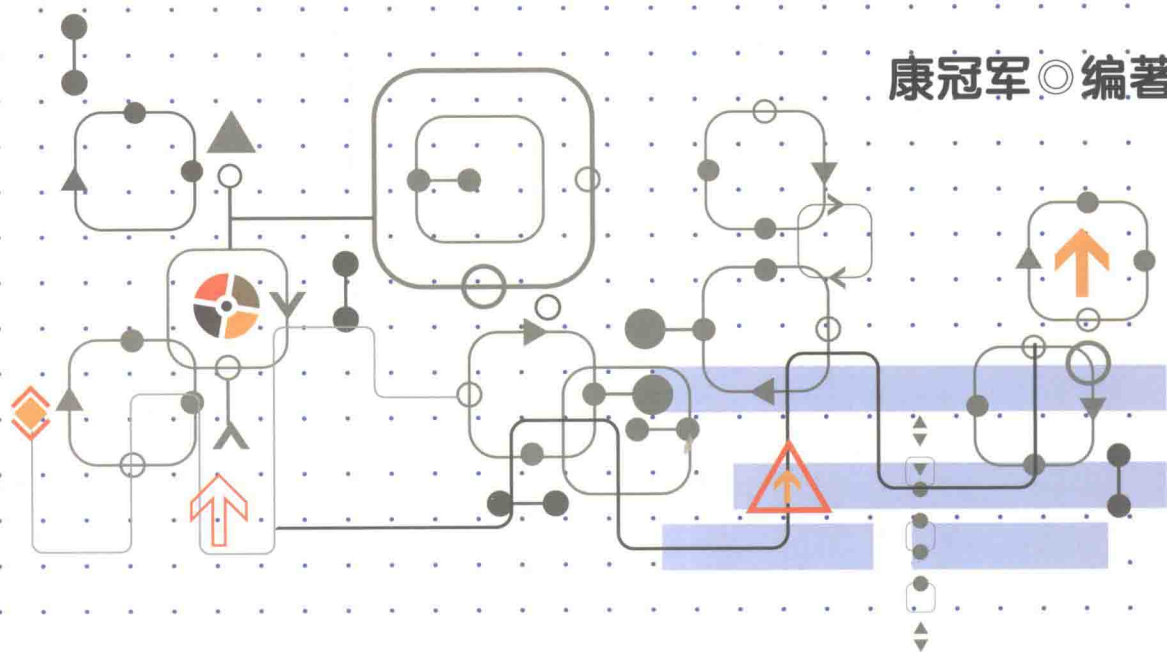


移动开发系列

基于Swift的 iOS App开发全景扫描: 诗意SagePoet 从诞生到上线

康冠军◎编著



以实际上线 iOS App SagePoet 为案例

记录**从零开始直到上线销售的全部过程**及实现细节
项目相关性强，自成体系

 中国工信出版集团

 电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

移动开发系列

基于 Swift 的 iOS App 开发全景扫描： 诗意 SagePoet 从诞生到上线

康冠军 编著



電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

基于 Swift 的 iOS App 开发全景扫描: 诗意 SagePoet 从诞生到上线/康冠军编著. —北京: 电子工业出版社, 2017. 7

(移动开发系列)

ISBN 978-7-121-31796-5

I. ①基… II. ①康… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2017) 第 129529 号

策划编辑: 张 迪

责任编辑: 张 迪 (zhangdi@phei.com.cn)

印 刷: 北京天宇星印刷厂

装 订: 北京天宇星印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787 × 1 092 1/16 印张: 12.25 字数: 314 千字

版 次: 2017 年 7 月第 1 版

印 次: 2017 年 7 月第 1 次印刷

定 价: 39.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010)88254888, 88258888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: (010)88254464, zhangdi@phei.com.cn。

About

关于本书

本书是基于一款苹果在线应用商店实际 iOS 应用——SagePoet App 实例的开发出品过程而写就的书，记录了从零开始直到上线销售的全部过程及实现细节。麻雀虽小，五脏俱全，这本书涉及一款 iOS 应用诞生过程的各个方面，能够自成体系。通过此书的学习，读者基本能够达到实现独立开发 iOS 应用并发布到苹果应用商店进行销售的能力。本书以实际的 SagePoet 应用开发过程为脉络，逐步展开 iOS 平台 App 开发相关的必要知识和工具等内容，帮助读者掌握必需的开发知识和开发流程，读者也可据此进行拓展自己感兴趣的内容。

