软件需求规格说明

Version 2.0

修改历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2024-10-23 | 1.0 | 初稿 |  |
| 2024-12-26 | 2.0 | 终稿 |  |
|  |  |  |  |

目 录

[1. 简介 1](#_Toc156499902)

[1.1 目的 1](#_Toc156499903)

[1.2 范围 1](#_Toc156499904)

[1.3 定义、缩写词以及简写 1](#_Toc156499905)

[1.4 参考文献 1](#_Toc156499906)

[1.5 内容组织 1](#_Toc156499907)

[2. 综合描述 1](#_Toc156499908)

[2.1 产品前景 1](#_Toc156499909)

[2.2 产品功能 2](#_Toc156499910)

[2.3 用户特征 2](#_Toc156499911)

[2.4 一般性限制 2](#_Toc156499912)

[2.5 假设和依赖 2](#_Toc156499913)

[3. 详细需求 2](#_Toc156499914)

[3.1 功能需求 2](#_Toc156499915)

[3.2 外部接口需求 3](#_Toc156499916)

[3.3 性能需求 3](#_Toc156499917)

[3.4 质量属性 3](#_Toc156499918)

[3.5 其他需求 3](#_Toc156499919)

[4. 支持信息 4](#_Toc156499920)

# 简介

## 目的

本文档旨在为“学术直通车”软件的开发者提供全面的指导。通过介绍平台的基本信息、目标用户以及需求标准，帮助开发者深入理解该平台的核心功能与独特价值，进而设计出更加契合用户需求的解决方案，以提升学术资源的获取与利用效率，推动知识的共享与应用。

## 范围

本文档所描述的开发产品名为“学术直通车”。该平台致力于为普通用户、科研人员和开发人员提供一个全面的学术资源整合工具，打破arXiv、GitHub和公众号之间的壁垒。

对于普通用户，学术直通车提供注册与登录功能，用户可以通过绑定邮箱和手机号完成注册，便捷地访问平台各项功能。用户个人页面允许修改昵称、密码和个人简介，展示其学术画像，并能够访问其他用户的主页，了解他们的研究兴趣与贡献。此外，用户可以订阅特定领域、学者或组织，以获取个性化的学术资源推荐，并追踪热门学术动态。

在学术资源展示方面，普通用户可以查看资源的详细信息和相关讨论，获取AI生成的论文总结，帮助快速理解研究主题。用户还可收藏和分类感兴趣的资源，并查看浏览历史，便于管理和查找。

对于科研人员，学术直通车提供了浏览学术论文的推荐功能，使他们能够及时了解科研动态。同时，开发人员可以浏览相关的代码仓库，获取工业界的实践信息。

论坛管理功能允许用户发布讨论和问题，促进知识交流。用户还可以通过私信与其他用户沟通，增进互动与合作。推送管理方面，用户可以接收最新的订阅通知和论坛动态，确保不会错过重要信息。

学术直通车专注于学术资源的整合与获取，为用户提供选择和评估学术资源的有效工具，助力他们在科研与开发中高效前行。

## 定义、缩写词以及简写

1.3.1 定义

* **软件需求规格说明书（SRS）**：描述学术直通车平台的需求和规范的文档，包括功能需求、性能需求、用户界面需求等。
* **功能需求**：描述学术直通车平台需要满足的具体功能和用户行为。
* **普通用户**：指注册并使用学术直通车平台的普通个人用户，能够访问和利用平台的各项功能。
* **科研人员**：专注于科学研究，使用学术直通车平台获取最新学术论文和科研动态的用户。
* **开发人员**：关注实际应用与实现，利用学术直通车平台了解相关开源代码和工业界动态的用户。

1.3.2 缩写词及简写

* **UI**：用户界面（User Interface）
* **SRS**：软件需求规格说明书（Software Requirements Specification）
* **AI**：人工智能（Artificial Intelligence）
* **GitHub**：开源代码托管平台（GitHub）
* **arXiv**：学术论文预印本库（arXiv）
* **公众号**：微信公众号（WeChat Official Account），提供与学术相关的内容和讨论。

