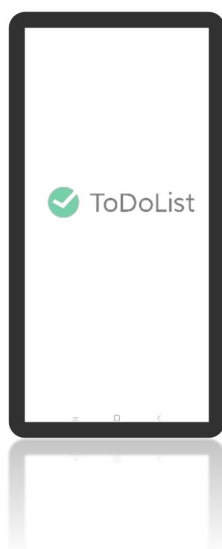


华南师范大学

《移动智能应用开发》课程项目  
系统开发说明文档



项 目 题 目：料未来 APP-智能推荐 ToDoList

所 在 学 院：计算机学院

项 目 组 长：朱培挺

小 组 成 员：赵文焯，李恒，龙泓汀

开 题 时 间：2021 年 10 月 13 日

## 目录

一、产品设计方案.....	4
1.1 项目实施可行性报告.....	4
1.1.1 行业市场分析.....	4
1.1.2 竞争分析.....	6
1.1.3 竞争优势.....	6
1.2 产品定位及目标.....	7
1.2.1 产品定位.....	7
1.2.2 目标群体.....	7
1.2.3 用户故事.....	8
1.3 产品内容总策划.....	8
1.3.1 应用开发流程规划.....	8
1.3.2 设计与测试规范.....	10
1.3.3 开发日程表.....	11
1.4 技术解决方案.....	11
1.5 推广方案.....	12
1.5.1 产品展示会.....	12
1.5.2 网络推广.....	12
1.5.3 线下广告推广.....	13
1.6 运营规划书.....	14
1.6.1 盈利模式.....	14
1.6.2 风险评估.....	14
1.6.3 产品发展战略.....	16
二、产品实现方案.....	17
2.1 系统的主要功能.....	17
2.2 模块设计.....	19
2.2.1 用户注册模块.....	19
2.2.2 用户登录模块.....	22
2.2.3 个人信息管理模块.....	25
2.2.4 任务管理模块.....	27
2.2.5 任务查看模块.....	35
2.2.6 帮助模块.....	38
2.2.7 反馈模块.....	39
2.3 关键技术和技术难点.....	41
2.3.1 组件间的信息传递以及路由跳转.....	41
2.3.2 推荐算法的实现思路与遇到的困难.....	41
2.4 用户体验记录和分析.....	43
2.5 已完成的改进和存在的问题.....	44
三、测试大纲和测试报告.....	45
3.1 性能测试.....	45
3.2 功能测试.....	48
3.3 测试总结.....	49
四、产品安装和使用说明.....	49

4.1 产品安装方法.....	49
通过下载软件对应的 apk 格式文件，在文件管理系统处点击打开安装即可。.....	49
4.2 产品使用说明.....	49
4.2.1 适用机型.....	49
4.2.2 使用方法.....	49

# 一、产品设计方案

## 1.1 项目实施可行性报告

随着时代的进步，有节奏的自律生活成为了人们追求的事物。而 Todolist（计划表）则是一项有助于人们实现自律生活的有效工具。目前市面上 Todolist 的软件较多，但美中不足的是，这些软件功能较单一，且越来越难以满足当下我们的需求。我们处在快节奏的生活中，制作计划的精力并不算太足够，如果软件能够帮助用户自动规划生活，也许会提高用户的体验；同时，我们也期望能够找到志同道合的人和自己一道去完成 Todolist。因此，智能推荐的 Todolist 也许是更迎合当下生活的软件。人们要是装配上这类 APP，即可通过算法与自身的力量，料算出自己未来的生活。

### 1.1.1 行业市场分析

PEST 模型是良好的宏观经济市场分析模型。在 PEST 模型中，分别从国家政策、经济、社会与技术四个维度对产品所在的市场进行宏观分析。据此，本小组对即将开发的产品进行了宏观经济分析，结果如下表所示。

PEST 模型分析-智能推荐 Todolist APP		
维度	情况	解读
政策	①国家推出双减等减负政策，有意识地减少人们过度倾注的精力，让人们拥有更多的属于自己的时间； ②同时，国家主流舆论也一直提倡“斜杠青年”、“拒绝‘躺平’”等概念，	在政策大环境的鼓励下，人们会拥有足量的自己的时光，同时也有自己的需求，需要进行规划使用。

	呼吁人们在生活中有所追求。	
经济	<p>①近年来愿意主动使用 APP 的规划时间的人群主要为中青年为主，这类人群的消费能力较高；</p> <p>②近年来，围绕着 todolist 这一主题进行开发的 APP 已经产生了许多变现模式，而这些 APP 有许多仍存活至今。</p>	对于当下市场而言，需求方有足够的消费能力，而且从市场上同类产品多元的变现模式来看，市场已经证明了该类产品的商业变现的可行性，有助增大料未来 APP 商业变现的可能性。
社会	<p>①从百度指数的查询来看，“抗压能力”的搜索频次随年份呈现上升的趋势；</p> <p>②目前人们越来越愿意交给智能推荐简化自己的思考，从休闲娱乐、饮食到购物出行；</p> <p>③在哔哩哔哩搜索词条“时间管理”，以 up 主蜡笔与小勋的时间管理视频为例，仅一条就有 157 万的播放量，10.2 万的收藏，接近十分之一的收藏率。</p>	从百度指数的结果来看，目前社会对于抗压能力的要求与认知愈发普遍。而从哔哩哔哩平台中该时间管理视频的火爆来看，人们发现计划制定是一个非常重要的时间管理的工具。而良好的时间管理可以有助于抗压，智能推荐有助于实现良好的时间管理。
	①从实现的角度来看，要实现该算法主要依赖数据库、前后端交互等技术，目前这类技术已经比较成	

技术	<p>熟，网上有许多教程可以借鉴；</p> <p>②从算法的角度来看，本 APP 的推荐算法主要基于频率统计、用户偏好加权等角度进行设计，逻辑比较简单，无需较大的运算量，或者是考虑调用已成型的 API 进行推荐，实现较为简单，效果预计比较可预见。</p>	目前从实现的角度来看，是可以实现的。
----	---	--------------------

### 1.1.2 竞争分析

根据本小组在应用市场上的调研，目前市场上的 todolist 可以分为以下几类：打卡类、UI 设计类、数据统计类、用户自定义类。虽然他们会有重叠，但基本上具体的描述如下所示。

打卡类：如 ToDo 打卡 APP，可以通过 todo 进行签到，并且拍照打卡，或者像 forest 这一软件进行种树打卡；

UI 设计类：如时光序 APP，主要通过不同主题色卡等设计作为卖点，吸引用户进行时间管理设计；

数据统计类：如 iToDo APP，能够对用户不同时段的使用时长进行统计；

用户自定义类：如微软的 microsoft to-do APP，能够根据用户的订阅任务进行智能推荐。

### 1.1.3 竞争优势

本产品竞争的优势为能够实现更高精度的智能推荐，以目前的市场产品为例，

市场上的产品大多是告诉用户自己在哪些事情付出了多少时间,但是并未能实现推荐的效果,哪怕是微软的,也是推荐即将到期的任务或者是推荐目前所列的事情中最重要的内容,但它不能推荐用户未来去做什么,无法预料未来。

## 1.2 产品定位及目标

### 1.2.1 产品定位

料未来 APP 是一款多用户、多标签的,能基于用户已有的任务与完成情况,推荐未来不同类别任务的的 todoList APP,为用户降低了规划未来、培养习惯的成本。

产品功能与特性:

