学号\_\_2012302660013\_

密级\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

武汉大学本科毕业论文

个人求职及企业招聘信息app系统

院（系）名 称：国际软件学院

专 业 名 称 ：空间信息与数字技术

学 生 姓 名 ：谢向飞

指 导 教 师 ：张昀 教授

二○一三年六月

**郑 重 声 明**

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名： 日期：

摘要

目前已经存在的移动求职应用，比如智联招聘、领英等知名产品，面向求职者提供的是全职职位、实习职位、兼职职位等多种职位类型的招聘。这样虽然覆盖的范围广，面向的用户群大，但是对于只想应聘兼职这一垂直类目的人群而言，这些产品就显得过于繁琐、复杂，用户体验并不友好。本项目采用目前已经非常成熟的移动应用开发技术，只针对兼职这一垂直类目，根据这类的职位的特点，面向广大的兼职职位应聘者提供更加友好的服务。最终的产品包括用于求职者的Android客户端应用，以及用于企业的Web网站。在产品上，对职位、企业按照热度、距离、城市范围等多种方式进行分类检索，并提供对职位、企业的搜索；在设计上，采用Google Material Design设计语言，打造简约、易用的用户界面和流畅的人机交互。通过这两方面的结合，开发出精而美的应用，极大地提高用户体验。

关键词：兼职；用户体验；Android

**ABSTRACT**

Nowadays, there are many mobile applications that are created to offering users the service of finding jobs, such as 51Job and LinkedIn. Jobs on these platforms consist of many categories, such as full-time, part-time and internship. So, the target users of these applications are those who want to find a job of any categories. However, when a person want to find a part-time job, all of full-time jobs and internship jobs are interferences to him or her. Thus, the process to find a part-time job will not be really easy and efficient. If taking into account the traits of the part-time jobs, it is possible to make the process of finding a job easier and more efficient, that is, better experience. This program aims to create an application, through which users want to find a part-time job. All jobs on this platform are part-time. The program consists of two products, one is an Android application, through which users can find jobs they want, and another is a Website, through which enterprises can commit their jobs for users. On the one hand, jobs and enterprises in the application can be retrieved by many means, quantity of applicants, distance or districts. And also, users is able to search jobs or enterprises by keywords. On the other hand, Google Material Design language, which is a very popular design rules and also the official standard of the newest Android version, is used to develop the mobile application. Through the both means, with better human-machine interaction, the experience to find a part-time job will become easier and more efficient for target users.

**Key words:** part-time job; UE; Android

Contents

[1 绪论 12](#_Toc447019808)

[1.1 选题的背景 12](#_Toc447019809)

[1.1.1 国内外研究现状 12](#_Toc447019810)

[1.2 课题的研究方法 12](#_Toc447019811)

[1.3 课题的预期结果 13](#_Toc447019812)

[2 总体设计方案 14](#_Toc447019813)

[2.1 平台服务流程 14](#_Toc447019816)

[2.2 技术架构设计 14](#_Toc447019817)

[2.2.1 Android客户端技术架构 14](#_Toc447019821)

[2.2.2 Web端技术架构 14](#_Toc447019822)

[2.2.3 服务器端技术架构 14](#_Toc447019823)

[3 详细设计与实现 16](#_Toc447019824)

[3.1 用户需求分析 16](#_Toc447019827)

[3.1.1 兼职职位的受众规模 16](#_Toc447019828)

[3.1.2 已有系统的现状与缺陷 16](#_Toc447019829)

[3.1.3 改进的措施 17](#_Toc447019830)

[3.2 产品功能 17](#_Toc447019831)

[3.2.1 Android客户端的功能 18](#_Toc447019832)

[3.2.2 企业Web端的功能 18](#_Toc447019833)

[3.3 Android客户端的实现 19](#_Toc447019834)

[3.3.1 用户界面设计 19](#_Toc447019835)

[3.3.2 开发环境与开发工具 20](#_Toc447019836)

[3.3.3 数据交互以及格式规范 22](#_Toc447019837)

[3.3.4 编码设计与规范 22](#_Toc447019838)

[3.3.5 LBS支持 23](#_Toc447019839)

[3.4 服务器端的实现 23](#_Toc447019840)

[3.4.1 整体架构设计 23](#_Toc447019841)

[3.4.2 MVC分层架构设计 24](#_Toc447019842)

