

48 在 GitHub 发布小程序作品：想收 offer 收到手软吗？

更新时间：2019-10-18 10:13:09



“人只有献身于社会，才能找出那短暂而有风险的生命的意义。

——爱因斯坦”

上一节我们学习了如何在现场面试时，向面试官展示自己的小程序作品。本节将介绍如何在自己的简历中使用 GitHub 展示自己的小程序作品。

GitHub 是什么应该不用再做介绍了吧，不知道的同学请阅读 [百度百科中 GitHub 词条](#) 或自行度娘。

1. GitHub 入门

在编写本节时，原本想做一个 GitHub 的详细入门讲解，但在收集资料的过程中发现网上已经有了非常丰富和完整的 GitHub 入门教程。因此，这里就不再重复造轮子了，直接将我收集整理的优质 GitHub 入门教程推荐给各位同学阅读：

官方教程

如果你能看懂简单的英文，那么毫无疑问 GitHub 官方的 [Hello World 教程](#) 是你入门的不二选择。

这份官方教程只需要 10 分钟就能阅读和实践完毕。

此外，你还可以在 GitHub 的 [Help 页面](#) 中探索和学习 GitHub 的各项功能。

慕课网教程

如果你感觉阅读英文的官方教程实在是一个不可能完成的任务，那么可以学习这个视频入门课程：

版本控制入门 – 搬进 [Github](#)。

这份入门教程你可能需要 2 小时才能观看和实践完毕。

GitHub 上的电子书

在 [GitHub](#) 上有很多基于开源协议的电子书，其中有一本 [GitHub 漫游指南](#) 内容很丰富。

该电子书内容很杂，有时间、有兴趣可以作为补充读物。

Tips:

以下是我十几年来采用的学习方式，它帮助我学会了汇编语言、C 语言、C++ 语言、Java 语言、C# 语言、各种前端框架，以及我们现在正在讲的小程序，供各位同学参考：

- 任何时候，如果我能找到一手的官方资料，我一定首选阅读官方资料。

任何第三方的资料或教程都是二手材料，一定没有官方资料权威，毕竟官方是产品开发者，他们才是最懂自己产品的人。

- 如果阅读完官方资料有疑惑或者理解不清楚的地方，我会再去网上搜索第三方资料。如果我的问题在至少 2 个不同的第三方资料中都有一致的解答，我会认为我找到问题的答案。

请注意，在中文世界中，你在不同的网站搜索到的结果很可能是复制粘贴的同一个第三方资料！不要认为在搜索引擎中点开两个网页看到了一样的内容就认为找到答案了，请仔细辨别。

2. 在 GitHub 发布小程序作品

在 [GitHub](#) 发布小程序作品，并不是按照前面几个入门教程讲述的方法，将“会员制社交电商小程序”的源代码上传到 [GitHub](#) 就完事了。

我们要发布的是自己的作品，而不是源代码。

自己的作品包含两层意思，第一层是“自己的”，第二层是“作品”。

“自己的”意思很直白，你发布的源代码要是你自己（阅读专栏的内容然后完成每一节实践作业）编写的，而不是下载专栏源代码然后直接上传到自己的 [GitHub](#) 项目中。

如果你直接抄袭专栏源代码和专栏内容作为自己的作品去面试，这是造假！

假的永远是假的，哪怕你成功入职了，在实际工作中你的真实水平是无法造假的，别忘了还有试用期呢。

一旦在试用期被发现应聘时表现的水平与试用期实际展现的水平不一致，相信绝大部分 **Leader** 和 **HR** 在试用期考核结果中都不会给转正的评语，在试用期被 **Fire** 几乎是必然的结果。

我就 **Fire** 过几十个简历造假的人，请不要抱有任何侥幸心理。“实事求是”是程序员最重要的品质之一。

“作品”的意思是：你需要在 [GitHub](#) 项目的 **README.md** 文件中，将这个项目使用的技术、功能、用途、代码结构等用你自己的语言进行详细介绍，另外别忘了介绍这个作品的作者（就是你自己呀）。

如果你的现场表达能力不是非常强，在 **GitHub** 上编写详细介绍可以将你现在的水平达到 **100%** 的程度，同时还可以完整梳理你在面试现场介绍项目的逻辑思路，在面试前多按你编写好的详细介绍演练几次，可以有效避免你在面试时，由于紧张导致翻车的情况（我见过有同学在面试时由于太紧张，讲话逻辑完全混乱、语无伦次，完全无法展示出自己实际的编程水平，这样太遗憾了）。

README.md 文件需要包括以下基本内容：

- 项目简介
- 使用技术
- 项目代码结构
- 功能清单
- 项目截图
- 作者联系方式（作者介绍）

“会员制社交电商小程序”的 **README.md** 内容可以在消化专栏第四章内容后，根据自己的理解，用自己的语言组织编写。

图 10 和 图 11 是 **GitHub** 中一些项目的 **README.md** 文件内容，供参考。

图 10 GitHub 说明实例 1

README.md

JavaQuark社区

license Apache 2

简介

- 1、JavaQuarkBBS是一款使用Java编写的简易社区系统。
- 2、采用前后端分离的机制实现。前台项目通过http访问RESTFulAPI获取信息渲染页面。
- 3、项目技术分层明显,模块分离,采用springboot构建模块。
- 4、前台页面来自FlyUI的开源社区模板

