



华南师范大学  
South China Normal University

# 华师拼拼 APP 系统 开发说明文档

组别: 第二组

指导教师: 李慧

学科专业: 计算机科学与技术

所在院系: 计算机学院

二零二一年一月

# 目录

|                    |    |
|--------------------|----|
| 一、产品设计方案 .....     | 3  |
| 1 项目实施可行性报告 .....  | 3  |
| 1.1 行业市场分析 .....   | 3  |
| 1.2 同类产品分析 .....   | 4  |
| 1.3 自身条件分析 .....   | 4  |
| 2 产品定位及目标 .....    | 4  |
| 2.1 用户群定位 .....    | 4  |
| 2.2 产品定位 .....     | 5  |
| 2.3 风格形象 .....     | 5  |
| 2.4 竞争定位 .....     | 5  |
| 3 产品内容总策划 .....    | 5  |
| 3.1 设计与测试规范 .....  | 5  |
| 3.2 开发程表 .....     | 5  |
| 4 技术解决方案 .....     | 6  |
| 5 推广方案 .....       | 6  |
| 6 运营规划 .....       | 6  |
| 6.1 广告服务 .....     | 6  |
| 6.2 会员服务 .....     | 6  |
| 二、产品实现方案 .....     | 6  |
| 1 系统的主要功能 .....    | 6  |
| 2 软件主要功能用例建模 ..... | 7  |
| 3 软件系统结构化分析 .....  | 7  |
| 4. 系统数据库设计 .....   | 9  |
| 5. UI 界面设计 .....   | 9  |
| 5.1 用户中心 .....     | 10 |
| 5.2 发布拼单 .....     | 11 |
| 5.3 参与拼单 .....     | 11 |
| 5.4 信息管理 .....     | 12 |
| 5.5 记录笔记 .....     | 13 |
| 6 系统环境 .....       | 13 |

|                      |    |
|----------------------|----|
| 6.1 设备要求 .....       | 13 |
| 6.2 软件要求 .....       | 13 |
| 6.3 接口要求 .....       | 13 |
| 7 关键技术和技术难点 .....    | 14 |
| 7.1 关键技术 .....       | 14 |
| 7.2 技术难点 .....       | 20 |
| 8 用户体验记录和分析 .....    | 20 |
| 9 已完成的改进和存在的问题 ..... | 21 |
| 9.1、已完成改进 .....      | 21 |
| 9.2、存在问题 .....       | 23 |
| 三、测试大纲和测试报告 .....    | 23 |
| 四、产品安装和使用说明 .....    | 24 |

## 一、产品设计方案

### 1 项目实施可行性报告

#### 1.1 行业市场分析

##### （1）总体市场状况：手机网民规模

2017 年，我国移动互联网用户达到 7.53 亿，移动互联网渗透率为 54.10%；预计到 2020 年，我国移动互联网用户将达 8.88 亿，渗透率达 64.50%；日本和韩国，移动互联网渗透率已经在 90%左右，新加坡移动互联网渗透率也超过了 81%；2013 年 5 亿移动互联网用户中有 3.02 亿网购人口，线上支付比例 60.4%；到 2017 年，线上支付人口人数比例达到了 71%。预计随着线上支付的普及，及对线上购物的认可，线上购物人口将进一步增加，线上零售消费额占比也将有所增加。

##### （2）生活服务类共享经济持续发展

共享经济是指利用互联网等现代信息技术，以使用权分享为主要特征，整合海量、分散化资源，满足多样化需求的经济活动总和。共享经济是信息革命发展到一定阶段后出现的新型经济形态，是整合各类分散资源、准确发现多样化需求、实现供需双方快速匹配的最优化资源配置方式，是信息社会发展趋势下强调以人为本和可持续发展、崇尚最佳体验与物尽其用的新的消费观和发展观。2020 年我国共享经济规模依然保持高速增长，市场结构更趋合理。从市场结构来看，2018 年我国在交通出行、共享住宿、生活服务、共享医疗等 7 大领域积极推动共享经济的发展，拓宽共享业务的布局。2018 年生活服务市场规模为 15894 亿元，占全部共享经济交易总规模的 54.02%，位列第一；其次是生产能力领域，交易规模为 8236 亿元，占比 27.99%。

### （3）外卖市场的增长空间

外卖市场的增长空间。居民餐饮支出增长稳健，对比发达国家有较大空间。外卖是对人们做饭和外出就餐的替代，预计市场规模仍有翻倍空间。目前外卖逐渐回归中速增长，长期势头依然看好。随着互联网普及、城镇化、年轻一代崛起，外卖的渗透率将持续提高。同时消费升级与消费场景多元化，不断推高外卖客单价。而生活节奏加快、品类扩张、懒人经济，外卖消费频次也将不断提升。

### （4）社交类产品发展迅猛

社交是人类生活中不可少的一部分，随着智能手机的出现，社交产品从 PC 端开始走向移动端，而用户也在不断增长。根据数据显示，2018 年中国移动社交用户规模为 7.37 亿，预计未来两年仍将稳步增长，到 2020 年有望突破 8 亿人。2019 年多款面向年轻用户群体的社交产品备受资本青睐，它们以新技术、新玩法、新场景等要素为切入点，抢占移动社交细分赛道。根据分析认为，资本向来较为重视年轻人市场，定位于 00 后、95 后等年轻群体的社交产品更易收获资本的支持。

## 1.2 同类产品分析

本产品提供外卖拼单、大学生社交等一系列基于华南师范大学实名制的学生服务，目前市场还没有功能完全重合的 app 出现。而将本产品的功能拆分来看，外卖产品如美团外卖解决了用户需要网罗周围所有餐厅但免去获得餐厅电话而叫外卖的需求。以前叫外卖的传统方式是通过电话预定且前提是需要客户提前知道附近餐厅的电话，而美团外卖的出现帮用户省去了需要获取周围附近餐厅电话的成本，同时也极大程度地帮助商铺触及更大的用客户群。同时，美团外卖提供的配送服务帮助商铺节省外卖人力成本。社交产品如微信作为一款社交工具软件，微信是非常简洁好用的。首页聊天功能明确，可以做到用完即走。微信朋友圈的评论和点赞都是只能看到自己的老友，对方仅有的好友是看不到的；另外微信是没有好友推荐的，表明微信是用来维系经营自己已有的社交关系。

