

X X X X 大 学

# 毕 业 论 文（设计）

论文题目 校园餐饮评价与美食推荐的社区 APP 的设计与实现

姓 名 x x x 学 号 x x x

院 系 信息与计算机学院 专 业 计算机科学与技术

指导教师 x x 职 称 副教授

中国 · XX

二 0 二 0 年 六 月

XXXX 大学学士学位论文（设计）开题报告

课题名称	校园餐饮评价与美食推荐的社区APP的设计 与实现			课题来源	学生自拟
学生姓名	XXX	专业	计算机科学与技术	学号	XXXXXXX
指导教师姓名	XXX	职称	副教授		
研究内容	1、移动应用的需求分析。 2、后端与移动端的架构设计。 3、Oauth2认证协议（密码模式）的实现。 4、软件工程的实践。 5、解决实际编码带来的问题。				
研究计划	2019/12/1-2020/1/31 调研、预确定软件架构、学习OAuth2认证与安卓设计。 2020/2/1-2020/2/15 搭建移动端与后端架构、划分功能模块。 2020/2/16-2020/3/15 在初步的架构中进行编码，完成认证模块。 2020/3/16-2020/4/15 逐步完成各个模块编码，并进行测试。 2020/4/16-2020/4/30 增加应用细节、撰写毕业设计论文。				
特色与创新	1、立意新颖，为解决吃什么的问题。 2、增加“吃喝聊天室”，拉近人与人之间的距离。 3、应用具有实际应用价值。				
指导教师	同意开题		教研室意	同意开题	
学院意见	同意开题				

# 目录

1 绪论.....	1
1.1 选题背景和意义.....	1
1.2 设计目标.....	2
1.3 论文的研究思路.....	2
1.4 软件开发设计方案.....	2
2 需求分析.....	2
2.1 软硬件需求.....	2
2.2 功能需求.....	3
2.3 用户需求.....	4
3 概要设计.....	4
3.1 系统架构设计.....	4
3.1.1 后端架构.....	4
3.1.2 移动端架构.....	6
3.2 系统功能总体设计.....	7
3.3 数据库 E-R 关系 .....	8
3.4 界面设计.....	9
4 详细设计.....	12
4.1 数据库详细设计文档.....	12
4.2 移动端与后端通信文档.....	16
4.2.1 用户模块.....	16
4.2.2 帖子模块.....	17
4.2.3 评论模块.....	18
4.3 流程图.....	19
5 编码与测试.....	20
5.1 开发工具简介.....	20
5.2 开发界面总览.....	21
5.3 测试.....	22
5.4 改进与维护.....	22
6 打包上线.....	22
7 结束语.....	23

参考文献.....	24
英文文摘.....	25
致谢.....	25

# 校园餐饮评价与美食推荐的社区 APP 的设计与实现

学生：xxx，指导教师：xxx

(XXXX 大学 信息与计算机学院 合肥 230036)

**摘要：**为解决一到饭点，不知吃什么的问题，以及校园餐饮缺乏点评的机制，利用目前流行的移动端和后端技术打造一个关于吃喝的社区应用，在这里用户可以发送信息，也可以浏览信息，通过大量的汇聚信息，用户可以参考他人，或与他人共同交流。依托目前移动设备和网络的普及，应用可以广泛方便地安装到设备上，以快速增加信息总量。本文介绍了基于安卓的校园餐饮评价与美食推荐的社区应用，给出了该应用从设计到实现的总体流程。旨在设计并实现一个解决“吃什么”问题和探索校园美食的可以发帖、评论、消息和在线聊天的社区应用。

**关键词：**安卓，社区，吃什么，设计，实现

## 1 绪论

### 1.1 选题背景和意义

2010 年以来，得益于基础设施建设和移动终端低廉的价格，移动互联网迅猛发展，国内移动终端的持有量早已超越传统 PC。移动互联网的迅猛发展极大地改变了人们的生活方式，各种各样的移动应用出现在各大应用市场，不同的应用聚焦不同的用户痛点。

本校有六个食堂，拥有大量的窗口，但是校园餐饮却缺乏评价机制，同时同学们就餐有时也并不了解窗口质量。如果有一个评价机制，同学们就可以对某些窗口提前了解，并且通过这个机制，同学们可以探索校园美食。

民以食为天，不同于往日物质匮乏的时代，当今人们面对各种各样的食物，一到饭点却不知吃什么，你问我我们你，陷入了“吃什么”这样一个问题怪圈。

针对以上两个问题，利用现在流行的后端技术与移动端技术打造一个社区应用，并加以特色功能，解决用户“吃什么”的问题。

## 1.2 设计目标

本应用是一个 C/S 架构，采用移动端做为终端有利于用户快速使用，随时随地参与社区发帖与讨论。用户可以发布校园餐饮评价和美食推荐两类帖子的社区应用，作为一个社区应用，用户可以评价回复，同时还有在线聊天室、一周热榜和美食图库等功能。在完成功能的基础上，更要注重细节，比如 UI 交互、用户体验感。本人负责该应用的全部设计与开发。

## 1.3 论文的研究思路

第一章简单说明设计的背景与思路，第二章给出后端和移动端的架构以及两端的交互方式，第三章给出各个功能模块的原型，第四章给出根据原型进行实际开发的流程，第五章介绍打包上线流程，第六章进行总结。

## 1.4 软件开发设计方案

一个完整的社区应用需要后端与移动端，为了快速实现功能，应该选择主流技术，利用其优势。当前后端 Java 技术栈占据主流，十几年的发展，Java 拥有庞大的生态。移动端由于安卓的开发与齐全的开发文档，市场比例占据绝对优势，采用安卓有利于开发与用户安装。

整体上利用 Spring 家族框架搭建后端，安卓利用谷歌推荐的 Jetpack 架构，以快速搭建开发架构。在此基础上进行功能模块的表结构和移动端 ui 设计，然后进行开发与测试。然后进行细节上的调整与性能优化，最终将后端代码进行集成打包部署在服务器，安卓利用 Android Studio 进行打包提供给用户。

# 2 需求分析

## 2.1 软硬件需求

- (1)、移动端选择安卓，运行在安卓 8.0 以上即 Android SDK  $\geq 28$ 。
- (2)、数据库：MySQL。
- (3)、开发环境：OpenJDK 11、ArchLinux、ManjaroLinux。
- (4)、开发工具：Android Studio、IDEA、DataGrip。

(5)、硬件需求：matebook13、旧笔记本。

## 2.2 功能需求

### (1)、用户模块

作为一个社区类应用，用户是整个应用的基石，用户在注册登录后有一个单独的个人空间，保存着自己的发帖记录、关注和粉丝信息、以及头像、昵称等可修改信息。用户登录凭证失效时，应提醒用户重新登录。

