扑克牌游戏系统

实验报告文档

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **描述** | **作者** |
| 2022-06-06 | <1.0> | 初始化 | 尹浩朗 |
| 2022-06-13 | <1.1> | 细化1 | 尹浩朗 |
| 2022-06-20 | <1.2> | 细化2 | 尹浩朗 |
| 2022-07-08 | <1.3> | 细化3 | 尹浩朗 |
| 2022-07-13 | <2.0> | 终版 | 尹浩朗 |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写和缩写 4

1.4 参考资料 5

1.5 概述 5

2. 体系结构目标和约束 5

2.1 迭代计划 5

2.2 质量约束 6

2.3 实现约束 7

2.3.1 服务端设计约束 7

2.3.2 客户端设计约束 7

2.3.3 网络通信设计约束 7

2.4 免费开源构件 7

3. 用例视图 7

4. 逻辑视图 11

4.1 包图 11

4.2 系统顺序图 12

4.3 领域模型 13

5. 流程视图 14

5.1 类图 14

5.2 交互图 15

5.2.1 检索词提示 15

5.2.2 热门搜索词 16

5.2.3 智能检索初始化 17

5.2.4 智能检索 17

5.2.5 图谱检索 18

6. 部署视图 19

7. 实施视图 19

8. 质量 19

8.1 安全性 19

8.2 可靠性 21

8.2.1 日志和错误处理 21

8.2.2 系统性能 21

8.2.3 效益指标 21

8.2.4 业务指标 21

实验报告文档

# 简介

## 目的

为加快推动数据资源共享，推动建设事项咨询数据中心，提升事项咨询的精准服务能力和智能应用水平。《广东省人民政府关于印发广东省“数字政府”建设总体规划（2018-2020年）的通知》要求：推行网上办事智能在线咨询服务，建立公众参与机制，鼓励引导群众分享办事经验，将在线客服插件化，为各政务服务页面或APP提供客服入口，接入微信或短信平台，实现服务结果主动提醒。

## 范围

智能咨询平台项目拟解决市民与事项指南对接的最后一公里问题，对于事项指南信息的精准导航和查找具有实质性意义，同时对于解决市民长期以来在这方面的投诉和抱怨具有重要意义。具体实现如下目标：（1）有助于减少咨询电话，提升业务咨询效率，有效提升咨询业务的公共服务水平。（2）将国家的相关惠民、惠企政策及时准确地推送给市民和企业，为办事群众和企业在广州市人社局政务服务事项中快速匹配到符合自身资格条件的事项，随时随地地了解事项详情，并找到对应的业务办理入口。

## 定义、首字母缩写和缩写

|  |  |  |  |
| --- | --- | --- | --- |
| 中文名 | 英文名 | 缩写 | 释义 |
| SWTextrank | SWTextrank | SWTR | 一种提取文本关键词的方法 |
| 知识图谱 | KnowledgeGraph | KG |  |
| 词向量 | WordVector | WordVector | 将词语用多维向量表示，可以用于计算相似度 |
| BERT | BERT | BERT | 一种NLP模型，在本项目中用于提供词向量 |
| Neo4j | Neo4j | Neo4j | 一种图数据库 |
| MongoDB | MongoDB | MongoDB | 一种NoSQL数据库 |
| 倒排索引表 | Inverted Table |  | 倒排索引表 |
| HanLP | HanLP | HanLP | 一种实体提取和关系提取的NLP技术 |
| 字典树 | Trie | Trie | 一种数据结构，用于前缀匹配 |
| 实体-关系-实体 | ERE | ERE | 即Entity-Relation-Entity三元组，知识图谱中的概念 |
| 事项 | Item | Item | 事项指南+事项规则=事项 |
| 事项指南 | itemGuide | itemGuide | 政府办事的指南，包括办理地点、办理时间、申报材料等 |
| 事项规则 | rule | rule | 事项的分类路径，例如个人业务-就业创业 |
| 区划 | region | region | 例如白云区、小谷围街道等 |
| MERN | MERN | MERN | MERN(MongoDB、Express、React、Node.js)技术栈 |
| 分词 | word segmentation | word segmentation | NLP技术的一种，将句子中的词语分割出来 |
| 规则库 | ruleBase | ruleBase | 事项规则的集合 |
| 倒排索引 | inverted index | inverted index | 由属性值来确定记录的位置的索引，用于检索 |

## 参考资料

## 概述

# 体系结构目标和约束

## 迭代计划

|  |  |
| --- | --- |
| 时间 | 迭代目标 |
| 2022.05.31- 2022.06.06 | 初始化阶段，完成系统总体用例图、设想、补充性规格说明、数据字典、迭代计划等 |
| 2022.06.07- 2022.06.13 | 细化每个用例并编写用例规约，进行智能检索系统前台开发。产出领域模型、系统顺序图、UML类图等迭代1制品。 |
| 2022.06.14- 2022.06.21 | 应用GoF设计模式，进行智能检索系统后台开发 |
| 2022.06.22- 2022.06.29 | 设计UML状态图并完善总体设计模型，进行信息检索功能开发 |
| 2022.06.30- 2022.07.06 | 智能导航功能开发 |
| 2022.07.06- 2022.07.13 | 基于知识图谱的信息检索功能开发 |

