

软件需求规格说明书

软需五杰组 开源项目分析平台

组长：魏鼎坤

版本号：2.0

组员：周正，沈韵涵，龙麟，裴宇航

日期：2022.12.04

目录

- 版本变更历史 3
- 1 项目介绍 4
 - 1.1 背景与编写目的 4
 - 1.2 预期读者与阅读建议 4
 - 1.3 项目范围 4
- 2 项目概述 5
 - 2.1 产品概述 5
 - 2.1.1 产品来源及背景 5
 - 2.1.2 产品前景 5
 - 2.2 产品功能概述 5
 - 2.3 用户类别及特征 6
 - 2.4 设计和实现上的约束 6
 - 2.4.1 数据存储 6
 - 2.4.2 网络服务吞吐 6
 - 2.4.3 数据安全 6
 - 2.5 假设和依赖 6
 - 2.6 用户文档 6
- 3 系统特征 7
 - 3.1 后台管理功能 7
 - 3.2 数据分析功能 7
 - 3.3 数据展示功能 7
 - 3.4 优化数据获取 7
 - 3.5 非功能需求 7
 - 3.6 项目重构 7
- 4 类图与数据流图 8
 - 4.1 类图 8
 - 4.2 数据流图 9
- 5 外部接口 10
 - 5.1 软硬件接口 10
 - 5.2 用户界面 10
 - 5.3 通信接口 10
- 6 非功能性需求 10
 - 6.1 权限与安全 10
 - 6.2 质量属性 10
 - 6.3 性能需求 11
- Appendix..... 12
 - 1 参考文献 12
 - 2 业务规则与业务算法 12
 - 2.1 业务算法 12
 - 2.2 业务规则 13
 - 3 用例图，用例，序列图与通信图 14
 - 3.1.1 管理员用例 14

3.1.2	管理员用例图	15
3.2.1	全体用户用例	16
3.2.2	全体用户用例图	23
3.3	序列图	24
3.4	通信图	30
4	其他 UML 图.....	32
4.1	上下文图	32
4.2	构件图	32
4.3	部署图	33
4.4	状态机图	33
5	术语表	34

版本变更历史

版本号	修订者	修订日期	审核者	审核日期	修订原因
1.0	全体组员	2022.11.09	魏鼎坤	2022.11.11	原始文档
2.0	魏鼎坤 龙麟 沈韵泓	2022.12.02	魏鼎坤	2022.12.04	用户需求变更； 用例迭代； UML 图迭代

1 项目介绍

1.1 背景与编写目的

近年来，开源项目大量涌现，满足特定需求的开源项目种类繁多。为了能够方便地对开源项目进行横向与纵向的比较，本项目提出开源项目分析平台这一方案，旨在解决上述问题。

本项目组对不同用户群体代表进行了访谈，总结出了系统的功能性需求与非功能性需求。为了保证后续开发流程的顺畅，本项目组编写本软件需求说明书。

1.2 预期读者与阅读建议

本项目需求说明书的预期读者主要包括用户、管理员与开发者。

对于用户与管理员读者，本说明书能够为之提供开源项目分析平台的基本功能等信息；对于开发者，本说明书能够为后续的设计、实现、测试工作提供基础。

在阅读本说明书时，建议不同读者群体根据自身需求分清主次，对相应部分进行阅读。

1.3 项目范围

本项目旨在对代码托管平台 **Github** 上的开源项目进行分析，平台提供的数据通过调用 **Github** 相关接口等方式获得，或由获得的数据计算得出。本项目不提供托管在 **Github** 外的开源项目的相关数据。

2 项目概述

2.1 产品概述

2.1.1 产品来源及背景

本项目以 2021-2022 秋冬学期软件需求工程课程大作业提供的 repo 为基础，完善功能并实现一些新需求后更新迭代 2.0 版本。我们小组选取的原型为“Project 2 开源项目分析平台”，一个旨在帮助开发者通过单一平台对不同托管平台上的开源项目进行项目分析、比较，以更好进行项目管理的分析网站。该项目目前已实现：项目信息检索模块、更新频率折线图模块、贡献比例饼状图模块以及 issues 柱状图模块；但在一些方面，如选择数据源模块等，仍有所欠缺。同时，为了使产品更具市场价值，老师以及助教还提出了一些新需求，在本项目中，我们将对所有待更新、完善和实现的功能进行恰当的优先级定义，经过多次版本迭代，最终呈现出一个较为完善的产品。

2.1.2 产品前景

本项目将为“开源项目分析平台”增强数据分析功能、数据展示功能，优化数据获取、系统性能，并完善用户系统。升级过后平台将能够满足用户对特定开源项目（Pytorch）的各项分析需求，并提供可视化支持，以深入挖掘其中开源数据所包含的隐含信息；同时还降低了用户获取开源项目相关数据的门槛，使得用户能够在日常网络环境下享受到流畅的分析体验。综上所述，本产品将为有意研究开源项目的开发者提供强大且便捷的分析力。

本产品目前虽仅针对围绕 Pytorch 项目的分析需求进行开发，但同时将保证良好的可拓展性，若后续有额外需求可灵活增添相关内容的支持。

2.2 产品功能概述

➤ 用户管理功能模块

用户管理功能主要用于区分不同用户的权限级别。系统将为不同用户分配不同级别的权限，普通用户仅能查看后台已录入的项目的数据信息、提出数据更新请求，后台管理员则额外拥有添加/删除监视项目的权限。

➤ 项目数据可视化模块

系统将支持饼状图、折线图、柱状图等多种形式的可视化分析。

◆ 查看项目更新频率

◆ 查看贡献者活跃信息与社区发展速度

系统支持统计一段时间内来自不同贡献者的代码提交数，指出项目的核心贡献者，比如展示哪些贡献者提交代码较为频繁、哪些贡献者只是偶尔提交代码。同时根据活跃人数、提交量等的变化评估社区发展速度和发展趋势。

◆ 查看 issue 信息

提出的 issue 按更新时间在时间轴上的分布、提取 issue 中的关键字分析热度。

◆ 查看贡献者 company 信息

◆ 查看项目讨论数据

系统将通过关键字识别等原理收集项目设计讨论数据，以周为单位可视化设计讨论的数量是如何随时间变化的，并使用 code smell 衡量最新版代码设计质量。

◆ 自定义数据分析

用户可自选项目针对指定属性进行横向比较分析。

此外，本系统也支持对来自多个开源项目的分析数据进行同屏对比。

➤ 数据爬取与分析模块：

用于自动从 GitHub 接口爬取数据，分析后存储至服务端数据库。

2.3 用户类别及特征

本系统将主要面向有意研究 Pytorch 开源项目的开发者，该类用户的研究主题明确且单一，偏向轻度使用，但对于数据覆盖的全面性以及分析的专业性可能有较高的需求。在开发过程中可针对用户的行为特征做相应优化调整。

2.4 设计和实现上的约束

2.4.1 数据存储

项目产品使用标准 MySQL 数据库系统作为引擎，按照数据产生、转换和存储的策略，通过将数据导入数据库的方式进行数据的存储操作。此外本项目对于数据的实效性并无过高需求，因此可考虑本地存储加定期更新的实现策略。

2.4.2 网络服务吞吐

考虑到本项目将主要面向内部开发人员使用，且数据吞吐量、对于负载的要求较低，因此只需确保同时为 20 名用户提供服务的能力。

2.4.3 数据安全

本项目中的数据安全主要涉及用户账号的管理以及权限的区分。拟通过利用安全的框架，在加密的基础上，运用多种方案和技术实现对用户账号的管理，同时使用时将对使用者身份进行验证，只有授权用户能够对后台数据进行相关操作。

2.5 假设和依赖

假设与依赖编号	假设与依赖描述
AS-1	平台提前获取相关开源项目的数据并储存于本地数据库中
AS-2	平台可以得到足够的服务器支持
AS-3	平台用户掌握了电脑的基本操作
AS-4	平台多轮的迭代开发过程有相应的人员负责维护与更新
AS-5	平台的开发人员具有足够的开发与学习能力

2.6 用户文档

本项目的文档分为描述类文档（用于提供整个项目的功能与使用方式）、过程类文档（用于引导初次使用的用户）和参考类文档（分专题提供详细的解释和常见问题的解决方法）

3 系统特征

3.1 后台管理功能

UC-SP-1: 管理员添加监视项目

UC-SP-2: 管理员删除监视项目

3.2 数据分析功能

UC-1: 用户查看项目贡献分析

UC-2: 用户查看活跃度分析

UC-3: 用户查看讨论数据分析

UC-4: 用户查看项目 issue 分析

UC-5: 用户查看 PyTorch 项目分析

用例图请见 Appendix

3.3 数据展示功能

UC-6: 用户查看项目基本数据分析

UC-7: 用户以多项目横向对比方式进行数据展示

UC-8: 用户筛选数据进行数据展示

UC-9: 用户对数据进行排序

用例图请见 Appendix

3.4 优化数据获取

1. 细化数据过滤
2. 数据源缓存
3. 优化信息获取方式

3.5 非功能需求

性能需求:

1. 优化加载速度
2. 提高并发度

质量属性:

