

项目结构 &技术&搭建

项目结构 &技术&搭建

结构介绍：

1、后端项目

- 1.1、项目演示
- 1.2、使用技术
- 1.3、数据库表概览
- 1.4、数据库表前缀说明
- 1.5、搭建：

6、后台前端项目

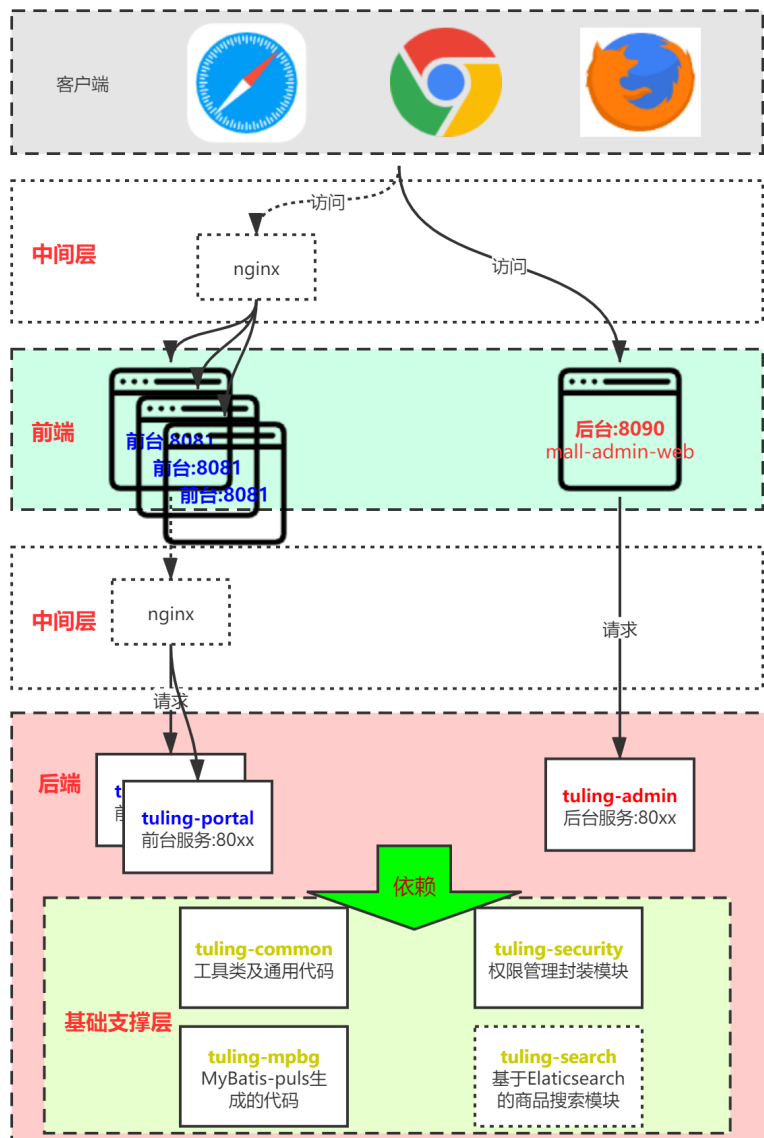
- 5.1、项目演示
- 5.2、使用技术

6、前台前端项目

- 5.1、项目演示
- 5.2、使用技术

结构介绍：

<https://www.processon.com/view/link/600a81ab5653bb1d8e0bdab6>



1、后端项目

1.1、项目演示

专门为前端提供rest接口

后台项目演示: <http://localhost:8081/>

1.2、使用技术

技术	说明	学习视频
Spring Boot	容器+MVC框架	https://www.tulingxueyuan.cn/video/XLSP819x0x819xXL100179x
Spring Security	认证和授权框架	https://www.tulingxueyuan.cn/series/XL100195xxxx1
MyBatis	ORM框架	https://www.tulingxueyuan.cn/video/XLSP582x0x582xXL100179x
MyBatis-Plus	MyBatis增强工具	本项目使用
MyBatis-Plus Generator	数据层代码生成	本项目使用
Swagger-UI	文档生产工具	https://www.tulingxueyuan.cn/video/XLSP856x0x856xXL100179x
Redis	分布式缓存	https://www.tulingxueyuan.cn/video/XLSP771x0x771xXL100184x
Druid	数据库连接池	https://www.tulingxueyuan.cn/video/XLSP873x0x873xXL100179x