SagePoet 简介

SagePoet 是作者独立创作的一款 iOS App——苹果在线应用商店的上线产品。SagePoet 以中国古代唐诗为素材，对唐诗进行了整理归纳，还可以检索学习，并设计了猜字连句的小游戏，帮助用户加深对诗的字句搭配的理解，通过游戏互动提高用户对诗的兴趣和掌握能力。SagePoet 应用部分界面如图 A-1 和图 A-2 所示。



图 A-1 SagePoet 游戏界面



图 A-2 SagePoet 诗集界面

使用说明

本书适合零基础及对 iOS 开发有一定了解的 iOS 开发爱好者、高校相关专业学生、诗词爱好者等。本书采用的开发环境基于最新 IDE 版本 Xcode 7.3.1 及 iOS9.3，使用的语言为 Swift2.0。Xcode7.3 是苹果于 2016 年 3 月最新发布的发展工具版本，iOS9.3 是苹果 2016 年 3 月最新发布的移动操作系统。Swift2.0 是苹果在 2016 年更新发布的开发语言。简单说来，本书采用 Swift2.0 语言在 Xcode7.3 开发工具里开发出最高适用 iOS9.3 版本操作系统的 iOS app 应用 SagePoet。iOS、Xcode、Swift 版本变化如表 A-1 所示。

表 A-1 版本历史表

iOS	Xcode	Swift
9.3	7.3	2.0
9.0	7.0	1.0
8.0	6.0	
7.0	5.0	

目 录

第0章 构思	1
0.1 用户是谁	2
0.2 用户需求	2
0.3 如何实现用户价值	3
第1章 布局	4
1.1 用例分析	4
1.2 布局设计	5
1.3 界面详细设计	12
第2章 素材	16
2.1 唐代古诗	16
2.2 古诗拼音	16
第3章 工具	17
3.1 Apple ID + Apple Developer Program	17
3.2 Xcode	24
3.3 Swift 语言	30
3.3.1 Swift 简介	30
3.3.2 Swift 特点介绍	30
3.3.3 Swift 语言详解	33
第4章 数据模型	73
4.1 数据编码	74
4.2 数据对象与数据格式化	77
4.3 数据文件读/写目录	86
4.4 数据模型 dataModel	88
4.5 SQLite 数据库文件	94
4.6 CoreData	95
4.7 数据导入数据库	98
第5章 控制器	105
5.1 PoemsViewController	105

5.2	DetailForPoemController	115
5.3	GameViewController	131
第6章	调试	145
6.1	调试基础	145
6.2	常见 crash 原因归类	147
6.3	instrument 分析	147
第7章	修饰	152
7.1	修饰基本要素	152
7.2	背景	157
7.3	动画	158
7.4	音效	160
7.5	appIcon	161
7.6	LaunchImage	163
第8章	多设备支持	165
8.1	自动布局	165
8.2	屏幕方向	166
第9章	国际化	168
9.1	界面国际化	168
9.2	appname 国际化	171
第10章	上线	174
10.1	appID	174
10.2	证书	177
10.3	Provisioning Profiles	181
10.4	iTunes 连接设置项目	183
10.5	打包上传	186
10.6	ready for sale	187

第 0 章

构 思

做一个苹果 iOS 的 App，为什么需要构思？

首先，设计制作一个 iOS App 的工作其实是一个项目过程。所谓项目过程，可以把它理解为为了实现一个特定的目标而进行的一次性的、独特的、创造性的工作过程。项目过程与日常工作过程是有区别的，日常工作过程是指非一次性的、可以不断重复的工作过程。例如，工厂的生产过程不属于项目过程，而是日常工作过程，因为这种过程属于不断重复的过程，并非一次性的、也没有创造性的特点。尤其要着重理解项目过程的创造性特点，项目过程一般都需要进行一定创新工作。项目过程与日常工作过程的区别如表 0-1 所示。