## 1.3 自身条件分析

美团外卖目前提供的拼单服务不尽完善，产生了无法提供陌生人之间拼单服务的痛点。本产品基于华南师范大学学生的环境下，在一定程度上保证了外卖拼单交易的安全性，同时满足了陌生人之间拼单的需求，是生活服务类共享经济发展的产物。而本产品与同类社交产品不同之处在于能在一个用户熟悉的环境（华南师范大学）扩充用户的社交，在一定程度上满足了用户扩充社交圈的需求，同时又易于用户接受。

## 2 产品定位及目标

### 2.1 用户群定位

华南师范大学在校大学生。

## 2.2 产品定位

本产品的核心功能为外卖拼单，在此基础上扩充出学习拼团、运动拼团、社交服务、发布闲置、记录笔记等功能，满足大学生校园生活中各类目前还没有被充分满足与解决的需求。

## 2.3 风格形象

本产品以青春活泼的界面设计与流畅优雅的用户交互体验，旨在给在校大学生以最简约的方式满足自己的拼单及社交需要。

## 2.4 竞争定位

本产品与目前已有的校园社交类产品（如超级课程表）不同，提供了外卖拼单服务，是目前大学生校园生活一块还没有被满足的痛点，以此增添用户粘性；同时提供的社交服务是在此基础上建立的，以实名制的方式让用户在舒适的范围内扩充交友群体。

# 3 产品内容总策划

## 3.1 设计与测试规范

本产品设计功能如下：

- 1、发布拼单信息功能。拼单发起人根据自己的拼单意愿在首页发布相应拼单信息；
- 2、拼单类别多样化，包括但不限于拼外卖，拼学习，拼运动等模块；
- 3、拼单功能。拼单参与者根据自己的意愿加入拼单发起人发布的拼单中；
- 4、评论交流功能：拼单参与者找到自己感兴趣的相应拼单后，在与拼单发起者达成合作之前，可针对拼单的具体内容与发起者进行交流沟通，以确定是否进行拼单合作；
- 5、发帖功能：拼单发起者在发布意愿之前，可通过产品提供的交流社群进行发帖咨询，通过交流结果再考虑是否发起拼单。拼单参与者可通过交流社群发帖，展开相关模块拼单的讨论或提出疑问。同时，用户可通过发帖功能发布自己的闲置物品进行交易；
- 6、记录笔记功能：用户可通过产品中的记录笔记功能，方便快捷地记下一些待办事项，给学生的日常生活提供了方便。

## 3.2 开发程表

| 日期                    | 完成进度                         |
|-----------------------|------------------------------|
| 2020/10/1-2020/10/20  | 成员讨论开发内容，交流意见后制定开发与分工部署      |
| 2020/10/21-2020/11/21 | 实现产品的四大基本功能，即拼单，参与拼单，讨论与发帖功能 |

|                       |                           |
|-----------------------|---------------------------|
| 2020/11/22-2020/12/10 | 完善产品的 UI 设计，并对上一阶段的开发进行优化 |
| 2020/12/10-2020/12/20 | 完成产品的初步测试与改进              |
| 2020/12/21-2021/01/10 | 完成对产品的最后修改与测试             |

## 4 技术解决方案

本产品基于 Android 开发，主要的开发环境为 Android Studio，基于 Kotlin 语言。

## 5 推广方案

本产品拟采用以下推广方案：

平台推广：通过微信公众号，抖音，B 站等流量平台进行广告推广，包括视频推广，文章推广，首页推广等。

搜索推广，利用以百度为代表的搜索引擎提高搜索排名的措施来推广产品

客户推广：推荐好友赠送平台福利，每邀请一个新用户注册可获得限时一周的会员服务

## 6 运营规划

本产品对于普通高校用户提供免费的基本服务，商业运营模式如下：

### 6.1 广告服务

本产品可为其他平台或商业机构提供推广服务，并收取相应的广告费用。

### 6.2 会员服务

本产品提供会员收费服务，会员用户可以享受优先服务，如置顶该用户发布的拼单信息或发的讨论贴，会员还可以拥有属于自身的专属标签与专属装扮。

## 二、产品实现方案

### 1 系统的主要功能

华师拼拼 APP 包括 6 大模块：启动模块、拼单服务、闲置交易、社交讨论、记录笔记、用户信息。各模块的主要功能描述：

- (1) 启动模块。包含华师拼拼客户端注册页、登录页，引导用户在 APP 中进行信息注册，以便于使用 APP 后续提供的全部功能，和用户信息的持久化保存。
- (2) 拼单服务。用户可以在华师拼拼客户端发布拼单帖子，填写拼单帖子的标题、描述

和图片，其他用户可以通过这些信息参与拼单。校园拼单服务解决了其他外卖 APP 中起送费高、配送费贵，与其他拼单 APP 中拼单的安全性不确定的痛点。

- (3) 闲置交易。用户可以在华师拼拼客户端发布闲置物品交易帖，填写交易帖的标题、描述和图片，其他用户可以根据这些信息与卖方进行私下的聊天交流，来决定是否进行交易，与如何进行交易等等。
- (4) 社交讨论。用户可以在华师拼拼客户端发布普通讨论帖，填写讨论帖的标题、描述和图片，与华师内志同道合的同学讨论问题，进行安全性更高，共享信息覆盖面更广的社交活动。
- (5) 记录笔记。用户可以在华师拼拼客户端找到记录笔记的功能，作为学生生活中的便捷小工具整合在本应用中，进一步提高了华师拼拼功能的丰富性。

