

WeBlog 个人博客系统需求规格说明书

1 引言

1.1 项目背景

WeBlog 个人博客系统主要用于发表个人博客,记录个人生活日常、学习心得以及技术分享等,供他人浏览、查阅、评论等。个人博客往往作为技术人员的个人知识发布管理平台,以便于记录某项技术使用教程,也有利于与同行业的从业人员进行交流。在软件开发领域,也需要这样一款博客应用,作为开发者分享和交流的平台。

本项目致力于构建一个面向计算机技术人员的个人博客平台,每个注册用户都可以发布、管理自己的博客,也可以浏览、评论他人的博客,在博客社区中分享自己的开发经验或交流相关的计算机技术。系统还包含了搜索功能和关注功能,用户可以方便地寻找到自己需要的类型的技术文章,并及时得到来自关注博主的消息。此外,由于博文采用富文本形式展现,平台还包含了文件存储的功能,用户可以将上传的图片嵌入到自己的博客中。

1.2 用户特征和水平

本软件主要面向计算机技术人员,计算机专业的学生、从业者、爱好者。编写博客的人应该有基础的 markdown 标记语言的使用技能,愿意分享技术。

2 项目概述

2.1 软件系统描述

本软件系统是一个个人博客平台,旨在帮助用户创建和管理个人博客,以供他人浏览、评论和分享。该系统旨在成为技术人员个人知识发布和管理平台,使他们能够记录技术使用教程,并与同行业的从业人员进行交流。

该软件系统的应用目标主要是方便个人博客的创建和管理,让用户可以在博客平台上自由发表、管理自己的博客,同时也能够浏览和评论他人的博客。此外,该系统还提供搜索和关注功能,使用户可以更轻松地找到自己感兴趣的博客和博主,博主发布新博文时,用户可及时通过邮件获得博主的消息。

该软件系统的作用范围主要是面向技术人员和对技术感兴趣的用户，但任何了解 markdown 标记语言的人都可以使用该系统创建和管理自己的博客。以便更好地记录和分享自己的知识和经验。

2.2 用户群体

本软件系统的最终用户特点如下：

1. 目标用户：计算机技术人员、计算机专业的学生、从业者、爱好者以及编写技术博客的人。
2. 技术水平：用户需要具备一定的计算机技术水平，包括基本的计算机操作知识、对编程语言的了解以及熟悉 Markdown 标记语言的使用。
3. 意愿程度：用户愿意分享自己的技术知识和经验，积极参与技术交流和讨论。
4. 操作人员和维护人员的教育水平和技术专长：操作人员和维护人员需要具备一定的计算机技术水平和专业知识，能够熟练运用本软件的功能和工具，解决可能出现的技术问题。
5. 预期使用频度：由于目标用户是计算机专业人士和爱好者，预期使用频度较高，尤其是在技术交流和知识分享的活动中，使用频度可能更高。

2.3 功能描述

2.3.1 用况图

项目的初步用况图如下图所示：

| | |
|------|---|
| 后置条件 | 用户看到博客的列表 |
| 基本路径 | 1. 用户第一次进入首页； 2. 用户点进某位注册用户，可以看到该用户的博客 |

| | | |
|----------------------|----------------------|-----|
| 用况编号：3 | 用况名：浏览博文 | 作者： |
| 用况描述：访客或者注册用户浏览具体的博文 | | |
| 执行者 | 访客或者注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 用户在博客列表点击某博文 | |
| 后置条件 | 用户进入博客的详情页面 | |
| 基本路径 | 用户在博客列表点击博文进入博文的详情页面 | |

| | | |
|----------------------------|---------------------------------------|-----|
| 用况编号：4 | 用况名：查询博文 | 作者： |
| 用况描述：访客或者注册用户属于关键词或标签来查询博文 | | |
| 执行者 | 访客或者注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 用户在搜索框中输入关键词或者添加标签 | |
| 后置条件 | 展示博文的列表 | |
| 基本路径 | 1.用户在搜索框中输入关键词并点击搜索； 2.用户选择标签并点击搜索 | |

| | | |
|----------------------|-----------------|-----|
| 用况编号：5 | 用况名：登录 | 作者： |
| 用况描述：注册用户输入账号和密码进行登录 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 登录界面 | |
| 后置条件 | 密码正确，登录成功，跳转到首页 | |
| 基本路径 | 首页点击登录跳转到登录页面 | |