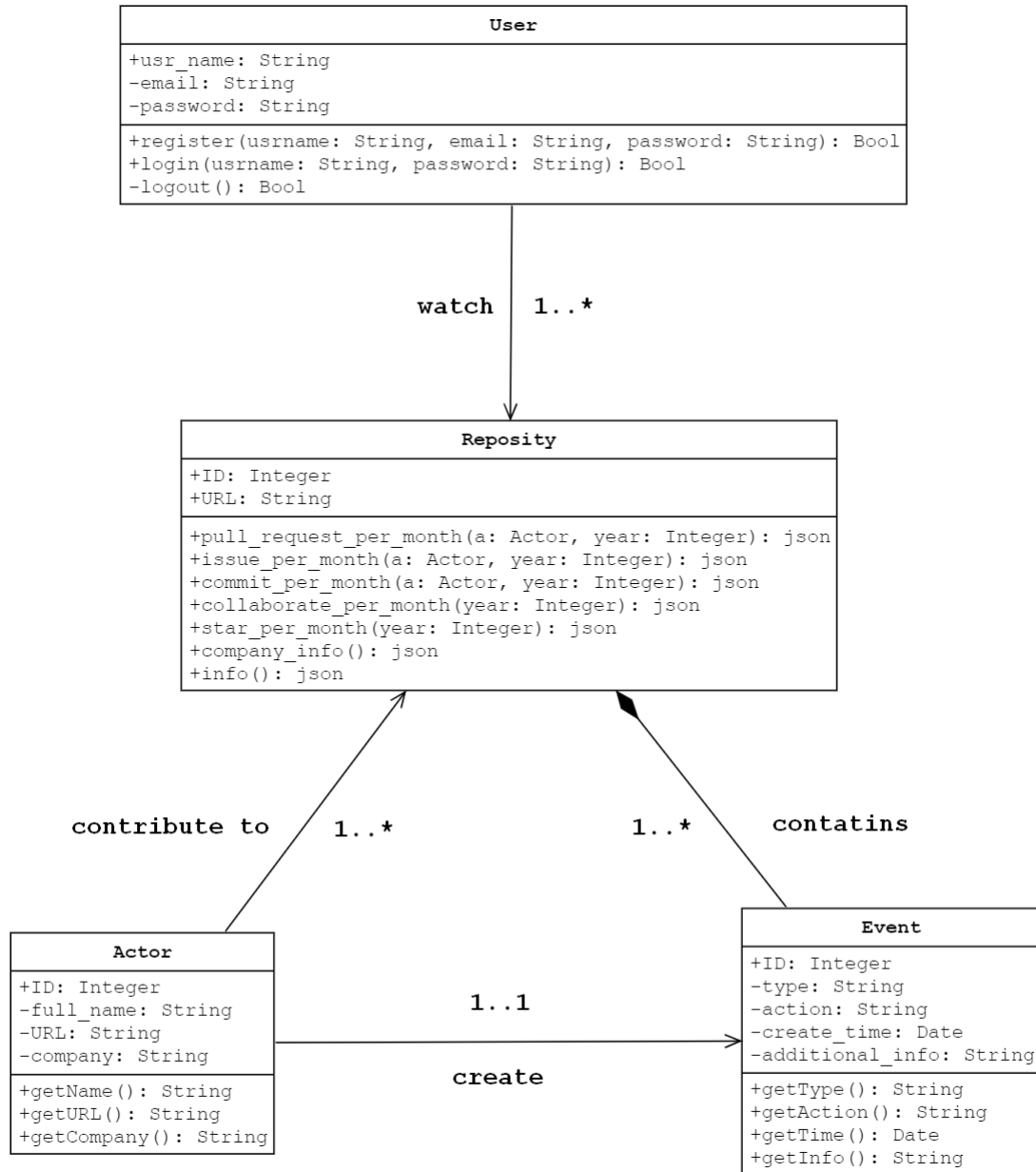
1. 低饱和配色方案
2. 友好的交互提示

3.6 项目重构

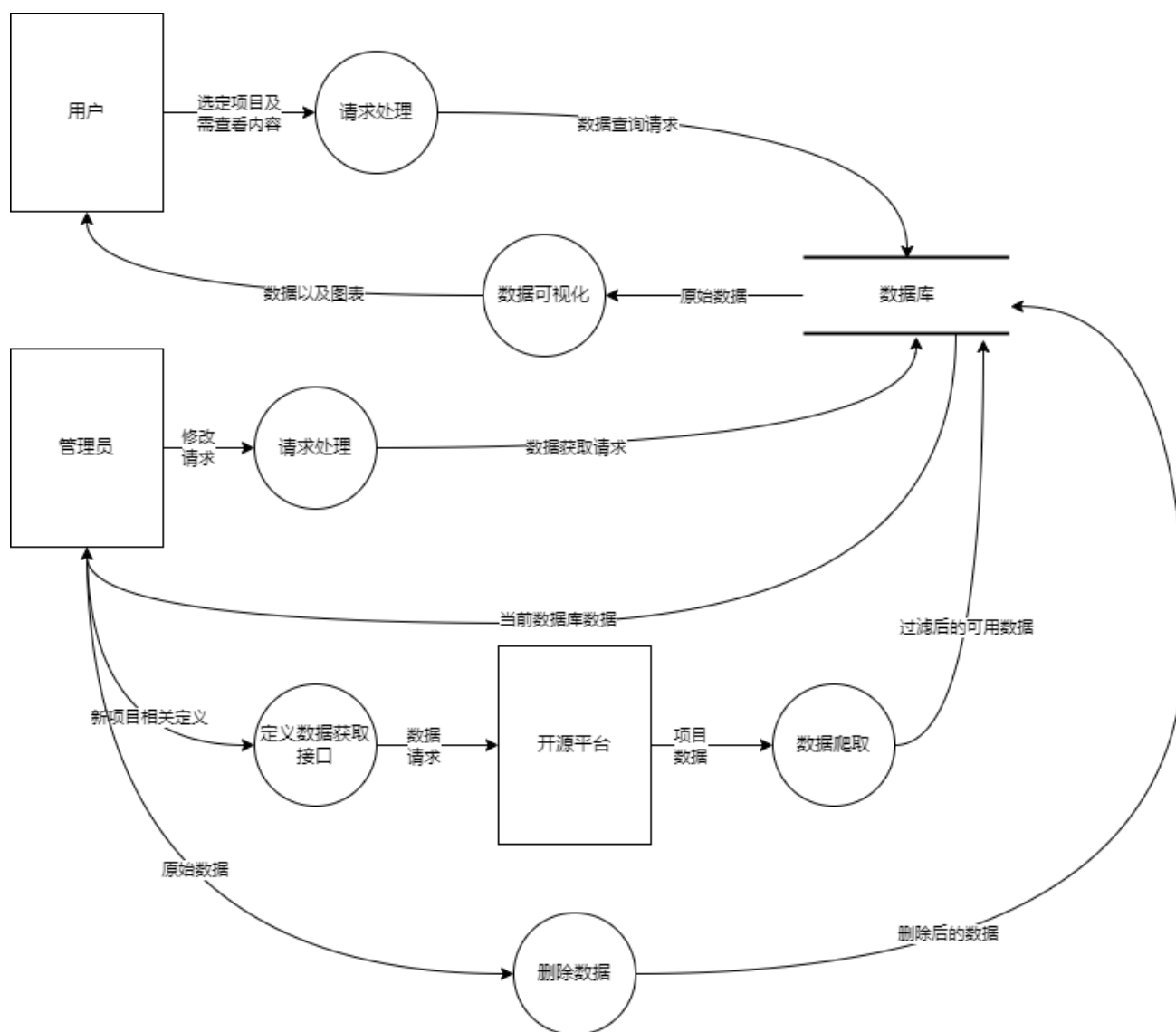
1. 页面设计
2. 前后端分离
3. 数据库重新设计
4. 完善项目规范

4 类图与数据流图

4.1 类图



4.2 数据流图

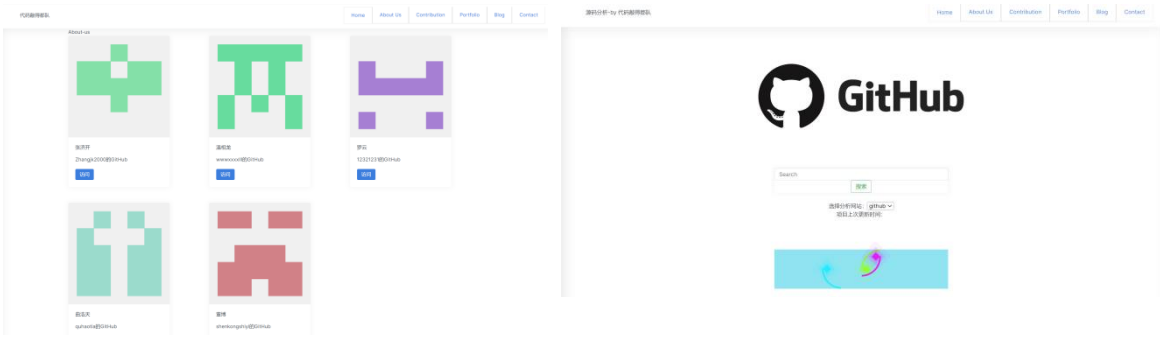


5 外部接口

5.1 软硬件接口

接口名称	软/硬件	类型	详细信息
阿里云	硬件接口	服务器	阿里云服务器提供了 web 的进程接口，处理器 i7，内存 4G，存储 512G
客户端主机	硬件接口	客户端	要求内存 2G 以上，带网卡，处理器至少在 i5 及以上
服务器软件	软件接口	服务器	操作系统 Ubuntu，使用软件 Node.js 和 MySQL
客户端软件	软件接口	客户端	操作系统 Windows、MacOS 或 Linux，使用主流浏览器

5.2 用户界面



5.3 通信接口

本系统使用的网络环境为浙江大学校园网，使用网络通信标准协议。

6 非功能性需求

6.1 权限与安全

本系统中，由于支持所有用户查看已抓包的仓库，但仅注册用户可以添加抓包仓库，所以涉及到权限与安全管理

- 密码安全：对密码要求数字、英文大小写字符、特殊字符混用，不少于 8 位
- 管理员权限：仅管理员能够对监视项目进行增删操作，且做好日志记录
- 游客权限：游客仅可以查看已抓包的仓库、发出更新数据请求，拒绝任何直接修改访问数据库的操作

6.2 质量属性

在本系统中，登陆用户能够用可视化的方式跟踪自己的操作

- 添加抓包仓库后，在主页仓库列表中添加红点，表示正在抓包
- 完成抓包后，弹出框提示用户

6.3 性能需求

本系统中，我们尽可能多的支持用户访问本系统并获得较好的交互体验

- 支持所有主流浏览器访问本系统
- 支持并发访问和并发上传（抓包需求）
- 对用户的操作进行迅速响应，页面渲染要足够迅速

Appendix

1 参考文献

- [1] 《软件工程：实践者的研究方法》，[美] Roger S. Pressman，机械工业出版社
- [2] 《软件需求》，[美] Karl Wieggers, Joy Beatty，机械工业出版社
- [3] IEEE Software Engineering standards: <http://standards.ieee.org/software>

2 业务规则与业务算法

2.1 业务算法

针对项目设计讨论的筛选，拟通过识别不同设计话题关键字，对相关讨论进行打分，最终得分超过一定阈值的则判定为项目设计讨论。不同关键字对应不同分数，具体打分依据可参照下表：

Category	Occurrences	Description
	Agreed on	
code	11	Implementation issues
maintainability	14	Future plans, OS support, code standards...
testing	1	Tests and testability
robustness	13	Robustness, safety, security
performance	2	Performance, runtime optimization
configuration	4	Configuration files, flags and options
documentation	1	Documentation in-code and off-code
clarification	7	Generic question

以“Should we manage to improve the **performance** of **robustness** of our project?”为例，该讨论最终得分应为：2 (performance) + 13 (robustness) = 15 pts

2.2 业务规则

标识符	规则定义	规则类型	静态或动态	来源
BR-01	游客仅能查看 Pytorch 相关的数据分析	约束	静态	流量控制
BR-02	每个可登录用户都有其唯一对应的 ID	事实	静态	管理策略
BR-03	注册用户可自选项目进行横向比较	约束	动态	流量控制
BR-04	授权用户可提出增加项目申请	约束	动态	流量控制
BR-05	授权账户应由管理员分配或审核	约束	静态	管理策略
BR-06	添加的监听项目数目不得超过上限	动作触发规则	动态	流量控制
BR-07	按时间轴展示数据应有最小有效单位	动作触发规则	动态	可视化策略
BR-08	授权用户申请添加项目时应提供准确、有效指向信息	动作触发规则	动态	管理员
BR-09	系统或管理员应定期更新数据库并对用户提交的申请做出反馈	动作触发规则	动态	管理员
BR-10	所有超过 5 年的资料和信息应该由管理员手动清除	动作触发规则	动态	管理员
BR-11	用户忘记密码且无法通过预留手段找回密码时，应报告管理员进行密码重置	动作触发规则	动态	管理策略
BR-12	管理员所有的操作都应记录在网站的日志中	动作触发规则	动态	管理员
BR-13	只有用户信息被管理员手动导入系统或注册成功后才可登录	约束	动态	管理策略
BR-14	同屏比较项目数目有最大上限	动作触发规则	动态	流量控制