## 质量约束

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **一级指标** | **二级指标** | **三级指标** | **指标值** |
| **1** | 系统性能 | 系统并发数 | 系统支持的峰值用户并发数（个） | 100 |
| **2** | 响应时间 | 系统对用户操作的响应时间（秒） | 3 |
| **3** | 效益指标 | 数据交换准确程度（%） | 系统按照标准接收数据量÷全部提供数据量×100% | 99% |
| 4 | 非正常停机率（%） | 非正常停机时间÷全年运行时间×100% | <5% |
| 5 | 业务指标 | 服务能力 | 系统平均用户数量×用户平均使用时间×系统在线时长（万件/每季度） | 60 |
| 6 | 好评率 | 用户对查询系统的好评数÷系统查询使用量×100% | 85% |

注： 系统平均用户量按照系统并发数100，反推出系统平均用户数量大于70

用户平均使用时间按照10分钟计算

服务能力=4（月/季度）\*30（天）\*12（小时）\*(60分钟/10分钟)\*70 (件）=60 （万件/每季度）

## 实现约束

### 服务端设计约束

服务器操作系统选用CentOS7/Ubuntu18.04之一

Web服务器使用Nginx的Express框架和Python的Flask框架

数据库使用MongoDB和Neo4j

### 客户端设计约束

支持Chrome/Edge等主流浏览器

前端使用React框架，Redux状态管理器

### 网络通信设计约束

使用Http/Https进行通信

使用Restful风格实现Web Service接口

## 免费开源构件

1. MongoDB
2. React
3. Redis
4. Python
5. NodeJS

# 用例视图

|  |  |
| --- | --- |
| 用例名称 | 事项智能检索 |
| 用例编号 | SEARCH001 |
| 用例简述 | 市民输入关键词搜素事项 |
| 前置用例 | 无 |
| 后置用例 | 热门搜索展示、检索词展示、图谱检索、智能检索 |
| 用例图 |  |
| 主要流程 | 1. 系统显示热门搜索词 2. 市民选择检索方式 3. 市民输入检索词 4. 系统根据市民输入的检索词提示检索词补全 5. 市民点击确认搜索按钮 6. 系统根据检索词和检索方式，返回检索结果。 7. 市民选择想要查看的事项指南，点击自己所在的地区 8. 系统跳转到对应的事项指南详情页面 |
| 替代流程 | 1. 检索词存在特殊字符，提示错误 2. 检索结果为空，显示“未找到相关的事项指南” 3. 用户点击热门搜索词，系统以热门搜索词为检索词执行一次事项检索并返回结果 |
| 业务规则 | 1. 检索词只能包含数字、字母、汉字，不能包含其他特殊字符 2. 热门搜索词是一定时间内的检索词按被检索次数排序 3. 同名事项在检索结果中合并，并列出存在事项的地区 |
| 其他 | 无 |

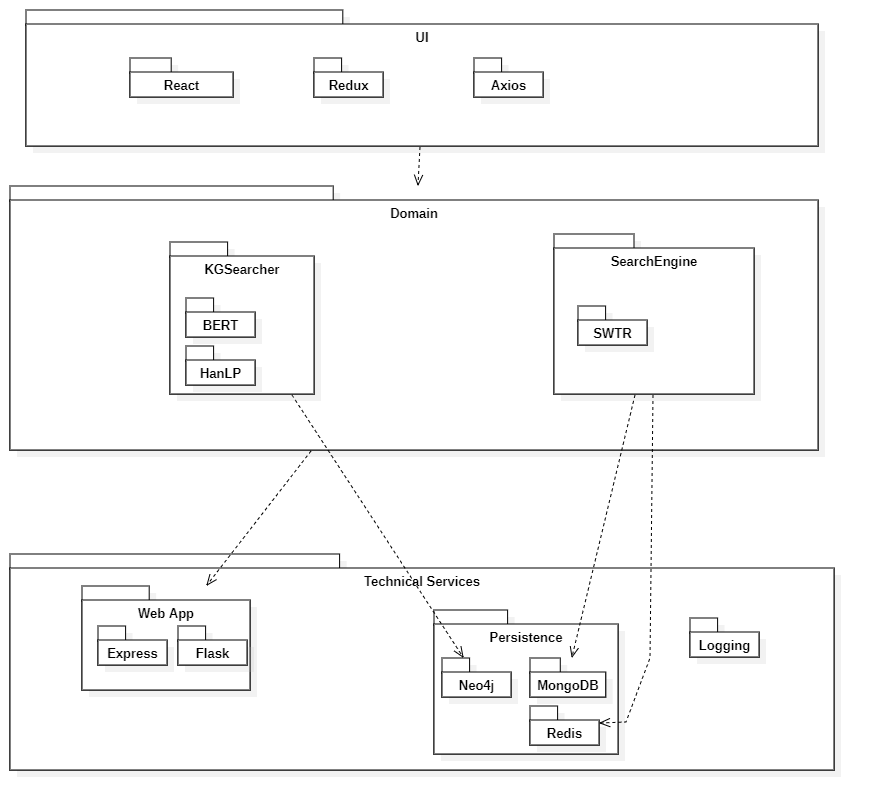
|  |  |
| --- | --- |
| 用例名称 | 智能检索 |
| 用例编号 | SEARCH002 |
| 用例简述 | 市民使用智能检索方式检索事项 |
| 前置用例 | 事项检索系统 |
| 后置用例 | 无 |
| 用例图 |  |
| 主要流程 | 1. 市民输入关键词 2. 市民选择智能检索方式并请求检索 3. SearchEngine对关键词进行分词 4. SearchEngine用分词在倒排索引中检索文档 5. SearchEngine返回检索结果，即事项指南列表 6. 前端显示检索结果 |
| 替代流程 | 无 |
| 业务规则 | （1）SearchEngine应在初始化时建立辅助数据库和倒排索引 |
| 其他 | 无 |