| | | |
|------------------|---------------------------|-----|
| 用况编号：6 | 用况名：管理个人信息 | 作者： |
| 用况描述：用户更改个人的个人信息 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 用户进入自己的个人页面 | |
| 后置条件 | 个人信息修改成功 | |
| 基本路径 | 用户在个人页面修改个人信息，最后点击确定，修改成功 | |

| | | |
|----------------------------------|-----------------------------|-----|
| 用况编号：7 | 用况名：订阅或取消订阅博主 | 作者： |
| 用况描述：用户可以在别的注册用户的博客中选择订阅或者取消订阅博主 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 用户进入到别的博主的博客页面 | |
| 后置条件 | 用户订阅或者取消订阅成功 | |
| 基本路径 | 用户进入到别的注册用户的博客页面，点击订阅或者取消订阅 | |

| | | |
|----------------------|------------------------------|-----|
| 用况编号：8 | 用况名：评论博文 | 作者： |
| 用况描述：注册用户可以在博文下面进行评论 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 无 | |
| 前置条件 | 用户进入浏览博文页面 | |
| 后置条件 | 博文下方出现用户的评论 | |
| 基本路径 | 用户在博文下方的评论框中书写自己的评论，之后点击评论按钮 | |

| | | |
|----------------------------|-------------|-----|
| 用况编号：9 | 用况名：点赞、收藏博文 | 作者： |
| 用况描述：注册用户在浏览博文中可以选择点赞和收藏博文 | | |

| | |
|------|------------------------|
| 执行者 | 注册用户 |
| 相关用况 | 无 |
| 前置条件 | 用户进入浏览博文页面 |
| 后置条件 | 点赞或者收藏成功 |
| 基本路径 | 用户在浏览页面中点击点赞按钮或者收藏博文按钮 |

| | | |
|-----------------------|-----------------------|-----|
| 用况编号：10 | 用况名：创建博文 | 作者： |
| 用况描述：注册用户可以创建一个自己新的博文 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 添加标签、发布博文、设置博文权限、编辑博文 | |
| 前置条件 | 用户在编辑博文页面 | |
| 后置条件 | 用户编辑完成 | |
| 基本路径 | 用户点击编辑按钮或者发布博客按钮 | |

| | | |
|-----------------------|----------------|-----|
| 用况编号：11 | 用况名：添加标签 | 作者： |
| 用况描述：注册用户可以给编辑的博文添加标签 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 创建博文 | |
| 前置条件 | 用户在编辑博文页面 | |
| 后置条件 | 用户添加成功标签 | |
| 基本路径 | 进入到编辑博文页面，添加标签 | |

| | | |
|-----------------------|-----------|-----|
| 用况编号：12 | 用况名：设置权限 | 作者： |
| 用况描述：注册用户可以给编辑的博文设置权限 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 创建博文 | |
| 前置条件 | 用户在编辑博文页面 | |

| | |
|------|----------------|
| 后置条件 | 用户成功设置权限 |
| 基本路径 | 进入到编辑博文页面，设置权限 |

| | | |
|------------------------|------------------|-----|
| 用况编号：13 | 用况名：发布博文 | 作者： |
| 用况描述：注册用户可以给编辑好的博文进行发布 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 创建博文 | |
| 前置条件 | 用户在编辑博文页面 | |
| 后置条件 | 用户成功发布，跳转到个人页面 | |
| 基本路径 | 进入到编辑博文页面，点击发布按钮 | |

| | | |
|--------------------|--------------------------|-----|
| 用况编号：14 | 用况名：编辑博文 | 作者： |
| 用况描述：注册用户可以编辑自己的博文 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 创建博文，修改博文 | |
| 前置条件 | 用户点击创建博文 | |
| 后置条件 | 用户成功编辑博文 | |
| 基本路径 | 点击创建博文，进入到编辑博文页面，编辑自己的博文 | |

| | | |
|--------------------|--------------------|-----|
| 用况编号：15 | 用况名：管理博文 | 作者： |
| 用况描述：注册用户可以管理自己的博文 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 浏览个人博文列表，修改博文 | |
| 前置条件 | 用户进入个人页面 | |
| 后置条件 | 用户管理完成，返回个人页面 | |
| 基本路径 | 用户点击个人中心进入个人页面进行管理 | |