## 参考文献

1. arXiv 论文预印本库，[https://arxiv.org/](https://arxiv.org/#tdsub)
2. GitHub 开源代码托管平台，[https://github.com/](https://github.com/#tdsub)
3. Papers With Code，[https://paperswithcode.com/](https://paperswithcode.com/#tdsub)
4. Google Style Guides，[https://google.github.io/styleguide/](https://google.github.io/styleguide/#tdsub)
5. 中华人民共和国科学技术进步法<https://www.gov.cn/flfg/2007-12/29/content_847331.htm>
6. FAIR原则，<https://www.go-fair.org/fair-principles/>

## 内容组织

本文档将按照总分的模式组织内容。第二部分将从整体视角描述学术直通车的产品概况，包括其使用前景、可提供给用户的功能、预期效果以及平台的限制和依赖关系。第三部分将详细讲述平台的需求信息，涵盖功能需求和性能需求等内容。附录中将展示用户地图和用户故事，以帮助理解不同用户的使用场景和需求。

# 综合描述

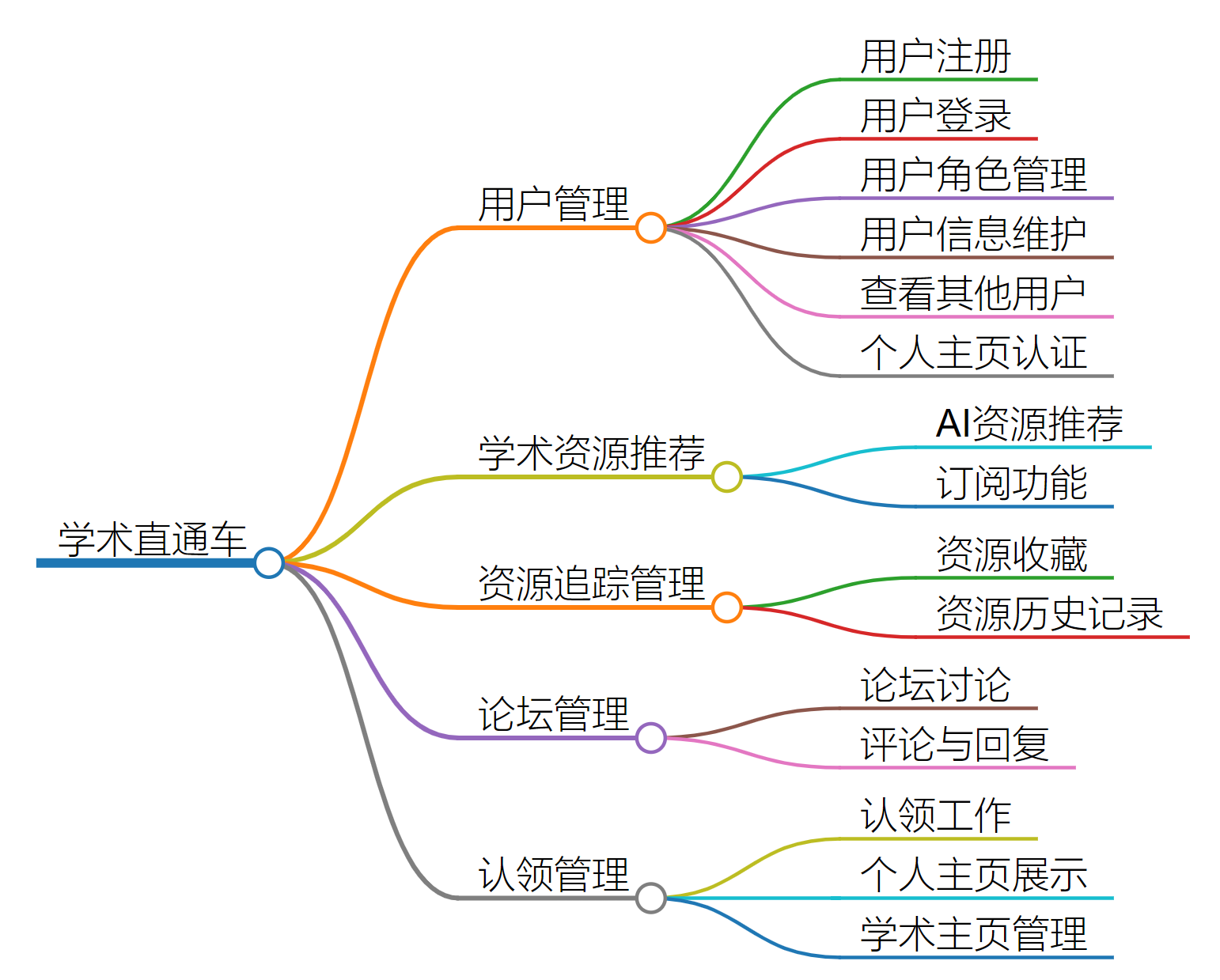
## 产品前景

在当前快速发展的学术环境中，科研工作者、开发者和知识爱好者面临着信息孤岛的挑战，尤其是在 arXiv、GitHub 和各类公众号之间。学术直通车的推出旨在打破这些壁垒，提供一个全面的平台，使用户能够便捷地获取和整合丰富的学术资源。通过高效的检索功能，用户不仅可以快速查找最新的学术论文，还能直接关联相关的开源代码仓库，甚至同步获取与研究主题相关的公众号内容，从而实现知识从理论到实践的无缝连接。

目前，市场上存在一些开放的学术资源平台，但它们往往各自为政，缺乏有效整合。许多科研工作者在寻找相关资料时，需耗费大量时间在不同平台间切换，无法高效获取所需信息。学术直通车旨在解决这一痛点，提供一个集成化的解决方案，帮助用户高效跟踪学术前沿动态，发现最新的代码实现，并参与社区讨论。我们的平台将通过智能关联论文与代码、公众号内容同步以及个性化推荐，全面提升用户的学术研究效率。

