需求规格说明文档

车一晗 181250009

黄婉红 181840096

纳思彧 181250107

王博 181250133

2021年3月25日

**摘要**

本文档为Heap小组在2021年春学期《软件工程与计算三》课程作业迭代二中为项目所撰写的需求规格说明文档。

**目录**

[1 引言 4](#_Toc1471033971)

[1.1 编写目的 4](#_Toc1836206333)

[1.2 对象与范围 4](#_Toc1779831341)

[1.3 参考文献 4](#_Toc1325629124)

[1.4 名词与术语 4](#_Toc1853333090)

[2 总体描述 5](#_Toc1866427542)

[2.1 项目前景 5](#_Toc754066665)

[2.1.1 背景与机遇 5](#_Toc1297437708)

[2.1.2 业务需求 5](#_Toc486606718)

[2.2 项目功能 5](#_Toc781381650)

[2.3 用户特征 6](#_Toc818890145)

[2.4 约束 6](#_Toc2011457039)

[2.5 假设和依赖 6](#_Toc870883399)

[3 详细需求描述 7](#_Toc1836232688)

[3.1 对外接口需求 7](#_Toc75296179)

[3.1.1 用户界面 7](#_Toc635012370)

[3.1.2 通信接口 10](#_Toc1806660647)

[3.2 功能需求 10](#_Toc1274209196)

[3.2.1 用户新增节点、关系 10](#_Toc927029288)

[3.2.1.1 特征描述 10](#_Toc587384431)

[3.2.1.2 刺激/响应序列 11](#_Toc187806558)

[3.2.1.3 相关功能需求 11](#_Toc1811342863)

[3.2.2 用户编辑节点、关系 11](#_Toc511318569)

[3.2.2.1 特征描述 11](#_Toc1649117536)

[3.2.2.2 刺激/响应序列 11](#_Toc1294479370)

[3.2.2.3 相关功能需求 12](#_Toc157943833)

[3.2.3 用户删除节点、关系 12](#_Toc272213539)

[3.2.3.1 特征描述 12](#_Toc952781863)

[3.2.3.2 刺激/响应序列 12](#_Toc1766699409)

[3.2.3.3 相关功能需求 13](#_Toc1808063641)

[3.2.4 用户使用工作域 13](#_Toc1232009237)

[3.2.4.1 特征描述 13](#_Toc341921885)

[3.2.4.2 刺激/响应序列 13](#_Toc14881823)

[3.2.4.3 相关功能需求 13](#_Toc1010696109)

[3.2.5 导入和导出 14](#_Toc173856193)

[3.2.5.1 特征描述 14](#_Toc1423275831)

[3.2.5.2 刺激/响应序列 14](#_Toc176547684)

[3.2.5.3 相关功能需求 15](#_Toc1562008481)

[3.3 非功能需求 17](#_Toc1836439239)

[3.3.1 安全性 17](#_Toc1399315189)

[3.3.2 性能 17](#_Toc1196963226)

[3.3.3 可靠性 17](#_Toc1881617933)

[3.3.4 易用性 18](#_Toc508414209)

[3.3.5 可移植性 18](#_Toc80179250)

[3.3.6 约束 18](#_Toc1100408081)

[3.4 数据需求 18](#_Toc429449403)

[3.4.1 数据定义 18](#_Toc63578654)

[3.4.2 默认数据 18](#_Toc1267065219)

[3.4.3 数据格式要求 19](#_Toc1117292081)

[3.4.4 其他需求 19](#_Toc730995999)

# 1 引言

## 1.1 编写目的

本文档描述了COIN知识关系图谱系统的功能需求与非功能性需求。开发小组的软件系统实现与验证工作都以此文档为依据。

注：除特殊说明之外，本文档所包含的需求都是高优先级需求。

## 1.2 对象与范围

本文档的读者是Heap团队内部的开发和管理人员。

COIN知识关系图谱旨在展示在线编辑可视化的“节点-关系-节点”的关系图，用户能够自主通过在线编辑或导入文件格式的方式编辑知识图谱，同时能够将生成的知识图谱导出为图片或其他文字格式。

## 1.3 参考文献

《项目计划文档》, Heap;

《软件架构设计文档》, Heap;