表 0-1 项目过程对比日常工作过程

	项 目 过 程	日常工作过程
可重复性	×	✓
创造性	✓	×
独特性	✓	×

也因此，作为项目过程的 iOS App 的开发制作工作不可能是那种说干就干就一定能干好的。如果不经构思，即使偶尔干好了，也很有可能失去了最初做这个 iOS App 的初衷及目标，做出来的 iOS App 很大可能上是随意的、不够创新的，甚至完全不是用户所想要的，这就失去了该 iOS App 项目的目标意义。从这点来看，构思对于一个 iOS app 来说是极其重要的，因为如果没有构思，又谈何创新呢？

其次，苹果公司对任何一款上线销售的 iOS App 都是要进行严格审核的，如果不经良好的构思，是难以满足苹果公司对 iOS App 的审核标准要求，可能被拒绝上线。透过苹果公司的经典产品系列 iPhone，就能大概了解苹果公司的价值追求所向，就是要给用户极致艺术般的产品体验，同时让高科技不断融入创新的产品，提供给用户方便的使用体验，使用户不用去面对并理解那些复杂的高科技，只管尽情享受苹果公司产品带来的创新、艺术化的高科技产品就可以了。苹果公司对 iOS App 的审核要求也

一样遵从这种创新价值方向，也就是 iOS App 的内容与风格都需要体现苹果公司所推崇的这种价值观。

最后，构思的工作是构建一个 iOS App 灵魂的过程，体现了 iOS App 存在的价值。一个好的构思将会在很大程度上决定了一个 iOS App 的优秀程度。就如同一篇文章的立意，只有立意高远，这篇文章才能达到一定的高度、深度及新意。

构思，实际上也是创意，是指为满足一项独特的需求，实现一项特别的目标而进行的思维过程。通过项目构思需要解决 iOS App 的目标、功能、范围等三大问题，以及对项目涉及的各主要因素和大体轮廓进行规划构想，并进行初步的规定。简单说来就是要考虑好 3W 问题：

- W1：Who？用户是谁？即目标市场定位问题。
 - W2：What？用户想要什么？也就是用户需求问题，体现了应用的用户价值。
 - W3：How？如何提供用户价值？也就是怎样通过产品或服务传递用户价值。
- 下面通过 SagePoet 实例应用来分析构思。

0.1 用户是谁

用户是谁？作为一款采用中国古诗词为素材打造的 app 应用 SagePoet，针对的用户主要是诗词爱好者、中小學生及文字游戏爱好者。

0.2 用户需求

用户有什么需求？SagePoet 的用户需要进行古诗的收集、归纳，以及学习、温习。如果有一种不闷的学习方式，也即休闲游戏的方式寓学于乐会更好。用户需求如表 0-2 所示。

表 0-2 用户需求表

需求/用户	诗词爱好者	中小學生	文字游戏爱好者
需求 1：诗词修学	诗集整理归纳研修，查询	整理诗词，方便查询学习	
需求 2：文字游戏	文化游戏，打发无聊时刻	通过游戏，寓学于乐，提升语言理解深度	开动脑筋，猜字连连诗句

0.3 如何实现用户价值

注重 user experience 用户体验的原则，围绕用户满意度为宗旨。通过产品提供用户满意的功能，解决用户的问题，并提供一些超出预期的性能，帮助提升用户体验和用户满意度。

SagePoet 应用因此计划通过帮助用户收集整理归纳 poems 形成诗集，并提供检索查询的功能来解决用户随时可以通过手机应用进行古诗的查询、温习等需求问题。

同时 SagePoet 计划设计一种猜字连句的 game 游戏，使得用户能够寓学于乐，加强用户对古诗更深入的理解，因此给用户提供一种更好的体验和更高的价值。

布局

布局主要考虑两方面的问题：一个是应用的功能组成问题，也就是解决布局的内容问题；其二就是界面设计问题，也就是解决布局的形式问题。

布局的内容方面，即应用的功能组成问题离不开用户的需求，应用提供的每一项功能都是基于用户需求的。通过用例分析，对用户的需求进行更深入具体的定位，使布局可以完全容纳体现所有用例，而不至于遗漏某些用户需求。