## 2 软件主要功能用例建模

华师拼拼的主要功能是拼单与发帖讨论。“拼”单参与者通过系统提供的“拼”单交流功能对待拼“拼”单进行交流咨询，通过系统的发帖功能对相应模块“拼”单进行交流讨论。“拼”单发起者针对相应“拼”单中的“拼”单参与者提出的疑问进行答复和沟通，同样可以通过系统的发帖功能对相应模块“拼”单进行交流讨论。对应功能的用例图如图 1 所示。

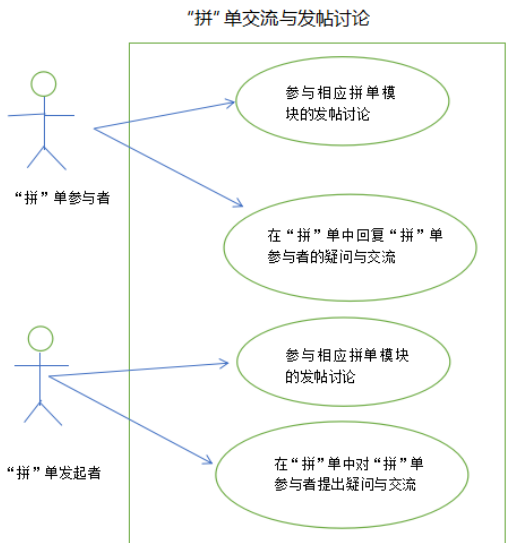


图 1 软件主要功能用例图

## 3 软件系统结构化分析

系统第 0 层数据流图如图 2 所示

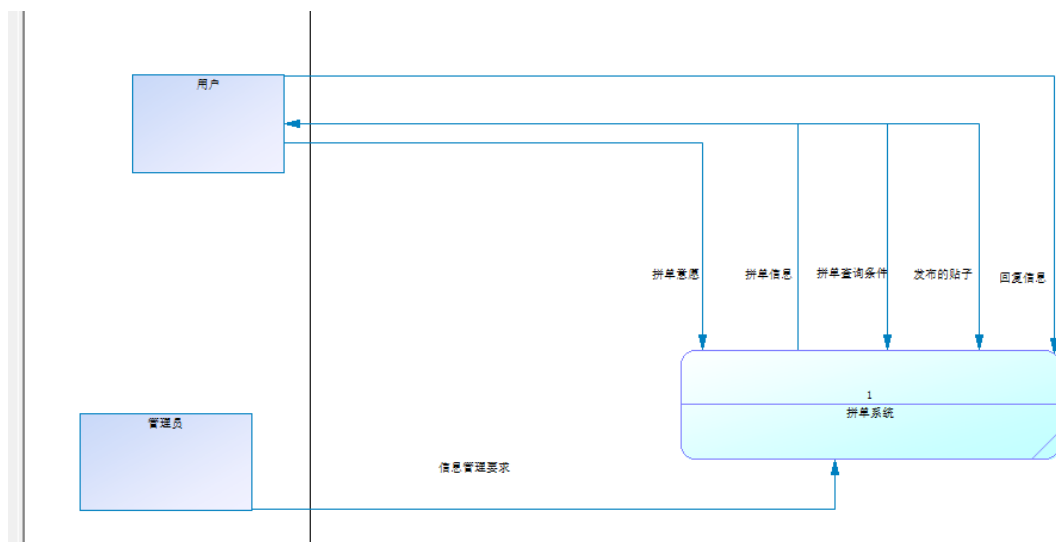


图 2 第 0 层数据流图

系统第 1 层数据流图如图 3 所示

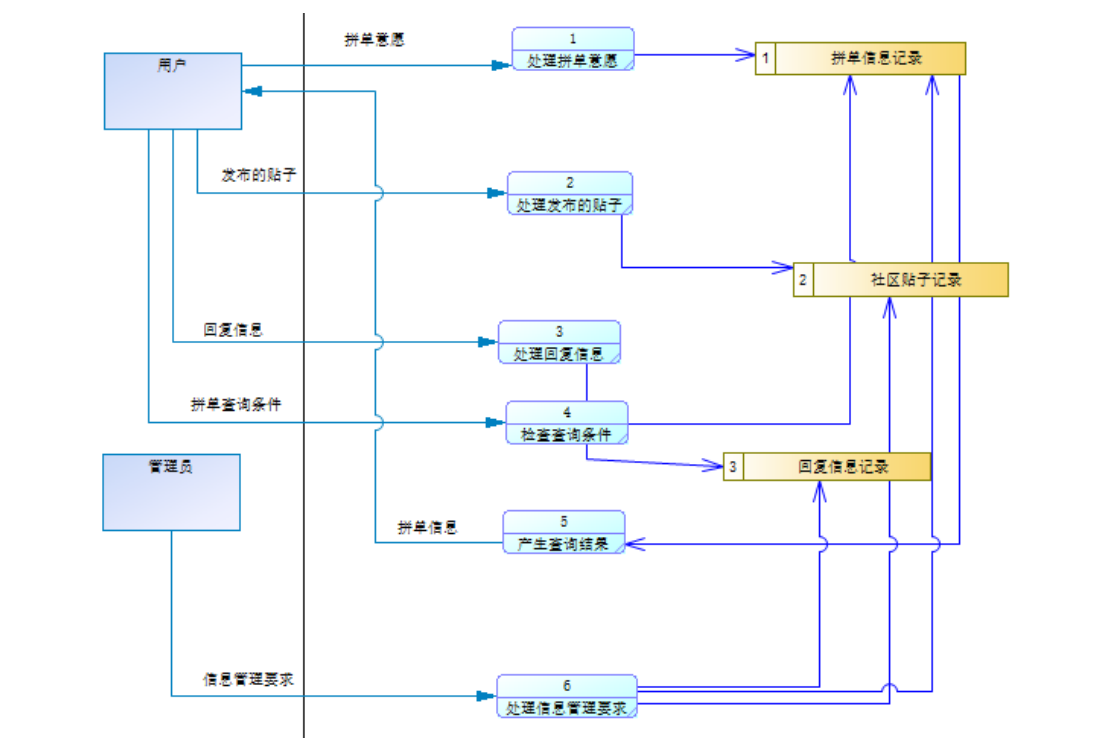


图 3 第一层数据流图

对第 1 层数据流图进行分解，依次得到第 2 层数据流图如图 4、5、6 所示

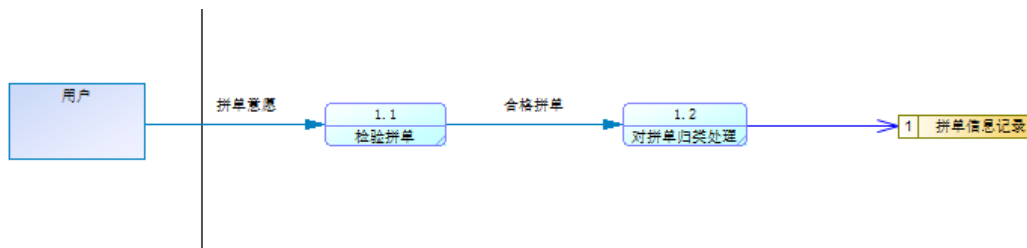


图 4 用户处理拼单意愿



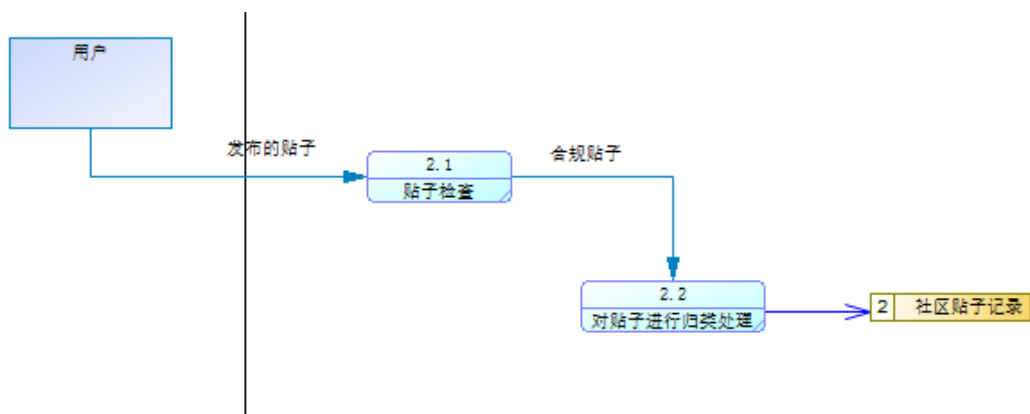


图 5 用户处理拼单帖子

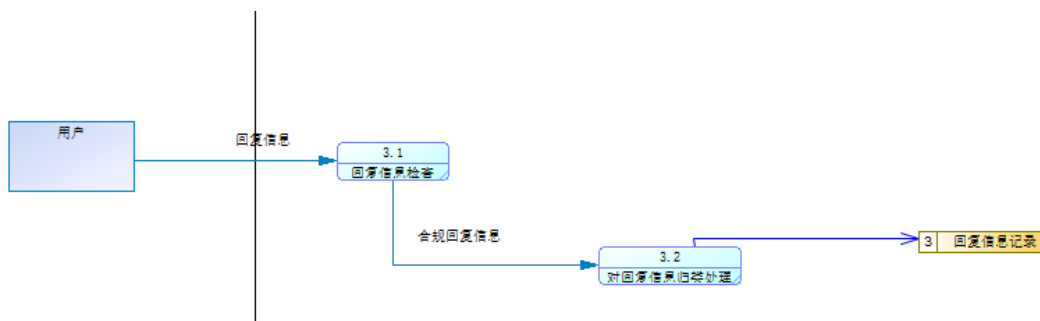


图 6 用户处理回复信息

## 4. 系统数据库设计

软件系统数据库设计的相关 E-R 图如图 7 所示

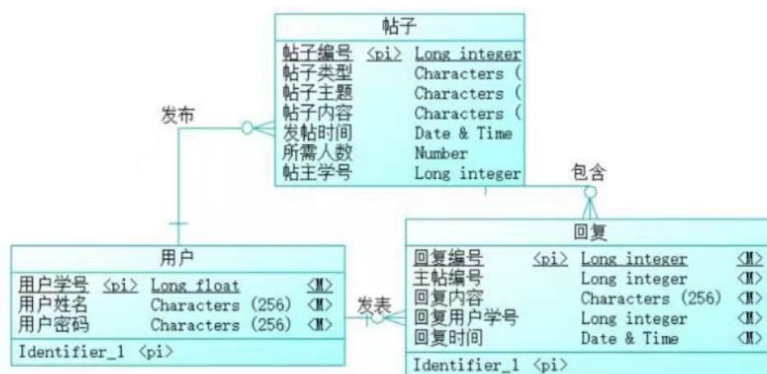


图 7 系统数据库设计 E-R 图

## 5. UI 界面设计

以下为华师拼拼 APP 在手机上的实测截图：

## 5.1 用户中心



图 8 登录界面



图 9 注册界面



图 10 管理界面

## 5.2 发布拼单



图 11 拼单帖子展示



图 12 编辑拼单帖子

