

数据库期末大作业报告

姓名	学号	负责的工作
许志勇 (组长)	21307169	后端功能实现, 协助数据库设计
许嘉璋	21307275	前端功能实现, 协助数据库设计
邝佳轩	21307177	数据库数据搜集, 数据库模式设计

应用介绍

我们小组基于 OpenGauss 实现了中大校园食堂的菜品评价系统——**中大食评**。经过前期小组讨论和调研，我们的数据库包括 6 个实体，并基于 Django 的框架实现 Web 的交互界面，旨在提供涵盖五个校园的食堂菜品信息查看、菜品评价等功能。项目地址：https://github.com/lafe-xzy/database_project/tree/main

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
2. **食堂管理员**: 添加菜品、管理评论、管理菜品信息

功能需求

1. **用户管理**: 用户注册功与登录功能，学生和管理员需要不同的登录权限
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。
5. **数据管理**: 食堂管理人员应该能够对菜品信息进行增删、管理菜品信息。

数据库设计

出于我们小组在概念设计阶段对该大作业——中大校园食堂的菜品评价系统**中大食评**的功能需求的考虑，包括在实现中期时经过协商讨论决定增加的数据库实体，我们小组的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)。接下来的数据库设计主要拆分成对具体实体的属性设计分析，和实体-联系模型的设计分析两个部分进行讲解。

实体属性设计分析

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 served_time_id 是标识供应时间唯一性的属性，served_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

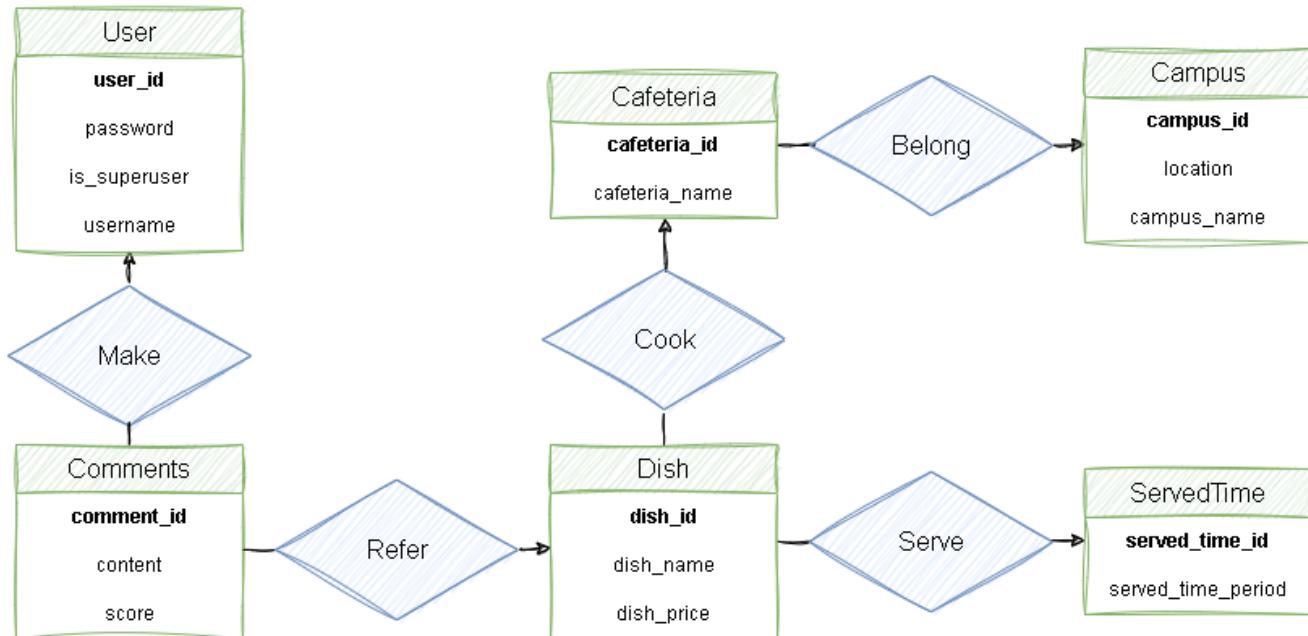
设计一个数据库模型的两个主要缺陷通常是**冗余和不完整**，因此为了避免**不完整**缺陷的产生，我们在数据库设计中多次使用了例如餐厅属性 campus_id、菜品属性 cafeteria_id、评论属性 user_id 等单值外键来辅助关联实体之间的联系，而为了避免**冗余**缺陷对于数据库存储和效能的不良影响，我们继续建立实体-联系模型来解决这一问题。

