

# iOS 高级面试简历指导

## 前言

这篇文章我来总结一下 iOS 程序员的简历应该怎么写，有哪些注意事项，如何拿下大厂 offer 的必备技术点；

花点时间，耐心看完，一定会对你有所帮助！

## 目录

一：iOS 开发者定位分析

二：iOS 开发基本要求和加分项

三：个人简历信息模板分析

---

## 一：iOS 开发者定位分析

### 1、简历注意点（了解自己）

- 1、不要夸张自身的简历；面试一问便知！
- 2、简历制作尽量不要用“招聘网站”的自动生成简历模板
- 3、根据自身的特点，自身优势，制作属于自己的简历（1）优势：学历，技术背景等，哪一块突出，哪一块就是优势！

### 2、iOS 开发者定位分析（仅个人看法）

#### 1：开发者定位-初级 iOS 开发者

- 1.能完成基本的 UI 界面搭建
- 2.了解网络,能利用网络框架上传
- 3.熟悉 iOS 常用内置的 API
- 4.能使用常用的第三方框架
- 5.能根据业务逻辑,完成数据与 ui 联系

#### 2：开发者定位-中级 iOS 开发者

- 1.能够快速完成界面搭建
- 2.能够根据项目业务逻辑以及项目需求封装出常用的工具类
- 3.熟练常用 iOS 内置 API
- 4.能根据业务需要快速选择合适的第三方,并快速完成第三方学习.

- 5.了解常用第三方框架底层并能做二次封装
- 6.具备基本的架构能力.能够根据项目以及业务方向设计健壮的项目架构
- 7.较好的编程习惯
- 8.具备一定的产品意识 .....

### 3.开发者定位—高级 iOS 开发者

- 1.第三方库安全性评估
- 2.架构风险评估,项目风险评估
- 3.具备创新并开发出新框架的能力
- 4.有自身技术专项领域.在这个领域有积累有造诣,解决公司核心业务.比如项目安全.图形图像处理,音视频.能够帮助公司攻克技术难度,做技术研发. .... (其他能力不一一概括了)

### 3、开发者定位小总结

总之我们要足够了解自己,才能在简历上有一个清晰的定位,以及在求职过程中,有一个明确的位置选择; 例如:有一家公司招的高级开发者,但个人能力目前达不到高级开发,这样去面试结果可想而知,而且还会浪费一次机会! 当面试中认知到自己的不足,那么就要朝着更高的层次去学习提升!

## 二: iOS 开发基本要求和加分项

### 1、基本需求:

- 1.多线程
- 2.网络
- 3.主流第三方框架实现原理
- 4.OC 本质
- 5.数据结构/算法
- 6.数据库
- 7.设计模式
- 8.项目经验
- 9.runtime / runloop
- 10.c/c++
- 11.界面开发能力

### 2、技术加分项:

- 1.网络直播课堂研发经验[IM,直播,音视频研发经验]
- 2.初创公司从业经历
- 3.Swift 项目经验
- 4.音视频编解码,

- 5.视频直播,短视频开发经验
- 6.AVFoundation
- 7.OpenGL/ARKit/GPUImage 图像处理
- 8.直播产品经验
- 9.安卓端开发经验
- 10.5 年以上 iOS 开发.大型 APP 项目
- 11.有多种编程语言经验
- 12.GitHub.上有原创开源框架/项目
- 13.负责过成熟项目架构设计
- 14.函数式编程(block 灵活运用]
- 15.英语能力[CET-6]
- 16.逆向实战项目经验
- 17.项目管理经验
- 18.跨平台开发经验(Weex/Flutter/RN/Hybrid)
- 19.博客
- 20.出书

### 3、简历要求注意

- 1.切记生搬硬套.
- 2.简历必须符合自身技术能力.
- 3.最好的简历必须要有技术作为支撑.

## 三：个人简历信息模板分析

### 1、简历版块构成

- 1.个人信息
- 2.自我简介
- 3.技术列表
- 4.项目经历
- 5.教育背景
- 6.工作背景
- 7.兴趣爱好

---

#### ①.个人信息，模块需要包含如下信息：

- 姓名
- 性别

- 年龄(年龄在 30-35 岁左右，建议不用直接写年龄)
- 电话
- 常用邮箱
- 常住地
- 学历+专业(985/211 学历优势，靠前写；计算机专业靠前写；)
- 英语能力
- 工作年限
- 籍贯
- 个人博客