| | | |
|--------------------|-------------------|-----|
| 用况编号：16 | 用况名：浏览个人博文列表 | 作者： |
| 用况描述：注册用户浏览自己的博文列表 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 管理博文 | |
| 前置条件 | 用户进入个人页面 | |
| 后置条件 | 用户浏览页面完成 | |
| 基本路径 | 用户点击个人中心浏览个人的博文列表 | |

| | | |
|----------------------|-----------------------|-----|
| 用况编号：17 | 用况名：修改博文 | 作者： |
| 用况描述：注册用户可以修改已经发布的博文 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 管理博文、编辑博文 | |
| 前置条件 | 用户进入想要更改的博客详情页 | |
| 后置条件 | 用户修改完成，返回个人页面 | |
| 基本路径 | 用户点击编辑按钮进行更改，更改完毕重新发布 | |

| | | |
|---------------------|-----------------|-----|
| 用况编号：18 | 用况名：删除博文 | 作者： |
| 用况描述：注册用户删除自己所发布的博文 | | |
| 执行者 | 注册用户 | |
| 相关用况 | 修改博文 | |
| 前置条件 | 用户进入个人页面 | |
| 后置条件 | 用户删除成功自己发布的博文 | |
| 基本路径 | 用户在对应的博文中点击删除按钮 | |

3 功能需求

3.1 系统功能模型的建立

依据以上的需求，WeBlog 个人博客系统的数据源和数据潭包括：访客、注册用户。访客向系统提出搜索博文请求、浏览博文请求、注册请求等，其是数据源；同时它也从系统中获取搜索博文结果、博文内容等数据，所以也是数据潭。同理，注册用户也同时是数据源和数据潭。该系统的顶层数据流程图如图 1 所示。

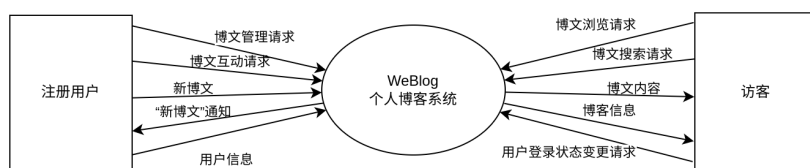


图 1 顶层数据流程图

WeBlog 个人博客系统主要是对两类实体的查询和管理：一类和博文相关，如对博文的增删改查等；另一类和用户相关，主要是对注册用户账号的管理。在顶层数据流图的基础上依照上述讨论，对顶层加工“Weblog 个人管理系统”进行分解，形成的 0 层数据流图如图 2 所示。

