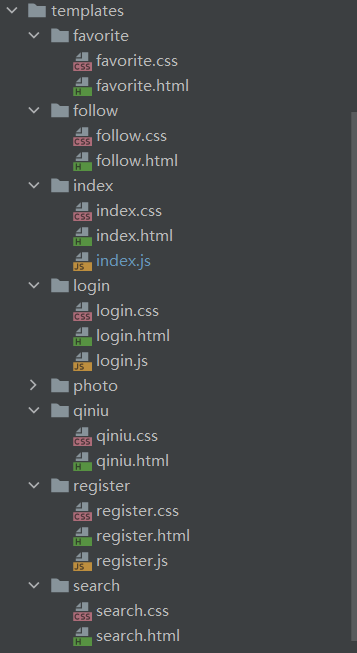
架构设计文档

项目介绍视频:https://url.v.tencent.com/164g4OZK （因为文件太大，所以以在线视频形式分享）

1. **分工**：个人组队/唐杨
2. 本项目为前后端不分离项目，运行方便。同时也有着前后端分离的思想,比如使用fetch函数调用后端接口。
3. **前端**：



3.1 前端技术栈

·HTML/CSS/JavaScript

·使用fetch API与后端进行数据交互

·部分页面使用Thymeleaf进行服务器端渲染

3.2 功能与界面

3.2.1 视频播放

使用Video.js实现视频播放器功能。

实现功能：播放、暂停、进度条控制。



3.2.2 内容分类展示

动态展示不同分类的视频内容。

与后端交互获取分类数据。



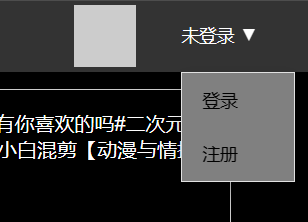


3.2.3 用户交互

视频收藏、点赞、分享。



用户登录与注册。



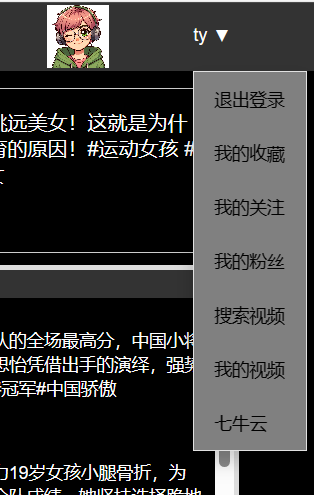




视频推荐根据用户喜好。



提供用户选择下拉框



关注和粉丝展示:



点击我的收藏，我的视频以及用户头像或姓名显示对应视频：



搜索视频：

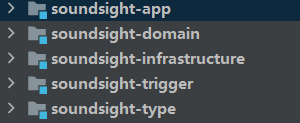


点击”七牛云”:



具体功能可见视频演示

1. **后端：**



4.1 技术选型

Spring Boot

MyBatis

Redis

ElasticSearch

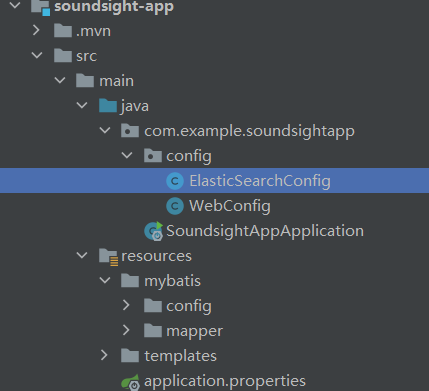
MySQL

4.2 架构设计

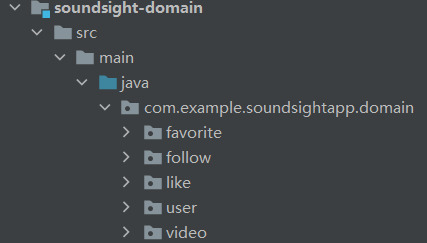
4.2.1 DDD架构实践

项目被划分为应用层，领域层，基础设施层，触发器层，通用层。

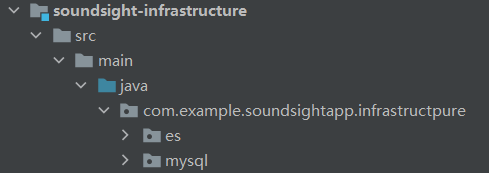
· 应用层:提供es,web,mybatis等相关配置,存放mybatis的mapper文件以及前端页面。



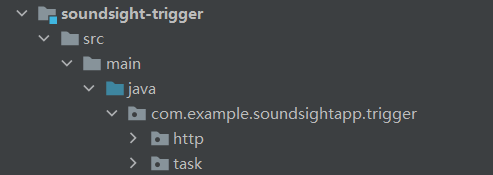
· 领域层：划分了收藏，关注，点赞，用户，视频五个邻域，便于管理。



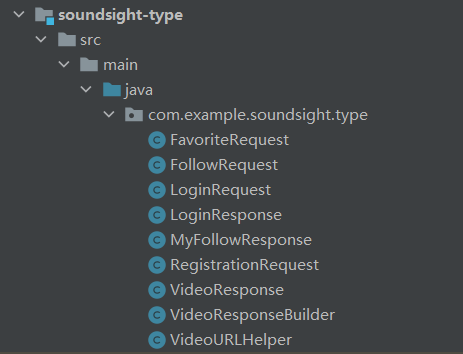
· 基础设施层：基础设施层依赖邻域层，实现邻域层中的有关接口。



· 触发器层：http包负责对外提供接口，task层处理定时任务。



· 通用层：提供一些前后端交互的通用类。



4.3功能实现:

4.3.1视频播放：播放，暂停，进度条拖拽

前端使用Video.js,后端从数据库获取视频url

4.3.2内容分类：

前端传给后端分类，数据库中不同类型视频存在不同表中，后端从对应的表中获取。

4.3.3热门视频：

后端通过定时任务，定时扫描数据库，把点赞数+收藏数>= 3000的加入热门视频表。

4.3.4推荐视频：

用户注册时选择感兴趣的类型，后续根据这个从对应的表中轮询查询。

4.3.5上下键视频切换：

下一个：每次从后端获取两个视频，放完再获取。

上一个：前端播放视频，存在前端数组中，上一个时从数组中取，最多存10个。

4.3.6用户注册和登录：

就是简单的增删改查了。

4.3.7收藏视频：

前端发送收藏请求，后端存到数据库。点击我的收藏时，从数据库中查询。

4.3.8推荐功能：

后端根据用户注册时选择的喜好轮询对应类型的视频表。

4.3.9点赞功能：

后端增加mysql中点赞数，用redis保存是否点赞。

4.3.10分享功能：

点击分享按钮，自动复制url,分享给他人访问即可调到该页面。

4.3.11关注功能：

使用redis存放关注和被关注关系。

4.3.12搜索功能:

后端使用ElasticSearch实现搜索功能

4.3.13前端懒加载：

之前查询用户发布的视频速度太慢，改为使用滚动懒加载优化用户体验。

**5. 数据库设计**

5.1表结构设计

**1. 用户表（Users）**

描述: 存储用户基本信息的表。

字段:

id (INT, 主键, 自增): 用户的唯一标识。

username (VARCHAR(50), 必填): 用户名，唯一。

password(VARCHAR(50), 必填): 用户密码。

avatar\_code(INT, 必填): 用户头像序号

preferences(VARCHAR(100)): 用户喜好。

约束:

PK\_Users PRIMARY KEY (id)

UN\_Username UNIQUE (username)

索引:

IDX\_Users\_Username ON username (用于优化基于用户名的查询)

**2. 视频表（不同类型视频存在不同表中，这里以sports\_video为例）**

描述: 存储视频信息的表。

字段:

id (INT, 主键, 自增): 视频id。

desc(VARCHAR(255), 必填): 视频描述。

like(INT): 点赞数。

favorite(INT): 收藏数。

isHot(INT): 是否已被加入热门表。

type(VARCHAR(20), 必填)：视频类型，冗余字段，便于后端查询。

约束:

PK\_Users PRIMARY KEY (id)

**3. 用户发表视频表（user\_video）**

描述: 存储用户和他发布的视频。

字段:

id (INT, 主键, 自增): 唯一标识。

userId(INT): 用户id

video\_id(INT): 视频 id

video\_type(varchar(20)): 视频类型

约束:

PK\_Users PRIMARY KEY (id)

索引:

IDX\_Users\_UserId ON userId(用于优化基于用户id的查询)

**4. 热门视频表（hot\_video）**

描述: 存储热门视频。

字段:

id (INT, 主键, 自增): 唯一标识。

video\_id(INT): 视频 id

video\_type(varchar(20)): 视频类型

约束:

PK\_Users PRIMARY KEY (id)

**5. 用户收藏表（favorite）**

描述: 存储用户和他收藏的视频。

字段:

id (INT, 主键, 自增): 唯一标识。

user\_id(INT): 用户id

video\_id(INT): 视频 id

video\_type(varchar(20)): 视频类型

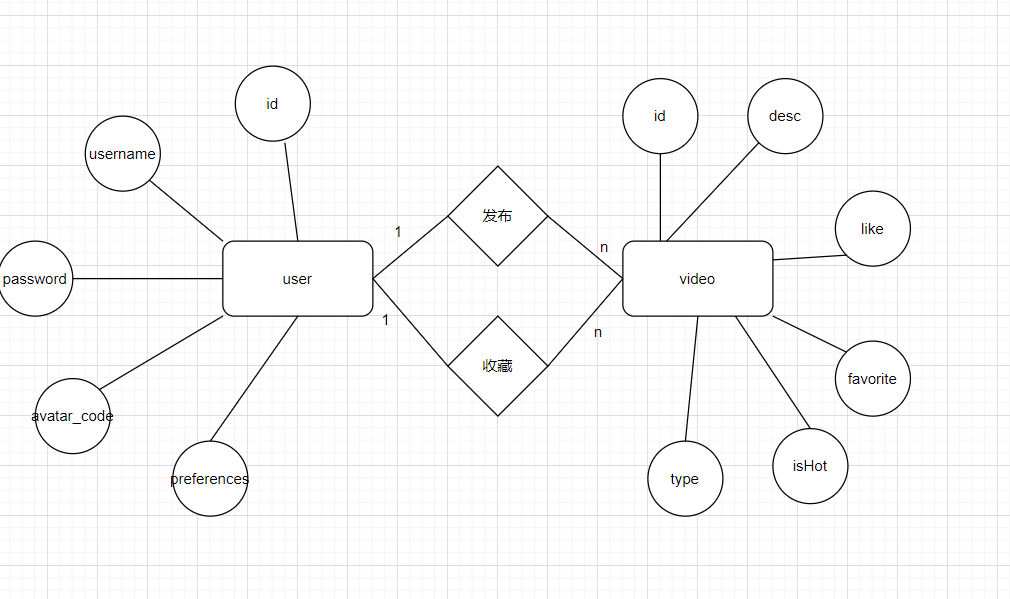
约束:

PK\_Users PRIMARY KEY (id)

索引:

IDX\_Users\_UserId ON user\_id(用于优化基于用户id的查询)

5.2 数据库ER图



5.3 数据库优化

索引优化(见上)

**6. 部署**

访问124.221.174.197