### 仅供参考：

**个人信息模板** [切记必须符合自身真实情况]

姓名: 苏大强	工作年限: 5年
性别: 男	英语能力: CET-6(可流畅阅读英文文档)
年龄: 28	学历: 中南大学
电话: 18373*****9	专业: 计算机科学与技术专业
邮箱: sudaqiang@163.com	个人技术博客: www.sudaqiang.com
籍贯: 湖南.长沙	常驻地: 北京

## ②.自我简介版块:

自我介绍：是通过一段简短的语言，快速让面试官知道自己的技术优点，技术特点；描述的是自己的能力，而不是自己的个性！

### 例子：个人介绍(仅供参考)

- 1.本人有五年多的Object -C编程经验,覆盖PC端, App端, Web端,并熟悉H5.
- 2.具有团队管理经验,跟后台沟通.接口的编写, 原型设计、U设计.研发、测试以及上线、包括跟运营沟通都有参与。
- 3.解决问题能力突出, 热衷算法。
- 4.精通MVC框架, MWM框架, 使用过RAC框架,用过React Native框架。常用的第三方有AFNetwork、SDWebImage、FMDB、Masonry、Cocoapods等很多,熟悉Hotfix,
- 5.擅长优化App,优化ipa包,喜欢抽象、封装、代码风格极简。喜欢注释,用代码块高效开发。
- 6.做过直播和IM

### 【重点】：个人信息模块需要包含如下信息:[切记必须符合自身真实情况]

- 个人突出经验(开发经验，知名项目经验,知名企业经历等)
- 个人技术优势(逆向,音视频,跨平台开发经验,Swift 完整上线项目,其他平台开发经验)
- 个人学习成果(gitHub 开源项目，开源框架，个人技术博客，公众号，社群以及博客粉丝等。

- 个人学习成果(个人技术书籍出版,国外技术文献翻译书籍出版,国外技术文献翻译文档)
- 个人管理经验(某项目架构设计以及项目管理等)
- 自我技术优势:可以从技能列表中筛选,例如组件化/响应式编程等
- 自我简介宗旨:突出自我优势!

### 模板: 自我简介(仅供参考)

- 本人有5年Objective-C开发经验,2年Swift开发经验.熟悉C/C++.独立完成2个项目,团队研发4个项目.并负责项目组项目上架.熟悉项目上架流程.
- 曾任职\*\*\*事业部项目iOS团队负责人.具有项目团队管理经验,与项目组后台/设计/产品/测试/运营保持良好的沟通.
- 本人精通MVC/MVVM框架,熟悉RAC/RXSwift框架,对响应式编程使用熟练.
- 工作之余研究iOS逆向开发,音视频处理以及对主流第三方框架AFNetwork/Alamofire网络框架, SDWebImage/YYImage图片加载框架, FMDB/WCDB数据库框架等底层原理.
- 个人技术博客: \*\*\*\*
- GitHub: \*\*\*\*

---

## ③.个人技术列表解析

### 网上案例: (仅供参考)

- 1.熟练Objective -C和Swift.有完整的iOS App开发经验。
- 2.熟悉iOS逆向开发。
- 3.熟悉React Native跨平台开发。
- 4.熟悉iOS组件化开发。
- 5.熟练使用CocoaPods开发。
- 6.有支付, 地图、音频开发、交易线、移动社交, 直播的开发经验。
- 7.熟练使用MVVM、单例、KVO、MVC、适配器等设计模式。
- 8.熟练使用xib自动化布局以及Masonry第三方布局。
- 9.熟悉iOS中Block、Runtime消息转发机制等。对iOS程序的编写也有比较深的技术储备。
- 10.熟练使用Cocoa Touch, UIKit等.熟悉各种Core Frameworks.
- 11.熟悉TCP、UDP、Http、XMPP协议,熟悉Socket等网络编程,以及GET/POST数据请求等操作。
- 12.熟悉NSThread、GCD、NSOperation等多线程编程技术。
- 13.熟悉使用Instruments进行内存、性能、渲染检测,对代码性能进行优化,以及使用UnitTest和UITest进行单元测试和自动化测试。
- 14.熟悉NSUserDefaults、Plist、Keychain、NSCoder等数据存储技术。
- 15.对于iOS的UI界面的动画效果的编写也很熟悉,能够开发一些页面比较复杂的动画。
- 16.对UI控件有优化经验,有大量适配不同屏幕,不同分辨率的手机的经验。

注意点: 个人技术优势放前面

### 【重点】: 技能列表模块注意细节:[切记必须符合自身真实情况]

- 所有技术点必须符合目前自身实际能力
- 建议面试前针对技术点准备 3-5 个值得探讨的技术点(注意点)

