

项目开发文档

项目名称	兰州大学点餐平台
项目组成员	蒋嵩林 白舒睿 肖锦恒
专业班级	2018 级计算机基地
指导教师	蔡川
起止日期	2020 12. 8-2020 12. 21

2020 年 12 月

目 录

1. 项目开发计划.....	- 1 -
1.1 项目概述.....	- 1 -
1.2 项目成员及分工.....	- 1 -
1.3 开发环境.....	- 1 -
1.4 实施计划.....	- 1 -
2. 需求分析	- 2 -
2.1 功能需求.....	- 2 -
2.2 性能需求.....	- 2 -
3. 总体设计	- 3 -
3.1 系统设计目标.....	- 3 -
3.2 系统体系结构.....	- 3 -
3.3 可复用资源.....	- 5 -
4. 详细设计	- 6 -
4.1 系统管理.....	- 6 -
5. 数据库设计	- 10 -
5.1 数据库需求分析.....	- 10 -
5.2 数据库概念结构设计（E-R 图）	- 11 -
5.3 数据库逻辑结构设计（Sample）	- 11 -
6. 系统详细设计与实现（核心代码）	- 13 -
总结	- 18 -

1. 项目开发计划

1.1 项目概述

本项目是使用 php, html, JavaScript 等编程语言, 基于 web 开发出的兰州大学点餐平台项目, 本项目的使用群体包括了客户, 商家和管理员三种, 并实现了客户点餐下单, 商家发布菜品查看订单, 管理员系统管理等功能

1.2 项目成员及分工

蒋嵩林 用户模块开发
白舒睿 商家模块开发
肖锦恒 管理员模块开发

1.3 开发环境

Mysql+Apache+php

1.4 实施计划

1.4.1 项目开发过程阶段划分

1. 系统需求汇总
2. ER 图与数据库设计
3. 客户端登录与商家目录界面开发
4. 购物车与结算模块开发
5. 商家模块 (订单查询管理, 商家信息修改等) 开发
6. 管理员 (系统级用户, 订单管理等) 模块开发
7. 功能复查与 bug 检修

1.4.2 项目工作任务分解

本项目使用了 github 作为远程代码仓库, 并使用 git 实现版本控制, 三个组员分别负责三个不同级别用户模块的开发, 并在开发过程中, 通过引入不同级别的用户需求, 不断修改优化数据库, 并将使用到的函数高度集成封装, 便于其他组员在后续开发过程中, 调用已有函数, 做到代码重用。

2. 需求分析

2.1 功能需求

普通用户可以通过校园卡号申请注册用户， 可以添加修改自己的个人信息性别、年龄、电话（必填）、QQ号·email），并可以搜索菜品。包括点击菜品名称能进入菜品详细介绍页面包括已售数量菜品状态

商家可以发布菜品的基本信息（标题、图片、简介、价格），可以修改、删除菜品，可以查看订单信息及订餐者信息（新订单提示）并可以修改订单，包括改变订单状态为已接单 / 已完成 / 已取消） 订单、菜品搜索。

系统管理员可以管理用户，包括浏览、删除用户，可以管理商家，如添加、关闭商家。并能够推荐和取消推荐商家，推荐商家优先 出现在用户首页

2.2 性能需求

1、系统可靠性和安全性

点餐平台的用户，商家，管理员密码存储使用了 sha1 加密算法。系统设置有 customer, merchants 以及 administrator 三种不同的用户级别，用户在登陆系统后，系统可以通过判断登录者属于的相关用户级别，来赋予不同用户不同的操作权限，如高级别用户如系统管理员可以修改商家和用户的信息等。

2、系统的开放性和系统的可扩充性

系统基于 B/S 的客户端应用包含了丰富的图形操作功能，并对所有功能都做了封装，使其与客户端应用程序完全分开，用户只要点击相关功能，即可实现所有已有的图形操作，而不需要了解所用功能的实现过程。

系统中的函数做到了高度封装，具有易于修改易于维护的特点，

3. 总体设计

3.1 系统设计目标

根据真实的点餐平台的需求，仿照着现有的真实点餐平台如饿了么和美团，整体功能参考了老师所提供的 PDF 上的购物车功能，我们将整个点餐平台分为三个功能部分，分别是商家、顾客和管理员，为每个身份的登陆人员都设计了一套属于他们的界面以及他们所需要的功能。商家可以在系统上接单，修改处理菜品及订单，顾客可以在平台上下单支付查看订单信息，管理员可以管理整个平台上的商家和顾客的信息，删除或者添加商家和用户。