## 5.3 参与拼单



图 13 拼单帖内回复



图 14 与发布拼单用户私聊

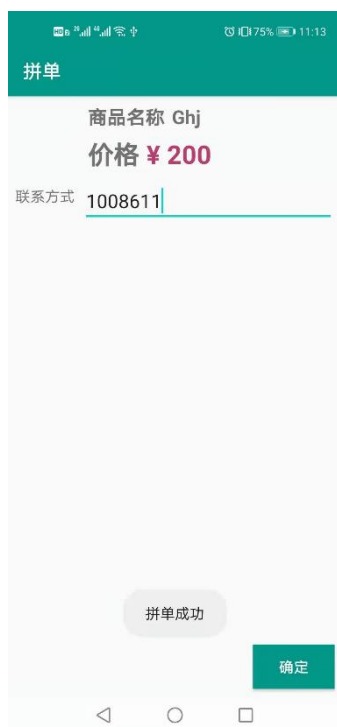


图 15 参与拼单



图 16 拼单消息列表

## 5.4 信息管理

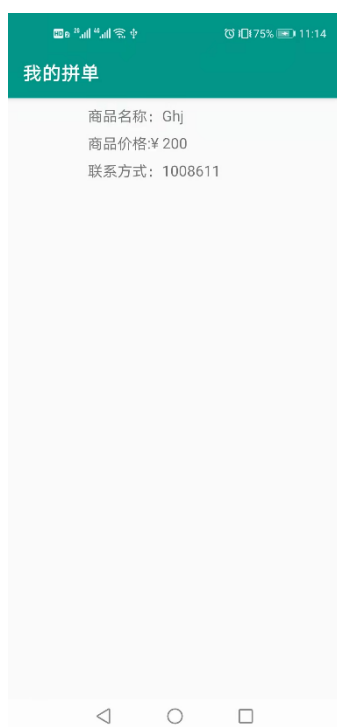


图 17 用户参与的拼单

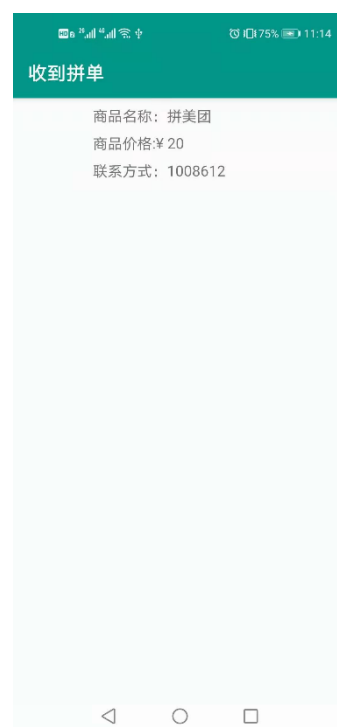


图 18 用户接受的拼单

# 5.5 记录笔记



图 19 笔记功能入口

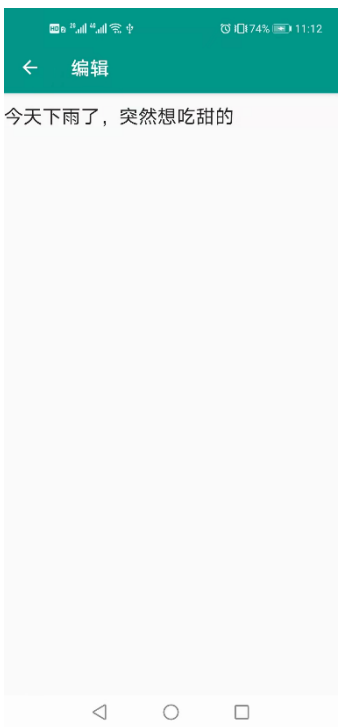


图 20 编辑笔记



图 21 笔记列表

# 6 系统环境

## 6.1 设备要求

要求设备安装 Android7.0 及以上版本或 IOS10.0 以上版本的操作系统

## 6.2 软件要求

设备需要安装微信 3.0 以上版本

## 6.3 接口要求

- 1、位置定位接口：准确对学生们的位置进行定位确认，区分相应的校区，并根据位置的远近计算出最优的拼单策略。
- 2、系统通知接口：当有最新消息提醒时，比如：拼单成功或失败、有最新且适合的拼单信息推送，会及时在通知栏提醒使用者，并发出震动。

## 7 关键技术和技术难点

### 7.1 关键技术

#### 7.1.1 后端云 BaaS

BaaS (Backend as a Service——后端即服务) 是一种新型的云服务, 旨在为移动和 Web 应用提供后端云服务, 包括云端数据/文件存储、账户管理、消息推送、社交媒体整合等。开发者无需过多研究服务器端程序, 而只需调用云计算平台提供的 API, 使用相应 SDK, 就能迅速完成上述功能。

华师拼拼 APP 选用了 BMOB 后端云。通过 BMOB 无需自己搭建服务器给该 APP 快速添加一个安全灵活的后台管理系统, 主要提供数据存储服务和用户管理的两大常用功能。