[1.1.1 数据库设计规范 26](#_Toc447019843)

[3.6 系统安全性 27](#_Toc447019851)

[3.6.1 Android客户端的安全 27](#_Toc447019852)

[3.6.2 服务器端的安全 28](#_Toc447019853)

# 绪论

## 选题的背景

如今，越来越多的大学生在课余时间选择做一份或几份兼职工作，也有越来越多的企业、商家有意向提供这样的职位，而且，兼职类职位的受众用户在百万级别。当前主流的招聘平台，如大街网、智联招聘等，提供的是全职、兼职、实习等几乎全部类目的职位。尽管这些平台对不同类目的职位做了分类，但对只想寻找兼职类职位的用户来说，其他类型的职位仍然是一种无意义的干扰。这在一定程度上降低了这部分用户的使用体验。平台方要兼顾所有类目的职位，难以只针对兼职类职位做出针对性的改善以提高用户体验。因此，开发一款只针对兼职职位的垂直类应用，可以专注于这一特定类目进行针对性的优化，从而极大地提高目标用户的使用体验。

### 国内外研究现状

在产品上，国外的求职类应用的代表是领英，提供的是所有种类职位的招聘，并且提供了基于职场的社交服务。基于这些，领英也是一款大而全的应用，面向的用户群体也很广泛。国内主流的求职平台包括大街网、智联招聘，同样地，这些平台也是横向面向所有求职者。在垂直细分领域，拉勾网就是专注于互联网行业的招聘，猎聘网专注于高端招聘，这些平台能够根据某一行业或某类职位的特点进行针对性优化，从而大幅地提高其用户体验，因此在近几年中都取得了迅猛的发展。

在技术上，移动应用开发已经经历了超过5年的快速进步，因此，相关的开发技术非常成熟。Android SDK已经更新到了23.0版本，新的UI设计规范Material Design被广泛地使用，各类开源库也层出不穷。在Web开发中，React与AngularJS等新一代前端框架正在代替JQuery等成为主流。

## 课题的研究方法

对于本课题，主要采用的研究方法是大量观察已有产品的特点，综合大量产

品的优缺点，从而发现亟需提高用户体验的地方。另一方面，通过用户调研等方式了解兼职这一垂直领域相对于全职、实习等所独有的特点，从这些特点下手采取对应措施来提升用户体验。

在开发技术上，调研并采用目前主流、成熟的开发方案，通过Wiki、开发者社区以及相关论文等了解、研究相关技术。在做整体的应用架构设计时，通过了解一些现有的知名应用的技术选型和软件架构，再结合本应用的特点以及开发者的个人能力，确定应用的最终技术结构。

## 课题的预期结果

本课题最终开发出一款能够商用的求职应用，能够针对兼职类职位招聘这一垂直细分领域在一定程度上大幅提升相应的用户体验。最终的产品包括一款面向求职者的Android应用，一个面向企业的Web站点，和一个用于后台审核人员的管理网站。

# 总体设计方案



## 平台服务流程

企业通过网站申请注册，在填写完企业信息并上传营业执照后，等待平台人工审核。审核通过后，企业可开始发布职位、接收求职者的申请等。

求职者注册并验证手机号码后，可进入应用浏览、收藏职位和企业，完善简历后才可申请职位。

## 技术架构设计



### Android客户端技术架构

在Android客户端开发中，采用Android SDK version 23.0版本，官方Android Studio作为开发IDE。遵照Google Material Design 规范来设计应用的用户界面和交互流程。

这样做的目的在于采用官方推荐的开发方案，也是目前主流的方案。这些技术已经经过了科学有效的论证和实践，用户体验非常友好。这些技术已经非常成熟，不仅容易使用，而且存在大量的第三方辅助库和开源库，可以大大地加快开发进度。

### Web端技术架构

在Web开发中，使用Facebook公司开发的AngularJS作为前端框架，进行DOM操作、数据绑定以及其他各类操作，降低与服务器端的耦合性，实现前后端完全分离。在UI设计上，采用Material Angular开源库设计具有Material Design 风格的页面。

### 服务器端技术架构

为了进一步降低系统耦合性，提高系统整体的可用性，整个服务器端包含三个完全独立的子系统：Android客户端服务器、企业Web端服务器以及用于后台审核的Web端服务器。这样，当其中一个模块发生内部错误时，不会影响其他模块的正常使用。