在功能上，学术直通车将提供多维度的资源检索和实时更新，确保用户不会错过任何重要的学术资源。同时，平台也将鼓励用户之间的互动与交流，通过论坛和私信功能促进知识分享和合作。我们相信，学术直通车不仅是一个资源整合的平台，更是每位追求卓越的科研人员和开发者探索知识世界的最佳伙伴。通过这一平台，用户将能够从理论研究到实践创新，轻松搭建起通往学术成功的桥梁。

## 产品功能



## 用户特征

根据用户的需求和技能水平，学术直通车的用户可分为三类：普通用户、科研人员和开发者。

1. **普通用户**

普通用户通常对学术资源的需求较高，但对计算机技术的掌握有限。他们可能来自各个行业，如金融、教育或传统工业，寻求一种简单易用的平台，以便快速获取与自己工作相关的学术论文和知识。这类用户希望通过学术直通车实现“零门槛”的访问，从而提高工作效率和质量。他们对复杂的技术细节并不关心，更看重的是平台的便捷性和易用性。

1. **科研人员**

科研人员对学术资源有更深入的理解，通常具备相关领域的专业知识。他们希望能够快速找到最新的研究成果、关联的开源代码以及相关讨论。这类用户更关注科研动态，希望通过学术直通车及时获取前沿的研究信息，促进自己的研究进展。他们需要更丰富的功能来支持自己的研究需求，比如数据分析工具和可视化功能。

1. **开发者**

开发者主要关注与学术成果相关的代码实现和应用。他们具备一定的计算机技能，能够理解和使用开源代码，且希望通过平台追踪最新的技术趋势和相关项目。学术直通车为他们提供了一个整合的环境，便于快速获取所需的学术资源和开发工具。这类用户需要更加专业和技术化的功能，以帮助他们在开发和应用学术资源时提高效率。

这样的划分确保了学术直通车能够满足不同用户的需求，促进学术资源的高效共享和利用。

## 一般性限制

#### 2.4.1 技术工具

学术直通车将以网页形式提供服务，确保用户可以方便访问。开发团队使用 Git 进行代码版本管理，并将其托管于 GitHub，以实现动态更新。同时，采用 Trello 进行任务分配和进度跟踪，确保项目高效推进。Docker 技术将用于将开发环境中的应用无缝迁移至部署环境。

在前端方面，使用 HTML、CSS 和 JavaScript 构建用户界面，具体实现将基于 Vue 框架以创建单页应用并提供响应式设计，并通过 AJAX 进行异步数据加载和交互。

后端将使用 Python 及 Django 框架，提高开发效率及安全性。数据存储方面，选择 MySQL 数据库，利用第三方库如 django-cors-headers 和 django-storages 处理用户上传的数据。用户身份验证和访问控制将使用 JWT（JSON Web Token）技术。

