“Java模块基础技术实训”课程报告

新闻信息管理系统的设计与实现

班级：软件20级4班

学号：120203504005

姓名：于有为

同组人：丁俊铭、王晨、黄振辉、马岩、白广印

成绩：

# 1系统开发的目的和意义

**1.1开发目的**

随着科技的不断发展，简单传统的记事本记账在复杂的超市货品销售中愈发不便，我们的系统操作简单，容易部署，为中小型超市提供了便捷的管理系统。

**1.2系统意义**

(1)、保证信息的准确性和时效性，随时通过计算机对系统进行管理，使得商品信息处于最新状态，保证统计数据和分析的准确。

(2)、减轻劳动强度、提高工作效率、增加商品管理的透明度。超市商品管理信息系统可以发挥计算机的强大功能，让管理人员从大量繁琐的手工劳动中解放出来，将单调枯燥的纸张涂写变成灵活的电子信息操作。并充分利用网络优势,加快超市商品信息和文档的发布、传送和获取，改善和优化备货工作，使各个部门之间的工作联系紧密、井然有序、清楚明晰，大大减轻工作负担，提高工作效率，增加商品管理的透明度。

(3)、降低管理成本，由于采用本系统的原因，使店长管理工作一改往日埋在各种表格中的被动局面，基本实现了无纸化办公，根除了很多日常管理中的浪费现象，体现出现代化商店管理的优势。

(4)、规范化管理，由于采用了计算机统计分析商品信息，一些原始信息在录入时就必须要求准确录入，这就要求管理人员在提供信息时要准确，同时要求在录入时也要十分认真，这样做是为了将错误降到最低，提高系统工作的准确率，从而实现商品管理工作规范化。

# 2主要的技术和开发工具简介

本系统开发中的后端技术为SpringBoot，前端为Vue，编译器为IntelliJ IDEA 2021.3.3。

开发主机配置为：

处理器: 英特尔酷睿i5-9600KF @ 3.70GHz六核

主板: 华擎B365M-ITX/ac(英特尔 B365芯片组)

内存: 16GB DDR4 2667MHz ( 8GB + 8GB)

显卡: NVIDIA GeForce GTX 1660 SUPER ( 6GB/英伟达)

硬盘: KINGSTON SA200OM81000G ( 1000GB)