### (2)、发帖

用户可以发校园餐饮评价与美食推荐两类帖子，同时这两类帖子又可进行细分。用户在发帖时，标题和内容必选，并且可以进行评分或者推荐打分，或者选择附带图片。每一个帖子都有评论和点赞信息，用户评论和点赞后，帖子应该动态显示。

### (3)、评论与消息

每一个帖子都有对应的评论模块，用户可以评论帖子与发帖人互动，也可以回复评论人。当评论出现变化时，应动态的反应在页面上。同时，应在应用的消息模块中反映新的评论，或者是其他人的回复。

### (4)、点赞

用户可以给帖子和评论点赞，在点赞逻辑中，要进行区分。同时，应在消息中反映新的点赞信息。

### (5)、关注

关注是用户之间的关系，关注的同时带来粉丝关系，关注的用户发帖，应在动态中按照时间先后顺序排列。

### (6)、吃喝聊天室

吃喝聊天室是一个即时的在线聊天室，用户可以在这里发送文字信息，同时用户的 ID、昵称、头像也应传递。

### (7)、一周热榜

在一周时间内，根据点赞数将帖子中的前 10 进行罗列，提供给用户。数据应是动态的在后端处理数据库数据时进行条件选择。

### (8)、美食图库

在美食推荐贴中，会附带一些图片，美食图库作为一个收集美食推荐贴中的美食

图片而存在。

## 2.3 用户需求

- (1)、界面设计合理，人性化，容易让人理解。
- (2)、页面加载快速。
- (3)、应用不能经常闪退。
- (4)、运行要流畅，不能出现卡顿现象。

## 3 概要设计

### 3.1 系统架构设计

#### 3.1.1 后端架构

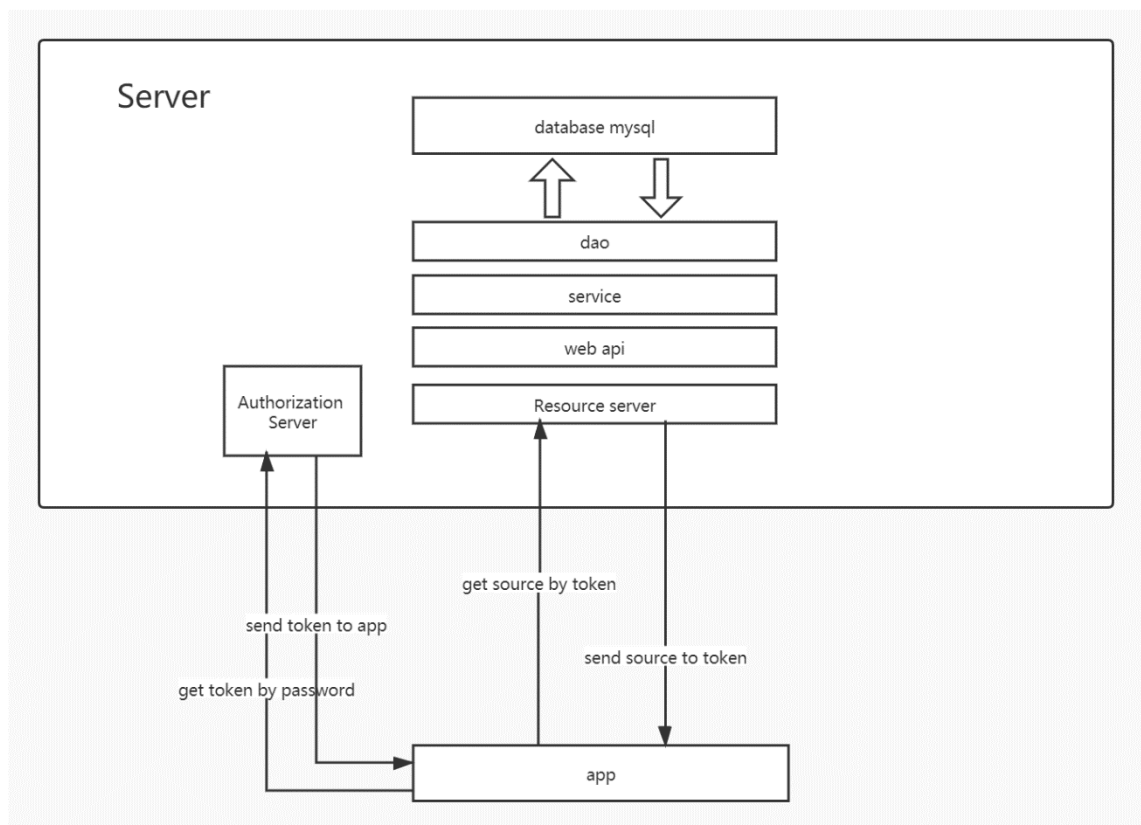


图 3-1 后端架构图

#### 1、架构概述

持久层使用 MySQL 作为数据库，对数据进行存储。后端数据处理分为 dao、service、web api 三层，在 dao 层访问数据库，在 service 层对数据进行处理，在



web api 层将 service 处理的数据以 JSON 的方式发送出去, 接口风格采用 restful 风格。后端对移动端的认证采用 OAuth2 协议中的密码模式, 即移动端使用标识和凭证向后端授权服务器获取 token(存储用户认证信息的一段字符串), 移动端获取 token 后, 凭 token 向资源服务器获取资源。

## 2、所用技术

### (1)、OAuth 2.0 协议:

OAuth 2.0 是一个针对于授权的行业标准协议, OAuth 2.0 侧重于客户端开发人员的简单性, 同时为 web 应用程序、桌面应用程序、移动电话和客厅设备提供特定的授权流。标准的 OAuth 2.0 协议拥有 4 种模式, 我在该设计中采用 OAuth 2.0 协议中的密码模式。

### (2)、Spring Boot:

Spring Boot 使创建独立的、生产级的基于 Spring 的应用程序变得很容易, 您可以“直接运行”这些应用程序。相较于以往的 SSM 模式, 整合进类似 Tomcat 的应用服务器中, Spring Boot 利用其独特的“约定优于配置”思想, 使得 Spring Boot 创建一个后端应用更加迅速, 摆脱繁琐的配置。

### (3)、Spring Security OAuth2:

Spring Security OAuth 为使用标准 Spring 和 Spring 安全编程模型以及配置习惯用法的 OAuth (1a) 和 OAuth2 来使用 Spring 安全性提供了支持。当然, 该框架与标准的 OAuth2 不完全一致, 使用应以实际情况。

