大家好，我是来自湖南岳阳的\*\*，接下来我要讲的是我前几天做的一个项目，微人事.人事管理是一种常见的企业型后台管理工具,它的主要目的是加强各个部门之间的协调和提高工作效率.项目功能较多,由于时间问题,我做的模块主要有登录,员工资料,导入导出,在线聊天,前后端项目打包。

微人事这个项目采用的是当下最流行的前后端分离的开发方式，前段使用的是Vue+ElementUI来构建，Vue的话是一套用于构建用户界面的渐进式框架，与其他大型框架不同，它是自底层向上逐层应用，只需关心视图层(View),最主要的是便于前后端项目整合,而element属于Vue的一个桌面端组件库.后端使用的Spring-boot开发,Spring-boot的话不同于Spring框架,它内部已经集成Tomcat,而且它最主要的不是xml配置,而是直接用java来配置Bean,省去了许多的配置文件,简化开发.并且在我们项目做完后方便打包成jar部署.(自动配置:springBoot开启注解使用的是启动类中的@SpringBootApplication注解,其实它主要由@Configuration,@ComponentScan,EnableAutoConfiguraton组成),从某种意义来说,前两个注解是由Spring框架自带的,SpringBoot最最核心的就是@EnableAutoConfiguation注解，它能根据类路径下的jar和动态配置来加载配置和注入Bean.)

现在来到项目,首先进入的是登录模块,登录模块我采用的是安全认证的方式,也就是SpringSecurity安全框架,实现这个功能我一共写了四个配置类:

首页我先实现了FilterInvocationSecuritymetadatasource接口的getAttribute方法来返回当前请求路径所需要的角色集合.(判断当前请求地址所需哪些角色才能访问)

然后实现了AccessdecisionManager接口的decide方法用于拿刚刚返回的角色集合与当前登录用户所具备的角色进行对比,如果用户具备所需角色就直接retrun该方法,如果不具备那么抛出权限不足的异常.

接下来实现AccessdeniedHandel的handle方法对刚刚抛出的异常进行处理,将权限不足等信息存入RespBean中返回给用户

最后在配置SecurityConfig类:SecurityConfig类继承自WebSecurityConfigurationAdapter类,该类提供一系列重载方法configure(),我们所需要实现参数分别为auth,HTTPSecurity,WebSecurity的configure重载方法()

auth:用于将HrService注入,以及使用.passwordEncoder方法对登陆进加密

(HrService必须实现UserDetailService)

Web:用于设置不拦截的请求路径antMacherts方法

http:用于注入返回角色集合和与角色进行对比的两个配置类,然后提供登陆成功和登陆失败的两个方法,将返回信息写入RespBean显示给用户,最后通过.ExceptionHandling()方法将抛出的异常类进行注入,也就是我刚刚所说的第三个抛出权限不足的配置类AccessDeniedHandler

动态加载菜单模块:

主要是根据不同的用户显示不同的菜单,实现是通过SpringSecurity框架的getcurrent方法获取登陆用户Id,然后根据用户Id查询菜单显示出来

员工资料模块:

员工信息查询附带条件查询以及分页,实现的话在控制层接受所需要的条件参数以及分页参数,然后根据条件查询数据库返回数据,然后让Vue框架自动渲染视图(条件查询需先在页面加载时查询所有下拉框的值,所需要表为:Nation,Political,PositionStatus,Joblevel,department,执行查询所有操作)

新增员工：

新增员工主要是给数据库执行一个新增操作,还有新增员工后,给该员工填写的email发一封入职邮件.发送邮件的话我们是通过创建子线程的方式执行的发送,因为发送邮箱需要好几秒,创建子线程的话就不需要等上一封邮件发完后才继续发送第二封,不管第一封发没发送完,第二次发送可以直接执行发送.。具体实现先写一个html模板,用于展示给员工查询,接下来实现Runnable接口创建子线程,该类中比先注入javaMailSend类用于执行发送和templateEngine类用于给html模板的员工信息赋值.接下来在run方法中编写代码,先用javamailSend的createMineHelper返回一个对象,在通过创建MimeMessageHelper类为邮件设置,发送者和发送人(New的时候用的有参构造,第一个参数为刚刚创建的对象,第二个设置为true),在创建Context类的set方法设置员工信息,然后通过第一步注入的templateEngine类提供process方法把Context作为参数放入,第一个参数为html模板,第二个为Context.然后用hepler的text方法吧process返回值放入,最后执行javamailSend的send方法,把createMineHelper方法的返回对象放入就行了

导出:

导出的话必先获取到所需要导出的数据,我们这边做的是一个导出员工数据,具体实现为:

先查询所有员工记录存入集合,然后New 一个HSSFWorkbook对象,意为创建一个文档,创建该对象需引入依赖poi.然后该类提供一系统get方法用于获取文档信息,如果要配置的话该文档信息的话可以通过刚刚get方法的返回对象用的set方法配置.

接下来通过HSSFWorkbook的createSheet方法创建一个Excel表单,可以用这个返回对象的set方法设置一些格式,样式还有列的宽高度,设置完这些基本信息后开始为Excel创建行和列,并为其赋值,具体实现用刚刚创建的对象的createRow,createCell方法,然后使用一个for循环,把查询出来的员工集合赋值入cell中,这样导出的文档信息和数据就基本配置完了,最后创建一个ByteArrayOutputStream对象作为HSSFWorkbook的写入方法参数,执行write方法.

导入：

导入先从页面拿到需要导入的数据,跟上传文件的意思一样,拿到一个file文件,接下来创建一个HSSFWorkBook对象,通过该类的getSheetAt方法指定参数获取到文件中那一行的数据,0是第一行,需舍弃第一行,因为第一行为表头,无法插入数据库,获取到该行对象后使用for循环,利用该行对象的getcell方法,获取该行每一个格子中的值,赋值进员工对象当中,需先创建员工集合,因为不但要循环遍历每一列,也要遍历每一行,也就是cell和row,当for循环遍历完后也就是数据全部从文档中拿出来并且赋值当员工集合当中了,最后在将数据插入数据库就行了,因为插入的数据是一个集合,所以我在mybatis的sql映射文件中用到了一个foreach标签,用来循环遍历集合进行插入.然后导入成功

在线聊天:

在线聊天包括登录用户之间进行的聊天以及系统管理员发送的系统消息,具体实现步骤为:

导入websocket依赖,配置websocketconfig实现websocketmessageBorkerConfiguration()的两个方法,第一个方法用户配置消息代理的前缀以及被@MessageMapping注解标识的方法进行处理,第二个方法用于配置用户在服务器进行消息传递时的前缀,最后在该类加商EnableWebSocketMessageBroket注解开启消息代理和@Configuration标识改类为一个注解类

配置完Config后编写Controller,主要编写的是 一个系统消息和用户之间进行的聊天请求,其实就是点对点消息与消息群发.

点对点:页面加载时查询所有用户,然后使用注解@MessageMapping在发送方法的上面,改方法用于获取所需要发送的用户,然后通过SpringSecurity提供的getCurrent方法获取当前用户登录的信息,也就是发送人.最后调用SimpMessagetemplate的ConverandSendtoUser方法进行发送

消息群发:

页面加载时获取所有系统消息,然后判断该用户是否具备admin权限也就是发送系统消息的权限,与点对点不同的是系统消息进行了数据库的插入,群发的消息保存于系统消息表与消息内容表当中,最后使用@MessageMapping和@SendTo注解配合使用实现消息群发

这些就是该项目的一些功能,以及这些功能的具体实现了,其中登录模块和在线聊天的实现较为复杂,文中只是片面讲解,具体实现可以查看源码.