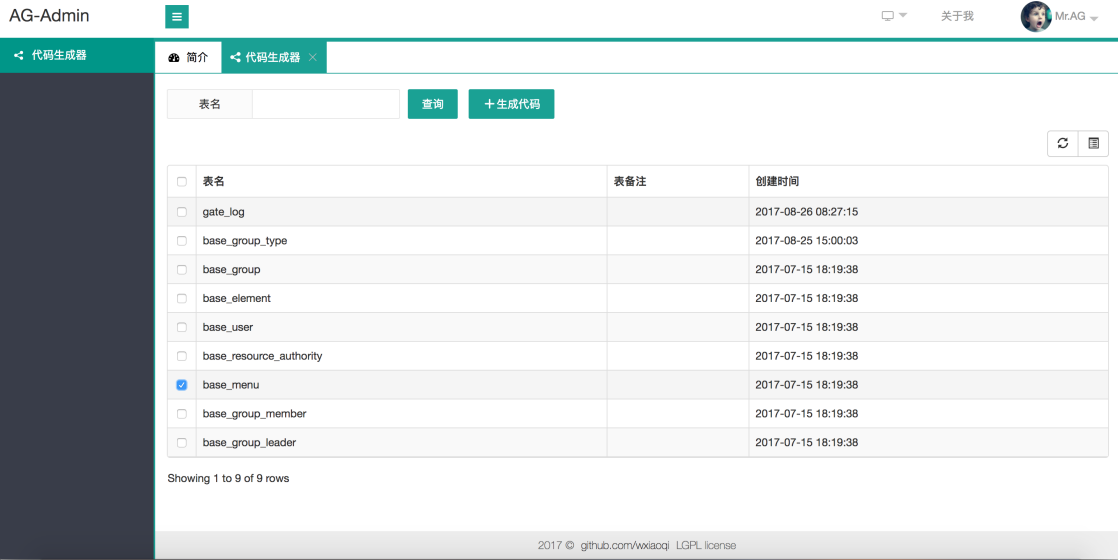
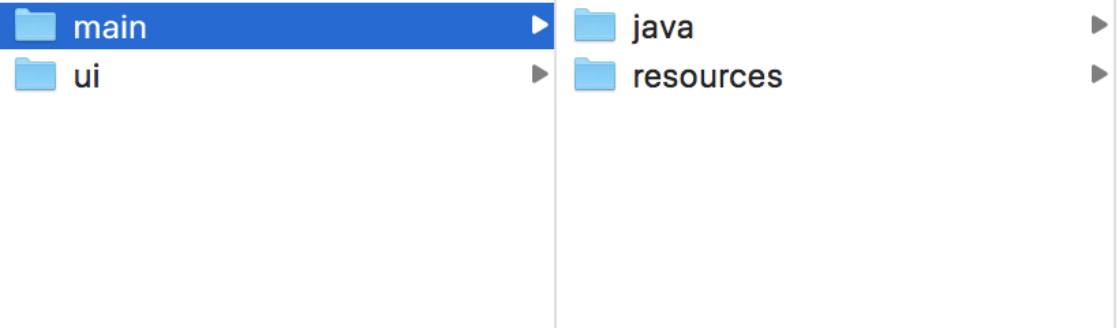


