



“福云”网站

需求分析文档_V3.1

一迭代记录表格

(1) 当前版本

项目	福云
文档	《软件需求说明书》--SRS
说明	V3. 1
作者	文档组
最后更新时间	2022-11-06

(2) 更新概要

版本号	更新时间	更新人员	更新摘要
V1. 0	2022. 10. 24	疏没有正七价	形成基本大纲

V2.0	2022.10.25	王首茜、温嘉骏	撰写部分内容
V2.1	2022.10.26	张静	补充项目 logo
V2.2	2022.10.26	杨磊	更新思维导图
V2.3	2022.10.30	王首茜、温嘉骏	更新项目功能
V2.4	2022.11.01	王首茜、温嘉骏	更新验收标准
V2.5	2022.11.02	徐钊、温嘉骏	更新系统体系结构
V2.6	2022.11.04	陈政浩	补充原型图
V3.0	2022.11.04	杨磊、温嘉骏	审核校验文档
V3.1	2022.11.06	陈政浩	添加目录与页码

注：本文档为需求分析阶段文档，随着产品实现的进行，文档会添加数据库设计部分的说明文档，以及对文档进行细微改动。

目录

一、介绍.....	4
1.1 引言	4
1.2 项目	5
1.3 说明	6
二、综合描述	7
2.1 项目背景	7
2.2 需求分析	7
2.3 产品功能 -- 设计类图	9
2.4 运行环境	14
三、系统体系结构设计	16
3.1 系统结构分析（包图）	16
3.2 结构设计（框架模式图）	16
3.3 接口设计	17
3.4 技术选型	19
3.5 界面模型-原型设计	20
四、验收标准	26
4.1 系统结构设计 -- 验收标准	26
4.2 功能模块 -- 验收标准	27
4.3 数据库设计验收标准	29

一、介绍

1.1 引言

1.1.1 编写目的

为明确软件需求、安排项目规划与进度、组织软件开发与测试，撰写本文

1.1.2 软件需求分析 —— 理论、目标

(1) 理论依据

软件需求分析（Software Requirement Analysis，以下简称 SRA）是研究用户需求得到的说明文档，旨在完全理解用户对软件需求的完整功能后，确认对应的软件功能实现后，建立可确认的、可验证的一个基本依据。

SRA 是一个项目的开端，也是项目实施最重要的关键点。据有关的机构分析结果表明，设计的软件产品存在不完整性、不正确性等问题 80% 以上是需求分析错误所导致的，而且由于需求分析错误造成根本性的功能问题尤为突出。因此，一个项目的成功软件需求分析是关键的一步。

(2) SRA 的主要实现目标

① 对实现软件的功能做全面的描述，帮助用户判断实现功能的正确性、一致性和完整性，促使用户在软件设计启动之前周密地、全面地思考软件需求；

② 了解和描述软件实现所需的全部信息，为软件设计、确认和验证提供一个基准；

③ 为软件管理人员进行软件成本计价和编制软件开发计划书提供依据；

(3) 具体内容要求

① 软件的功能需求，

② 软件与硬件或其他外部系统接口，

③ 软件的非功能性需求，

④ 软件的反向需求，

⑤ 软件设计和实现上的限制，

⑥ 阅读支持信息。

SRA 应尽量提供软件实现功能需求的全部信息，使得软件设计人员和软件测试人员不再需要需求方的接触。这就要求软件需求分析内容应正确、完整、一致和可验证。

此外，为保证软件设计质量，便于软件功能的休整和验证，软件需求表达应具备无歧义性，可追踪性和可修改性。

1.1.3 参考文献

- [1] 邹欣. 构建之法-现代软件工程[M]. 第三版. 北京:人民邮电出版社, 2017
- [2] [美]Roger S.Pressman 著, 郑人杰等译. 软件工程[M]. 第八版. 北京:机械工业出版社, 2017
- [3] GB-T8567-2006, 《计算机软件文档编制规范》[S]
- [4] GB-T 9385-2008. 《计算机软件需求规格说明规范》[S]

1.2 项目

1.2.1 项目标志



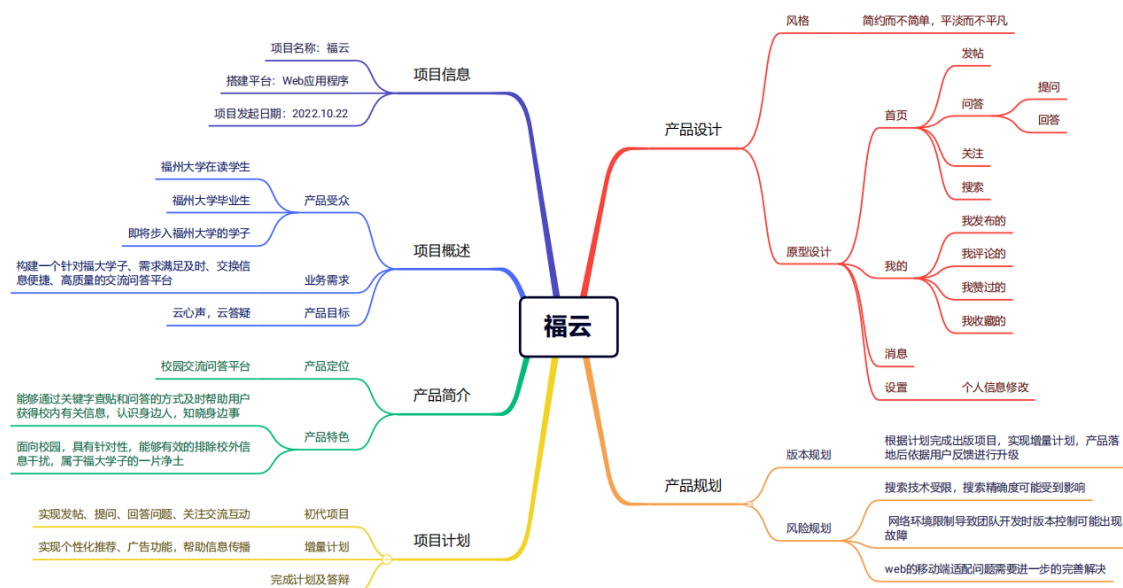
1.2.2 产品范围

本产品在综合考虑具体用户的实际情况以及可能存在的用户需求，拟为使用 Web 应用的用户提供以下服务：

- 实时分享动态
- 随便看看帖子
- 发起问题讨论
- 引用问题并发表自己的见解
- 在帖子的讨论区进行评论交流互动

1.3 说明

1.3.1 思维导图



本文档面向多种读者对象：

- (1) 项目经理：项目经理可以根据该文档了解预期产品的功能，并据此进行系统设计。
- (2) 设计员：对需求进行分析，并设计出系统，包括数据库的设计。
- (3) 程序员：配合《设计报告》，了解应用功能。
- (4) 测试员：根据本文档对软件产品进行功能性测试和非功能性测试。
- (5) 销售人员：了解预期产品的功能和性能。
- (6) 用户：了解预期产品的功能和性能，并与分析人员一起对整个需求进行讨论和协商。

