中山大学饭堂菜品评价系统-需求分析和设计规格文档

目录

一、背景

本文档为许嘉瑋（学号 21307275）于中山大学 2024 年春季软件工程期末大作业中为其设计的“中山大学饭堂菜品评价系统”编写的测试文档。

二、需求分析

作为大学生，食堂的菜品是我们一日三餐的选择之一。而由于每个校区存在多个食堂，每个食堂又存在许多窗口，因此对于许多人来说在众多菜品中做选择是一件困难的事情，既要考虑价格，又想避免踩坑。因此我们希望通过设计“中大食评”这样一个应用，帮助我们学校的学生和教职工，能够更好地了解菜品的信息，也为食堂提供了改进菜品和服务的方向和依据。因此我们认为该数据库应用应该具有如下的功能

用户角色

1. 学生和教职工：浏览菜品、搜索菜品、评价菜品、查看评价

功能需求

1. 用户管理：用户注册功与登录功能
2. 菜品展示：对各个菜品的详细信息进行显示，包括菜品所在校区、所在食堂、供应时段、一些评价信息。
3. 菜品搜索：用户应该能够通过菜品名字进行模糊搜索，或者根据校区、食堂、供应时段等进行筛选。
4. 菜品评价：用户应该能够添加自己对某个菜品的评价，包括评分信息和评价内容，方便其它用户根据菜品评分选择菜品。

三、总体设计

系统基于 MySQL + tomcat + eclipse 实现。前端使用了 jsp、css 和 JavaScript，后端使用了 Java。

四、详细设计

模块划分如下：

用户管理模块：实现用户注册、登录功能。

菜品管理模块：展示菜品信息，支持搜索和筛选。

评价管理模块：实现菜品评价的查看与添加功能。

五、架构设计

图 6-8 一个通用的分层体系结构

系统采用了分层体系结构，参考了如下模型：

具体地，在 jsp 文件夹下各 jsp 文件实现了用户界面，在 web.xml 下实现用户界面管理，在包 servlet 下实现了核心应用功能，最后在底层的 MySQL 中初始化数据库。