《项目启动文档》, Heap

《软件工程与计算（卷二）》 骆斌 丁二玉 刘钦

《软件工程与计算（卷三）》 骆斌 刘嘉 张瑾玉 黄蕾

## **1.4 名词与术语**

COIN：COnstructing and vIsualizing kNowledge graph知识图谱可视化系统

# 2 总体描述

## 2.1 项目前景

### 2.1.1 背景与机遇

在众多知识表示方式中，知识图谱作为一种语义网络拥有极强的表达能力和建模灵活性：知识图谱是一种语义表示，可以对现实世界中的实体、概念、属性以及它们之间的关系进行建模；其次，知识图谱是其衍生技术的数据交换标准，其本身是一种数据建模的“协议”，相关技术涵盖知识抽取、知识集成、知识管理和知识应用等各个环节。通过构建知识图谱，可以极大化的辅助相关知识的理解。

### 2.1.2 业务需求

BR1：系统正式上线半个月后，至少获取1000次有效访问。

BR2：系统正式上线一年内，数据库发生的意外性故障少于1次

BR3：系统正式上线一年内，获取效率至少提高50%

## 2.2 项目功能

SF1：对知识图谱进行在线编辑（增、删、改）

SF2：能够导入特定格式文件生成图谱

SF3：能够将图谱导出图片或其他文本格式

SF4：能够对图谱进行特定筛选和搜索，能够显示历史搜索

SF5：能够统计图谱的基础信息

SF6：能够修改图谱展示的样式

SF7：能够以不同排版模式展示图谱

## 2.3 用户特征

|  |  |
| --- | --- |
| 用户 | 系统的访问者，希望通过导入或在线编辑的方式使用知识图谱，考虑到用户计算机水平层次不齐，因此界面应该尽可能简单，做到对用户易用。 |

