**项目介绍文档**

**“呼叫邻里”农村招聘小程序v1.01**

小程序AppID: wx874904c38445429a

**开发人员**: 林德福2014580

刘志昊2014581

**文档编写**: 林德福2014580

刘志昊2014581

**编写日期**: 2022.05.24

目录

[一、项目陈述 1](#_Toc105524040)

[1.1 问题陈述 1](#_Toc105524041)

[1.2 利益相关者需求分析 1](#_Toc105524042)

[1.3 需要实现的功能梳理 2](#_Toc105524043)

[二、系统需求 2](#_Toc105524044)

[2.1 功能性需求列举 2](#_Toc105524045)

[2.2 非功能性需求列举 3](#_Toc105524046)

[2.3 屏幕显示需求 4](#_Toc105524047)

[三、系统分析与设计 4](#_Toc105524048)

[3.1 利益相关者 4](#_Toc105524049)

[3.2 样例图分析 5](#_Toc105524050)

[3.3 时序图分析 5](#_Toc105524051)

[四、系统结构的实现与应用 8](#_Toc105524052)

[4.1 结构风格 8](#_Toc105524053)

[4.2 核心算法与数据结构 8](#_Toc105524054)

[4.3 开发环境 8](#_Toc105524055)

[五、开发总结与未来工作 8](#_Toc105524056)

[5.1 开发总结 8](#_Toc105524057)

[5.2 未来工作 9](#_Toc105524058)

## 一、项目陈述

### 1.1 问题陈述

长期来说，在乡村建设过程中，优质人才资源外流、人才结构失衡等问题日益凸显，乡村新业态的蓬勃发展、乡村集体产业的转型增值，迫切需要具有现代管理理念的经营型人才的加入，也需要乡村治理者具有更高的经营能力，职业经理人、各类专业经营人才的缺乏。希望设计些小程序，及时对接家乡人才需求，传递给在外的学子，或者想返乡的创业的人，以及当地想找工作的人。

短期来说，县域的农产品加工厂大多受季节影响，用短期工需求较多，农民朋友们也会有农忙与农闲的时候，双方都需要一个工具帮助对接起来，既解决农业企业的短期用工问题，又能帮助当地村民增收。

因此，我们结合农村招聘中需要兼顾地域与当地农耕需要的特点，开发了一款针对于短期用工需求进行招聘的微信小程序：“呼叫邻里”，可以帮助需要招收短期工的雇主与需要兼顾农耕的乡亲们对接需求，为农闲时的农民创造更多的创收机会，为需要在农村开展业务的雇主快速找到雇员。

### 1.2 利益相关者需求分析

农村招聘程序的核心思想是对不同的工作人员的核心任务进行明晰。

1. 作为应聘者：我们希望其能够正常登录小程序，并且实时的编辑发布自己的简历，查看相关岗位信息并投递简历，可以及时的跟招聘人员进行交流，最好可以有一些技能培训和简历制作指导。
2. 作为招聘者：我们希望可以正常登录小程序，根据用工需求发布职位，实时查看收到的简历情况，对简历进行筛选，联系应聘者进行交流。
3. 作为平台客服：我们希望能够在工作时间接收到平台用户的反馈内容，并能够及时回复。

### 1.3 需要实现的功能梳理

**主要功能：**

1. 支持帐号登录体系，帐号登录、登出等；
2. 招工信息查询，可选择职位、薪酬待遇、行业类别、工作地点等；
3. 招工需求发布，可区分个人或企业身份，并能编辑相关用工信息；
4. 求职信息发布，个人可编辑相关求职信息发布；
5. 简历投递，可向心仪的招工企业或个人雇主投递简历；
6. 简单管理端，企业或个人雇主可对收到的求职信息进行处理，如简历查看，对信息打标记等。

**选做功能：**

1. 简历辅导，提供相关辅导资源，如视频、图文教程等；
2. 职业培训，提供相关培训资源，如视频、图文教程等；
3. 法律咨询，简单的劳动法知识宣传，如视频、图文等；
4. 简单客服咨询，可与招工企业或个人雇主进行客服咨询，支持文本、图片、文件传输（可适当使用官方或外部组件实现该能力）；
5. 支持订阅消息提醒。