|  |  |
| --- | --- |
| 用例名称 | 图谱检索 |
| 用例编号 | SEARCH003 |
| 用例简述 | 市民使用图谱检索方式检索事项 |
| 前置用例 | 事项检索系统 |
| 后置用例 | 无 |
| 用例图 |  |
| 主要流程 | 1. 市民输入关键词 2. 市民选择图谱检索方式并请求检索 3. HanLP对市民输入的关键词进行实体抽取和关系抽取 4. BERT将（3）中的实体和关系转化为词向量 5. KGSearcher以词向量相似度为标准在数据库中检索相关的实体和关系 6. KGSearcher返回检索结果，即事项指南列表 7. 前端将检索结果显示给用户 |
| 替代流程 |  |
| 业务规则 | 无 |
| 其他 | 无 |

|  |  |
| --- | --- |
| 用例名称 | 检索词提示 |
| 用例编号 | SEARCH004 |
| 用例简述 | 市民输入检索词时给予提示 |
| 前置用例 | 事项检索系统 |
| 后置用例 |  |
| 用例图 |  |
| 主要流程 | （1）市民输入关键词  （2）系统在字典树中检索以关键词为前缀的检索词  （3）系统返回推荐检索词列表  （4）前端在输入框的下拉框显示推荐检索词列表 |
| 替代流程 | （1）推荐检索词列表为空，显示无 |
| 业务规则 | （1）SearchEngine初始化时应建立检索词库和字典树 |
| 其他 | 无 |

