

基于 Android 的校园外卖点餐 APP 的设计与实现*

姚良鹏 邓舒婷**

南宁学院信息工程学院, 南宁 530200

摘要 随着社会发展, 外卖服务在人们生活中扮演着重要角色, 校园外卖市场具有爆发式增长, 竞争日益激烈, 校园外卖点餐 APP 的开发意义重大, 本研究结合 AndroidStudio 和 Spring Boot 框架等技术开发, 用户模块、商家模块、管理员模块以及商家与用户联动模块等, 每个模块都经过详细, 用户界面设计简单直观, 易于操作, 满足用户和商家的日常业务需求。

关键字 外卖点餐, Spring Boot, APP, Android

Design and Implementation of Campus Takeout Ordering APP based on Android

Liangpeng Yao Shuting Deng
School of Information Engineering of Nanning University
Nanning 530200, China;
1979995109@qq.com

Abstract—With the development of society, food delivery services play an important role in people's lives. The campus food delivery market has experienced explosive growth and increasingly fierce competition. The development of campus food delivery ordering apps is of great significance. This research is developed by combining technologies such as Android Studio and Spring Boot, including user modules, merchant modules, administrator modules, and the linkage module between merchants and users. Each module has been detailed. The user interface design is simple and intuitive, easy to operate, and meets the daily business needs of users and merchants.

Keywords—Online food ordering, Spring Boot, APP, Android

1 引言

移动互联网终端的飞速发展, 为餐饮企业提供了有效途径, 提高了运营效率、拓展了业务。目前, 点餐软件越来越收到人们的青睐, 尤其是手机 APP 的推出进一步改善了用户的点餐体验。校园外卖点餐 APP 为大学生提供了方便快捷的外卖点餐服务, 解决了学生因为忙碌或者不想出门而无法及时就餐的问题。该 APP 通过线上点餐的方式, 促进了数字化校园建设, 提高了校园生活的智能化水平。

2 相关技术

本系统专为智能手机、平板等移动设备设计, 基于受欢迎的 Android 移动操作系统, 采用成熟的 AndroidStudio 和 Spring Boot 框架等技术开发, 采用高级语言 Java 进行编程, 解决数据处理、采集、转换等问题, 快速集成第三方组件, 提高了开发效率和代码的可读性。

* **基金资助**: 基金资助: 南宁学院 2024 年教学改革项目 (2024XJG11) 资助。

** **通讯作者**: 邓舒婷 1979995109@qq.com

3 校园外卖点餐 APP 的功能分析

校园外卖点餐 APP 主要有商家、管理员、用户和游客四个用户角色, 每种角色具有不同的权限。功能结构如图 1 所示。

3.1 商家

商家功能是 APP 端界面设计, 主要管理模块包含菜品管理、订单管理、评价查看等。商家是平台提供资源的重要群体, 商家主要功能登录账号、注册账号、添加餐品信息、添加菜品、添加菜品类型、查看订单评论、查询客户订单、处理订单状态、修改个人信息、退出登录等。

3.2 管理员

系统管理员功能是 PC 端界面设计, 主要模块包含店铺审核、菜品信息管理、会员管理等。管理员在平台拥有查看和管理的权限, 能够访问和控制平台的全部数据信息。管理员可以执行查询用户和商家的个人信息、管理用户和商家的个人信息、审核商家上传菜品等。