三个子系统使用相同的技术架构。MySQL作为数据存储系统，采用Python作为服务器端开发语言，并使用基于Python的Flask作为后端HTTP框架。这样做有利于快速开发，从而可以将更多精力集中在业务逻辑上。

# 详细设计与实现



## 用户需求分析

### 兼职职位的受众规模

根据国家统计局 2014年2月24日发布的统计信息中第十项的数据统计，2014年全年普通本科、专科在校生人数达2468.1万。在这一兼职工作主力人群中，有超过68%的人在课余时间做过兼职类工作。再加上社会上的其他人群，可以很明确地得出结论：中国兼职行业的人数规模是在千万级别的。面对这一庞大的规模，开发独立、体验良好的应用是有很大的潜在市场需求的。

### 已有系统的现状与缺陷

以目前主流的平台大街网的Android客户端应用为例，来分析已有系统针对兼职招聘用户体验有待提升的地方。

从上图可以看出，大街网对于兼职类目只列出了所有的职位，并没有列出所有提供兼职类职位的企业。一些大量提供兼职类职位的企业在这里得不到展示，因此求职者无法浏览某个企业所提供的全部兼职职位。求职者必须跳出兼职页面找到企业，然后浏览企业发布的所有种类的职位，自行从中发现兼职职位。这将多出很多步骤，用户体验也必会有所下降。

另一方面，从图片中可以看出，大街网对于兼职类职位的分类排序的依据有地点、行业、薪水结算周期三种。然而，对于求职者来说，这三种并不是最主要的参考依据。一名典型的求职者，如大学生，想利用课余时间抽空参加兼职，当寻找一份工作时，工作地点与求职者所处位置的距离远近、该职位有多少人已经申请过等这些因素，往往决定了他们是否会选择这份工作。

### 改进的措施

针对上节分析的结果，可以从以下几方面进行改进。

1. 提供浏览所有发布兼职职位的企业的入口。

在全局导航目录里，提供一个与浏览所有职位同级别的入口，通过该入口求职者可以浏览所有企业。这样求职者可以针对企业的进行职位选择。

1. 对职位和企业依照距离、申请人数等进行分类。

对职位列表和企业列表按照发布时间、与求职者所处位置的距离以及职位的申请人数等多种方式分类排序，使得求职者可以结合自己的喜好等浏览到更适合的职位。

1. 求职者能够查看某一企业所提供的全部职位。

在进入企业详情的页面中，求职者可以浏览该企业所发布的全部兼职职位，这可以使得求职者完全按照企业发现职位，相对于大街网等来说多了一个更直观的入口。

1. 提供对企业、职位进行搜索。

求职者能够搜索某个职位或者企业，这将提供很大的自主选择权。

1. 求职者能够锁定在某一城市范围内进行浏览。

对于大多数求职者来说，他们只会选择在本城市内工作的职位，因此，其他城市的职位完全没有必要展示出来。

## 产品功能

通过大量调研分析和观察现有产品所存在的不足，最终设计出的产品以提升用户体验为全局目标，各个子系统所具备的功能均服务于这个目标。

### Android客户端的功能

求职者首先需要提供手机号码进行注册，应用通过短信验证码的方式验证用户有效性，注册后的有效用户可以使用账号登录客户端应用进行后续的浏览、申请等一系列操作。已注册的用户可以通过找回密码的功能来重新设定一个新密码。

登陆后的求职者可以进入个人中心，在这里，可以修改基本的用户名、头像等。通过个人中心，求职者可以简历、编辑、查看自己的简历。求职者也可以查看所有自己已经收藏的职位和企业以及申请过的职位，并且可以进行取消收藏某些职位、企业等操作。

求职者可以在首页浏览所有职位和企业，企业和职位的分类排序方式有时间、距离、申请或收藏人数等三种方式。可以通过首页顶部的搜索栏根据名称搜索职位或者企业。

求职者可以修改城市范围，也包括全国这一范围。

在职位列表点击某个职位可以进入职位详情，该页面详细描述职位的全部信息。求职者可以在该页面选择申请、收藏或者取消收藏该职位。求职者具备申请资格的前提是至少已经建立了简历，并且在简历中填写了姓名、电话、邮箱等信息。