### (4)、Spring Data JPA:

作为更大的 Spring 数据家族的一部分, 使得轻松实现基于 JPA 的存储库变得很容易。此模块处理对基于 JPA 的数据访问层的增强支持。它使得构建使用数据访问技术的 spring 应用程序变得更加容易。

作为一个更加抽象的 ORM 框架, Spring Data JPA 拥有独特的优势即单表查询, 简单的代码可以替代极其繁琐的步骤, 领域模型设计思想的实现, 使得 Spring Data JPA 可以无视数据库的类型。但是也不无缺点, 关联查询非常麻烦。

### (5)、MySQL:

开放源码的关系数据库管理系统。

### (6)、Maven:

是一个软件 (特别是 Java 软件) 项目管理及自动构建工具, 由 Apache 软件基金

会所提供。基于项目对象模型（缩写：POM）概念，Maven 利用一个中央信息片段能管理一个项目的构建、报告和文档等步骤。

### 3.1.2 移动端架构

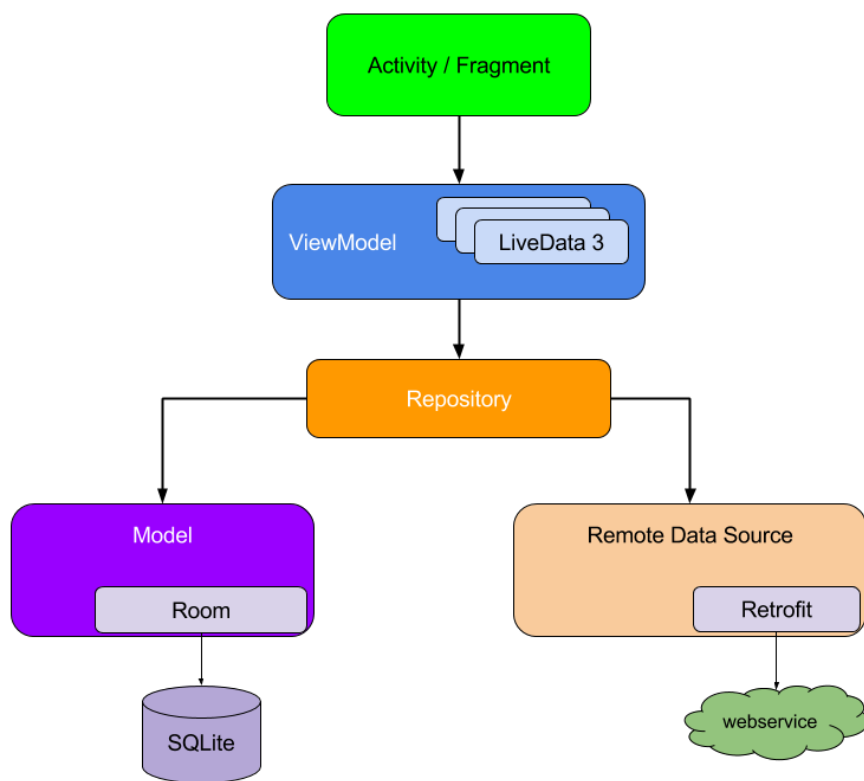


图 3-2 移动端架构图

#### 1、架构概述

采用单 Activity 多 fragment 模式，利用 Navigation 进行页面跳转。实际即底部导航栏，利用安卓提供的 UI 组件进行页面布局 and 页面设计。在 ViewModel 中进行数据处理，供视图层进行调用，将业务与视图剥离。利用 Retrofit 向服务器异步访问数据，访问到数据后进行处理，然后在主线程渲染到页面上。充分利用谷歌提供的 MVVM 实现，简便开发，专注于业务。

#### 2、所用技术

##### (1)、Jetpack

Jetpack 是一套库、工具和指南，可帮助开发者更轻松地编写优质应用。这些组件可帮助您遵循最佳做法、让您摆脱编写样板代码的工作并简化复杂任务，以便您将

精力集中放在所需的代码上。

(2)、MVVM

MVVM 是一种软件架构模式，MVVM 就是将其中的 View 的状态和行为抽象化，让我们将视图 UI 和业务逻辑分开。当然这些事 ViewModel 已经帮我们做了，它可以取出 Model 的数据同时帮忙处理 View 中由于需要展示内容而涉及的业务逻辑。

(3)、Retrofit

一个安卓和 Java 平台类型安全的 HTTP 客户端。与主流的 Java 框架相同，Retrofit 也使用了接口和注解作为其基础方式，使得 HTTP 访问变得异常方便。

(4)、网络编程

为实现吃喝聊天室模块，需要 Java 网络编程，在服务器端单独开启一个线程处理客户端吃喝聊天室发送的信息。客户端在接收到服务端信息时，需要将信息重新序列化并最终显示在手机页面上。

