**Django学生信息管理系统**

## 数据库设计

CREATE DATABASE sms;

USE sms;

CREATE TABLE admin (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

adminName VARCHAR(50),

adminPassword VARCHAR(50)

);

CREATE TABLE student (

stuId INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

stuName VARCHAR(50),

stuClass VARCHAR(50),

stuIntro TEXT

);

## 一、创建Django项目和应用

python manage.py startapp sms

## 配置数据库

1. **settings.py配置**

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'NAME': 'sms',

'USER': 'root,

'PASSWORD': 'root,

'HOST': '127.0.0.1',

'PORT': '3306,

}

}

1. **安装mysqlclient库**

pip install mysqlclient

## 创建模型

在 sms/models.py 文件中定义模型：

from django.db import models  
  
class Admin(models.Model):  
 adminName = models.CharField(max\_length=100)  
 adminPassword = models.CharField(max\_length=100)  
 class Meta:  
 db\_table = 'admin'  
  
class Student(models.Model):  
 stuId = models.IntegerField(primary\_key=True)  
 stuName = models.CharField(max\_length=100)  
 stuClass = models.CharField(max\_length=100)  
 stuIntro = models.TextField()  
 class Meta:  
 db\_table = 'student'

## 创建视图

在 sms/views.py 文件中创建视图：

# sms/views.py  
from django.db.models import Count # 导入 Count  
from django.shortcuts import render, redirect  
from .models import Admin, User, Student  
from django.core.paginator import Paginator  
  
# 检查用户是否是管理员  
def is\_admin(request):  
 return 'admin\_id' in request.session and request.session['is\_admin']  
  
# 检查用户是否已登录（管理员或普通用户）  
def is\_logged\_in(request):  
 return 'admin\_id' in request.session or 'user\_id' in request.session  
  
# 管理员注册视图  
def admin\_register(request):  
 if request.method == 'POST':  
 admin\_name = request.POST.get('adminName')  
 admin\_password = request.POST.get('adminPassword')  
 admin = Admin(adminName=admin\_name, adminPassword=admin\_password, is\_admin=True)  
 admin.save()  
 return redirect('admin\_login')  
 return render(request, 'admin\_register.html')  
  
# 普通用户注册视图  
def user\_register(request):  
 if request.method == 'POST':  
 user\_name = request.POST.get('userName')  
 user\_password = request.POST.get('userPassword')  
 user = User(userName=user\_name, userPassword=user\_password, is\_admin=False)  
 user.save()  
 return redirect('user\_login')  
 return render(request, 'user\_register.html')  
  
# 管理员登录视图  
def admin\_login(request):  
 if request.method == 'POST':  
 admin\_name = request.POST.get('adminName')  
 admin\_password = request.POST.get('adminPassword')  
 try:  
 admin = Admin.objects.get(adminName=admin\_name, adminPassword=admin\_password)  
 request.session['admin\_id'] = admin.id  
 request.session['is\_admin'] = True # 标记为管理员  
 return redirect('student\_list') # 登录成功后跳转到学生列表页面  
 except Admin.DoesNotExist:  
 return render(request, 'admin\_login.html', {'error': '用户名或密码错误'})  
 return render(request, 'admin\_login.html')  
  
# 普通用户登录视图  
def user\_login(request):  
 if request.method == 'POST':  
 user\_name = request.POST.get('userName')  
 user\_password = request.POST.get('userPassword')  
 try:  
 user = User.objects.get(userName=user\_name, userPassword=user\_password)  
 request.session['user\_id'] = user.id  
 request.session['is\_admin'] = False # 标记为普通用户  
 return redirect('student\_list') # 登录成功后跳转到学生列表页面  
 except User.DoesNotExist:  
 return render(request, 'user\_login.html', {'error': '用户名或密码错误'})  
 return render(request, 'user\_login.html')  
  
  
def dashboard(request):  
 # 检查用户是否登录  
 if 'admin\_id' not in request.session and 'user\_id' not in request.session:  
 return redirect('admin\_login') # 如果未登录，重定向到管理员登录页面  
  
 # 学生总数  
 total\_students = Student.objects.count()  
  
 # 班级分布  
 class\_distribution = Student.objects.values('stuClass').annotate(count=Count('stuId')).order\_by('stuClass')  
  
 # 将班级名称转换为整数并排序  
 class\_distribution = sorted(  
 class\_distribution,  
 key=lambda x: int(x['stuClass']) # 按班级名称的整数值排序  
 )  
  