(7) 其他人员：如老师、助教、其他同学等可以据此了解产品的功能和性能。

在阅读本文档时，首先要了解产品的功能概貌，然后可以根据自身的需要对每一功能进行适当的了解。

1.3.3 文档约定

(1) 页面边距

页面的左边距为 3.17cm，右边距为 3.17cm，装订线靠左。

(2) 标题最多分三级:

① 一级标题: 一、 二、 三、… (宋体加粗二号)

② 二级标题: 1.1 1.2 1.3 … (黑体加粗三号)

③ 三级标题: 1.1.1 1.1.2 1.1.3 … (黑体加粗四号)

(3) 正文

正文字体为宋体小四号, 1.5 倍行距。无特殊情况下, 字体颜色均采用黑色。

(4) 人工编号

出现序号的段落不采用自动编号功能而采用人工编号, 各级别的序号依次为 (1)、①等, 部分情况序号 1 为 (一)。

(5) 段落首行缩进 2 个字符

(6) 本文所引用的图片均采用 PNG 格式进行引用。

(7) 页码格式为“当前页/总页数”

二、综合描述

2.1 项目背景

21 世纪互联网的浪潮仍在持续冲击影响着社会生活, 我们的生活方式也在潜移默化地因之改变。目前值得关注的是网站的迅速发展, 网站作为一种获取信息的重要方式, 越来越占据了信息市场上的更大份额, 学校网站聚合社会力量, 是分配教育资源的关键桥梁, 也是学生查询资料的重要途径。

本次待开发的产品, 计划实现一个面向福大校内、基于 web 端的交流互动平台, 力在达到福大内全知全解, 为福大学子提供交互一体、无须下载、易于访问的直观界面, 帮助福大学子收获愉快的大学生活体验。

2.2 需求分析

2.2.1 Need

福州大学占地 170 万平方公里, 在校全日制学生近 5 万人, 学生与学生之间、学生与学校之间存在大量信息交换需求。目前我校信息交换主要依赖两种方式: 第一是社群模式, 学生自主创建相关需求群, 比如失物招领群等等, 这种模式最大的优点就是及时性, 但缺点在于规模小, 无论是 QQ 亦或微信群聊都有人数的限制; 第二种模式是公众号模式, 最大优点就是可大规模传播信息, 但缺乏及时性, 存在需求不被关注

的现象。

从市场规模来说，截止至 2020 年 3 月底，我国网民规模已达 9.04 亿，其中手机网民规模达 8.97 亿人，目前移动互联网人群地主要用户正是大学生。因此，在这个移动终端应用异常爆发的移动互联网时代，基于 web 端来实现服务扩展，能够有效的破解传统。如今，人们在手机上就能满足网络社交、查找信息等不同需求。在这样的发展大趋势下，学校作为最贴近大学生生活的入口平台，成为了必争之地。

基于以上情况，我们小组想构建一个针对福大学子、需求满足及时、交换信息便捷、高质量的交流问答平台——福云，一个大规模及时性信息传播工具，我们希望实现用户间的高效信息传递，实现“云心声，云答疑”相结合，并以此作为该产品的第一性原则，所有的功能后续以此为中心围绕展开。

2.2.2 Approach

通过在平台上发布帖子分享日常生活或寻求校友帮助，通过搜索关键词查帖或在他人帖子下问答获取相关校内信息，私聊模式下可深度答疑和实现社交。

2.2.3 Benefit

“福云”通过学校官方认定的账号密码登录，绑定用户信息，确认用户身份，可以实现聚焦福大学子并为其提供服务的针对性；整合校内信息资源，排除校外人员信息干扰，便捷高效；简单易学的使用方法大大降低了用户的学习成本，容易上手。

2.2.4 Competition

与“福云”类似的 App 还有网络问答社区“知乎”APP，作为面向互联网且创作者聚集的原创内容，知乎可以提供多种多样的信息，对于概念性的解释、发散思维的整合，是其一有优点。然而同样，劣势有：其一受众太广，无法对于福大学子提供校内针对性的服务；其二，及时性不够，如果没有一定量的粉丝基础就存在回应量贫瘠的现象，新人发帖无法快速回应，问题不能够得到及时解决。

2.2.5 Delivery

与学校内入驻商家合作，比如在首页设置板块投放商家宣传广告或者刊登有关招聘信息，与其他消费 App 联动，在学校内找学生做授权代理人，提高产品在学生内部的知名度。

2.3 产品功能 —— 设计类图

2.3.1 用户场景分析用户

典型用户	80 后 男 在职工作者
描述	①工作节奏快，生活压力大； ②毕业后缺少渠道了解母校信息； ③即将参加计算机院庆。
困难	传统院庆乏味枯燥，流于形式，毕业后很难了解母校有关信息，希望参加院庆时能够获得更好的体验感。

典型用户	90 后 女 在读研究生
描述	①内向慢热，不善交际； ②生活三点一线，单调刻板； ③学业压力大，交友圈小；
困难	跨专业上岸，交友渠道狭窄，苦于寻找同学一起讨论相关课题，想努力改变现状却无计可施，课业与生活问题难以解决。

典型用户	00 后 女 在读本科生
描述	①性格开朗，希望在学校里交到志同道合的新朋友； ②热爱生活，积极乐观。
困难	大一新生，对学校不甚了解，缺乏合适的渠道了解学校相关情况，希望尽快熟悉学校地理环境、建筑分布等情况。