实体-联系图设计分析

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

在完成了对实体集中的冗余属性对的删除之后，我们按照先前理论设计的实体集来建立实体-联系表：



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

功能实现

开发平台和框架

- 数据库：OpenGauss, OpenEuler
- 前端：Bootstrap
- 后端：Django

前端网页设计



- index**: 网站首页，轮播3张中大饭堂图片、随机显示8条已有评论，可从导航栏进入登录界面（登录前）、账户管理界面（登录后）、网站开发团队简介界面；还可前往搜索、查看更多菜品及其评论。
- log_in**: 登录界面，可返回主界面、前往账户注册界面。
- sign_up**: 注册界面，可返回登录界面。
- search**: 菜品筛选、展示界面，可点击前往查看每个菜品的详细信息及其评论，或在侧边进行筛选、查找，也可返回主界面。
- detail**: 菜品详细信息界面，登录后可发表评论，否则只能查看评论。
- user_info**: 账户信息界面，可查看账户已发表过的评论。
- about_us**: 网站开发团队简介界面，展示了团队成员简要信息及相关工作。

源代码参见附件news/templates/xxx.html

后端功能实现

我们的后端使用 Django 框架的 psycopg2 模块实现与 OpenGauss 的连接，并利用 Django 框架处理前端请求。Django 基于 MVC 模型，因此后端需要处理好 Model+View+Controller 的关系，对应到具体实现就是在 model.py 中按照 E-R 图定义数据库的 5 个实体，在 view.py 中处理前端请求，在 urls.py 定义好路由信息。这里主要介绍 view.py 中主要的函数，更多实现可见代码注释。

搜索功能

搜索界面支持基于菜名的模糊搜索和基于食堂、供应时间的组合搜索。

首先需要为前端提供可用来组合搜索的条件，直接返回食堂和供应时段的信息；

```
# 获取所有食堂、用餐时间，用于搜索界面的筛选
campus_list = Campus.objects.all()
cafeteria_list = Cafeteria.objects.all()
served_time_list = ServedTime.objects.all()
```

接着获取前端传过来的请求的内容：

```

if request.method == 'GET':
    query = request.GET.get('q', '')
    # 获取搜索的菜名
    campus = request.GET.getlist('campus', [])
    # 获取筛选条件的校区
    cafeteria = request.GET.getlist('cafeteria', [])
    # 获取筛选条件的食堂
    served_time = request.GET.getlist('served_time', [])
    # 获取筛选条件的用餐时间

```

执行搜索时需要对搜索结果进行过滤：

模糊搜索使用 Django 内置的过滤方法：“dish_name__icontains” 表示基于指定字符串的模糊搜索。

组合搜索使用 Django 内置的过滤方法：“cafeteria_id__in” 表示存在一个条件满足所给列表，因为勾选的食堂可能不止一个。

```

# 构建查询条件
filters = {}
if query:                      # 基于菜名的模糊查询
    filters['dish_name__icontains'] = query
if len(cafeteria) > 0:          # 基于食堂的查询
    filters['cafeteria_id__in'] = cafeteria
if len(served_time) > 0:         # 基于用餐时间的查询
    filters['served_time_id__in'] = served_time

# 执行查询
search_result = Dish.objects.filter(**filters)

```

最后返回前端信息：

```

return render(request, 'search.html', {'search_result': search_result, 'context': context})