OSS	对象存储	本项目使用
Lombok	简化对象封装工具	本项目使用

1.3、数据库表概览

mall项目目前有71张数据表，业务逻辑有一定复杂度，当然我们在开发中不一定会全用到，这里所有的表是我们整个图灵商城的完整版

cms_help	pms_comment_replay	sms_home_new_product
cms_help_category	pms_feight_template	sms_home_recommend_product
cms_member_report	pms_member_price	sms_home_recommend_subject
cms_preference_area	pms_product	ums_admin
cms_preference_area_product_relation	pms_product_attribute	ums_admin_login_log
cms_subject	pms_product_attribute_category	ums_admin_permission_relation
cms_subject_category	pms_product_attribute_value	ums_admin_role_relation
cms_subject_comment	pms_product_category	ums_growth_change_history
cms_subject_product_relation	pms_product_category_attribute_relation	ums_integration_change_history
cms_topic	pms_product_full_reduction	ums_integration_consume_setting
cms_topic_category	pms_product_ladder	ums_member
cms_topic_comment	pms_product_operate_log	ums_member_level
oms_cart_item	pms_product_verify_record	ums_member_login_log
oms_company_address	pms_sku_stock	ums_member_member_tag_relation
oms_order	sms_coupon	ums_member_product_category_relation
oms_order_item	sms_coupon_history	ums_member_receive_address
oms_order_operate_history	sms_coupon_product_category_relation	ums_member_rule_setting
oms_order_return_apply	sms_coupon_product_relation	ums_member_statistics_info
oms_order_return_reason	sms_flash_promotion	ums_member_tag
oms_order_setting	sms_flash_promotion_log	ums_member_task
pms_album	sms_flash_promotion_product_relation	ums_permission
pms_album_pic	sms_flash_promotion_session	ums_role
pms_brand	sms_home_advertise	ums_role_permission_relation
pms_comment	sms_home_brand	

1.4、数据库表前缀说明

- cms_*: 内容管理模块相关表（暂不关心）
- oms_*: 订单管理模块相关表
- pms_*: 商品模块相关表
- sms_*: 营销模块相关表（暂不关心）
- ums_*: 会员模块相关表

1.5、搭建:

1. 后端开发环境、工具

- 环境: jdk1.8 maven3.6+ 、 mysql 、 redis
 - 工具: idea2019（mybatis插件、 **lombok插件**、translation插件）、 Postman、Navicat、RedisStudio（github） redisdesktop
- i. 实在不会提供2份文档给你:

- [mall在Linux环境下的部署（基于Docker容器）.note](#)
- [mall在windows环境下运行.note](#)

2. 提供后端脚手架

- 基于SpringBoot+MyBatis-Plus的快速开发脚手架，拥有完整的权限管理功能，可对接Vue前端，开箱即用。

i. 项目结构

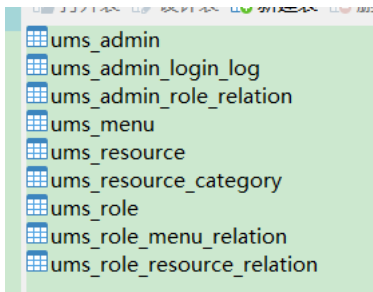
```

1 src
2 |─ common -- 用于存放通用代码
3 | |─ api -- 通用结果集封装类
4 | |─ config -- 通用配置类
5 | |─ domain -- 通用封装对象
6 | |─ exception -- 全局异常处理相关类
7 | |─ service -- 通用业务类
8 |─ config -- SpringBoot中的Java配置
9 |─ domain -- 共用封装对象
10 |─ generator -- MyBatis-Plus代码生成器
11 |─ modules -- 存放业务代码的基础包
12 | |─ ums -- 权限管理模块业务代码
13 | | |─ controller -- 该模块相关接口
14 | | |─ dto -- 该模块数据传输封装对象
15 | | |─ mapper -- 该模块相关Mapper接口
16 | | |─ model -- 该模块相关实体类
17 | | |─ service -- 该模块相关业务处理类
18

