# 云计算 1607C 王顺虎

# JAVA工程师

1. **项目介绍(按照实训二做的项目写)**

|  |
| --- |
| **面试官您好，现在我给您介绍一下我在上家公司做的一个关于社交和招聘租房项目，在这个项目中呢，用户可以在平台进行交友，找工作，租房，等之类的，整个项目主要是采用SpringBoot+SpringCloud微服务框架实现的，通过eureka微服务注册中心实现多个微服务之间的调用，另外这么多的微服务不可能说是访问不同的微服务就使用不同的端口，这样显然是不现实的，所以我们就使用的Zuul网关，实现对请求的拦截和转发，然后数据库方面我们用的是MySQL关系型数据库和redis，MongoDB非关系型数据库，使用SpringDataJpa实现与数据库进行交互。这就是我对上家公司的项目的一个简单的介绍，您还有什么要问的吗？** |

1. **功能模块介绍(按照自己得实际项目写)**

|  |
| --- |
| 1. 用户微服务：用户这个模块呢？主要是登录和注册功能，注册的话有一个短信验证的功能，我们用的是阿里大于进行的发短信，用户注册的时候点击获取验证码，后台生成一个6位随机数，一式两份Redis中一份，Mq 中一份，发短信模块呢，进行监听，监听到之后会调用阿里大于发短信的接口，用户接受短信填写验证码，跟Redis中进行对比，验证通过就是在数据裤用户表中加一条数据，登录这块，首先就是先判断这个用户是不是存在，如果是数据库有着一条数据，那么就是通过JWT生成一个token令牌，将这个token返回给前台 ，如果用户访问哪个微服务的时候，而当前操作是需要用户的权限的，那么这个微服务就是需要配置拦截器，但是这个拦截器不是对前进行拦截，而是对请求中的token进行解析，判断角色字段，将用户和管理员分别放在作用域中，如果访问当前的方法需要用户或者是管理员权限，就通过获取作用域中的token，判断这个token是否存在，不存在就不让执行抛出异常，存在就放行，程序继续执行 2. 问答微服务：问答模块呢？主要是实现让用户实现在平台可以发布在面试中遇到的问题，请求帮助，对于好的回答可以进行点赞，这里用到了Redis防止重复点赞，用户也可以对已经有的问题进行答复，因为问答或者评论数据量是很大的并且是属于一层层的数据，所以我们没有用MySQL，而是用到了MongoDB这个文档型数据，后台我们在创建回答的时候会先判断他的上级Id是否存在，如果是存在的话上级的用户的回复数加1，MongoDB提供了一个inc方法这个方法是实现了当前字段的自增。 3. 搜索微服务：搜索这块呢我们用的是es，首先是将数据库中数据同步到es中，用的是logstash， 4. 交友微服务：用户点击交友这一按钮，会列出一些其他用户，对于自己喜欢的人，可以对其进行关注，对于不喜欢的人呢可以对其进行拉黑，后台操作的话就是判断是否喜欢，喜欢向好友喜欢表中添加后双方的Id，还要判断是否已经喜欢和是否互相喜欢，若互相喜欢该两条记录的状态为1表示相互喜欢，如果是不喜欢，还需要判断是否已经是好友，若已经是好友，还要判断是否是相互喜欢，是则删除单向喜欢数据，修改相反喜欢的状态为0，再向非好友中添加数据。如果不是好友，直接向非好友表中添加，对应好友粉丝数和自己的关注数要进行修改。 |

1. **开发中遇到的技术难点以及解决办法**

|  |
| --- |
| 难点1：XX项目XX模块需要对分布式事务进行解决  解决方法1：  难点2：使用hand操作**es**项目跨域问题需要解决  解决方法2：  。。。。。。。。。。。 |

1. **项目技术亮点分析**

|  |
| --- |
| (1)项目框架：项目采用得**spring boot+springcloud**微服务框架，好处是**spring boot**快速开发，快速整合，配置简化、内嵌服务容器。SpringCloud是一套非常完整的微服务解决方案，俗称“微服务全家桶”，几乎内置了微服务所使用的各种技术，可以不必集成第三方依赖。  (2)**redis技术**：项目中问答模块使用到redis了，具体是这样使用的用户对问题进行回答或者评论，其他用户就可以对评论进行点赞，但是又不能重复点赞，所有就用到了Redis，用户点赞后会在Redis中存一个当前用户的一个状态，再次点赞的时候会判断Redis中有没有当前用户的状态，有的话不让重复点赞。验证码也是放在Redis中，因为Redis可以设置过期时间。  (2)**MongoDB技术**：MongoDB 将数据存储为一个文档，数据结构由键值(key=>value)对组成。MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。，可以在MongoDB记录中设置任何属性的索引，如果负载的增加（需要更多的存储空间和更强的处理能力） ，它可以分布在计算机网络中的其他节点上这就是所谓的分片  (3)**JWT：**这JWT呢首先它是由三部分组成的头部、载荷、签名，头部就是jwt的基本信息，例如类型和签名所使用的算法，然后就是对头部信息进行加密，形成第二部分。载荷：就是存放有效信息的，比如说是当前用户的id和名字，还有角色，然后也是对信息进行加密，形成第二部分，然后将第一部分和第二部分进行拼接形成一个字符串，对这个字符串在进行加密这样就形成第三部分，将这三部分用.拼接就形成了token。  (4) **es:** es，相对于solr的好处就是可以试实时建立索引，而solr实时建立索引会发生堵塞，不足之处就是维护团队没有solr的维护团对那么完善 |

1. **项目答辩中亮点简历审核评定标准**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核内容** | **简历样式是否规范** | **项目内容是否完整** | **项目中是否有亮点** | **技术难点解决办法是否高效** | **存在问题** |
| **项目经理 （评定）** |  |  |  |  |  |
| **就业处干事（评定）** |  |  |  |  |  |
| 简历是否合格 | | | | 合格 | 不合格 |

1. **项目答辩成绩评定标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **考核内容** | **亮点简历与项目描述是否匹配** | **项目流程是否清晰** | **学生讲解是否流畅** | **项目介绍是否到位** | **技术亮点考核是否合格** | **存在问题** |
| **项目经理 （评定）** |  |  |  |  |  |  |
| **就业处干事（评定）** |  |  |  |  |  |  |
| 项目答辩是否合格 | | | | 合格 | 不合格 |  |