If things had gone backstage management system RuoYi the Use Only ▲

If you build a manual based on the system development environment

1. Preparation

1.1 Pre-environment preparation - install Maven

If you build a project based on Maven management, you need to install the corresponding version first.

1.2 Development tools

If the system uses Eclipse as a development tool. But not limited to Eclipse . Only the operations required to build a development environment in Eclipse are introduced here .

2. Development environment construction

2.1 Development tool configuration

2.1.1 Configure Maven

Go to Window->Preferences->Maven->Installations page, set the installed Maven

Graphical user interface, application, table, Excel

Description automatically generated

2.1.2 Configure Maven repository path

Go to Window->Preferences->Maven->User Settings page and configure the warehouse path

Graphical user interface, text, application, email

Description automatically generated

2.1.4 Turn off verification

Enter Window-> Preferences-> Validation page, check the " Suspend All validators ", close check

Graphical user interface, text, application

Description automatically generated

2.2 Import project

Import the project through Eclipse , the steps are as follows:

(1) Click on the project area on the left— >Import...

Graphical user interface, text, application, email

Description automatically generated

(2) Choose RuoYi

Graphical user interface, text, application, email

Description automatically generated

(3) Click Finish

(4) The code of RuoYi is exported to Eclipse and can be seen in the project view at this time.

Graphical user interface, application

Description automatically generated

3. Run Ruoyi system

3.1 Necessary configuration

3.1.1 Modify database connection

Edit the application-druid.yml file in the src/main/resources directory to modify the database address account information.

Execute sql/ ry\_20180423.sql , the date of the quartz.sql files varies with the version

3.1.2 Development environment configuration

Edit the application.yml file in the src/main/resources directory ,

The default port is 80

Graphical user interface, text, application

Description automatically generated

3.1.3 Code generation configuration

Edit the application.yml file in the src/main/resources directory ,

Text

Description automatically generated

The default is module , you can modify it according to the actual situation. The generated table must be annotated

Note: If you have special requirements for the template, you can modify it yourself. Edit the src/main/ resources/templates/vm directory

Graphical user interface

Description automatically generated with low confidence

3.1.4 Logging Configuration

Edit the logback.yml file in the src/main/resources directory

< property name ="log.path" value ="/home/ruoyi/logs" />

Change to the path you need

3.2 Startup and verification (background)

Start RuoYiApplication.java and the following figure appears, indicating that the startup is successful

A picture containing text

Description automatically generated

Open the browser and enter: http://localhost:8080/captchaImage

If the return information can be displayed correctly, the background setup is successful.

3.3 Start-up and verification (foreground)

#Enter the project directory

cd ruoyi-ui

npm install --registry=https://registry.npm.taobao.org

npm run dev

Open the browser and enter: http://localhost:80 (default account admin/admin123 )

If the login page can be displayed correctly, and the login can be successfully completed, and the menu and page display are normal, it indicates that the environment is successfully set up

Note: The node environment needs to be configured to execute the npm command

**(download redis server and run also configuration mysql server and client and then run application and then download node.js and change port in ruoyi-ui>vue.config.js to 8080, if you face @Handler dispatch failed; nested exception is java.lang.NoClassDefFoundError: javax/xml/bind/DatatypeConverter@ Error then you should add bellow code to ruoyi>ruoyi-framework>pom.xml**

**<dependency>**

**<groupId>javax.xml.bind</groupId>**

**<artifactId>jaxb-api</artifactId>**

**</dependency>**

**)**

**create DATABASE `ry-vue` CHARACTER SET utf8mb4 COLLATE utf8mb4\_unicode\_ci;**

**mysql -u root -p ry-vue --default-character-set utf8mb4 < quartz.sql**

**mysql -u root -p ry-vue --default-character-set utf8mb4 < ry\_20210908.sql**

**give database mysql user name and password> application> srouce>yaml file**

**inside ruoyi>ruoyi-ui>npm install**

**inside rouyi>ruoyi-ui>npm run dev**

4. Deploy Ruoyi system

4.1 war deployment method

4.1.1 Modify the pom.xml file. The jar revised to war

Text, letter

Description automatically generated

If it is a sub-module, you need to modify ruoyi-admin

Text, letter

Description automatically generated

4.1.2 Remove the tomcat module from the spring-boot-starter dependency

Text

Description automatically generated

<exclusions>

<exclusion>

<artifactId>spring-boot-starter-tomcat</artifactId>

<groupId>org.springframework.boot</groupId>

</exclusion>

</exclusions>

4.1.3 Deploy to tomcat 's webapps directory

The default is RuoYi.war

Graphical user interface, application

Description automatically generated

4.1.4 Start-up and verification

Run startup.bat and the following figure appears, which means the deployment is successful

Text

Description automatically generated

4.2 Jar deployment

Execute the command: java – jar RuoYi.jar

Script execution: ry.sh start start stop stop

4.2 Front-end deployment

# Packing formal environment

npm run build:prod

# Packaged pre-release environment

npm run build:stage

After the build is packaged successfully, the dist folder will be generated in the root directory , which contains the built and packaged files, usually static files such as \*\*\*.js , \*\*\*.css , and index.html . Publish to your nginx or static server, where index.html is the entry page of the background service.

Demo address: ruoyi.vip

Document address: doc.ruoyi.vip

NS1Page

< All the above information is owned by Ruoyi >

All Rights reserved, RuoYi 2021