系统结构图

- 1、quark-common :采用了Springdata+MySql实现基础服务抽象,DAO层, Entity以及DTO
- 2、quark-admin: 采用springboot+shiro搭建的细粒度的基于URL的权限管理系统,进行帖子管理,回复管理,用户管理等操作
- 3、quark-rest: 使用springMVC搭建RESTFul服务,采用WebSocket协议+stomp协议搭建推送服务,实现一对一推送与一对多推送,面向各
- 4、quark-portal: 前台社区系统,使用springMVC进行页面跳转与拦截,采用前后端分离的机制实现。前台展示模块通过http协议访问RES使用LayUI, jQuery渲染页面渲染页面
- 5、quark-chat:采用Netty+WebSocket协议搭建的聊天室服务,通过JSON传递数据, Ping-Pong心跳检测机制保证链路可用性。
- 6、使用Redis进行了热点缓存, Ehcache进行数据库的二级缓存提高应用的效率

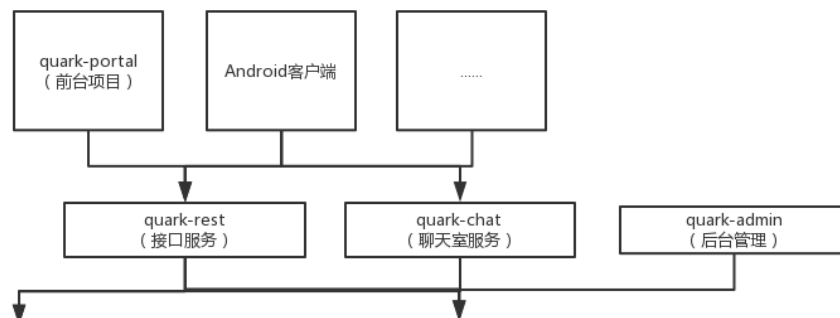


图 11 GitHub 说明实例 2

技术选型

前端

- web服务器: nginx
- 后台渲染: node.js
- M站框架集: vue, vuex, vue-router
- M站UI组件库: vux
- 后台管理框架集: react, redux, react-router-redux
- 后台管理UI组件库: antd
- 数据可视化: echarts
- 富文本编辑器: ueditor
- 打包工具: webpack
- 构建工具: gulp

后台

- web框架: iris
- 路由: httprouter
- 持久层框架: gorm
- 数据库: mysql

go依赖的第三方库

库	说明
gopkg.in/kataras/iris.v6	iris web框架
github.com/jinzhu/gorm	gorm 持久层框架
github.com/satori/go.uuid	uuid生成工具

项目结构

目录或文件	说明
docs	文档
...	...

如果还想详细的说明功能设计、数据库设计等内容，可以单独在源代码中建立一个 `doc` 文件夹，在其中存放所有详细的说明文档（如图 12 所示），同时在 `README.md` 文件添加这些文档的查看链接。

“会员制社交电商小程序”的详细设计内容，可以在消化专栏第五章至第十章内容后，根据自己的理解，用自己的语言组织编写。

图 12 GitHub 说明实例 3

← → ↺

GitHub, Inc. [US] | github.com/linlinjava/litemall/tree/master/doc

🔍 ☆ 👤

Branch: master | litemall / doc /

Create new file | Upload files | Find file | History

linlinjava doc

Latest commit 6aa5fdf 25 days ago

..		
pic	doc	2 months ago
pic1	doc	2 months ago
pic2	更新文档	8 months ago
FAQ.md	doc	2 months ago
README.md	doc	3 months ago
admin.md	doc: 增加接口文档	3 months ago
api.md	chore[litemall-admin, litemall-admin-api]: 管理后台API调整及文档	3 months ago
mobmall.md	笔误校正 (#179)	3 months ago
note.md	chore[litemall-admin]: 团购活动菜单项调整至推广菜单中	9 months ago
platform.md	doc	25 days ago
project.md	doc	2 months ago
wxmall.md	doc: 增加接口文档	3 months ago

📖 README.md

Table of Contents

- 介绍
- 更新日志
- 贡献指南
- FAQ
- API
- 1. 系统架构
- 2. 基础系统
- 3. 小商场
- 4. 管理后台
- 5. 轻商城

README.md 使用 Markdown 语法进行编写，我使用的 Markdown 编辑器是 Typora ，推荐给大家，各位同学可以到其官网下载：[Typora 官网](#)。

在根据本节内容完成你自己的作品后，你将自己的 **GitHub** 地址放到简历中才会提高拿到 **offer** 的几率。

下节预告

下一节，我们将回顾已经学习过的专栏内容，对专栏学习进行一次复盘。

实践环节

实践是通往大神之路的唯一捷径。

本节实操内容：

- 在阅读专栏的内容，完成每一节实践作业后，你应该已经编写出了你自己的“会员制社交电商小程序”。请将你自己编写小程序的源代码上传到 **GitHub**，并在 **README.md** 中编写项目介绍。

}



47 为什么你发布小程序的申请被
驳回了？

49 如果有一件事是必须要做的，
那就是复盘