# 3系统的主要功能

**3.1首页**

用户登录成功后进入首页，会显示当前账号的个人信息以及当前时间。

**3.2管理员管理**

只有超级管理员可使用管理员管理，用来管理系统管理员。

**3.3用户管理**

所有管理员均可使用，可以对用户信息进行管理。

**3.4仓库管理**

所有管理员均可使用，可以对仓库进行管理。

**3.5物品分类管理**

所有管理员均可使用，可以对物品的分类进行管理。

**3.6物品管理**

所有人均可使用，普通用户仅能查询，其他管理员可以对物品进行新增、入库出库等管理。

**3.7记录管理**

所有人均可使用，对物品操作记录进行查看。

# 4系统的架构和设计思路

本系统使用了MSCM架构。

（1） controller

控制层

相当于MVC的C层，controller通过service的接口来控制业务流程，也可通过接收前端传过来的参数进行业务操作。

（2） model

数据模型层

相当于MVC的M层，存放实体类，与数据库中的属性值基本保持一致。

（3） service

业务逻辑层

主要是针对具体的问题的操作，把一些数据层的操作进行组合，间接与数据库打交道（提供操作数据库的方法）。

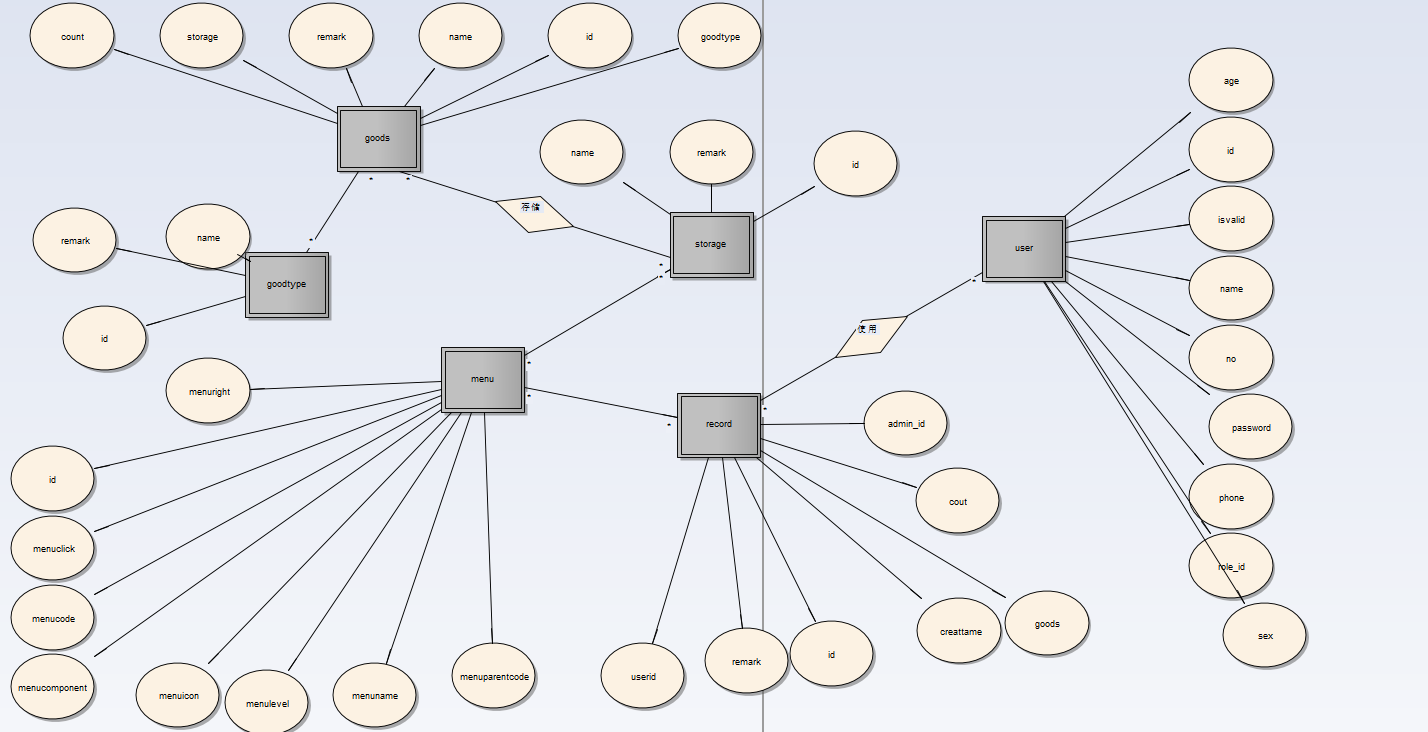
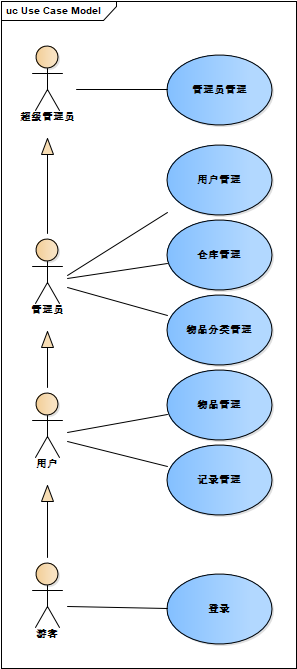
要做这一层的话，要先设计接口，在实现类。