此外，平台将利用大模型提供的 API 接口，实现 AI 辅助的论文阅读功能。

#### 2.4.2 开发规范

遵循 Google Open Source Software Style Guides，以确保代码的一致性和规范性。开发人员将使用工具如 isort 和 autopep8 来辅助代码格式的优化。

#### 2.4.3 国家标准

本平台将严格遵循《中华人民共和国科学技术进步法》（2021年修订），确保在科研成果的传播与应用中，促进科学技术的创新、发展与普及。我们承诺维护科研诚信，避免剽窃、数据造假等不当行为，保护知识产权和科研成果的合法权益。

同时，平台将遵循《FAIR原则》，确保数据和信息的管理符合以下标准：

* 可找到性（Findable）：所有数据应具有唯一的标识符，以便用户能够快速定位所需资源。
* 可获取性（Accessible）：确保数据的可获取性，用户能够在合理的条件下访问和下载所需的数据。
* 可互操作性（Interoperable）：数据应使用通用的标准和格式，以便不同系统之间能够有效交换和使用数据。
* 可重用性（Reusable）：数据应附有适当的元数据和许可证信息，以支持其在未来的研究中被重复使用。

通过遵循这些标准，我们旨在为科研工作者、开发者和知识爱好者提供高质量、可持续的学术资源，促进学术界的合作与创新。

#### 2.4.4 硬件限制及数据转换格式

开发团队需考虑最终服务器硬件性能，设定网页的访问负载。将通过模拟测试和压力测试评估服务器能力，使用 Apache JMeter 工具监测性能指标，并考虑采用 CDN 缓存静态资源和负载均衡。

鉴于后端模型推理的时效性，前端页面的渲染将采用异步请求和懒加载技术，以降低显示延迟。数据处理将以 JSON 和 JSONL 格式为主。

#### 2.4.5 软件运行环境

为了确保安全性，开发团队需对服务器进行严格的安全配置，包括防火墙和访问控制列表。监控和日志记录系统将被设置，以便及时响应潜在问题。

部署环境将使用 Linux Ubuntu 22.04.2，Python 版本为 3.11.5，用户可通过主流浏览器如 Chrome、Firefox 和 Edge 访问平台。

## 假设和依赖

我们假设用户对学术资源的检索和使用有基本的需求，且用户对科研领域、开源代码和相关知识的兴趣较高。用户可能来自不同背景，包括科研人员、开发者和普通知识爱好者，因此他们对平台的期望和需求各异。平台依赖于用户提供的学术资源、开源项目及公众号内容来构建知识库，确保信息的丰富性和实时性。

平台的功能和推荐效果将受到用户参与度和互动质量的影响。用户在平台上活跃度的提高将促进学术资源的有效共享和社区讨论。同时，我们依赖于稳定的网络连接来提供流畅的用户体验，以确保实时信息的快速检索和更新。