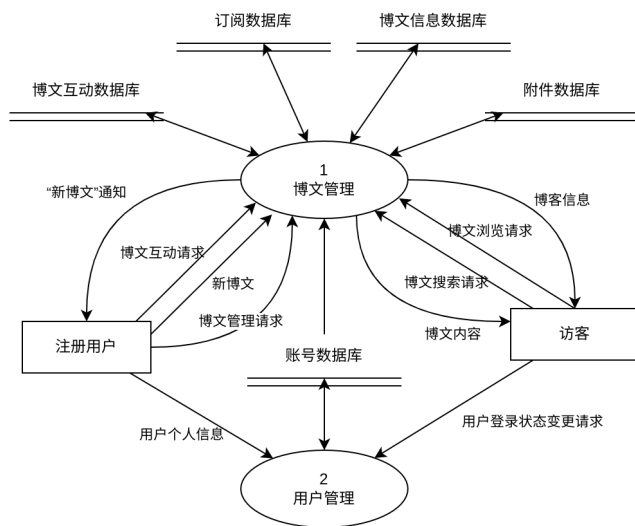


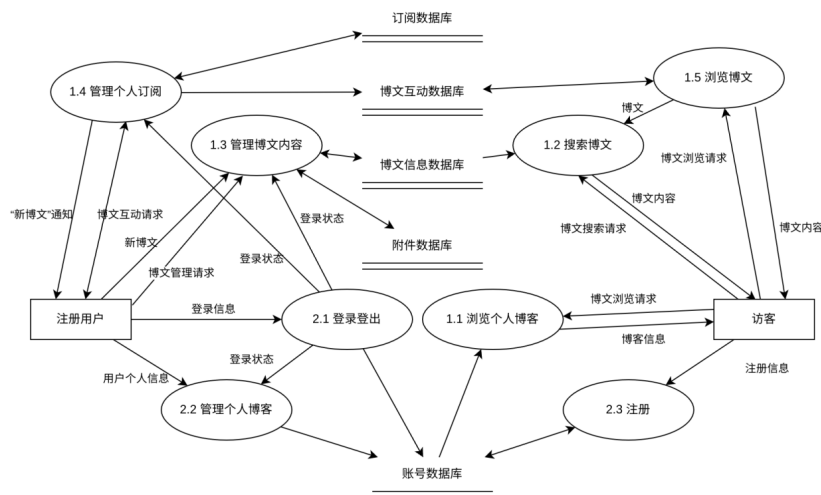
图 2 零层数据流程图

在零层数据流图中的两个加工的功能仍然不够单一且易于理解，因此可以对它们进行进

批注 [徐1]: “3 功能需求”这一部分做了完全的修改。

因为“数据流图的变换与精化”部分有额外的需要，对原来的数据流图有一定的调整，所以这里的对数据流图的描述等也跟着改了。

不过如果书中的内容没有问题的话，这里也就不会有问题。不必花时间在这部分的内容细节上。



用户信息 = 用户个人信息 + 用户账号信息
 用户个人信息 = 昵称 + 联系方式 + 邮箱 + 毕业院校
 用户账号信息 = 注册用户ID + 密码
 用户登录状态变更请求 = [注册信息 | 登录信息]
 注册信息 = 联系方式 + 昵称 + 邮箱 + 密码
 登录信息 = 联系方式 + 密码
 博文管理请求 = [博文修改请求 | 博文删除请求]
 博文修改请求 = 博文ID + 博文信息
 博文删除请求 = 博文ID
 新博文 = 博文信息
 博文信息 = 博文ID + 博文标题 + 博文简介 + 博文内容 + 注册用户ID + {标签信息} + 博文权限
 标签信息 = 标签ID + 标签名称 + 标签描述
 博文互动请求 = 博文点赞评论请求 + 博客关注请求
 博文点赞评论请求 = 博文点赞请求 + 博文评论请求
 博文点赞请求 = 点赞者ID + 博文ID

博文评论请求 = 评论者ID + 博文ID + 评论内容
博文关注请求 = 关注者ID + 被关注者ID
“新博文”通知 = 注册用户邮箱 + 博文ID + 注册用户昵称(博文作者) + 邮件标题
内容
博文搜索请求 = 搜索关键字 + {标签ID}
博文浏览请求 = 博文ID + 注册用户ID
博文内容 = 博文信息 + {评论信息} + 点赞数 + 收藏数 + 浏览数
博客信息 = 注册用户信息(博主) + {博文信息} + 点赞数 + 访问次数

3.2.2 数据存储条目

账号数据库 = {注册用户信息}
博文信息数据库 = {博文信息}
附件数据库 = {附件信息}
附件信息 = 附件ID + 附件大小 + 文件名 + 附件地址
博文互动数据库 = {点赞信息} + {评论信息}
点赞信息 = 博文ID + 注册用户ID
评论信息 = 评论ID + 博文ID + 注册用户ID + 评论内容 + 评论ID(被回复的评论)
订阅数据库 = {注册用户ID(被订阅者) + 注册用户ID(粉丝)}

3.2.3 数据项条目

| 数据项名 | 数据类型 | 意义 |
|---------|--------------------------------|------------------------|
| 和注册用户相关 | | |
| 注册用户 ID | BIGINT | 用户的唯一标识符 |
| 昵称 | 长度小于 10 的字符串 | 用于对博主个人信息的展示 |
| 联系方式 | 长度为 11 的只包含数字的字符串 | 登录的凭证 |
| 邮箱 | 长度小于 50 的字符串 | 关注的博主发新博文时将通 过此邮箱通知 |
| 毕业院校 | 长度小于 20 的字符串 | 用于对博主个人信息的展示 |
| 密码 | 长度小于 50 的字符串, 仅包含 ASCII 字 符 | 登录的凭证 |
| 和博文相关 | | |
| 博文 ID | BIGINT | 博文的唯一标识符 |
| 博文标题 | 长度小于 50 的字符串 | 在展示博文列表中用 |
| 博文简介 | 长度小于 200 的字符串 | 在展示博文列表中用 |
| 博文内容 | 长度小于 5000 的字符串 | 在查看博文内容时使用 |
| 博文权限 | “公开” 或 “不公开” | 决定博文是否可以被其他人 查询到或查看 |
| 标签 ID | BIGINT | 标签的唯一标识符 |
| 标签名称 | 长度小于 20 的字符串 | 在搜索中用到 |
| 标签描述 | 长度小于 50 的字符串 | 在查看标签信息时用到 |