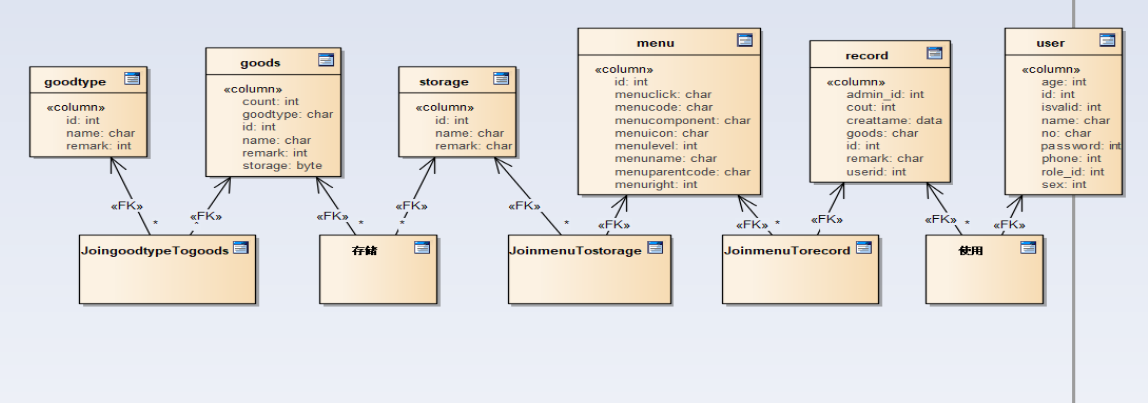
（4） mapper

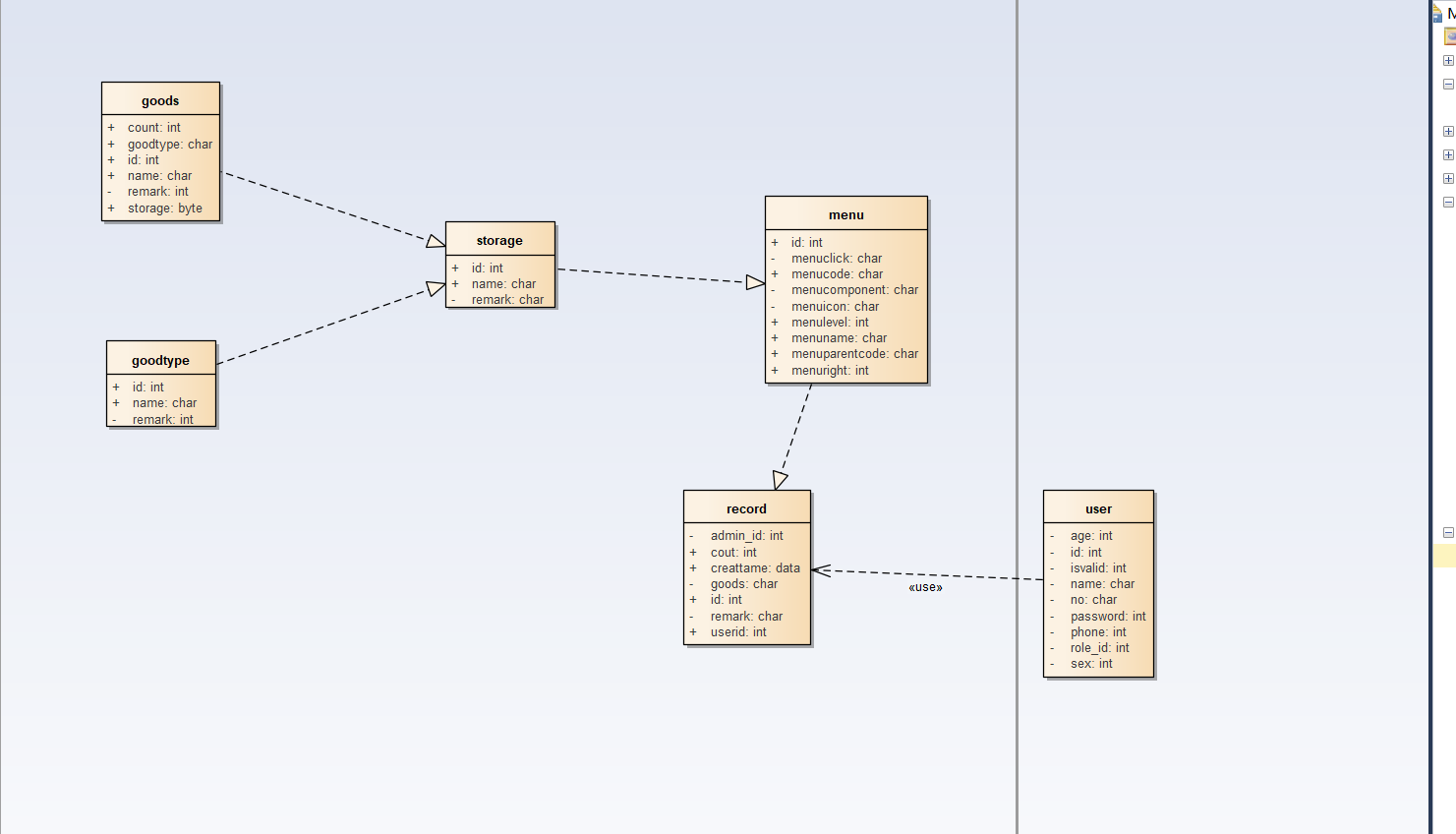
数据存储对象

相当于DAO层，mapper层直接与数据库打交道（执行SQL语句），接口提供给service层。

以下是管理系统的用例图。







# 5程序实现和重要模块的代码

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.IdType;

import com.baomidou.mybatisplus.annotation.TableId;

import com.baomidou.mybatisplus.annotation.TableField;

import java.io.Serializable;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

@Data

@EqualsAndHashCode(callSuper = false)

@ApiModel(value="User对象", description="")

public class User implements Serializable {

private static final long *serialVersionUID* = 1L;

@ApiModelProperty(value = "主键")

@TableId(value = "id", type = IdType.*AUTO*)

private Integer id;

@ApiModelProperty(value = "账号")

private String no;

@ApiModelProperty(value = "名字")

private String name;

@ApiModelProperty(value = "密码")

private String password;

private Integer age;

@ApiModelProperty(value = "性别")

private Integer sex;

@ApiModelProperty(value = "电话")

private String phone;

@ApiModelProperty(value = "角色 0超级管理员，1管理员，2普通账号")

private Integer roleId;

@ApiModelProperty(value = "是否有效，Y有效，其他无效")

@TableField("isValid")

private String isvalid;

}

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.IdType;

import com.baomidou.mybatisplus.annotation.TableId;

import java.io.Serializable;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