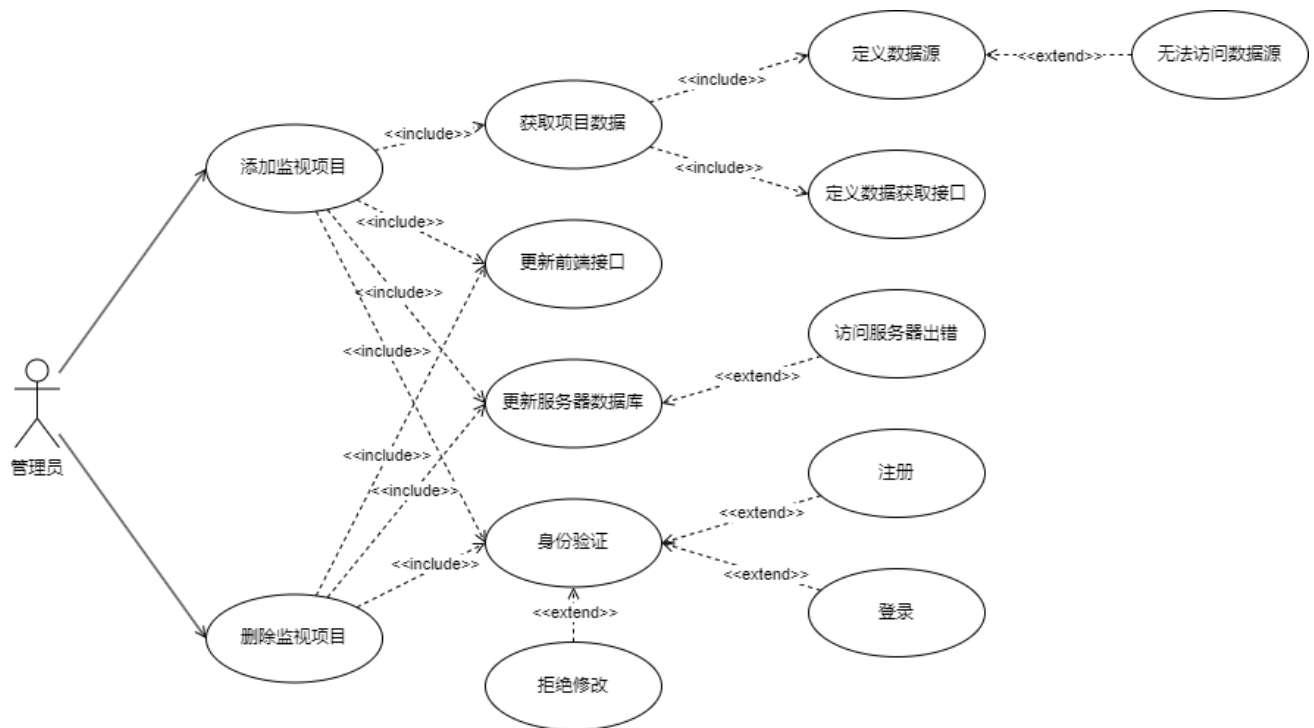
3 用例图，用例，序列图与通信图

3.1.1 管理员用例

ID and Name:	UC-SP-1 管理员添加监视项目		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	管理员	Secondary Actors:	开源平台，服务端数据库
Description:	在获取检索结果后，管理员可以通过网址或关键词搜索到开源平台上的项目，便于对感兴趣的项目添加监视		
Trigger:	用户决定对检索到的开源项目添加监视		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已检索到非空的结果		
Postconditions:	POST-1. 该用户已添加监视的项目中增加了该项目 POST-2. 服务端数据库添加了该项目对应的数据		
Normal Flow:	1.0 添加监视项目 1. 用户选择需要添加监视的项目进行添加监视操作 2. 系统应用开源平台的接口获取该项目的最新详细数据 3. 系统将获取的数据持久化到本地数据库内 4. 系统显示信息：项目添加成功		
Alternative Flows:	无		
Exceptions:	1.0.E1 项目数据爬取失败 1. 系统让客户选择重新爬取项目数据（2a）或取消添加监视（5a） 2a. 用户选择重新爬取项目数据 2a. 系统重新从开源平台爬取项目数据到本地数据库，得到爬取成功（3a）或失败（4a）的爬取结果 3a. 系统显示爬取成功 3b. 系统重新回到 normal flow 的 4 继续运行 4a. 显示错误信息和结束查看选项（5a） 5a. 用户选择结束查看 5b. 系统结束用例		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	BR-04 BR-06		
Other Information:	本系统中所有添加监视的项目数据都需要存储在本地		
Assumption:	用户已经正常连接网络		

ID and Name:	UC-SP-2 管理员删除监视项目		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	管理员	Secondary Actors:	开源平台，服务端数据库
Description:	管理员选择取消对项目的监视		
Trigger:	用户决定对检索到的开源项目取消监视		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已检索到非空的结果		
Postconditions:	POST-1. 该项目从监视列表中移除 POST-2. 服务端数据库删除了该项目对应的数据		
Normal Flow:	1.0 删除监视项目 1. 用户选择需要删除监视的项目进行删除监视操作 2. 系统将移除服务端数据库中的 相关数据 3. 系统显示信息：项目删除成功		
Alternative Flows:	无		
Exceptions:	无		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	BR-04 BR-06		
Other Information:	无		
Assumption:	用户已经正常连接网络		

3.1.2 管理员用例图



3.2.1 全体用户用例

ID and Name:		UC-1 用户查看项目贡献分析	
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	贡献分析是项目详细数据分析的一个具体选项，包括贡献者代码提交数分析、核心贡献者分析		
Trigger:	用户决定查看项目贡献分析		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到项目的贡献者代码提交数分析和核心贡献者分析的图表和数据		
Normal Flow:	4.0 查看项目贡献分析 1. 用户选择查看项目贡献分析的选项 2. 系统显示该项目的贡献者代码提交数分析和核心贡献者分析的图表和数据		
Alternative Flows:	无		
Exceptions:	无		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	无		
Other Information:	无		
Assumption:	无		
ID and Name:		UC-2 用户查看活跃度分析	
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	活跃度分析是项目详细数据分析的一个具体选项，包括贡献者活跃度分析、社区发展速度分析		
Trigger:	用户决定查看项目活跃度分析		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到项目的贡献者活跃度分析和社区发展速度分析的图表和数据		
Normal Flow:	5.0 查看项目活跃度分析 1. 用户选择查看项目活跃度分析的选项 2. 系统显示项目贡献者活跃度分析和社区发展速度分析的图表和数据		
Alternative Flows:	无		
Exceptions:	无		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	BR-01		
Other Information:	无		
Assumption:	无		

ID and Name:	UC-3 用户查看讨论数据分析		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	讨论数据分析是项目详细数据分析的一个具体选项，包括代码设计质量衡量、讨论-时间关系分析、设计话题分析		
Trigger:	用户决定查看项目讨论数据分析		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到项目的代码设计质量衡量、讨论-时间关系分析、设计话题分析的图表和数据		
Normal Flow:	6.0 查看项目讨论数据分析 1. 用户选择查看项目讨论数据分析的选项 2. 系统显示该项目的代码设计质量衡量、讨论-时间关系分析、设计话题分析的图表和数据		
Alternative Flows:	无		
Exceptions:	无		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	BR-01		
Other Information:	无		
Assumption:	无		

ID and Name:	UC-4 用户查看项目 issue 分析		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	issue 分析是项目详细数据分析的一个具体选项，包括 issue 时间分布分析、issue 关键字热度分析		
Trigger:	用户决定查看项目 issue 分析		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到 issue 时间分布分析和 issue 关键字热度分析的图表和数据		
Normal Flow:	7.0 查看项目讨论数据分析 1. 用户选择查看项目 issue 分析的选项 2. 系统显示项目的 issue 时间分布分析与关键字热度分析的图表和数据		
Alternative Flows:	无		
Exceptions:	无		
Priority:	中		
Frequency of Use:	一般		
Business Rules:	BR-01		
Other Information:	无		
Assumption:	无		

ID and Name:	UC-5 用户查看 PyTorch 项目分析		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	PyTorch 项目分析是该系统的必要功能，因此 PyTorch 项目的数据已经预先爬取存储在本地数据库中，并显示在用户已添加监视的项目列表中，用户不需要再进行检索。PyTorch 的分析中包含人数占比可视化分析与 module 占比分析这两个独特的分析功能。		
Trigger:	用户决定查看 PyTorch 项目的数据及分析		
Preconditions:	PRE-1. 用户已登录		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到 PyTorch 项目的基本数据和图表 POST-2. 如果该用例正常运行，用户可以看到 PyTorch 项目的人数占比可视化分析与 module 占比分析的数据和图表		
Normal Flow:	8.0 查看 PyTorch 项目分析 <ol style="list-style-type: none"> 1. 用户从已添加监视的项目列表中选择 PyTorch 项目 2. 系统显示 PyTorch 项目的基本数据和图表供用户进行概览，同时呈现人数占比可视化分析与 module 占比分析的数据和图表 		
Alternative Flows:	无		
Exceptions:	无		
Priority:	高		
Frequency of Use:	一般		
Business Rules:	BR-01		
Other Information:	PyTorch 项目的其他分析维度（贡献分析、活跃度分析、讨论数据分析、issue 分析）的实现根据项目进度而定；如果能够实现这些分析功能，则该用例可以看作对 UC-3 的继承		
Assumption:	PyTorch 项目的相关数据已经预先爬取到本地数据库内		

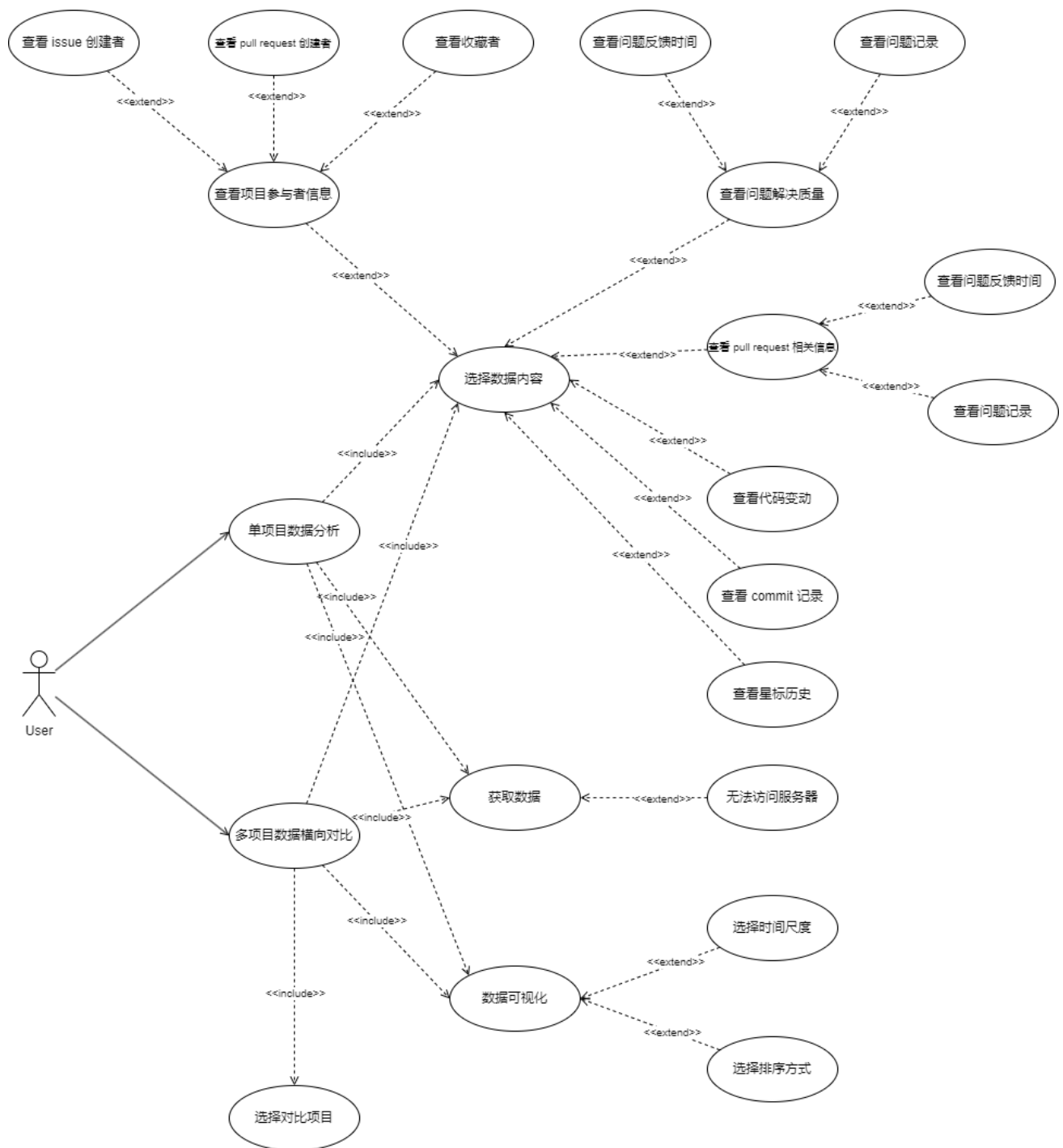
ID and Name:	UC-6 用户查看项目基本数据分析		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	用户事先将一些感兴趣的项目数据爬取到本地数据库，当需要查看项目的数据分析时，选择对应项目，系统就会显示数据基本图表（包括更新频率折线图、贡献比例饼状图、issue 柱状图），并提供更加详细的数据分析选项（包括贡献分析、项目讨论数据分析、issue 分析等）供用户进一步选择以及进阶的数据展示选项（包括多项目横向对比、图表数据删选等）		
Trigger:	用户决定选择项目以查看数据及分析		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户需要查看的项目已经添加到监控，其数据已经被爬取到本地数据库		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到项目的基本数据和图表 POST-2. 如果该用例正常运行，用户可以选择详细分析选项进行进一步分析 POST-3. 如果该用例正常运行，用户可以选择进阶图表展示选项进行高级图表展示		
Normal Flow:	1.0 查看项目基本数据分析 1. 用户从已添加监控的项目中选择需要查看的项目 2. 系统显示该项目的基本数据和图表供用户进行概览，以及详细分析选项供用户选择		
Alternative Flows:	无		
Exceptions:	3.0.E1 用户选择的项目在本地数据库中不存在 1. 系统显示信息：项目数据未在本地 2. 系统让客户选择重新爬取项目数据（3a）或结束查看（6a） 3a. 用户选择重新爬取项目数据 3b. 系统重新从开源平台爬取项目数据到本地数据库，得到爬取成功（4a）或失败（5a）的爬取结果 4a. 系统显示爬取成功 4b. 系统重新回到 normal flow 的 2 继续运行 5a. 显示错误信息和结束查看选项（6a） 6a. 用户选择结束查看 6b. 系统结束用例		
Priority:	低		
Frequency of Use:	一般		
Business Rules:	BR-01		
Other Information:	数据基本图表和数据分析选项可以随着迭代开发而增加		
Assumption:	无		