3.2 系统体系结构

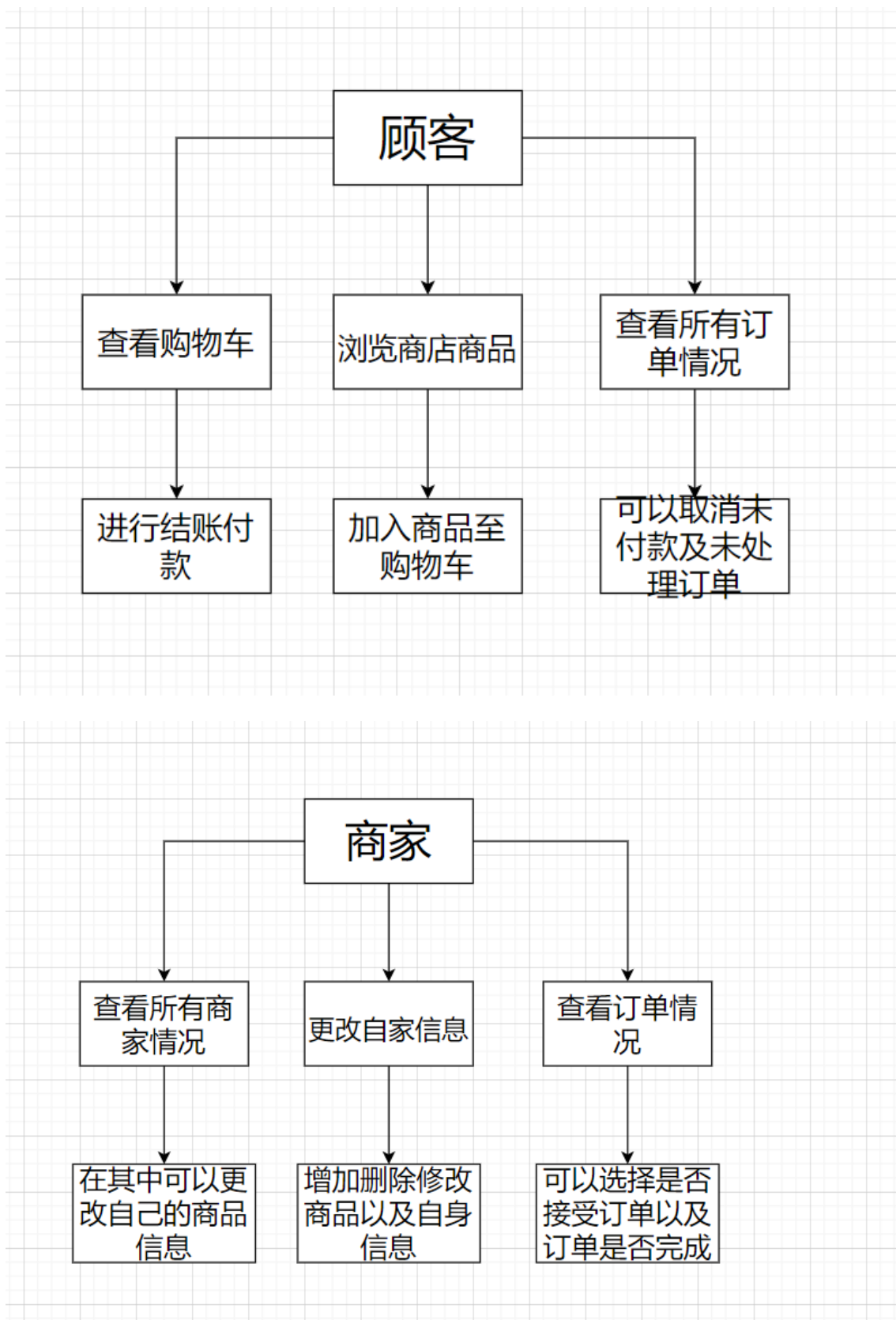
体系结构按照我们所预想的系统设计目标进行实现。

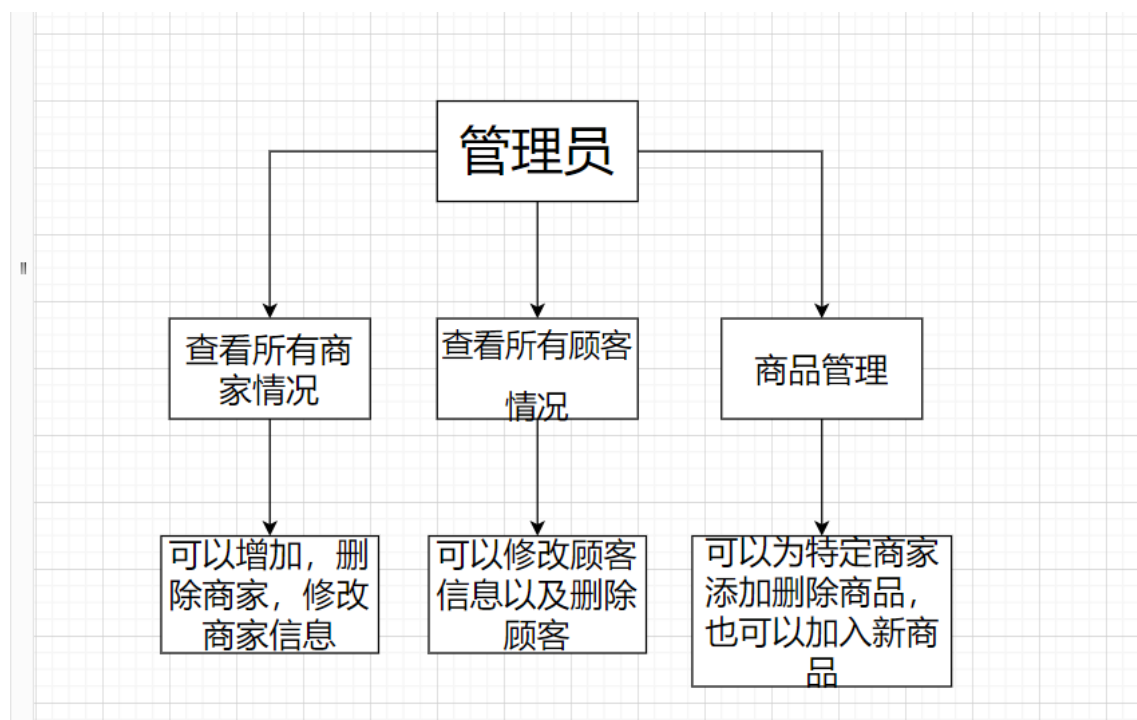
商家在登陆后可以在其功能界面当中查看所有的商家，并且在其中修改、删除或者新增自己的商品，商家还可以修改自己本身的信息，包括自己的密码。商家有对于点餐订单的管理，订单一共有三种状态，分别是新订单，处理中订单和已经处理完成的订单，商家可自行选择是否接单以及选择每个订单完成状态。

对于管理员它可以查看系统当中所有商家的信息并且对其进行修改，也可以为每个商家删除或者新增修改商品，它还可以在系统中新增商家以及删除商家。对于顾客，它也可以删除或者修改顾客的信息，通过顾客名来查询顾客，所有的顾客也都会列在顾客管理的列表当中。最后，它也可以修改自己的密码。

对于顾客，其登陆的页面与商家和管理员不同。登陆成功后，会进入到一个页面可以查看所有的商家及其商品信息。顾客可以点击每个菜品并且将其加入到自己的购物车当中，已加入购物车的商品总价会显示在屏幕右上方，也可以在右上方点击进入购物车查看详细的信息。在右上方中顾客也可以查看到他所点过的所有订单的信息，订单中有四种状态，分别是未支付的订单，正在等待商家处理的订单，商家正在准备的订单以及完成的历史订单。在右上角顾客也可以改变自己的个人信息和密码。

3.2.1 系统功能模块图





3.2.2 开发命名规则

命名规则采用利用每个页面和函数所提供的功能来进行命名的原则，所有名字采用英文缩写来进行命名，如果需要多个单词，我们利用下划线来连接，方便后续的阅读代码，进行查看。

3.3 可复用资源

我们将许多功能都编写成为了一个个的函数来方便进行重复使用，如连接数据库，在页面上展现网页的页眉以及页脚，对于表格的输出等等，整体代码清晰明了，十分简洁，功能丰富，适合于阅读。

