

- 系统详细设计说明书
 - 系统功能概述
 - 前端功能
 - Web服务端
 - 数据库端
 - 系统功能模块结构
 - 用户管理模块
 - 菜单管理模块
 - 原料管理模块
 - 系统界面设计
 - 登录页面 (login.html)
 - 用户信息列表页面 (user_info_list.html)
 - 添加用户页面 (user_add.html)
 - 菜单信息列表页面 (menu_info.html)
 - 菜单查找页面 (menu_find.html)
 - 添加菜单页面 (auth_add.html)
 - 系统物理模型
 - 表结构
 - 系统安全体系设计
 - 用户管理与控制
 - 存储与恢复
 - 系统运行环境设计与部署结构
 - 开发环境
 - 生产环境
 - 源代码列表及说明

系统详细设计说明书

系统功能概述

系统主要分为前端功能、Web服务端和数据库端三个部分。

前端功能

通过HTML页面与用户交互，使用表单提交数据，展示查询结果。

Web服务端

使用Django框架处理请求，进行数据验证和业务逻辑处理，与数据库交互。

数据库端

使用SQLite数据库存储用户信息、菜单信息和原料信息。

系统功能模块结构

用户管理模块

- 登录页面
- 用户信息列表页面
- 添加用户页面

菜单管理模块

- 菜单信息列表页面
- 菜单查找页面
- 添加菜单页面

原料管理模块

- 原料信息列表页面

系统界面设计

登录页面 (login.html)

```
<form method="post">
  {% csrf_token %}
  <label for="user">Username:</label>
```

```
<input type="text" id="user" name="user">
<label for="password">Password:</label>
<input type="password" id="password" name="password">
<button type="submit">Login</button>
</form>
```

用户信息列表页面 (user_info_list.html)

```
<table>
  <tr>
    <th>Name</th>
    <th>Phone</th>
    <th>Address</th>
  </tr>
  {% for user in data_list %}
  <tr>
    <td>{{ user.name }}</td>
    <td>{{ user.phone }}</td>
    <td>{{ user.address }}</td>
  </tr>
  {% endfor %}
</table>
```

添加用户页面 (user_add.html)

```
<form method="post">
  {% csrf_token %}
  <label for="user">Username:</label>
  <input type="text" id="user" name="user">
  <label for="password">Password:</label>
  <input type="password" id="password" name="password">
  <label for="phone">Phone:</label>
  <input type="text" id="phone" name="phone">
  <label for="address">Address:</label>
  <input type="text" id="address" name="address">
  <button type="submit">Add User</button>
</form>
```

菜单信息列表页面 (menu_info.html)

```
<table>
  <tr>
```

```
        <th>Name</th>
        <th>Price</th>
        <th>Marketer</th>
    </tr>
    {% for menu in data_list %}
    <tr>
        <td>{{ menu.name }}</td>
        <td>{{ menu.price }}</td>
        <td>{{ menu.marketer }}</td>
    </tr>
    {% endfor %}
</table>
```

菜单查找页面 (menu_find.html)

```
<form method="post">
    {% csrf_token %}
    <label for="name">Menu Name:</label>
    <input type="text" id="name" name="name">
    <button type="submit">Find Menu</button>
</form>
{% if name and price %}
<p>Menu: {{ name }} - Price: {{ price }}</p>
{% else %}
<p>Menu not found.</p>
{% endif %}
```

添加菜单页面 (auth_add.html)

```
<form method="post">
    {% csrf_token %}
    <label for="name">Menu Name:</label>
    <input type="text" id="name" name="name">
    <label for="price">Price:</label>
    <input type="text" id="price" name="price">
    <label for="marketer">Marketer:</label>
    <input type="text" id="marketer" name="marketer">
    <button type="submit">Add Menu</button>
</form>
```

系统物理模型

表结构

```
class UserInfo(models.Model):
    name = models.CharField(max_length=50)
    password = models.CharField(max_length=64)
    phone = models.CharField(max_length=50)
    address = models.CharField(max_length=30)

class MenuInfo(models.Model):
    name = models.CharField(max_length=50)
    price = models.CharField(max_length=10)
    marketer = models.CharField(max_length=50)

class MarketerInfo(models.Model):
    material = models.CharField(max_length=50)
    staff = models.CharField(max_length=50)
```

系统安全体系设计

用户管理与控制

- 仅授权用户可以登录系统。
- 用户密码使用加密存储。

存储与恢复

- 数据库定期备份，确保数据安全。

系统运行环境设计与部署结构

开发环境

- 操作系统：Windows 10 / Ubuntu 20.04
- 开发工具：Visual Studio Code / PyCharm
- 依赖库：Django 3.x

生产环境

- 服务器：Ubuntu 20.04
- Web服务器：Gunicorn / Nginx
- 数据库：SQLite

源代码列表及说明

```
# models.py
from django.db import models

class UserInfo(models.Model):
    name = models.CharField(max_length=50)
    password = models.CharField(max_length=64)
    phone = models.CharField(max_length=50)
    address = models.CharField(max_length=30)

class MenuInfo(models.Model):
    name = models.CharField(max_length=50)
    price = models.CharField(max_length=10)
    marketer = models.CharField(max_length=50)

class MarketerInfo(models.Model):
    material = models.CharField(max_length=50)
    staff = models.CharField(max_length=50)
```

```
# views.py
from django.shortcuts import render, redirect
from django.http import HttpResponse
from .models import UserInfo, MenuInfo

def login(request):
    if request.method == "GET":
        return render(request, "login.html")
    else:
        username = request.POST.get("user")
        password = request.POST.get("password")
        return HttpResponse(f"hello {username}")

def user_info_list(request):
    data_list = UserInfo.objects.all()
    return render(request, "user_info_list.html", {"data_list": data_list})

def user_add(request):
    if request.method == "GET":
        return render(request, "user_add.html")
```

```

user = request.POST.get("user")
password = request.POST.get("password")
phone = request.POST.get("phone")
address = request.POST.get("address")

if user and password:
    UserInfo.objects.create(name=user, password=password, phone=phone,
address=address)
    return redirect("/user/info")

def menu_info(request):
    data_list = MenuInfo.objects.all()
    return render(request, "menu_info.html", {"data_list":data_list})

def menu_find(request):
    if request.method == "GET":
        return HttpResponse("you shouldn't go there this time")

    name = request.POST.get("name")
    data_list = MenuInfo.objects.all()

    for item in data_list:
        if item.name == name:
            return render(request, "menu_find.html", {"name":name,
"price":item.price})
    return HttpResponse("can't find this name in the menu")

def auth_way(request):
    return render(request, "auth_way.html")

def auth_add(request):
    if request.method == "GET":
        return HttpResponse("you shouldn't go there this time")

    name = request.POST.get("name")
    price = request.POST.get

("price")
    marketer = request.POST.get("marketer")

    if name and price and marketer:
        MenuInfo.objects.create(name=name, price=price, marketer=marketer)
    return redirect("/menu/info")

```

```

# urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('login/', views.login),
    path('user/info/', views.user_info_list),
    path('user/add/', views.user_add),
    path('menu/info/', views.menu_info),
    path('menu/info/find/', views.menu_find),
    path('auth/', views.auth_way),

```

```
path('auth/add/', views.auth_add),  
]
```