总之，学术直通车旨在创建一个动态的学术资源生态系统，依托于用户的主动参与和对知识的渴求，连接不同领域的学术内容，推动知识的发现与应用。

# 详细需求

## 功能需求

1. 用户管理

1.1. 用户注册：用户可以注册账户，需提供用户名、电子邮件地址、手机号和密码。

1.2. 用户登录：已注册用户可以登录账户，以访问平台的各项功能。

1.3. 用户角色管理：系统支持多种用户角色，包括普通用户和管理员，每种角色具有不同的权限。

1.4. 用户信息维护：用户可以随时修改个人信息，包括电子邮件地址、密码和个人简介等。

1.5. 查看其他用户：用户可以在论坛中访问其他用户的主页，展示的信息包括用户名、昵称、邮箱地址（可选）、订阅的标签（可选）、个人简介等、关联的学术主页等。

1.6. 学术主页关联：用户可以通过邮箱验证等方式，将个人主页关联到 Google Scholar等学术主页，以展示更多信息。

2. 学术资源推荐

2.1. 订阅功能：用户可以选择订阅特定领域、学者或组织，系统将根据其偏好推送相关内容。

2.2. 资源推荐：基于用户的搜索历史和订阅设置，系统将推送个性化的学术资源和 Github项目。

2.3. 订阅动态：基于用户订阅的学者和组织，系统将按时间顺序列出其近期动态。

2.4. 热点追踪：基于用户的搜索历史和订阅领域，系统将推送最新的热门学术资源。

1. 学术资源展示

3.1. 资源展示：用户可以查看学术资源的详情，包括对应论文或仓库链接、资源作者、领域标签、浏览量和被引等统计数据、摘要或Readme等。

3.2. 资源相关讨论：用户可以在学术资源详情页浏览并跳转到相关的讨论。

3.3. AI 论文总结：系统使用 AI 辅助总结论文内容，包括论文要点、方法简介和实验结果等。

4. 资源追踪管理

4.1. 资源收藏：用户可以将感兴趣的论文和代码仓库等收藏至个人账户中。

4.2. 收藏资源分类：用户能够对收藏的资源进行分类管理，如按领域、重要性等进行整理。

4.3. 资源历史记录：系统记录用户对学术资源的搜索和浏览历史，并提供查看服务。

5. 论坛管理

5.1. 论坛讨论：用户可以在平台论坛发布与论文、代码和研究主题相关的讨论和问题。

5.2. 评论与回复：用户可以对论坛帖子进行评论和回复。

5.3. 私信功能：用户可以通过私信与其他用户进行一对一的交流。

6. 推送管理

6.1. 订阅通知：系统可以通过邮箱等方式定期向用户推送相关的通知。

6.2. 论坛通知：系统可以通知与用户相关的论坛讨论动态。

## 外部接口需求

**用户界面接口：**学术直通车将提供一个直观且友好的网页用户界面，支持在主流浏览器（如 Chrome、Firefox、Safari 等）上正常显示和操作。界面设计将采用响应式布局，以确保在不同屏幕尺寸和设备（包括桌面和移动设备）上提供流畅的用户体验。用户可通过简单的导航快速访问所需功能，如论文检索、代码关联和个性化推荐。

**硬件接口：**本平台不与特定硬件组件直接交互，但确保其网页能够在各种硬件设备上顺畅运行，优化以适应不同的处理能力和显示要求。

**软件接口：**

* **前端技术**：使用 HTML、CSS 和 JavaScript，结合 React 框架构建动态用户界面，以实现丰富的用户交互体验。
* **后端技术**：采用 Python 编程语言，并基于 Django 框架进行开发，以提高开发效率和安全性。
* **数据存储**：使用 MySQL 数据库存储用户信息、论文和代码相关数据，确保数据的安全性与完整性。
* **第三方集成**：与相关第三方库（如 django-cors-headers 和 django-storages）集成，以支持用户上传的文件管理和处理。
* **数据可视化**：利用 D3.js 或 Chart.js 等 JavaScript 数据可视化库展示学术资源与社区讨论的动态。
* **通信接口：**
  + 采用标准的 HTTP 和 HTTPS 协议进行前端与后端之间的数据传输，以确保安全和可靠。
  + 数据以 JSON 格式进行编码，支持多平台间的交换和解析。
  + 与第三方库和服务的通信遵循其 API 规范，确保接口的稳定性和兼容性。

## 性能需求

**用户访问响应时间**

* **需求**：平台应快速响应用户请求，以确保良好的使用体验。
* **原理**：通过优化前端加载和后端处理流程，提升页面和操作的响应速度。

**系统稳定性**

* **需求**：平台在高流量时段需保持高可用性，确保用户能够持续访问。
* **原理**：采用冗余服务器和负载均衡技术，确保服务的稳定性和连续性。

**数据处理能力**

* **需求**：平台应能高效处理并展示大量文献和相关代码的查询请求。
* **原理**：利用数据库优化和索引策略，提高数据检索和处理速度。

**内容同步速度**

* **需求**：平台需及时同步最新的公众号文章和相关研究信息。
* **原理**：通过高效的API调用和缓存机制，确保信息的更新及时。

**并发用户数**

* **需求**：平台应支持高并发用户访问，尤其在任务发布和评估高峰期，确保系统性能不受影响。
* **原理**：高并发支持依赖于优化的系统架构及强大的服务器性能。采用集群架构及数据库优化手段，能够有效管理并发请求。