## 2.4 约束

CON1：系统使用Web界面。

CON2：系统使用迭代式开发。

CON3：在开发中，开发者要提交计划、软件需求规格说明文档、设计描述文档和测试报告

## 2.5 假设和依赖

AE1：用户至少运行在100M带宽的网络环境中，网络通畅。

AE2：不会有高并发场景。

# 3 详细需求描述

## 3.1 对外接口需求

### 3.1.1 用户界面

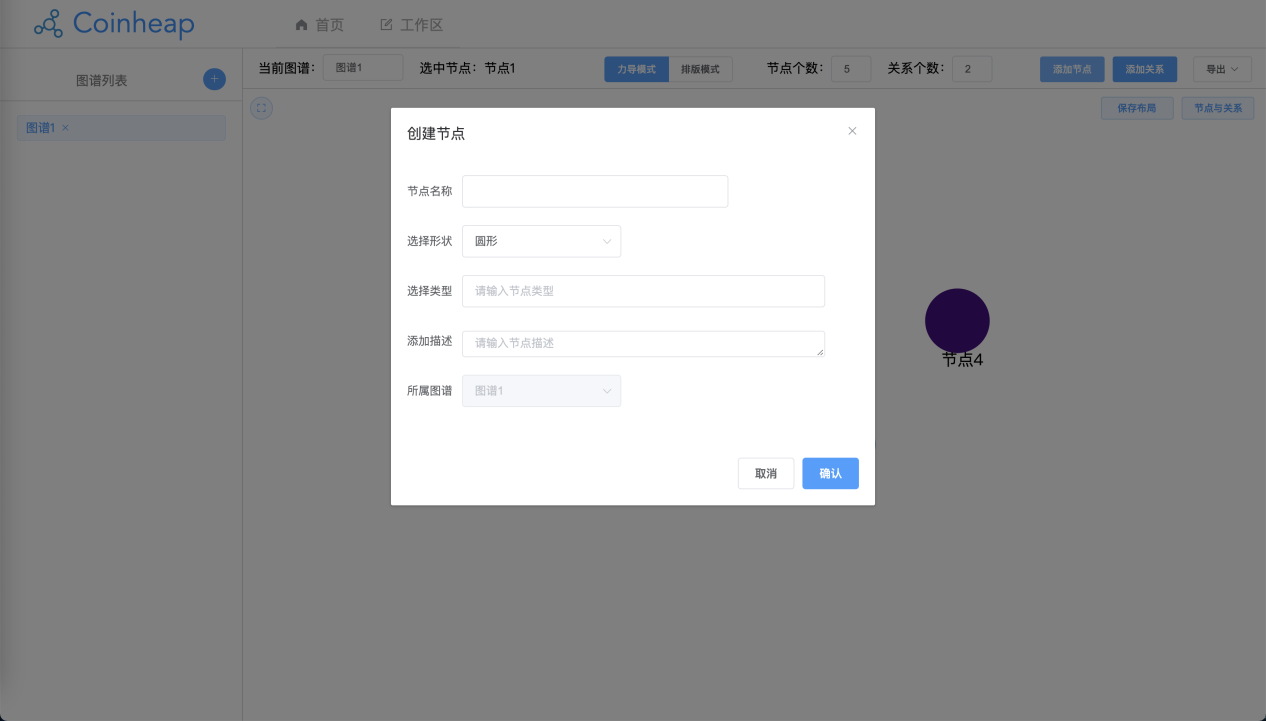
**首页**：

导入文件、创建新图谱



**工作区**：

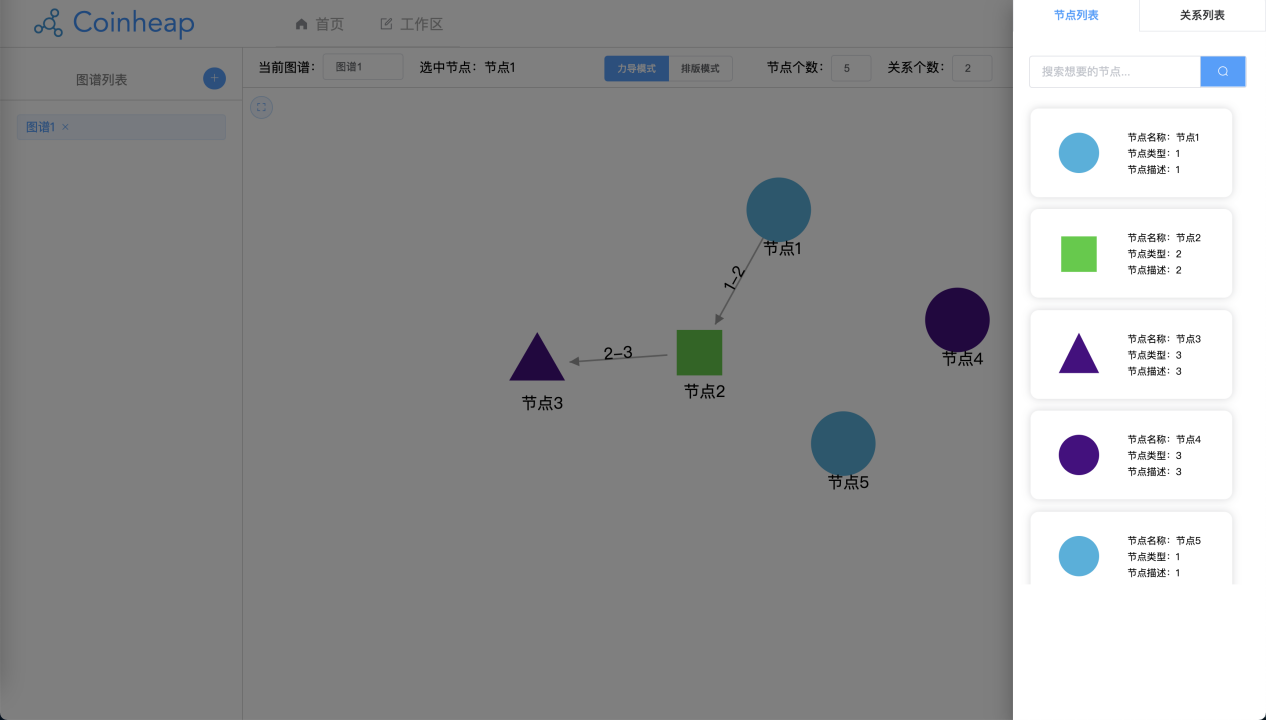
创建节点

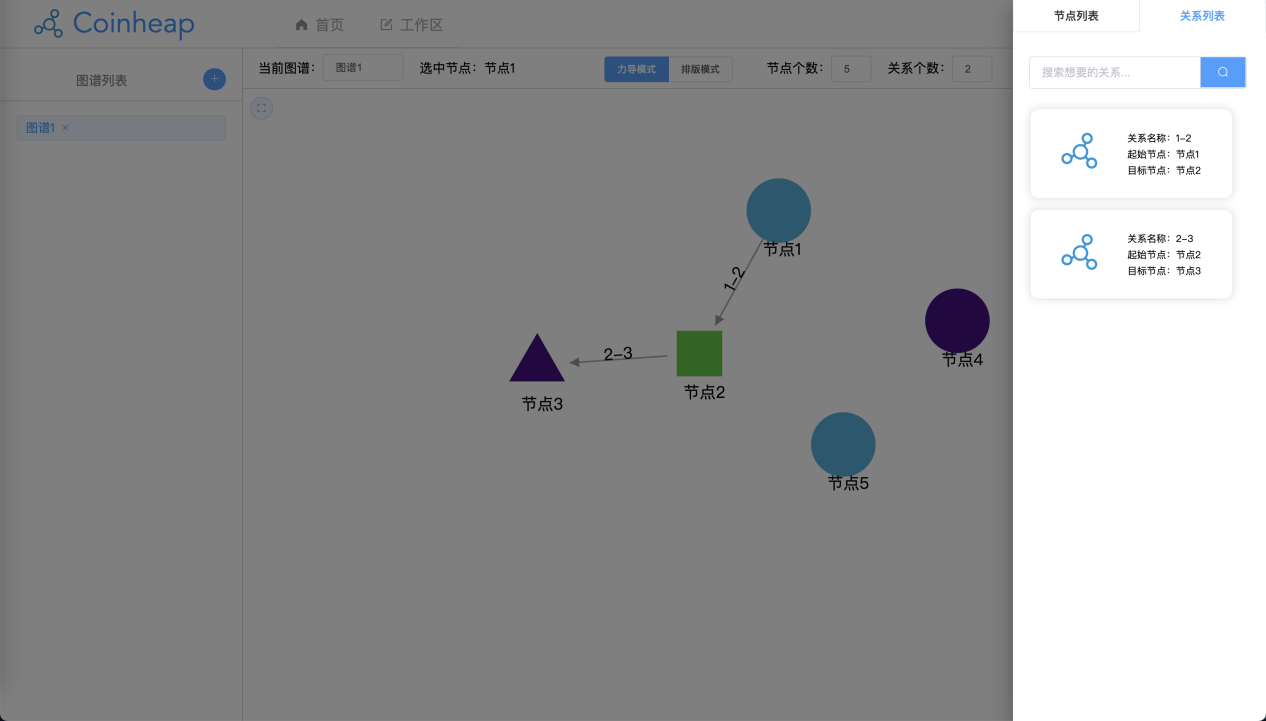


添加关系：

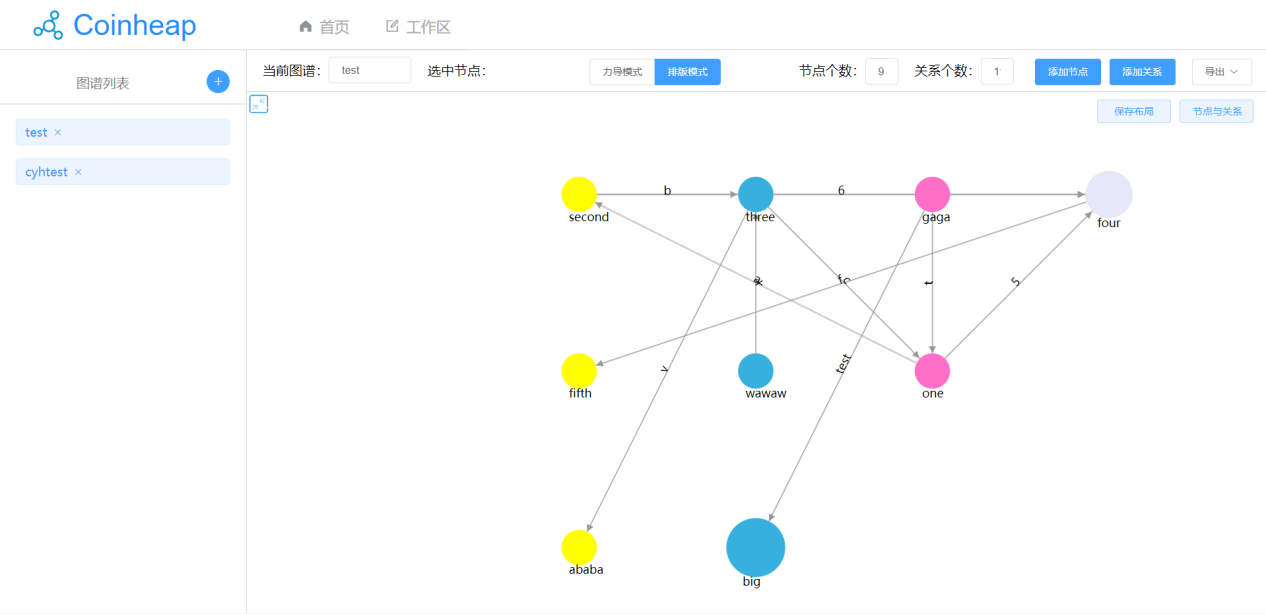


节点与关系统计：





排版模式：



## 

## 3.1.2 通信接口

CI：客户端与服务器使用HTTP的方式进行通信。

## 3.2 功能需求

### 3.2.1 用户新增节点、关系

#### 3.2.1.1 特征描述

用户选择添加节点，输入节点相关信息，或单击对已有的节点创建关系连接

优先级：高

#### 3.2.1.2 刺激/响应序列

刺激：用户选择新增节点

响应：系统返回节点信息一览，请求用户填写

刺激：用户输入节点信息

响应：系统保存节点信息

刺激：用户选择两个节点创建连接

响应：系统为选择节点创建关系

#### 3.2.1.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.createNode | 通过输入节点信息来新增节点 |
| RelationshipController.createLink | 通过输入节点起始终止id来新增关系 |