求职者可以通过点击企业列表中的单项或者通过点击职位介绍中的所属企业进入企业详情页面，该页面详细描述企业的信息，同时也列出该企业所发布的全部职位。求职者可以在该页面收藏或者取消收藏该企业。

### 企业Web端的功能

企业用户可以使用邮箱进行注册，注册后的企业首先需要提供企业营业执照和企业法人身份证件照片进行审核，审核通过的企业方可进入网站平台。

审核后的企业可以在网站上完善企业信息、发布职位、接收申请、查看求职者简历或者选择拒绝求职者的申请。

## Android客户端的实现

### 用户界面设计

Android客户端的界面设计以Google Material Design规范为基本原则，目标是打造出简约、易用的用户体验。

在具体的实施过程中，采用NavigationView侧边栏导航抽屉提供全局导航服务，在该处呈现顶级目录，即职位、企业、收藏、用户中心入口等。



图 3.1

在职位、企业列表展示中，采用卡片流式布局，通过顶部选项卡的方式进行分类，使用户的快速浏览过程更加流畅。

其他方面，使用SnackBar代替原来的Toast显示通知消息，使用Toolbar代替原来的ActionBar作为顶部标题栏。

### 开发环境与开发工具

技术开发过程中，基于Windows 8.1操作系统，采用JDK7作为JAVA开发包和运行时环境。

使用Android Studio作为集成开发环境。这是官方推荐的IDE，相对于经典的Eclipse和ADT插件的搭配方式，Android Studio的智能提示、模糊搜索等功能更加强大，String与Color资源可以实时预览，并且完美整合了版本控制和大量插件。这将大幅提升开发效率。

项目使用Gradle[[1]](#footnote-1)作为自动构建工具，相对于Maven、Ant等传统构建工具，Gradle抛弃了基于XML的配置，而是采用领域特定语言Groovy[[2]](#footnote-2)的配置，大大简化了代码行数，提高了可读性。除此之外，Gradle的第三方开源库数量非常大，对于Github上的开源项目支持得非常好。

项目使用Git作为版本控制工具，该工具可以很方便地加入到Android Studio中，每一个版本的代码都同步转储到Github的代码仓库中。这样，在进行版本迭代的同时，也对代码进行了备份。

为了使得开发过程更加快速、敏捷，从而将更多精力用于业务逻辑，在Android客户端开发中使用了大量流行的第三方开源库，包括网络库、UI库、持久化存储等多种。

GSON：Google开源的JAVA JSON库，相对于Android原生JSON库、fastjson[[3]](#footnote-3)等库使用起来更加简单。因此在项目中使用该库进行网络数据的序列化、反序列化操作。

OKHttp：Android SDK 19.0版本默认在HttpURLConnection中使用的HTTP网络库，支持多种HTTP方法，文件上传等非常方便。为了兼容低版本，因此在项目中单独引入该库，进行HTTP交互、上传图片、求职者简历文件等。

ButterKnife：Android视图注入工具，可以通过JAVA注解的方式注入视图、事件方法等多种类型，从而将开发者从繁琐的通过资源ID绑定View类的过程中解放出来。

Fresco：Facebook开源的异步图片加载库，目前开源社区性能最好的第三方库之一。同时支持加载网络图片资源和本地图片资源，还提供了多级缓存，在视图层还支持图片圆角、动画等等。项目使用该库加载企业图标、用户头像以及简历中的照片等。

ActiveAndroid：本地SQLite数据持久化库，可以摆脱具体的SQL语句，极大简化简历数据库连接、游标遍历记录、基本CRUD[[4]](#footnote-4)操作、以及释放数据库连接资源等一系列过程。项目中使用该库存储全国城市数据、用户配置数据和部分职位、企业等的离线数据。

### 数据交互以及格式规范

项目使用HTTP协议与服务器进行交互、获取数据等。

在客户端与服务器的交互中，使用JSON作为数据格式协议。相对于XML等其他数据格式，JSON的结构更加容易映射到一般编程语言的数据结构中，特别是在服务器使用Python的情况下，JOSN格式可以很好地对应到Python的dict，也就是字典类型中，因此，序列化与反序列化的过程更加直观、简单。

用户初次打开应用时，后台将会从服务器下载城市列表数据并持久化存储在本地SQLite数据库中。这样，用户在以后的选择城市操作中可以直接从本地加载，避免多次从服务器获取大量的数据，不仅节省流量，还可以减少等待时间。