2.3.2 功能说明

本软件主要有以下功能，随着 Web 应用的更新，功能会有所更改

(1) 登录功能

- 1) 用户可以使用已有的初始账号密码进行登录
- 2) 用户可以在个人主页更新联系方式和个人信息

(2) 关注功能

- 1) 已登录的用户可以关注感兴趣的用户，

(3) 发布帖子

- 1) 用户可以在平台上发帖/回帖进行分享讨论
- 2) 用户可以在发帖时可以设计标签，以便高效检索

(4) 提出问题和解答问题功能

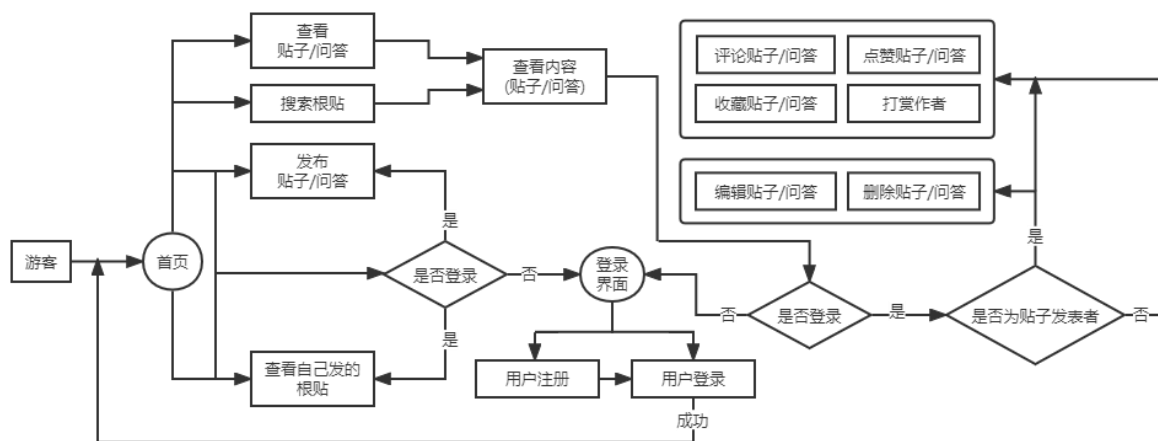
- 1) 用户可以发表问题等待回答
- 2) 用户可以对提出的问题进行回帖解答

(5) 对帖子进行点赞、评论以及打赏

- 1) 已登录的用户可以对感兴趣的帖子进行点赞，或进行评论留下自己的想法。
- 2) 若用户觉得解答帖子对自己有帮助，可选择打赏功能回报解答者

(6) 首页搜索与推荐

- 1) 用户与游客在首页可通过搜索功能查找感兴趣的帖子和问题
- 2) 首页内容可根据用户喜好与热度进行排序展示



2.3.3 用例描述

1. 用例：登录

主要参与者：用户

目标：让用户通过已有的初始账号密码登录本 Web 应用

前提条件：通过浏览器进入 Web 应用

触发器：用户想要登录该 Web 应用并进行使用

场景：用户：进入该 Web 应用，输入账号密码进行登录

异常：无法打开 Web 应用：手机运行或网络异常，解决手机本身问题

优先级：必须实现

使用频率：首次登录或者重新登录

使用方式：通过控制面板的接口

2. 用例：关注

主要参与者：用户

目标：让用户将自己感兴趣的用户添加到自己的关注列表

前提条件：用户已进行授权登录

触发器：用户想之后能快速找到自己感兴趣的用户

场景：

（1）用户：浏览帖子时对某一用户感兴趣

（2）用户：点击关注按钮将该用户添加到自己的关注列表

异常：关注失败：检查登录状态以及网络是否出现问题

使用频率：一天若干次 ~ 几天一次

使用方式：通过功能按钮的接口

3. 用例：发布帖子

主要参与者：用户

目标：让用户通过发帖的方式分享自己想分享的事

前提条件：用户已登录 Web 应用并完成信息完善

触发器：用户想要分享自己的日常

场景：

（1）用户：点击首页的发布按钮

（2）用户：添加标签并编辑帖子内容后，点击“发布”选项进行发布