4. 详细设计

详细设计是我们软件开发流程终不可或缺的一部分，它是在概要设计的基础上，对概要设计的每个功能模块进一步细化。

4.1 系统管理

4.1.1 功能结构图（Sample）

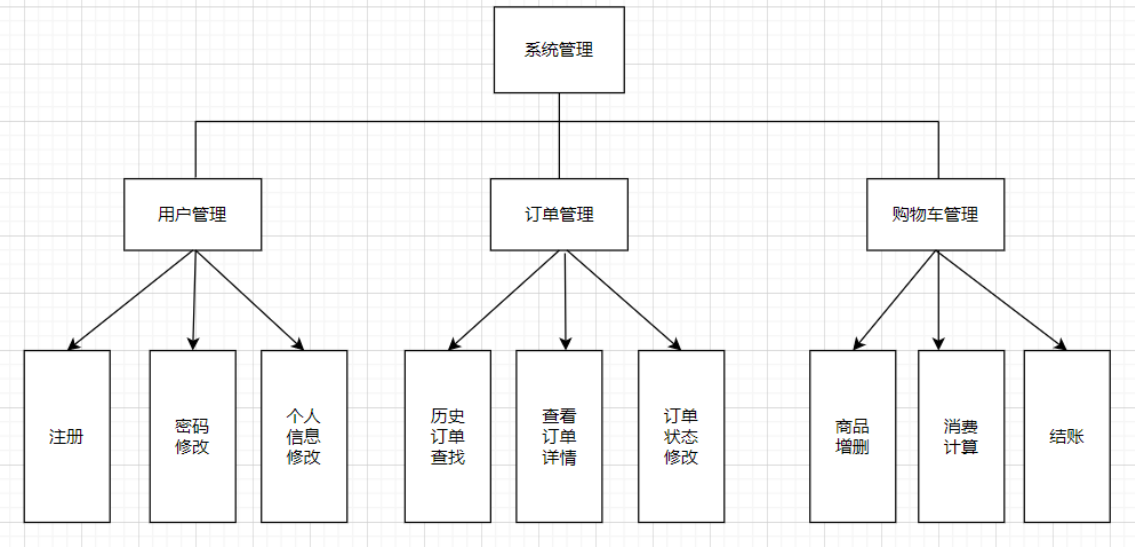
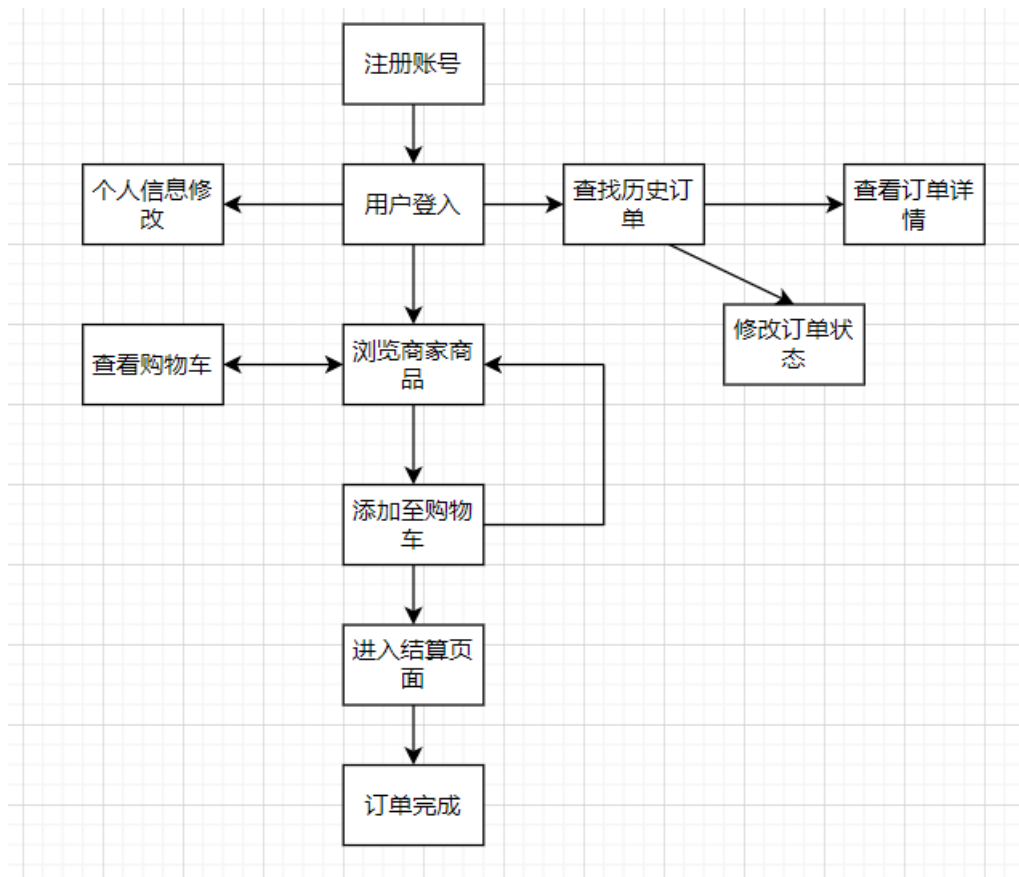