针对移动应用开发的特性, BMOB 还提供了主动推送服务、离线数据操作功能、数据关联操作以及数据多线程操作和回调操作等三个功能, 不仅解决了用户网络不稳定的情况, 同时可以高效快速实现类似发帖问答相关的功能, 提升用户体验。

#### 7.1.2 数据读取

在华师拼拼 APP 开发过程中, 最经常用到的就是数据读取, 主要的流程是在 APP 中增加/删除/修改/查询时, APP 端通过专门的接口代码连接到相应的数据库, 并根据所需要增加/删除/修改/查询的数据提供的对应数据表名/ID/来进行操作。

关键代码展示如下:

##### 7.1.2.1 读取云服务器数据, 刷新拼单广场相应的适配器信息

```
layoutManager = LinearLayoutManager(activity)

rv_posts!!.layoutManager = layoutManager
adapter = PostAdapter(postList, 1)
rv_posts!!.adapter = adapter
pull_to_refresh!!.setRefreshing(false)
```

##### 7.1.2.2 修改用户信息

```
val bmobUser = BmobUser.getCurrentUser(MyUser::class.java)

newUser.update(bmobUser.objectId, object : UpdateListener() {
    override fun done(e: BmobException?) {
        if (e == null) {
```

```

        Log.d(TAG, "更新用户信息成功")
    } else {
        Log.d(TAG, "更新用户信息失败:" + e.message)
    }
}