3.2 系统功能总体设计

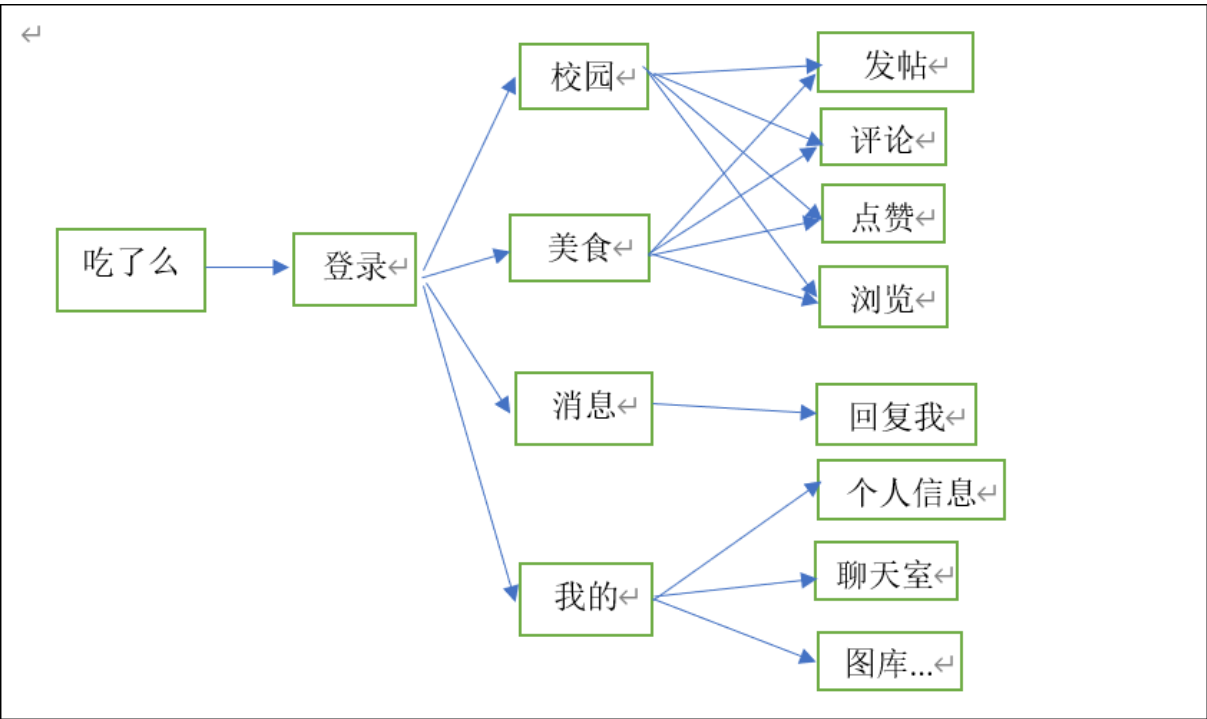


图 3-3 系统功能概况图

### 3.3 数据库 E-R 关系

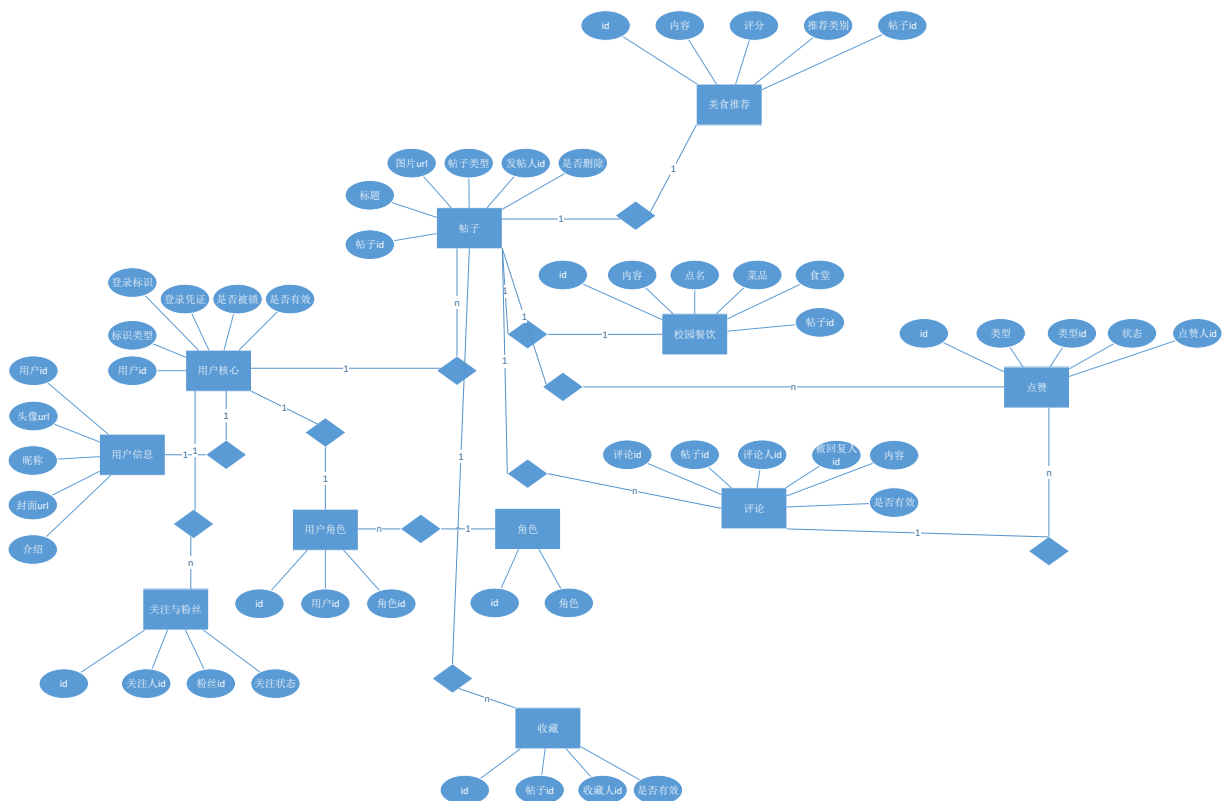


图 3-4 数据库 E-R 关系图

### 3.4 界面设计

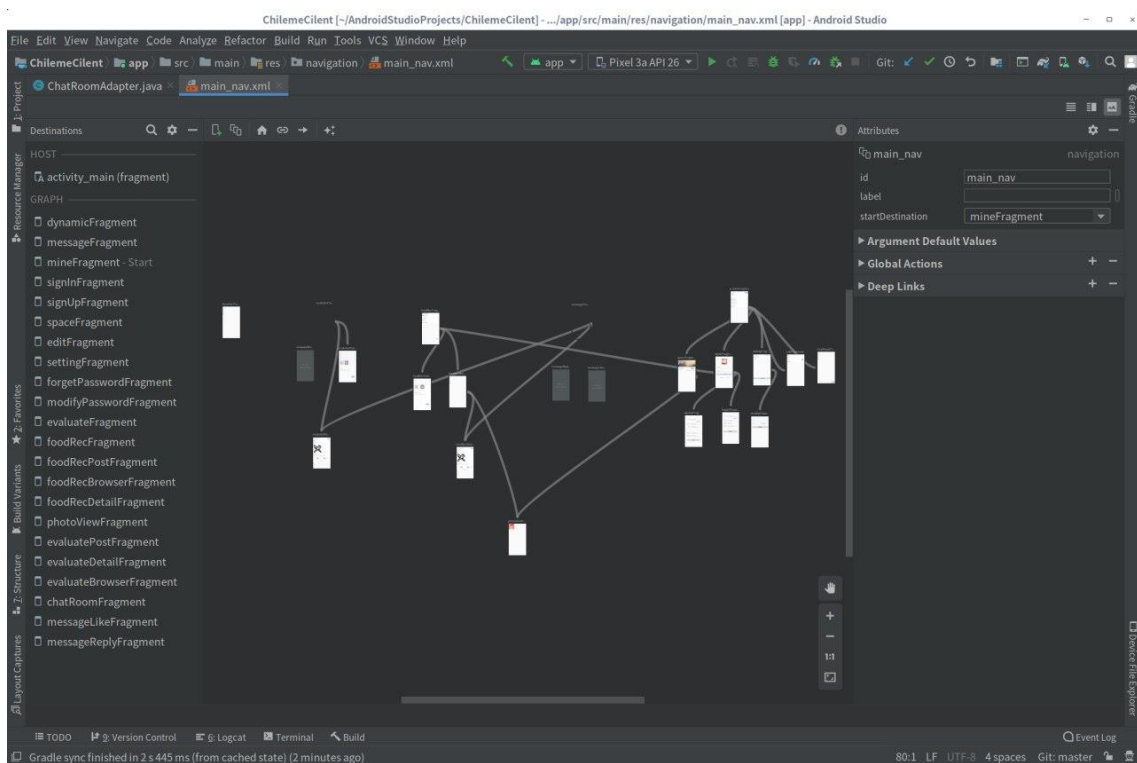


图 3-5 界面设计总览



图 3-6 发帖界面设计



图 3-7 美食推荐界面设计



图 3-8 消息界面设计



图 3-9 我的界面设计

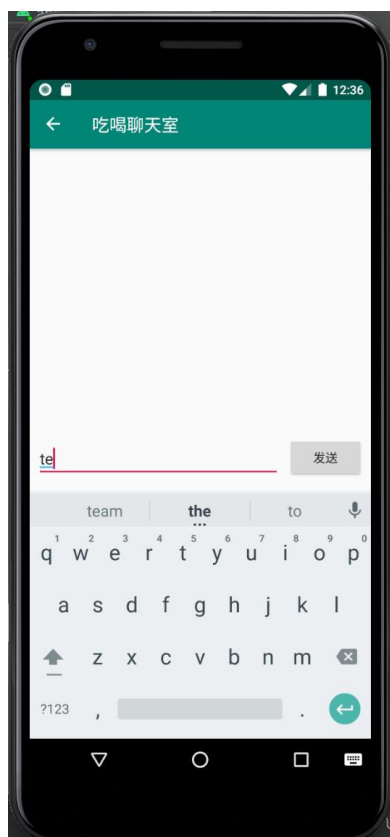


图 3-10 吃喝聊天室界面设计

## 4 详细设计

### 4.1 数据库详细设计文档

表 4-1 用户核心表设计 **user\_core**

字段	属性	备注
user_id	int(11) (主键自增)	用户编号
identity_type	varchar (255)	登录标识类型
identifier	varchar (255)	登录标识
credential	varchar (255)	登录凭证
enabled	tinyint(1)	是否有效
locked	tinyint(1)	是否被锁
security_question	varchar (255)	密保问题
security_answer	varchar (255)	密保答案

表 4-2 用户信息表设计 **user\_info**

字段	属性	备注
id	int(11) (自增主键)	字段编号
user_id	int(11) (外键)	用户 id
avatar_url	varchar(255)	存储的头像链接
nick_name	varchar(255)	昵称
intro	varchar(255)	个人介绍
cover_url	varchar(255)	存储的封面链接

表 4-3 用户角色表 **user\_role**

字段	属性	备注
id	int(11) (自增主键)	字段编号
uid	int(11) (外键)	用户 id



rid	int(11) (外键)	角色 id
-----	--------------	-------

**表 4-4 角色表 role**

字段	属性	备注
id	int(11) (自增主键)	角色 id
name	varchar(255)	角色
name_zh	varchar(255)	中文解释

**表 4-5 帖子表 post**

字段	属性	备注
post_id	int(11) (自增主键)	帖子 id
post_type	tinyint(1)	帖子类型 0 校园 推荐
headline	varchar(11)	帖子标题
user_id	int(11) (外键)	用户 id
image_url	varchar(11)	帖子图片 无则为空
enable	tinyint(1)	是否有效 0 有效 1 无效

**表 4-6 校园餐饮评价表 campus\_evaluate**

字段	属性	备注
id	int(11) (自增主键)	字段编号
hall_id	int(11) (外键)	食堂 id
store_name	varchar(255)	店名
dish	varchar(255)	菜品
content	varchar(255)	评价内容
post_id	int(11) (外键)	帖子 id
score	float	评价分数

**表 4-7 美食推荐表 food\_recommend**

字段	属性	备注
id	int(11)（自增主键）	字段编号
rec_classify	int(11)（外键）	推荐类型
content	varchar(255)	内容