图 4-1-1 系统管理结构图

4.1.2 业务逻辑图



4.1.3 功能项描述

普通用户功能模块

- 申请注册用户（校园卡号）
- 修改用户信息用户可以添加修改自己的个人信息性别、年龄、电话（必填）、QQ号、email）浏览页面：
- 查看商家信息鼠标移到商家log。能看到该商家信息（店名联系方式地址等）
- 查看供应菜品可以搜索菜品。点击菜品名称能进入菜品详细介绍页面包括已售数量菜品状态

商家模块

- 发布菜品：菜品基本信息（标题、图片、简介、价格）
- 菜品管理：修改、删除菜品
- 菜品显示：菜品基本信息已售数量修改菜品状态（售罄）
- 查看订单：商家可以查看订单信息及订餐者信息（新订单提示）

- 修改订单：改变订单状态为已接单 / 已完成 / 已取消） 订单、菜品搜索。

用户订餐功能模块

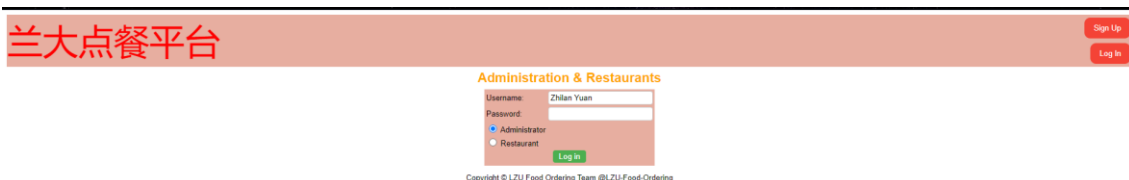
- 餐车功能：用户可以将菜品加入自己的餐车，可以查看修改餐车
- 下单功能：用户可以购买选中的菜品。
- 订单管理：用户可以查看自己的订单状态，查看历史订单。
- 订单搜索功能：用户可以在自己的订单页面，用订单号或者商品 名搜索历史订单

系统管理部分

- 用户管理：系统管理员能够浏览、删除用户。
- 商家管理：系统管理员能够添加、关闭商家。
- 商家推荐：系统管理员能够推荐和取消推荐商家，推荐商家优先 出现在用户首页

4.1.4 界面设计图 （部分展示）

管理员登陆页面：



管理员菜单：

兰大点餐平台

Administrator: admin

Log Out

Go to main site

Restaurant Management:

Add a new Restaurant

Update a Restaurant

Delete a Restaurant

Customer Management

Food Management:

Add a new food

Update, Delete food

Change admin password

Copyright © LZU Food Ordering Team @LZU-Food-Ordering

管理员添加商家页面：

兰大点餐平台

Log Out

Add a Restaurant

Please Confirm:

Restaurant Name:

Password:

Phone:

Address:

Recommend: No

Add Restaurant

Back to administration menu

Copyright © LZU Food Ordering Team @LZU-Food-Ordering

客户查看历史订单页面：

兰大点餐平台


Log Out

Order Management

Search Orders by: Food Name

8/8

New Orders:




Handle Orders: ACCEPT

Change

- Name: sweet and sour chop
- Order id: 39
- Item sold price: 49.99
- Date: 2020-12-12
- Buyer Name: hahah
- Buyer Student ID: 1
- Buyer Address: 38420
- Status: PAID


Handling Orders:



Handle Orders: ACCEPT

Change

- Name: sweet and sour chop
- Order id: 20
- Item sold price: 49.99
- Date: 2020-12-11
- Buyer Name: hahah
- Buyer Student ID: 1
- Buyer Address: 38420
- Status: CANCEL



Handle Orders: ACCEPT

Change

- Name: sweet and sour chop
- Order id: 32
- Item sold price: 49.99
- Date: 2020-12-11
- Buyer Name: hahah
- Buyer Student ID: 1
- Buyer Address: 38420
- Status: CANCEL

