**Reggie外卖系统开发手册**

**1.环境配置 数据库表的建立**

1.新建Springboot项目

2. 获取/制作界面原型

3.前端资源的静态资源映射

4.在MySql中建立数据库表项

**2.创建实体类**

创建所有类别的包：

关系：application : controller🡪service🡪mapper🡪entity

自下向上配置，再向下开发

说明：

1.Controller：后端与前端最直接的通信位置，实现上层逻辑

2.Service: 实现模块化的逻辑，为上层Controller提供服务

3.Mapper：与数据库交互（DAO）

4.Entity：封装实体

**3.主要代码开发**

分析一般从页面发送请求开始

思路：

1.前端传参在url中可见，可变参数传递：@RequestParam

Json参数传递：request body

2.后端返回值经过response body被转化为json（需求在response中可见）

3.Header中可以看见请求路径

**3.1登录与退出**

3.1.1后台登陆（员工）

逻辑为：passwd MD5

Username查询，比对密码，查看员工状态，将id存入session并返回1（成功）

3.1.2 退出后台

清理session的用户id 返回结果

3.1.3阻止直接页面访问强行登录（过滤器/拦截器）

自定义过滤器 判断本次是否需要过滤 本次登录与否 跳转/放行

**3.2员工管理**

3.2.1新增员工

页面将数据传到服务器（json），controller接收后用service保存，再用Mapper持久化

需要保持id唯一，捕获错误

3.2.2.员工分页查询

Ajax->Controller🡪Service🡪Mapper—>controller->返回响应

3.2.3启用/禁用员工账号

只有admin才能操作（前端）（list.html）

流程与之前一致，修改数据库即可

Json{id，status}

前端js处理数字会有精度缺失，需要转化类型—消息转换器

3.2.4编辑员工信息

先请求服务端，回显，再进行修改

**3.3菜品分类管理**

开发简化：

公共字段填充@TableField

🡪统一处理字段，避免重复代码

3.3.1新增

与之前一样，先建立所有的包和配置

3.3.2分页

3.3.3删除

3.3.4修改分类，与之前一样，需要回显（此处前端已经写好）

**3.4 菜品开发**

3.4.1文件的上传与下载

3.4.2新增菜品 先获取数据库的选项，显示在前端页面（图片又进行一次传输）

选择后再传回后端

一共三次通信过程

DTO(Data Transfer Object)展示层与服务层之间传输数据

3.4.3分页查询

两次通信：数据（ajax）、图片

3.4.4修改菜品

修改的逻辑与之前一致

类别回显，菜品信息回显，图片

信息传回

**3.5套餐管理**

3.5.1新增

Ajax🡪1.套餐分类 2.菜品分类 3.菜品数据

4.图片（需要回显）5.传回数据保存

3.5.2分页 数据与图片

3.5.3删除

单个删除/批量删除（请求地址一样，传入id）

只能删除停售的套餐

**4.移动端开发**

4.1 手机验证码 //此功能忽略

由于此功能无法开发，使得过滤器无法识别，所以在使用后续相关功能之前，请先登录admin使得过滤器放行

4.2 地址展示与修改 与之前一样

4.3面向用户的菜品展示

逻辑：获取分类数据

Default显示：获取第一个类别下的菜品

购物车显示

调用的方法与之前是一致的，需要额外写的是购物车界面的显示

Tip: ctrl + F5清除浏览器缓存

4.4购物车

菜品：选择菜品规格，再加入购物车

套餐：直接加入

购物车中可以修改内容

逻辑：点击添加后ajax，添加至服务端

点击购物车，请求数据

点击清空，请求清空

4.5用户下单

逻辑：

按下去结算，前端自动跳转至支付页面

Ajax请求：用户地址、用户购物车（函数已写）

点击去支付，将订单保存（模拟支付功能）

**5.项目优化**

5.0开发使用git来进行版本控制

5.1.1缓存菜品数据（redis）

改造Dish Controller 的list方法，加入缓存

改造save和update方法，清除缓存

5.1.2缓存套餐数据（Spring Cache + redis）

改造SetmealController的list @Cacheable

Save和delete @ CacheEvict

// 未实现，已学习

5.2.1Mysql主从复制

完成双方的配置，使得建立连接，且确认双方的身份

5.2.2读写分离

**6.部署**

企业的部署需要多台服务器，此处忽略。

我们将Spring前后端代码均服务部署到windows，Mysql也部署到windows

Redis部署在linux上

Linux和windows电脑端可以直接访问

手机端需要先开放电脑服务端的端口（新建入站规则）

网址输入方式如下：ip : port + 路径