ID and Name:	UC-7 用户以多项目横向对比方式进行数据展示		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	多项目横向对比是项目数据进阶展示的一个具体选项，用户可以选择若干个项目的若干个指标进行对比展示		
Trigger:	用户决定进行多项目横向对比		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到多个项目对于某些指标的对比展示		
Normal Flow:	9.0 以多项目横向对比方式进行数据展示 1. 用户选择多项目横向对比的选项 2. 系统以勾选列表的方式呈现用户已添加监视的其他项目 3. 用户在列表中勾选需要用于对比的若干个其他项目 4. 系统以勾选列表的方式呈现需要用于对比的指标 5. 用户在列表中勾选需要用于对比的指标 6. 系统以对比图表的方式呈现数据展示结果		
Alternative Flows:	无		
Exceptions:	9.0.E1 除当前项目外无其他项目可选 1. 系统显示信息：无其他项目可选 2. 系统显示取消对比的选项 3. 用户选择取消对比 4. 系统结束用例		
Priority:	高		
Frequency of Use:	频繁		
Business Rules:	BR-02 BR-03 BR-11 BR-13		
Other Information:	同时可选的对比项目和对比指标的个数限制根据具体实现而定		
Assumption:	用例描述中用户勾选的对比项目以及对比指标的个数均未超过限制		

ID and Name:	UC-8 用户筛选数据进行数据展示		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	筛选数据是项目数据进阶展示的一个具体选项，用户可以对当前项目的若干个指标进行时间范围的筛选，并通过缩放功能进行进一步的时间范围调整		
Trigger:	用户决定筛选数据进行显示		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以看到筛选时间范围内的数据		
Normal Flow:	10.0 筛选数据进行数据展示 <ol style="list-style-type: none"> 1. 用户选择筛选数据的选项 2. 系统以勾选列表的方式呈现当前项目的所有指标 3. 用户在列表中勾选若干指标用于筛选 4. 系统呈现起止时间的选择框 5. 用户设定起止时间 6. 系统呈现起止时间范围内所选指标与时间的关系图 7. 用户可以通过调整时间范围的滚轮进行时间范围的缩放与调整 		
Alternative Flows:	无		
Exceptions:	无		
Priority:	中		
Frequency of Use:	频繁		
Business Rules:	BR-07		
Other Information:	无		
Assumption:	无		

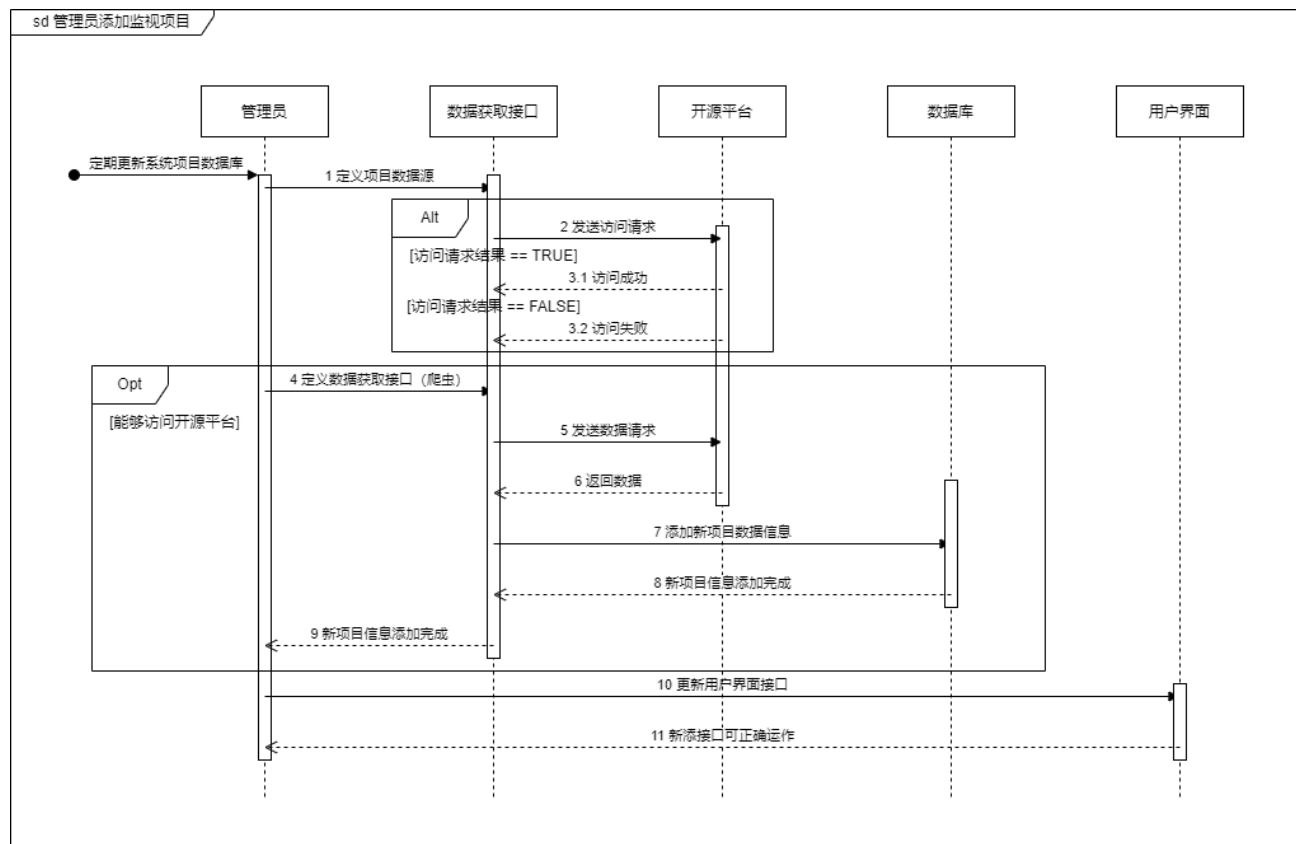
ID and Name:	UC-9 用户对数据进行排序		
Created By:	软需五杰组	Date Created:	2022/11/10
Primary Actor:	用户	Secondary Actors:	本地数据库
Description:	数据排序是		
Trigger:	用户决定筛选数据进行显示		
Preconditions:	PRE-1. 用户已登录 PRE-2. 用户已打开项目的基本数据分析页面，即已完成 UC-3		
Postconditions:	POST-1. 如果该用例正常运行，用户可以以表格的形式看到按照所选指标进行排序后的数据表		
Normal Flow:	11.0 数据排序据进行数据展示 1. 用户选择数据排序的选项 2. 系统以勾选列表的方式呈现当前项目的指标 3. 用户在列表中勾选若干指标用于排序，用户选择指标的的顺序对应于排序指标的优先级顺序 4. 系统以表格形式呈现排序后的数据		
Alternative Flows:	无		
Exceptions:	无		
Priority:	中		
Frequency of Use:	频繁		
Business Rules:	无		
Other Information:	用户在勾选指标时，勾选框内可以以数字的形式标识排序指标的优先级顺序		
Assumption:	无		