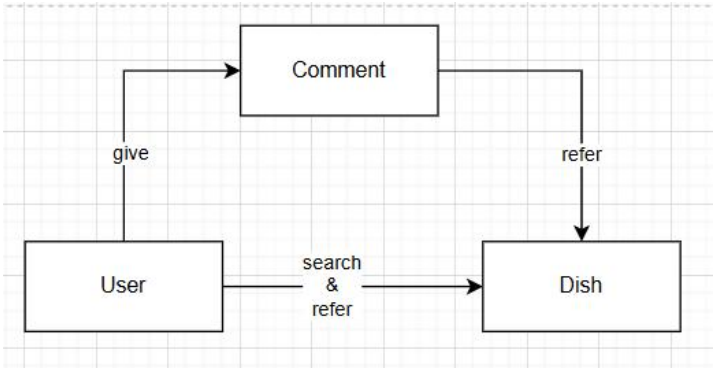
六、类的设计

类可以认为有如下设计：

用户类：管理用户的登录、注册状态信息

菜品类：管理菜品的各项信息

评价类：管理评价信息



（类图：系统中类和关联关系）

七、子系统及其接口设计

用户管理子系统：

接口：用户注册、登录接口，获取用户信息接口

输入：用户名、密码

输出：用户 ID、注册或登录结果

菜品管理子系统：

接口：菜品搜索接口、菜品详情接口

输入：搜索条件（菜品名、校区、食堂、供应时间）

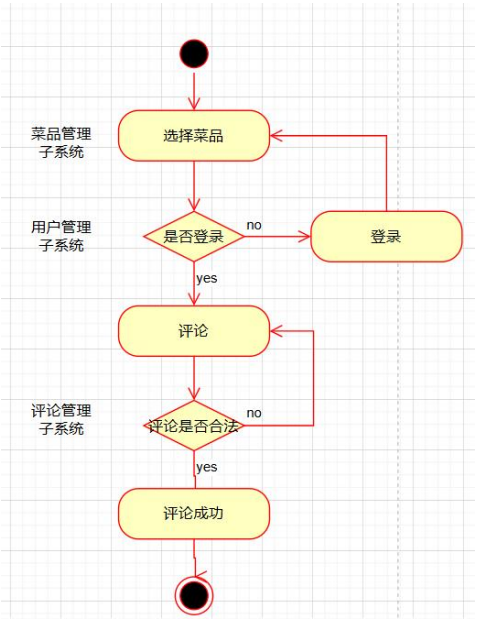
输出：符合条件的菜品列表、菜品详细信息

评价管理子系统：

接口：添加评论接口、获取评论接口

输入：菜品 ID、用户 ID、评分、评论内容

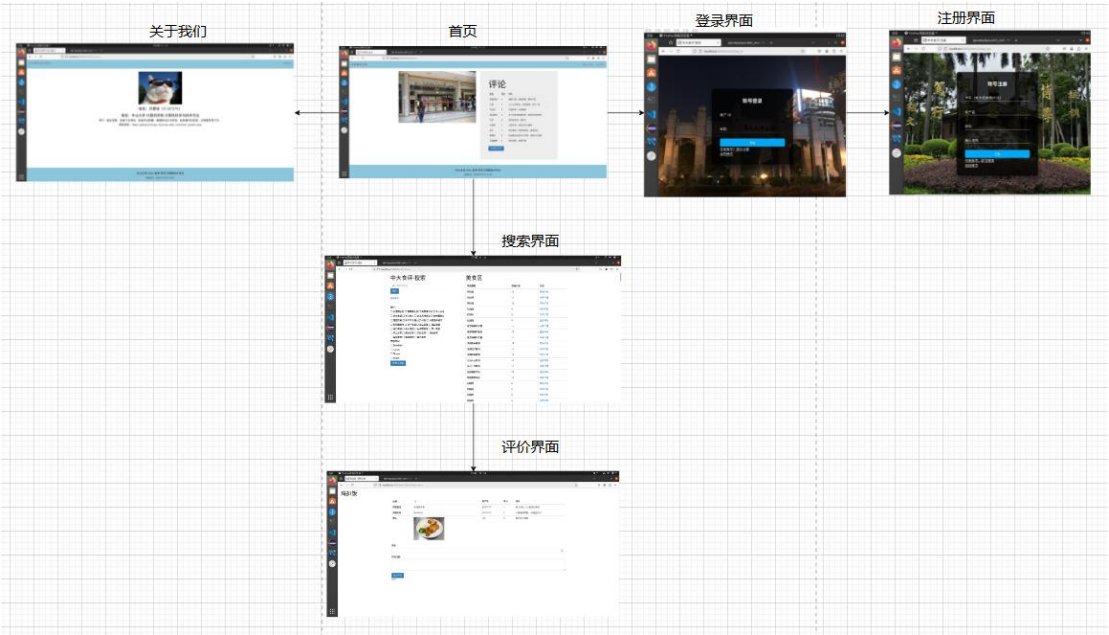
输出：评论添加结果、评论列表



（活动图：显示系统关系）

八、部件设计

网站首页：用户概览整个系统、并能通过一些按钮跳转到其它界面
登录页面、注册界面：表单提交用户名和密码等信息以获得一些权力。
菜品展示页面：显示菜品列表，支持搜索和筛选。
评价展示页面：显示评价列表，支持添加评价。



（页面跳转逻辑图）

九、数据库设计

数据库最终呈现中包含了 6 个实体，分别命名为 Campus (校区)、Cafeteria (餐厅)、Dish (菜品)、ServedTime (供应时间)、Comments (评价) 和 User (用户)

数据库中，实体集以及它们的所属属性如下所示，注意主码使用下划线标明：

• Campus (校区)

包含属性 (campus_id, location, campus_name)

其中主码 campus_id 是标识校区项唯一性的属性，location 标识该校区所在的位置，campus_name 标识该校区的具体校区名字

• Cafeteria (餐厅)

包含属性 (cafeteria_id, cafeteria_name, campus_id)

其中主码 cafeteria_id 是标识餐厅项唯一性的属性，cafeteria_name 标识该餐厅的具体名字，campus_id 是辅助餐厅归属校区的外键属性

• Dish (菜品)

包含属性 (dish_id, dish_name, dish_price, cafeteria_id, served_time_id)

其中主码 dish_id 是标识菜品唯一性的属性，dish_price 标识该菜品的价钱，cafeteria_id 是辅助查找菜品归属餐厅的外键属性，served_time_id 是辅助查找菜品具

体供应时间的外键

• ServedTime (供应时间)

包含属性 (served_time_id, served_time_period)

其中主码 serverd_time_id 是标识供应时间唯一性的属性，serverd_time_period 标识该供应时间的具体时段

• Comments (评价)

包含属性 (comment_id, dish_id, user_id, score, content)

其中主码 comment_id 是标识评价唯一性的属性，dish_id 是辅助找到被评价菜品的外键属性，user_id 是辅助查找所评论用户的外键属性，score 是该评价的所打的分数，具体范围在 0 - 5 之间，content 是该评价的具体内容，且可以是空值

• User (用户)

包含属性 (user_id, password, is_superuser, username)

其中主码 user_id 是标识用户唯一性的属性，password 标识用户登录系统所使用的密码，is_superuser 标识该用户是特权用户还是普通用户，username 标识该用户的昵称

设计一个数据库模型的两个主要缺陷通常是冗余和不完整。为了避免不完整缺陷的产生，我们在数据库设计中多次使用了例如餐厅属性 campus_id、菜品属性 cafeteria_id、评论属性 user_id 等单值外键来辅助关联实体之间的联系；而为了避免冗余缺陷对于数据库存储和效能的不良影响，我们继续建立实体-联系模型来解决这一问题。