### 3.2.2 用户编辑节点、关系

#### 3.2.2.1 特征描述

用户点击节点，编辑节点相关信息（如类型），或单击连接，编辑连接信息

优先级：高

#### 3.2.2.2 刺激/响应序列

刺激：用户点击节点或关系

响应：系统弹出编辑按钮

刺激：用户点击编辑

响应：系统请求对应信息键入

刺激：用户确认信息键入

响应：系统为选择的节点或关系更新信息

#### 3.2.2.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.updateNode | 通过编辑节点信息来编辑节点 |
| RelationshipController.updateLink | 通过修改关系信息来编辑关系 |
| EntityController.updateType | 通过修改节点类型来编辑关系 |
| EntityController.countNodeByType | 统计同类型节点个数 |

### 3.2.3 用户删除节点、关系

#### 3.2.3.1 特征描述

用户点击节点或关系，删除节点或关系，删除节点时，会一并删除和该节点有关联的关系

优先级：高

#### 3.2.3.2 刺激/响应序列

刺激：用户点击节点或关系

响应：系统弹出删除按钮

刺激：用户点击删除

响应：系统删除该节点或关系

#### 3.2.3.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.deleteNode | 通过输入节点id删除节点及其附属关系 |
| RelationshipController.deleteLink | 通过输入关系id删除关系 |