3.2.2 全体用户用例图

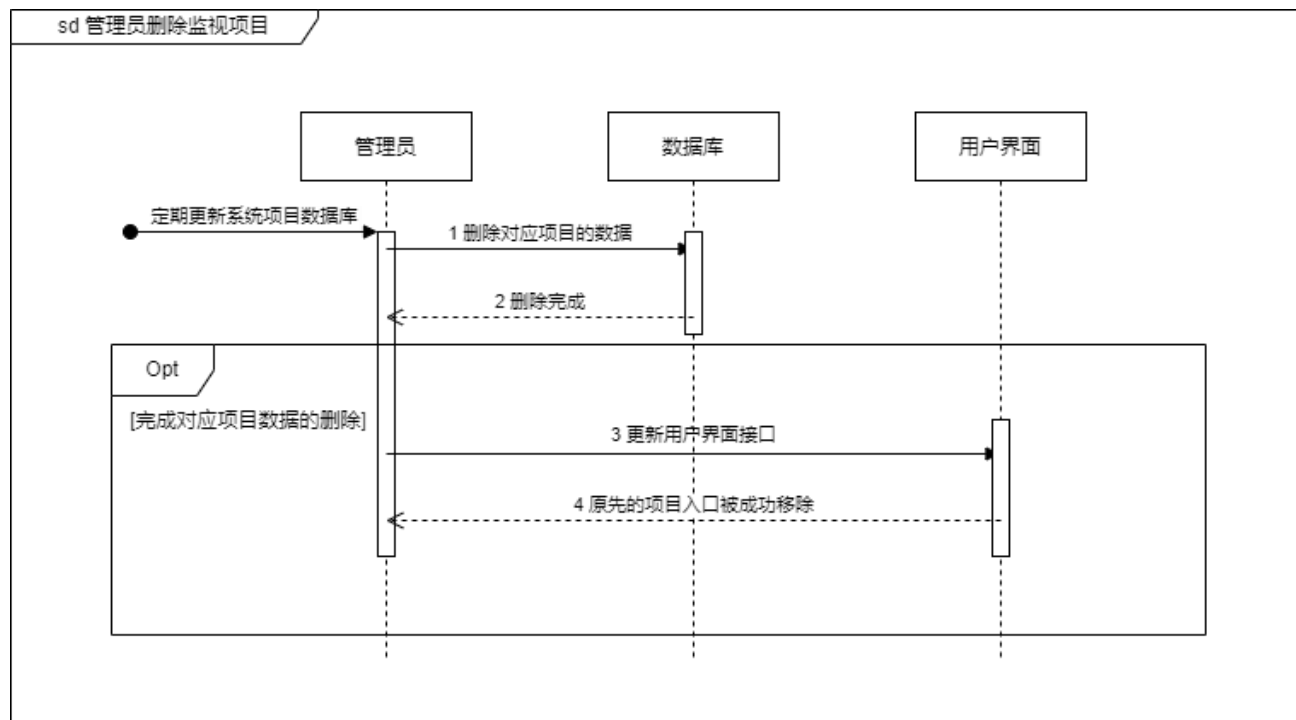


3.3 序列图

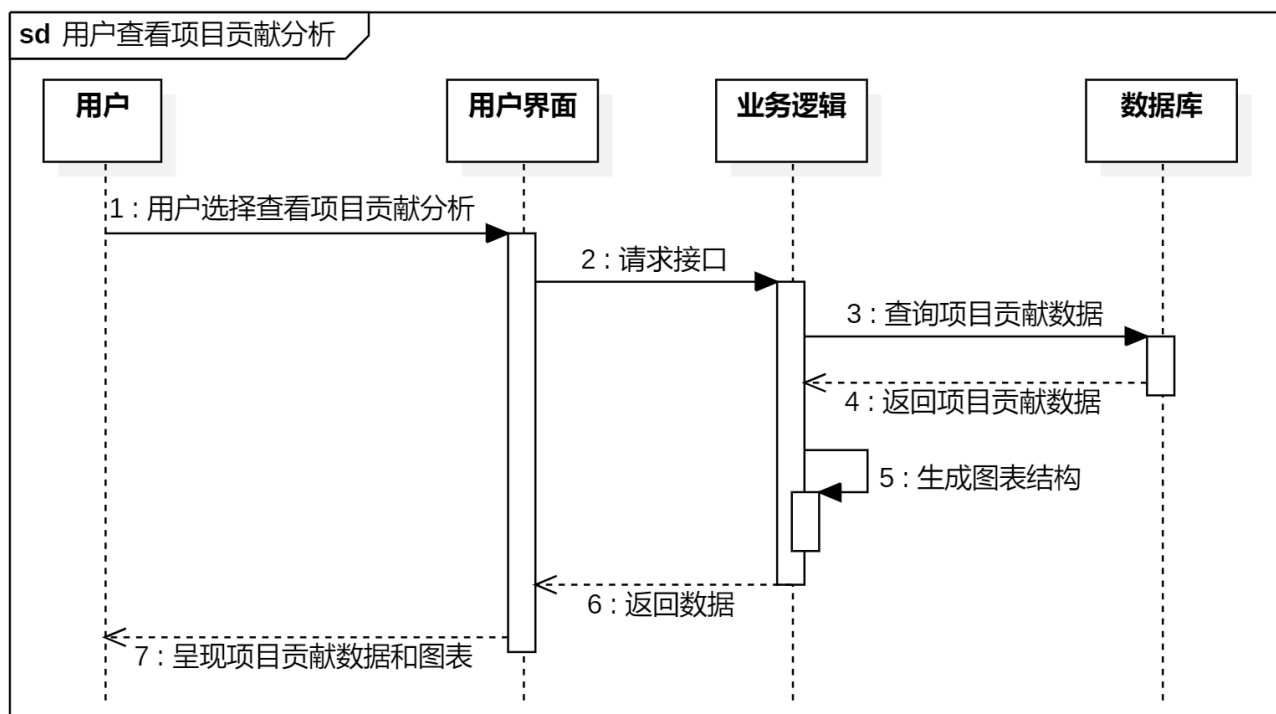
UC-SP-1



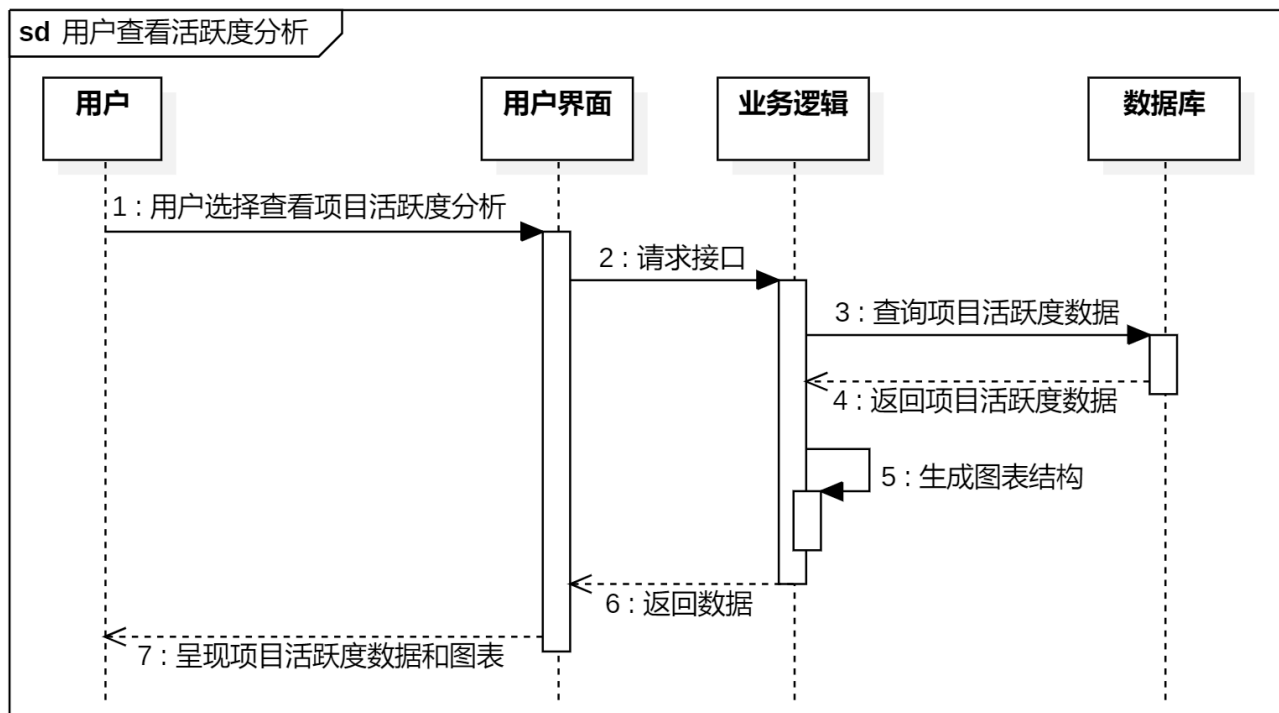
UC-SP-2



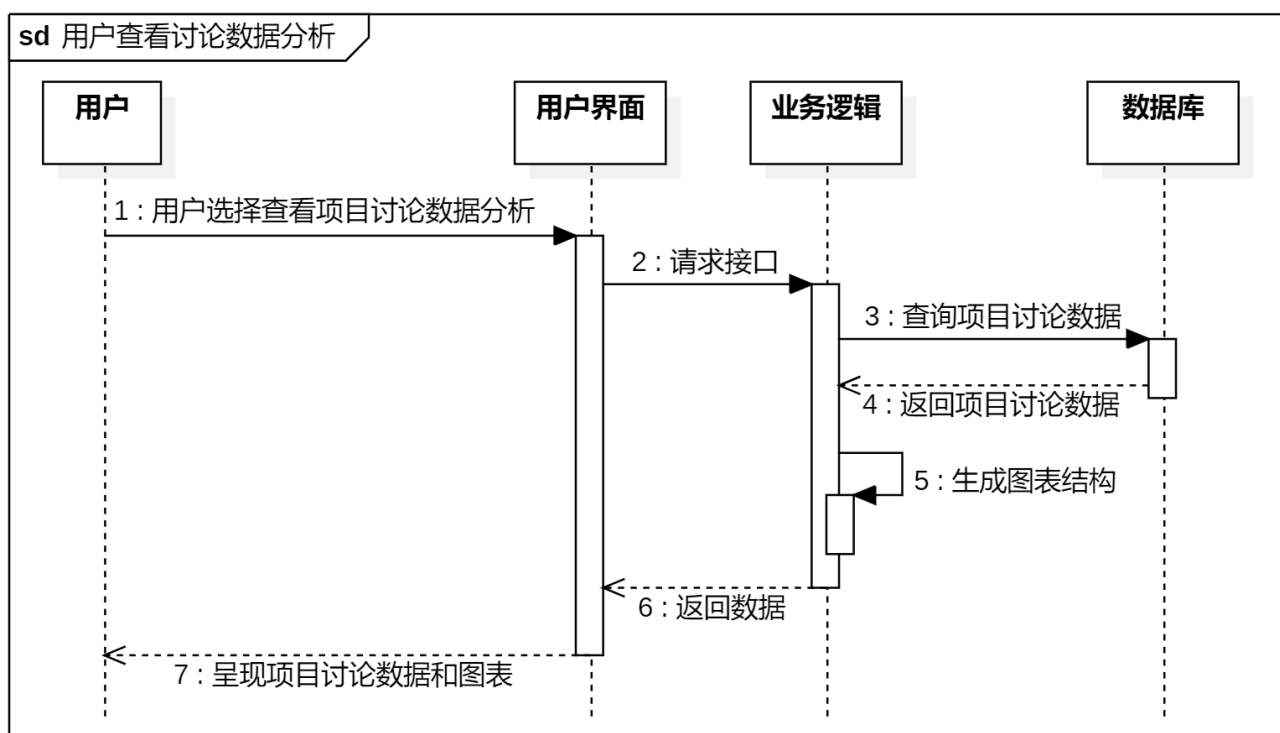
UC-1



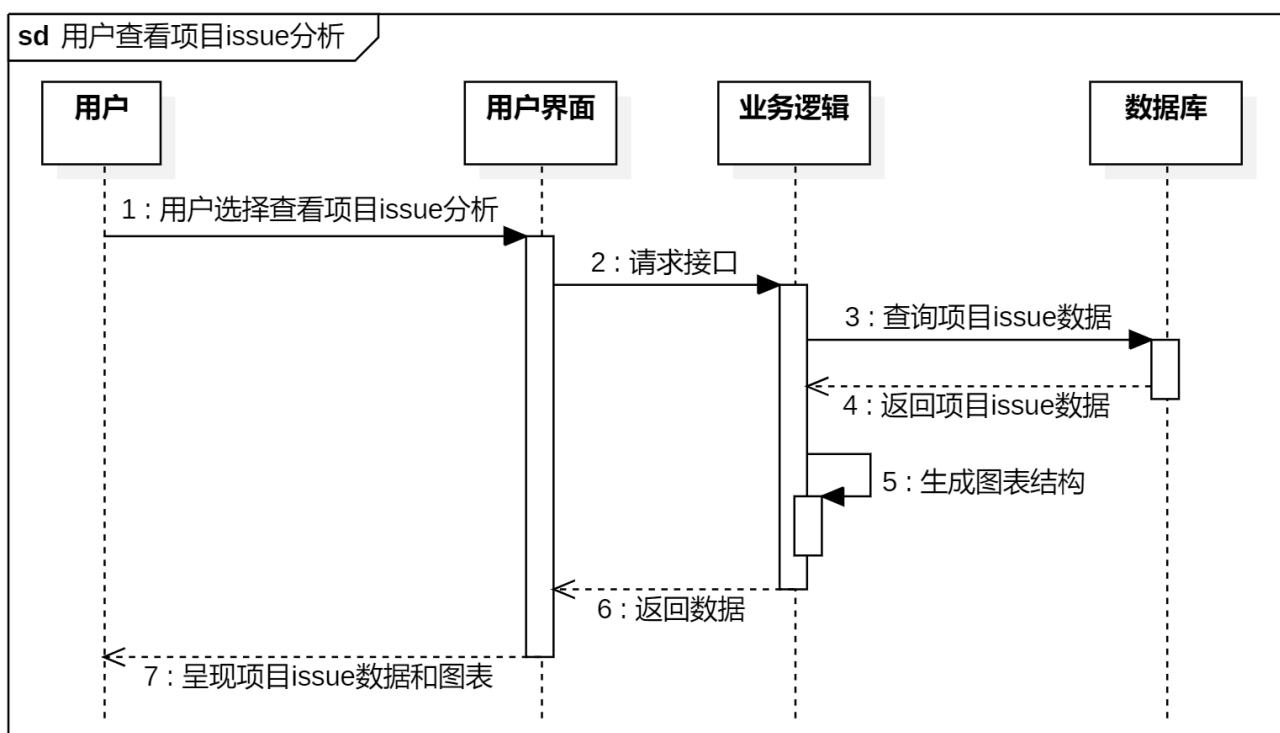
UC-2