```

菜品信息展示与评论功能

菜品详情页需要菜品的各种信息。这里涉及到多个表的连接，以根据外键依赖获得对应的菜品所在食堂、供应时间的信息。

首先需要进行菜品表和食堂表、供应时间表的自然连接，获取相应菜品信息。

```

dish = Dish.objects.get(dish_id=dish_id)
dish_name = dish.dish_name
dish_price = dish.dish_price
dish_cafe = dish.cafeteria_id.cafeteria_name
dish_served_time = dish.served_time_id.served_time_period
comments = Comments.objects.filter(dish_id=dish_id)

```

如果使用了 POST 方法，说明用户添加了评论，此时需要在 Comments 表执行插入操作：

```
comment = Comments.objects.create(
    comments_id=next_id, score=score, content=content, dish_id=dish,
    username_id=username)
comment.save()
```

登录注册功能

用户登录和注册的请求均为 POST 请求，从 POST 请求获取用户输入的用户名和密码：

```
username = request.POST.get('username')
password = request.POST.get('password')
```

对于注册则根据用户名和密码尝试在用户表中添加用户，默认为普通用户。如果添加失败，如用户名已存在，则会返回错误代码，由前端解析。

```
try:
    # 创建用户
    user = User.objects.create_user(username=username, password=password)
    user.save()
    if user:      # 如果注册成功，跳转到登录页面
        return_code = 1
        return redirect('log_in')
except IntegrityError:
    return_code = 2          # 用户名已存在
    return render(request, 'sign_up.html', {'return_code':return_code})
```

对于登录功能则验证输入的用户名和密码是否正确，正确则登录，跳转至主界面，否则返回错误代码，如用户不存在或者密码错误，由前端解析。

```
try:
    # 验证用户名、密码是否正确
    user = auth.authenticate(username=username, password=password)
    if user:
        auth.login(request, user)    # 正确则登录
        all_dish = Comments.objects.select_related('dish_id').all()
        return_code = 1
        return render(request, 'index.html', {'comment_data':
all_dish, 'return_code':return_code})
    else:
        raise auth.PermissionDenied
except auth.PermissionDenied:        # 用户名不存在或密码错误
    return_code = 2
    return render(request, 'log_in.html', {'return_code':return_code})
```

功能展示

本部分仅展示图片并简单介绍，具体功能可见视频。

主界面——评论展示

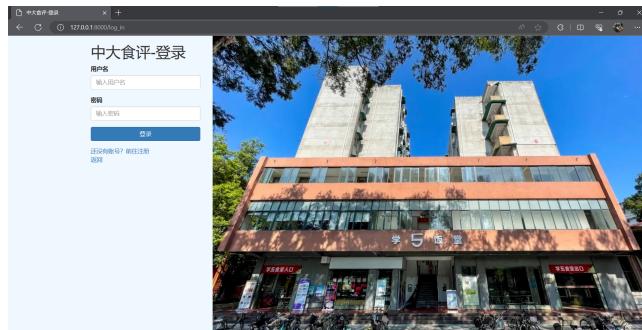


The screenshot shows a web application window titled "中大食评-主页" (Zhuoda Food Review - Home Page) at the URL "127.0.0.1:8000". The page features a large image of a modern building with red columns and green signage for "春晖园". To the right is a "评论" (Reviews) section with a table of food items and their ratings and descriptions. At the bottom, there's a footer with the text "中山大学 2023 秋季 数据库期末大作业".

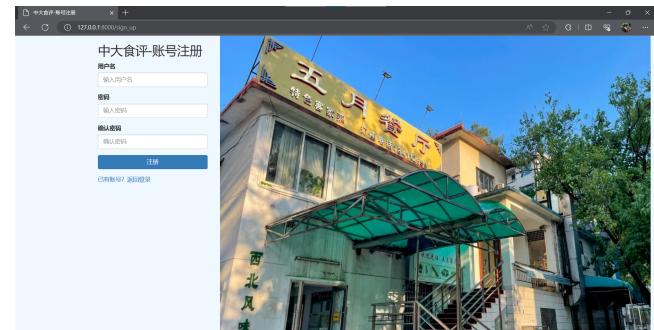
菜品	评分	评价
芭提雅煎牛扒	3.0	菜品一般，可吃但不出彩
辣子鸡	4.0	非常好吃，菜品种类繁多，食材新鲜
韭菜炒猪红	4.0	口感浓郁，味道独特
葱油饼	2.0	口感一般，不太出众
大盘鸡拌面	1.0	严重难以下咽，拒绝再尝试
老排骨	4.0	菜品的外观很美，很有食欲
炸鸡块	2.0	不怎么好吃，味道不佳，没有食欲
面条	2.0	味道平庸，需要增加创新