异常：

（1）无法打开该板块：检查网络是否出现问题，用户是否登录成功

（2）发布帖子失败：检查网络是否出现问题

使用频率：一天若干次 ~ 几天一次

使用方式：通过功能按钮的接口

4. 用例：提出问题和解答问题

主要参与者： 用户

目标： 让用户提出疑问并相互交流解答疑问

前提条件： 用户已登录 Web 应用

触发器：

- (1) 用户想发布问题寻求他人解答
- (2) 用户想要解答他人发起的问题

场景：

- (1) 用户：编辑问题详情进行发布
- (2) 用户：对应问题下编辑解答帖进行发布

异常：

- (1) 无法打开该模块：检查网络状态是否出现问题
- (2) 无法发布问题或解答帖：检查登录状态或网络是否出现问题

使用频率： 一天若干次 ~ 几天一次

使用方式： 通过功能按钮的接口

5. 用例：帖子点赞、评论以及打赏

主要参与者： 用户

目标： 用户发表对帖子的看法以及对帖子的认可

前提条件： 用户已登录 Web 应用且正在浏览某一帖子

触发器： 用户想要对当前帖子发表自己的看法

场景：

- (1) 用户：评论区输入评论内容进行发表
- (2) 用户：点击“点赞”按钮进行点赞
- (3) 用户：点击“打赏”按钮跳出帖子作者的第三平台收款码

异常：

- (1) 无法进行评论以及点赞功能：检查登录状态以及网络是否出现问题
- (2) 点击打赏按钮无反应：帖子作者未开通打赏功能

使用频率： 一天若干次 ~ 几天一次

使用方式： 通过功能按钮的接口

6. 用例：搜索

主要参与者： 用户

目标： 让用户更快的定位到自己想了解的内容

前提条件： 用户打开 Web 应用

触发器： 用户有明确想要浏览的帖子内容

场景：

(1) 用户：搜索框键入搜索关键词

(2) 用户：点击“搜索”按钮或键入回车，页面展示搜索内容

异常：

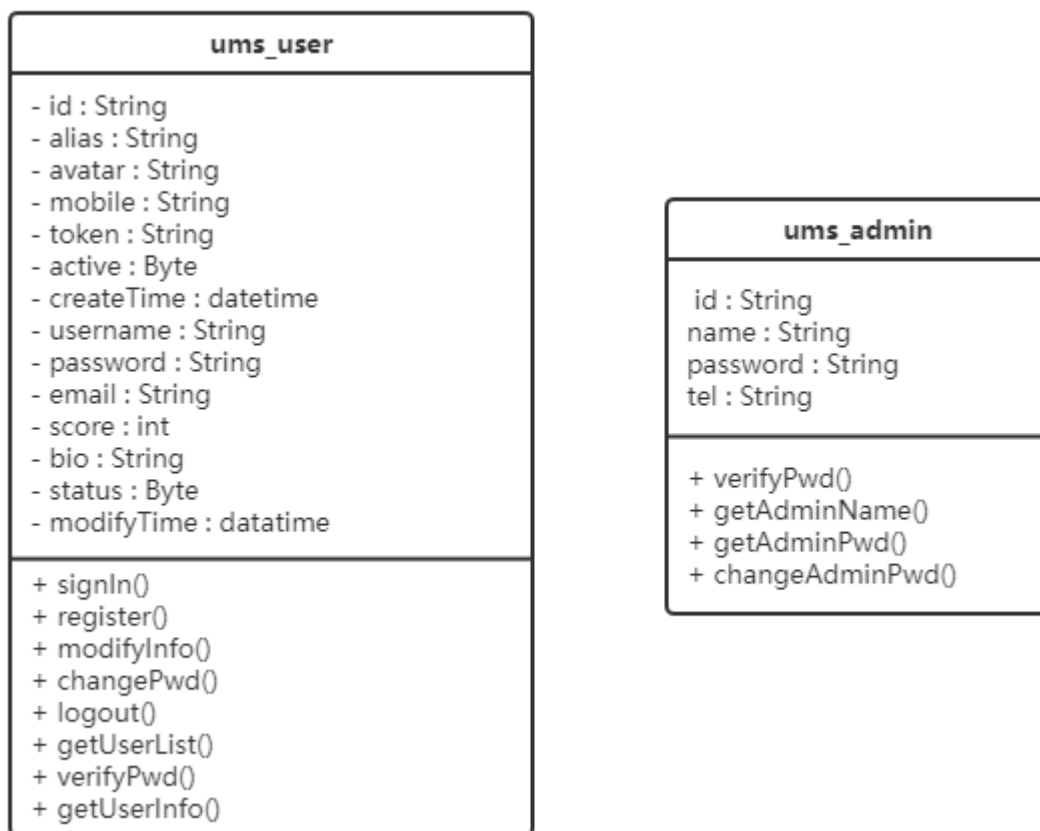
(1) 搜索结果界面加载失败：检查网络是否出现问题

使用频率： 一天若干次 ~ 几天一次

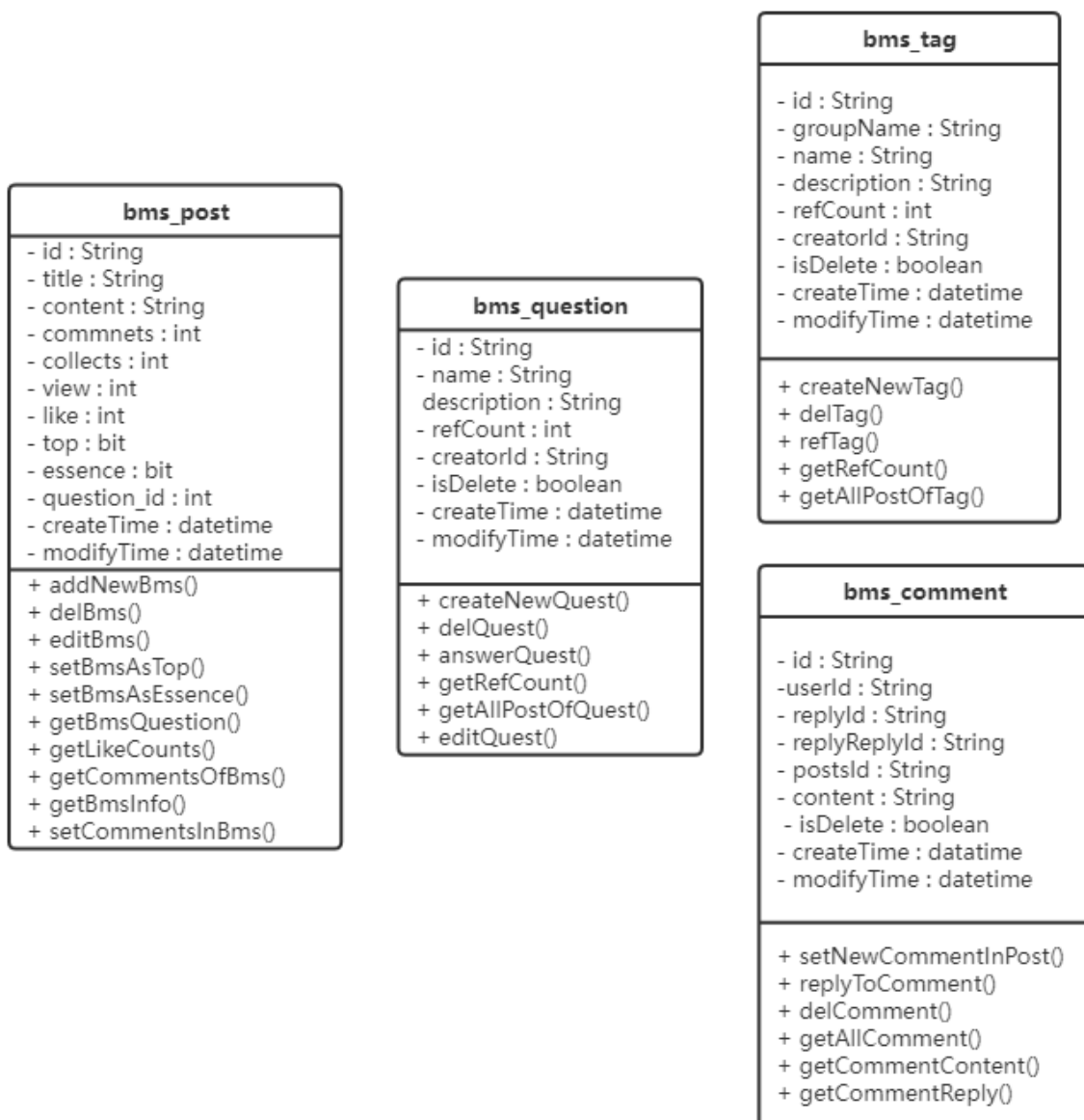
使用方式： 通过功能按钮的接口

2.3.4 类图

用户类和管理员类



其余相关类



2.4 运行环境

2.4.1 软、硬件要求

运行环境:

硬件设备: 手机/PC

软件环境: Windows/Mac OS/Linux/Android

2.4.2 设计与实现约束

(1) 开发环境约束

网络环境限制导致团队开发时版本控制可能出现故障; web 的移动端适配需要进行

一定的调整

(2) 开发周期较短

鉴于总的开发时间较短，需要进行合理的时间规划，以保证产品的准时交付。

(3) 所采用的方法与技术有限

由于对搜索引擎接触了解较少，对于产品的搜索功能模块采用模糊查询实现，效率和精确度可能会有所受限。没有购买服务器的预算，部署环境配置较低，当多用户访问网站进行的多请求高并发场景下大多是带宽瓶颈，故本项目不采用分布式微服务以及 k8s 进行部署。

(4) 采用一定的代码规范约束

为了保证代码能够顺利整合，开发之前需要制定一定的代码规范约束。

2.4.3 假设与依赖

(1) 假设

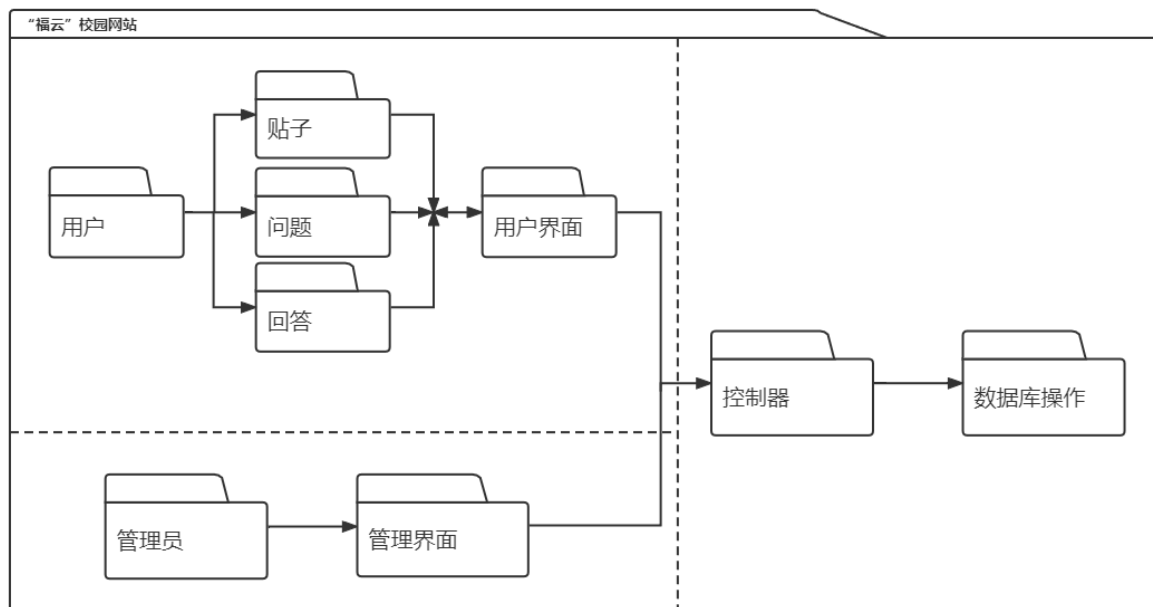
- 1) 可操作性：假定使用本 Web 应用的用户在经过一段时间熟悉之后，可以灵活地操作本 Web 应用来满足自己的需要。
- 2) 用户支持：假定在 Web 应用在开发的各个环节中得到用户的有效支持和配合。
- 3) 技术支持：假定开发初期，小组成员充分认识本系统的需求，认真学好相关知识。开发过程中遇到技术问题，可以及时得到其他同学或者老师的指导与帮助。
- 4) 人员配合：假定小组主要成员基本不会出现变动，并且在项目开发过程中不会因为突发情况的发生而导致项目成员无法正常参与开发工作。
- 5) 时间限定：假定项目的截止时间不会提前。
- 6) 需求限定：假定项目需求基本确定之后，不会有太大改变。

(2) 依赖

本产品的使用率取决于官方发布平台的可靠性、权威性、安全性，对于发布平台具有很强的依赖性。院或校官方平台对该软件的重视程度，即给予该软件的曝光率，很大程度影响了本产品的使用率。同时，校园网带宽的局限性以及用户注册率也会影响产品的体验感和使用率。