Copyright © LZU Food Ordering Team @LZU-Food-Ordering

顾客选购商品页面：



localhost/show_cart.php?new=1

购物车页面：



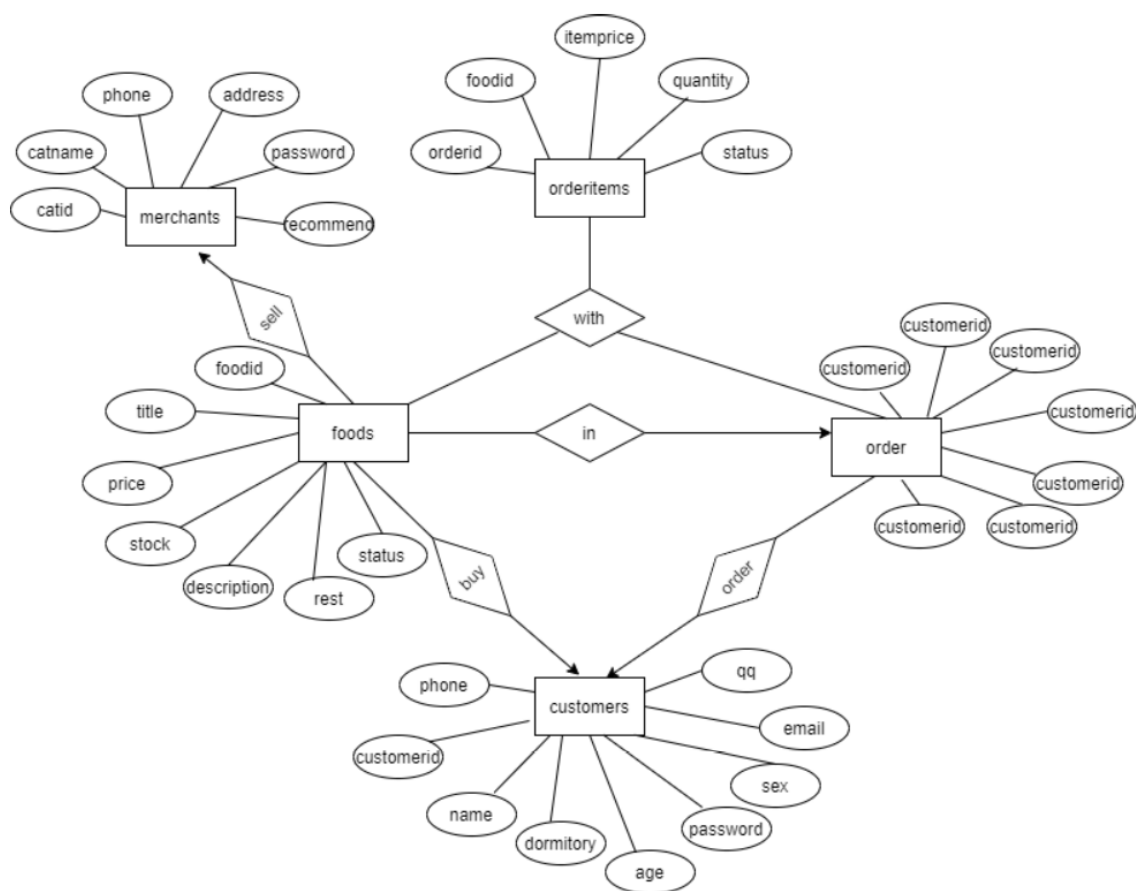
localhost/show_cart.php

5. 数据库设计

5.1 数据库需求分析

要求记录管理员，客户以及管理员的账号信息（如账号，密码等），记录订单信息及订单中的商品信息，记录各种商家以及商家出售的各种食物的相关信息。

5.2 数据库概念结构设计（E-R 图）



5.3 数据库逻辑结构设计（Sample）

Order_items 表：

#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/> 1	orderid 🔑	int(10)		UNSIGNED	否	无		
<input type="checkbox"/> 2	foodid 🔑	int(10)		UNSIGNED	否	无		
<input type="checkbox"/> 3	item_price	float(4,2)			否	无		
<input type="checkbox"/> 4	quantity	tinyint(3)		UNSIGNED	否	无		
<input type="checkbox"/> 5	item_status	varchar(10)	latin1_swedish_ci		否	无		

