**概要设计说明书**

**《xxx系统》**

**编写日期：2022年6月10日**

**项目组：**

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **角色** |
| **202031061495** | **何宗欣** | **组长 后端** |
| **202031061524** | **黄淳** | **组员 前端** |
| **202031061525** | **段飞** | **组员 后端** |
| **202031061526** | **江昕** | **组员 前端** |
| **202031061522** | **钟佳豪** | **组员 前端** |

**修改日志**

|  |  |  |
| --- | --- | --- |
| 修改者 | 修改日期 | 备注说明 |
| 江昕 | 2022/5/21 | 立项 |
| 江昕 | 2022/6/10 | 对接 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目 录**

[1. 引言 5](#_Toc469413310)

[1.1 编写目的 5](#_Toc469413311)

[1.2 定义 5](#_Toc469413312)

[1.3 参考资料 5](#_Toc469413313)

[2. 范围 5](#_Toc469413314)

[2.1 系统主要目标 5](#_Toc469413315)

[2.2 主要软件需求 5](#_Toc469413316)

[2.3 设计约束、限制 5](#_Toc469413317)

[3. 软件系统结构设计 5](#_Toc469413318)

[3.1 软件体系结构 5](#_Toc469413319)

[3.1.1 软件程序结构图 5](#_Toc469413320)

[3.1.2 模块描述 5](#_Toc469413321)

[3.2 功能需求追溯 6](#_Toc469413322)

[4. 数据设计 6](#_Toc469413323)

[5. 接口设计 6](#_Toc469413324)

[5.1 用户界面设计规则 6](#_Toc469413325)

[5.2 内部接口设计 6](#_Toc469413326)

[5.3 外部接口设计 6](#_Toc469413327)

[6. 出错处理设计 6](#_Toc469413328)

# 引言

## 编写目的

本说明的目的是对高校交流论坛系统进行概要的功能说明，以便用户及项目开发人员了解产品大致的设计与实现。以下叙述将结合文字描述、数据流图、变换图等来描述高校交流论坛系统的体系结构设计、接口设计，数据设计等方面。本说明的预期使用者有需求分析人员、设计人员、开发人员、项目管理人员、测试人员和用户。

## 定义

西柚知事,微信小程序平台,校园论坛

## 参考资料

《软件工程：实践者的研究方法（本科教学版第8版）

面向对象分析与设计（UML）

# 范围

## 系统主要目标

西柚知事是一款针对年轻人的论坛类社交软件，大家可以在这里分享自己有趣的想法，以及咨询自己的烦心事或畅聊每天的趣事。现在市场上大部分的论坛类社交软件都不仅仅是帮助用户交流，解决问题，还掺杂了一些诸如推送广告、推销等的附带功能，为了解决这一问题我们推出西柚知事，主打论坛类社交，无其他附带功能，为用户提供一个纯粹的交流平台。我们希望用户可以在使用我们的产品时不被交流之外的事情烦扰。

## 主要软件需求

功能需求：

(1)主页：显示程序功能主页面。

(2)天气预报：授权地理位置查看当前所在地当日及未来3天天气状况预报。

(3)添加发帖：发布自己想要发布的信息，必须先同意《平台约定》后输入联系方式、当前年级、专业说明、选择发布的模块、选择是否匿名发送自己想要发送的信息。选择发布的模块有：失物招领、交友墙、拼车、兼职、闲置交易、相约学习、校园活动、吐槽专区、资源分享、我想要的、最新消息、问题反馈、问题解答。

(4)查看、回复、收藏信息：点击失物招领、交友墙、拼车、兼职、闲置交易、相约学习、校园活动、吐槽专区、资源分享、我想要的、最新消息、问题反馈、问题解答模块进入模块查看对应模块其他用户发布的信息，并且能够在信息下方选自实名或者匿名评论该条信息。在回复旁能够有个收藏按钮收藏信息。

