

COIN需求规格说明文档

1. 目录

COIN需求规格说明文档

- 1. 目录
- 2. 变更记录
- 3. 引言
 - 3.1 目的
 - 3.2 范围
 - 3.3 参考文献
- 4. 总体描述
 - 4.1 背景与机遇
 - 4.1 业务需求
 - 4.2 系统特性
 - 4.3 用户特征
 - 4.4 约束
 - 4.5 假设和依赖
- 5. 详细需求描述
 - 5.1 用户界面
 - 5.2 通信接口
 - 5.3 功能需求
 - 5.3.1 用户导入知识图谱
 - 5.3.1.1 特征描述
 - 5.3.1.2 刺激/响应序列
 - 5.3.1.3 相关功能需求
 - 5.3.2 用户在线编辑知识图谱
 - 5.3.2.1 特征描述
 - 5.3.2.2 刺激/响应序列
 - 5.3.2.3 相关功能需求
 - 5.3.3 用户导出知识图谱
 - 5.3.3.1 特征描述
 - 5.3.3.2 刺激/响应序列
 - 5.3.3.3 相关功能需求
 - 5.4 非功能需求
 - 5.4.1 安全性
 - 5.4.2 性能
 - 5.4.3 可靠性
 - 5.4.4 易用性
 - 5.4.5 约束
 - 5.5 数据需求
 - 5.5.1 数据定义
 - 5.5.2 数据格式要求

2. 变更记录

修改人员	日期	审批	变更原因	版本
冯泊涓	2021-3-7		初稿	v1.0
冯泊涓	2021-3-21		终稿	v1.1

3. 引言

3.1 目的

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

3.2 范围

本项目旨在构建一个系统平台通过在线编辑基本图元以可视化的方式构建个结构完整、业务覆盖全面的知识图谱知识图谱，并初步具备一定知识推理能力。

3.3 参考文献

1. IEEE 标准
2. 软件开发的技术基础
3. 团队与软件开发实践

4. 总体描述

4.1 背景与机遇

知识图谱（Knowledge Graph）的概念由谷歌 2012 年正式提出，旨在实现更智能的搜索引擎，并且于 2013 年以后开始在学术界和业界普及。

目前，随着智能信息服务应用的不断发展，知识图谱已被广泛应用于智能搜索、智能问答、个性化推荐、情报分析、反欺诈等领域。知识图谱以其强大的语义处理能力与开放互联能力，可为万维网上的知识互联奠定扎实的基础，使 Web 3.0 提出的“知识之网”愿景成为了可能。

知识图谱是一种结构化的语义知识库，用于迅速描述物理世界中的概念及其相互关系。通常知识图谱通过对错综复杂的文档的数据进行有效的加工、处理、整合，转化为简单、清晰的“实体,关系,实体”的三元组，最后聚合大量知识，从而实现知识的快速响应和推理。

4.1 业务需求

BR1：系统正式上线半个月后，至少获取1000次用户访问。

4.2 系统特性

SF1：图形化展示知识图谱
SF2：知识图谱的编辑修改
SF3：知识图谱导出

4.3 用户特征

|||| ---- | ---- | |用户|系统的访问者，希望可以直接便捷地浏览、查询、编辑知识图谱，并希望知识图谱具有简洁、美观的特性。

用户计算机水平层次不齐，因此界面设计应该简洁、易用。|

4.4 约束

CON1：系统是基于Web的数据库应用系统
CON2：系统开发使用螺旋模型
CON3：系统架构采用分层模型

4.5 假设和依赖

AE1: 用户至少运行在 5M 带宽的网络环境中，网络通畅。

AE2: 不会有高并发场景。

5. 详细需求描述

5.1 用户界面

5.2 通信接口

用户浏览器与服务器使用 HTTP 协议进行通信。

5.3 功能需求

5.3.1 用户导入知识图谱

5.3.1.1 特征描述

用户选择导入本地知识图谱文件，查看该图谱的可视化信息

优先级：高

5.3.1.2 刺激/响应序列

刺激：用户选择上传

响应：系统显示上传文件弹窗

刺激：用户选择本地文件上传

响应：系统可视化显示该知识图谱

5.3.1.3 相关功能需求

编号	描述
usr.importJSON	上传知识图谱
usr.getAll	获得知识图谱信息

5.3.2 用户在线编辑知识图谱

5.3.2.1 特征描述

用户选择增加、修改或删除知识图谱中的实体和关系，在线编辑知识图谱并查看修改后的知识图谱

优先级：高

5.3.2.2 刺激/响应序列

用户增加/修改知识图谱中的实体/关系

刺激：用户选择添加实体/关系

响应：系统提示输入实体/关系信息，包括实体/关系名称、实体/关系类型

刺激：用户输入实体/关系信息并确定输入

响应：系统显示修改后的可视化知识图谱

用户删除知识图谱中的实体/关系

- 刺激：用户选择删除实体/关系
- 响应：系统提示是否删除该实体/关系
- 刺激：用户确定删除实体/关系
- 响应：系统删除该实体/关系并更新数据库

5.3.2.3 相关功能需求

编号	描述
usr.addEntity	增加新实体
usr.addRelation	增加实体关系
usr.updateEntityName	更改实体名
usr.updateRelationName	更改关系名
usr.deleteEntity	删除实体
usr.deleteRelation	删除关系

5.3.3 用户导出知识图谱

5.3.3.1 特征描述

- 用户选择特定格式导出知识图谱文件
- 优先级：中

5.3.3.2 刺激/响应序列

- 刺激：用户选择导出知识图谱
- 响应：系统提示选择知识图谱导出格式
- 刺激：用户选择文件格式并选择导出知识图谱
- 响应：系统导出知识图谱文件

5.3.3.3 相关功能需求

无

5.4 非功能需求

5.4.1 安全性

Safety1：用户操作过程中数据不被偷窥、窃取、篡改

5.4.2 性能

- Performance1：系统的响应时间不超过 10ms
- Performance2：系统的超时错误率不超过 0.1%

5.4.3 可靠性

Reliability1: 系统提供7*24小时运行

5.4.4 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

5.4.5 约束

C1: 系统采用分层模型开发

C2: 系统使用 Neo4j 数据库存储数据

C3: 系统使用 Java 语言进行开发

5.5 数据需求

5.5.1 数据定义

DR1: 系统需要存储用户上传的知识图谱数据

5.5.2 数据格式要求

Format1: 用户上传知识图谱文件格式为.csv

Format2: 知识图谱实体属性为: 姓名, 格式为String

Format3: 知识图谱关系名不以数字开头, 且不包含特殊字符

Format4: 用户导出知识图谱文件类型为xml或图片