布局的形式方面，即应用的界面设计问题，涉及美观、人机工程学等方面因素。这些因素最终直接关系到用户体验的层面。因为即使应用帮助用户实现了某项用例功能，但由于界面设计不好，使其操作体验不佳，该应用也是失败的应用。

下面通过 SagePoet 实例应用来分析应用的布局设计问题。

1.1 用例分析

经过对用户的调查分析，整理出 SagePoet 的 4 个基本用例描述：

- 对古诗进行集中整理成诗集 Poems。
- 对古诗按作者进行归纳、检索。
- 选择一首诗进行查看、温习、背诵。
- 随机对一首诗玩猜字连句的文字游戏。

每一个用例都代表一项具体的用户需求，注意用例描述中的动词与名词，动词代表用户的操作，用户每一次操作的目的是希望能得到一种预期的结果或体验。名词代表用户操作的对象，以及操作后得到结果或期望获得的体验。SagePoet 应用用例如表 1-1 所示。

表 1-1 用例分析表

用 例	操作（动词）	操作对象（名词）	结果/体验（名词）
用例 1	集中整理	所有古诗	古诗集 Poems
用例 2	归纳、检索	所有古诗	按作者分类成区 Sections
用例 3	选择、查看、温习	某一首古诗	某一首古诗的细节 Details
用例 4	玩猜字连句游戏	某一首古诗的汉字	寓学于乐的体验

1.2 布局设计

尽管布局设计表面上看来属于平面设计的范畴，并不涉及高精尖的高科技，但是布局设计对一款 iOS App 的成败来说是至关重要的，因为布局设计影响一款 iOS App 用户使用的各个方面，最终在很大程度上直接影响用户的各种使用体验。如苹果公司这种极其重视用户体验的公司，对产品外观是极其重视的，苹果公司的任何一个产品系列，都有一种统一的风格，轻盈、简洁、时尚而不乏艺术性。苹果公司绝对不会让用户直接面对那些复杂的高精尖技术，而是通过巧妙的、精细的布局设计，把轻盈、简洁、便于使用同时不乏艺术性的外在产品呈现给客户，而那些高精尖科技就像隐形般不露痕迹地融进那经过精妙布局设计的产品里，因为用户需要的不是直接的高科技，而是产品使用的简单方便及舒适性。为此苹果公司，对于 iOS App 的布局设计还专门出了一个规范性的标准文件——《iOS Human Interface Guideline》，所有违背该标准的 iOS app 都将被拒绝上线。

布局设计的几大原则问题如下所示。

界面设计原则之一：就是最好为每一个用例设计一个独立的操作界面。界面设计一定是要基于用例分析来进行的，应该把某用例所涉及的所有操作，以及操作对象、操作对象结果和体验等都包含进该用例对应的界面设计中去。

界面设计原则之二：符合人机工程学。人机工程学包含多个方面内容，对于应用界面来说，主要宗旨就是要求界面设计人性化，使得各种用户操作和内容呈现皆符合人的各种生理、心理特点，顺应用户的各种思想和行为，最起码要做到不能给用户带来操作上的困难和理解上的困扰。苹果关于界面设计问题提供了一个规范性的指引文件“iOS Human Interface Guideline”，所有违背该文件的应用都有可能被苹果拒绝上线运营销售。

布局设计基础要求如下所示。

- 简洁清晰：尽管漂亮的界面和流畅的动画是用户体验的亮点，但是内容才是界面的核心。界面设计必须以突出内容为主，避免大量装饰性元素把内容淹没了。如何做到布局贴切内容？有以下几个要点。

(1) 利用整个设备屏幕布置内容；

(2) 去除不必要的装饰，如阴影效果、渐变效果、Bezel 效果等装饰。因为过多的这些装饰只会加重界面元素的效果甚至超过内容的重要性，这就不得体了。界面元素的作用永远是对内容的支持，不能与内容竞争对用户的关注。