 # 提取排序后的班级名称和学生数量  
 class\_names = [str(item['stuClass']) for item in class\_distribution] # 转换为字符串  
 class\_counts = [item['count'] for item in class\_distribution]  
  
 # 最近添加的学生  
 recent\_students = Student.objects.order\_by('-stuId')[:5] # 获取最近添加的5名学生  
  
 # 渲染仪表盘页面，并传递数据  
 return render(request, 'dashboard.html', {  
 'total\_students': total\_students,  
 'class\_names': class\_names,  
 'class\_counts': class\_counts,  
 'recent\_students': recent\_students  
 })  
  
# 学生列表视图（管理员和普通用户均可访问）  
def student\_list(request):  
 if not is\_logged\_in(request):  
 return redirect('admin\_login') # 如果未登录，重定向到管理员登录页面  
  
 students = Student.objects.all()  
 paginator = Paginator(students, 10)  
 page\_number = request.GET.get('page')  
 page\_obj = paginator.get\_page(page\_number)  
 return render(request, 'student\_list.html', {'page\_obj': page\_obj})  
  
# 创建学生视图（管理员和普通用户均可访问）  
def student\_create(request):  
 if not is\_logged\_in(request):  
 return redirect('admin\_login') # 如果未登录，重定向到管理员登录页面  
  
 if request.method == 'POST':  
 stu\_id = request.POST.get('stuId')  
 stu\_name = request.POST.get('stuName')  
 stu\_class = request.POST.get('stuClass')  
 stu\_intro = request.POST.get('stuIntro')  
 student = Student(stuId=stu\_id, stuName=stu\_name, stuClass=stu\_class, stuIntro=stu\_intro)  
 student.save()  
 return redirect('student\_list')  
 return render(request, 'student\_create.html')  
  
# 更新学生视图（管理员和普通用户均可访问）  
def student\_update(request, stuId):  
 if not is\_logged\_in(request):  
 return redirect('admin\_login') # 如果未登录，重定向到管理员登录页面  
  
 student = Student.objects.get(stuId=stuId)  
 if request.method == 'POST':  
 student.stuName = request.POST.get('stuName')  
 student.stuClass = request.POST.get('stuClass')  
 student.stuIntro = request.POST.get('stuIntro')  
 student.save()  
 return redirect('student\_list')  
 return render(request, 'student\_update.html', {'student': student})  
  
# 删除学生视图（管理员和普通用户均可访问）  
def student\_delete(request, stuId):  
 if not is\_logged\_in(request):  
 return redirect('admin\_login') # 如果未登录，重定向到管理员登录页面  
  
 student = Student.objects.get(stuId=stuId)  
 student.delete()  
 return redirect('student\_list')  
  
# 普通用户列表视图（仅管理员可访问）  
def user\_list(request):  
 if not is\_admin(request):  
 return redirect('admin\_login') # 如果不是管理员，重定向到管理员登录页面  
  
 users = User.objects.all()  
 paginator = Paginator(users, 10)  
 page\_number = request.GET.get('page')  
 page\_obj = paginator.get\_page(page\_number)  
 return render(request, 'user\_list.html', {'page\_obj': page\_obj})  
  
# 创建普通用户视图（仅管理员可访问）  
def user\_create(request):  
 if not is\_admin(request):  
 return redirect('admin\_login') # 如果不是管理员，重定向到管理员登录页面  
  
 if request.method == 'POST':  
 user\_name = request.POST.get('userName')  
 user\_password = request.POST.get('userPassword')  
 user = User(userName=user\_name, userPassword=user\_password, is\_admin=False)  
 user.save()  
 return redirect('user\_list')  
 return render(request, 'user\_create.html')  
  
# 更新普通用户视图（仅管理员可访问）  
def user\_update(request, userId):  
 if not is\_admin(request):  
 return redirect('admin\_login') # 如果不是管理员，重定向到管理员登录页面  
  
 user = User.objects.get(id=userId)  
 if request.method == 'POST':  
 user.userName = request.POST.get('userName')  
 user.userPassword = request.POST.get('userPassword')  
 user.save()  
 return redirect('user\_list')  
 return render(request, 'user\_update.html', {'user': user})  
  
# 删除普通用户视图（仅管理员可访问）  
def user\_delete(request, userId):  
 if not is\_admin(request):  
 return redirect('admin\_login') # 如果不是管理员，重定向到管理员登录页面  
  
 user = User.objects.get(id=userId)  
 user.delete()  
 return redirect('user\_list')

