WAR部署方案

WAR部署方案

正式环境部署

部署方案采用nginx+tomcat部署方案 后端服务发布部署到tomcat中 前端项目由于build后都是静态问题,部署到nginx中

- 一、后台项目jeecg-boot打war包(jeecg-boot-module-system)
- (1) 后台项目jeecg-boot-module-system打war包之前要进行如下改动
- 1、pom.xml文件中项目打包格式设置为war

war

具体配置如下:

```
<modelVersion>4.0.0</modelVersion>
  <groupId>org.jeecgframework.boot</groupId>
  <artifactId>jeecg-boot-module-system</artifactId>
  <version>2.0.0</version>
  <packaging>war</packaging></packaging>
```

2、pom.xml文件删除插件spring-boot-maven-plugin

下面配置删除

3、增加项目web容器部署的支持:

修改类/src/main/java/org/jeecg/JeecgApplication.java

代码如下:

```
package org.jeecg;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.builder.SpringApplicationBuilder;
import org.springframework.boot.web.servlet.support.SpringBootServletInitializer;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

@SpringBootApplication
@EnableSwagger2
public class JeecgApplication extends SpringBootServletInitializer {

    @Override
    protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {
        return application.sources(JeecgApplication.class);
    }

public static void main(String[] args) {
        System.setProperty("spring.devtools.restart.enabled", "true");
        SpringApplication.run(JeecgApplication.class, args);
```

}
}

- 4、修改配置文件(数据库和redis配置)
 - 1、修改数据库连接 application-prod.yml
 - 2、修改缓存redis配置 application-prod.yml
 - 3、修改上传附件配置 application-prod.yml

```
Quick Access
  datasource:
     869
                   masters
     87
                      url: jdbc:mysql://127.0.0.1:3306/jeecg-boot?characterEncoding=UTF-8&useUnicode=true&useSSL=false
     90
91
92
93
94
95
96
97
98<sup>©</sup>
99
                      driver-class-name: com.mysql.idbc.Driver
                      #多数振源配置
                      # タヌ北京 原始日
#multi-datasource1:
#url: jdbc:mysql://localhost:3306/jeecg-boot??useUnicode=true&characterEncoding=uff8&autc
                      #password: root
                        driver-class-name: com.mysql.jdbc.Drive
            #redis 配置
           redis:
    100 1019
1029
103
              host: 118 89 223 144
                                                                                                        ► redic 和智

        max-active:
        8
        #最大连接数据库连接数设0为没有限制

        max-idle:
        8
        #最大等待连接中的数量设0为没有限制

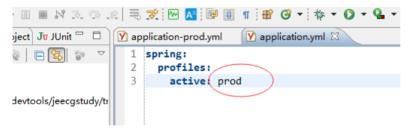
        max-wait:
        -1ms
        #最大建立连接等待时间。如果超过此时间将接到异常。设为-1表示无限制。

        min-idle:
        0
        #最大建立连接等待时间。如果超过此时间将接到异常。设为-1表示无限制。

        #最大建立连接等待时间。如果超过此时间将接到异常。设为-1表示无限制。
        #最大建立连接等待时间。如果超过此时间将接到异常。设为-1表示无限制。

    104
    105
106
107
                 shutdown-timeout: 100ms
   password: 'port: 6379
110 #mybatis plus 设置
mybatis-plus:
111 mapper-location
           mapper-locations: classpath*:org/jeecg/modules/**/xml/*Mapper.xml
    114 #jeecg专用配置
    #Jeecg 专用配盘
1150 jeecg :
1160 path :
#文件上传根目录 设置
           #X計上時報日本 原血
upload: /opt/jeecg-boot/upload
#Webapp文件路径
                                                                                                        上传附件路径配置
    118
    119
120
121
               webapp: /opt/jeecg-boot/webapp
```

• 4、切换配置为线上配置 application.yml



然后 maven package 打war包

- 二、后台项目jeecg-boot部署tomcat
- 1、设置tomcat端口号 8080,设置tomcat编码 URIEncoding="UTF-8"
- 2、部署项目到tomcat安装目录webapps/jeecg-boot工程目录下

部署完后通过http://localhost:8080/jeecg-boot 可以访问项目,提示token错误说明部署成功!!