(3) 使用半透明的界面来暗示当前界面下的内容，如图 1-1 所示

(4) 使用大量空间留白以突出内容及使用色彩来简化界面。使用一种关键色彩来突出重点内容信息，并提示可交互性，如图 1-2 所示。

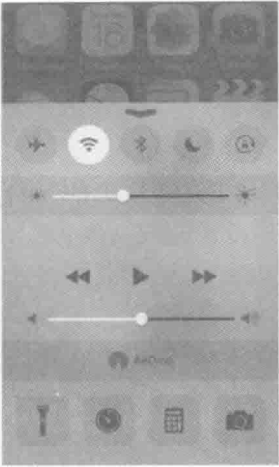


图 1-1 界面半透明设计



图 1-2 空间留白及关键色彩使用

(5) 使用无边界按钮。无边界按钮可以和内容融为一体，不会与内容竞争用户的关注度，无边界按钮一般使用色彩、标题及上下文关系来提示用户可交互性。

- 原生风格：尽量使用苹果提供的原生界面设计元素。苹果的原生界面元素自身也提供了一些可供自定义的属性。如何体现原生风格？方法就是使用苹果提供的 UIKit 框架。UIKit 框架提供了许多苹果原生界面元素。界面元素总体分为四大类，即 Bars，Content views，Controls，Temporary views。Bars 一般用于上下文相关信息的情形，如状态栏、导航栏、工具栏、切换栏等。注意所有的界面元素都继承了 UIView，都是 View 的一种类型 Type，如图 1-3 和图 1-4 所示。

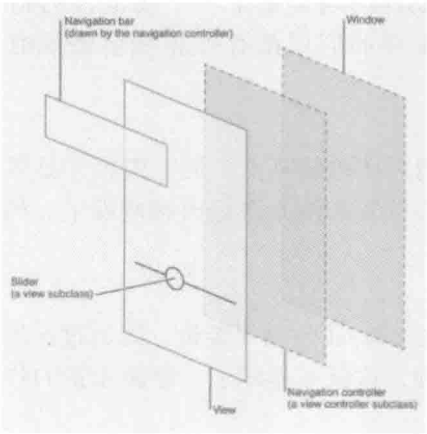


图 1-3 UIKit 展示

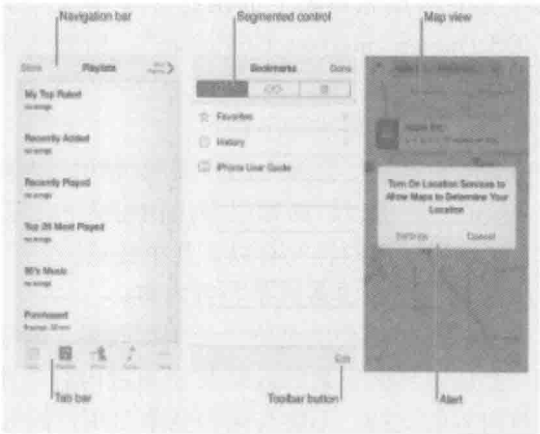


图 1-4 UIKit 展示

- 44x44Px: 方便用户触摸的最小屏幕区域尺寸。比这个尺寸小的操作控件, 就会让用户难以触摸到预期的区域, 如图 1-5 所示。

布局设计流程方法如下所示。

第一步: 布局设计之前请回顾一下 App 的核心职能, 布局一定要围绕 App 的核心职能来设计。请严格确保二者的相关性。

第二步: 使用 iOS 原生风格界面, 在此基础上再适当增加自定义风格的细节装饰, 小心不要过分装饰。确保苹果用户的用户体验。

第三步: 使界面布局设计尽可能适合不同的设备及屏幕显示模式, 如此才能尽可能让用户喜欢使用你的 App。设备屏幕显示模式有以下两种, 竖向 (portrait)、横向 (landscape), 如图 1-6 和图 1-7 所示。