- ①根据用户的特定标签进行未来某些事情类别推荐;
- ②根据用户在过去的时段中规划的事情的类别标签进行关于未来某些时段的事情类别的推荐;
- ③用户可自行选择推荐算法;
- ④UI 流畅舒适简洁,操作简易;
- ⑤具有 todoList 产品的基本事项记录的功能;

### 1.2.2 目标群体

- ① 有意愿掌握生活节奏的人群;
- ② 对自身的时间分配感兴趣,但平时又没有太多精力做过多时间规划的人群。
- ③ 对 APP 统计的数字不敏感,但又想根据自己平时的节奏规划生活的人群。

### 1.2.3 用户故事

①曹小姐是一名高校教师，她日常的生活内容主要是科研工作、居家带娃、备课上课等事情，她此前下载过一些计划表类的 APP 进行事项提醒，也比较满意效果，但是对于她来说，每天要进行事项的添加，制作成一个事项提醒是一个麻烦的事情，因为很多琐碎的事项很难去想起，想起了也很难去规划；后来，她通过下载料未来 APP，通过记录了此前的事项完成情况之后，得到了明晰的时间分区推荐，知道自己可以在什么时段做什么事情，进而无需自己多做计划就得到了一个能够完成日常推荐的工具，她感觉自己的生活更轻松了，无需过度刻意地去制定计划；

②小陈是一名高中的学生，他日常的生活节律主要是学习，每天的主要玩手机的时间是有限的两小时，他希望能够做好未来天数的规划，但他又想用手机获取足够的资讯以及进行练习，因此他之前有用过手账本以及手机计划表等 app 进行规划，但手账本的设计太死板，原有的计划表 APP 需要自己做较多的考量才能做好规划，他后来下载了料未来 APP，根据推荐的计划表进行学习，自身的时间规划时间大大减少，提高了学习与生活的效率；

## 1.3 产品内容总策划

### 1.3.1 应用开发流程规划

#### 1.3.1.1 明确需求

全组人员一起开需求研讨会，主要讨论需求的合理性及存在的必要性，研究并对可行性进行评估，并将最后的需求一步步拆分、细分，最后分解成一个个明确的需求功能点。

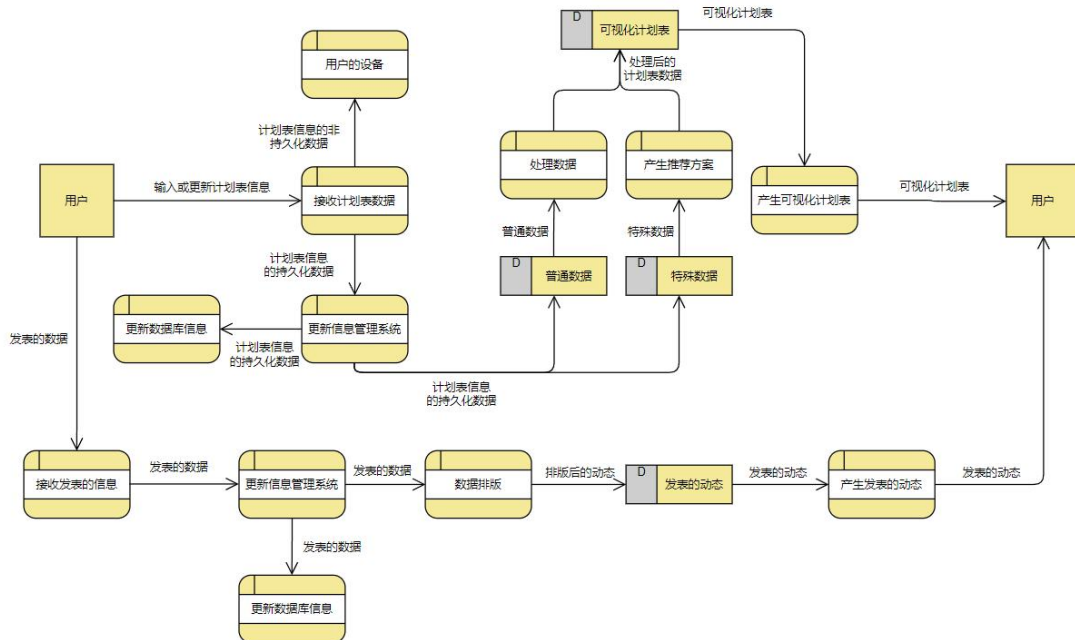
最初定下的主要功能有：

①用户的事项制定功能，该功能包含了事项的分类，修改，删除，标记事实完成与事实未完成的功能；



- ②用户对自己账户信息（账号，密码）的管理功能；
- ③智能推荐事项类别，以及推荐最近一天做什么的功能；

经过讨论，料未来的数据流图如下图所示。



### 1.3.1.2 UI 设计

小组开 UI 设计讨论会，设计出美观大方的 UI 界面，做好配色、元素设计等，使 UI 效果图基本上与最终的 APP 页面效果一致。本团队决定采用 XUI 与 QMUI 作为 UI 设计与前端开发的组件库，它们提供了一系列常用的组件和丰富多彩的样式主题，比如最常见的 TextView、Button、EditText、ImageView、Spinner、Picker，还有其他诸如 LoadingView、StateLayout、BadgeView、MarqueeView 等等。

### 1.3.1.3 组件化开发

开发小组根据实际功能需求，对软件进行组件化开发。这种开发模式耦合度低，易于团队合作开发。后端采用了 Java 语言以及 Sqlite（本地存储）+Bmob（云端存储）的解决方案。

#### 1.3.1.4 算法研究

对算法进行规划，查询论文调研，设计不同的算法推荐方案，比如根据用户往期规划的不同时间段的事项的标签进行统计后，根据频率进行推荐；又或者根据用户设定的某些参数进行加权统计，对方案进行调整等等，并根据论文调研的结果设计评价推荐算法的评估体系，根据评估的效果决定优化的方向。

目前本团队根据查询的资料，初定的推荐方案的思路有以下几个：

①针对冷启动问题，本项目采用了基于用户信息标签进行推荐、基于用户对过去的计划进行预推荐的方案；

②针对具体的推荐方案的解决方向，本团队拟采用以下两个方向的推荐算法，一是基于用户标签的推荐算法，二是基于用户标签的统计推荐算法。

③评估推荐方案好坏的指标，本团队倾向于采用准确率、召回度、汉明距离、内部相关性以及流行度进行评估推荐的准确度。

#### 1.3.1.5 测试与改进

对整个 APP 进行性能与功能的测试，发现 bug 进行修复，直到测试通过。

### 1.3.2 设计与测试规范

#### 1.3.2.1 设计规范

①模块化设计：一个功能封装为一个模块。

②统一代码、文件命名风格。

③内部设计规范：

主题色为绿色，RGB（109，196，156）；

按钮样式尽量为圆角。

### 1.3.2.2 测试规范

测试资源：

测试任务开始前，检查各项测试资源，确定功能需求、性能需求后开展测试。

测试要点：

#### ①功能测试

确保手头的功能需求文档为当前最新版本，同时需要确保所有的软件功能都已实现且逻辑正常。

#### ②性能测试

性能测试方面必须满足硬件压力条件下的测试需要。

### 1.3.3 开发日程表

开发时间段	工作内容
2021/10/08-2021/10/12	产品设计方案编写
2021/10/11-2021/10/23	明确需求
2021/10/24-2021/11/28	UI 设计与前端实现
2021/11/29-2021/12/24	后端开发
2021/12/25-项目结束	测试