- 注意技术点书写正确，以及表述必须能完整表述其技术的名称+功能+亮点
- 简历尽量以 PDF 格式形式(开发者简历尽量简约即可)
- 技术切换以方向为技能例如：建议不要写精通逆向(而是细数对于逆向你具备实际能力)
- 技术列表中排序根据开发年限: iOS 核心开发技术->技术亮点->使用/了解的技术

## iOS 技术列表简介，各方向都有简单列出（仅供参考）

### 底层基础：

- 熟练掌握 Runtime , Runloop 底层原理以及底层源码分析
- 熟练掌握 KVC , KVO 使用及其底层原理
- 熟练掌握多线程原理以及 NThread, NSOperation 使用
- 熟练掌握 GCG 高级使用以及 GCD 源码分析
- 基于 GCD 自定义 NSOperation
- 熟练网络(OC/JS 交换/NSURLProtocol/Socket )
- 具备封装网络框架构建能力
- 了解项目单元测试(逻辑测试/异步测试/性能测试/UI 测试/OCMock 依赖注入)
- 项目持续集成/自动化打包
- 掌握项目性能优化(内存优化/界面优化/启动优化/网络优化)
- 了解常用架构模式(创建型模式/结构型模式/行为型模式).
- 了解直播项目架构设计
- 熟练掌握组件化开发

---

### 基本框架：

- 熟悉 AFNetworking 网络框架高级使用并了解底层原理
- 熟悉 Alamofire 网络框架高级使用并了解底层原理
- 熟悉响应式编程 RAC 框架高级使用并了解底层原理
- 熟悉响应式编程 RXSwift 框架高级使用并熟悉底层源码
- 熟悉微信开源框架 WCDB 使用并了解底层源码实现
- 熟悉 FMDB 数据库使用并了解底层实现原理
- 熟练掌握项目缓存策略,熟悉 YYCache 使用以及底层缓存策略.
- 熟悉图片加载框架 SDWebImage , YYWebImage , YYImage 使用以及底层图片加载/解码原理.
- 熟悉 YYAsyncLayer 异步绘制框架以及底层异步绘制原理
- 熟悉美团开源框架 Graver 框架的使用以及底层原理

---

### 图形视觉

- 熟悉 Mac OS 平台下 OpenGL 图形图像开发
  - 熟悉 OpenGL ES 渲染技巧(背面剔除/深度测试/多边形偏移/颜色混合)
  - 熟练掌握 OpenGL ES 着色语言 GLSL.可以编写 shader
  - 熟悉 iOS 下 OpenGL ES GLKit 框架
  - 熟悉 OpenGL ES 多重纹理混合/光照混合/颜色混合等。
  - 熟悉 OpenGL ES 实现视频渲染以及加滤镜
  - 熟悉 OpenGL ES 实现图片滤镜实现
  - 熟悉 OpenGL ES 加载 3D 模型
  - 熟悉 Metal 框架以及渲染流程.
  - 熟悉 Metal shading Language 能实现图形图像处理
  - 了解 SpriteKit 框架
  - 熟练使用 GPUImage 上层框架并了解 GPUImage 底层源码.
- 

## 音视频

- 熟悉 AVFoundation 音视频采集/视频播放/视频帧获取/切换前后摄像头/静态图片捕捉
  - 基于 AVFoundation 封装直播/小视频音视频数据采集工具类
  - 熟悉视频投屏技术 AirPlay
  - 熟悉 AVFoundation 人脸识别/二维码识别
  - 了解视频封装格式.视频/音频编码格式
  - 了解直播/小视频客户端架构
  - 了解 CDN 网络
  - 熟悉视频编码原理( IBP 帧/帧内预测压缩/帧间预测压缩)
  - 熟悉 H264 视频编码原理以及 H264 数据结构
  - 熟练掌握 VideoToolBox 视频硬编解码,并基于 videoToolBox 封装视频编码/解码工具类
  - 熟练掌握 AudioToolBox 音频编解码,并基于 AudioToolBox 封装音频编码/解码工具类
  - 熟悉音频裸流 PCM 播放/ AAC 解码播放.
  - 熟练基于 OpenGL ES 在 iOS 平台下渲染视频并添加滤镜.
- 

## 逆向安防

- 熟悉 iOS 加密手段(哈希算法/对称加密算法/非对称加密算法/OpenSSL)
- 熟悉 APP 签名(数字签名/代码签名/iOS 双重签名机制)
- 了解 Entitlements/CSR 文件/Provisioning Profile
- 熟悉重签名技术( codesign 重签名/xcode 重签名/shell 脚本自动重签名)
- 熟悉逆向项目进攻(、Framework 注入 / MachOview / LC\_LOAD\_DYLIB / DYLD / yololib / Dyllib 注入)