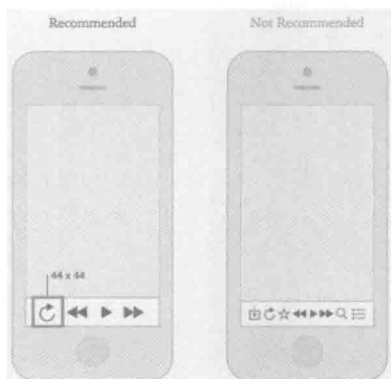


图 1-5 合理的像素界面元素展示

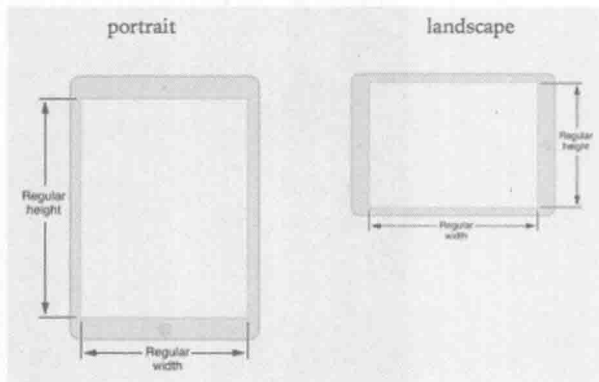


图 1-6 iPad 界面显示模式

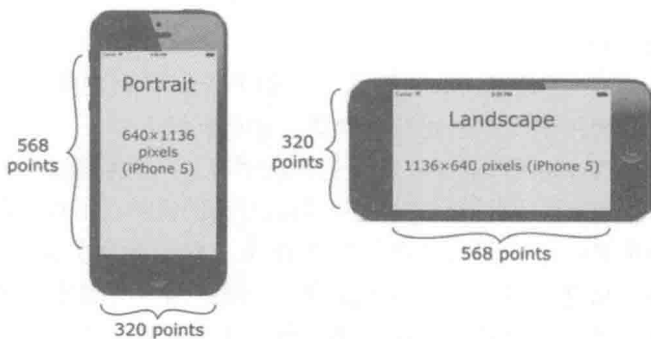


图 1-7 iPhone5 界面显示模式

下面以 SagePoet 来分析界面设计。

根据 SagePoet 的 4 个用例, 初步设想 4 个用户界面, 分别是游戏界面 Game、诗集界面 Poems, 针对某首诗的查看学习界面 Detail, 以及归纳检索界面 Search, 如图 1-8 所示。

总结出 SagePoet 的 4 个界面之后, 进一步对这 4 个界面之间的关系进行分析。界面之间的相互关系可以分为并行关系、导航关系、附属关系等主要 3 种。下面以 SagePoet 作为