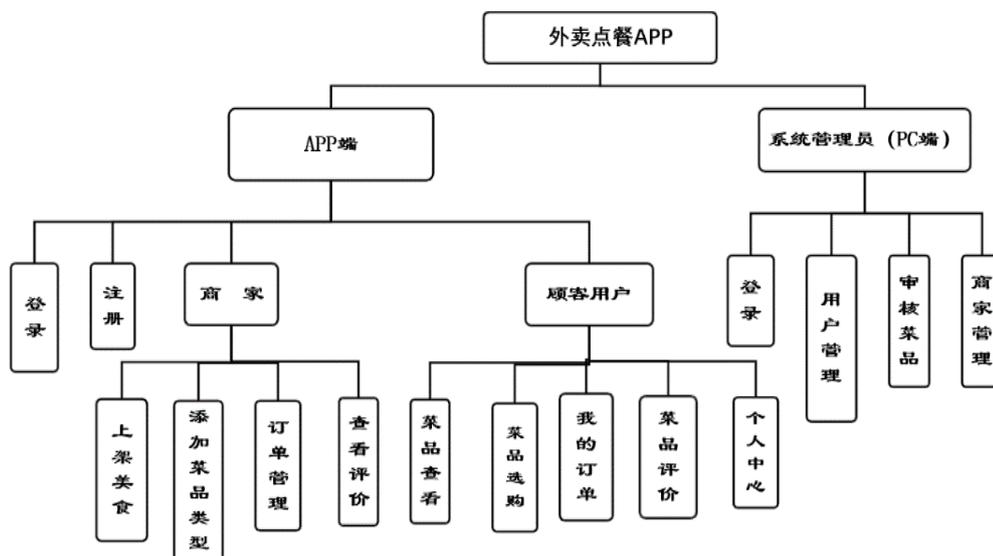


图 1 功能结构图

3.3 用户

顾客用户界面使用的是APP端，主要功能包括查看店铺菜单、下单、评价等。外卖点餐APP为用户提供了丰富且易于操作的功能选项。用户可以进行商品搜索、商品选择、商品购买、查看个人信息、管理收货地址、查看订单情况、收藏喜爱的商品、管理购物车、访问个人中心，根据自己的喜好收藏店铺和菜品。在订单模块中，可以查看和取消未备餐的订单。

4 校园点餐APP核心功能的实现

4.1 注册功能

注册功能分为商家注册和用户注册，注册需要信息验

证、注册请求发送、注册成功和失败处理等。如果系统发现用户名未注册且输入信息符合规范，提交注册信息；商家注册需要系统管理员审核才能成为校园点餐APP的商家。如果发现用户名已存在或输入信息有误，则注册失败并显示错误消息，如图2注册界面图所示。

4.2 登录

商家和用户使用帐号和密码进行登录，用户的密码和登录角色匹配，就能进入对应的后台页面。如果密码与所使用的账户不匹配，登录失败，启动错误提示。登录界面图如图2所示。



图 2 登录界面图



图 3 商家添加美食图



图 4 景添加菜品类型图

4.3 商家模块

(1) 上架美食

商家上架美食信息模块主要包括美食名称、美食价格、选择美食类型、上传美食图片等内容。商家添加美食图如图 3 所示。

商家点击提交按钮时,调用 createTopicPost() 方法提交数据到服务器,并且将用户输入的美食信息封装成 AjaxParams 对象。在请求成功后,提示用户提交成功,并在 2 秒后关闭活动。在处理选择图片的回调事件中,根据选择的图片获取美食图片的绝对路径,并将路径拆分获取图片的文件名。然后通过 FinalHttp 库将美食图片上传到服务器,并在上传成功后将服务器返回的图片地址显示在界面上。核心代码如下:

```
private void createTopicPost(boolean isShow){
    AjaxParams params = new AjaxParams();
```

```
    httpPost(Constants.URL + "addFood", params,
    Constants.actionId.resultCode, isShow, "正在添加...");
    @Override
    public void onActivityResult(final int requestCode, int
    resultCode, Intent data) {
        if (resultCode == RESULT_OK) {
            switch (requestCode) {
                case REQUEST_CODE_SCAN_GALLERY:
                    urlInfor = data.getData();
                    String imagePath = getRealPathFromUri(this, data.getData0());
                    Break;
                    super.onActivityResult(requestCode, resultCode, data);}}}
```