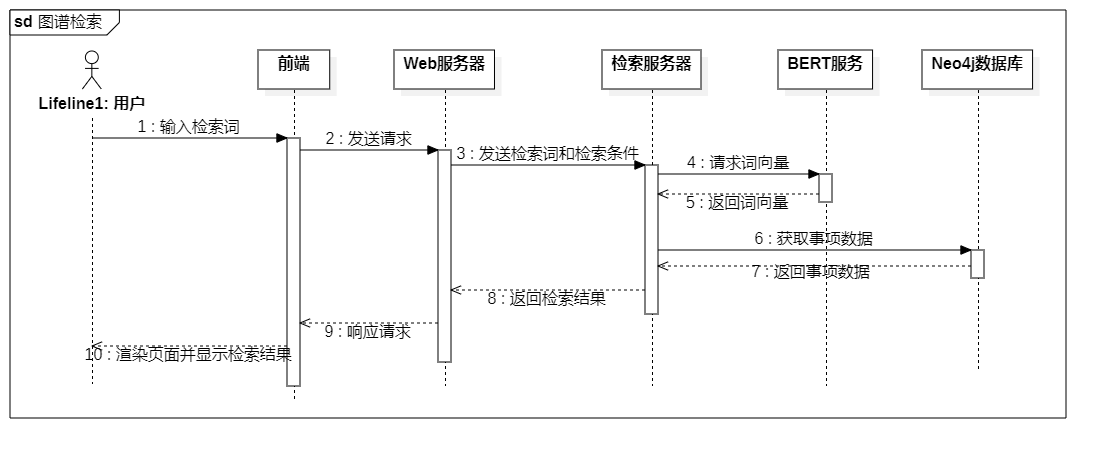
# 逻辑视图

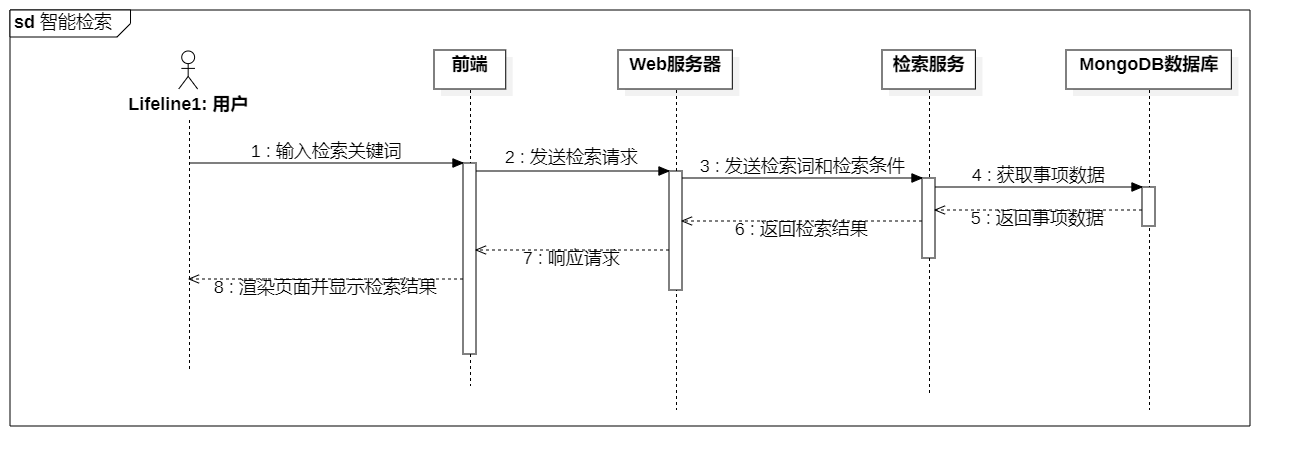
## 包图



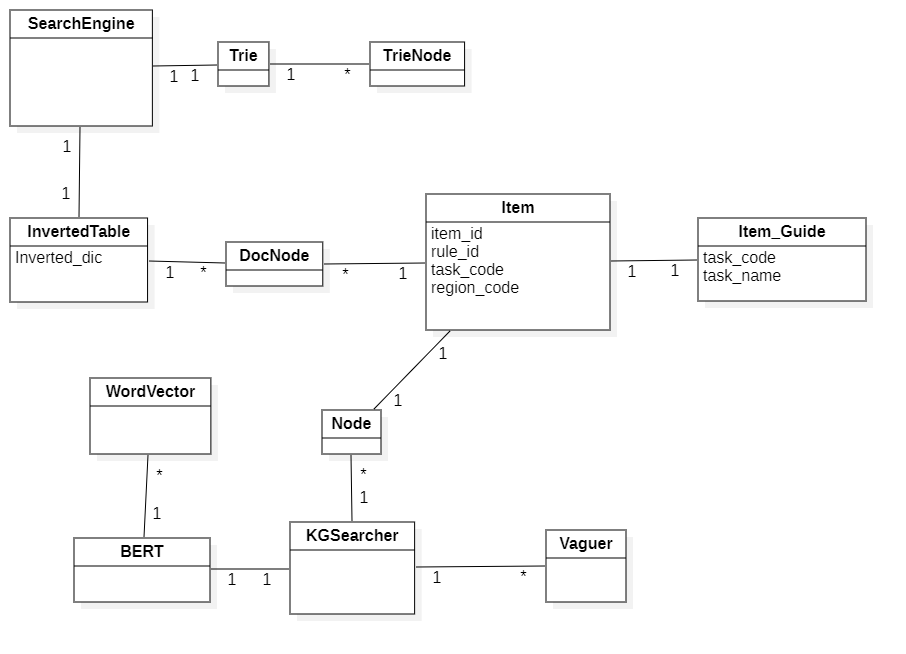
|  |  |
| --- | --- |
| 包名 | 注释 |
| React | 一个基于js的web前端框架 |
| Redux | 一种前端状态管理容器 |
| Axios | 一种前端发送Http请求的框架，用于与后端交互 |
| KGSearcher | 图谱检索模块 |
| HanLP | 实体抽取、关系抽取模块 |
| BERT | 词向量模块 |
| SearchEngine | 智能检索模块 |
| SWTR | 一种基于TextRank优化的关键词提取算法 |
| Web App | Web服务器 |
| Express | 基于NodeJS的一种web框架，用于提供智能检索接口 |
| Flask | Python的一种web框架，用于提供图谱检索后端接口 |
| Persistence | 持久化层 |
| Neo4j | 一种图数据库 |
| MongoDB | 一种NoSQL数据库 |
| Redis | 一种NoSQL数据库，一般用于缓存 |
| Logging | 后端日志记录 |