### 编码设计与规范

在具体的编码过程中，实施了良好的编码规范和结构设计，使得开发过程高效、敏捷，良好的编码对于以后的代码修改、维护、调试也非常有利。

项目对源码包结构做了细分。在根包名目录下，将Activity、Fragment、Model、Network、Utility等不同用途的类放在同一个包中。对于Activity、Fragment等常用类，按照面向对象的思想，分别抽象出了一个共同基类，在这个抽象基类中，定义了一系列通用的方法，比如简单地显示进度条对话框、SnackBar显示简单的文字内容、Activity跳转、Fragment切换等等。

在变量、方法等的命名中，遵守驼峰命名法[[5]](#footnote-5)。

### LBS支持

应用需要定位用户目前的位置，以便可以推送附近的职位和企业。项目使用百度公司的LBS Android SDK，该工具提供了定位服务，并且可以指定定位使用的精度、定位方式如GPS或者基站等诸多选项。除此之外，对于定位的结果不仅提供了经纬度，还提供了分层级的省份、城市、街道等详细地址、POI[[6]](#footnote-6)数据等非常详尽的内容。该工具由百度开放平台免费提供给开发者。

## 服务器端的实现

为了提高服务器系统的可用与高效，项目将Android应用的服务器与Web端的服务器实现为两个独立的实例。两个实例完全采用相同的架构和设计，只在业务逻辑方面有所不同。因此，为了方便描述，在本节内容中，将Android应用服务器与Web端服务器统称为服务器。

### 开发工具和运行环境

服务器整体部署在Ubuntu 14.04 LTS操作系统上，采用基于Python语言的Flask作为Web框架，并且使用了Flask自带的Web Server。在数据库层，采用了MySQL。

服务器采用Python 2.7开发，主要目的是提高开发速率和编码效率。另外，Python语言的字典数据构与JSON数据格式天然融合，JSON操作比Java只能通过类对象方式映射简单得多。开发过程中使用JetBrains公司开发的PyCharm Community作为IDE工具，该软件是目前使用最广泛的Python IDE。

服务器部署在阿里云Ubuntu主机上，系统版本为14.04 Long Term Support。该系统本身自带Python 2.7环境，以及SSH[[7]](#footnote-7)、Vim[[8]](#footnote-8)等非常常用的工具。在编码过程中使用了pip[[9]](#footnote-9)作为Python包的依赖管理工具，安装额外的第三方包。选择该环境作为服务器主机使得部署过程更加快速、简单。

Flask是一个用Python编写的轻量级Web框架，支持扩展插件以及插件热插拔。Flask社区拥有丰富的插件，包括身份验证、数据库ORM等等。Flask将装饰器与视图函数绑在一起，这使得开发过程变得更加简洁。除此之外，Flask的BluePrint组件可以很容易地将一个大应用拆分成各个子模块，因此，项目使用Flask按照业务分类将整个系统切分成多个BluePrint子模块，避免编写出代码量过大的单文件和业务过于复杂的单模块。

项目使用的MySQL版本是5.6，因为该版本开始支持JSON格式数据存储，这对于如企业的分条介绍、位置信息的层级表示都非常有利。MySQL是目前大量中小型应用使用的数据库系统，是传统的关系型数据库，并且支持多种存储引擎，目前主流的引擎有InnoDB和MyISAM。项目使用InnoDB作为存储引擎，主要考虑到能在大量并发的情境下能有更好的性能。

### MVC分层架构设计

MVC指的是Model-View-Controller，即数据-视图-业务逻辑，是一种软件架构设计典范，将业务逻辑、数据、界面显示等各层分离开来，在修改其中一层的代码时不会影响到其他各层。项目使用MVC模式进行设计，尽量做到各个层次完全隔离。

在视图层，主要内容是URL设计和数据交互格式规范；在数据层，主要是数据库设计和基本的的CRUD操作；在业务逻辑层，主要是实现功能需求，以及用户验证等等。

#### RESTful URL 设计

REST，即Representational State Transfer，表现层状态转移，是一种资源描述规范，符合该规范的HTTP URL就被称为RESTful URL。REST可以使URL具有很强可读性的，具有自描述性，利于提供开放API，便于第三方系统集成，提高互操作性。项目使用REST原则来设计交互URL。