高效开发利器：代码生成器

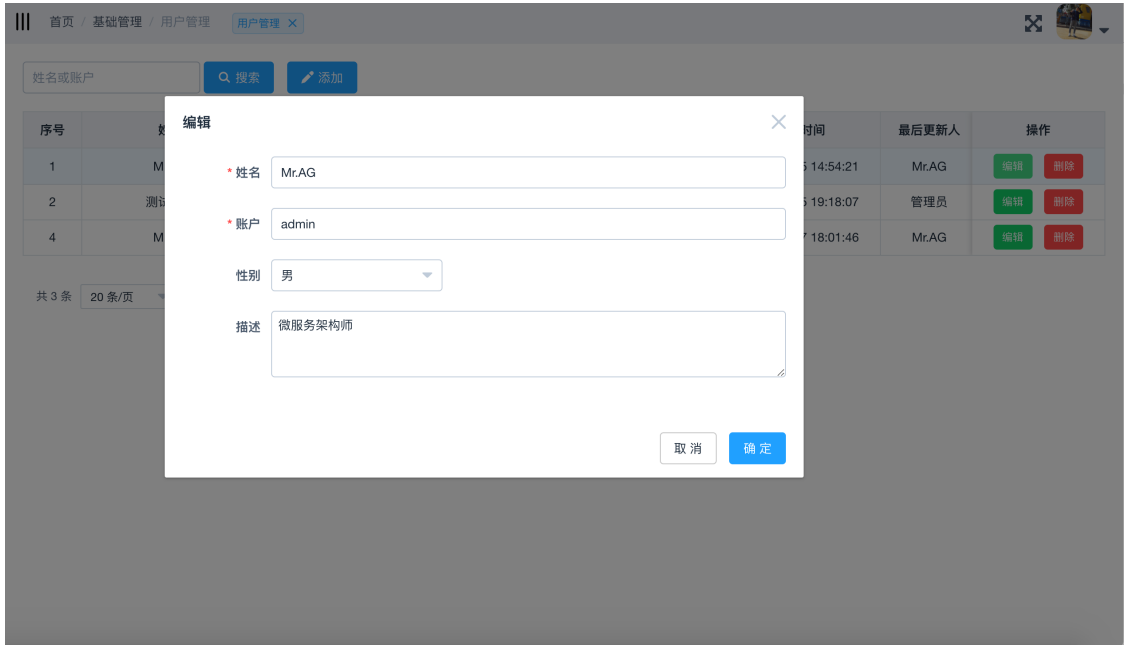
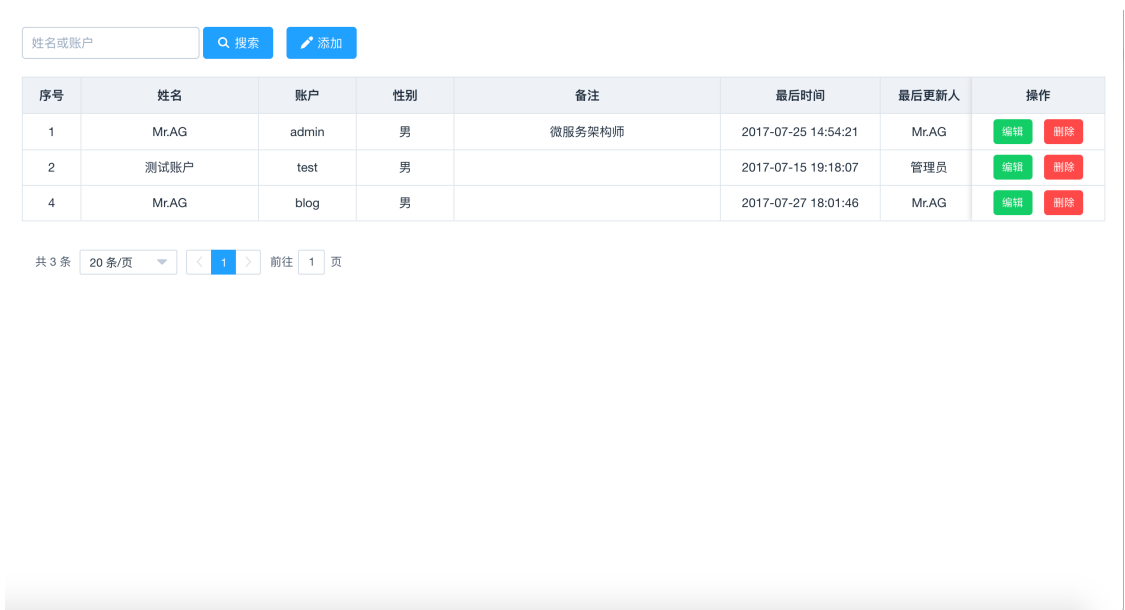


可以自动生成前端和后端代码，放入指定位置即可快速地完成增删改查。表要写注释

- 生成代码



- 最终运行效果



代码生成器指南

配置数据库和基础配置

- 修改application.yml的数据库信息

```
# Tomcat
server:
  tomcat:
```

```

        max-threads: 10
        min-spare-threads: 10
    port: 7777

# mysql
spring:
    datasource:
        type: com.alibaba.druid.pool.DruidDataSource
        driverClassName: com.mysql.jdbc.Driver
        url: jdbc:mysql://localhost:3306/ag_admin?
        useUnicode=true&characterEncoding=UTF-8
        username: root
        password: 123456
    jackson:
        time-zone: GMT+8
        date-format: yyyy-MM-dd HH:mm:ss
    thymeleaf:
        mode: LEGACYHTML5
        cache: false

# Mybatis配置
mybatis:
    mapperLocations: classpath:mapper/**/*.xml

```

- 修改生成代码的描述信息

```

#代码生成器, 配置信息

#包名
package=com.github.wxiaoqi.security
#作者
author=Mr.AG
#Email
email=463540703@qq.com
#表前错误的Unicode字符串!
tablePrefix=base_
#前端模块前缀
mainModule=admin