(5)查看、删除自己发布的信息和回答：能够查询自己发布的信息和自己对他人发布信息的评论，并且能够自己选择删除。

(6)“我的”个人中心：查看基本资料和功能列表，可以修改昵称、头像，查看我的消息、我的收藏，软件使用反馈。

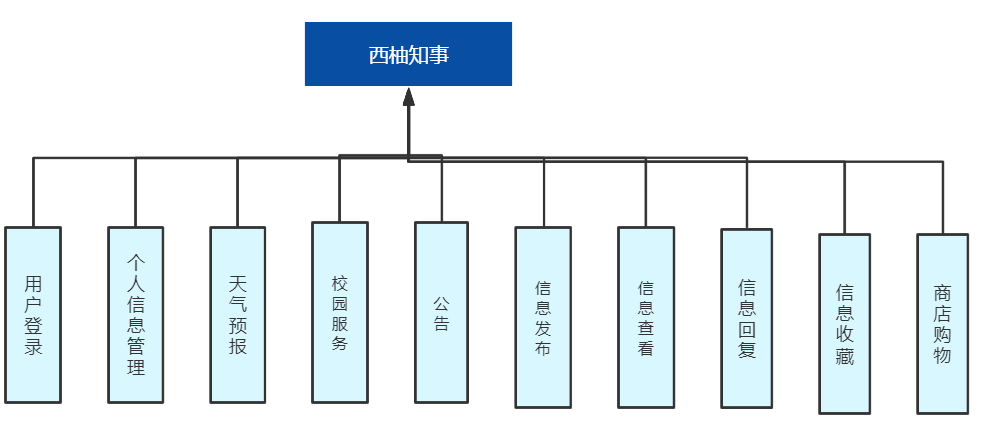
## 设计约束、限制

采用免费的开发软件，开发成本低，但运行效率受限。技术上，采用现有的小组成员掌握的技术保证项目的最终实现。环境上，在安卓手机端保证运行正确。

# 软件系统结构设计

## 软件体系结构

### 软件程序结构图



### 模块描述

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 模 块  名 称 | 用户登录 | | | 子系统名 称 | 用户注册 | 系统  名称 | 用户登录 |
| 接　口  说　明 | 输入 | Input（） | | | | | |
| 输出 | Output（） | | | | | |
| 功　能  说　明 | 用户登录管理（包括注册） | | | | | | |
| 运行环  境说明 | （1）内存：1G以上  （2）磁盘空间：50G以上  （3）CPU：1GHz以上  （4）硬盘空间：3G以上 | | | | | | |
| 调用关  系说明 | 调用模块 | | 登录，注册 | | | | |
| 被调用模块 | | 登录，注册 | | | | |

## 功能需求追溯

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 模块1 | 模块2 | …… | 模块M |
| 用户登录 | 注册 | 登录 | √ | √ |
| 个人信息管理 | 添加 | 删除 | √ | √ |
| 公告 | 编辑 | 发布 | √ | √ |

# 数据设计

用户信息

|  |  |
| --- | --- |
| 数据名称 | 数据类型 |
| 用户名 | char型 |
| 密码 | char型 |
| 昵称 | char型 |
| 等级 | int型 |
| 点赞数 | int型 |
| 帖子ID | int型 |
| 发帖用户名 | char型 |
| 帖子标题 | text型 |
| 帖子内容 | text型 |
| 帖子标签 | char型 |
| 帖子创建时间 | time型 |
| 帖子点赞数 | int型 |

# 接口设计

## 用户界面设计规则

用户登录界面：当用户访问正确的网址时，进入用户登录界面。该界面至少包含用户名输入栏和密码输入栏以便接受用户的输入，还应该有“登录”按钮使得用户可以提交用户名和密码和“注册”按钮。

注册界面：用户点击用户登录界面的“注册”按钮后可以进入注册界面。该界面应提供“用户名”、“昵称”、“密码”、“确认密码”栏供用户输入，还应该提供“提交”按钮让用户提交用户信息并完成注册。

主界面：用户提供正确的用户名和密码后，跳转到该界面。该界面根据帖子库中问题的热度，帖子的时间等信息为用户推送相关的问题。并且需要提供搜索问题的输入栏和“个人中心”的跳转按钮。

搜索结果显示界面：通过读取主界面的搜索关键字在帖子信息库中匹配相关问题，并将相关帖子显示到该界面。

查看帖子界面：用户在看到帖子后可以点击该帖子便可跳转到该界面，在该界面展示问题的详细信息，并且需要提供给用户“点赞”按钮和评论栏。

个人中心界面：用户在主界面点击个人中心跳转按钮可以跳转到该界面。该界面显示用户的昵称、签名、称号、发帖数、回答数等基本信息，并且提供给用户查看“我的发帖”、“我的回帖”、“创建帖子”、“帮助\提示”和“退出”等按钮方便用户进行相应操作。

## 内部接口设计

## 外部接口设计

### 用户接口

安卓系统 Android 4.4以上

IOS系统 IOS 11.0 以上

### 硬件接口

（1）内存：1G以上

（2）磁盘空间：50G以上

（3）CPU：1GHz以上

（4）硬盘空间：3G以上

* + 1. *软件接口*

（1）操作系统：Microsoft Windows 10

（2）软件设备：微信小程序开发工具、MySQL8.0

# 出错处理设计

6.1出错信息

用一览表的方式说朗每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。

6.2补救措施

说明故障出现后可能采取的变通措施，包括：

a.后备技术说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术；

b．降效技术说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录；

c．恢复及再启动技术说明将使用的恢复再启动技术，使 软件从故障点恢复执行或使软件从头开始重新运行的方法。

6.3 系统维护设计

说明为了 系统维护的方便而在程序内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。 各个程序之间的对应关系，可采用如下的矩阵图的形式；