三、系统体系结构设计

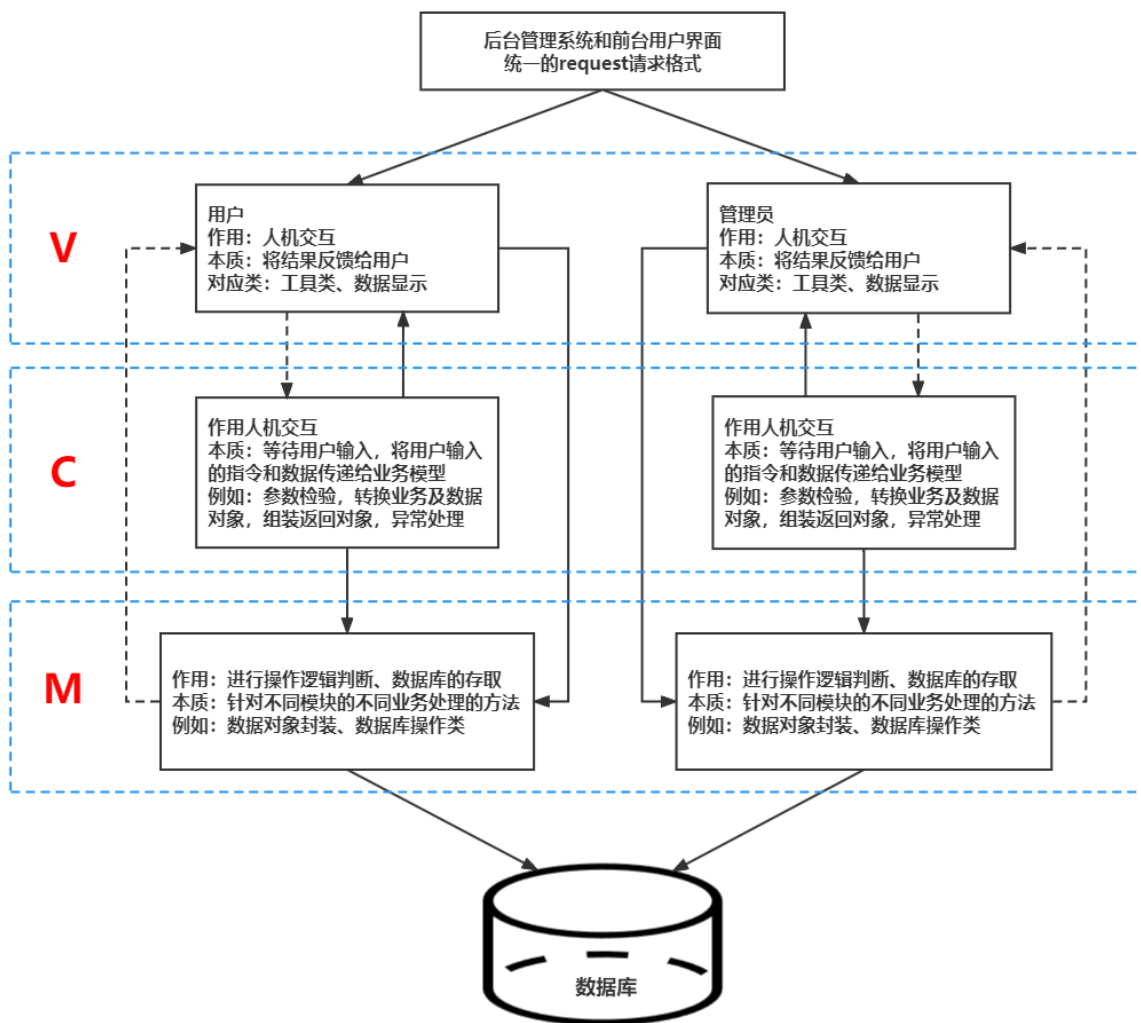
3.1 系统结构分析（包图）



包图：一种维护和描述系统总体结构模型的重要建模工具。对复杂系统进行建模时，通常需要处理大量的类、接口、构件、节点和图（跟书包中的书和文具类似），这就有必要将这些元素进行分组，即把语义相近并倾向于同一变化的元素组织起来加入同一个包中，以方便理解 and 处理整个模型

3.2 结构设计（框架模式图）

在结构设计方面，我们决定采用 MVC 框架模式。MVC 采用单一入口模式进行项目部署和访问，准确处理好模块与模块之间的联系。针对 MVC 架构对类和操作进行分析，得到的分析结果如下图：



3.3 接口设计

3.3.1 URL 设计

(1) 动词 + 宾语

客户端发出的数据操作指令都是“动词 + 宾语”的结构。比如，GET /articles 这个命令，GET 是动词，/articles 是宾语。动词通常就是五种 HTTP 方法，对应 CRUD 操作。根据 HTTP 规范，动词一律大写。

(2) 动词的覆盖

有些客户端只能使用 GET 和 POST 这两种方法。服务器必须接受 POST 模拟其他三个方法（PUT、PATCH、DELETE）。这时，客户端发出的 HTTP 请求，要加上 X-HTTP-Method-Override 属性，告诉服务器应该使用哪一个动词，覆盖 POST 方法。

(3) 宾语必须是名词

宾语就是 API 的 URL，是 HTTP 动词作用的对象。它应该是名词，不能是动

词。

(4) 复数 URL

为了统一起见，建议都使用复数 URL，比如 GET /articles/2 要好于 GET /article/2

(5) 避免多级 URL

除了第一级，其他级别都用查询字符串表达。

3.3.2 状态码

(1) 状态码必须精确

客户端的每一次请求，服务器都必须给出回应。回应包括 HTTP 状态码和数据两部分。

HTTP 状态码就是一个三位数，分成五个类别

(2) 2xx 状态码

200 状态码表示操作成功。POST 返回 201 状态码，表示生成了新的资源；DELETE 返回 204 状态码，表示资源已经不存在。202 Accepted 状态码表示服务器已经收到请求，但还未进行处理，会在未来再处理，用于异步操作。

(3) 3xx 状态码

API 用到的 3xx 状态码，主要是 303 See Other，表示参考另一个 URL。它与 302 和 307 的含义一样，也是“暂时重定向”，区别在于 302 和 307 用于 GET 请求，而 303 用于 POST、PUT 和 DELETE 请求。收到 303 以后，浏览器不会自动跳转，而会让用户自己决定下一步怎么办。

(4) 4xx 状态码

4xx 状态码表示客户端错误，主要有下面几种。

400 Bad Request：服务器不理解客户端的请求，未做任何处理。

401 Unauthorized：用户未提供身份验证凭据，或者没有通过身份验证。

403 Forbidden：用户通过了身份验证，但是不具有访问资源所需的权限。

404 Not Found：所请求的资源不存在，或不可用。

405 Method Not Allowed：用户已经通过身份验证，但是所用的 HTTP 方法不在其权限之内。

410 Gone：所请求的资源已从这个地址转移，不再可用。

415 Unsupported Media Type：客户端要求的返回格式不支持。比如，API 只能返回 JSON 格式，但是客户端要求返回 XML 格式。

422 Unprocessable Entity : 客户端上传的附件无法处理, 导致请求失败。

429 Too Many Requests: 客户端的请求次数超过限额。

(5) 5xx 状态码

5xx 状态码表示服务端错误。一般来说, API 不会向用户透露服务器的详细信息, 所以只要两个状态码就够了。

500 Internal Server Error: 客户端请求有效, 服务器处理时发生了意外。

503 Service Unavailable: 服务器无法处理请求, 一般用于网站维护状态。

3.3.3 服务器回应

(1) 不要返回纯文本

API 返回的数据格式, 不应该是纯文本, 而应该是一个 JSON 对象, 因为这样才能返回标准的结构化数据。所以, 服务器回应的 HTTP 头的 Content-Type 属性要设为 application/json。