## 质量属性

**可用性**

* 平台应保持高可用性，确保用户在大多数时间内能够顺利访问和操作系统。通过使用冗余设计和负载均衡技术，减少系统故障带来的影响，提供不间断的服务。

**可扩展性**

* 系统应具备良好的可扩展性，能够方便地增加新功能和模块，以适应用户需求的变化和技术的进步。这包括支持微服务架构，使得新功能的开发和部署更加灵活高效。

**安全性**

* 必须实施强有力的安全措施，防止未授权用户访问系统和敏感数据。包括用户身份验证、数据加密和安全审计等，确保数据传输和存储的安全性。

**可靠性**

* 系统应表现出高可靠性，能够在正常运行条件下稳定执行，具备快速故障检测和自动恢复的能力，以确保用户操作不受影响。

**互操作性**

* 平台应支持与多种外部系统的数据交换，以实现信息的有效整合。这包括采用标准的API协议和数据格式，使得与其他科研工具和平台的集成更加顺畅。

**可维护性**

* 系统的维护应尽量简便，以减少修复错误和进行更新的工作量。应提供清晰的文档和支持工具，使开发人员能够快速定位和解决问题。

**可移植性**

* 应能在不同的操作环境中顺利部署，且迁移过程应尽可能简单。系统应支持容器化部署，便于在不同硬件和软件平台之间的迁移和复制。

**可重用性**

* 关键模块应具备良好的重用性，以便在不同项目中重复使用。这包括将通用功能抽象成库或服务，降低新项目的开发成本。

**可测试性**

* 系统应易于测试，以确保在发现缺陷时能够高效进行评估和修复。应提供自动化测试框架，以支持持续集成和交付的流程，确保代码的质量。

**易用性**

* 用户应能够快速学习和操作平台，界面应直观易懂，降低使用门槛。通过用户反馈不断优化用户体验，使得用户能够高效准备输入、理解输出，并轻松进行操作。

## 其他需求

**国际化需求**

* 目前，学术直通车主要面向中国用户，支持中文界面。随着用户群体的扩大，系统需具备多语言支持的能力，允许用户选择适合的语言。系统应确保界面文本和功能按用户语言设置进行本地化，提供顺畅的用户体验。同时，文档和帮助资源也应支持多语言，以便用户能够获取必要的指导。

**数据库需求**

* 数据库应提供高性能的数据存储和检索，采用优化的索引结构以确保快速访问。定期实施数据库备份和恢复策略，防止数据丢失并确保数据完整性。此外，数据库需支持简单的增删改操作，以适应高频访问需求，同时具备良好的移植性与部署便捷性，便于用户灵活调整。安全权限管理至关重要，需确保未授权访问被有效阻止，并支持基于角色的访问控制。

**配置需求**

* 系统应提供用户友好的配置选项，允许管理员轻松调整系统设置、安全设置、用户角色及权限。配置界面应直观易用，以满足不同用户和组织的个性化需求，确保系统能在多种环境下平稳运行。此外，系统应支持导入和导出配置文件，以便用户能够快速备份和恢复设置。

**日志和监控需求**

* 学术直通车需生成详尽的操作日志，记录用户活动和系统事件，便于管理员进行维护和审计。监控系统应实时跟踪系统性能、资源使用情况及错误情况，及时向管理员发送警报和通知，确保问题能够快速响应和处理。同时，日志数据应支持查询和分析，以帮助发现潜在问题和优化系统性能。

**容错和修复需求**

* 系统应具备自动化的容错机制，以确保在出现故障时能够迅速恢复。应设计冗余和备份方案，以最小化用户中断时间，同时提供自我修复功能，降低人工干预的需求，提升系统的稳定性和可靠性。系统还应支持故障转移功能，当一个组件发生故障时，能够自动切换到备用组件，以保证持续的服务可用性。

**操作和管理需求**

* 平台应提供详细的用户手册和技术文档，以支持用户在操作过程中遇到问题时能快速找到解决方案。此外，管理员应具备管理用户账户、监控系统状态、配置服务和维护数据安全的工具，以确保系统高效运行。

**用户支持需求**

* 学术直通车应设有用户支持渠道，包括在线帮助、FAQ、技术支持等，以便用户在遇到问题时能够快速获得帮助。系统应支持用户反馈和建议的提交，以便不断优化和改进产品。