主界面滚动展示食堂图片，并随机展示评论

登录注册界面



登陆界面



注册界面

搜索界面

The screenshot shows a search interface for a food review system. On the left, there's a search bar with the placeholder "输入搜索关键词" (Input search keyword) and a blue "搜索" (Search) button. Below the search bar are sections for "餐厅:" (Restaurants) and "供应时间:" (Supply time), each with checkboxes for various options like Breakfast, Lunch, Dinner, and Snack. A "筛选 & 组合" (Filter & Combine) button is also present. On the right, a table titled "美食区" (Food Area) lists various dishes with their prices and a "查看详情" (View details) link. The table includes rows for 鸡扒饭 (12.0), 红烧肉 (8.0), 紫苏啫啫鸡套餐 (15.0), etc.

菜品名称	菜品价格	详情
鸡扒饭	12.0	查看详情
鸡扒饭	12.0	查看详情
鸡扒饭	12.0	查看详情
红烧肉	8.0	查看详情
红烧肉	8.0	查看详情
红烧肉	8.0	查看详情
紫苏啫啫鸡套餐	15.0	查看详情
紫苏啫啫鸡套餐	15.0	查看详情
紫苏啫啫鸡套餐	15.0	查看详情
冬阴功手撕鸡	15.0	查看详情
冬阴功手撕鸡	15.0	查看详情
冬阴功手撕鸡	15.0	查看详情
台山十三捞鸡	15.0	查看详情
台山十三捞鸡	15.0	查看详情
芭提雅煎牛扒	18.0	查看详情
芭提雅煎牛扒	18.0	查看详情
回锅肉	6.0	查看详情
回锅肉	6.0	查看详情

搜索界面

左侧上方可以对菜名进行模糊搜索，左侧下方可以对餐厅和用餐时段进行组合搜索，下方演示对第三食堂菜品“茄子”相关菜品的搜索结果。

This screenshot shows the same search interface as the previous one, but with the search term "茄子" entered into the search bar. The results table on the right now shows four entries for "炒茄子" at a price of 6.0 each, with a "查看详情" link for each row.

菜品名称	菜品价格	详情
炒茄子	6.0	查看详情

搜索结果

菜品详情页

鸡扒饭

价格	12.0
供餐餐厅	松涛园食堂
供餐时间	Breakfast
图片	

评分：

评论内容：

[提交评论](#) [返回](#)

菜品详情-鸡扒饭

管理界面

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#) [Change](#)
- Users [+ Add](#) [Change](#)

NEWS

- Cafeterias [+ Add](#) [Change](#)
- Campuss [+ Add](#) [Change](#)
- Comments [+ Add](#) [Change](#)
- Dishes [+ Add](#) [Change](#)
- Served times [+ Add](#) [Change](#)

Recent actions

My actions

- + 21307275 User
- + 21307177 User
- + 21307169 User

菜品详情-鸡扒饭

个人信息界面与关于我们界面

中大食评-账户信息		
		
用户名	21307177	
21307177	1.0	给了6分，十分感谢老师的
21307177	5.0	非常好的
21307177	4.0	数据可以，分量也够
21307177	3.0	味道一般，有点咸
21307177	5.0	菜品解释，口味也佳
21307177	2.0	卖家和蔼大方，发货很快
21307177	4.0	味道不错，性价比很高
21307177	1.0	难以下咽，完全不能吃
21307177	5.0	口味丰富，十分美味
21307177	3.0	没有特色，质量一般
21307177	4.0	菜品不错，种类多样
21307177	2.0	口感欠佳，味道过于单一化
21307177	5.0	菜品独特，食材也很棒

登陆界面



注册界面