



看看考位助手 介绍文档

@微信小程序应用开发赛

参赛小组：辛苦了<se.1>

刘曦恺 王景楠 杨天诚

团队介绍

我们是 辛苦了<se.1> 组！



组长

刘曦恺

中南大学计算机学院

设计和实现了小程序的主要功能



一般组员A

王景楠

中南大学计算机学院

提供了验证码识别的机器学习相关技术支持



一般组员B

杨天诚

中南大学计算机学院

参与产品设计、项目管理和推广

1. 看看考位助手 解决了什么问题?

托福、雅思和 GRE 考试，在国内考位是比较紧张的，这导致很多考生需要提前、频繁地查询当地有没有考位，反复地使用容易忘记的ID号和密码登录考试官网查看考位信息。如果报名迟了，只能更加紧盯考试官网，以期望抢到不定时释放出来的考位，甚至求助于考位“黄牛”。

有没有人做出考位相关的工具解决这个问题呢？没有。打着考位查询旗号的网站、公众号和小程序全部都是“黄牛”的人工代报名服务。均需要收费 200 元左右。

本小程序基于爬虫技术和 Nodejs+Python 的高效系统设计，结合方便使用的微信小程序平台，实现了

- 1.考位查询，无需反复登录考试系统即可查询全国考位。
- 2.考位订阅，选择想要考试的时间地点，每当释放出考位时系统就会发送微信消息到你的手机。

实属考生的好帮手。

这是看看考位助手的一个真实反馈，来自西安的刘同学：



疫情期间，雅思考位频频取消，放出的考位却又寥寥。如果不能及时考到雅思，我就必须推迟留学。为了抢到尽早的考位，我甚至找过黄牛，他们也不能给我一个准信。我的好朋友说他开发的考位助手正在测试阶段，让我试试。我在小程序里订阅了陕西、6月6日、西安外国语考场，每天都通过小程序刷新查看。几天后的早上，微信推送了一条订阅消息，我去雅思官网一看，真的放出了考位，并且成功报上了名。我俩都非常开心！

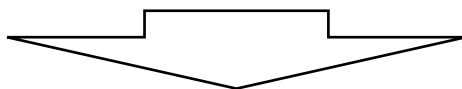
2. 看看考位助手 如何使用?



微信搜一搜

看看考位助手

搜索或扫码进入小程序



选择
托福/雅思



主页 Grid 导航

- 选择托福、雅思
- 联系开发者
- 展示公告弹窗

雅思/托福子页面

- 选择城市、考位查询
- 考位订阅
- 个人信息
- FAQ

主页

选择
联系开发者

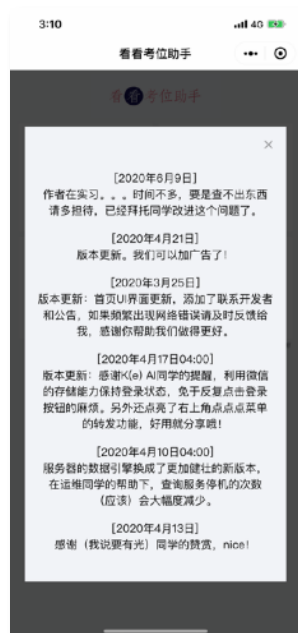


联系开发者子页面

- 利用微信的客服功能，用户获得技术支持

主页

选择
通知公告



公告弹窗

- 更新信息
- 考位放出通告
- 开发者的话

微信

考位释放
微信收到通知



微信服务通知

- 利用订阅消息功能，通知用户释放的考试日和考场

3. 技术实现

小程序端

开发工具上，我们选用了微信开发者工具，使用原生的开发方式，没有使用框架。并且开启了云开发的模式，将通知公告和页面需要下发的可变文本放在了云函数、云数据库里，保证页面的高效展示。