## 系统顺序图





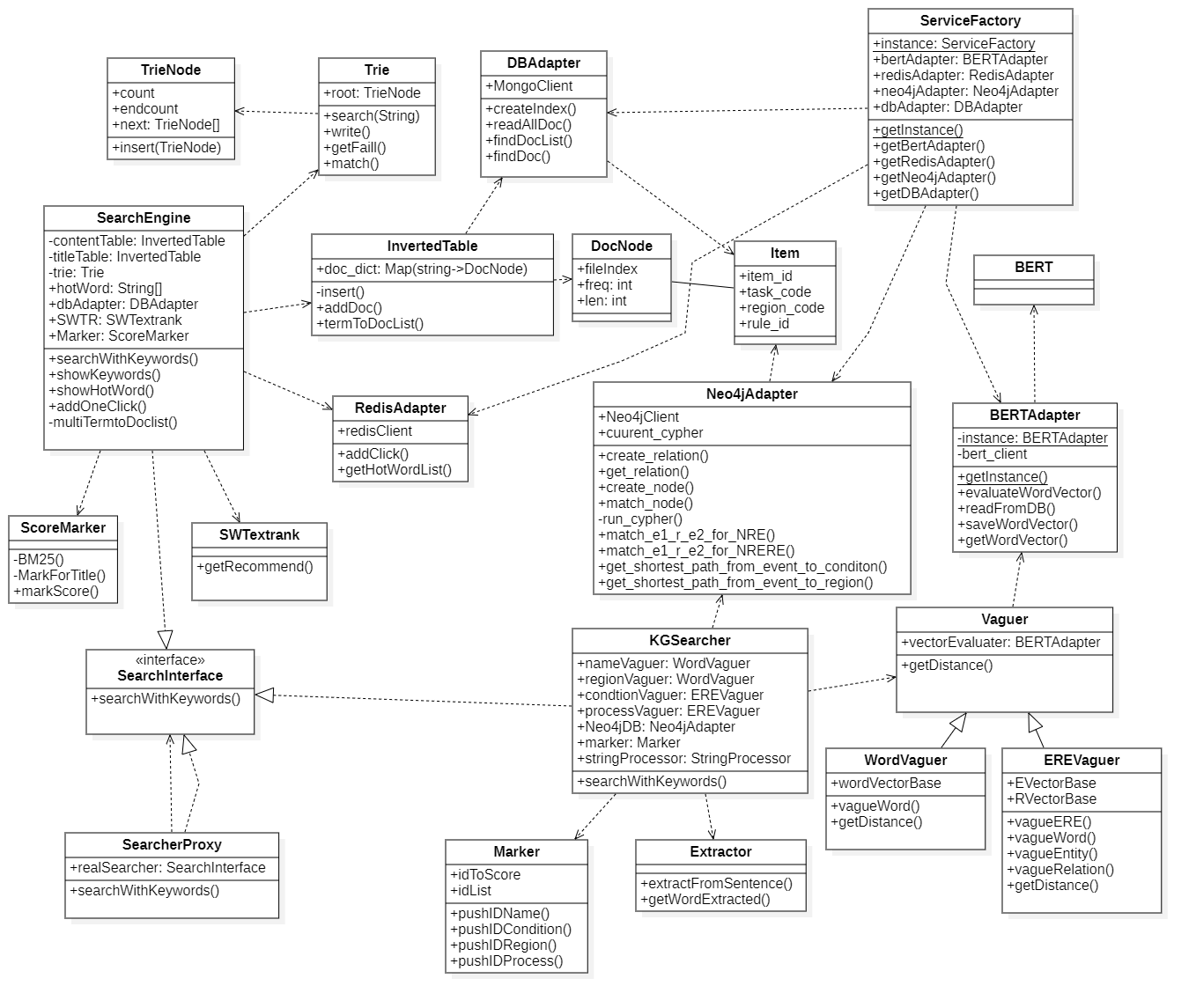
## 领域模型



|  |  |
| --- | --- |
| 类名 | 注释 |
| KGSearcher | 图谱检索核心类，控制器 |
| BERT | 用于提供词向量 |
| WordVector | 词向量 |
| Vaguer | 实体抽取、关系抽取 |
| Item | 事项 |
| ItemGuide | 事项指南 |
| InvertedTable | 倒排索引表 |
| SearchEngine | 智能检索核心类，控制器 |
| Trie | 字典树 |

# 流程视图

## 类图

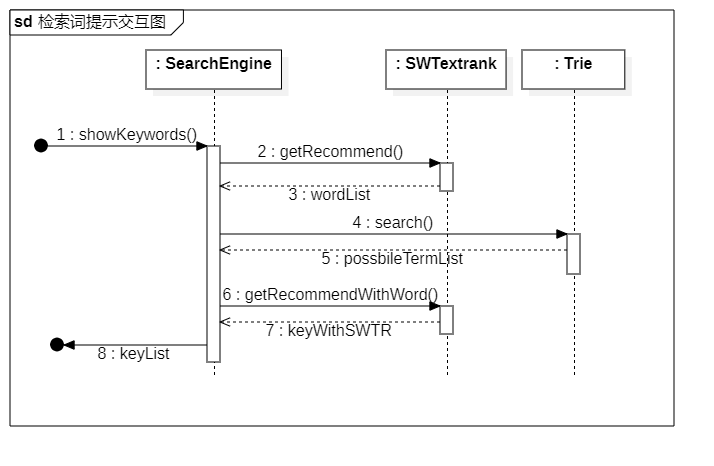


GoF设计模式：

1. 代理模式，SeacherProxy作为代理人，将智能检索和图谱检索的请求分发到SearchEngine和KGSearcher中，详见5.2.4和5.2.5的交互图
2. 适配器模式，与数据库的交互和BERT的交互使用Adapter作为中介，防止变异
3. 单实例类。Adapter和工厂类在全局只有一个实例
4. 工厂模式，所有Adapter由一个单实例的工厂类创建，其他类从工厂类中获取需要的Adapter，降低了耦合度。

## 交互图

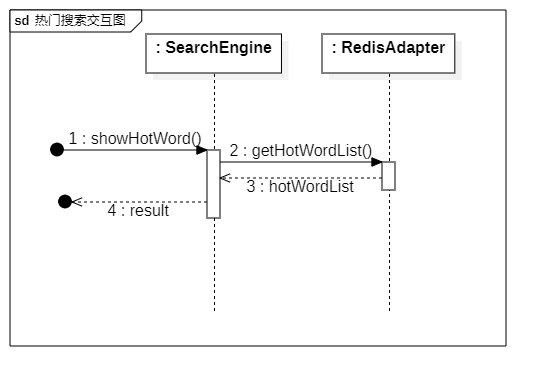
### 检索词提示



补充说明：

1. 检索词传入SearchEngine，先通过SWTextRank提取关键词，返回关键词列表
2. 对每一个关键词，在字典树中查找以关键词为前缀的词，返回可能词列表
3. 将可能词列表传入SWTR计算权重，按权重降序排列并返回

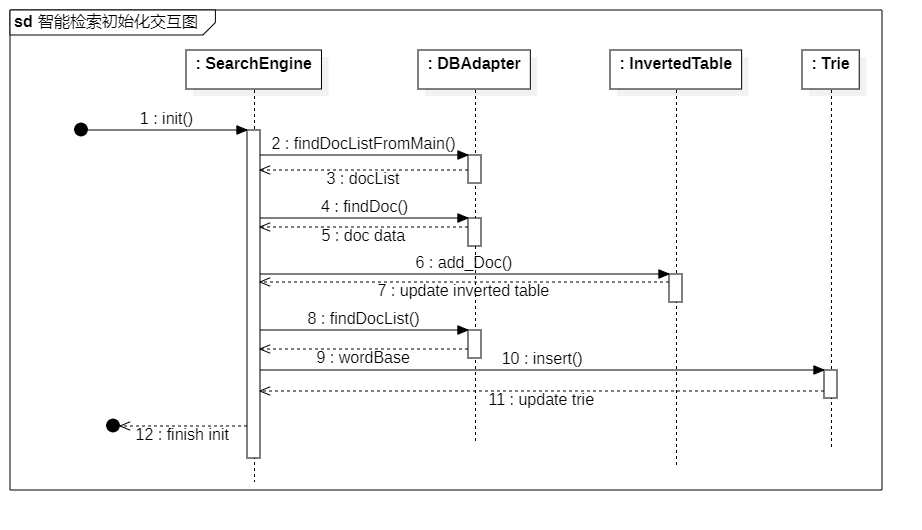
### 热门搜索词



补充说明：

1. 检索词和检索次数以key-value形式存放在Redis中
2. SearchEngine从Redis中取出检索词，并按照近三日的检索次数排序
3. 最终返回热门搜索词列表