*/\*\**

*\* <p>*

*\**

*\* </p>*

*\**

*\**

*\*/*

@Data

@EqualsAndHashCode(callSuper = false)

@ApiModel(value="Storage对象", description="")

public class Storage implements Serializable {

private static final long *serialVersionUID* = 1L;

@ApiModelProperty(value = "主键")

@TableId(value = "id", type = IdType.*AUTO*)

private Integer id;

@ApiModelProperty(value = "仓库名")

private String name;

@ApiModelProperty(value = "备注")

private String remark;

}package com.wms.entity;

import lombok.Data;

@Data

public class RecordRes extends Record{

private String username;

private String adminname;

private String goodsname;

private String storagename;

private String goodstypename;

}

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.FieldFill;

import com.baomidou.mybatisplus.annotation.IdType;

import com.baomidou.mybatisplus.annotation.TableId;

import java.time.LocalDateTime;

import com.baomidou.mybatisplus.annotation.TableField;

import java.io.Serializable;

import com.fasterxml.jackson.annotation.JsonFormat;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

*/\*\**

*\* <p>*

*\**

*\* </p>*

*\**

@Data

@EqualsAndHashCode(callSuper = false)

@ApiModel(value="Record对象", description="")

public class Record implements Serializable {

private static final long *serialVersionUID* = 1L;

@ApiModelProperty(value = "主键")

@TableId(value = "id", type = IdType.*AUTO*)

private Integer id;

@ApiModelProperty(value = "货品id")

private Integer goods;

@ApiModelProperty(value = "取货人/补货人")

@TableField("userId")

private Integer userid;

@ApiModelProperty(value = "操作人id")

private Integer adminId;

@ApiModelProperty(value = "数量")

private Integer count;

@ApiModelProperty(value = "操作时间")

@JsonFormat(pattern = "yyyy-MM-dd HH:mm:ss", timezone="GMT+8")

@TableField(fill = FieldFill.INSERT)

private LocalDateTime createtime;

@ApiModelProperty(value = "备注")

private String remark;

@TableField(exist = false)

private String action;

}

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.TableField;

import java.io.Serializable;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

*/\*\**

*\* <p>*

*\**

*\* </p>*

*\**

*\**

*\*/*

@Data

@EqualsAndHashCode(callSuper = false)

@ApiModel(value="Menu对象", description="")

public class Menu implements Serializable {

private static final long *serialVersionUID* = 1L;

private Integer id;

@ApiModelProperty(value = "菜单编码")

@TableField("menuCode")

private String menucode;

@ApiModelProperty(value = "菜单名字")

@TableField("menuName")

private String menuname;

@ApiModelProperty(value = "菜单级别")

@TableField("menuLevel")

private String menulevel;

@ApiModelProperty(value = "菜单的父code")

@TableField("menuParentCode")

private String menuparentcode;

@ApiModelProperty(value = "点击触发的函数")

@TableField("menuClick")

private String menuclick;

@ApiModelProperty(value = "权限 0超级管理员，1表示管理员，2表示普通用户，可以用逗号组合使用")

@TableField("menuRight")

private String menuright;

@TableField("menuComponent")

private String menucomponent;

@TableField("menuIcon")

private String menuicon;

}

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.TableName;

import com.baomidou.mybatisplus.annotation.IdType;

import com.baomidou.mybatisplus.annotation.TableId;

import java.io.Serializable;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

*/\*\**

*\* <p>*

*\**

*\* </p>*

*\**

*\*/*

@Data

@EqualsAndHashCode(callSuper = false)

@TableName("goodsType")

@ApiModel(value="Goodstype对象", description="")

public class Goodstype implements Serializable {

private static final long serialVersionUID = 1L;

@ApiModelProperty(value = "主键")

@TableId(value = "id", type = IdType.AUTO)

private Integer id;

@ApiModelProperty(value = "分类名")

private String name;

@ApiModelProperty(value = "备注")

private String remark;

}

package com.wms.entity;

import com.baomidou.mybatisplus.annotation.IdType;

import com.baomidou.mybatisplus.annotation.TableId;

import com.baomidou.mybatisplus.annotation.TableField;

import java.io.Serializable;

import io.swagger.annotations.ApiModel;

import io.swagger.annotations.ApiModelProperty;

import lombok.Data;

import lombok.EqualsAndHashCode;

*/\*\**

*\* <p>*

*\**

*\* </p>*

*\**

*\**

*\*/*

@Data

@EqualsAndHashCode(callSuper = false)

@ApiModel(value="Goods对象", description="")

public class Goods implements Serializable {

private static final long serialVersionUID = 1L;

@ApiModelProperty(value = "主键")

@TableId(value = "id", type = IdType.AUTO)

private Integer id;

@ApiModelProperty(value = "货名")

private String name;