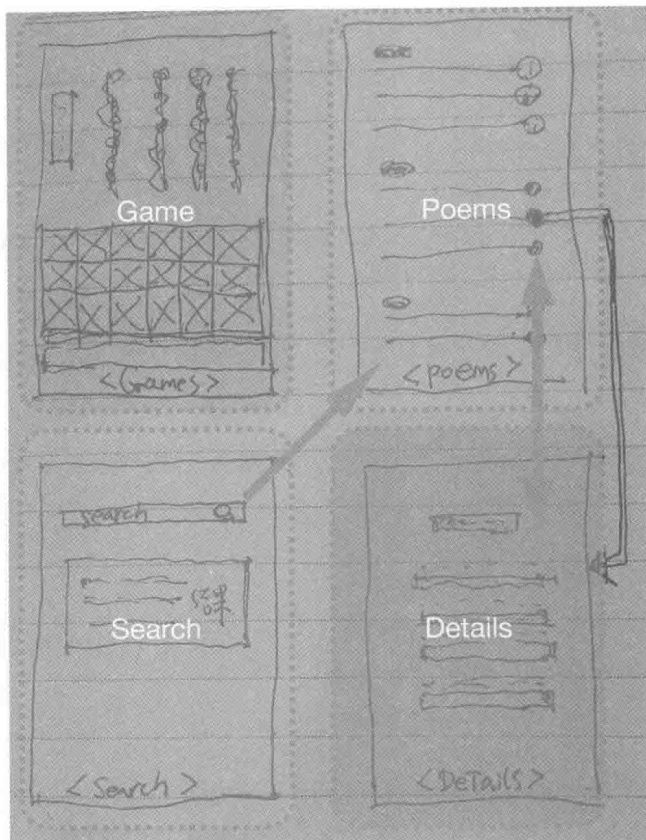


图 1-8 界面布局设计

案例对界面关系进行分析。

- 并行关系：SagePoet 的游戏 Game 界面与诗集 Poems 界面之间的关系属于并行关系。这两个界面代表的用例操作之间没有交集，是相互独立的。
- 导航关系：诗集 Poems 界面与某一首诗的查看学习 Detail 界面属于导航关系。因为用户是在诗集界面里选择了一首诗后才会进入到该首诗的查看学习界面的，属于导航关系。
- 附属关系：诗集 Poems 界面与归纳检索界面属于附属界面关系，归纳检索操作对象完全是基于诗集来进行的，因此归纳检索界面是附属于诗集界面的。

针对不同的界面关系，需要进行不同的界面布局处理，如下所示：

- 并行关系处理：采用苹果提供的 Tab Bar Controller 来构建处理并行关系的界面布局。
- 导航关系处理：采用苹果提供的 Segue 形式来构建处理导航关系的界面布局。
- 附属关系处理：采用整合界面的形式来处理附属关系的界面布局。

针对 SagePoet 的 4 个设想界面，以及其相互之间的关系，下面做相应的处理如下：

- 用 Tab Bar Controller 组织并行关系的 Game 界面与诗集 Poems 界面。
- 从 Poems 界面设置一个 Segue 导航到相应的 Detail 界面。

- 合并处理 Poems 界面与 Search 界面，即把 Search 操作嵌入到 Poems 界面内实现。

处理后 SagePoet 的 4 个界面变为 3 个界面，分别是 Game、Poems、Detail 等 3 个界面。因此在后续界面设计工作中只需要构建实现这 3 个界面即可。下面开始实际操作。

打开 Xcode，新建一个项目（如有疑问请参照第 3 章工具关于使用 Xcode 新建项目流程），取名为 SagePoet，完成新建项目流程后，在 Xcode 左边栏的项目资源导航栏内熟悉一下项目的文件组成，然后选中 Main.storyboard 文件（该文件是应用的界面设计文件）后，Xcode 主窗口即会显示了该文件内容，默认包含一个 ViewController，如图 1-9 所示。

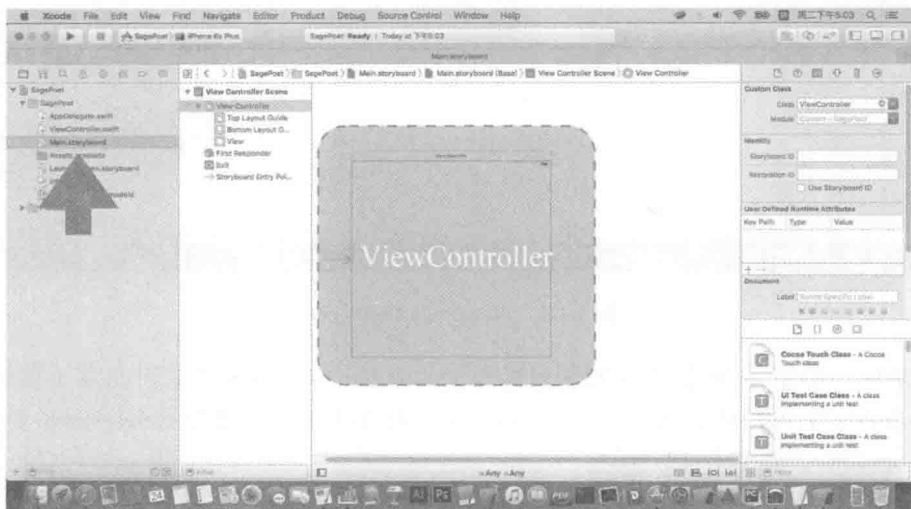


图 1-9 Main.storyboard 文件编辑界面

把默认的 ViewController 删除，从右边栏下方对象资源库一栏内找到 Tab Bar Controller，并将其拖进主界面窗口，如图 1-10 所示。

