

产品源代码部署运行发布指南

开源企业 2023-06

1 准备工作

1.1 Java 微服务环境

按照下列软件版本要求，前往相关官网下载安装。

- 1) JDK \geq 1.8 （推荐 1.8 版本）
- 2) Mysql \geq 5.7.0 （推荐 5.7 版本）
- 3) Redis \geq 3.0
- 4) Maven \geq 3.0
- 5) Node \geq 12
- 6) Nacos == 2.1.2
- 7) Rocketmq == 4.9.4
- 8) Seata == 1.5.2

1.2 平台基座

平台基座是开源企业的应用开发平台和应用运行环境, 包括组织架构平台和开发者平台两部分；开源企业产品应用的运行，首先需要确保两平台的正常运行。前往新致开源企业官网，注册下载所需版本的部署文件压缩包、部署运行指南文档，按照指南进行部署和运行。

2 后端运行

2.1 代码导入

首先导入 common 依赖工程代码，然后导入后端工程代码。现以 Eclipse 导入后端工程代码为例，菜单 File -> Import，然后选择 Maven -> Existing Maven Projects，点击 Next> 按钮，选择“工作目录\back”，然后点击 Finish 按钮，即可成功导入；Eclipse 会自动加载 Maven 依赖包，初次加载会比较慢。

2.2 数据库导入

检查应用数据库表是否创建，开源企业提供可直接导入的初始数据库文件。

2.3 配置修改

2.3.1 服务器配置

编辑 resources 目录下 application-dev.yml 文件中的 server.port 配置项。

2.3.2 redis 配置

编辑 resources 目录下 application-dev.yml 文件中的 redis.host、redis.password、redis.port 等配置项，必须与平台基座配置一致。

2.3.3 nacos 配置

编辑 resources 目录下 bootstrap.yml 文件中 nacos 节点下所有配置项，必须与平台基座配置一致。

2.3.4 客户端 IP 配置

编辑 resources 目录下 application-dev.yml 文件中的 spring.cloud.nacos.discovery.ip 配置项，为服务注册到 nacos 中的 IP 地址。

2.3.5 动态数据源 system 库配置

编辑 resources 目录下 application-dev.yml 文件中的 spring.datasource.dynamic.0.url、spring.datasource.dynamic.0.username、spring.datasource.dynamic.0.password 配置项，为任意一租户数据库。

2.3.6 rocketmq 配置

编辑 resources 目录下 application-dev.yml 文件中的 spring.cloud.stream.rocketmq.binder 下所有配置项。

2.4 Maven 构建

在集成开发环境中执行 Maven 插件的 clean、package 等动作。

2.5 打开运行 com.geeke.MyApplication.java

3 前端运行

3.1 进入项目目录

```
cd client
```

3.2 安装依赖

```
npm install
```

3.3 开发环境 API 配置

按需调整 config\dev.env.js 文件 BASE_API 参数配置。

3.4 启动本地开发项目

```
npm run dev
```

3.5 浏览器运行

若能正确展示登录页面，使用默认账号/密码（system/123456）能成功登录，菜单及页面展示正常，则表明环境搭建成功。

4 产品发布

4.1 后端发布

4.1.1 工程打包

在 back 目录下，使用命令行执行打包：mvn clean package -Pdev，其中-Pdev 参数为环境可选项，会自动创建 target 目录，该目录中包含了生成的可执行 jar 包文件。

4.1.2 服务运行

采取 jar 部署方式，在 jar 包文件部署目录下，使用命令行执行：java -jar xxx.jar，命令行中可按需添加虚拟机内存参数。

4.2 前端发布

4.2.1 调整配置

按需调整 client\config\prod.env.js 文件 BASE_API 参数配置。

4.2.2 执行打包

在 client 目录下使用命令行执行打包：npm run build。

打包成功之后，会自动生成 dist 目录，里面就是打包好的文件，通常是 js、css、index.html 等静态文件。

dist 目录发布到 tomact 或者 nginx 即可，目录中的 index.html 是应用入口页面。

4.2.3 自动化发布

开发者可自行选择开源或商业 CI/CD 工具来自动构建和发布，相关配置操作按 CI/CD 工具指南进行。