post_id	int(11)（外键）	帖子 id
score	float	推荐指数
location	varchar(255)	位置

**表 4-8 评论回复表 comment**

字段	属性	备注
id	int(11)（自增主键）	字段编号
post_id	int(11)（外键）	帖子 id
from_id	int(11)（外键）	评论人或回复人 id
to_id	int(11)（外键）	被回复人 页面要+@
content	varchar(255)	评论或回复内容
status	tinyint(1)	状态 0 有效 1 无效

**表 4-9 点赞表 like**

字段	属性	备注
id	int(11)（自增主键）	字段编号
type	tinyint(1)	点赞类型 0 帖子 1 评论或回复
type_id	int(11)	帖子 id 或回复 id
user_id	int(11)（外键）	点赞用户 id
status	tinyint(1)	状态 0 有效 1 无效

**表 4-10 收藏表 my\_favorite**

字段	属性	备注
id	int(11)（自增主键）	字段编号

post_id	int(11) (外键)	帖子 id
user_id	int(11) (外键)	收藏用户 id
status	tinyint(1)	状态 0 有效 1 无效

**表 4-11 关注粉丝表 attention\_fans**

字段	属性	备注
id	int(11) (自增主键)	字段编号
attention_id	int(11) (外键)	被关注用户 id
fans_id	int(11) (外键)	粉丝 id
status	tinyint(1)	状态 0 有效 1 无效

**表 4-12 食堂分类表 hall\_classify**

字段	属性	备注
hall_id	int(11) (自增主键)	食堂 id
hall_name	varchar(255)	食堂名称

**表 4-13 推荐分类表 rec\_classify**

字段	属性	备注
hall_id	int(11) (自增主键)	食堂 id
hall_name	varchar(255)	食堂名称

**表 4-14 浏览历史表 browser\_history**

字段	属性	备注
id	int(11) (自增主键)	字段编号
post_id	int(11) (外键)	帖子 id
user_id	int(11) (外键)	浏览用户 id
status	tinyint(1)	状态 0 有效 1 无效

## 4.2 移动端与后端通信文档

说明：返回格式为 timestamp、code、message 和 data 四个属性封装的 json 字符串。

timestamp：时间戳；

code：内部结果码；

message：结果信息，专门编写一个枚举类存放 code 和 message 实例；

data：数据，要传递的实际数据。

实例：（json）

```
{
  "timestamp": "2020-04-11 12:16:13",
  "code": 1,
  "message": "成功! ",
  "data": {
    "userId": 1,
    "nickName": "nikeName",
    "intro": null,
    "avatarUrl": "image/a3a5c2df-86fd-4f1a-85f4-64187a87c2f0.jpg",
    "coverUrl": null,
    "attentionNumber": 0,
    "fansNumber": 0,
    "postNumber": 8,
    "historyNumber": 0,
    "myFavoriteNumber": 4
  }
}
```