```

3. 导入提供的脚手架

4. 执行sql脚本：执行sql/tiny.sql



4. 修改本地的数据库 配置信息和 redis配置信息

5. 运行Main方法

6、后台前端项目

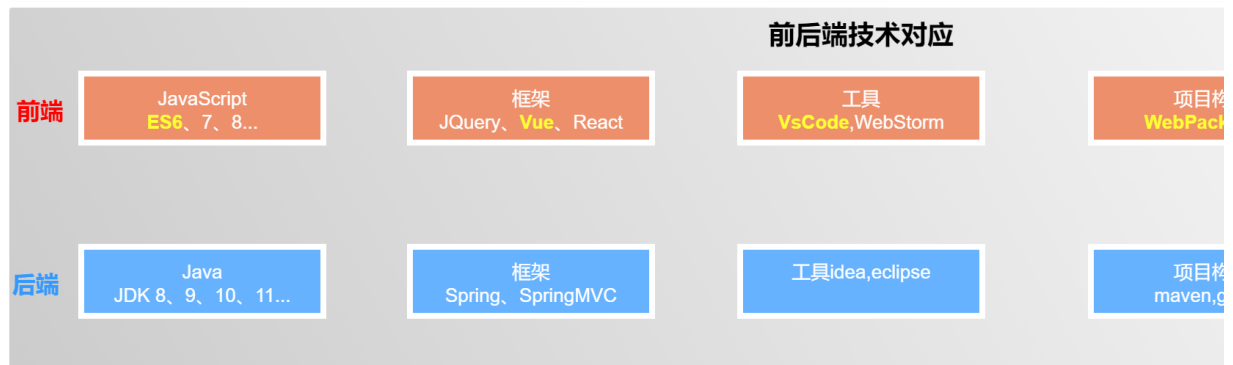
5.1、项目演示

后台项目演示：<http://localhost:8090/#/login>

核心功能：

- 商品模块
 - 商品管理
 - 商品分类管理
 - 商品类型管理
 - 品牌管理
- 订单模块
 - 订单管理

5.2、使用技术



技术选型

技术	说明	官网
Vue	前端框架	https://vuejs.org/
Vue-router	路由管理器	https://router.vuejs.org/
Vuex	全局状态管理框架	https://vuex.vuejs.org/
Element	前端UI框架	https://element.eleme.io/
Axios	前端HTTP框架	https://github.com/axios/axios
v-charts	基于Echarts的图表框架	https://v-charts.js.org/
Js-cookie	cookie管理工具	https://github.com/js-cookie/js-cookie
nprogress	进度条控件	https://github.com/rstacruz/nprogress
vue-element-admin	项目脚手架参考	https://github.com/PanJiaChen/vue-element-admin

项目布局

```

1 src -- 源码目录
2   └─ api -- axios网络请求定义
3   └─ assets -- 静态图片资源文件
4   └─ components -- 通用组件封装
5   └─ icons -- svg矢量图片文件
6   └─ router -- vue-router路由配置
7   └─ store -- vuex的状态管理
8   └─ styles -- 全局css样式
9   └─ utils -- 工具类
10  └─ views -- 前端页面
11    └─ home -- 首页
12    └─ layout -- 通用页面加载框架
13    └─ login -- 登录页
14    └─ oms -- 订单模块页面
15    └─ pms -- 商品模块页面
16    └─ sms -- 营销模块页面

```

搭建：

- 环境工具：
 - [VSCode 开发工具安装说明\(Vue\)](#)
 - 1. 安装vscode
 - 2. 安装vscode插件
 - 3. 解压前端开源项目：mall-admin-web-master-前端.zip
 - 4. 在VSCode中选择“文件” ---> “打开 文件夹” 选择解压后的项目所在目录

■ 5.安装nodejs (因为要用npm功能)

- 该项目为前后端分离项目，访问本地访问接口需搭建后台环境。(已完成<http://localhost:8099/>)
- 在VsCode终端窗口中运行命令：npm install,下载相关依赖;

```
PS D:\ideaProjects\mall-admin-web-master> npm install

> node-sass@4.13.0 install D:\ideaProjects\mall-admin-web-master\node_modules\node-sass
> node scripts/install.js

Cached binary found at C:\Users\zengjin\AppData\Roaming\npm-cache\node-sass\4.13.0\win32-x64-72_binding.node

> uglifyjs-webpack-plugin@0.4.6 postinstall D:\ideaProjects\mall-admin-web-master\node_modules\webpack\node_modules\uglifyjs-webpack-plugin
> node lib/post_install.js

> node-sass@4.13.0 postinstall D:\ideaProjects\mall-admin-web-master\node_modules\node-sass
> node scripts/build.js

Binary found at D:\ideaProjects\mall-admin-web-master\node_modules\node-sass\vendor\win32-x64-72_binding.node
Testing binary
Binary is fine
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.3 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.3: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

added 1279 packages from 733 contributors in 69.943s
```

- 运行项目之前需要将config/dev.env.js文件中的BASE_API改为<http://localhost:你的后端项目的端口/>

The screenshot shows the VS Code interface. On the left, the file explorer displays the project structure, with `dev.env.js` highlighted under the `config` directory. The main editor shows the content of `dev.env.js`, where the `BASE_API` is set to `"http://localhost:8099"`. Below the editor, the terminal window shows the output of the `npm install` command, including the installation of `node-sass` and `uglifyjs-webpack-plugin`.

- 在VsCode终端窗口中运行命令：npm run dev,运行项目;

```
PS D:\ideaProjects\mall-admin-web-master> npm run dev
```

- 访问地址：<http://localhost:8090> 即可打开后台管理系统页面;

The screenshot shows the terminal output after running `npm run dev`. It displays a green message: `DONE Compiled successfully in 20592ms`. Below this, an information icon is followed by the text: `Your application is running here: http://localhost:8090`.

6、前台前端项目

5.1、项目演示

<http://localhost:8081/>

5.2、使用技术

技术选型

技术	说明	官网
Vue	前端框架	https://vuejs.org/

Vue-router	路由框架	https://router.vuejs.org/
Vuex	全局状态管理框架	https://vuex.vuejs.org/
Element	前端UI框架	https://element.eleme.io/
Axios	前端HTTP框架	https://github.com/axios/axios
Js-cookie	cookie管理工具	https://github.com/js-cookie/js-cookie
nprogress	进度条控件	https://github.com/rstacruz/nprogress