```

启动

启动: GeneratorBootstrap
访问地址: <http://localhost:7777/index>

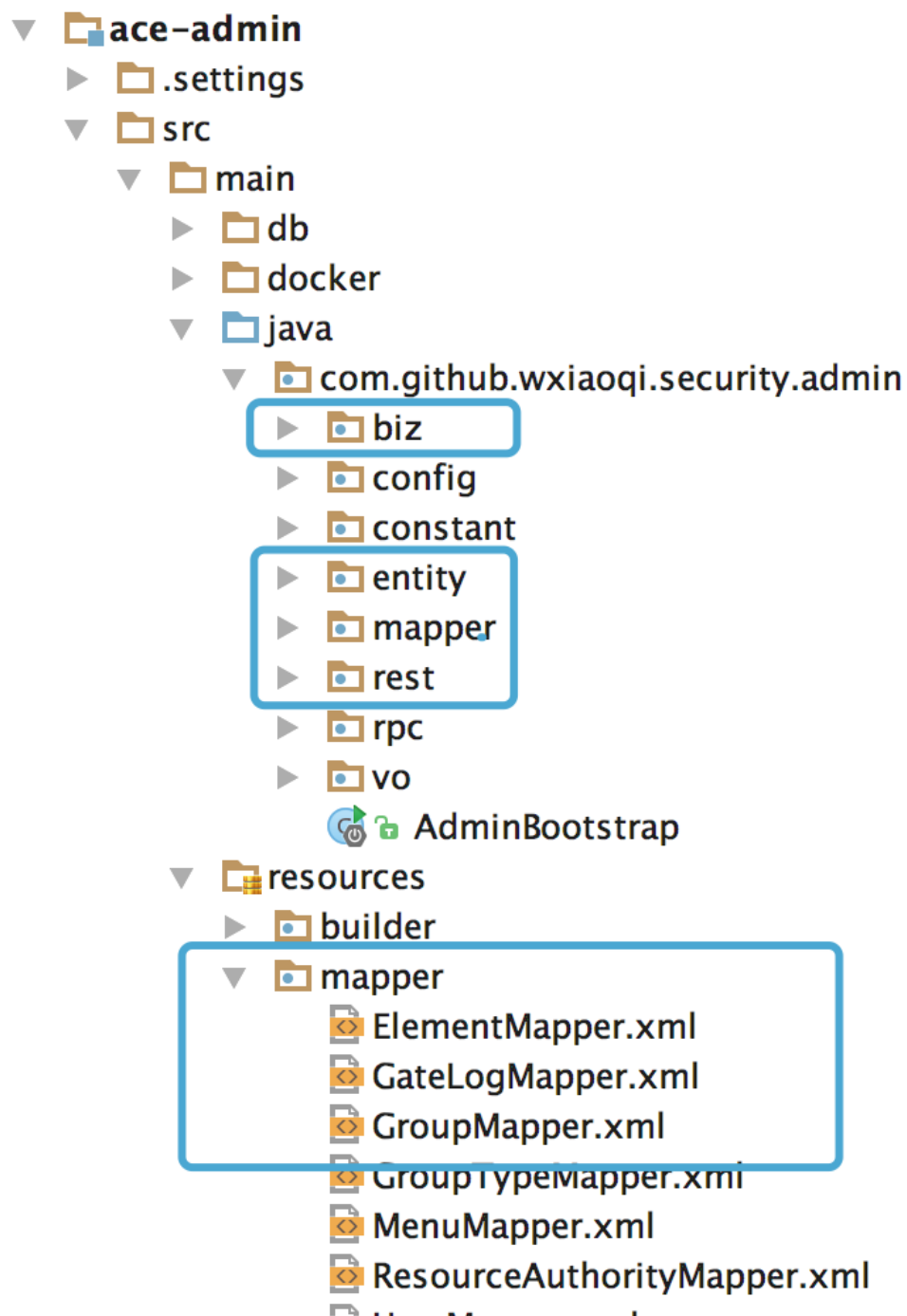
额外说明

记得在数据库中表描述和字段描述都要写上，否则生成的页面表单和列表没有名称

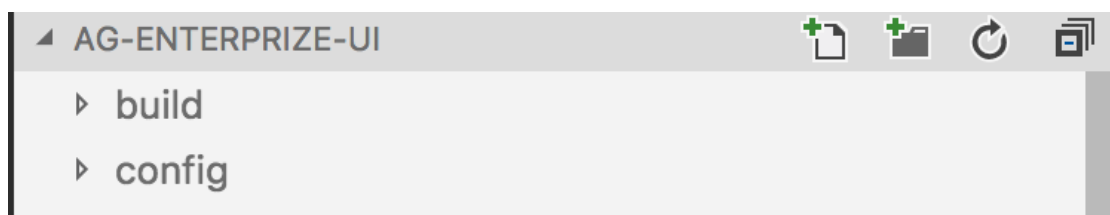
生成代码的使用应用

后端代码

直接将代码copy到后端对应服务包路径即可



前端代码



- dist
- node_modules

▾ src

▾ api

api, 即与后端服务对接的接口

- admin
- auth

JS article.js

JS login.js

JS qiniu.js

JS remoteSearch.js

~/prod/AG

- assets
- components
- directive
- filters
- mock

▾ router

记得同步在这里添加路由

JS _import_development.js

JS _import_production.js