(2) 添加菜品类型

商家可以通过增加新的菜品类型来区分菜品。商家添加美食类型图如图 4 所示。

(3) 订单管理

商家管理员通过查看订单,进行备餐发货,发送订单后,商家管理员可以把订单状态修改为“已发送”。商家修改订单状态图如图 5 所示。



图 5 修改订单状态图



图 6 查看订单评论图



图 7 商家信息图

通过调用 updateOrderTable 方法和 updateOrderBingTaiTable 方法来请求修改订单信息的接口,用于处理修改普通订单状态。修改订单的核心代码如下:

```
private void updateOrderTable(boolean isShow,
OrderInforModel orderInforModel) {
    AjaxParams params = new AjaxParams();
    params.put( action flag", "updateOrderTable );
    params.put("tableMessage",
    mListData.get(posIndex).getTableNo0);
    params.put( orderId", orderInforModel.getId0 + ")
    httpPost(Constants.URL + Constants.APP.FoodAction, params,
    Constants.actionId.resultState, isShow, "正在加载");}
```

(4) 查看评价

商店可以查看订单评价,了解顾客用户对产品的满意度。商家查看订单评论图如图 6 所示。

4.4 顾客用户模块

(1) 商家、菜品信息查看

顾客用户在首页查看商家店铺信息,点击进入店铺,可以查看到该店的各种菜品介绍和价格。商家信息图如图 7 所示。

通过发送 HTTP 请求,从服务器获取商家的食物信息。在回调方法中解析服务器返回的数据,并将食物按照不同的类型进行分类,展示在食物类型列表和默认食物列表中。商家信息展示的核心代码如下:

```
private void addTypeMessage(boolean isShow){
    AjaxParams params = new AjaxParams();
    params.put("storeid", MemberUserUtils.getUid(this));
    params.put("name", metMoney.getText().toString());
    httpPost(Consts.URL + "addType", params,
    Consts.actionId.resultState, isShow, "正在加载..");}
```

(2) 菜品选购

顾客用户可以在食物列表中选择菜品,并通过点击增加按钮(setJia()方法)将选中的菜品数量加1,或通过点击减少按钮(setJian()方法)将选中的菜品数量减1。同时,更新相应的界面和数据。通过点击结算按钮,跳转到支付页面,用户可以查看购物车中已选中的菜品和总价格。在TotalMoney()方法中,计算选中食物的总价格,并展示在界面上。添加菜品到购物车的用户选购图如图8所示。



图 8 用户选购图



图 9 订单图



图 10 评价图

用户选中的食物和相关信息提交给服务器,进行订单的创建和支付操作。发送 HTTP 请求,调用 httpPost 方法,将参数传递给服务器的 addOrder 接口进行订单提交。用户选购的核心代码如下:

```
private void OrderAction(boolean isShow) {
    String foodIdMsg = "";
    for (int i = 0; i < listChoice.size(); i++) (
    foodIdMsg += listChoice.get(i).getId0+", "
    AjaxParams params = new AjaxParams();
    params.put("ordermsgid", foodIdMsg.substring(0, foodIdMsg.length()-1)+"");
    params.put("orderinfor", choiceFoodMsg.substring(0, choiceFoodMsg.length()-1)+" ");
    params.put("ordermoney",
    this.getIntent().getStringExtra("payMoney "));
    params.put("orderuserid"
    MemberUserUtils.getUid(this));
    params.put("orderusername",
    MemberUserUtils.getName(this))
```

```
params.put("orderstoreid",
    this.getIntent().getStringExtra("storeid"));
    params.put("orderaddress", choiceAddree.getText().toString());
    httpPost(Consts.URL + "addOrder", params,
    Consts.actionId.resultFlag, isShow, "正在支付..");}
```