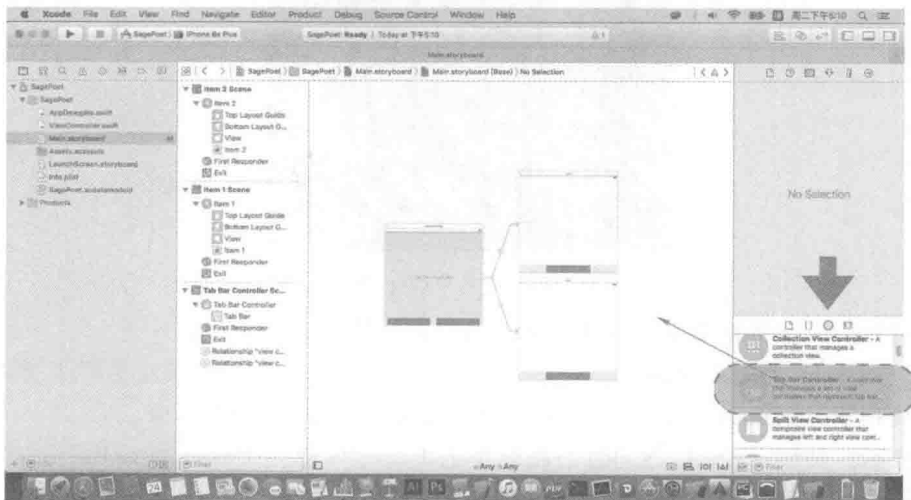


图 1-10 Tab Bar Controller

然后选中 item2ViewController（窗口右下方），删除它。再从右边栏下方对象资源库一栏内找到 Navigation Controller，并把其拖进主界面窗口，如图 1-11 所示。

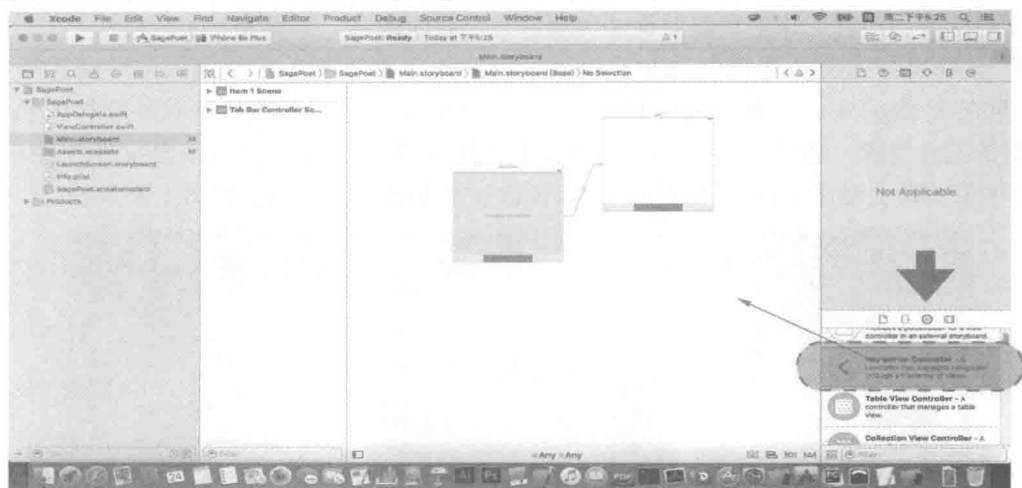


图 1-11 Navigation Controller

拖进来后，按住【Ctrl】键同时用鼠标单击 Tab Bar Controller 不松开从其上面拖到 Navigation Controller 上再松开，然后在弹出的 segue 选择菜单内单击选择 Relationship Segue 下方的 view controllers，如图 1-12 所示。



图 1-12 Relationship Segue

选好设置后，主编辑界面窗口各界面之间的关系如图 1-13 所示。

Game 与 Poems 界面都存在了，只差一个某首诗的 Detail 界面，从右边栏下方对象资源库一栏内找到 View Controller，并把其拖进主界面窗口，如图 1-14 所示。

拖进来后，按住【Ctrl】键同时用鼠标单击 Poems 界面内 Prototype Cells 下方的那个白色条框不松开，从其上面拖到新进的 View Controller 上再松开，如图 1-15 所示。