（注：由于时间原因，可能无法将所有的功能都实现，因而优先实现主要功能）

## 二、系统需求

### 2.1 功能性需求列举

**2.1.1 注册登录**

1. 注册功能（个人注册、企业注册）
2. 登录功能（个人登录、企业登录）

**2.1.2 个人登录**

1. 招工信息查询，可选择职位、薪酬待遇、行业类别、工作地点等；
2. 招工需求发布，并能编辑相关用工信息；
3. 求职信息发布，个人可编辑相关求职信息发布；
4. 简历投递，可向心仪的招工企业或个人雇主投递简历；
5. 求职咨询，联系招工需求发布人员进行沟通；
6. 联系客服。

**2.1.3 个人页（个人登录）**

1. 简历辅导，提供相关辅导资源，如视频、图文教程等；
2. 职业培训，提供相关培训资源，如视频、图文教程等；
3. 法律咨询，简单的劳动法知识宣传，如视频、图文等；
4. 我的简历，查看和编辑已有简历；
5. 我的发布，查看和修改自己发布的招工情况，包括应聘人数、查看收到的简历、对简历进行标记等；
6. 我的应聘，查看自己投递简历的岗位情况。

**2.1.4 企业登录**

1. 招工需求发布，并能编辑相关用工信息；
2. 联系应聘者进行线上交流；
3. 联系客服。

**2.1.5 个人页（企业登录）**

1. 企业简介编辑；
2. 我的发布，查看和修改自己发布的招工情况，包括应聘人数、查看收到的简历、对简历进行标记等；

**2.1.6 收藏页**

1. 收藏岗位；
2. 收藏公司。

### 2.2 非功能性需求列举

1. 使用NodeJS技术编写后端代码，结合微信小程序云托管提供的部署服务，快速实现功能。
2. 数据库使用MySQL，可以与云存储适配。
3. 应用微信小程序开发工具，与云托管提供的前端API，编写前端代码。

### 2.3 屏幕显示需求

1. 注册登录功能界面；
2. 不同登陆者的主界面；
3. 个人页面（企业、个人）；
4. 调用微信提供的客服API实现客服功能聊天页面；（未完成）
5. 简历编写页面；
6. 招工信息页面；
7. 应聘信息页面；
8. 其他页面（法律自诩、技能培训等）（未完成）

## 三、系统分析与设计

### 3.1 利益相关者

应聘者

招聘者

客服

### 3.2 样例图分析

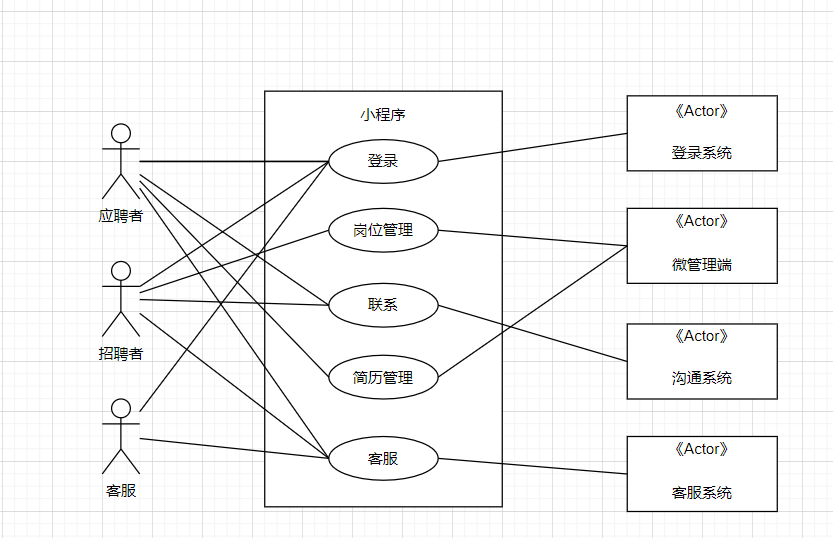


图1 小程序样例图

样例图如图所示，招聘者会根据登录的方式（个人、企业），在登陆管理中被赋予不同的功能。应聘者肯定都为个人，因此简历管理服务仅与应聘者相关，而岗位管理服务中会包括简历筛选、应聘人数查看、建立标记等；由于应聘者跟招聘者需要进行沟通，因此需要有联系服务，且二者都能使用；客服需要登录，因此也与登录服务有联系，三者都与客服服务有联系。（在具体实现上，因为客服系统可以直接调用微信小程序开发提供的API实现，自动进行客服人员添加并完成登陆绑定，因此不需要特地做一个客服登录分支。）