### 4.2.1 用户模块

（1）、根据登录标识查找用户 id

接口地址：/api/v1/user/{identifier}

方式和返回：GET JSON

请求参数：

表 4-15 登录请求参数表

名称	类型	说明
identifier	String	登录标识

#### (2)、注册接口 创建用户

接口地址: /api/v1/user

方式和返回: POST JSON

请求参数:

表 4-16 注册请求参数表

名称	类型	说明
identityType	String	登录类型 默认 phone
identifier	String	登录标识
credential	String	登录凭证
securityQuestion	String	密保问题
securityAnswer	String	密保答案

#### (3)、返回用户信息 展示在我的页面上

接口地址: /api/v1/mine/{usrId}

方式和返回: POST JSON

请求参数:

表 4-17 获取信息请求参数表

名称	类型	说明
userId	String	用户 id

### 4.2.2 帖子模块

#### (1)、发帖 接口

接口地址: /api/v1/evaluate

方式和返回: POST JSON

请求参数:

表 4-18 发帖请求参数表

名称	类型	说明
headline	String	标题
storeName	String	店名
dish	String	菜品
content	String	点评
hallId	int	食堂
evaluateScore	float	打分
postUserId	int	发帖人 id

(2)、头像 与 封面 图片的上传与下载接口

接口地址: /file/upload

方式和返回: POST JSON

请求参数:

表 4-19 文件上传请求参数表

名称	类型	说明
id	int	上传用户 id
typeCode	int	文件类型

### 4.2.3 评论模块

(1)、根据 帖子 id 和 访问用户 id 获取所有评论

接口地址: /api/v1/comment

方式和返回: GET JSON

请求参数:

表 4-20 获取评论请求参数表

名称	类型	说明
postId	int	帖子 id
userId	int	用户 id

(2)、增加一条评论

接口地址: /api/v1/comment

方式和返回: POST JSON

请求参数:

表 4-21 增加评论请求参数表

名称	类型	说明
postId	int	帖子 id
content	String	内容
fromUid	int	评论或回复人 id
toUid	int	被回复人 id

### 4.3 流程图

#### (1)、登录流程

登录流程在本项目中是一个关键的流程,用户打开应用时,应根据是否存在 token 判断用户是否登录过。如果没有,则提示登录;如果存在 token,则要判断 token 是否临近到期,如果没有临近到期,则可以直接拿 token 和用户 id 去服务器获取信息,如果临近到期,则要提示用户重新登录。用户登录过程中,则要根据用户的登录标识与凭证向服务器获取 token,如果登录凭证正确,则返回 token,否则重新登录。

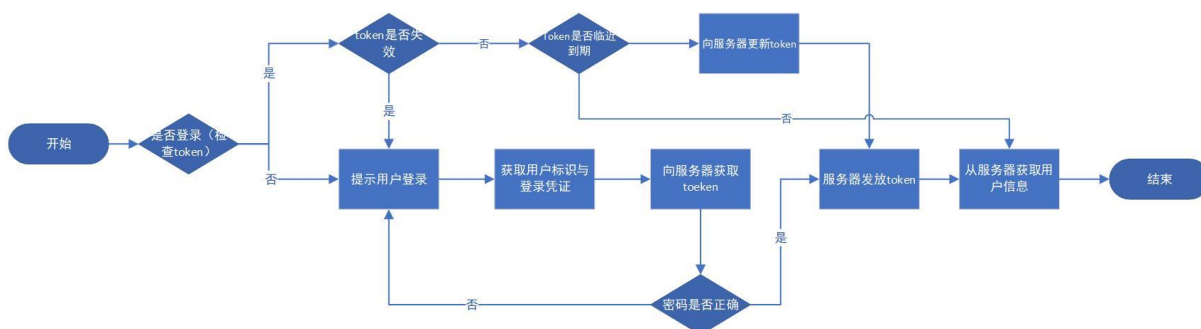


图 4-1 登录流程图

#### (2)、在线聊天室流程

在线聊天室的流程并不复杂,但是实现的时候,需要多线程与网络编程,其中需要注意得一点是在安卓的主线程也就是 UI 线程不能进行耗时操作,比如 IO 与网络。具体流程参见下图。



图 4-2 在线聊天室流程图

### (3)、发帖流程图

发帖流程是应用中比较常用的流程，这个流程在编码中的难点在于图片的上传。具体是在将信息发送到服务器并进行存储后，向客户端返回帖子 id，客户端拿到帖子 id 后再根据是否有图片，选择发送图片，当然，在发送的同时要附带帖子 id。之所以这样设计，在于服务端的文件上传接口与帖子接口进行了拆分，方便后续编码。

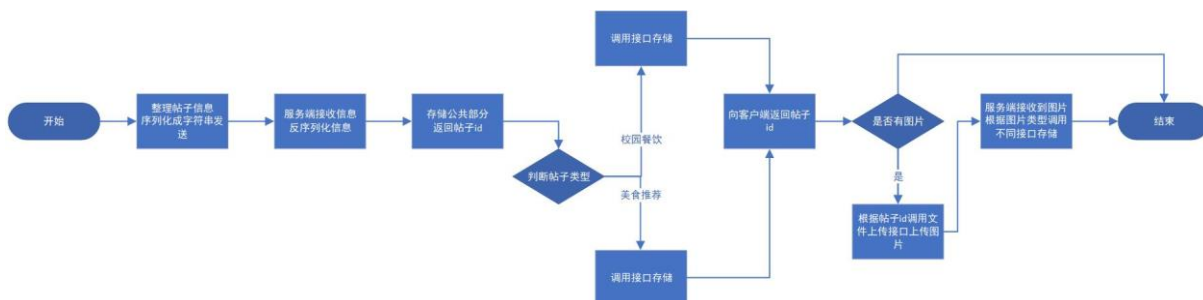


图 4-3 发帖流程图

## 5 编码与测试

### 5.1 开发工具简介

**Android Studio:** 是谷歌推出的一个 Android 集成开发工具，基于 IntelliJ IDEA。Android Studio 提供了集成的 Android 开发工具用于开发和调试。由于谷歌的支持，这个专为安卓开发的集成开发工具非常好用。

**IDEA:** 是一种商业化销售的 Java 集成开发环境（Integrated Development Environment, IDE）工具软件，由 JetBrains 软件公司（前称为 IntelliJ）开发，提供 Apache 2.0 开放式授权的社区版本以及专有软件的商业版本，开发者可选择其所需来下载使用。本次用作后端开发环境。