## 配置 URL 路由

在 sms/urls.py 文件中配置 URL 路由（如果没有urls.py，需要新建）：

# sms/urls.py  
  
from django.urls import path  
from sms import views  
  
urlpatterns = [  
 path('admin\_register/', views.admin\_register, name='admin\_register'),  
 path('admin\_login/', views.admin\_login, name='admin\_login'),  
 path('user\_register/', views.user\_register, name='user\_register'),  
 path('user\_login/', views.user\_login, name='user\_login'),  
 path('dashboard/', views.dashboard, name='dashboard'),  
 path('student\_list/', views.student\_list, name='student\_list'),  
 path('student\_create/', views.student\_create, name='student\_create'),  
 path('student\_update/<int:stuId>/', views.student\_update, name='student\_update'),  
 path('student\_delete/<int:stuId>/', views.student\_delete, name='student\_delete'),  
 path('user\_list/', views.user\_list, name='user\_list'), # 普通用户列表  
 path('user\_create/', views.user\_create, name='user\_create'), # 创建普通用户  
 path('user\_update/<int:userId>/', views.user\_update, name='user\_update'), # 更新普通用户  
 path('user\_delete/<int:userId>/', views.user\_delete, name='user\_delete'), # 删除普通用户  
]

在项目的 urls.py 文件中包含应用的 URL 配置：

from django.contrib import admin  
from django.urls import path, include  
from django.views.generic.base import RedirectView  
  
urlpatterns = [  
 # 配置 Django 管理后台的 URL，访问 /admin/ 可以进入管理后台  
 path('admin/', admin.site.urls),  
 # 配置根 URL 重定向，当访问网站根目录时，重定向到 /admin\_login/ 页面  
 path('', RedirectView.as\_view(url='/admin\_login/')),  
 # 配置包含子应用 sms 的 URL，当访问以 / 开头的 URL 时，会调用 sms.urls 中的 URL 配置  
 path('', include('sms.urls')),  
]

## 创建模板

在 templates 目录下创建以下 HTML 模板：

**admin\_register.html代码如下：**

{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>管理员注册</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <div class="title-container fade-in">  
 <h1 class="text-center mb-4"><i class="fas fa-user-plus"></i> 管理员注册</h1>  
 </div>  
 <form method="post" class="card p-4 shadow">  
 <div class="mb-3">  
 <label for="adminName" class="form-label">管理员名称:</label>  
 <input type="text" name="adminName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="adminPassword" class="form-label">管理员密码:</label>  
 <input type="password" name="adminPassword" class="form-control" required>  
 </div>  
 <button type="submit" class="btn btn-primary">注册</button>  
 <p class="mt-3">已有账号？ <a href="{% url 'admin\_login' %}">点击登录</a></p>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

**admin\_login.html代码如下：**

<!-- templates/admin\_login.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>管理员登录</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">管理员登录 <i class="fas fa-sign-in-alt"></i></h1> <!-- 添加登录图标 -->  
 {% if error %}  
 <div class="alert alert-danger">  
 <i class="fas fa-exclamation-circle"></i> {{ error }} <!-- 添加错误提示图标 -->  
 </div>  
 {% endif %}  
 <form method="post" class="card p-4 shadow">  
 {% csrf\_token %}  
 <div class="mb-3">  
 <label for="adminName" class="form-label">  
 <i class="fas fa-user-shield"></i> 管理员名称: <!-- 添加管理员图标 -->  
 </label>  
 <input type="text" name="adminName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="adminPassword" class="form-label">  
 <i class="fas fa-lock"></i> 管理员密码: <!-- 添加锁图标 -->  
 </label>  
 <input type="password" name="adminPassword" class="form-control" required>  
 </div>  
 <button type="submit" class="btn btn-primary">  
 <i class="fas fa-sign-in-alt"></i> 登录 <!-- 添加登录按钮图标 -->  
 </button>  
 <p class="mt-3">  
 <i class="fas fa-user-plus"></i> 没有账号? <a href="{% url 'admin\_register' %}">点击注册</a> <!-- 添加注册图标 -->  
 </p>  
 <p class="mt-3">  
 <i class="fas fa-user"></i> 普通用户登录? <a href="{% url 'user\_login' %}">点击这里</a> <!-- 添加普通用户图标 -->  
 </p>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

**student\_list.html代码如下：**

{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>学生列表</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">学生列表 <i class="fas fa-users"></i></h1> <!-- 添加学生列表图标 -->  
  
 <!-- 仪表盘按钮（所有已登录用户均可访问） -->  
 <a href="{% url 'dashboard' %}" class="btn btn-primary mb-3">  
 <i class="fas fa-tachometer-alt"></i> 仪表盘 <!-- 添加仪表盘图标 -->  
 </a>  
  