Orders 表：

	#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/>	1	orderid 	int(10)		UNSIGNED	否	无		AUTO_INCREMENT
<input type="checkbox"/>	2	amount	float(6,2)			是	NULL		
<input type="checkbox"/>	3	date	date			否	无		
<input type="checkbox"/>	4	order_status	char(10)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	5	ship_name	char(60)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	6	ship_customerid	char(80)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	7	ship_dormitory	char(30)	latin1_swedish_ci		否	无		

Merchants 表:

	#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/>	1	catid 	int(10)		UNSIGNED	否	无		AUTO_INCREMENT
<input type="checkbox"/>	2	catname	char(60)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	3	phone	char(20)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	4	address	char(100)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	5	recommend	int(10)		UNSIGNED	是	NULL		
<input type="checkbox"/>	6	password	char(40)	latin1_swedish_ci		否	无		

Foods 表:

	#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/>	1	foodid 	int(10)		UNSIGNED	否	无		AUTO_INCREMENT
<input type="checkbox"/>	2	title	char(100)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	3	catid	int(10)		UNSIGNED	是	NULL		
<input type="checkbox"/>	4	price	float(4,2)			否	无		
<input type="checkbox"/>	5	stock	int(10)		UNSIGNED	是	NULL		
<input type="checkbox"/>	6	status	int(10)		UNSIGNED	是	NULL		
<input type="checkbox"/>	7	description	varchar(255)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	8	rest	varchar(32)	latin1_swedish_ci		否	无		

Customers 表:

	#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/>	1	customerid 🔑	varchar(12)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	2	name	char(60)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	3	dormitory	varchar(20)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	4	password	varchar(50)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	5	sex	int(11)			是	NULL		
<input type="checkbox"/>	6	age	int(11)			是	NULL		
<input type="checkbox"/>	7	phone	varchar(20)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	8	qq	varchar(20)	latin1_swedish_ci		是	NULL		
<input type="checkbox"/>	9	email	varchar(50)	latin1_swedish_ci		是	NULL		

Admin 表:

	#	名字	类型	排序规则	属性	空	默认	注释	额外
<input type="checkbox"/>	1	username 🔑	char(16)	latin1_swedish_ci		否	无		
<input type="checkbox"/>	2	password	char(40)	latin1_swedish_ci		否	无		

6. 系统详细设计与实现（核心代码）

例:

1.打印商家的食物列表函数:

```
function display_foods($food_array)
{
    //display all foods in the array passed in
    if (!empty($_GET['key']) && !is_array($food_array)) {
        echo "<p>No foods currently available with the keyword: " . $_GET['key'] . "</p>";
    } else if (!is_array($food_array)) {
        echo "<p>No foods currently available in this merchant</p>";
    } else {
        echo "
        <form align=\"right\" method=\"get\" action=\"show_cat.php\">
```

```

    Search Food:
    <input type="hidden" name="catid" value="" . $_G
ET['catid'] . ">
    <input type="text" name="key">
    <input type="submit">
    </form>";
    //create table
    echo "<table width="100%" border="0">";

    //create a table row for each food
    foreach ($food_array as $row) {
        $url = "show_food.php?foodid=" . urlencode($row['fo
oodid']);
        echo "<tr><td>";
        if (@file_exists("images/{$row['foodid']}.jpg")) {
            $title = "";
            do_html_url($url, $title);
        } else {
            echo "&nbsp;";
        }
        echo "</td><td>";
        $title = htmlspecialchars($row['title']) . " by " .
htmlspecialchars($row['rest']);
        do_html_url($url, $title);
        echo "</td></tr>";
    }

    echo "</table>";
}

echo "<hr />";
}

```

2.打印食物细节函数:

```

function display_food_details($food)
{

```



```
// display all details about this food
if (is_array($food)) {
    echo "<table><tr>";
    //display the picture if there is one
    if (@file_exists("images/{$food['foodid']}.jpg")) {
        $size = GetImageSize("images/{$food['foodid']}.jpg"
    );
        if (($size[0] > 0) && ($size[1] > 0)) {
            echo "<td><img src=\"images/" . htmlspecialchars(
$food['foodid']) . ".jpg\"
                height=\"125px\" style=\"border: 1px solid bl
ack\"/></td>";
        }
    }
    echo "<td><ul>";
    echo "<li><strong>Name:</strong> ";
    echo htmlspecialchars($food['title']);
    echo "</li><li><strong>Stock:</strong> ";
    echo htmlspecialchars($food['stock']);
    echo "</li><li><strong>Sold:</strong> ";
    echo htmlspecialchars(get_sold_amount($food['foodid']
));
    echo "</li><li><strong>Provided by:</strong> ";
    echo htmlspecialchars($food['rest']);
    echo "</li><li><strong>Our Price:</strong> ";
    echo number_format($food['price'], 2);
    echo "</li><li><strong>Description:</strong> ";
    echo htmlspecialchars($food['description']);
    echo "</li><li><strong>Status:</strong> ";
    if ($food['status'] == 1) {
        echo "Available";
    } else {
        echo "Unavailable";
    }
    echo "</li></ul></td></tr></table>";
} else {
    echo "<p>The details of this food cannot be displayed
at this time.</p>";
}
echo "<hr />";
```