## 1.4 技术解决方案

本产品基于 Android 开发，主要的技术方案是：QMUI+XUI+JDK+Android Studio+SQLite+Bmob，针对不同的功能模块开展组件化开发。

## 1.5 推广方案

### 1.5.1 产品展示会

邀请相关老师、有兴趣的学生和开发团队在学校报告厅等公共场所，探讨产品的发展前景、和优缺点，进行产品的宣传也借此了解用户的需求，让用户清晰了解产品的功能作用。产品展示会可一举二得，既得到了用户的认同，又显露了自身优势。具体展示会的开办形式视具体情况而定，对于场地资金等显示条件的不足应当做出相应的调整。可以通过一些文化节，双创周等校园活动进行宣传，或者社团活动日通过摆地摊的形式进行宣传。

### 1.5.2 网络推广

网络推广具有高效性和便利性，且网络推广对资金的需求较小，本产品的推广主打网络宣传，我们可以在互联网上的各种应用平台上线。主要的形式可以有以下几种：

(1) 拍摄宣传视频，介绍产品用法，描绘应用场景，上传到各大短视频和其他热门应用平台。

(2) 基于网络论坛所进行的口碑传播，比如小红书，微博，豆瓣，虎扑等等，分享使用攻略，以及产品衍生的故事。

(3) 利用小程序等衍生产品的形式进行推广。

(4) 开设公众号、简书等博客进行相应的宣传，主要推送用户的使用故事与体验，引导用户在上述平台进行分享；

(5) 创作一些时间管理的攻略向同质内容的自媒体进行投稿，一方面赚取稿费，一方面获取广告的效应。

### 1.5.3 线下广告推广

#### 1.5.3.1 人员推广

可以通过在校园内派发问卷调查或者访谈等形式进行市场调研，还可以让用户直接使用我们的应用，让用户有更加直观的体验，我们通过用户的反馈也进一步了解并发现市场的需求，可以根据调研结果有针对性的制定个性化的营销计划，提供相应的功能模块的同时通过传单等形式对产品进行宣传，宣传范围可以从校园内逐步向外辐射扩散。

#### 1.5.3.2 广告宣传

利用海报广告形式，绘制各类活动宣传海报，向学校提出申请，在特定的地方进行粘贴如雍园附近的校道等地方，并配有专人在附近进行宣传介绍。

#### 1.5.3.3 产品联动

当用户量积累到一定程度，亟待破圈时，可以利用与其他的品牌 IP 的联动，策划一些文创联名产品，实现双赢。

## 1.6 运营规划书

### 1.6.1 盈利模式

#### 1.6.1.1 VIP 会员制

特定的会员身份可以开启某种特定的推荐功能，或者是打破共享计划表人数的某种限制，实现协同办公。

#### 1.6.1.2 售卖线下文创周边产品

设计手账本、Tshirt 等产品作为周边进行售卖，赚取相应的利润。。

#### 1.6.1.3 广告投放

积累了一定用户基础后，通过首页轮播图、文章末尾广告推送进行广告的投放。

### 1.6.2 风险评估

#### 1.6.2.1 运营风险

运营风险主要为平台用户少，流量少致使难以盈利。目前许多具有类似功能的产品都因客户少、没流量致使难以继续维护。团队通过分析目前市场上产品的失败原因、发展路线与经营模式等，从失败案例中汲取经验，总结出预防此类风险的措施以及出现该风险后的应急措施。

首先，目前许多产品通病在于操作繁琐，同时一上线即有大量的广告，降低了用户体验，对此，我们决定在平台运营初期不投放广告，确保了平台的简便性与内容的简洁。这在一定程度上增强了产品的吸引力，先留住第一批接触平台的用户。

此外，加强平台的宣传。目前许多产品不温不火，部分原因并不是用户不愿意使用，而是用户不知道有这个产品。对此，我们制定了多途径的推广策略，提高产品的曝光度，用产品故事与产品特性，以及一定频率的运营活动提高用户的粘性，让更多的人看到与使用料未来 APP。

### 1.6.2.2 市场风险

（1）技术门槛较低，同类新技术、新产品的出现，会争夺市场份额

为防止市场被抢占，我们计划在第一年内完成对华师内学生的覆盖，即先行主导市场。如果能让将来成为老师的学生成为平台的忠实用户，我们的平台会更容易接触到不同年龄段的用户，甚至走进课堂。同时，在推广运营期间不断优化平台，扩展功能，坚持市场调查，不断提高客户满意度。同时，及时关注类似的平台或软件，关注其创新功能与营销思路，以便及时对自身平台进行创新与完善，避免平台被市场淘汰。

（2）平台研发更新力度不足，难以实现升级换代。

平台研发更新力度不足。为防止平台运营期间出现技术研发或更新上的困难，团队的技术部需要引进较强的编程人员。这一步计划在平台开发初期开展，在平台运营期间不断补足。而技术成员则优先从华师计算机学院的高年级本科生或研究生中选取，给予一定薪酬，提高团队的技术研发水平。如果还是出现困难，则计划向社会上更加专业的技术人员寻求帮助，给予薪酬换取技术研发工作。

### 1.6.2.3 财务风险

由于平台开发初期流量较少且需要在平台开发和宣传上投入大量资金，这可能导致团队在资金上出现困难。除团队事先筹备的资金外，在研发推广过程中，由市场部积极与社会企业、机构等进行协商、签订合同以拉取赞助，使团队顺利渡过前期。若赞助拉取不够或资金数量较少，则考虑进行贷款。

### 1.6.3 产品发展战略

作为初发行软件、想要迅速提升我们的知名度就必须从线上线下两方面寻找知名度较高的合作伙伴，利用他们的流量和顾客量为我们带来第一批用户，提高我们的影响力。通过制定的推广策略逐步扩大产品在各大高校学生间的影响力，由此可以获得一批数量较大的系统使用群体，而由于本产品符合这类群体的目标需求，因此在众多使用体验中可逐渐形成本产品的口碑，通过口碑效应可进一步提高产品的影响力，整个过程可以划分为前期和后期。

前期：线下利用社团分享与相关活动进行宣传。同时绘制各类活动宣传海报，申请在校道的宣传栏、饭堂饭桌等地方粘贴宣传海报，发展华师校内客户群体；线上利用知名公众号的流量以小程序为切入点打开市场。

后期：华师校内市场开拓基本完毕，对于抖未来的产品打磨也有了用户使用的基础，可以进行下一步的出圈。针对校内外目标对象，我们通过与线下组织合作，线上宣传打开市场。一方面，线下与目标社团进行合作宣传，同时也与知名公众号、学习区 UP 主，大 V 等知名用户进行线上的合作宣传，通过邀请码注册向高知名度合作伙伴支付额外宣传报酬，利用他们的流量迅速打开平台的影响力。同时开放学生优惠，以学生身份如学校邮箱或学生证认证，即可获得学生优惠。

在用户反馈上面，高度关注第一批用户的体验感；以老用户的自发宣传代替额外的宣传投入。让第一批用户成为我们的宣传者。在形成一定的用户规模后，我们从功能优化方向转型成为运营优化方向，逐渐从产品固有的功能研发打磨转向用户运营以及双赢的盈利模式的探讨，在此阶段我们的主要盈利来源将转向广告、周边以及会员的收入。