### 3.3 时序图分析

一、是登录时序图：



图2 登录时序图

由于微信开放小程序登录API，所以本处直接用NodeJS编写后端代码调用微信小程序开发提供的API进行实现，因此官网上的时序图符合本文所需。

说明：

1. 调用 [wx.login()](https://developers.weixin.qq.com/miniprogram/dev/api/open-api/login/wx.login.html) 获取 **临时登录凭证code** ，并回传到开发者服务器。
2. 调用 [auth.code2Session](https://developers.weixin.qq.com/miniprogram/dev/api-backend/open-api/login/auth.code2Session.html) 接口，换取 **用户唯一标识 OpenID** 、 用户在微信开放平台帐号下的**唯一标识UnionID**（若当前小程序已绑定到微信开放平台帐号） 和 **会话密钥 session\_key**。
3. 完成登录。

二、查看岗位信息与发布岗位功能时序图

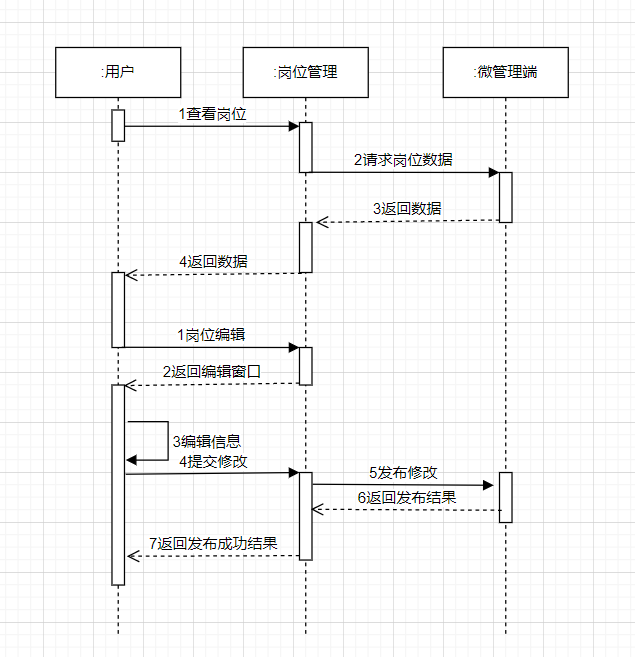


图3 岗位管理时序图

由于岗位的查看与发布都通过岗位管理服务实现，因此在同一张时序图上进行呈现。三个对象分别代表了用户，服务端和后端。用户发起命令，由服务端进行相应，然后对后端进行操作。

后端链接着数据库，通过数据库操作，按照用户命令进行数据的增删改查，返回到服务端，服务端对数据进行处理后返回给用户。用户对数据的修改，停留在前端，修改后的结果在提交后才反馈给服务端，服务端对数据做处理后，提交给微管理端对数据库进行统一操作。

## 四、系统具体实现

### 4.1 开发环境

后端：本地调试使用Node.js版本为v18.2.0；云端Node版本未知，云端MySQL 5.7

前端：微信小程序开发工具，调试基础库版本2.24.5。

操作系统：Windows 11 家庭版

使用非LTS（长期支持版本）的Node.js的原因是，Node.js v18更新了原生的fetch方法，开发时方便在Node环境中调试网络接口。除此之外本程序的最终版本没有使用到最新Node的特性，因此后端服务使用低一些的Node版本应该也可以运行。

微信小程序运行时的基础库版本取决于客户端的微信版本，因此过于老旧的微信版本可能无法运行本程序。请确保运行时的基础库版本大于2.21.2。

### 4.2 整体架构

（图4.1 系统整体架构图）

本程序前端使用微信小程序技术实现，后端采用云托管技术托管了一个Node.js服务，属于传统的前后端分离架构。前端主要负责界面的渲染与呈现、界面与用户交互的逻辑。后端负责处理前端发送的网络请求，根据请求内容进行相应的运算和数据库操作。

### 4.3 前端技术

本程序的前端采用了微信小程序开发技术，主要使用wxml、wxss和JavaScript语言编写而成。另外，微信小程序还提供了一些可以影响程序编译的JSON配置文件，所以也用到了JSON语法。