**Git:** 是一个免费的开源分布式版本控制系统，可以快速高效地处理从小项目到非常大的项目。在本次项目中，使用 Git 将代码进行管理并且同步到远程仓库，为开发带来了便利。



## 5.2 开发界面总览

(1)、后端 仓库地址: <https://github.com/wangqingzh/ChilemeServer>

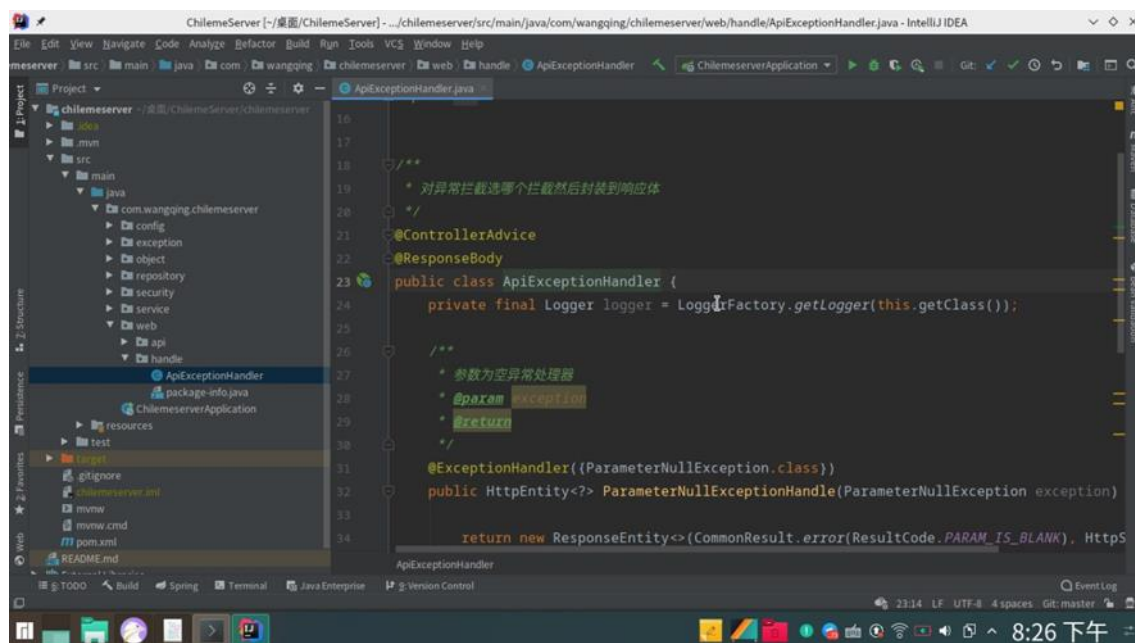


图 5-1 后端开发环境

(2)、移动端 仓库地址: <https://github.com/wangqingzh/ChilemeClient>

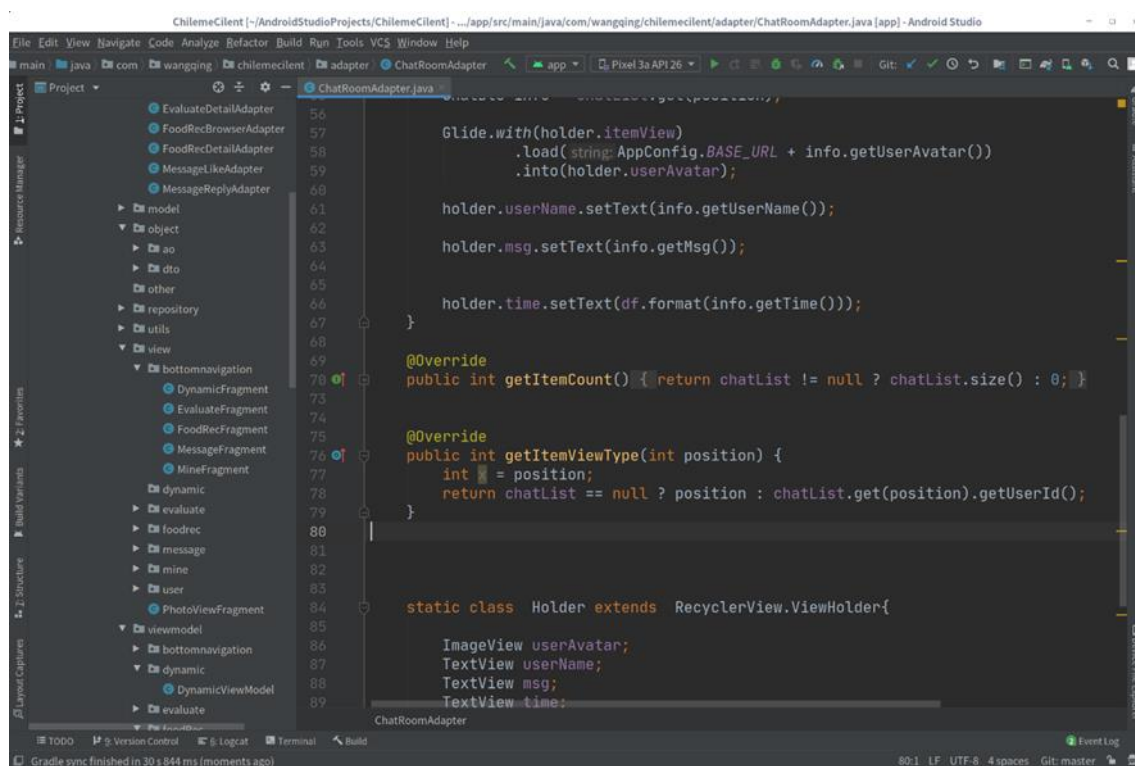


图 5-2 移动端开发环境

### 5.3 测试

由于项目是个人完成，测试其实贯穿于编码的全过程，在此过程中使用到了 Postman 对后端接口进行了数据测试，同时在移动端经常进行打断点进行调试。逐步将一个个模块完成，然后在移动端手机界面上进行功能测试。

### 5.4 改进与维护

应用从一个想法到实现也经历了几个月的时间，最初的设想较现在的实现更为复杂，在实际编码完成过程中，确实遇到了一些困难。由于很多技术都是第一次学习，一边查阅一边实践，移动端的开发也是第一次，中间遇到了一些困难，但是也都一一解决。

目前主要功能已基本实现，但是细节方面仍有改进的可能性，另外功能方面，还可以继续增加一些以丰富应用，在项目的构建方面，代码一开始的分层设计较为合理，但是由于代码量的增加，再加上赶时间，中间没有进行代码重构，代码实际质量并不理想，应再重新进行优化和重构。

没有完美的产品，任何产品都要进行更新和优化，我的应用同样如此，更应在细节方面继续维护。

## 6 打包上线

利用 Maven 对 Spring Boot 项目进行打包，由于 Spring Boot 中继承了内嵌的 tomcat 服务器，上线时不再考虑应用服务器的搭建。移动端利用 Android Studio 对项目进行构建打包，安装即用。在打包时要注意移动端和后端的连接配置，以防打包后无法连接。