## 二、产品实现方案

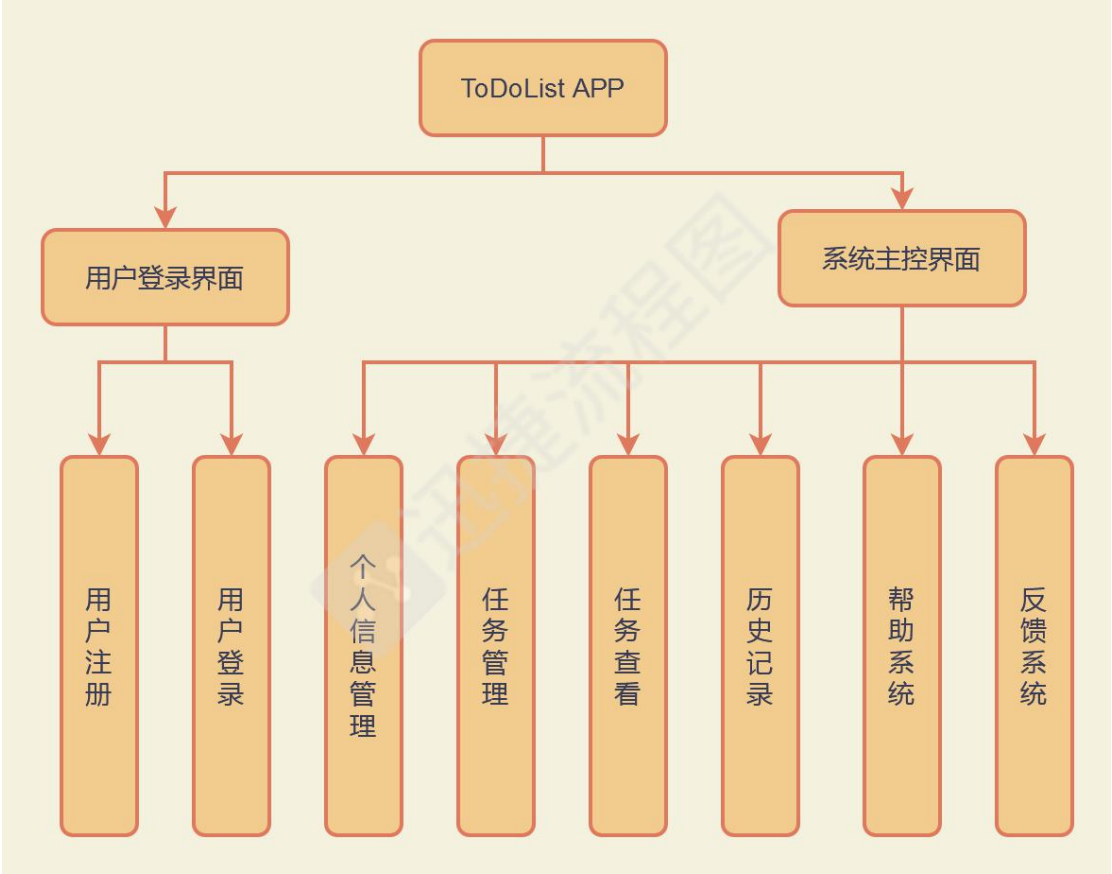
### 2.1 系统的主要功能

经过讨论与开发，结合实际情况，该软件的主要功能模块如下表所示（没有列出未实现的算法实现功能）。

模块名称	功能描述
用户注册	用户名未被使用，则将信息持久化，返回注册成功提示 用户名已存在，则返回用户已存在信息
用户登录	效验正确则读取该用户响应的任务信息并返回 效验失败则返回错误提示信息
个人信息管理	个人信息和用户头像修改
任务管理	任务的增删改查（包括标签、完成进度、完成时间、开始时间等）
任务查看	任务的属性可视化以表格、图表形式呈现 按照预先设定好的优先顺序对用户任务进行排序 根据当前时间高亮显示待完成任务
帮助	向用户提供应用使用说明 定期更新应用的 Q&A

反馈	<p>用户可以在此向开发者提供反馈</p> <p>开发人员定期读取、分析反馈信息，对 APP 进行改进</p>
----	---

其功能结构如下图所示。



## 2.2 模块设计

### 2.2.1 用户注册模块

#### 2.2.1.1 界面设计



数据要求

功能类型：查询数据和增加数据

数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空项	数据检验
AccountID	用户账户	11	文本框	Y	Y
Password	用户密码	18	password	Y	Y
Confirm_password	确认密码	18	password	Y	Y

### 2.2.1.2 模块内部逻辑

activity\_register.xml 用于显示注册界面的内容

RegisterActivity.java 用于注册数据处理

处理关键点：

检验用户输入账户、密码的合法性

检验两次密码输入结果是否相同

调用数据库接口检测用户名是否已存在

调用数据库接口添加新用户数据

处理过程：

当账号输入框内容变化时，检验账户输入的合法性。当密码输入框内容变化时，检验密码输入的合法性，若输入合法再检验密码与确认密码是否相同。当确认密码输入框内容变化时，检验密码与确认密码是否相同。若输入内容均非空且合法，则将注册按钮设置为可点击。点击注册按钮后，在数据库中查找输入的用户账户，如果用户账号已存在，则提示用户账户已存在；否则将用户账户与密码存入数据库，提示注册成功并使页面跳转到登录界面。

### 2.2.1.3 存储分配

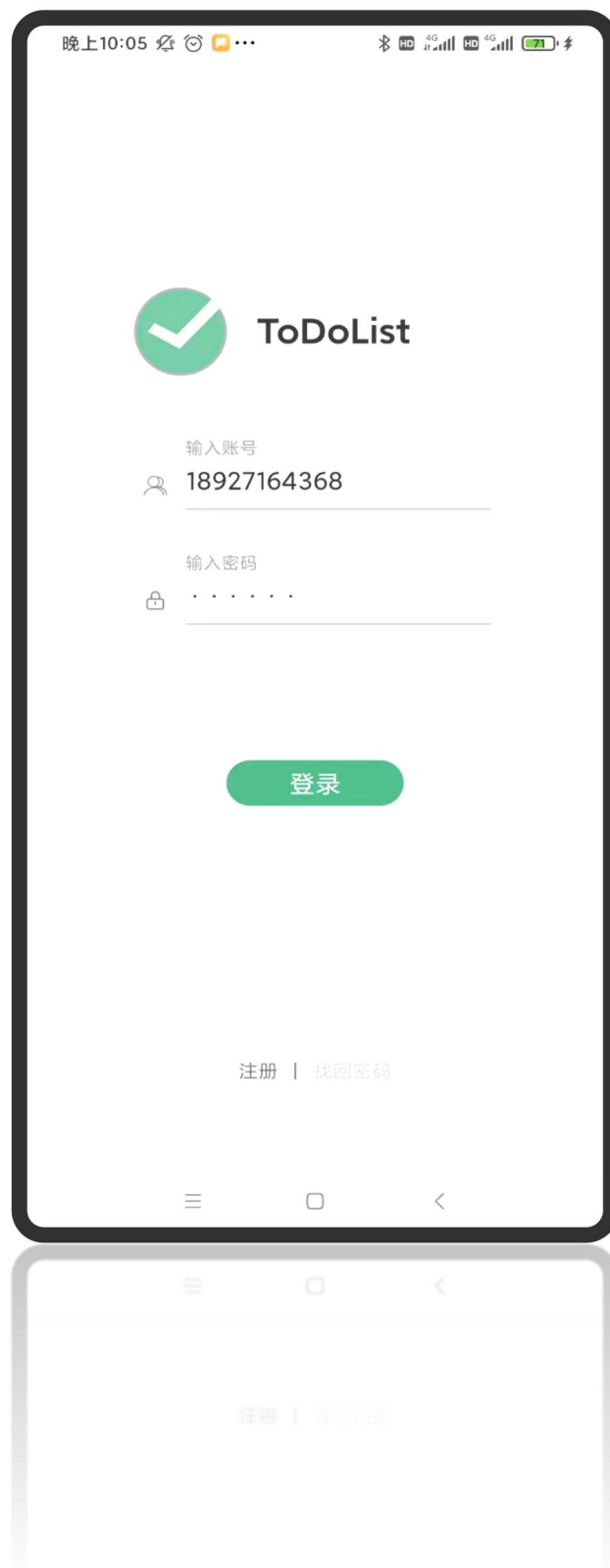
用户信息表（account）

编号	字段名称	字段含义	字段类型	字段长度	默认值	允许空	主键
----	------	------	------	------	-----	-----	----