- 熟悉 Runtime 运行时机制，通过 SEL 方法号寻获 IMP 函数实现
  - 了解 MachO 文件,熟悉应用加载过程。
  - 熟悉逆向项目进攻(Hook 技术)以及防护反 Hook
  - 熟练逆向调试工具(LLDB / Cycript / MokeyDev / Logos )
  - 熟练掌握越狱开发(应用砸壳/Theos)
  - 熟悉越狱攻防手段(修改 RESTRICT 段防护 Tweak/二进制修改破坏防护/dy1d 源码分析/检测 RESTRICT 段防护二进制修改)
  - 熟练掌握汇编语言(还原高级代码/指针反汇编/Block 反汇编/OC 方法反汇编)
  - 逆向开发项目(逆向微信抢红包,实现自动拆红包,抢红包.)
  - 逆向开发项目(逆向钉钉打卡,实现异地 wi-fi 打卡)
  - 熟悉项目安全防护基础反调试(ptrace 反调试/ sysctl 反调试/破坏 ptrace , sysctl 防护/Ptrace&sysctl 提前执行)
  - 熟悉项目动态调试/静态调试。
  - 熟悉项目安全防护检测(MethodSwizzie 检测/签名检测/代码检测)
  - 熟悉项目安全防护(定位防护代码/破解 fishhook 防护手段)
  - 熟悉项目安全防护(字符串加密/隐藏核心函数调用/汇编防护)
  - 熟练使用 LLDB 调试工具.
- 

#### ④：项目经历

**项目经验：** > 误区:> \* 1.写的越详细越好 > \* 2.啰嗦,找不到重点 > \* 3.冗余信息过重,掩盖亮点 > \* 4.排版不清晰 > \* 5.认为独立开发才是好开发! > \* 6.项目经验与工作经历混为一谈

#### 项目案例简单分析(无参考)

项目周期:1 年/日期 项目名称: **项目人员:5 名** **项目描述:**(项目分类, 项目成绩) **项目技术点:**(架构, 框架, 底层技术出发) 1、\*\*\*\* 2、\*\*\*\*\* 3、\*\*\*\*\*

---

#### 返回：iOS 开发高级进阶必看学习库

#### 结语

本文仅供参考，每个人的经历都是不一样的，需要每个人自我补充，自我完成简历，才会呈现更好的！

- 最后附上一份精品整理的《大厂-iOS 面试题答案合集.pdf》



在这里插入图片描述

- 需要的可加 iOS 高级开发群: [563513413](#), 群文件直接获取

参考目录:

(5) <i>Objective-C</i>	15
(6) 内存管理	35
(7) <i>KVO-KVC</i>	45
(8) <i>Block</i>	47
(9) <i>Swift</i>	59
(10) <i>UI</i>	62
(11) 报错警告调试	85
(12) 第三方框架及其管理	90
(13) 绘图与动画	95
(14) 数据存储	101
(15) <i>RunLoop</i>	110
(16) 网络	117
(17) <i>WebView</i> 与 <i>JS</i> 交互	128
(18) 多线程	137
(19) 多媒体	147
(20) 设计模式	148
(21) 安全机制	158
(22) <i>Runtime</i>	159
(23) 性能优化	178
(24) 通知与推送	184

——补充

[https://blog.csdn.net/qq\\_18612413](https://blog.csdn.net/qq_18612413)

## (5) Objective-C

面试笔试都是必考语法知识的。请认真复习和深入研究OC。

### 方法和选择器有何不同? (Difference between method and selector?)

- selector是一个方法的名字, method是一个组合体, 包含了名字和实现.

### Core Foundation的内存管理

- 凡是带有Create、Copy、Retain等字眼的函数, 创建出来的对象, 都需要在最后做一次release
- 比如CFRunLoopObserverCreate release函数: CFRelease(对象);

### malloc和New的区别

- new 是c++中的操作符, malloc是c 中的一个函数
- new 不止是分配内存, 而且会调用类的构造函数, 同理delete会调用类的析构函数, 而malloc则只分配内存, 不会进行初始化类成员的工作, 同样free也不会调用析构函数
- 内存泄漏对于malloc或者new都可以检查出来的, 区别在于new可以指明是哪个文件的那一行, 而malloc没有这些信息。
- new 和 malloc效率比较
- new可以认为是malloc加构造函数的执行。
- new出来的指针是直接带类型信息的。

你是否接触过OC中的反射机制? 简单聊一下概念和使用 [g.csdn.net/qq\\_42792413](http://g.csdn.net/qq_42792413)