服务端部署时，要先部署 MySQL 数据库，将表结构和一部分初始化数据导入数据库，其他相关设置提前根据 Spring Boot 源码中的配置进行设置，然后将打包好的 jar 包运行在后台。移动端安装进设备后，进行连接测试。

## 7 结束语

在本次论文设计中，对应用设计与实现进行了复盘，在一次简略的软件工程流程中，提高了设计与开发能力。由于应用的设计与实现，是边学习边实践，应用的细节难免有一些瑕疵。当然我会继续完善这款应用。从做这个应用的想法诞生到实现，前前后后经历了几个月，从当初不分日夜地查找资料和学习安卓设计，到后来实际操作，不断地编写调试代码，我从中间学到了许许多多技术上的技巧，尤其是数据库的设计、移动端和后端的交互。更重要的，在完成这份毕设的同时，我获得了自信，相信自己学有所用，必定会成为一名优秀的软件工程师。

## 参考文献

- [1] 郭霖. 第一行代码[M]. 北京: 清华大学出版社, 2016. 11
- [2] 杨斯博. 基于安卓平台的手机订餐系统的设计与实现[D]. 吉林大学, 2016.
- [3] 吴蔚. 安卓平台快递派送辅助软件开发[D]. 北京邮电大学, 2013.
- [4] google. Android Developers Reference [EB/OL].<https://developer.android.com/reference>,2020-2-5.
- [5] pivotal. Spring Security Reference[EB/OL].<https://docs.spring.io/spring-security/site/docs/5.3.2.RELEASE/reference/html5/>,2020-03-06.
- [6] Healthline Media; 'Healthline Together' Announces Migraine Healthline App for iOS and Android[J]. Journal of Engineering,2020.

## 英文文摘

# **Design and implementation of community APP for campus catering evaluation and food recommendation**

Lixiang Wang

( School of Information & Computer, Anhui Agricultural University , Hefei 230036)

**Abstract:** In order to solve the problem of not knowing what to eat at the meal point, we make use of the current popular mobile terminal and back-end technology to create a community application about eating and drinking, where users can send information or browse information. Through a large amount of information gathering, users can refer to others or communicate with others together. Depending on the current popularity of mobile devices and networks, applications can be installed on devices extensively and conveniently to rapidly increase the total amount of information. This paper introduces the community application of campus catering evaluation and food recommendation based on android, and gives the overall process from design to implementation of this application. It aims to design and implement a community application that can post, comment, message and chat online to solve the "what to eat" problem.

**Key words:** Android, community, what to eat, design, implement

## 致谢

感谢自己这几年来积累的计算机专业的基础知识和几个月来不断地学习进取。

感谢指导老师在此毕业设计过程中的指导工作，以及四年来所有教过我的老师，让我从一个小白到可以个人设计一个应用。

感谢这个开放的互联网，人人都可以分享知识，使得获取知识变得非常容易，并且有各种各样的实例可以参考。

## 信息与计算机学院(系)毕业论文(设计)指导教师评阅意见

学生姓名 xxx 年级 2016 级计算机科学与技术

评阅意见（论文选题意见，研究内容，实验设计，数据结果分析及论文写作等进行批阅）

评 阅 意 见	<p>该设计采用了目前比较流行的后端技术与移动端技术，来实现一个基于安卓的可用的社区应用。</p> <p>论文中详细地说明了该社区应用的设计背景，以及面向用户的需求。基于此，提供了整个应用从后端技术到移动端技术的架构设计。并且根据需求模块划分，给出了数据库设计和 UI 界面设计，在实现方面该设计提供了代码仓库，项目分层合理。该同学论文结构安排合理，格式符合学士论文要求。</p> <p>该同学动手能力强，学习积极主动，利用所学知识与技术较好地实现了 C/S 应用的开发，达到了导师预定计划要求。</p> <p>建议成绩_____</p> <p>评阅人_____</p>
------------------	---

2020 年 5 月 10 日

## 信息与计算机学院(系)毕业论文(设计)评阅教师意见

学生姓名 xxx 年级 2016 级计算机科学与技术

评卷人姓名 xx 职称 教授 学历 博士

评 阅 意 见	<p>该论文（设计）制作的是校园餐饮评价与美食推荐的社区应用的设计，使用目前比较流行的后端技术与安卓技术，比较完整地实现了一次软件开发的流程。开发的应用具有实际使用意义，立意新颖。</p> <p>该应用的设计到实现，论文描述地较为详尽，作为一个社区应用，该应用基本完成了评论与回复功能模块，还增加了一些新想法比如“吃喝聊天室”。可以看出，该设计与实现工作量大，对该同学今后的工作方向论文撰写都是一次很好的锻炼。从论文整体来看，该同学学习了很多技术，花费了一定的精力，系统中的功能有待进一步改进完善，继续深入研究。</p> <p>该同学论文结构安排合理、清晰，符合学士学位论文的要求。</p> <p>同意指导老师的意见。</p> <p>建议成绩_____</p> <p>评阅人_____</p>
------------------	--

2020 年 5 月 19 日

# XXXX 大学

## 毕业论文（设计）任务书

论文（设计）题目 校园餐饮评价与美食推荐的社区  
APP 的设计与实现

院 系 名 称 信息与计算机学院

专 业 （ 班 级 ） 16 计算机科学与技术-3

学 生 姓 名       X X X      

学 号       X X X X X X      

指 导 教 师       X X      

下发任务书日期

2020 年 3 月 1 日



一、毕业论文（设计）的主要内容
1、搜集资料，对应用设计进行分析。 2、选择合适的技术，设计应用架构。 3、划分功能模块，进行编码实现。 4、进行总结，根据格式撰写论文。
二、毕业论文（设计）的基本要求
1、搜集一定的文献资料。 2、独立设计并完成实例。 3、撰写标准格式的论文。

三、应收集的资料及主要参考文献		
1、OAuth2 授权协议资料。 2、安卓开发相关文档与API。 3、第一行代码。 4、安卓平台快递派送辅助软件开发。		
四、毕业论文（设计）进度计划		
起讫时间	工作内容	备注
2019/12/1-2020/1/31	调研、预确定软件架构、学习OAuth2认证与安卓设计。	
2020/2/1-2020/2/15	搭建移动端与后端架构、划分功能模块。	
2020/2/16-2020/3/15	在初步的架构中进行编码，完成认证模块。	
2020/3/16-2020/4/15	逐步完成各个模块编码，并进行测试。	
2020/4/16-2020/4/30	增加应用细节、撰写毕业设计论文。	