```

#### 7.1.2.3 查询详情页信息

```

tv_author!!.text = username

tv_content!!.text = title + " " + detail
tv_time!!.text = intent.getStringExtra("time")
tv_price!!.text = price
tv_location!!.text = intent.getStringExtra("location")

```

#### 7.1.2.4 查询评论区信息

```

override fun done(objects: List<Comment>, e: BmobException?) {

    if (objects.isEmpty()) {
        flipProgressDialog!!.dismiss()
    } else {
        Collections.reverse(objects)
        rv_comments!!.isNestedScrollingEnabled = false
        layoutManager = LinearLayoutManager(activity)
        rv_comments!!.layoutManager = layoutManager
        adapter = CommentAdapter(objects)
        rv_comments!!.adapter = adapter
        flipProgressDialog!!.dismiss()
    }
}

```

#### 7.1.2.5 增加评论

```

comment.save(object : SaveListener<String>() {

    override fun done(objectId: String, e: BmobException?) {
        if (e == null) {
            refreshCommentData(objectID)
            Toast.makeText(activity, R.string.add_commit_success,
                Toast.LENGTH_SHORT).show()
        } else {

```

```

        Toast.makeText(activity, R.string.add_commit_failed,
Toast.LENGTH_SHORT).show()
    }
}

```

#### 7.1.2.6 删除贴子

```

p.delete(object : UpdateListener() {
    override fun done(e: BmobException?) {
        if (e == null) {
            Toast.makeText(activity, R.string.delete_success,
Toast.LENGTH_SHORT).show()
            activity.finish()
        } else {
            Toast.makeText(activity, R.string.delete_failed,
Toast.LENGTH_SHORT).show()
        }
    }
}
)

```

### 7.1.3 启动模块

在用户启动华师拼拼 APP 后，首先要进行的就是用户的注册或登录。若为第一次使用华师拼拼 APP，则需要在注册模块中填写个人信息进行注册；若为已经注册过的用户，则需要在登录模块填写自己的账号和密码进行登录，才可以执行后续的功能。用户在注册时，先向 Bmob 后端进行注册请求，注册成功后，再注册到环信 IM，两者都注册成功才为成功注册账号。登录成功后用户信息保存在本地。

关键代码展示如下：

#### 7.1.3.1 登录操作

```

bu2.login(object : SaveListener<BmobUser?>() {

    override fun done(bmobUser: BmobUser?, e: BmobException?) {
        if (e == null) {
            Toast.makeText(activity, getString(R.string.signin_success),
Toast.LENGTH_SHORT).show()
            flipProgressDialog!!.dismiss()
            val login = "login"
            EventBus.getDefault().post(login)

```



```

        activity.finish()
    } else {
        toast("登录失败，请检查用户名和密码")
        flipProgressDialog!!.dismiss()
    }
}
})

```

### 7.1.3.2 注册操作

```

bu.signUp(object : SaveListener<MyUser?>() {

    override fun done(s: MyUser?, e: BmobException?) {
        if (e == null) {
            Toast.makeText(activity, getString(R.string.signup_success),
                Toast.LENGTH_SHORT).show()
            bu.login(object : SaveListener<BmobUser?>() {

                override fun done(bmobUser: BmobUser?, e: BmobException?) {
                    if (e == null) {
                        Toast.makeText(activity, R.string.signin_success,
                            Toast.LENGTH_SHORT).show()
                        flipProgressDialog!!.dismiss()
                        val login = "login"
                        EventBus.getDefault().post(login)
                        activity.finish()
                    } else {
                        toast("登录失败，错误代码: "+e.toString())
                        flipProgressDialog!!.dismiss()
                    }
                }
            })
        }
    }
})

```

## 7.1.4 拼单服务与社交讨论

本应用提供的核心功能为拼单服务。用户在填写相应的拼单帖子或讨论帖子信息后，系统将相应的信息上传至云端的数据库中进行保存，同时在其他用户的客户端进行展示。关键代码展示如下：

### 7.1.4.1 从发帖页回调后获取要上传的信息

```

val postTitle = data!!.getStringExtra("post_title")

```

```

val postDetail = data.getStringExtra("post_detail")
val postLocation = data.getStringExtra("post_location")
val postPrice = data.getStringExtra("post_price")
val paths = data.getStringArrayExtra("data_return_2")

```

#### 7.1.4.2 发帖成功后刷新拼单广场

```

override fun done(objectId: String, e: BmobException?) {

    if (e == null) {
        Toast.makeText(this@MainActivity, "发布成功", Toast.LENGTH_SHORT).show()
        flipProgressDialog!!.dismiss()
        mListener!!.refreshView()
    } else {
        Toast.makeText(this@MainActivity, "发布失败", Toast.LENGTH_SHORT).show()
    }
}

```

#### 7.1.4.3 从云服务器获取与其他人的私信

```

BmobIM.connect(userInfo, objectId, object : ConnectListener() {

    override fun done(uid: String, e: BmobException?) {
        if (e == null) {
            //连接成功
            rv_message.visibility = View.VISIBLE
            val conversations = BmobIM.getInstance().loadAllConversation()
            if (conversations.size==0){
                toast("你还没有任何消息纪录")
                return
            }
        }
    }
}

```

#### 7.1.4.4 获取私信成功则刷新消息页对应的适配器信息

```

val layoutManager = LinearLayoutManager(activity)

rv_message!!.layoutManager = layoutManager
mMessageAdapter = MessageAdapter(conversations, activity)
rv_message!!.adapter = mMessageAdapter