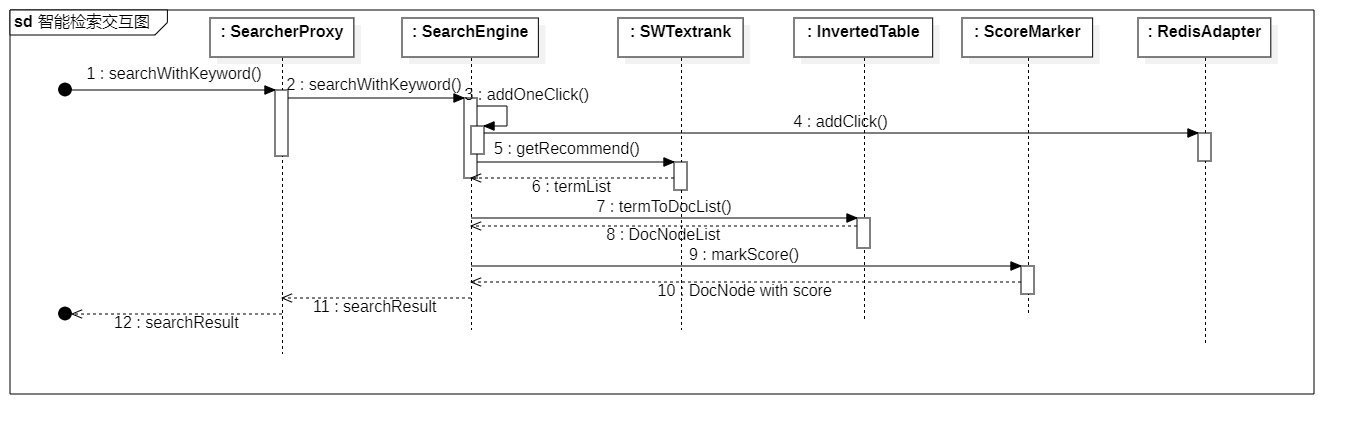
### 智能检索初始化



补充说明：

1. SearchEngine通过DBAdapter从主数据库中获取所有的事项指南，主数据库中没有分词信息
2. 通过（1）中的事项指南ID在辅助数据库中查找事项指南的分词结果并返回
3. 每个分词映射一个倒排索引表，将事项指南以分词为索引添加到对应的倒排索引表中
4. SearchEngine通过DBAdapter从主数据库中获取词库，并建立字典树