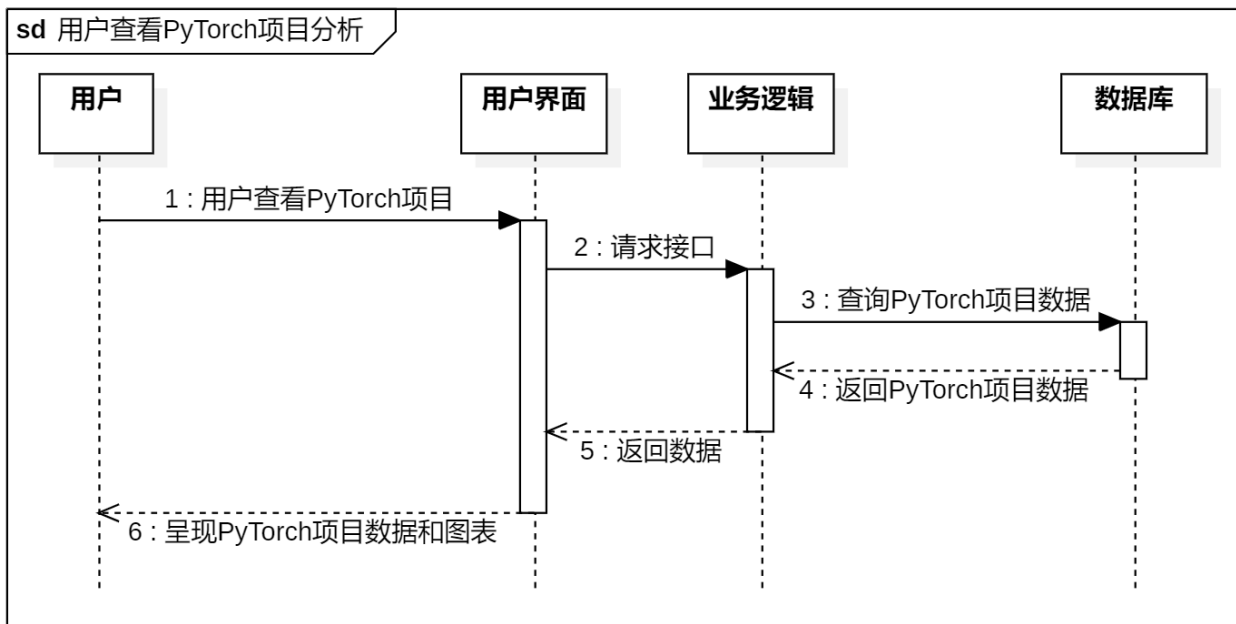
UC-3



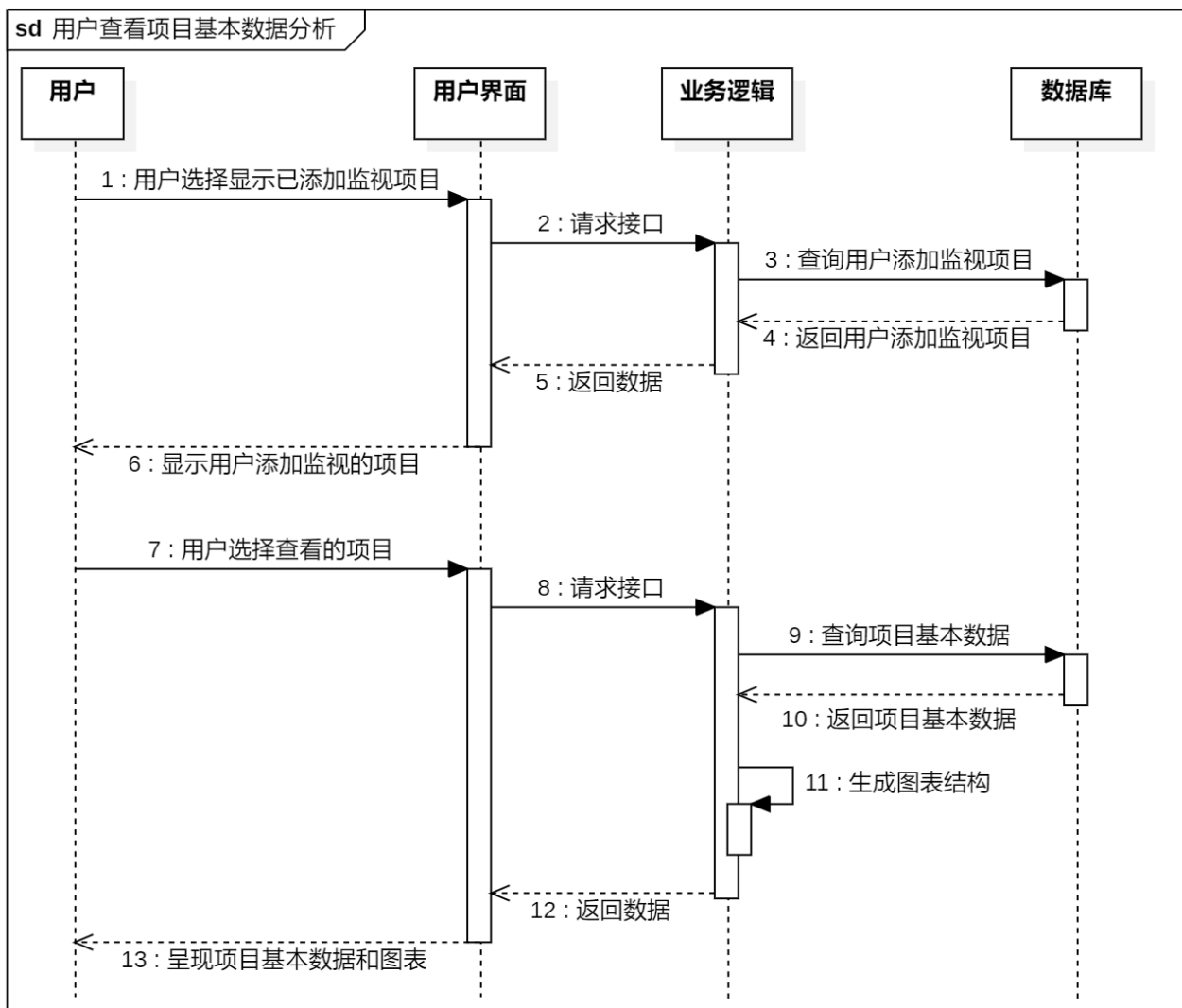
UC-4



UC-5

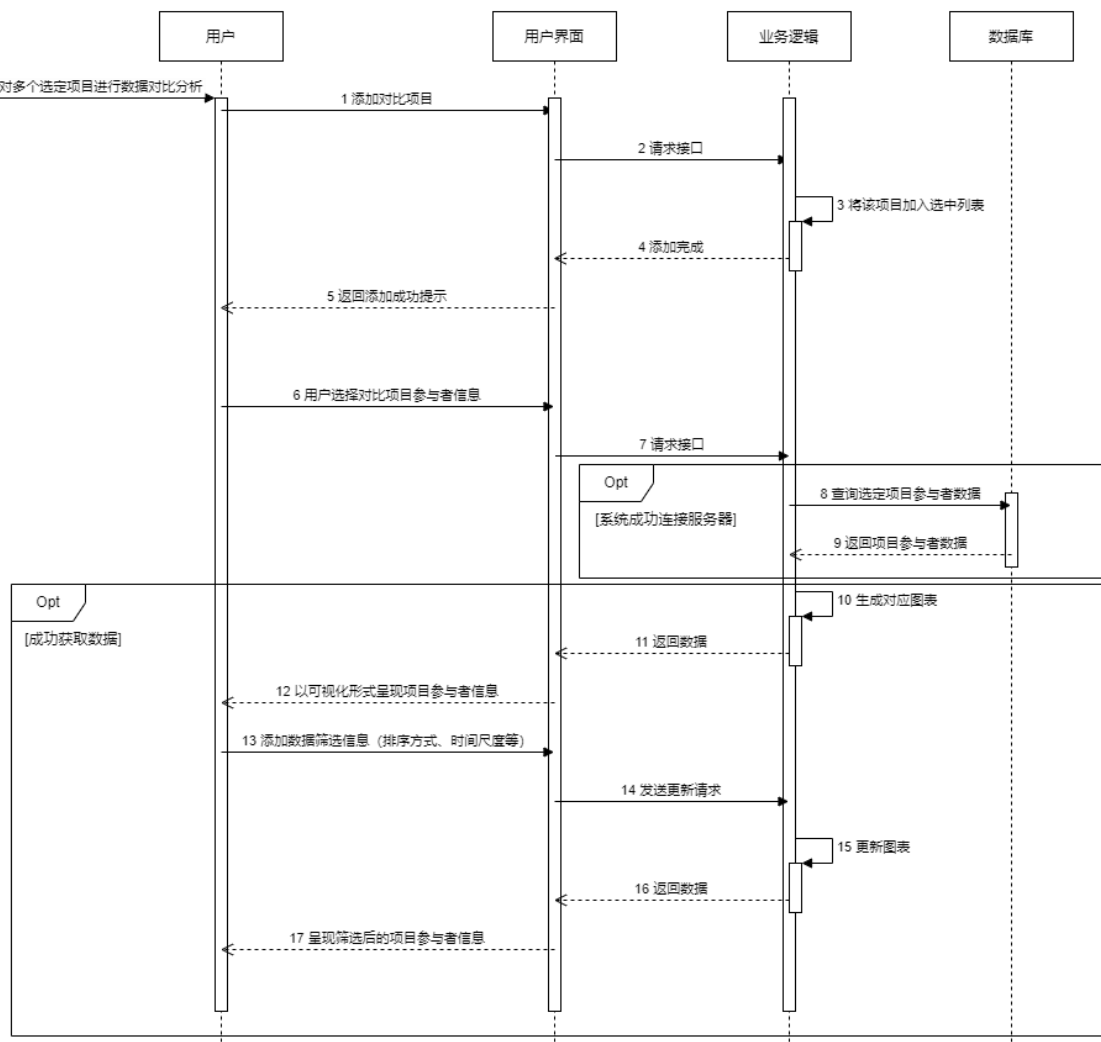


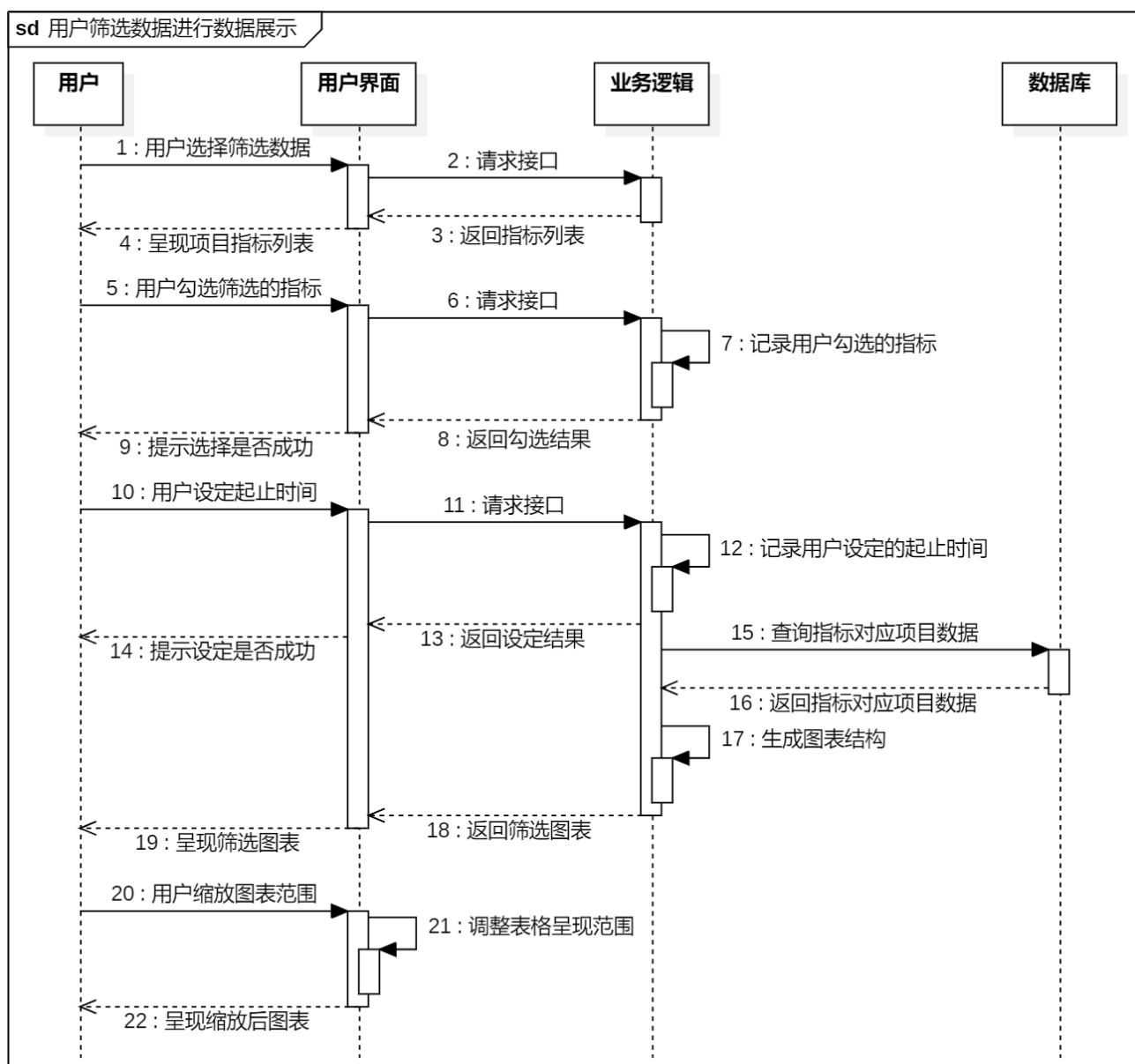
UC-6



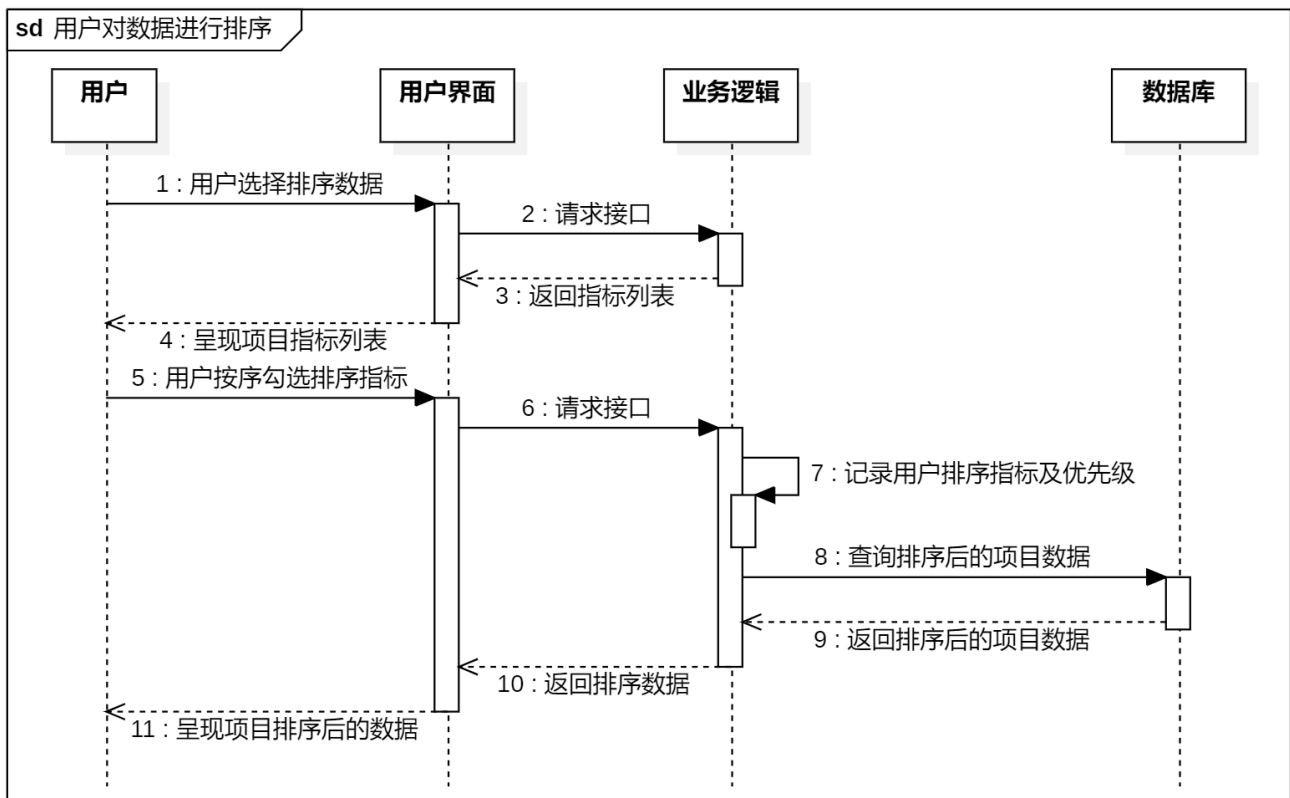
UC-7

sd 用户查看多项目数据横向对比 (以项目参与者信息为例)