| | | |
|-------|---------------|----------------------------|
| 搜索关键字 | 长度小于 20 的字符串 | 查询时用于匹配博文标题 |
| 和附件相关 | | |
| 附件 ID | BIGINT | 附件的唯一标识符 |
| 附件大小 | INT | 以比特为单位 |
| 文件名 | 长度小于 30 的字符串 | 附件的名称 |
| 附件地址 | 长度小于 100 的字符串 | 附件的 URI 地址, 访问该地址即可获取到附件内容 |

3.3 加工说明

| | |
|------|---|
| 加工编号 | 2.1 |
| 加工名 | 登录登出 |
| 功能描述 | 通过此加工，改变注册用户的登录状态； |
| 输入流 | 登录登出请求、注册用户信息 |
| 加工 | 对于登录，将访客输入的个人信息存储到数据库中；对于登录，检查访客输入的联系方式和密码是否匹配；如果匹配，则将访客的登录状态转变为已登录，否则提示失败；对于登出，将用户的登录状态设置为未登录。 |
| 输出流 | 登录状态 |

| | |
|------|------------------------------|
| 加工编号 | 2.2 |
| 加工名 | 管理个人博客 |
| 功能描述 | 注册用户管理个人信息，包括账号密码、公开信息、头像的管理 |
| 输入流 | 个人信息、登录状态 |
| 加工 | 处理来自注册用户的信息，检查后输出到账号数据库中 |
| 输出流 | 注册用户的个人信息 |

| | |
|------|----------------------------------|
| 加工编号 | 2.3 |
| 加工名 | 注册 |
| 功能描述 | 通过此加工，为访客添加个人信息，使其可以以注册用户身份与系统交互 |
| 输入流 | 注册请求、注册用户信息 |
| 加工 | 处理来自注册用户的信息，检查后输出到账号数据库中 |
| 输出流 | 注册用户的个人信息 |

| | |
|------|------------------------------------|
| 加工编号 | 1.1 |
| 加工名 | 浏览个人博客 |
| 功能描述 | 访客或注册用户进入自己或他人的博客主页，查看博主的个人信息和博文列表 |
| 输入流 | 查看博客请求、注册用户信息 |
| 加工 | 将用户信息、博文信息列表展示给访客 |
| 输出流 | 博客信息 |

| | |
|------|---|
| 加工编号 | 1.2 |
| 加工名 | 搜索博文 |
| 功能描述 | 基于规则搜索系统中的博文 |
| 输入流 | 搜索博文请求、博文信息 |
| 加工 | 根据来自访客的博的标签和标题文本，匹配博文数据库中符合要求的博文，将其准备到输出流中，输出给访客。 |
| 输出流 | 搜索博文结果 |

| | |
|------|---|
| 加工编号 | 1.3 |
| 加工名 | 管理博文内容 |
| 功能描述 | 修改博文的标题、简介、内容 |
| 输入流 | 管理博文请求、登录状态、博文信息 |
| 加工 | 根据来自访客的博的标签和标题文本，匹配博文数据库中符合要求的博文，将其准备到输出流中，输出给访客。 |
| 输出流 | 博文信息、附件信息 |

| | |
|------|---|
| 加工编号 | 1.4 |
| 加工名 | 管理个人订阅 |
| 功能描述 | 处理用户的关注请求，更新关注关系；在被关注者发布新博文时向关注者 |
| 输入流 | 关注请求、登录状态、订阅信息 |
| 加工 | 根据来自访客的博的标签和标题文本，匹配博文数据库中符合要求的博文，将其准备到输出流中，输出给访客。 |
| 输出流 | “新博文”邮件通知、订阅信息 |

| | |
|------|---|
| 加工编号 | 1.5 |
| 加工名 | 浏览博文 |
| 功能描述 | 对用户的浏览博文请求做出响应，返回博文内容 |
| 输入流 | 浏览博文请求、博文互动信息、博文信息 |
| 加工 | 对于用户的浏览博文请求，首先检查访客是否具有查看博文的权限，如果没有则告知其无权限；如果有，则将对应的博文内容返回给用户。如果用户在浏览博文中做出点赞收藏订阅等操作，将这些操作的信息输出到数据库中。 |
| 输出流 | 博文内容 |