客户端请求时, 也要明确告诉服务器, 可以接受 JSON 格式, 即请求的 HTTP 头的 ACCEPT 属性也要设成 application/json

(2) 发生错误时, 不要返回 200 状态码

状态码反映发生的错误, 具体的错误信息放在数据体里面返回。

(3) 提供链接

在回应中, 给出相关链接, 便于下一步操作。这样的话, 用户只要记住一个 URL, 就可以发现其他的 URL。这种方法叫做 HATEOAS。对于用户来说, 不需要记住 URL 设计, 只要从 api.x.com 一步步查找就可以了。

3.4 技术选型

3.4.1 前端

分析

本次待开发的应用兼顾 PC 端与手机端, 其主要以 webkit 内核为主, 对于 HTML5 等新技术支持的更好。同时, 移动端存在网络质量变化大的问题, 不适合大型的框架, 故选择较为轻量级的 Vue 框架。由于小组成员接触 JavaScript 较多, 且 Vue 对 JavaScript 支持较好, 故语言选择 JavaScript

技术选型

- HTML

- CSS
- TypeScript/JavaScript
- Vue

3.4.2 后端

分析

由于小组成员大多对 Java 掌握较为熟练，且为了方便进行单元测试，故后端技术栈选用 Springboot 作为开发框架，并整合 Mybatis 作为持久层框架，免除了传统 JDBC 繁杂的设置参数和获取代码的工作，将开发重心放在具体功能实现上。