```

#### 7.1.4.5 发送私信

```
val msg = BmobIMTextMessage()

msg.content = content
mConversationManager.sendMessage(msg, object : MessageSendListener() {
    override fun done(bmobIMMessage: BmobIMMessage, e: BmobException?) {
        if (e == null) {
            Log.d(TAG, "成功发送信息")
            val msg = Msg(content, Msg.TYPE_SENT)
            msgList.add(msg)
            adapter!!.notifyItemInserted(msgList.size - 1)
            msg_recycler_view!!.scrollToPosition(msgList.size - 1)
            input_text!!.setText("")
        } else {
            Log.d(TAG, "发送信息失败")
        }
    }
})
```

#### 7.1.4.6 拼单操作

```
order.save(object : SaveListener<String>() {

    override fun done(s: String, e: BmobException?) {
        if (e == null) {
            Toast.makeText(this@OrderActivity, "拼单成功",
Toast.LENGTH_SHORT).show()
            finish()
        } else {
            Toast.makeText(this@OrderActivity, "拼单失败",
Toast.LENGTH_SHORT).show()
        }
    }
})
```

#### 7.1.4.7 查询自己参与的拼单, 将数据传递至拼单列表

```
query.findObjects(object : FindListener<Order>() {

    override fun done(orders: List<Order>, e: BmobException?) {
        if (e == null) {
            Log.d(TAG, "查询成功: 共" + orders.size + "条数据。")
            if (orders.isEmpty()) {
```

```

        toast("你当前没有任何拼单信息")
        return
    }

    val layoutManager = LinearLayoutManager(this@MyOrderActivity)
    rv_my_order!!.layoutManager = layoutManager
    val adapter = OrderAdapter(orders)
    rv_my_order!!.adapter = adapter

```

#### 7.1.4.8 查询他人参与我发布的拼单, 将数据传递至拼单列表

```

query.findObjects(object : FindListener<Order>() {

    override fun done(orders: List<Order>, e: BmobException?) {
        if (e == null) {
            Log.d(TAG, "查询成功: 共" + orders.size + "条数据。")
            if (orders.isEmpty()) {
                toast("你当前没有任何拼单信息")
                return
            }
        }

        val layoutManager = LinearLayoutManager(this@GetOrderActivity)

        rv_get_order!!.layoutManager = layoutManager
        val adapter = OrderAdapter(orders)
        rv_get_order!!.adapter = adapter
    }

```

## 7.2 技术难点

Bmob 服务器提供了一些列的 API 接口, 当客户端发送请求时, Bmob 服务器接受请求返回 Bmob 指定的数据类型, 客户端接受数据显示在 UI 界面。在开始开发时应熟悉 Bmob 开发文档, 自己开发时会遇到未知的问题, 要上网查资料解决。关于 BMOB 数据库表结构的设计, 需要建立用户表、帖子表。用户表给学生使用, 需要用户名、密码等相关字段。帖子表需要标题、描述、图片等字段。要考虑用户表和帖子表之间的联系, 建立帖子表指向用户表的外键。

上传图片到 BMOB 服务器时, 需要从本地图库中取得图片的 Uri, 取得图片的绝对路径, 通过绝对路径将图片转换为 BmobFile 格式, 再执行上传。由于本项目资金有限, 故没有在 BMOB 服务器上实现上传图片的功能。

## 8 用户体验记录和分析

我们在产品开发的中期阶段, 向 20 个用户分发了调查问卷, 调查本应用存在的不足之处与改进的意见与建议。收到的答卷如下所示:

| 序号 | 提交答卷时间           | 您的建议或看法                                    |
|----|------------------|--|
| 1  | 2020/12/24 19:40 | 侧边抽屉用不惯，还是像微信 qq 那样菜单栏在下面吧                 |
| 2  | 2020/12/24 21:36 | 整挺好看哈                                      |
| 3  | 2020/12/26 13:28 | 希望整合一个闲置功能                                 |
| 4  | 2020/12/26 13:43 | 没有   |
| 5  | 2020/12/26 15:34 | 感觉什么讨论功能有点多余                               |
| 6  | 2020/12/28 10:12 | 最好加个消息页面，因为别人回复你的信息可能很多，只有帖子管理的话得一个一个翻，很麻烦 |
| 7  | 2020/12/28 10:46 | 牛逼   |
| 8  | 2020/12/28 11:27 | 没有性能优化                                     |
| 9  | 2020/12/28 11:47 | 坐等成品∠( ᐢ 」 ∠ )__                           |
| 10 | 2020/12/28 19:25 | 没有   |
| 11 | 2020/12/29 9:19  | 既然是学生 app 不如加个记笔记功能？                       |
| 12 | 2020/12/29 9:23  | 不错不错                                       |
| 13 | 2020/12/29 12:50 | 数据库会不会有密码泄露问题？                             |
| 14 | 2020/12/29 13:05 | ui 不够美观                                    |
| 15 | 2020/12/29 13:16 | 收到拼单有没有系统提醒的                               |
| 16 | 2020/12/29 13:23 | 无  |
| 17 | 2020/12/29 17:52 | 讨论功能有点鸡肋                                   |
| 18 | 2020/12/29 18:03 | 懒狗想在 app 里完成支付                             |
| 19 | 2020/12/29 18:15 | 不够简洁实用                                     |
| 20 | 2020/12/29 19:02 | 社交功能不够强大 希望再扩展这方面                          |

表 1 用户调查问卷反馈

## 9 已完成的改进和存在的问题

### 9.1、已完成改进

提取调查问卷中的有效信息，我们总结出了以下在完成中期阶段设计后需要改进的问题：

- 1、侧边栏不符合用户习惯；
- 2、增加消息列表界面；
- 3、增加闲置交易功能；
- 4、增加记录笔记功能；
- 5、应用性能优化；
- 6、APP 内提供支付功能。