4 非功能需求

4.1 性能需求

1. 系统必须有能力强支持 1000 个以上的并发用户
2. 系统必须有能力强支持 5000 个以上的用户注册
3. 系统不限制发布博文的字符数量
4. 系统中博客管理信息的生效时间应该应该小于 1 秒，最大生效时间应该小于 5 秒
5. 对于每个用户的访问，平均响应速度应该小于 1 秒，最大响应时间应该小于 5 秒
6. 在用户搜索文章时，系统返回搜索结果的时间应该小于 1 秒，最大时间应该小于 5 秒

为做到这一点，我们会应用到如下技术：

1. 微服务架构：微服务的架构可以方便地横向扩展瓶颈功能；
2. Redis 缓存：对于热点功能，我们可以考虑使用 redis 作为缓存，降低过多的数据库 IO 操作；
3. JMeter：性能测试工具，可用于找出性能瓶颈所在。

4.2 外部接口需求

外部接口分为系统接口、用户接口(UI)、硬件接口、软件接口、通信接口等。WeBlog 个人博客系统有如下接口需求：

通讯接口：通过网络交互，协议栈为 TCP/IP 协议和 HTTP 协议；

软件接口：本项目使用了 Mysql 数据库以及 Redis 内存数据库，需要与这两项软件应用进行数据交互；

用户接口：基于上面对软件功能需求的描述，我们设计了如下图所示的 UI：

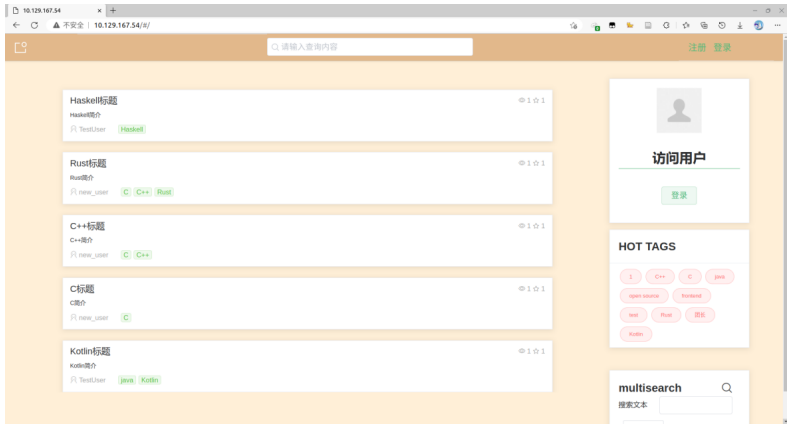


图 4 主页（未登录）

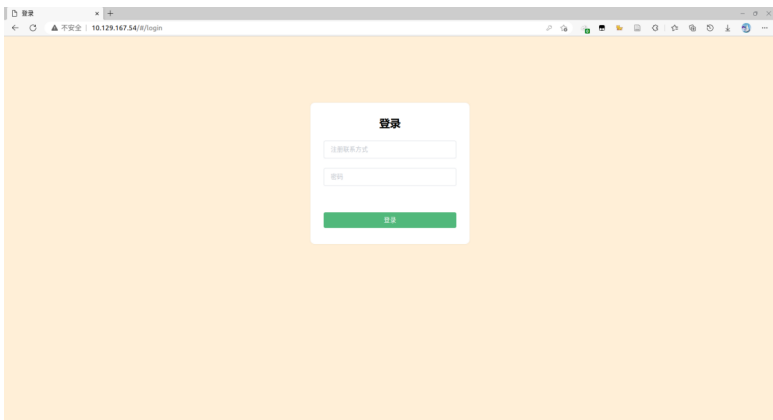


图 5 登录页面

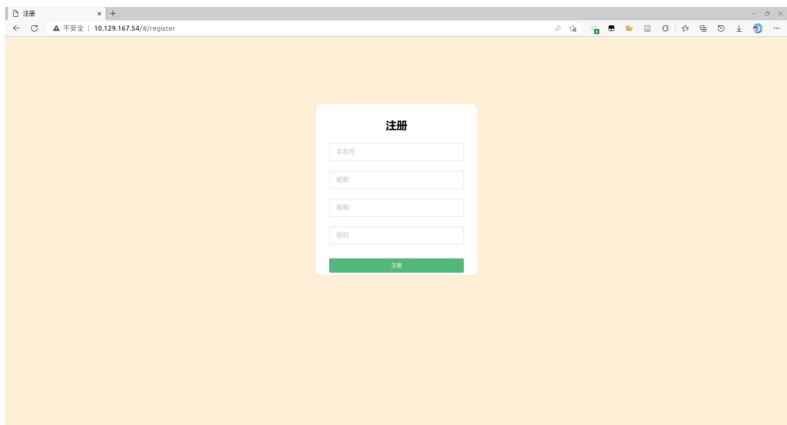
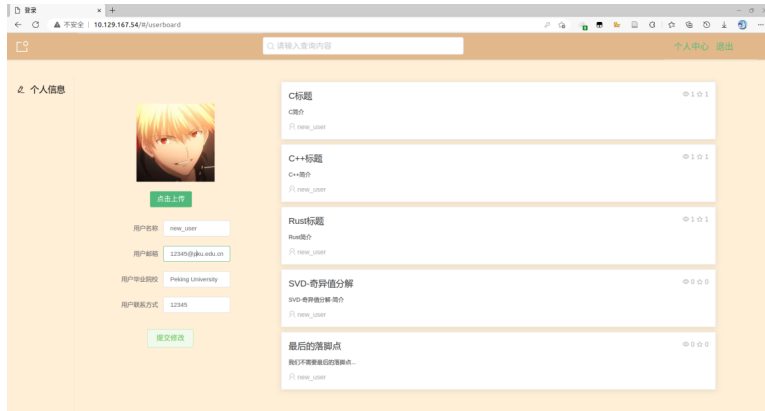
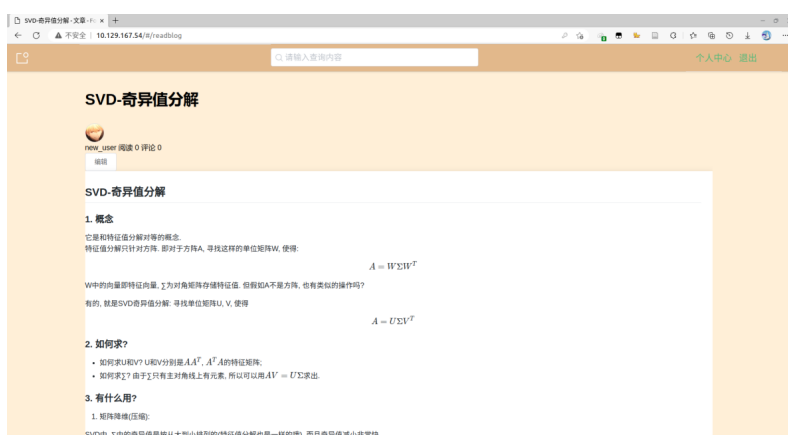


图 6 注册页面



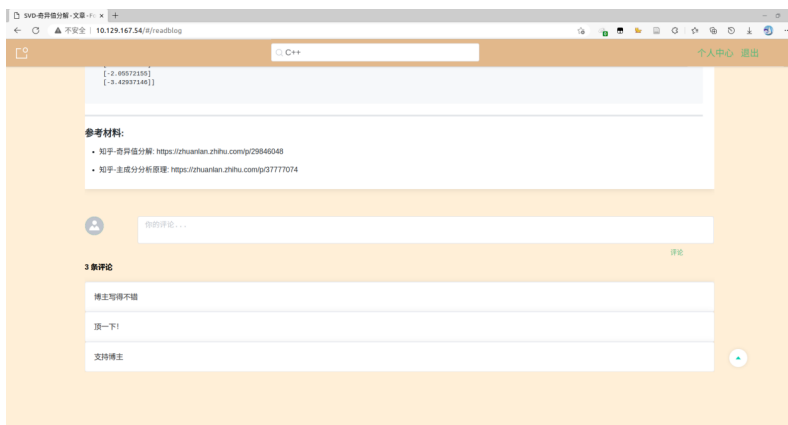
The registration page features a search bar at the top with the placeholder text "请输入查询内容". On the left, under the "个人信息" (Personal Information) section, there is a profile picture of a person with orange hair and a "点击上传" (Click to upload) button. Below the picture are input fields for "用户名" (Username) with the value "new_user", "用户邮箱" (User Email) with the value "12345@pku.edu.cn", "用户毕业院校" (User Graduation School) with the value "Peking University", and "用户联系方式" (User Contact Information) with the value "12345". A "提交修改" (Submit Modification) button is at the bottom of this section. On the right, there are five form fields for "C标题" (C Title), "C++标题" (C++ Title), "Rust标题" (Rust Title), "SVD-奇异值分解" (SVD Singular Value Decomposition), and "最后的落脚点" (The final point), each with a "new_user" placeholder and a "1 0 1" or "0 0 0" status indicator.