### 3.2.4 用户使用工作域

#### 3.2.4.1 特征描述

用户编辑知识图谱时均在工作域内进行，可对工作域进行编辑

优先级：高

#### 3.2.4.2 刺激/响应序列

刺激：用户新增工作域

响应：系统为用户新增工作域

刺激：用户编辑工作域

响应：系统弹出信息编辑，请求修改

刺激：用户删除工作域

响应：系统删除该域的所有节点和关系

刺激：用户使用工作域

响应：系统在界面上展示工作域的节点、关系信息

刺激：用户保存图谱布局

响应：系统将图谱上的节点位置持久化，并保存

刺激：用户选择不同的排版布局

响应：系统根据用户选择，分别采取力导图和排版模式渲染图谱

#### 3.2.4.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.getNodesByDomainId | 获取域内的所有节点信息 |
| RelationshipController.getLinkByDomainId | 获取域内知识图谱关系 |
| DomainController.createDomain | 新建工作域 |
| DomainController.updateDomain | 编辑工作域信息 |
| DomainController.deleteDomain | 删除域及域内的关系节点 |
| DomainController.selectDomainById | 根据id查找域 |
| DomainController.selectAllDomain | 返回所有域 |
| EntityController.getTypes | 得到关系类型 |
| EnitityController.updateXY | 保存节点坐标信息 |