(3) 我的订单

顾客用户可以在我的订单中查看订单的具体情况,并可以取消未备餐的订单。订单查看的订单图如图9所示。

(4) 价订单菜品

顾客用户对于购买过的订单可以进行打分评价和发表评论,评价订单菜品的评价图如图10所示。

(5) 个人中心

顾客用户在个人中心可以修改密码、收获地址等。用户下订单时可以标注自身的位置,从而使快递员能精准送达。用户地址管理图如图11所示。

4.5 系统管理员模块

系统管理员的后台管理模块是为 PC 端界面设计的,主要功能有:商家管理、用户管理、菜品审核等。

(1) 商家管理

系统管理员在后台可以对商家信息进行审核,经过审核后的商家才能发布商品。对于违反平台规则的商家也可以删除。

(2) 审核菜品

系统管理员需要对商家新发布或首页推广的菜品进行审核,审核通过后,菜品才到首页显示推送。处理前端请求的功能,更新菜品状态的功能。这个方法



图 11 用户地址管理图

根据传入的菜品 ID 和状态信息,在数据库中更新相应菜品审核的状态。核心代码展示:

```
@RequestMapping("updateOrderState")
public void updateOrderState(HttpServletRequest request,
    HttpServletResponse response, Orderfood mode) throws
    IOException {
    UpdateWrAPPER<Orderfood> updateWrAPPER = new
    UpdateWrAPPER<Orderfood>();
    updateWrAPPER.eq(id, model.getid()).set(
    model.getOrderstate(),
    orderstate);
    if (orderMapper.update(null, updateWrAPPER) > 0) {
    JsonObject jsonmsg = new JsonObject();
    jsonmsg.put("repMsg", "状态更新成功");
    jsonmsg.put("repCode", "666");
    response.getWriter().print(jsonmsg);
    }
    else {
    JsonObject jsonmsg = new JsonObject();
    jsonmsg.put("repMsg", "no");
    jsonmsg.put("repCode", "111");
    response.getWriter().print(jsonmsg);
    }
}
```

(3) 用户管理

系统管理可以对于违反平台规则的用户进行删除。删除用户。通过参数获取要删除的用户 id 和用户类型 utype, 然后使用 UserMapper 的 deleteById 方法删除数据库中对应 id 的用户记录。最后根据 utype 返回相应的逻辑视图。核心代码分析:

```
@RequestMapping("deleteUser")
public ModelAndView deleteUser(String id, int utype) {
    userMapper.deleteById(id);
    Map<String, Object> map = new HashMap<>();
    map.put("utype", utype);
    ModelAndView modelAndView = new ModelAndView();
    if (utype == 1) {
    modelAndView.addObject("listData", userMapper.selectByM
    ap(map));
    modelAndView.setViewName("list user");
    }
    else {
    modelAndView.addObject("listData", userMapper.selectByM
    ap(map));
    modelAndView.setViewName("list store");
    return modelAndView;
    }
}
```

5 结束语

校园外卖点餐 APP 的设计与实现,采用成熟的 Spring Boot 框架和 Android Studio 等技术进行开发,应用高级语言 Java 进行代码编写,开发效率高和代码具有可读性。校园外卖点餐 APP,每个模块都经过需求分析和详细设计,APP 的用户界面设计简单直观,易于操作,基本满足用户和商家的日常业务需求。校园外卖点餐 APP 可以大学生提供方便快捷的外卖点餐服务,促进了数字化校园建设,提高了校园生活的智能化水平。

参考文献

- [1] 李志伟. 基于 Android Studio 开发平台的简易阅读器设计与实现 [J]. 电脑编程技巧与维护, 2023, (09): 47-49.
- [2] 蔡丹. 基于 Android 的校园外卖系统设计与实现 [D]. 南昌: 江西财经大学, 2018.
- [3] 陈祥豪. 大学校园外卖发展现状及对策分析——以杭州师范大学为例 [J]. 经营与管理, 2022, (03): 75-81.
- [4] 李章恒. 校园外卖系统设计与实现 [D]. 济南: 山东大学, 2022.
- [5] 蔡丹. 基于 Android 的校园外卖系统设计与实现 [D]. 南昌: 江西财经大学, 2018.
- [6] 张京蕊, 张茹冰. 互联网外卖行业透明化可行性分析 [J]. 北京信息科技大学经济管理学院, 2020, (16): 46-47.
- [7] 刘申晓. 基于 MVC 框架的网上订餐系统的设计与实现 [D]. 郑州: 解放军信息工程大学, 2010.