@ApiModelProperty(value = "仓库")

private Integer storage;

@ApiModelProperty(value = "分类")

@TableField("goodsType")

private Integer goodstype;

@ApiModelProperty(value = "数量")

private Integer count;

@ApiModelProperty(value = "备注")

private String remark;

}

package com.wms.controller;

import com.baomidou.mybatisplus.core.conditions.query.LambdaQueryWrapper;

import com.baomidou.mybatisplus.core.metadata.IPage;

import com.baomidou.mybatisplus.core.toolkit.StringUtils;

import com.baomidou.mybatisplus.extension.plugins.pagination.Page;

import com.wms.common.QueryPageParam;

import com.wms.common.Result;

import com.wms.entity.Menu;

import com.wms.entity.User;

import com.wms.service.MenuService;

import com.wms.service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.HashMap;

import java.util.List;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-07-21*

*\*/*

@RestController

@RequestMapping("/user")

public class UserController {

@Autowired

private UserService userService;

@Autowired

private MenuService menuService;

@GetMapping("/list")

public List<User> list(){

return userService.list();

}

@GetMapping("/findByNo")

public Result findByNo(@RequestParam String no){

List list = userService.lambdaQuery().eq(User::getNo,no).list();

return list.size()>0?Result.suc(list):Result.fail();

}

//新增

@PostMapping("/save")

public Result save(@RequestBody User user){

return userService.save(user)?Result.suc():Result.fail();

}

//更新

@PostMapping("/update")

public Result update(@RequestBody User user){

return userService.updateById(user)?Result.suc():Result.fail();

}

//删除

@GetMapping("/del")

public Result del(@RequestParam String id){

return userService.removeById(id)?Result.suc():Result.fail();

}

//登录

@PostMapping("/login")

public Result login(@RequestBody User user){

List list = userService.lambdaQuery()

.eq(User::getNo,user.getNo())

.eq(User::getPassword,user.getPassword()).list();

if(list.size()>0){

User user1 = (User)list.get(0);

List menuList = menuService.lambdaQuery().like(Menu::getMenuright,user1.getRoleId()).list();

HashMap res = new HashMap();

res.put("user",user1);

res.put("menu",menuList);

return Result.suc(res);

}

return Result.fail();

}

//修改

@PostMapping("/mod")

public boolean mod(@RequestBody User user){

return userService.updateById(user);

}

//新增或修改

@PostMapping("/saveOrMod")

public boolean saveOrMod(@RequestBody User user){

return userService.saveOrUpdate(user);

}

//删除

@GetMapping("/delete")

public boolean delete(Integer id){

return userService.removeById(id);

}

//查询（模糊、匹配）

@PostMapping("/listP")

public Result listP(@RequestBody User user){

LambdaQueryWrapper<User> lambdaQueryWrapper = new LambdaQueryWrapper();

if(StringUtils.isNotBlank(user.getName())){

lambdaQueryWrapper.like(User::getName,user.getName());

}

return Result.suc(userService.list(lambdaQueryWrapper));

}

@PostMapping("/listPage")

// public List<User> listPage(@RequestBody HashMap map){

public List<User> listPage(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

System.out.println("name==="+(String)param.get("name"));

/\*System.out.println("no==="+(String)param.get("no"));\*/

/\*LambdaQueryWrapper<User> lambdaQueryWrapper = new LambdaQueryWrapper();

lambdaQueryWrapper.eq(User::getName,user.getName());

return userService.list(lambdaQueryWrapper);\*/

Page<User> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<User> lambdaQueryWrapper = new LambdaQueryWrapper();

lambdaQueryWrapper.like(User::getName,name);

IPage result = userService.page(page,lambdaQueryWrapper);

System.out.println("total=="+result.getTotal());

return result.getRecords();

}

@PostMapping("/listPageC")

public List<User> listPageC(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

System.out.println("name==="+(String)param.get("name"));

Page<User> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<User> lambdaQueryWrapper = new LambdaQueryWrapper();

lambdaQueryWrapper.like(User::getName,name);

//IPage result = userService.pageC(page);

IPage result = userService.pageCC(page,lambdaQueryWrapper);

System.out.println("total=="+result.getTotal());

return result.getRecords();

}

@PostMapping("/listPageC1")

public Result listPageC1(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

String sex = (String)param.get("sex");

String roleId = (String)param.get("roleId");

Page<User> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<User> lambdaQueryWrapper = new LambdaQueryWrapper();

if(StringUtils.isNotBlank(name) && !"null".equals(name)){

lambdaQueryWrapper.like(User::getName,name);

}

if(StringUtils.isNotBlank(sex)){

lambdaQueryWrapper.eq(User::getSex,sex);

}

if(StringUtils.isNotBlank(roleId)){

lambdaQueryWrapper.eq(User::getRoleId,roleId);