JS index.js

- store
- styles
- utils
- vendor

▾ views

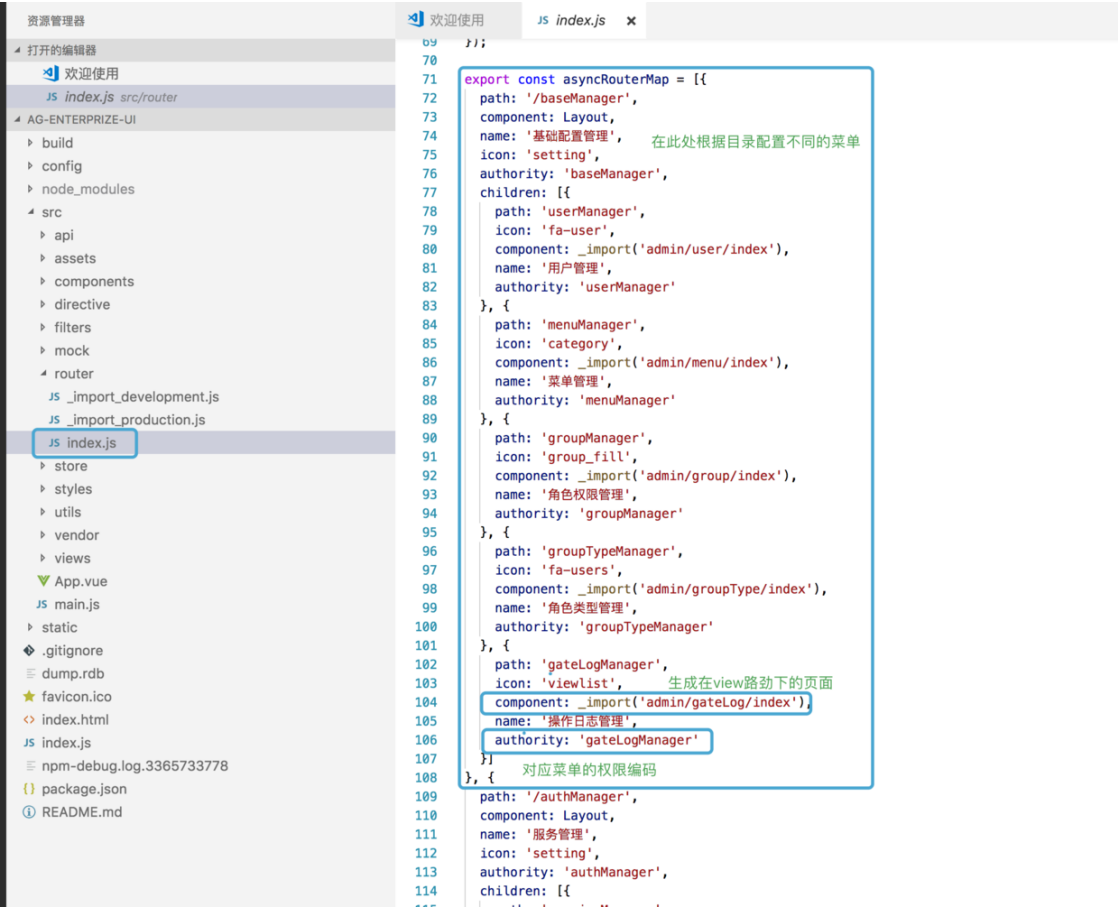
- admin
- auth
- charts

view, 即页面存放的地方

- components
- dashboard
- errlog
- error
- introduction

- 菜单配置
1. 前端添加路由配置

1. 前端添加路由配置



2. 管理端配置页面

+ 添加

编辑

删除

输入关键字进行过滤

权限管理系统

基础配置管理

用户管理

菜单管理

角色权限管理

角色类型管理

数据字典

操作日志

服务管理

服务权限管理

服务运维管理

服务监控管理

服务注册管理

服务链路监控

对应前端路由由权限编码

路径编码

userManager

标题

用户管理

父级节点

5

图标

fa-user

资源路径

/admin/user

类型

menu

排序

请输入排序

描述

请输入描述

前端组件

_import('admin/user/index')