图 7 个人中心



The personal center page displays the user's profile and a list of articles. The profile section on the left shows the user's name "new_user", a bio "阅读 0 评论 0", and a "编辑" (Edit) button. The main content area shows a list of articles, with the first one titled "SVD-奇异值分解" (SVD Singular Value Decomposition) by "new_user". The article content includes a title "SVD-奇异值分解", a section "1. 概念" (Concept), and a section "2. 如何求?" (How to calculate?). The "1. 概念" section explains the concept of SVD and includes the formula $A = W\Sigma V^T$. The "2. 如何求?" section explains how to calculate the SVD and includes the formula $A = U\Sigma V^T$. The "3. 有什么用?" (What is it used for?) section explains the applications of SVD.

图 8 博文详情页



The blog detail page shows the full content of the article "SVD-奇异值分解". At the top, there is a code block containing the matrix $A = \begin{bmatrix} -2.49572350 \\ -3.42537148 \end{bmatrix}$. Below the code block, there is a "参考材料:" (Reference material) section with two links: "知乎-奇异值分解: https://zhuanlan.zhihu.com/p/29846048" and "知乎-主成分分析原理: https://zhuanlan.zhihu.com/p/37777074". At the bottom, there is a "你的评论..." (Your comment...) section with a "评论" (Comment) button. Below the comment section, there is a "3 条评论" (3 comments) section with three buttons: "博主写得不错" (The blogger wrote well), "顶一下!" (Like!), and "支持博主" (Support the blogger).



图 10 博文编辑页

4.3 设计约束

1. 系统设计过程中主要使用 Java 语言编写系统内容；非必要不使用其他编程语言，以免造成不同微服务之间在 Nacos 的注册问题。
2. 系统设计时需考虑在单机器部署时能够响应 1000 次/秒的首页响应请求，在多机器部署的时候能响应至少 5000 次/秒的首页响应请求。并且任取 10 秒，一个特定应用所消耗的可用计算能力平均不超过 50%。
3. 系统设计时需前后端分离且所采用技术需为业界前沿技术，架构需采用微服务架构便于后续迭代和维护，在考虑稳定性的因素下，尽可能避免使用业界过期或是存在缺陷的技术进行前后端设计。
4. 系统在部署时应部署在 Linux 环境，因而设计时必须考虑实际生产环境和开发环境不同所带来的问题并进行处理。
5. 系统在设计时需考虑用户使用问题，应当以简洁明了为主。降低用户的使用和学习成本，且重要功能主题鲜明，界面美观。
6. 系统在设计时应该考虑容灾问题，具备出现一定故障时能够自动报警或不影响其他模块正常运行，不影响数据库和缓存的准确性。

4.4 质量属性

| 属性名称 | 详细要求 |
|------|------|
|------|------|

| | |
|---------|---|
| 性能及效率 | 当系统在单个机器上部署时，使集群至少能够响应 1000 次/秒的首页请求响应；在多个节点上部署时，能够响应 5000 次/秒的首页请求响应；所有界面应该在 5 秒之内完成界面框架的加载，7 秒内填充来自后端的数据 对于搜索功能，响应时间不应超过 5 秒. |
| 易用性，清晰性 | 系统界面设计应与主流博客平台相似，降低用户迁移的学习成本；从页面设计上，界面的主次设计鲜明，主要部分展示数据，次要部分包含全面的跳转链接. 用户上手时间不应超过 5 分钟，在 15 分钟内了解绝大多数功能. |
| 可扩展性 | 后端 API 将搜索,热搜,排序等功能抽象，可以在工期足够时采用不同的算法实现功能. |
| 可移植性 | 系统采用 docker 部署，因此可以在任意 x86-64 架构的 linux 系统 (包括 WSL)节点上部署. |
| 健壮性 | 微服务节点由 nacos 管理，节点失败重启时间不应超过 10 秒；失败不应影响到持久化存储的数据，尽量避免影响到缓存数据(如 redis). |