}

//IPage result = userService.pageC(page);

IPage result = userService.pageCC(page,lambdaQueryWrapper);

System.out.println("total=="+result.getTotal());

return Result.suc(result.getRecords(),result.getTotal());

}

}

package com.wms.controller;

import com.baomidou.mybatisplus.core.conditions.query.LambdaQueryWrapper;

import com.baomidou.mybatisplus.core.metadata.IPage;

import com.baomidou.mybatisplus.core.toolkit.StringUtils;

import com.baomidou.mybatisplus.extension.plugins.pagination.Page;

import com.wms.common.QueryPageParam;

import com.wms.common.Result;

import com.wms.entity.Menu;

import com.wms.entity.Storage;

import com.wms.service.StorageService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.HashMap;

import java.util.List;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-10-15*

*\*/*

@RestController

@RequestMapping("/storage")

public class StorageController {

@Autowired

private StorageService storageService;

//新增

@PostMapping("/save")

public Result save(@RequestBody Storage storage){

return storageService.save(storage)?Result.suc():Result.fail();

}

//更新

@PostMapping("/update")

public Result update(@RequestBody Storage storage){

return storageService.updateById(storage)?Result.suc():Result.fail();

}

//删除

@GetMapping("/del")

public Result del(@RequestParam String id){

return storageService.removeById(id)?Result.suc():Result.fail();

}

@PostMapping("/listPage")

public Result listPage(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

Page<Storage> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<Storage> lambdaQueryWrapper = new LambdaQueryWrapper();

if(StringUtils.isNotBlank(name) && !"null".equals(name)){

lambdaQueryWrapper.like(Storage::getName,name);

}

IPage result = storageService.pageCC(page,lambdaQueryWrapper);

return Result.suc(result.getRecords(),result.getTotal());

}

@GetMapping("/list")

public Result list(){

List list = storageService.list();

return Result.suc(list);

}

}

package com.wms.controller;

import com.baomidou.mybatisplus.core.conditions.query.LambdaQueryWrapper;

import com.baomidou.mybatisplus.core.conditions.query.QueryWrapper;

import com.baomidou.mybatisplus.core.metadata.IPage;

import com.baomidou.mybatisplus.core.toolkit.StringUtils;

import com.baomidou.mybatisplus.extension.plugins.pagination.Page;

import com.wms.common.QueryPageParam;

import com.wms.common.Result;

import com.wms.entity.Goods;

import com.wms.entity.Record;

import com.wms.service.GoodsService;

import com.wms.service.RecordService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import java.util.HashMap;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-10-16*

*\*/*

@RestController

@RequestMapping("/record")

public class RecordController {

@Autowired

private RecordService recordService;

@Autowired

private GoodsService goodsService;

@PostMapping("/listPage")

public Result listPage(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

String goodstype = (String)param.get("goodstype");

String storage = (String)param.get("storage");

String roleId = (String)param.get("roleId");

String userId = (String)param.get("userId");

Page<Record> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

QueryWrapper<Record> queryWrapper = new QueryWrapper();

queryWrapper.apply(" a.goods=b.id and b.storage=c.id and b.goodsType=d.id ");

if("2".equals(roleId)){

// queryWrapper.eq(Record::getUserid,userId);

queryWrapper.apply(" a.userId= "+userId);

}

if(StringUtils.isNotBlank(name) && !"null".equals(name)){

queryWrapper.like("b.name",name);

}

if(StringUtils.isNotBlank(goodstype) && !"null".equals(goodstype)){

queryWrapper.eq("d.id",goodstype);

}

if(StringUtils.isNotBlank(storage) && !"null".equals(storage)){

queryWrapper.eq("c.id",storage);

}

IPage result = recordService.pageCC(page,queryWrapper);

return Result.suc(result.getRecords(),result.getTotal());

}

//新增

@PostMapping("/save")

public Result save(@RequestBody Record record){

Goods goods = goodsService.getById(record.getGoods());

int n = record.getCount();

//出库

if("2".equals(record.getAction())){

n = -n;

record.setCount(n);

}

int num = goods.getCount()+n;

goods.setCount(num);

goodsService.updateById(goods);

return recordService.save(record)?Result.suc():Result.fail();

}

}

package com.wms.controller;

import com.wms.common.Result;

import com.wms.entity.Menu;

import com.wms.entity.User;

import com.wms.service.MenuService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.RestController;

import java.util.List;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-10-04*

*\*/*

@RestController

@RequestMapping("/menu")

public class MenuController {

@Autowired

private MenuService menuService;

@GetMapping("/list")

public Result list(@RequestParam String roleId){

List list = menuService.lambdaQuery().like(Menu::getMenuright,roleId).list();

return Result.suc(list);