REST专注于将要请求的对象描述为资源。对于资源的增删改查，对应于HTTP协议中的PUT、DELETE、POST、GET方法，对应到项目的一个场景就是发布职位、删除职位、更新职位信息、查看职位信息。RESTful URL中不应该出现名词，而应该都是动词，对于数据实体都采用JSON格式，分页等场景需要用到的参数则作为URL的查询参数附在URL后面。

下面的表格列出了项目中的一些典型的业务场景的URL，这些URL都是符合REST设计规范的。

表 1.1 服务器RESTful URL设计实例

|  |  |  |
| --- | --- | --- |
| URL | 方法 | 解释 |
| http://coocol.cc/collection/4357/job?count=10&start=30 | GET | 获取用户4357收藏的职位，从第30条开始，获取10条 |
| http://coocol.cc/enterprise | PUT | 创建一家新企业 |
| http://coocol.cc/profile/2321 | POST | 用户2312修改了用户资料 |
| http://coocol.cc/job/45 | DELETE | 删除ID为45的过期的职位 |

#### 视图层的URL路由

在视图层，项目使用Flask的BluePrint组件将视图函数做了模块划分。总共包含三个BluePrint，分别命名为user、job、enterprise，负责路由各自相关的URL，并且在不同的Python文件中编码实现，然后声明一个入口级的Flask对象app，最后，将这三个BluePrint注册到app中。这样，就完成了三个子视图模块的聚合。每个视图函数都接受请求并解析JSON数据，然后将具体的处理交给下一步的业务逻辑层去处理，最后JSON格式化相应的处理结果并返回。

#### 业务逻辑层的实现

业务逻辑层也分为user、job、enterprise三个不同的子模块，编码在不同的Python文件中，分别处理各自相应的事物。

业务层使用数据层提供的数据库访问接口与数据库通信，数据层只提供比较基本的CRUD操作。对于不同的业务使用的SQL语句也不相同，因此，该层也需要提供业务相关的较复杂的SQL语句。

项目使用阿里大鱼[[10]](#footnote-10)提供的短信服务向用户手机发送短信验证码，该服务只负责发短信，验证码生成以及验证均由服务器完成。

当用户请求附近范围的职位或企业时，业务层需要根据用户当前的经纬度和存储在数据库中的经纬度计算距离并按照大小排序。距离的计算模拟球面距离的计算，具体的公式为：

#### 数据层的实现

数据层主要是建立数据库连接，并进行一系列的增删改查。项目使用mysql-connector-python[[11]](#footnote-11)作为数据库驱动，该层负责建立数据库连接、与数据库交互、释放数据库连接资源等。

服务器所有的子系统，包括Android应用服务器和Web服务器都使用同一个数据库。主要原因是，项目中的所有数据因为业务逻辑耦合度较高。例如，企业发布、修改职位要写对应的数据表，而客户端检索职位、企业时又要从相同的数据表中获取数据，使用同一个数据库可以减少跨库操作的代价，而且也会减少数据的冗余。

在数据库设计中，所有的数据表均使用UTF-8编码。为了提高性能，数据库一概不设置外键约束，数据一致性的检查由业务层去完成。原因是对于外键约束，数据库每次的更新、插入等操作都需要检查外键值是否有效，这不利于插入、更新等操作很频繁的场景。



## 系统安全性

一个健壮的系统必须在一定程度上保证安全性，这包括防止泄露用户信息特别是密码、用户认证、防止第三方攻击如SQL注入等。项目通过在各个模块采取相应的安全措施，来提高整体系统的安全性。

### Android客户端的安全

Android系统上的每一个应用都是一个Java虚拟机实例，每个实例都运行在自己的沙箱中，对于其他应用是不可见的。Android会为每个实例创建一个新的执行用户，进一步提高应用的独立性与隔离性。因此，只要应用没有ROOT权限，就无法越过系统访问到其他应用空间。这些都是Android提供的安全机制，除此之外，应用还需要自己提供额外的安全措施。