1	AccountID	用户 id	String	11	√
2	Username	用户名	String	255	
3	Password	密码	String	18	
4	Sex	性别	String	255	保密
5	HeadPortrait	头像	Number	255	1
6	Describe	个人描述	String	255	√

## 2.2.2 用户登录模块

### 2.2.2.1 界面设计



数据要求

功能类型：查询数据

数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空	数据检验项
AccountID	用户账户	11	文本框	Y	Y
Password	用户密码	18	password	Y	N

### 2.2.2.2 模块内部逻辑

activity\_login.xml 用于显示登录界面的内容

LoginActivity.java 用于登录数据处理

处理关键点：

检验用户输入账户的合法性

调用数据库接口检测用户名与密码是否匹配

处理过程：

当账号输入框内容变化时，检验账户输入的合法性。当密码输入框内容变化时，检验密码输入的合法性。若用户账户输入合法，且密码非空时，将登录按钮设置为可点击。点击注册按钮后，在数据库中查找输入的用户账户，如果用户账号存在，则检验输入密码与该元组的密码是否匹配，如果匹配成功则记住密码并跳转到该账户的主页面；否则提示登录失败。

### 2.2.2.3 存储分配

用户信息表（account）

编号	字段名称	字段含义	字段类型	字段长度	默认值	允许空	主键
1	AccountID	用户 id	String	11			√
2	Username	用户名	String	255			

3	Password	密码	String	18	
4	Sex	性别	String	255	保密
5	HeadPortrait	头像	Number	255	1
6	Describe	个人描述	String	255	√



## 2.2.3 个人信息管理模块

### 2.2.3.1 界面设计



数据要求

功能类型：查询数据和修改数据

数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空项	数据检验
HeadPortrait	头像	1	单选框	Y	N
Username	用户名	255	文本框	Y	N
Sex	性别	255	单选框	Y	N
Describe	个人描述	255	文本框	N	N
Password	用户密码	18	password	Y	Y
Confirm_password	确认密码	18	password	Y	Y

### 2.2.3.1 模块内部逻辑

activity\_personal.xml 用于显示用户个人信息界面的内容

PersonalActivity.java 用于初始化个人信息界面显示内容、处理个人信息的修改

处理关键点：

调用数据库接口检测用户名与密码是否匹配

检验用户密码的合法性

检验两次密码输入结果是否相同

调用数据库接口修改已更改数据

处理过程：

用户点击头像，列出系统提供的默认头像。点击用户名或个人描述模块，弹出文本框编辑框。点击性别，显示性别下拉框。若用户作出任意修改，则更新数据库中内容。点击密码编辑框，进入密码更改界面。当密码输入框内容变化时，检验密码输入的合法性，若输入合法再检验密码与确认密码是否相同。当确认密码输入框内容变化时，检验密码与确认密码是否相同。若输入内容均非空且合法，则将更改密码按钮设置为可点击。点击更改密码按钮后，调用数据库接口更新用户密码。

### 2.2.3.2 存储分配

用户信息表（account）

编号	字段名称	字 段 含 义	字段类型	字 段 长 度	默 认 值	允 许 空	主键
1	AccountID	用户 id	String	11			√
2	Username	用户名	String	255			
3	Password	密码	String	18			
4	Sex	性别	String	255	保密		
5	HeadPortrait	头像	Number	1	1		
6	Describe	个 人 描 述	String	255		√	

### 2.2.4 任务管理模块

该模块可以拆分为任务添加模块、任务编辑模块两个子模块。

## 2.2.4.1 任务添加子模块

### 2,2.4.1.1 界面设计



### 数据要求

(1) 功能类型：新增任务计划、编辑任务计划、删除任务计划、查询与修改任务计划

(2) 数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空	数据检验项
Worktag	任务标签	255	单选框	Y	N
Workname	任务名称	255	文本框	Y	N
Is_all_day	全天任务	1	Switch 框	Y	N
startTime	开始时间	255	单选框	Y	N
endTime	结束时间	255	单选框	Y	N
workDescribe	任务描述	255	文本框	N	N

#### 2.2.4.1.2 模块内部逻辑

activity\_plan\_detail.xml 用于获取用户新建任务的内容

PlanDetailActivity.java 用于处理用户新建任务

处理关键点：

获取系统时间，初始化任务开始与结束时间

调用数据库接口新增任务数据

处理过程：

任务点击添加计划按钮，进入新建计划页面，获取系统当前时间用于初始化任务开始时间与结束时间，任务开始时间初始化为当前时间，任务结束时间初始化为当前时间后一小时。用户在页面内可选择任务标签、输入任务的标题与详细描述。默认任务为非全天任务，显示日期选择框与时间选择框，用户可将任务设置为全天任务，设置为全天任务后时间选择框将被隐藏，后台将任务时间区间设置为 0:00——23:59。当标题非空，用户可以保存计划，用户点击添加计划按钮后，将新建计划添加到数据库中。

## 2.2.4.1.3 存储分配.

任务信息表 (work)

编号	字段名称	字 段 含 义	字段类型	字 段 长 度	默认值	允 许 空	主键
1	WorkID	任务 id	String	255			√
2	AccountID	用户 id	String	11			
3	Worktag	任 务 标 签	Number	255			
4	workname	任 务 名 称	String	255			
5	workDescribe	任 务 描 述	String	18		√	
6	Is_all_day	全 天 任 务	Boolean	1			
7	startTime	开 始 时 间	Number	255			
8	endTime	结 束 时 间	Number	255			
9	performance	完 成 情 况	Boolean	1	false		
10	Is_delete	被删除	Boolean	1	false		

## 2.2.4.2 任务编辑子模块

### 2.2.4.2.1 界面设计





数据要求

功能类型：查询数据和修改数据

数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空	数据检验项
Worktag	任务标签	255	单选框	Y	N
Workname	任务名称	255	文本框	Y	N
Is_all_day	全天任务	1	Switch 框	Y	N
startTime	开始时间	255	单选框	Y	N
endTime	结束时间	255	单选框	Y	N
workDescribe	任务描述	255	文本框	N	N
performance	完成情况	1	单选框	Y	N
is_delete	标记删除	1	单选框	Y	N