### 

### 3.2.5 导入和导出

#### 3.2.5.1 特征描述

用户可通过导入规范格式文件自动生成图谱，或将已有图谱导出为指定格式文件

优先级：高

#### 3.2.5.2 刺激/响应序列

刺激：用户选择导入csv文件

响应：系统自动创建工作域，并按照文件内容填充关系

刺激：用户选择导出xml文件

响应：系统将图谱导出为同目录下的xml文件

#### 3.2.5.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| FileController.getCsvFile | 导入csv文件生成知识图谱 |
| FileController.exportGraphXML | 导出已有图谱为xml格式文件 |

### 3.2.6 用户获取图谱信息

#### 3.2.6.1 特征描述

用户可直观在图谱中获取图谱基础数据

优先级：高

#### 3.2.6.2 刺激/响应序列

刺激：用户统计图谱节点、关系个数

响应：系统返回该图谱的节点、关系个数

刺激：用户统计图谱特定类别的节点个数

响应：系统返回特定类别的节点个数

刺激：用户查看所有节点类型一栏

响应：系统返回图谱节点信息一栏

#### 3.2.6.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.countNode | 统计节点个数 |
| EntityController.countNodeByType | 统计特定节点个数 |
| RelationShipController.countLink | 统计关系个数 |
| EntityController.getTypes | 返回全类型一栏 |

### 3.2.7 用户搜索图谱

#### 3.2.7.1 特征描述

用户可通过侧边栏对现有节点、关系进行名称或类型的搜索

优先级：高

#### 3.2.7.2 刺激/响应序列

刺激：用户键入字段

响应：系统匹配响应的内容回显给用户

#### 3.2.7.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| EntityController.searchNodeByName | 根据名称搜索节点 |
| RelationShipController.searchLink | 根据名称搜索关系 |
| EntityController.getNodeByType | 根据类型显示节点 |

### 3.2.8 用户筛选图谱

#### 3.2.8.1 特征描述

用户可点击节点只显示与节点有关的图谱

优先级：中

#### 3.2.8.2 刺激/响应序列

刺激：用户选择筛选节点列表

响应：系统根据输入信息筛选图谱，其余部分虚化显示

刺激：用户选择不显示关系

响应：系统只显示节点散点

#### 3.2.8.3 相关功能需求

|  |  |
| --- | --- |
| **名称** | **描述** |
| RelationShipController.getGraphScreen | 根据节点进行图谱筛选 |

## 3.3 非功能需求

### 3.3.1 安全性

Safety1：系统要每隔10s对与其建立链接的网络环境进行探测并反馈报告

Safety2：系统可以在0.1s内识别并拒绝非法访问

### 3.3.2 性能

Performance1：系统的吞吐量可达到 50GB/s

Performance2：系统的响应时间不超过 10ms

Performance3：系统的超时错误率不超过 0.1%

### 3.3.3 可靠性

Reliability1：如果在同客户交互时，网络出现故障，系统不能出现故障

Reliability2：数据库的数据实时备份，在丢失或破损后可自动恢复

### 3.3.4 易用性

Usability1：无需用户使用手册或专业培训即可直接使用系统

Usability2：查询任何一条数据不会跳转超过3个页面，鼠标点击不会超过5次

### 3.3.5 可移植性

Portability1 : 系统或部件可以在4个人月内从windows10系统迁移至centos7操作系统

Portability2：数据库可在5个人月内从MySQL数据库迁移至Oracle数据库

### 3.3.6 约束

C1：系统采用分层模型开发

C2：系统前端使用Vue框架和JavaScript语言进行开发

C3：系统后端使用Java语言进行开发

## 3.4 数据需求

### 3.4.1 数据定义

DR1：系统需要存储用户创建的节点的详细关系信息

DR2：系统需要存储用户三个月内的活动记录

DR3：系统删除的所有数据仍需继续存储三个月的时间，以保证历史数据显示

的正确性

### 3.4.2 默认数据

系统从其他数据源导入新数据中的某一项为空时，默认为null

### 3.4.3 数据格式要求

Format1：日期格式统一精确到年份-月-日 yyyy-mm-dd

Format2：节点、关系的识别id统一为长整型数字

Format3：颜色格式为类似#FFFFFF的字符串格式

Format4：节点、关系的类型格式为-1~10内的整数

Format5：csv文件的格式为三列的“起始节点名,终止节点名,关系名”

### 3.4.4 其他需求

系统安装部署时，需同时导入一批数据，保证系统安装后即可供用户使用。同时该系统投入使用时，需对运维人员进行1个月的专业培训