**4.3.1 依赖管理**

微信小程序的依赖安装与传统使用Node构建的前端项目或基于HTML的前端项目有较大不同。本程序使用了两种方式管理依赖，一是直接导入，通过wxss文件的import语法和wxml的component语法，就可以直接将样式或组件依赖引入。本程序使用这种方法引入了colorUI组件库。这种引入方法的缺点是比较麻烦，需要手动添加引入语句。

此外，程序还使用了构建npm的方式引入依赖。方法是先使用npm init生成package.json，再使用npm安装依赖，最后使用微信小程序开发者工具自带的构建npm功能将安装的node\_modules编译成可以被小程序引入的形式。其默认文件夹名为miniprogram\_npm。本程序使用这种方法引入了weUI组件库。这种方法的缺点是，只有部分npm包提供了对微信小程序编译的支持，因此能够引入的包数量相对有限。

**4.3.2 微信小程序开发框架**

相比于传统前端开发框架，如Vue、React，微信小程序同样提供了组件化、模板语法、条件渲染、列表渲染、动态数据绑定等一系列功能，因此，有前端开发经验的开发者学习微信小程序开发会比较快。本程序在页面和组件中大量使用了这些功能。

此外，本程序还用到了微信小程序开发框架提供的一些特色功能，比如tabbar的配置、自带的UI组件、自带的网络请求封装（CallContainer）、自带的前端路由、与用户设备直接交互的API（如调起用户电话呼叫）等等。

### 4.4 后端技术

由于本小组为两人开发，因此采用前后端分离的云托管模式构建后端相比直接编写云函数后端更易分工，所以我们采用了云托管+Node.js服务的方式创建了小程序的后端。

在服务部署上，小程序使用的是微信提供的云托管服务。云托管服务可以自动关联开发人员的Github代码仓库，在本地编写完成相关代码之后，使用git命令将代码push到仓库中，可以直接触发流水线，进行后端服务的更新。经过研究，微信云托管应该是采用Docker技术，预先编写Dockerfile来实现这一自动化流程的。云托管赋予了项目CI/CD的能力，大幅提高了开发效率。同时，它还可以在一定范围内自动扩缩容Docker环境，来应对高负载场景，实现按量收费。

在后端代码编写上，本程序使用了Koa框架，原因其一是云托管有现成的可供修改的Koa后端模板，其二是Koa框架支持ES6 async await语法，可以提供一种更清晰的编程风格，便于后端代码的组织。另外，Koa框架的生态也比较完备，例如，本项目就使用了Koa-router依赖进行后端路由管理。

### 4.5 数据结构和数据库

（图4.2 数据库表关系图）

由于云托管默认只支持云端MySQL作为数据库，因此本程序使用它存储数据。通过使用Sequelize ORM（Object Relational Mapping，对象关系映射），将关系型数据库的数据表抽象成模型（类），数据表中的记录抽象成实例（类的实例），大幅简化了后端代码中的数据库操作，避免了SQL语句的书写。

本程序的数据模型中，使用Sequelize 的关联功能定义了多个关联关系，其中包括一对一关系、一对多关系和多对多关系。Sequelize 通过数据库中的外键记录实现一对一关系和一对多关系的保存。而对于多对多关系，Sequelize 则是使用额外一张表存储数据之间的关联，并使用每一行记录的值作联合主键，最大化节约了空间。

### 4.6 分模块具体实现

**4.6.1 登录和注册**

在用户第一次打开小程序时，前端代码会自动使用微信自带的封装方法CallContainer请求login接口。这个CallContainer方法会给网络请求附加一些额外信息，例如用户的wx\_openid。而每个用户的wx\_openid是唯一的，通过这个封装方法调用任何接口，后端都可以判断用户的身份。请求login接口后，如果后端发现是新用户，就会将用户的wx\_openid录入，并设置默认的灰色头像和“微信用户”昵称。

**4.6.2 表单页面**

表单数据的提交

## 五、开发总结与未来工作

### 5.1 开发总结

微信小程序的开发更偏向于前端开发，如果选择了云开发服务，可以将所有的函数通过上传部署到云端，所有云函数可以使用微信开发工具提供的API实现，比较方便。如果需要自己定义后端功能，可以选择云托管服务，将自己的后端代码托管到微信提供的云端服务器，减少对服务器的维护操作。

本次使用云托管服务，注意云托管服务文档中明确的调用API是动态IP地址，对于外网是不提供访问链接服务的，因此应该使用微信提供的对应API对该功能进行替代。

### 5.2 未来工作

继续完善小程序的界面与功能。