#### 2.2.4.2.2 模块内部逻辑

activity\_plan\_detail.xml 用于获取用户获取任务变更的内容

PlanDetailActivity.java 用于初始化界面内容，将更新任务内容同步到数据库：

处理关键点

调用数据库接口获取任务 id 对应的详细任务内容，用于界面初始化

调用数据库接口更新任务 id 元组中的其他任务数据

处理过程：

在任务展示界面选定任务可以编辑任务内容，编辑包括将任务设置为已完成、被删除以及跳转到编辑页面编辑任务的除主键外的其他属性。已完成内容默认排序未完成任务的下方，被删除的任务不在展示列表显示，用户可以前往回收站将任务恢复或彻底删除。用户任务编辑界面重新为任务设置标签，标题，描述，开始时间与结束时间等基本信息。

## 2.2.4.2.3 存储分配

任务信息表 (work)

编号	字段名称	字 段 含 义	字段类型	字 段 长 度	默认值	允 许 空	主键
1	WorkID	任务 id	String	255			√
2	AccountID	用户 id	String	11			
3	Worktag	任 务 标 签	Number	255			
4	workname	任 务 名 称	String	255			
5	workDescribe	任 务 描 述	String	18		√	
6	Is_all_day	全 天 任 务	Boolean	1			
7	startTime	开 始 时 间	Number	255			
8	endTime	结 束 时 间	Number	255			
9	performance	完 成 情 况	Boolean	1	false		
10	Is_delete	被删除	Boolean	1	false		

## 2.2.5 任务查看模块

### 2.2.5.1 界面设计



数据要求

功能类型：查询数据

数据描述：

页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空	数据检验项
Worktag	任务标签	255	单选框	Y	N
Reverse	倒序排列	1	单选框	Y	N
Order_by	排序元素	1	单选框	Y	N
performance	完成情况	1	单选框	Y	N
Date	筛选日期	1	单选框	Y	N

### 2.2.5.2 模块内部逻辑

fragment\_home.xml 用于选取筛选条件和排序方式，并按序显示符合筛选条件的用户任务

common\_vertical\_set\_dialog.xml 用于获取用户的排序设置

fragment\_time\_line\_func.xml 用于按天展示任务

HomeFragment.java 用于初始化界面内容，提供任务编辑接口

TimeLineFuncFragment.java 用于获取选取日期的任务数据

TimeLineAdapter.java 用于初始化任务卡片，获取当前时间高亮显示当前正在进行的任务，提供任务编辑接口

DividerItemDecoration.java 用于绘制时间轴

处理关键点：

根据用户选择的标签、任务状态、日期等基本信息筛选出符合条件的任务

根据用户的排序设置按序显示筛选后的任务

获取系统当前时间高亮显示正在进行的任务

处理过程：

根据用户选择的标签、任务状态、日期等基本信息从数据库中筛选出符合条件的

任务，再根据用户的排序设置将任务排序显示。再获取当前系统时间高亮高亮显示正在进行的任务。用户可以点击标签下拉框设置筛选的标签，用户选择”所有“标签时，将属性值设为空，不按标签筛选任务。用户可以点击设置按钮，进入详细设置页面，用户可以点击设置页面的单选框将排序设置为按”开始时间“或”结束时间“排序，设置为”正序“或”倒序“排序，设置显示”已完成“、“未完成“或所有任务。用户点击任务可以对任务进行编辑或删除。

### 2.2.5.3 存储分配

任务信息表 (work)

编号	字段名称	字 段 含 义	字段类型	字 段 长 度	默认值	允 许 空	主键
1	WorkID	任务 id	String	255			√
2	AccountID	用户 id	String	11			
3	Worktag	任 务 标 签	Number	255			
4	workname	任 务 名 称	String	255			
5	workDescribe	任 务 描 述	String	18		√	
6	Is_all_day	全 天 任 务	Boolean	1			
7	startTime	开 始 时 间	Number	255			
8	endTime	结 束 时 间	Number	255			
9	performance	完 成 情 况	Boolean	1	false		
10	Is_delete	被删除	Boolean	1	false		

展示设置表(select)

编号	字段名称	字段含义	字段类型	字段长度	默认值	允许空	主键
1	AccountID	用户 id	String	11			√
2	Worktag	任务标签	Number	255		√	
3	Reverse	倒序排列	Boolean	1	false		
4	Order_by	排序元素	Boolean	1	true		
5	performance	完成情况	Boolean	1		√	
6	Date	筛选日期	Date	255			

## 2.2.6 帮助模块

### 2.2.6.1 界面设计

数据要求

功能类型：为用户提供使用方法

数据描述：本模块无需存储数据

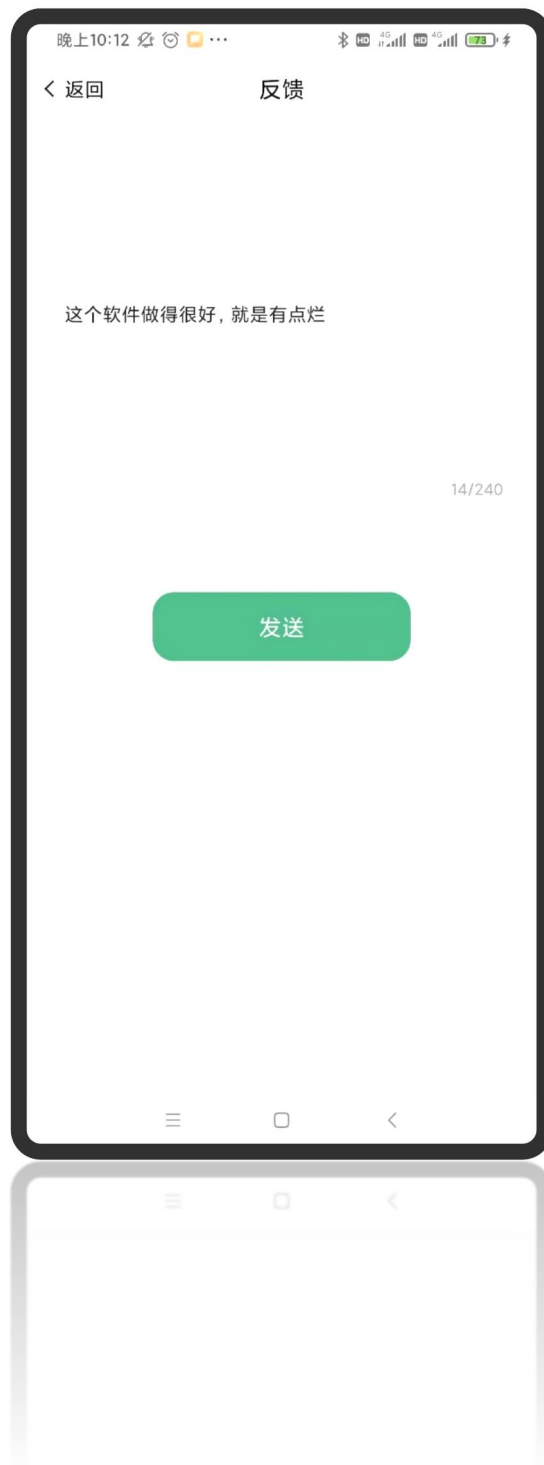
### 2.2.6.2 模块内部逻辑

处理过程：

根据用户点击帮助按钮的事件，弹出帮助信息页面，包含最基本的任务管理等功能的指引。

## 2.2.7 反馈模块

### 2.2.7.1 界面设计



数据要求

功能类型：用户点击提交反馈，弹出信息告知用户反馈成功

数据描述：页面显示录入字段如下：

字段名称	字段含义	长度	录入方式	是否非空	数据检验项
AccountID	用户 ID	11	自动获取	Y	N
FeedBack	反馈内容	255	文本框	Y	N