用户注册时，客户端会将提交的密码通过sha1[[12]](#footnote-12)的方式哈希处理后再发往服务器，保证用户密码不会明文在网络上传输。

注册成功后，客户端不会存储用户密码，而是存储由服务器结合用户密码使用特定算法生成的token字符串作为以后访问的凭证。这就避免了用户密码在本地泄露的可能性。

对于存储在本地的用户信息和用户配置信息，应用全部采用了Android的private模式，即意味着其他任何应用和用户都不会访问到这些数据。

### 服务器端的安全

#### 用户密码存储

服务器在存储用户密码的时候，需要保证用户密码安全。服务器绝对不能明文存储用户密码，明文存储意味着服务器运维人员可以看到用户的密码，而且，如果服务器数据库泄露，密码也将公之于众。

其次，服务器也不能使用加密算法对密码加密，因为加密算法是双向的，只要拿到了加密使用的算法，就能通过该算法进行解密。因此，服务器处理密码应当使用单向函数，md5、sha1等哈希算法就是常见的单向算法。

最后，服务器要保证，假如数据库被泄露了，攻击者应当无法根据数据库中存储的处理后的密码获取到用户原来输入的真实密码。

Rainbow table，即彩虹表，是一个已经利用常见哈希算法处理过一定长度的所有密码组合的预先计算好的表, 常用于破解哈希过的密码散列。一般主流的彩虹表都在100G以上。这是一种以空间换取时间的案例，攻击者可以通过枚举彩虹表中哈希后的结果与获取到的密码作比较，从而反推出原密码。

目前，常见的密码处理办法有以下几种。

数据库直接存储md5或sha1哈希处理后的结果作为密码。简单地用公式可以描述为：

(3.1)

显然，在彩虹表攻击面前，这种方式是极为不安全的。

另一种方法是服务器可以随机生成一串字符串，叫做salt，然后用这个字符串加上用户密码再做哈希，用公式描述为：

(3.2)

使用这种方式处理时，数据库也需要保存下salt用来验证。假如字符串足够随机、足够长的话，可以很大程度上抵抗彩虹表攻击。然而，当攻击者拿到salt的话，依然可以轻松破解。

因此，为了提供更高的安全性，项目使用了bcrypt开源工具处理密码。bcrypt是专门为密码存储而设计的算法，最大的好处是有一个参数（work factor), 可用于调整计算强度从而使得攻击时间代价大幅上升，而且work factor是包括在输出的摘要中的。随着攻击者计算能力的提高，使用者可以逐步增大work factor，而且不会影响已有用户的登陆。bcrypt经过了很多安全专家的仔细分析，使用在以安全著称的OpenBSD中。

#### 用户认证

Android客户端在请求职位、企业列表以及职位、企业的详细数据时服务器不需要验证请求用户的身份。这类数据不会涉及任何用户信息，而且，省去用户验证的步骤可以使得响应更加及时，这对于这些非常频繁的请求来说，将很大提高用户体验。

在涉及到用户相关的请求时，例如收藏职位、修改信息等，需要验证用户身份。客户端用户在填写用户名、密码登录后，服务器会返回一个字符串token，客户端在以后的请求中使用该token作为访问凭证。用户每次手动登陆后，该token都会得到更新。

Web端用户在请求访问时，服务器端使用session进行验证。

#### 数据库安全

数据库安全的主要内容是防止SQL注入。SQL注入是很常见的攻击方式了，是指把SQL命令插入到Web表单提交或输入域名或页面请求的查询字符串，最终达到欺骗服务器执行恶意的SQL命令。例如，在服务器的业务层拼装了这样一条SQL语句：

s = 'select id from user where phone = ' + phone

其中，phone是客户端提交的参数。如果参数是一个正常的电话号码值，那么SQL语句s会被正确地执行。但是，如果客户端提交的phone参数值是：

phone = '1354246; delete from user;'

那么，拼接后被执行的最终语句将是：

s = 'select id from user where phone = 1354246; delete from user;'

显然，user数据表会被清空，这将是灾难性的后果。

因此，数据层的数据库访问需要使用预查询语句，即PreparedStatement，避免所有拼接字符串的行为。预查询语句可以使得SQL语句在执行前已经被编译过了，最终执行时只需要提交相应的参数。没有了编译的过程，上述的恶意参数就不会被当作命令执行。