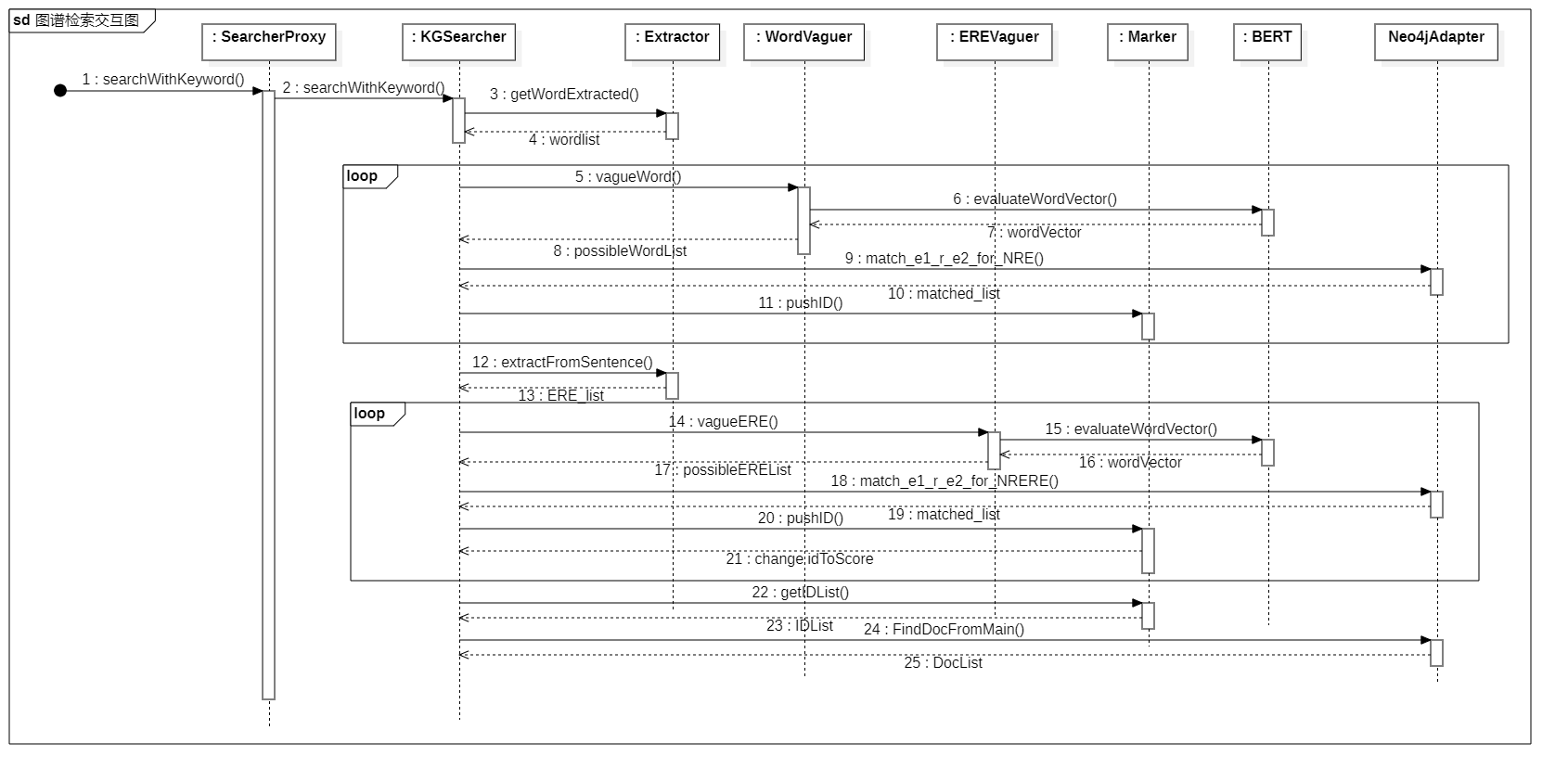
### 智能检索



补充说明：

1. SearchEngine调用addOneClick，将Redis中的检索词检索次数+1
2. SearchEngine将传入的检索词通过SWTR分词，获得分词列表
3. 以分词为索引查找倒排索引表，并通过倒排索引表检索事项，返回事项列表
4. 将返回的事项通过ScoreMarker进行打分，将事项指南按分数降序排列并返回

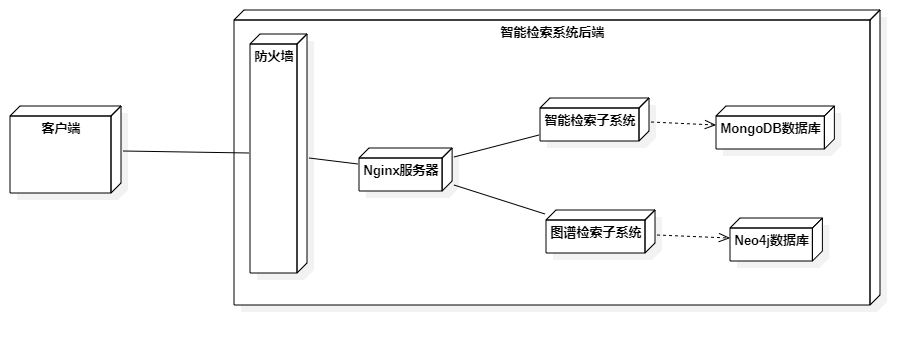
### 图谱检索



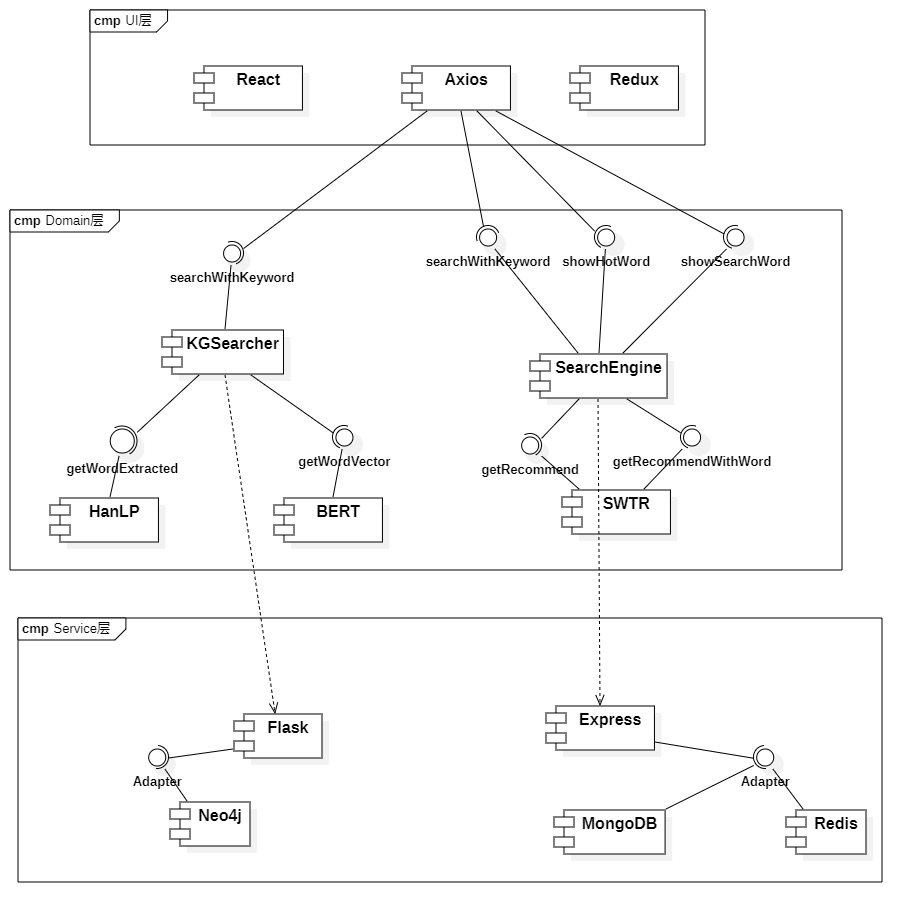
补充说明：

1. KGSearcher将检索词在Extractor中进行实体抽取，获得实体列表
2. 使用WordVaguer类对实体进行模糊化，词向量由BERT提供，返回实体模糊化后的模糊实体列表。例如实体“补贴”模糊化后可能是“补助”、“津贴”等
3. 通过Neo4jAdapter，将模糊实体列表在图数据库Neo4j中检索相关的实体和关系并返回
4. 将（3）中返回的数据传入Marker进行相关度评分
5. KGSearcher将检索词在Extractor中进行关系抽取，获得ERE列表
6. 使用EREVaguer类对关系进行模糊化，词向量由BERT提供，返回模糊关系列表
7. 通过Neo4jAdapter，将模糊关系列表在图数据库Neo4j中检索相关的实体和关系并返回
8. 将（7）中返回的数据传入Marker进行相关度评分
9. Marker以事项ID为索引，映射一个评分，并以评分降序排列。KGSearcher从Marker获取所有的事项ID，并通过Neo4jAdapter在图数据库中进行检索，返回事项列表