### 2.2.7.2 模块内部逻辑

处理关键点：

获取用户的 ID。

处理过程：

用户填写反馈内容，点击提交反馈的按钮后，后台会获取其用户 ID，以及反馈内容，存到云服务器的数据库中，交由开发团队了解具体情况。

### 2.2.7.3 存储分配

反馈处理记录表(feedback)

编号	字段名称	字段含义	字段类型	字段长度	默认值	允许空	主键
1	AccountID	用户 id	String	11			√
2	FeedBack	反馈内容	String	255			
3	IsOk	判定是否成功处理	Boolean	1	0		



## 2.3 关键技术和技术难点

### 2.3.1 组件间的信息传递以及路由跳转

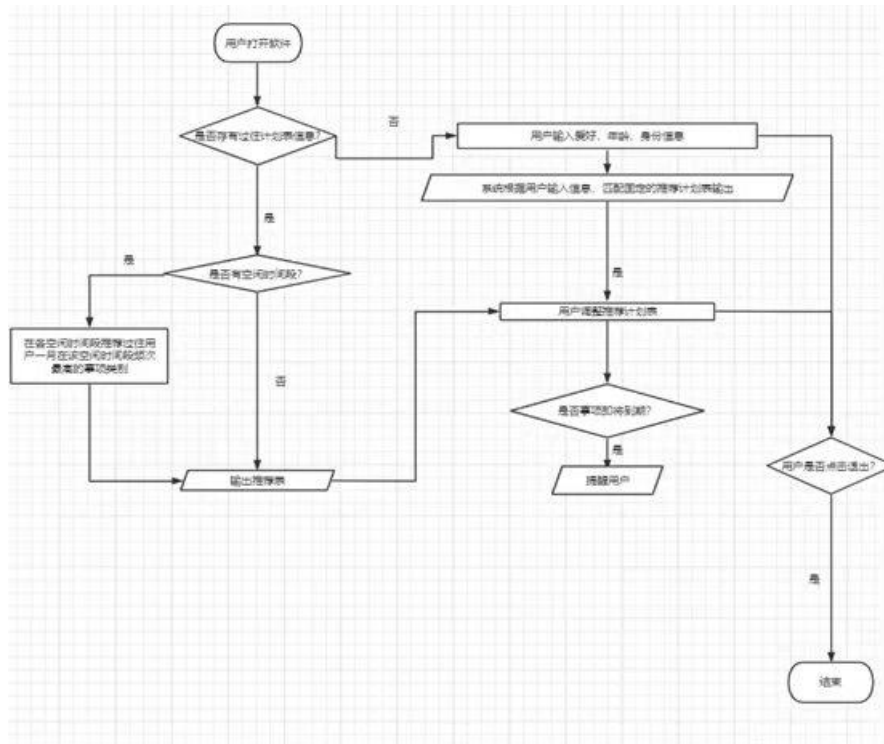
由于采用了组件化开发的方式，因此不同的组件之间需要降低耦合度，但是在课本教授的 intent 等工具中，由于各模块彼此独立，依赖关系较弱，其要实现路由跳转是比较困难的，就算实现了，性能也比较低下。因此开发小组采用了阿里的开源组件 ARouter 解决路由跳转的问题。ARouter 在编译期的时候，利用自定义注解完成了页面的自动注册，通过代码编译后，生成了相应的映射文件，初始化会将这些文件加载到内存中去，造成一个路由表，以供后面路由查找跳转之用。

同样是采用了组件化开发的原因，EventBus 代替了课本教授的广播成为了本软件各模块间信息交互的工具。广播在软件运转过程中时间与空间的开销大，而 EventBus 调度灵活，不依赖 Context，且有通知优化级，保证关注最重要的通知，同时通知当时不在场也能接收到事件，且父类对于通知的监听和处理可以继承给子类，有助于简化代码以及适配组件之间低耦合度的场景，所以被开发小组认定为软件组件化开发过程中的关键技术。

### 2.3.2 推荐算法的实现思路与遇到的困难

#### 2.3.2.1 推荐算法实现思路

推荐算法流程图如下图所示。



在冷启动阶段，用户在登录时需要输入自己的身份标签，如年龄，性别，职业等等，系统将不同的标签排列组合，每一种排列都预设了一个推荐的表格提供给用户，以解决冷启动阶段的推荐问题。

计划的推荐内容是计划的标签，即大概的事项类别，而不是推荐具体的内容，比如会推荐学习，生活，工作等等，用户也可以自定义标签，但不会推荐用户学编译原理这种具体的计划。

而用户可以根据自己的实际需要对系统的推荐计划表进行修改，后台将会记录用户过去近一个月的计划。以小时时段为单位，进行统计，将各时段（一天24个时段）在过去一个月中出现频次最高的计划推荐给用户。如果一个计划是跨时段的，则将其标签计入各时段的统计中。

经过接近一个月的打磨，则推荐的计划表就会越发拟合用户的行为习惯了。这是一个比较符合直觉且适合推荐的算法。

### 2.3.2.2 推荐算法实现过程中遇到的困难

推荐算法是本软件的一大亮点，但无奈在推荐算法的实现过程中遭遇了以下困难，开发小组难以解决。

①开发资金的问题。该问题主要集中在我们云端数据库 Bmob 的约束，其只能限定有 20 个字段，但我们的开发日常使用就已经占用了十多个字段了，而要实现精准的推荐，用户的标签量是要比较多的，但是数据库的字段限制了标签的量，导致推荐难以进行，同时数据库的容量也会有所限制，导致难以存储大数据。要想突破上述的约束，只能充钱，但显然，开发小组作为学生，经费有限。

②数据收集的问题。该问题主要产生在用户的数据收集量，我们要精准推荐，要足够的数据，但是没做到精准推荐，以我们学生的身份，很难让用户的数据量有所增长，这也是困难之一。

③后台服务器的运算速度问题。好的服务器处理大数据时是需要较高的运算性能的，但是开发人员手头并没有足够应付大量数据运算的服务器，这也掣肘了推荐算法的设计与应用。

## 2.4 用户体验记录和分析

我们向目标用户群体发放了调查问卷以及我们软件的 apk 文件，得到了如下方表格所示的反馈。

您的手机 号是？	对 APP 的 评分？	您认为 app 的哪 一项功能有待提 高？	您觉得 APP 有待提高的具体内容是什么？
13423585 282	10	挺好的，没感觉 有什么要提高了	
13428347 462	10	时间表管理功能	备份功能有时候会出现 bug
19124816 545	9	挺好的，没感觉 有什么要提高了	没发现什么问题
15913537 099	10	时间表管理功能	希望能导入课表
15984650 673	10	挺好的，没感觉 有什么要提高了	
19120332 275	9	时间表管理功能	能批量删除就好了
17725669 976	8	时间表管理功能	如果我的时间表能够在不同的设备都出现就好了

15218627059	9	挺好的，没感觉有什么要提高的	很好哈哈
13794947424	10	挺好的，没感觉有什么要提高的	
18927164368	9	UI 外观	希望登录界面的 UI 能够增加更多的元素，不让它太空