**性能监测需求**

* 系统应集成性能监测工具，实时分析系统负载、响应时间和资源使用情况，以确保系统始终处于最佳状态。定期生成性能报告，帮助团队识别瓶颈和优化方向。

# 支持信息

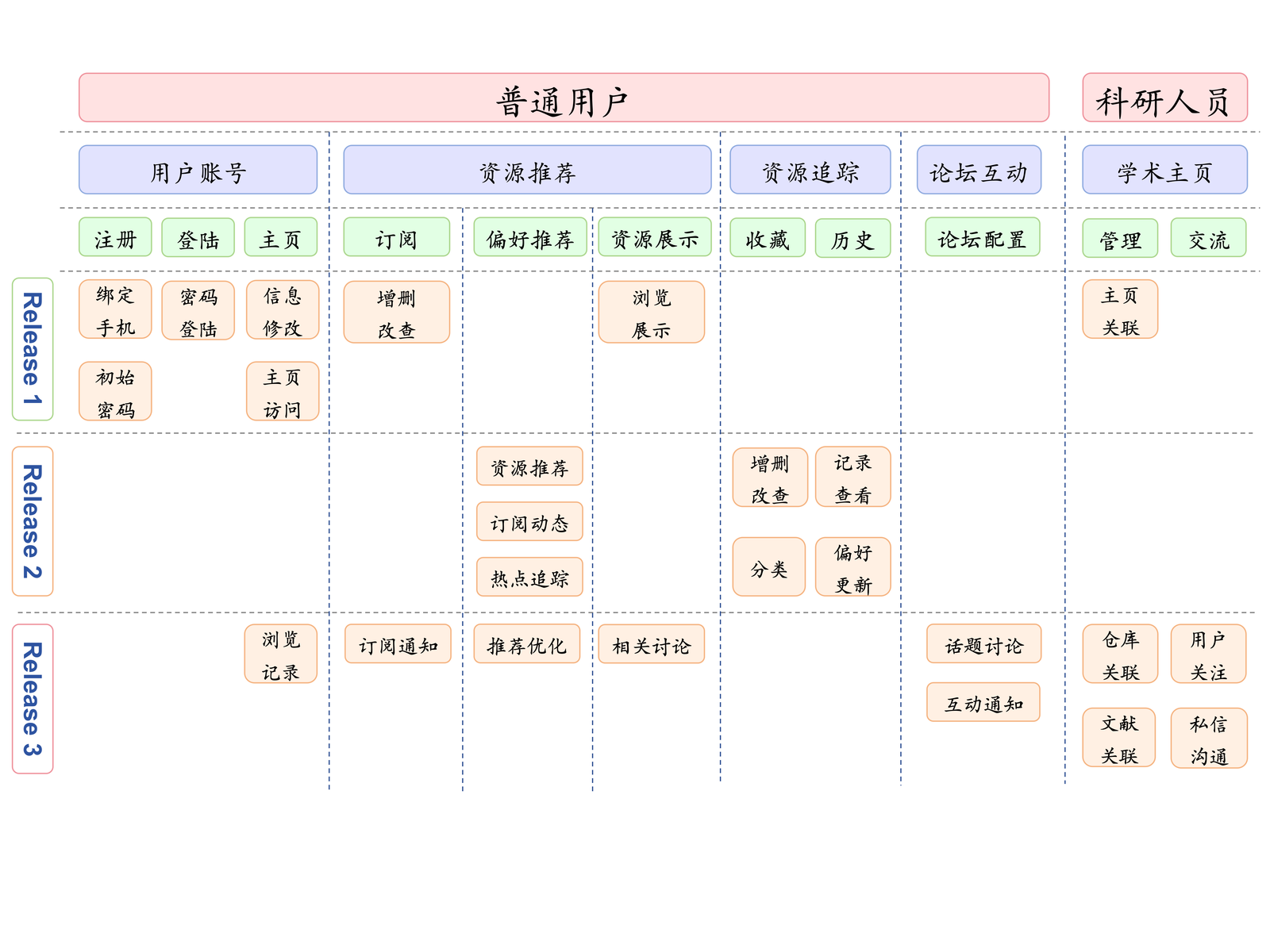
**必要术语表**

* **学术直通车**：本平台名称，旨在整合学术资源，提供便利的获取途径。
* **SRS**：软件需求规格说明书（Software Requirements Specification），描述系统的功能和性能需求。
* **API**：应用程序编程接口（Application Programming Interface），用于不同软件组件之间的交互。
* **GUI**：图形用户界面（Graphical User Interface），用户与系统交互的可视化界面。
* **UAT**：用户验收测试（User Acceptance Testing），验证系统是否满足用户需求的过程。
* **CDN**：内容分发网络（Content Delivery Network），用于加速内容传输的网络架构。
* **JWT**：JSON Web Token，一种用于用户身份验证和访问控制的令牌。