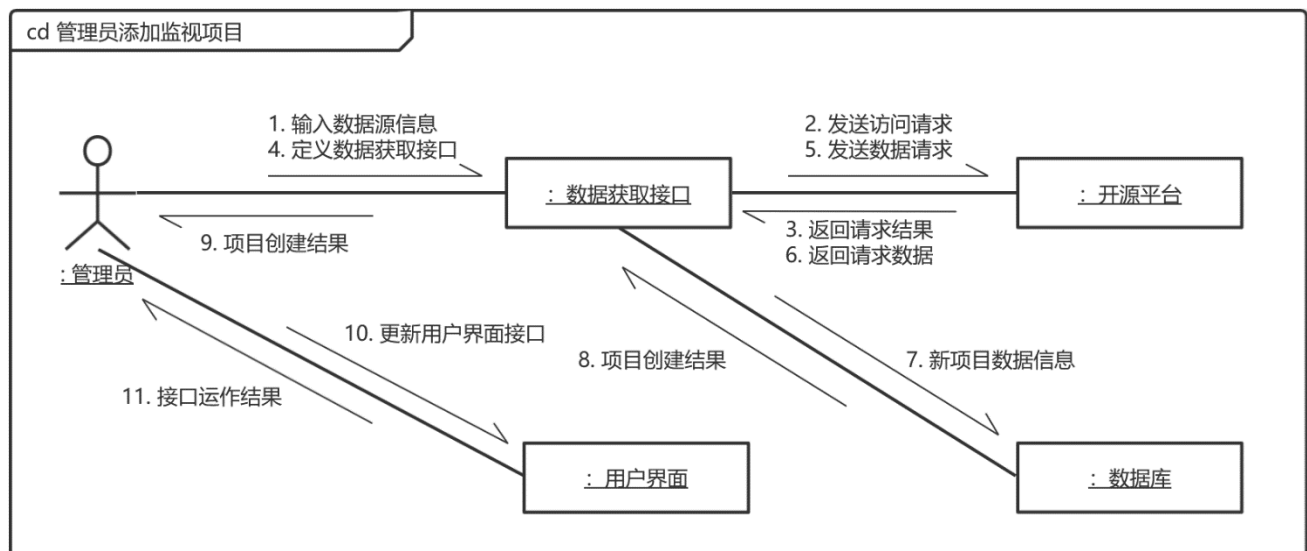




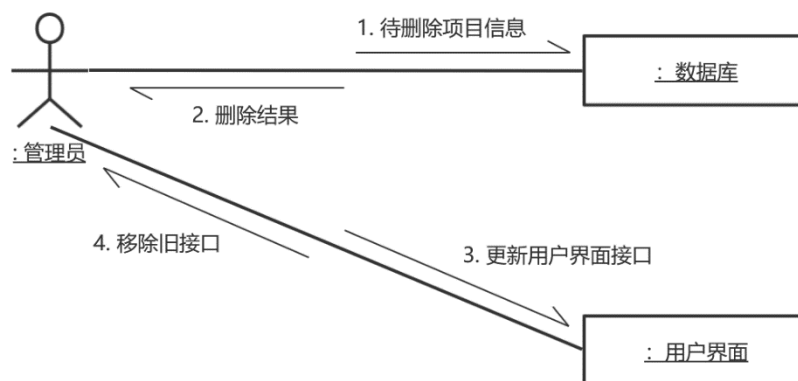
UC-9



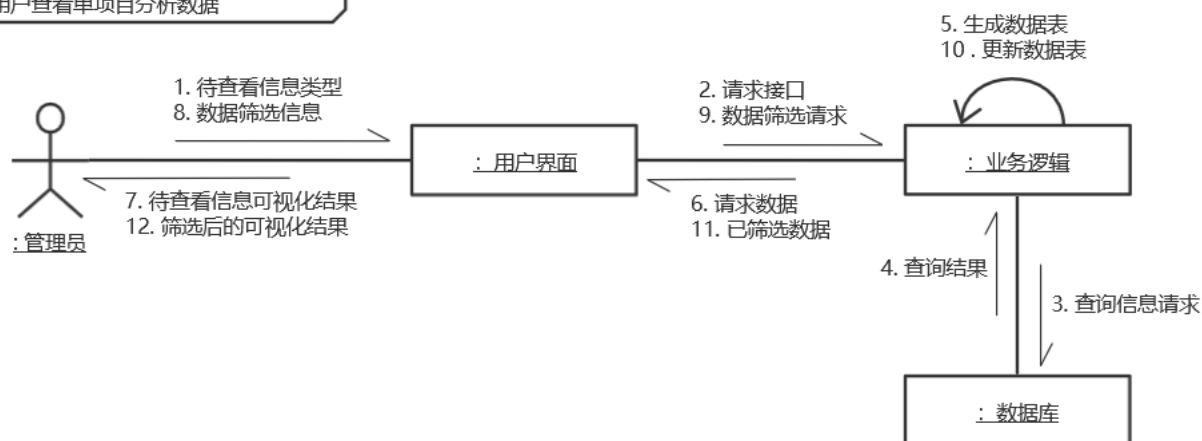
3.4 通信图



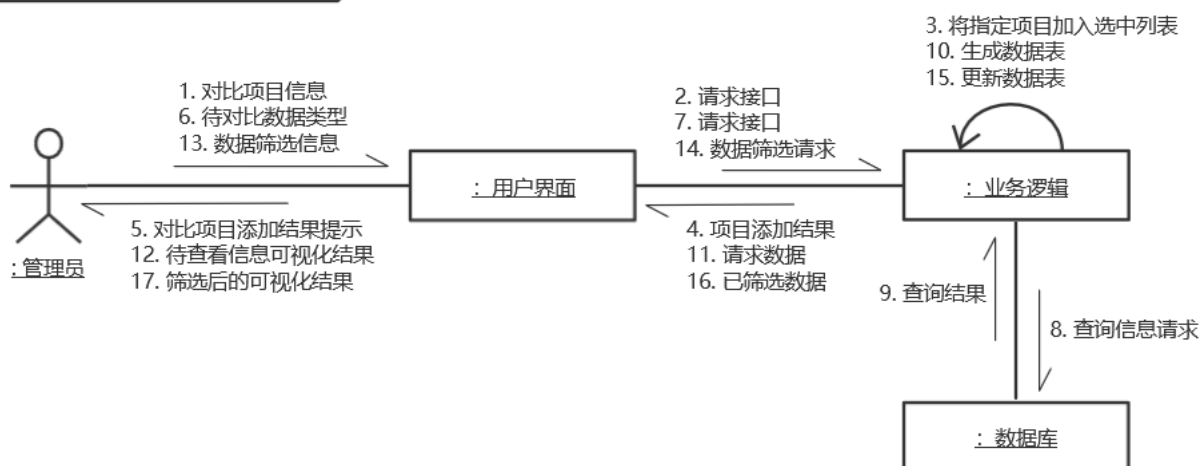
cd 管理员删除监视项目



cd 用户查看单项目分析数据

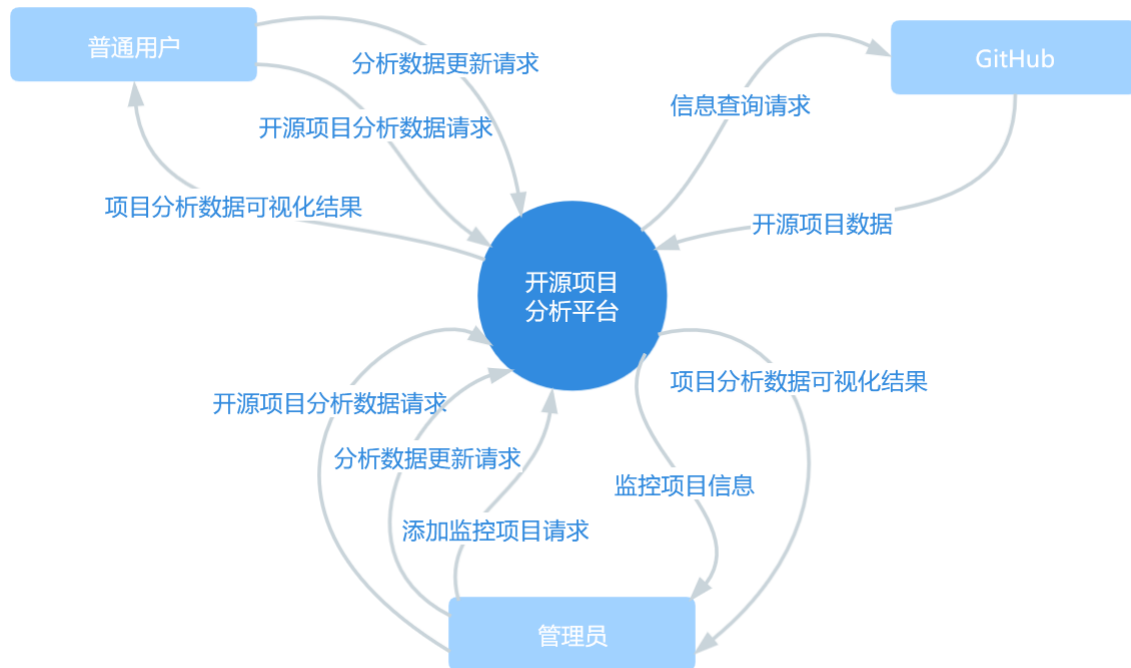


cd 用户查看多项目数据横向对比

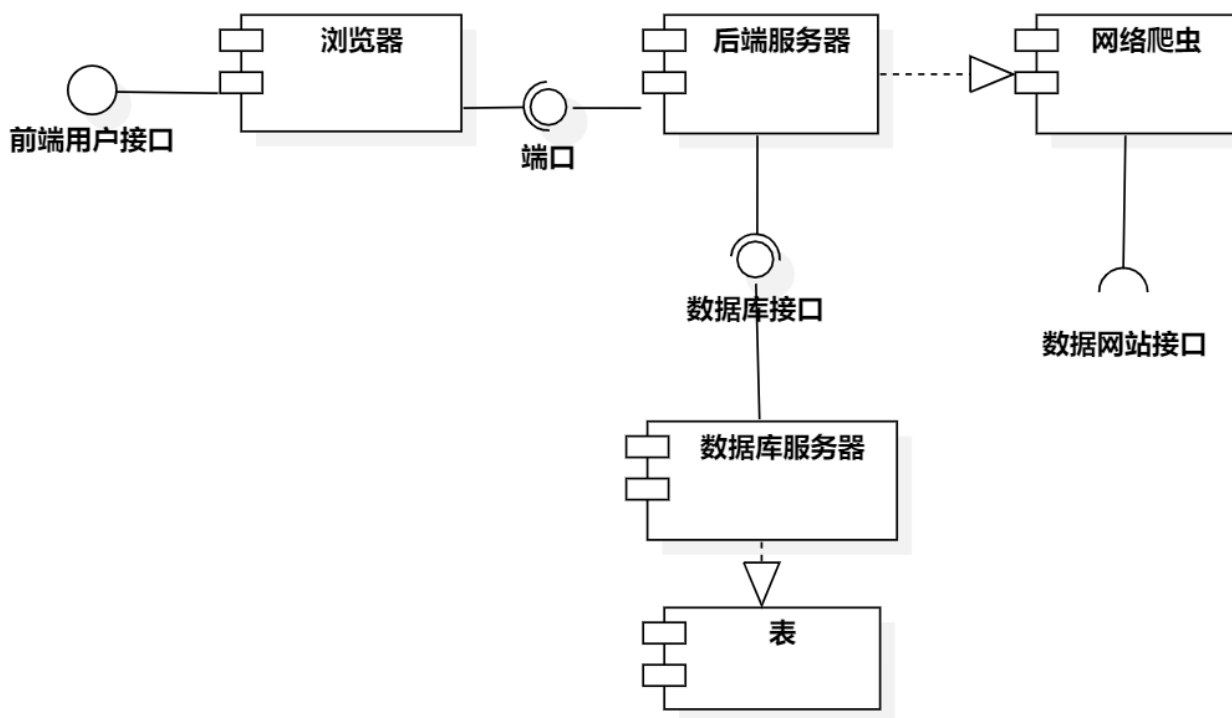


4 其他 UML 图

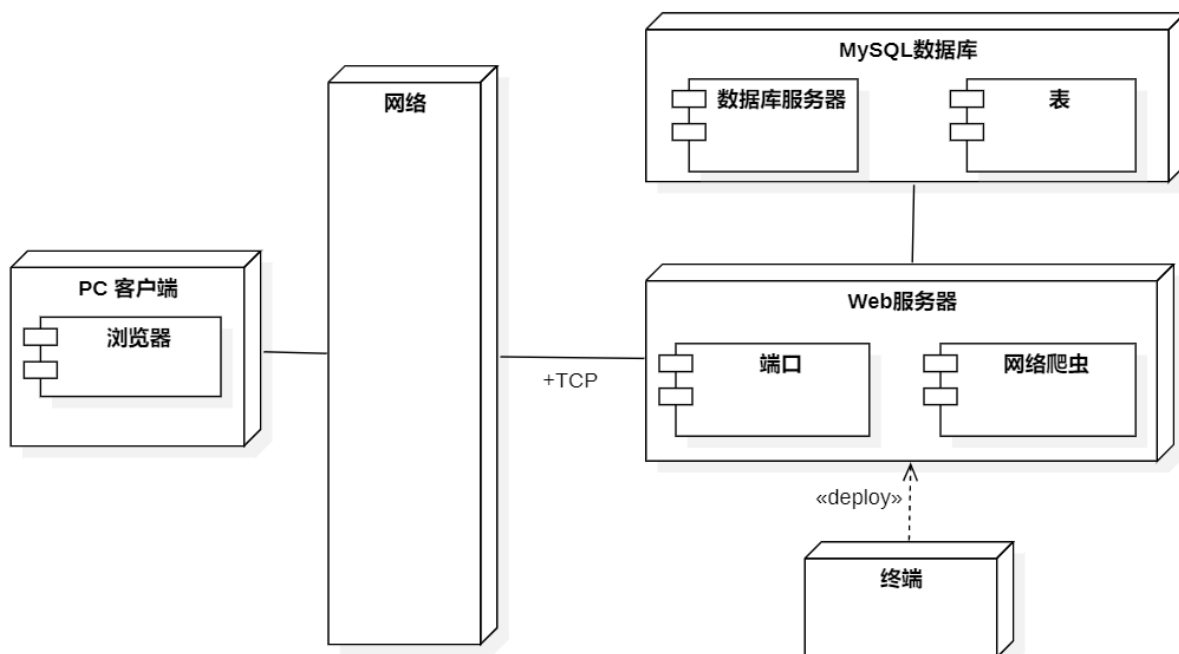
4.1 上下文图



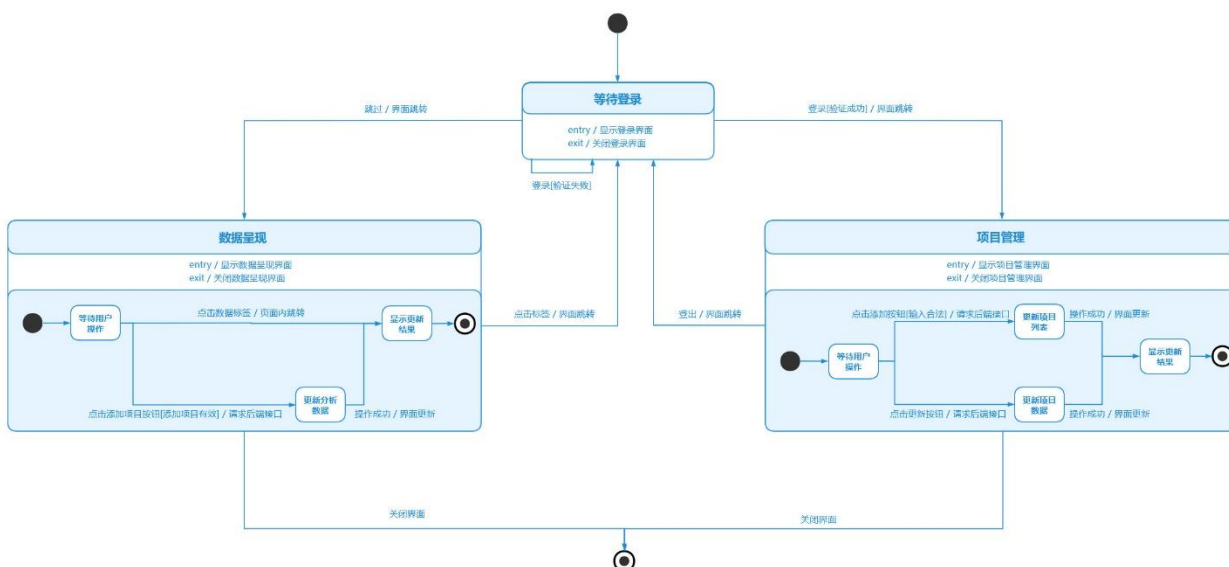
4.2 构件图



4.3 部署图



4.4 状态机图



5 术语表

名词	解释
软件	软件是一系列按照特定顺序组织的计算机数据和指令的集合。
软件工程	软件工程是 1. 将系统化的、严格约束的、可量化的方法应用于软件的开发、运行和维护，即将工程化应用于软件；2. 对在 1.中所述方法所进行的研究。
软件生存周期	软件生存周期是软件的产生直到报废的生命周期，周期内含问题定义、可行性分析、总体描述、系统设计、编码、调试和测试、验收与运行、维护升级到废弃等阶段。
软件过程	软件过程为一个为建造高质量软件所需完成的任务的框架，即形成软件产品的一系列步骤，包括中间产品、资源、角色及过程中采取的方法、工具等范畴。
业务需求	业务需求（Business Requirements）反映了组织机构或客户对系统或产品高层次的目标要求，它们在项目视图与范围文档中予以说明。
用户需求	用户需求（User Requirements）描述了用户使用产品必须要完成的任务，它们在用例模型中予以说明。
功能需求	功能需求（Functional Requirements）定义了开发人员必须实现的软件功能，使得用户能达成他们的目标，从而满足业务需求。
非功能需求	非功能需求（Non-functional Requirements）是从各个角度对系统的约束和限制，反映了应用对软件系统质量等的额外要求。
需求工程	需求工程是指应用已证实有效的技术、方法进行需求分析，确定客户需求，帮助分析人员理解问题并定义目标系统的所有外部特征的一门学科。它通过合适的工具和记号系统地描述待开发系统及其行为特征和相关约束，形成需求文档，并对用户不断变化的需求演进给予支持。
用例	用例（Use Case），或译使用案例、用况，是软件工程或系统工程中对系统如何反应外界请求的描述，是一种通过用户的使用场景来获取需求的技术。每个用例提供了一个或多个场景，该场景说明了系统是如何和最终用户或其它系统互动，也就是谁可以用系统做什么，从而获得一个明确的业务目标。编写用例时要避免使用技术术语，而应该用最终用户或者领域专家的语言。用例一般是由软件开发者和最终用户共同创作的。