## 2.5 已完成的改进和存在的问题

经过归纳与汇总，以及对个别问题的追踪提问，我们对用户反映的问题进行了归纳，并且针对各问题提出了解决方案并着手改进，结果如下。

问题 or 期待	改进情况
如果我的时间表能够在不同的设备都出现就好了	我们为此设置了账号云端备份功能，即用户可以选择将自己的账号备份到云端，再在其他设备下载回来
希望登录界面的 UI 能够增加更多的元素，不让它太空	我们把产品 Logo 和名称做了矢量放大，插了进去，提高了画面的美观程度
如果能实现批量删除就好了	目前还在改进中，计划在数据库事项表新增一个属性表示被选中，然后将用查询语句将被选中的元组进行修改，定义为被删除的状态
备份功能有时候会出现 bug，重复点击取出备份的按钮，会重复加载已经加载过的事项	已经修复了该 bug，将数据库的事项表新增一个属性表示加载成功，只有当前状态为未加载才能上云，未加载状态只有该事项{是新建默认未加载事项    该事项原本已加载但在用户本地被修改}才会出现
希望能导入课表	正在改进中，目前想到的方案是用 CV 有关的 API 将用户的课表识别并且转换为统一的格式，再录入我们的数据库，进而在前端显示

## 三、测试大纲和测试报告

### 3.1 性能测试

开发小组使用自动化测试工具 PerfDog 对 TodoList APP 进行了硬件性能上的测试。

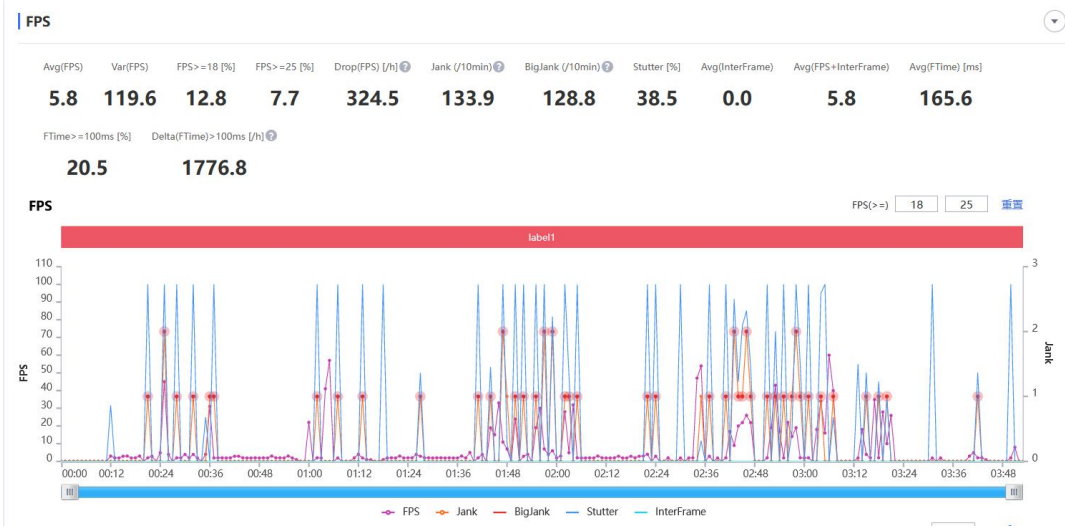
基于 TodoList 的功能，开发小组设计了针对标签设置、事件管理、备份数据到云端、登录注册等功能的测试，检查其在硬件参数上反映出来的流畅程度以及对资源的占用等性能指标。

测试硬件参数如下图所示，系统是 Android 10，CPU 是骁龙的一个中低端的芯片，闪存大小是 7.6GB。

OS	: 10
CPU Type	: bengal
CPU Info	: Qualcomm Technologies, Inc BENGAL
CPU Arch	: arm64-v8a
CPU CoreNum	: 8
CPU Freq	: 300MHz - 1804MHz 300MHz - 2016MHz
GPU Type	: Qualcomm Adreno (TM) 610
OpenGL	: OpenGL ES 3.2 V@444.0 (GIT@243b6bc
GPU Freq	: unavailable
Resolution	: 1080x2340
Screen Size	: 5.69 in
Ram Size	: 7.6 GB
LMK Threshold	: 1080MB
Swap	: 4047 MB
Root	: No
SerialNum	: 530242321120

根据下方的 FPS 数据可以看出，在 APP 运行过程中 Jank/BigJank 的比值为每 10mins 中 133.9/128.8，即说明在运行过程中，用户并不会会有明显的感到卡

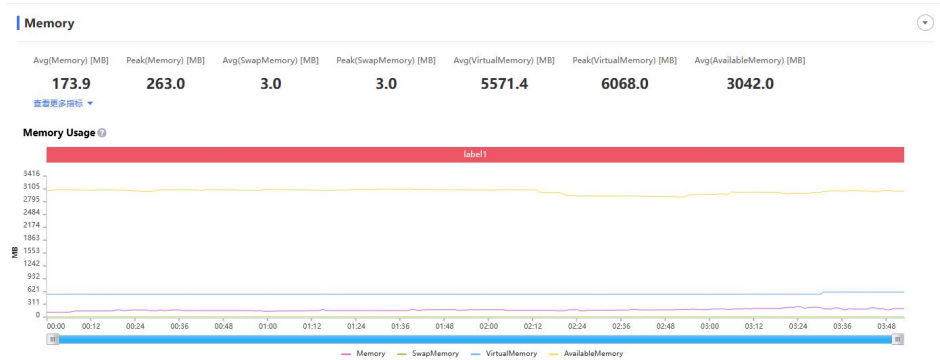
顿现象。其次 Jank 的情况基本发生在界面切换或者刷新过程中，所以并不影响用户的使用。总的来说，用户的常规使用比较流畅。



该软件的 CPU 资源占用情况如下图所示，可见本 APP 在日常使用中占用的不到 20%的 CPU 内存(当前测试已经是清除其他所有的非系统后台应用)，说明本 APP 是一个轻量级的应用，并不会对用户的其他正常 App 使用造成影响。



在整个测试过程中，APP 对内存资源的占用变动幅度不大，占用系统内存的比例也很小，这正说明了 ToDoList APP 的轻便灵巧。



### 3.2 功能测试

该软件的测试采用了黑盒测试，对于关键的功能点进行了测试，结果如下。

模块名称	功能描述	结果
用户注册	用户名未被使用，则将信息持久化，返回注册成功提示  用户名已存在，则返回用户已存在信息	通过
用户登录	效验正确则读取该用户响应的任务信息并返回  效验失败则返回错误提示信息	通过
个人信息管理	个人信息和用户头像修改	通过
任务管理	任务的增删改查（包括标签、完成进度、完成时间、开始时间等）	通过
任务查看	任务的属性可视化以表格、图表形式呈现  按照预先设定好的优先顺序对用户任务进行排序  根据当前时间高亮显示待完成任务	通过
帮助	向用户提供应用使用说明  定期更新应用的 Q&A	通过



反馈	用户可以在此向开发者提供反馈  开发人员定期读取、分析反馈信息，对 APP 进行改进	通过
----	--	----

### 3.3 测试总结

经过测试，本软件的基本功能能够交付给用户使用，且在性能上比较优越，因此可以交付。

## 四、产品安装和使用说明

### 4.1 产品安装方法

通过下载软件对应的 apk 格式文件，在文件管理系统处点击打开安装即可。

### 4.2 产品使用说明

#### 4.2.1 适用机型

Android 5.0 以上的机型

#### 4.2.2 使用方法

根据导航栏的页面分类，可以按照以下几类进行划分，用户可以在下列各页面使用对应的功能。

用户可以在“首页”页面与“计划”页面完成计划的增删改查以及排序等功能，可以在“我的”页面完成账号的登录注册退出、计划回收、标签设置、计划

云端备份、反馈、获取产品帮助文档、头像设置、昵称设置等功能。