其中，已经完成 1~4 的功能改进。5 与 6 为本应用后续完善的方向。

改进 1、 2：导航栏由侧边改为横向，并增加消息页

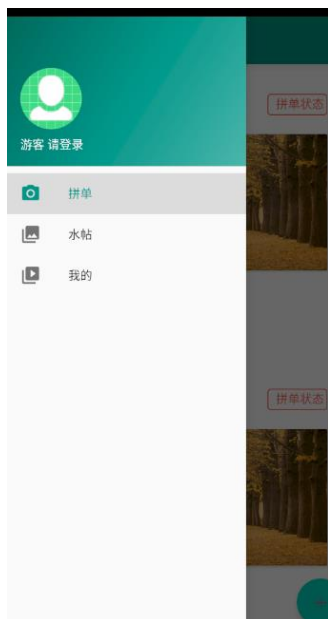


图 14 原始主页



图 15 改进后主页

改进 3: 用户也可以发布闲置信息, 交流学习或者自己感兴趣的事情, 发帖页面合并



图 16 原始判单页



图 17 改进拼单页

同时改进的还有发帖按钮, 除了更换意义更加明显的图标外, 新版本的发布按钮还会随着页面滑动进行隐藏, 支持代码如下:

```
rv_posts!!.addOnScrollListener(object : RecyclerView.OnScrollListener() {
    override fun onScrolled(recyclerView: RecyclerView?, dx: Int, dy: Int)
    {
        if (dy > 0)
            fb!!.hide()
        else if (dy < 0)
```

```
fb!!.show()
    }
})
```

改进 4：增加记笔记的功能

改进前的 app 无记笔记的功能，经调查后发现其他同学有记笔记的需求，于是在导航栏增加“更多”栏目，目前仅支持记笔记的功能，后续还可以根据同学们的需求进一步添加其它实用功能

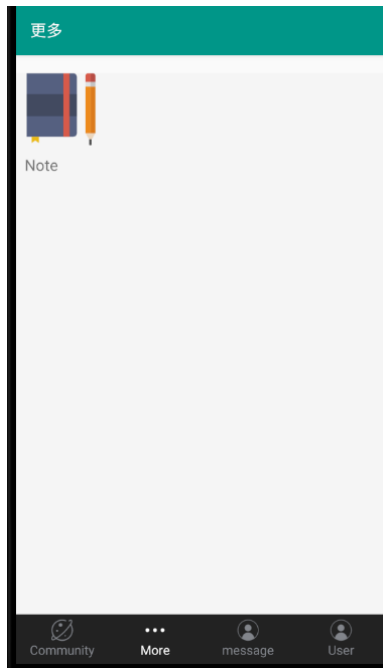


图 18 “更多”页面

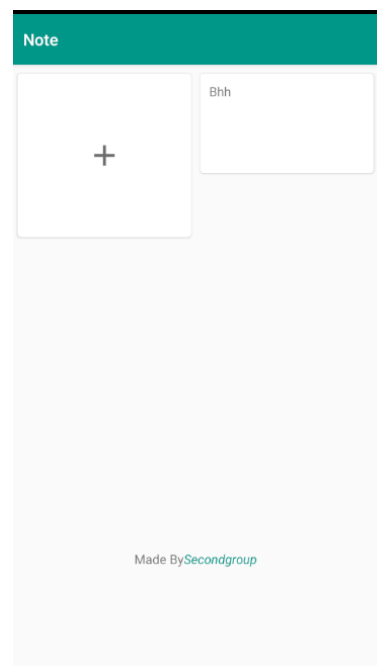


图 19 笔记页

## 9.2、存在问题

由于 Bomb 云服务器上传文件，即上传图片需要收取服务费用，由于经费问题该项暂时无法实现，导致图片均无法显示，这是后期需要进一步改进的。

## 三、测试大纲和测试报告

本次测试针对未经过登录用户使用 APP 的用例进行了测试。测试结果如下：

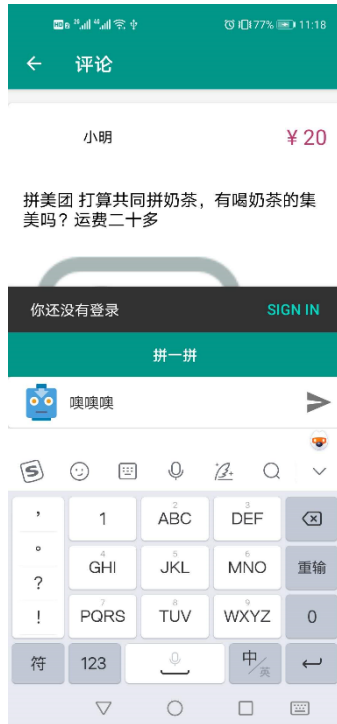


图 20 测试评论功能

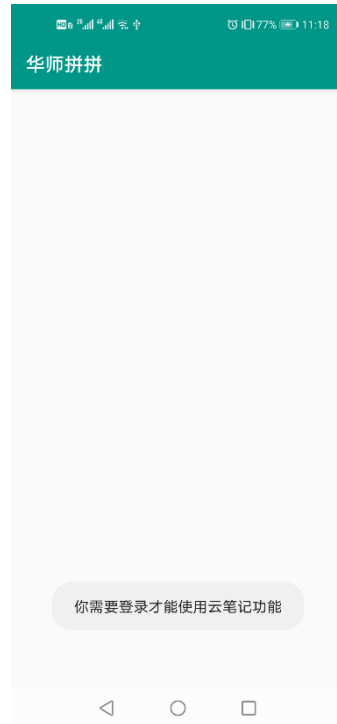


图 21 测试记录笔记功能

从上图可以看到，用户在未登录时不能使用评论和记录笔记功能，测试结果良好。

## 四、产品安装和使用说明

下载华师拼拼 APP 的安装包 SCNUPin.apk，选择安装后即可使用。

初次使用需要先在用户中心进行注册与登录，然后便可以使用 APP 内提供的功能，如发布拼单、参与拼单、发布闲置、参与讨论、发起私聊、记录笔记等功能。