用例图	用例图（Use Case Diagram）是用户与系统交互的最简表示形式，展现了用户和与他相关的用例之间的关系。通过用例图，人们可以获知系统不同种类的用户和用例。用例图也经常和其他图表配合使用。
时序图	时序图（Sequence Diagram）是一种 UML 交图。它通过描述对象之间发送消息的时间顺序显示多个对象之间的动态协作。它可以表示用例的行为顺序，当执行一个用例行为时，其中的每条消息对应一个类操作或状态机中引起转换的触发事件。
项目管理	项目管理是通过合理地组织和利用一切可以利用的资源，按照计划的成本和计划的进度，完成一个计划的目标，它包含团队管理、风险管理、采购管理、流程管理、时间管理、成本管理和质量管理等。
UML	UML（Unified Modeling Language）又称统一建模语言或标准建模语言，是始于 1997 年一个 OMG 标准，它是一个支持模型化和软件系统开发的图形化语言，为软件开发的所有阶段提供模型化和可视化支持。
软件质量	软件与明确的和隐含的定义的需求相一致的程度。
质量认证	质量认证也叫合格评定，是国际上通行的管理产品质量的有效方法。
ISO9000	ISO9000 质量保证体系是企业发展与成长之根本，ISO9000 不是指一个标准，而是一类标准的统称，是由 TC176（TC176 指质量管理体系技术委员会）制定的所有国际标准。
开源项目	开源，即开放源码（Open Source）被非盈利软件组织（美国的 Open Source Initiative 协会）注册

	为认证标记，并对其进行了正式的定义，用于描述那些源码可以被公众使用的软件，并且此软件的使用、修改和发行也不受许可证的限制。
Github	GitHub 是一个面向开源及私有软件项目的托管平台，因为只支持 Git 作为唯一的版本库格式进行托管，故名 GitHub。GitHub 于 2008 年 4 月 10 日正式上线，除了 Git 代码仓库托管及基本的 Web 管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。目前，其注册用户已经超过 350 万，托管项目也是非常之多，其中不乏知名开源项目如 PyTorch 等。
代码仓库	代码仓库亦称储存库、资源库、资源库、版本库、代码库、存放库，在版本控制系统中是指在磁盘存储上的数据结构，其中包含了文件、目录以及元数据。
开源社区	开源社区又称开放源代码社区，一般由拥有共同兴趣爱好的人所组成，根据相应的开源软件许可证协议公布软件源代码的网络平台，同时也为网络成员提供一个自由学习交流的空间。由于开放源码软件主要被散布在全世界的编程者所开发，开源社区就成了他们沟通交流的必要途径，因此开源社区在推动开源软件发展的过程中起着巨大的作用。
DevOps	DevOps（Development 和 Operations 的组合词）是一组过程、方法与系统的统称，用于促进开发（应用程序/软件工程）、技术运营和质量保障（QA）部门之间的沟通、协作与整合。