技术选型

- Java
- SpringBoot
- Mybatis

3.5 界面模型-原型设计

“福云”网站原型图如下：（采用 Axure 制作）

首页



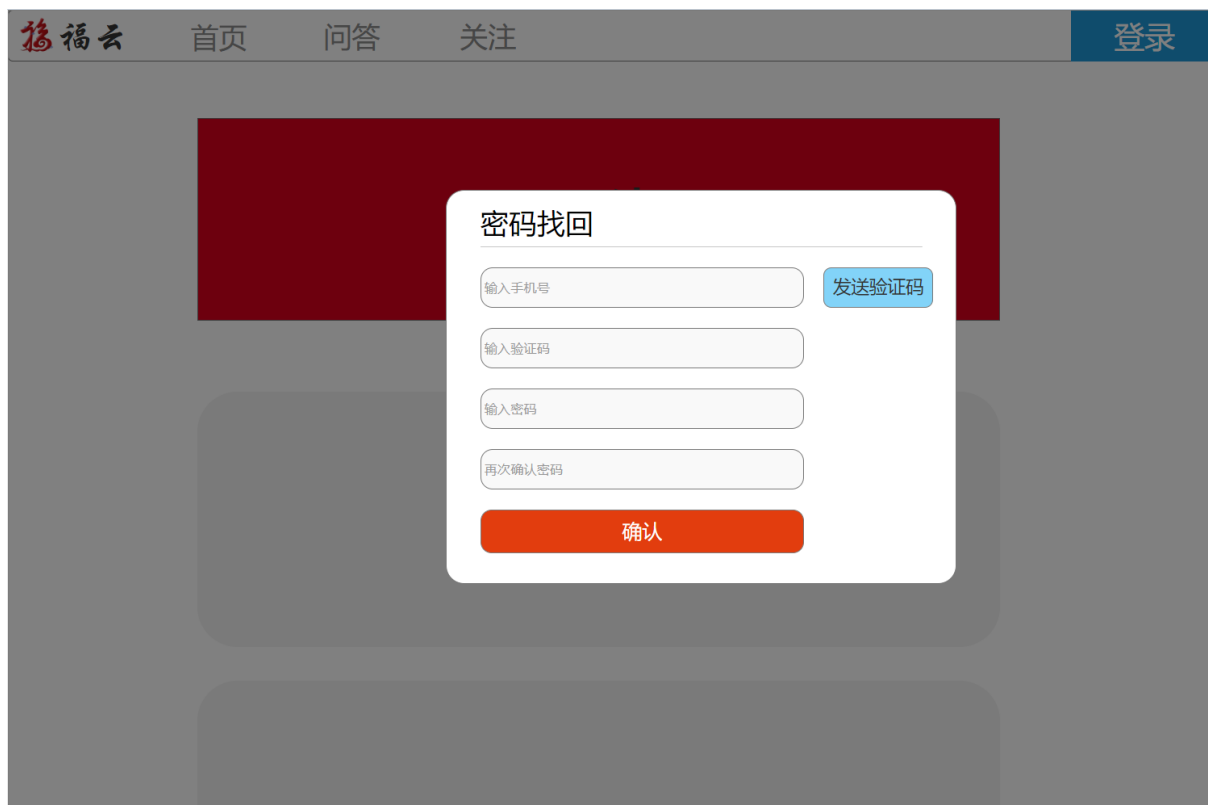
登录界面



注册页面



忘记密码页面



问答页面



问答回答帖子页面



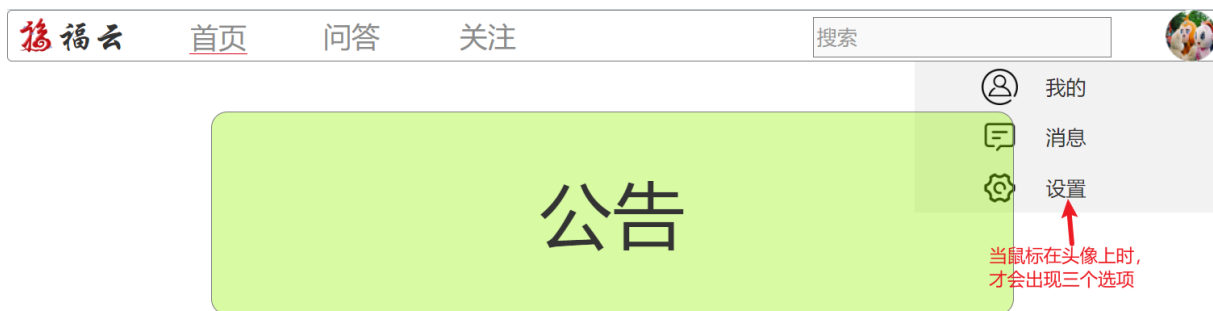
帖子详情页面



关注作者和评论的功能要登录后实现，所以在未登录时，点击会到登录页面。

下面为登录多出来的页面

登录后主页



编辑问题内容的页面



分享详情页面

福云

首页

问答

关注

搜索



如何变聪明捏?

 大聪明

 八天前

 10087

 1023

脑筋急转弯

多读书，多看报，少吃零食，多运动

 1111



 小笨蛋

我悟啦!!!!!!

2022-02-01 12:12:12

 回复



大聪明

关注TA

作者的其他文章 [更多](#)

一百块怎么活过一周教程 2022-01-10

常用标签

脑筋急转弯

我的页面

福云

首页

问答

关注





计小猿

粉丝: 555 关注: 30 获赞: 3000

Ta很懒，还没有签名哦~

我发布的

我评论的

我赞过的

我收藏的

设置页面

设置头像



修改用户名

输入新的用户名

确认

设置个性签名

消息页面

我的消息

全部

评论

@ 我

点赞

通知

四、验收标准

4.1 系统结构设计 -- 验收标准

4.1.1 满足 MVC 设计模式

MVC 设计模式是将整个系统划分为

(1) 表现层 (Presentation layer):

包含表示代码、用户交互 GUI、数据验证。该层用于向客户端用户提供 GUI 交互，它允许用户在显示系统中输入和编辑数据，同时系统提供数据验证功能。

(2) 业务逻辑层 (Business layer):

包含业务规则处理代码，即程序中与业务相关专业算法、业务政策等等。该层用于执行业务流程和制订数据的业务规则。业务逻辑层主要面向业务应用，为表示层提供业务服务。

(3) 数据持久层 (Persistence layer):

包含数据处理代码和数据存储代码。数据持久层主要包括数据存取服务，负责与数据库管理系统（如数据库）之间的通信。

三个层次的每一层在处理程序上有各自明确的任务，在功能实现上有清晰的区分，各层与其余层分离，但各层之间存有通信接口。

4.2 功能模块 -- 验收标准

(1) 用户身份操作

测试功能	测试项	输入/操作	检验点	测试功能
用户身份认证	账号密码输入	文本输入	输入限制	输入合法
	登录跳转	点击	身份确认与页面呈现	页面跳转正确
	功能菜单	选择操作方式	功能确认，结果显示	页面跳转正确
用户信息完善	联系方式	11 位阿拉伯数字	输入限制	输入合法
	头像	文件上传	结果显示	页面提示成功且成功上传

(2) 关注操作

测试功能	测试项	输入/操作	检验点	测试功能
------	-----	-------	-----	------

查看用户主页	进入用户主页	点击	跳转	页面跳转正确
关注用户	将用户添加至关注列表	点击	结果显示	页面提示成功
查看关注列表	进入关注列表	点击	跳转	页面跳转正确

(3) 发布帖子

测试功能	测试项	输入/操作	检验点	测试功能
进入发布模块	进入编辑页面	点击	跳转	页面跳转正确
编辑帖子	编辑标题	键入文本	输入限制	输入合法
	编辑内容	键入文本	输入限制	输入合法
	选择标签	点击	结果显示	提示成功
标签管理	添加标签	键入	输入限制	输入合法
发布	发布帖子	点击	结果显示、跳转	页面提示成功且跳转正确

(4) 问题模块

测试功能	测试项	输入/操作	检验点	测试功能
查看问题	进入问题详情	点击	跳转	页面跳转正确
	阅读问题详情	滑动	页面呈现	页面显示正确
发布问题	进入编辑页面	点击	跳转	页面跳转正确
	编辑标题	键入文本	输入限制	输入合法
	编辑详情	键入文本	输入限制	输入合法
回答问题	进入编辑页面	点击	跳转	页面跳转正确
	编辑标题	键入文本	输入限制	输入合法
	编辑内容	键入文本	输入限制	输入合法

	选择标签	点击	结果显示	提示成功
--	------	----	------	------

(5) 评论、点赞及打赏

测试功能	测试项	输入/操作	检验点	测试功能
评论	编辑评论内容	键入	输入限制	输入合法
	发表评论	点击	页面呈现、结果显示	页面提示成功且显示正确
点赞	点赞帖子	点击	结果显示	页面提示成功
打赏	打赏帖子	点击	页面呈现	页面显示正确

(6) 搜索与推荐

测试功能	测试项	输入/操作	检验点	测试功能
搜索	编辑关键词	键入	输入限制	输入合法
	查看搜索结果	点击	页面呈现	页面显示正确
推荐	首页推荐内容	滑动	页面呈现	页面显示正确

4.3 数据库设计验收标准

4.3.1 数据库设计

数据库设计应该至少要满足第三范式：

第一范式（1NF）：强调的是列的原子性，即列不能够再分成其他几列。

第二范式（2NF）：首先是 1NF，另外包含两部分内容，一是表必须有一个主键；二是没有包含在主键中的列必须完全依赖于主键，而不能只依赖于主键的一部分。

第三范式（3NF）：首先是 2NF，另外非主键列必须直接依赖于主键，不能存在传递依赖。即不能存在：非主键列 A 依赖于非主键列 B，非主键列 B 依赖于主键的情况。

4.3.2 业务算法与数据流

业务算法应该充分考虑到不同用户、不同应用场景下的用户需求差异。例如，对于不同用户等级，应该给予差异化功能实现。

- (1) 业务算法应该满足场景要求；
- (2) 业务算法应该很好的体现出设计模式的数据流向。