}

3.打印购物车函数:

```
function display_cart($cart, $change = true, $images = 1)
{
    // display items in shopping cart
    // optionally allow changes (true or false)
    // optionally include images (1 - yes, 0 - no)

    echo "<table border=\"0\" width=\"100%\" cellpadding=\"0\">
        <form action=\"show_cart.php\" method=\"post\">
        <tr><th colspan=\"\" . (1 + $images) . \"\" bgcolor=
\"#E7AEA0\"></th>
        <th bgcolor=\"#E7AEA0\">Restaurant</th>
        <th bgcolor=\"#E7AEA0\">Price</th>
        <th bgcolor=\"#E7AEA0\">Quantity</th>
        <th bgcolor=\"#E7AEA0\">Total</th>
        </tr>";

    //display each item as a table row
    foreach ($cart as $foodid => $qty) {
        $food = get_food_details($foodid);
        echo "<tr>";

        if ($images == true) {
            echo "<td align=\"left\">";
            if (file_exists("images/{$foodid}.jpg")) {
                $size = GetImageSize("images/{$foodid}.jpg");
                if (($size[0] > 0) && ($size[1] > 0)) {
                    echo "<img src=\"images/\" . htmlspecialchars($f
oodid) . ".jpg\"
                        style=\"border: 1px solid black\"
                        height=\"42px\"/>";
                }
            } else {
                echo "&nbsp;";
            }
            echo "</td>";
        }
    }
}
```

```
}

    echo "<td align=\"left\">
        <a href=\"show_food.php?foodid=" . urlencode($foodid) . "\">" . htmlspecialchars($food['title']) . "</a></td>
    >
        <td align=\"center\">" . htmlspecialchars($food['rest']) . "</td>
        <td align=\"center\">
¥" . number_format($food['price'], 2) . "</td>
        <td align=\"center\">";

        // if we allow changes, quantities are in text boxes
        if ($change == true) {
            echo "<input type=\"number\" name=\"" . htmlspecialchars($foodid) . "\" value=\"" . htmlspecialchars($qty) . "\" min=0 max=\"" . htmlspecialchars(get_stock($food['foodid'])) . "\" size=\"3\">";
        } else {
            echo $qty;
        }
        echo "</td><td align=\"center\">
¥" . number_format($food['price'] * $qty, 2) . "</td></tr>
\n";
    }
    // display total row
    echo "<tr>
        <th colspan=\"" . (4 + $images) . "\" bgcolor=\"#E7AEA0\">&nbsp;</th>
        <th align=\"center\" bgcolor=\"#E7AEA0\">
¥" . number_format($_SESSION['total_price'], 2) . "
        </th>
    </tr>";

    // display save change button
    if ($change == true) {
```

```
        echo "<tr>
            <td colspan=\"\" . (2) . \">&nbsp;</td>

            <td>&nbsp;</td>
        </tr></table>
        <div align=\"right\">
            <input type=\"hidden\" name=\"save\" value=\"true
\"/>

            <input type=\"submit\" value=\"Save Changes\"/>
        </div>";
    }
    echo "</form>";
}
```

总结

本次基于 php 的兰州大学点餐平台开发中，我们在功能中实现了基于订单管理，用户管理和菜品管理的点餐平台系统，对使用到的相关数据库进行了设计，并进行了功能和页面的优化，基本实现了用户点餐至购物车，下单付款；商家查看订单状态，并对餐品进行相关管理，以及管理员对整个系统进行管理的功能。之后的 bug 的检验以及维修工作，也在小组成员们分工合作下基本完成。系统后续还会存在许多可以优化方面，以及一些未能及时发现的 bug，希望发现者及时与我们联系，我们会悉心采纳你们的宝贵意见。