# 部署视图



# 实施视图



补充说明：

1. UI层主要由React，Redux，Axios组成，React是前端框架，Redux是全局状态容器，Axios用于前后端Http通信
2. Domain层，KGSearcher提供searchWithKeyword接口，即根据输入的检索词检索事项。KGSearcher的功能依赖HanLP提供的分词接口getWordExtracted和BERT的词向量接口getWordVector。SearchEngine提供searchWithKeyword，showHotWord，showSearchWord三个接口，分别为根据输入的检索词检索事项，显示当前热门搜索词，检索词提示三个功能。SearchEngine依赖SWTR提供词向量接口getRecommend和词相似度接口getRecommendWithWord
3. Service层，Flask为KGSearcher提供Web服务，Flask和Neo4j数据库通过Adapter接口进行交互，同理，Express为SearchEngine提供Web服务，通过Adpater和Redis以及MongoDB进行交互

# 质量

## 安全性

1. 物理和环境安全

物理和环境安全是整个信息系统安全的基础，只有保证物理层可用性，才能保证整个信息系统的可用性，进而提高整个系统的抗破坏性。物理安全需求主要包括：防盗窃、防破坏、防雷击、防火、防水和防潮、防静电、温湿度控制、电力供应、电磁防护等。

1. 网络和通信安全

1. 核心网络设备应具有高性能以满足业务高峰时的处理要求，并采用冗余配置，避免关键节点出现单点故障。网络结构应实现安全域划分，并在此基础上实现不同区域安全、可控的逻辑隔离。

2. 避免将重要安全域部署在网络边界处且直接连接外部信息系统，重要安全域与其他安全域之间采取可靠的技术隔离手段。

3. 对进出各安全域的信息和数据进行严格的控制，防止对安全域的非法访问，能够检测和清除网络边界的恶意代码。

4. 在各个重要安全域内，能及时发现和响应各种网络攻击与破坏行为。应对网络系统中的网络设备运行状况、网络流量、用户行为等进行日志记录和安全审计。

5. 对于安全域之间交互的信息和数据，保护其完整性、可用性、保密性，防止在传输过程中被窃取、篡改和破坏。

6. 数据跨网交换需满足与其他网络之间的数据协同安全保障需求。

（3）设备和计算安全

1. 由于目前的计算机架构缺乏相应的安全机制，整个计算平台很容易被攻击而进入一个不可控状态。所以，必须从底层硬件、操作系统和应用程序等方面采取综合措施，建立可信计算环境，才能从整体上提高设备和计算安全性。

2. 有关操作系统及基础软件应能及时升级、安装安全补丁，操作系统启动的服务应最小化，启用的账户权限应最小化，并进行分权管理。

3. 设置严格的用户登录策略和密码管理策略；设置认证失败处理和超时退出措施。启用操作系统日志审核并同时将审核日志传送到第三方日志服务器进行保存。

4.使用加密通道对主机操作系统进行远程管理，并设置严格的IP访问控制策略。

（4）应用和数据安全

1. 针对流行的WEB应用攻击行为，从边界专业防护技术措施和加强应用漏洞检查等措施进行综合防护。

2. 启用应用系统的日志审核并同时将审核日志传送到第三方日志服务器进行保存，使用密码技术的完整性功能来对日志记录进行完整性保护。

3. 使用加密通道来保护登录用户的业务操作，并设置严格的IP访问控制策略。重要应用系统需要设置双因素认证，对应用系统的运行状态进行集中监控。

4. 采用加密或其他有效措施实现系统管理数据、鉴别信息和重要业务数据传输及存储的保密性。能够检测到系统管理数据、鉴别信息、重要业务数据、重要审计数据、重要配置数据、重要视频数据、重要用户信息、重要可执行程序等传输及存储的完整性收到破坏并采取恢复措施。使用密码技术的完整性功能来保证业务应用系统访问控制策略、数据库表访问控制信息和重要信息资源敏感标记等信息的完整性。

5. 加强对文件、数据的保密性检测和审查，防止由于疏忽、误操作或其他原因导致保密信息或其他敏感信息外泄。

6. 应提供本地数据备份与恢复功能，数据备份满足相关要求和规范。

## 可靠性

### 日志和错误处理

后端访问记录和错误都进行持久化存储

### 系统性能

系统并发数，即系统支持的峰值用户并发数需要>=100

系统响应时间，即系统对用户操作的响应时间不高于3秒

### 效益指标

数据交换准确率，即系统正确接收的数据量/全部发送的数据量不小于99%

非正常停机率，即非正常停机时间/全年运行时间小于5%

### 业务指标

服务能力，即系统平均用户数量×用户平均使用时间×系统在线时长，不低于60万件/每季度

好评率，用户对查询系统的好评数/系统查询使用量，不小于85%。