- 三、前台项目build
- 1、修改 public/index.html

```
//图片预览请求地址
window._CONFIG['domianURL'] = 'http://localhost:8080/jeecg-boot';
window._CONFIG['imgDomainURL'] = 'http://localhost:8080/jeecg-boot/sys/common/view';
```

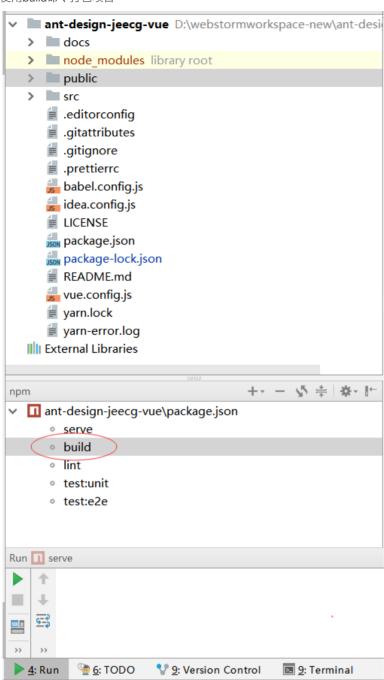
2、后台接口服务项目名默认是jeecg-boot,如果需要个性化可以修改src/utils/request.js 中baseURL参数

(一般情况下默认不需要修改)

具体代码如下:

```
// 创建 axios 实例
const service = axios.create({
  baseURL: '/jeecg-boot/', // api base_url
  timeout: 6000 // 请求超时时间
})
```

使用build命令打包项目



build完成后台会生成一个dist的目录该目录下即为build后的文件。

4、nginx部署前端项目

拷贝dist下的代码到nginx安装目录下html目录中,即可

四、nginx配置(conf/nginx.conf)

nginx监听80端口

```
server {
       listen
                   80;
       server_name 你的域名;
       #后台服务配置,配置了这个location便可以通过http://域名/jeecg-boot/xxxx 访问
       location ^~ /jeecg-boot {
                                 http://127.0.0.1:8080/jeecg-boot/;
          proxy_pass
                                Host 127.0.0.1;
          proxy_set_header
                                X-Real-IP $remote_addr;
          proxy_set_header
          proxy_set_header
                                X-Forwarded-For $proxy_add_x_forwarded_for;
       #解决Router(mode: 'history')模式下,刷新路由地址不能找到页面的问题
       location / {
           root html;
```

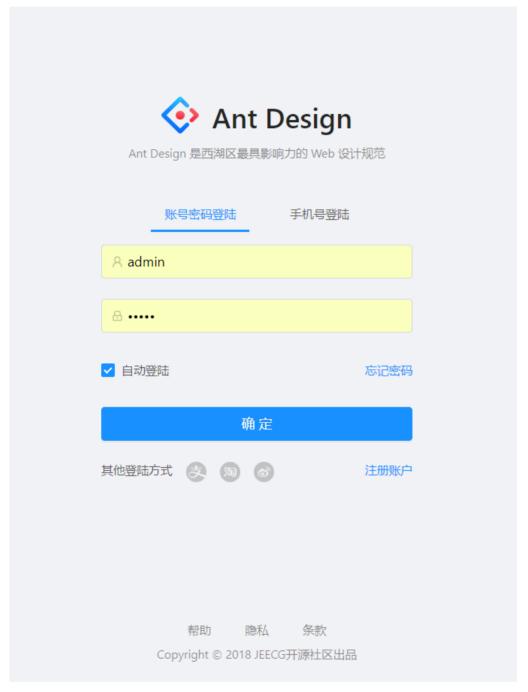
```
index index.html index.htm;
if (!-e $request_filename) {
    rewrite ^(.*)$ /index.html?s=$1 last;
    break;
}
}
```

五、nginx 开启压缩,提高首页访问效率

https://github.com/zhangdaiscott/jeecg-boot/issues/88

配置后启动tomcat, 启动nginx

通过http://你的域名/ 访问项目,出现如下页面,使用账户/密码: admin/123456 登录成功即可



上一篇:JAR部署方案 下一篇:极简部署方案