更新

取消

图例从iconfont中读取，主要配置自己的iconfont，并替换前端src/assets/iconfont/iconfont.js

3. 给对应角色组进行菜单授权

- 按钮配置

因为采用的是前后端分离的方式，所以在生成前端页面的时候，会自动加入权限编码。

1. 前端页面

AG-ENTERPRISE-UI

components

directive

filters

mock

router

store

styles

utils

vendor

views

admin

gateLog

group

groupType

menu

user

index.vue

index.vue

auth

charts

components

dashboard

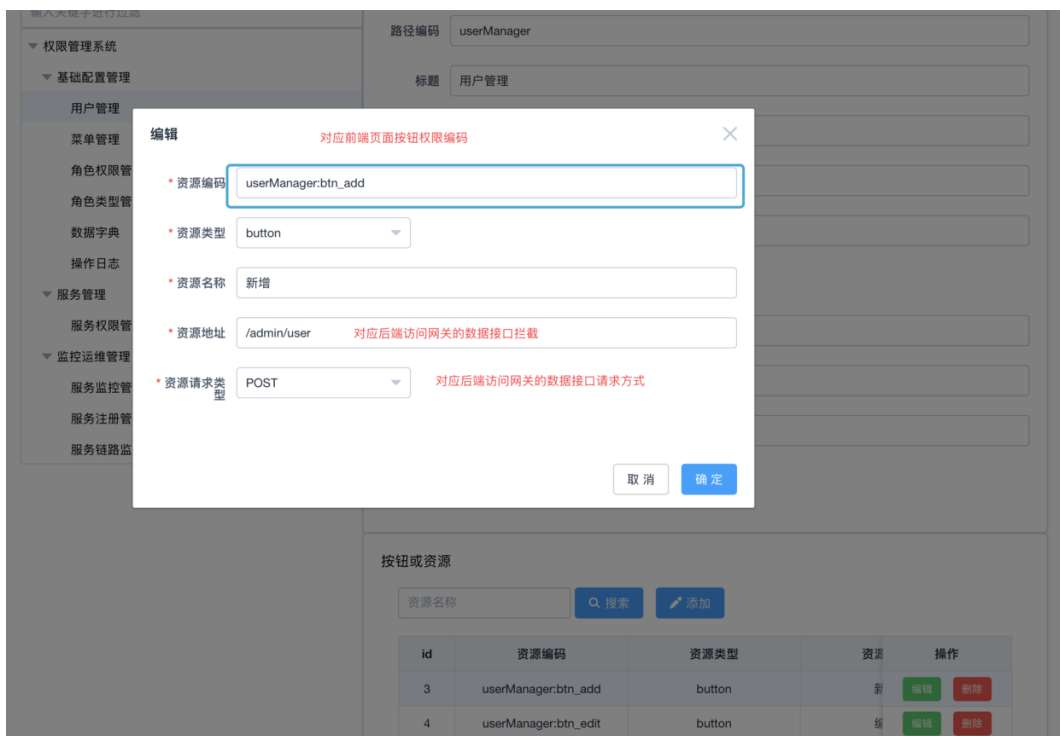
errlog

```
138 },
139 sexOptions: ['男', '女'],
140 dialogFormVisible: false,
141 dialogStatus: '',
142 userManager_btn_edit: false,
143 userManager_btn_del: false,
144 userManager_btn_add: false,
145 textMap: {
146   update: '编辑',
147   create: '创建'
148 },
149 tableKey: 0
150 },
151 },
152 created() {
153   this.getList();
154   this.userManager_btn_edit = this.$elements['userManager:btn_edit'];
155   this.userManager_btn_del = this.$elements['userManager:btn_del'];
156   this.userManager_btn_add = this.$elements['userManager:btn_add'];
157 },
158 computed: {
159   ...mapGetters([
160     'elements'
161   ])
162 },
163 methods: {
164   getList() {
165     this.$axios.get('admin/user/index')
166   }
167 }
168 }
```

从数据库中读取的权限配置，从而控制页面按钮的显示和隐藏

问题 输出 调试控制台 终端 1: zsh

2. 管理端配置



3. 给对应角色组进行菜单按钮的授权