## 系统测试与调优

测试是软件开发过程中必不可少的一项实践。单元测试针对小模块进行测试，这个小模块简单地可以是一个函数、一个类或者一个文件等。集成测试针对许多小单元组成的更大的模块进行测试，例如对应到的一个功能等。整体测试是测试整个系统的可用性。通过一系列步骤的测试，可以发现代码中的错误、性能的瓶颈以及功能上的逻辑漏洞。

### 单元测试

项目的单元测试由开发者在开发过程中负责完成，主要测试单个函数或者单个类。在Android客户端开发过程中，项目使用JUnit这一主流的JAVA平台测试工具，结合JAVA Annotation可以进行简易、快速的单元测试。在服务器开发中，使用Python官方标准库unittest做单元测试。最终，所有测试用例都通过后，在部署时移除掉测试代码和依赖库。

### Web服务器压力测试与调优

为了能使系统承受住一定量的并发访问量，需要对Web系统进行压力测试，然后综合测试结果进行调优。最终达到可用的目的。

项目使用JMeter对服务器进行了压力测试。JMeter是Apache[[13]](#footnote-13)组织开发的一个基于Java的开源的压力测试工具，该工具能用于HTTP服务器、数据库、FTP服务器等，以多线程的方式请求服务器，通过对响应时间以及超时时间的分析，从而评估服务器能够容纳的并发访问量。

首先，对服务器在使用各个软件默认配置的情况下进行压力测试，测试的结果如下表所示。

### 数据库压力测试与调优

# 结论

## 项目的成果

项目最终设计出了一个可以商用的企业兼职职位招聘系统。该系统的各个子模块均能按照预期运行。

Android客户端根据兼职职位相对于全职、实习等职位独有的特点，对职位、企业的浏览、发现等按照新的方式进行排序、设计。再结合更加符合Android Material Design规范的UI设计，使得整体的用户体验获得了一定程度的提升。Web网站能够准确地按照预定流程完成职位发布、接收申请以及其他操作。服务器能够按照预期高效、稳定地运行，使得整体成为一个完全可用的系统。

## 项目的研究意义

1. Gradle是新一代的项目自动构建工具，它可以使得项目的依赖管理、代码编译、测试、静态代码分析、项目部署等一系列过程完全自动化。 [↑](#footnote-ref-1)
2. Groovy是一种基于JVM的敏捷、简约的动态语言，语法习惯接近于Ruby。 [↑](#footnote-ref-2)
3. fastjson是阿里巴巴公司开源的JSON序列化、反序列化库，是目前效率较高、使用较广泛的库之一。 [↑](#footnote-ref-3)
4. CRUD即Create、Retrieve、Update、Delete的统称，指的是关系型数据库中基本的增删改查操作。 [↑](#footnote-ref-4)
5. Camel-Case，即驼峰命名法，是一中变量命名规范，对于单词使用首字母大写的方式进行命名，例如userPhoto、setName。常用于Java、C++等语言中。与Python等语言常用的下划线方式并列为两种最常见的命名规范。 [↑](#footnote-ref-5)
6. POI是Point of Interest的缩写，每个POI包含四方面信息，名称、类别、经度纬度、附近的酒店饭店商铺等信息。 [↑](#footnote-ref-6)
7. SSH指的是一种非常流行的远程登录会话的工具。 [↑](#footnote-ref-7)
8. Vim是Unix/Linux社区非常著名的基于终端的编辑器。开发、运维人员往往先用SSH登录远程主机，然后在终端中使用Vim进行编码等。 [↑](#footnote-ref-8)
9. pip是Python社区最流行的包依赖管理工具，可以通过非常简单的命令安装一个第三方包，开发者不用处理该包所依赖的其他包以及版本等问题，pip会自动处理这些。 [↑](#footnote-ref-9)
10. 阿里大鱼是阿里巴巴公司为开发者提供的通信服务，包括短信、邮件发送、语音通话等等，网站地址是http://www.alidayu.com [↑](#footnote-ref-10)
11. MySQL-connector-python是Oracle官方开发的MySQL数据库驱动，用于Python环境下。 [↑](#footnote-ref-11)
12. sha1是一种单向的哈希算法，安全性能比md5更高，使用经过该算法处理后的结果去推测原本的输入值是不可能的。 [↑](#footnote-ref-12)
13. Apache软件基金会是开发开源软件的非赢利性组织，该组织支持、开发了许多有名的开源软件以及工具，其中，最著名的是Apache Web服务器软件。 [↑](#footnote-ref-13)