在对每对实体建立可能的联系实例之前，我们需要从实体集中删除冗余属性，删除分析过程如下:

1. 在餐厅与校区之间建立联系实例时，由于餐厅内有外键属性 campus_id，而校区主键为 campus_id，因此餐厅的 campus_id 冗余属性被删除

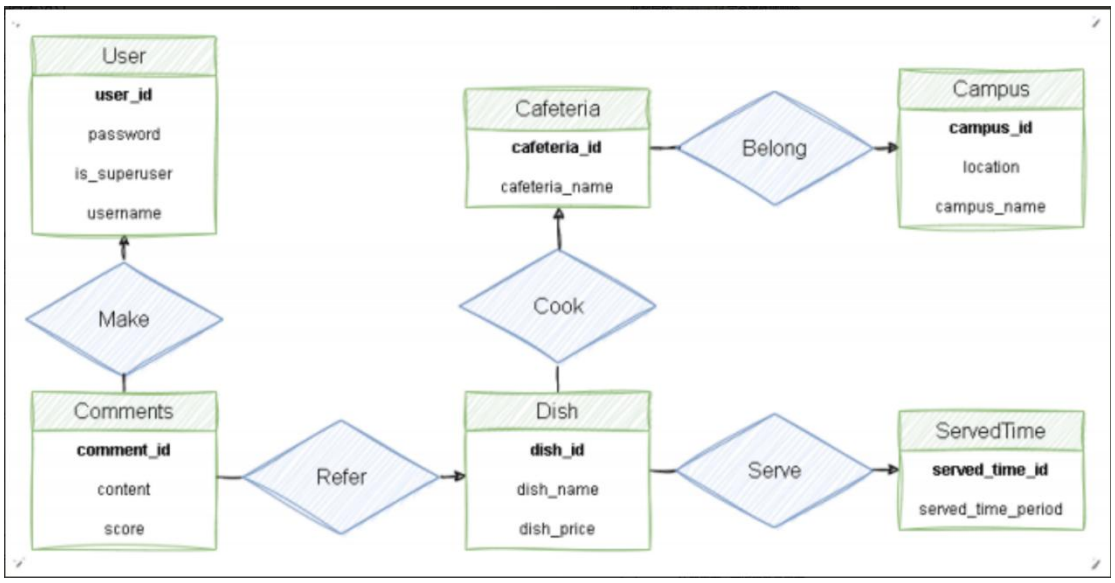
2. 在菜品与餐厅之间建立联系实例时，由于菜品内有外键属性 cafeteria_id，餐厅主键为 cafeteria_id，因此菜品的 cafeteria_id 冗余属性被删除

3. 在菜品与供应时间之间建立联系实例时，由于菜品内有外键属性 served_time_id，供应时间主键为 served_time_id，因此供应时间的 served_time_id 冗余属性被删除

4. 在评价与菜品之间建立联系实例时，由于评价内有外键属性 dish_id，而菜品主键为 dish_id，因此评价的 dish_id 冗余属性被删除

5. 在评价与用户之间建立联系实例时，由于评价内有外键属性 user_id，而用户主键为 user_id，因此评价的 user_id 冗余属性被删除

最终建立来建立实体-联系表如下：



（ER 图）

其中我们具体设计的联系集如下：

Belong：关联餐厅和校区的第一条删除过程，且餐厅与校区的映射基数属于一对多

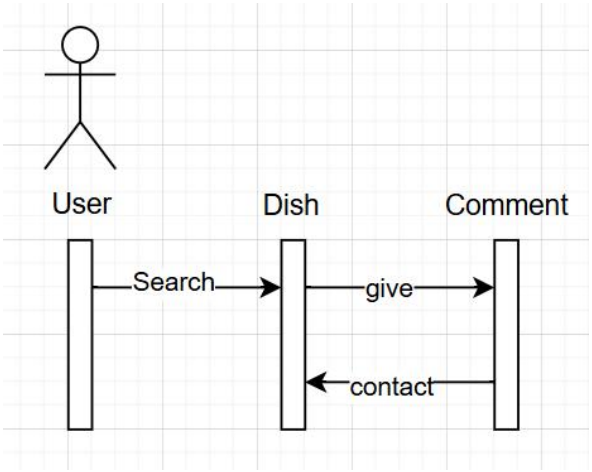
Cook：关联菜品和餐厅的第二条删除过程，且菜品与餐厅的映射基数属于一对多

Serve：关联菜品和供应时间的第三条删除过程，且菜品和供应时间的映射基数属于一对多

Refer：关联评价和菜品的第四条删除过程，且评价和菜品的映射基数属于一对多

Make：关联评价和用户的第五条删除过程，且评价和用户的映射基数属于一对多

十、时序图



十一、状态图