}

}

package com.wms.controller;

import com.baomidou.mybatisplus.core.conditions.query.LambdaQueryWrapper;

import com.baomidou.mybatisplus.core.metadata.IPage;

import com.baomidou.mybatisplus.core.toolkit.StringUtils;

import com.baomidou.mybatisplus.extension.plugins.pagination.Page;

import com.wms.common.QueryPageParam;

import com.wms.common.Result;

import com.wms.entity.Goodstype;

import com.wms.entity.Goodstype;

import com.wms.service.GoodstypeService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.HashMap;

import java.util.List;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-10-15*

*\*/*

@RestController

@RequestMapping("/goodstype")

public class GoodstypeController {

@Autowired

private GoodstypeService goodstypeService;

//新增

@PostMapping("/save")

public Result save(@RequestBody Goodstype goodstype){

return goodstypeService.save(goodstype)?Result.suc():Result.fail();

}

//更新

@PostMapping("/update")

public Result update(@RequestBody Goodstype goodstype){

return goodstypeService.updateById(goodstype)?Result.suc():Result.fail();

}

//删除

@GetMapping("/del")

public Result del(@RequestParam String id){

return goodstypeService.removeById(id)?Result.suc():Result.fail();

}

@PostMapping("/listPage")

public Result listPage(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

Page<Goodstype> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<Goodstype> lambdaQueryWrapper = new LambdaQueryWrapper();

if(StringUtils.isNotBlank(name) && !"null".equals(name)){

lambdaQueryWrapper.like(Goodstype::getName,name);

}

IPage result = goodstypeService.pageCC(page,lambdaQueryWrapper);

return Result.suc(result.getRecords(),result.getTotal());

}

@GetMapping("/list")

public Result list(){

List list = goodstypeService.list();

return Result.suc(list);

}

}

package com.wms.controller;

import com.baomidou.mybatisplus.core.conditions.query.LambdaQueryWrapper;

import com.baomidou.mybatisplus.core.metadata.IPage;

import com.baomidou.mybatisplus.core.toolkit.StringUtils;

import com.baomidou.mybatisplus.extension.plugins.pagination.Page;

import com.wms.common.QueryPageParam;

import com.wms.common.Result;

import com.wms.entity.Goods;

import com.wms.entity.Goods;

import com.wms.service.GoodsService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.HashMap;

*/\*\**

*\* <p>*

*\* 前端控制器*

*\* </p>*

*\**

*\** ***@author*** *wms*

*\** ***@since*** *2022-10-15*

*\*/*

@RestController

@RequestMapping("/goods")

public class GoodsController {

@Autowired

private GoodsService goodsService;

//新增

@PostMapping("/save")

public Result save(@RequestBody Goods goods){

return goodsService.save(goods)?Result.*suc*():Result.*fail*();

}

//更新

@PostMapping("/update")

public Result update(@RequestBody Goods goods){

return goodsService.updateById(goods)?Result.*suc*():Result.*fail*();

}

//删除

@GetMapping("/del")

public Result del(@RequestParam String id){

return goodsService.removeById(id)?Result.*suc*():Result.*fail*();

}

@PostMapping("/listPage")

public Result listPage(@RequestBody QueryPageParam query){

HashMap param = query.getParam();

String name = (String)param.get("name");

String goodstype = (String)param.get("goodstype");

String storage = (String)param.get("storage");

Page<Goods> page = new Page();

page.setCurrent(query.getPageNum());

page.setSize(query.getPageSize());

LambdaQueryWrapper<Goods> lambdaQueryWrapper = new LambdaQueryWrapper();

if(StringUtils.*isNotBlank*(name) && !"null".equals(name)){

lambdaQueryWrapper.like(Goods::getName,name);

}

if(StringUtils.*isNotBlank*(goodstype) && !"null".equals(goodstype)){

lambdaQueryWrapper.eq(Goods::getGoodstype,goodstype);

}

if(StringUtils.*isNotBlank*(storage) && !"null".equals(storage)){

lambdaQueryWrapper.eq(Goods::getStorage,storage);