**待确定的问题列表**

* 用户身份验证方法的最终选择，评估 JWT 与其他替代方案的优缺点。
* 日志和监控系统的具体实施细节，以确保平台的稳定性和安全性。
* 容错机制的设计，包括系统在出现故障时的恢复流程与策略。
* 各个页面的具体实现细节，确保用户界面的流畅性与可用性。
* 多语言支持的实施细节，以满足不同地区用户的需求。
* 平台监控与日志记录的具体标准，确保系统性能和用户活动的透明度。

**A.1 用户故事地图**



**A.2 用户故事**

**注册与登录：**

1. 作为普通用户，可以绑定邮箱和手机号完成注册、密码登录，以便访问平台的各项功能。

**个人页面部分：**

1. 作为普通用户，可以修改昵称、密码等，以便调整个人信息与设置。
2. 作为普通用户，可以修改个人简介，以便展示自己的学术画像。
3. 作为普通用户，可以访问其他用户的主页，以便了解他们的研究兴趣和贡献。
4. 作为普通用户，可以通过邮箱验证关联学术主页，以便展示自己的研究成果。

**学术资源推荐部分：**

1. 作为普通用户，可以订阅特定领域、学者或组织，以便获得个性化的学术资源推荐。
2. 作为研究人员，可以浏览学术论文推荐，以便了解学术界科研动态。
3. 作为开发人员，可以浏览代码仓库推荐，以便了解工业界实践动态。
4. 作为普通用户，可以查看订阅学者组织的动态，以便及时获取最新的研究进展。
5. 作为普通用户，可以追踪最新的热门学术资源，以便多方位综合把握学术前沿。

**学术资源展示部分：**

1. 作为普通用户，可以查看学术资源的详细信息，以便深入了解相关论文和项目。
2. 作为普通用户，可以在资源详情页浏览相关讨论，以便参与知识交流。

**资源追踪管理部分：**

1. 作为普通用户，可以收藏感兴趣的资源，以便随时访问。
2. 作为普通用户，可以查看资源浏览历史记录，以便快速找到之前浏览的资源。

**论坛管理部分：**

1. 作为普通用户，可以在论坛中发布讨论和问题，以便与其他用户交流。
2. 作为普通用户，可以对论坛帖子进行评论和回复，以便促进讨论和知识分享。

**通知推送部分：**

1. 作为普通用户，可以接收系统的订阅通知，以便获取最新的学术资源和动态。
2. 作为普通用户，可以收到论坛相关的通知，以便及时参与讨论和交流。

**资源认领部分：**

1. 作为研究人员或开发者，可以认领自己参与的学术资源（如论文、代码项目等），以便将其展示在个人主页中。
2. 作为研究人员或开发者，可以在学术主页中管理已认领的资源，以便更新资源信息或取消认领。
3. **系统总体架构与技术选型**

详见2.4.1技术工具一节。

1. **开发迭代计划**

|  |  |  |
| --- | --- | --- |
| 轮次 | 周次 | 任务 |
| 第一轮 | 第7-8周 | 搭建并部署前后端框架；实现用户注册登录、信息修改和个人主页功能；调研爬虫方案 |
| 第二轮 | 第9-10周 | 实现爬虫；构建领域、学者数据库 |
| 第三轮 | 第11周 | 实现资源展示页面和订阅系统；调研推荐方案；发布版本1 |
| 第四轮 | 第12周 | 实现资源推荐、订阅动态、热点追踪功能 |
| 第五轮 | 第13周 | 实现资源收藏和历史记录管理功能；发布版本2 |
| 第六轮 | 第14周 | 实现论坛相关功能，在资源页面列出对应讨论 |
| 第七轮 | 第15周 | 实现认领功能，用户可认领自己发布的文章/项目；发布版本3 |
| 第八轮 | 第 16 周 | 进行平台测试，编写使用教程与文档；发布最终版本 |