我们使用了 Vant 组件库，这是有赞公司开发的一个组件库，可以比较美观、方便地展示数据，让我们把工作重心放在功能的实现上。

我们接入了微信客服、微信订阅消息、微信广告等微信开放能力。并且用微信登录获取 uid，实现了用户鉴别。

后端服务器

前端接口服务

- 我们使用 NodeJS 开启一个 API 服务器，用于接受并且处理小程序发送来的请求。
- 进行的业务包括：返回数据库中的考位信息、考位订阅的增删查。

通知下发引擎

- 我们使用 NodeJS 编写了一个通知下发引擎。以及一个不断获取微信 API Token 的服务。
- 通知引擎利用 Redis 的 Subscription 功能，每当爬虫遍历考位信息完成的事件发出时，就遍历 Redis 中的考位信息字段以及 Mysql 中的订阅表，如有考位空闲就利用 Token 调用微信的通知接口，下发给订阅了此考位的用户。

Python 爬虫

- 爬虫使用 Selenium 运行一个 Geckodriver，即 Firefox 浏览器，手动登录后不断地请求考位信息。请求到的考位信息会暂时放在 pandas.dataframe 中。

- 为什么不模拟请求？因为神奇的报名系统会根据需要不断地变化对于请求中携带某个 Token 的要求，有时模拟请求是可行的，有时是不可行的，为了保证系统的稳定性，选择使用 Selenium。
- 每当所有考位遍历查询完成，爬虫就会将考位信息存入 Redis，并且发出消息，让通知引擎进行检查。

数据库

- 数据库方面我们使用了免费开源的 MySQL，建立两个表：
ielts_subscription 和 subscription 来存放用户订阅的考位。
- 我们还使用了内存数据库 Redis，存放需要经常写入取出的全部考位信息。考位信息不需要持久化，但我采取了特殊的设计保证即使爬虫的登录态过期，停止更新考位信息，Redis 中仍然会存在至少一个完整的考位表。这样小程序端就不会出现查询为空的情况。

4. 运营现状

本项目已经运营约 3 个月，累计用户 1300+，进入 5 月后因为组员开学、实习的原因缺少维护和推广，增长速度放缓。预计在升级利用了机器学习技术的爬虫、杨天诚重新开始推广后，会回到 4 月的增速。



用户达到 1000 后，我们接入了流量主功能，虽然用户的点击欲望不强，单凭曝光的收入不多，但对我们也是一个非常大的鼓励。我们也会考虑在订阅的流程上进行改进，让用户更愿意为我们点击广告。

账户收入

昨日收入 (元)

0.04

近7日收入 (元)

0.10

累计收入 (元)

11.57

5. 近期迭代计划

后端服务器

Python 爬虫

- 因为验证码的存在，现在的爬虫登录态结束后依赖组员手动连接服务器进行重新登录。现在王景楠正研究机器学习的验证码识别的模型训练，完成后可以实现考位查询不间断。对于小程序用户的查询体验和通知引擎的时效性都是一个质的飞跃。

网络延迟和服务性能

- 现在服务器是购买的 VPS，如果用户群不断扩大，考虑到用户体验以及 Selenium 对于硬件的消耗大，可以在家自建服务器或购买腾讯云的服务器，可以降低爬虫的查询延迟和用户的请求时间。目前已经完成自建服务器的技术预研，在家部署了一个公网可访问的 Ubuntu 服务器。
- 如果广告收入增长，可以覆盖服务器开销，优先考虑购买腾讯云的服务器。

小程序改版

用户界面

- 增设埋点、进行用户体验调研，让小程序本身更加易用。
- 完善提示信息，优化 FAQ 的文案。

业务流程

- 考虑开发一个促进用户点击广告的新流程。但要保持免费使用的初心。
- 添加卡片分享功能，用户可以更方便地向他人展示考位情况，促进用户分享行为，以促进用户增长。
- 添加跳转或是复制链接的功能，使用户收到通知后可以更快的触达报名系统