}

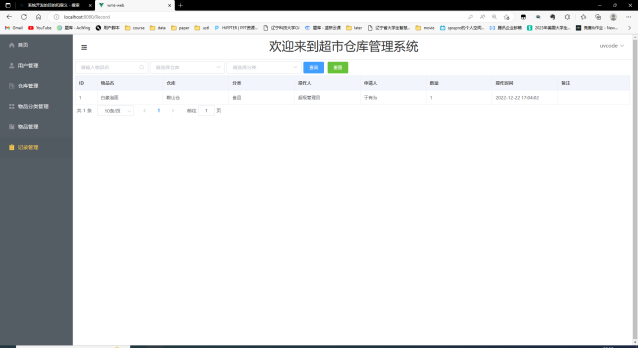
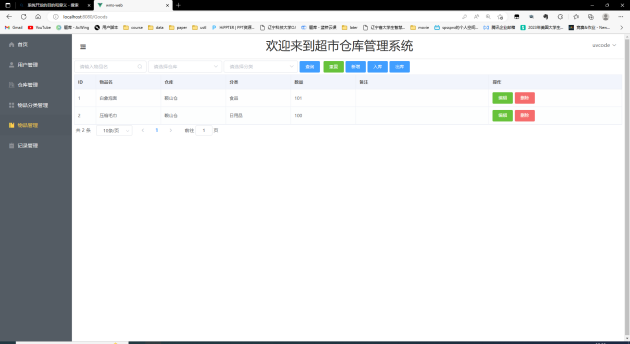
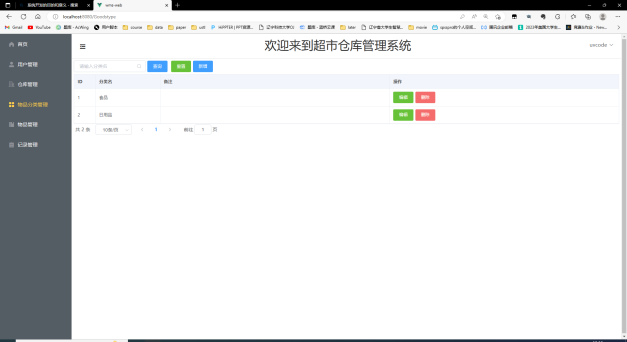
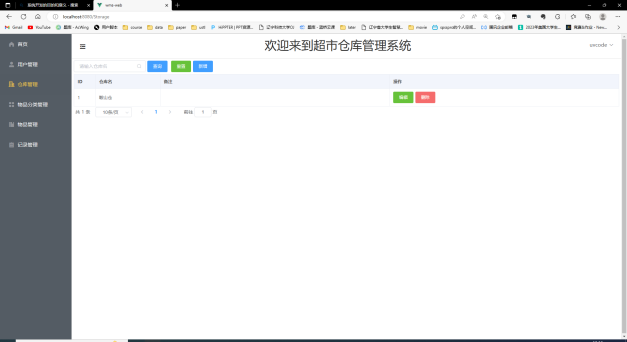
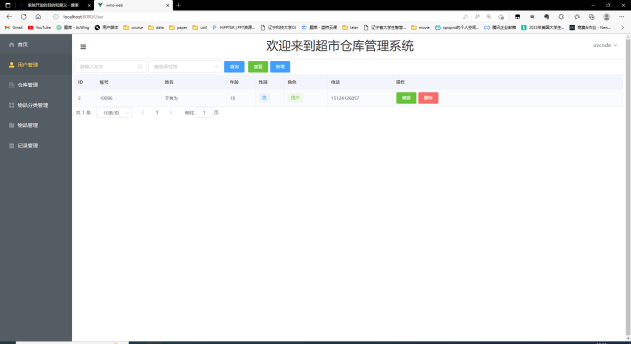
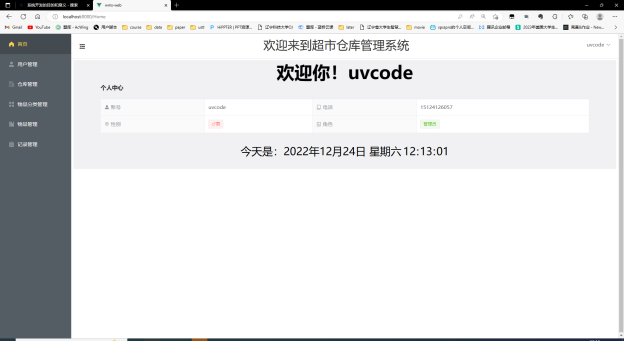
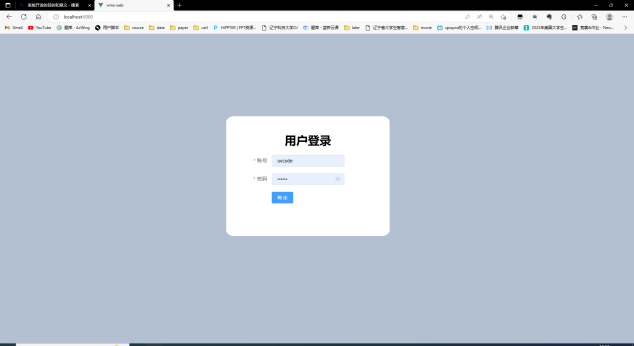
IPage result = goodsService.pageCC(page,lambdaQueryWrapper);

return Result.*suc*(result.getRecords(),result.getTotal());

}

}

运行截图



# 6总结

本次课程设计大多是跟着网上的流程做的，自己对于项目构建的具体过程还是不够熟练，希望可以通过以后的联系不断提升自己。