 <!-- 普通用户列表按钮（仅管理员可见） -->  
 {% if request.session.is\_admin %}  
 <a href="{% url 'user\_list' %}" class="btn btn-info mb-3">  
 <i class="fas fa-user-friends"></i> 普通用户列表 <!-- 添加普通用户列表图标 -->  
 </a>  
 {% else %}  
 <button class="btn btn-secondary mb-3" disabled>  
 <i class="fas fa-ban"></i> 无权限 <!-- 添加无权限图标 -->  
 </button>  
 {% endif %}  
  
 <!-- 添加学生按钮 -->  
 <a href="{% url 'student\_create' %}" class="btn btn-success mb-3">  
 <i class="fas fa-user-plus"></i> 添加学生 <!-- 添加添加学生图标 -->  
 </a>  
  
 <!-- 学生列表表格 -->  
 <table class="table table-bordered table-striped shadow">  
 <thead class="table-dark">  
 <tr>  
 <th><i class="fas fa-id-card"></i> 学生编号</th> <!-- 添加学生编号图标 -->  
 <th><i class="fas fa-user"></i> 学生姓名</th> <!-- 添加学生姓名图标 -->  
 <th><i class="fas fa-school"></i> 学生班级</th> <!-- 添加学生班级图标 -->  
 <th><i class="fas fa-info-circle"></i> 学生简介</th> <!-- 添加学生简介图标 -->  
 <th><i class="fas fa-cogs"></i> 操作</th> <!-- 添加操作图标 -->  
 </tr>  
 </thead>  
 <tbody>  
 {% for student in page\_obj %}  
 <tr>  
 <td>{{ student.stuId }}</td>  
 <td>{{ student.stuName }}</td>  
 <td>{{ student.stuClass }}</td>  
 <td>{{ student.stuIntro }}</td>  
 <td>  
 <!-- 修改按钮 -->  
 <a href="{% url 'student\_update' student.stuId %}" class="btn btn-sm btn-warning">  
 <i class="fas fa-edit"></i> 修改 <!-- 添加修改图标 -->  
 </a>  
 <!-- 删除按钮 -->  
 <a href="{% url 'student\_delete' student.stuId %}" class="btn btn-sm btn-danger">  
 <i class="fas fa-trash"></i> 删除 <!-- 添加删除图标 -->  
 </a>  
 </td>  
 </tr>  
 {% endfor %}  
 </tbody>  
 </table>  
  
 <!-- 分页导航 -->  
 <nav aria-label="Page navigation">  
 <ul class="pagination justify-content-center">  
 {% if page\_obj.has\_previous %}  
 <li class="page-item">  
 <a class="page-link" href="?page=1">  
 <i class="fas fa-angle-double-left"></i> 首页 <!-- 添加首页图标 -->  
 </a>  
 </li>  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.previous\_page\_number }}">  
 <i class="fas fa-angle-left"></i> 上一页 <!-- 添加上一页图标 -->  
 </a>  
 </li>  
 {% endif %}  
 <li class="page-item disabled">  
 <span class="page-link">  
 第 {{ page\_obj.number }} 页，共 {{ page\_obj.paginator.num\_pages }} 页。  
 </span>  
 </li>  
 {% if page\_obj.has\_next %}  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.next\_page\_number }}">  
 下一页 <i class="fas fa-angle-right"></i> <!-- 添加下一页图标 -->  
 </a>  
 </li>  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.paginator.num\_pages }}">  
 末页 <i class="fas fa-angle-double-right"></i> <!-- 添加末页图标 -->  
 </a>  
 </li>  
 {% endif %}  
 </ul>  
 </nav>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

**student\_create.html代码如下：**

{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>创建学生</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <div class="title-container fade-in">  
 <h1 class="text-center mb-4"><i class="fas fa-user-plus"></i> 创建学生</h1>  
 </div>  
 <form method="post" class="card p-4 shadow">  
 <div class="mb-3">  
 <label for="stuId" class="form-label">学生编号:</label>  
 <input type="text" name="stuId" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="stuName" class="form-label">学生姓名:</label>  
 <input type="text" name="stuName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="stuClass" class="form-label">学生班级:</label>  
 <input type="text" name="stuClass" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="stuIntro" class="form-label">学生简介:</label>  
 <textarea name="stuIntro" class="form-control" required></textarea>  
 </div>  
 <button type="submit" class="btn btn-primary">创建</button>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

**student\_update.html代码如下：**

1. {% load static %}  
   <!DOCTYPE html>  
   <html lang="zh-CN">  
   <head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1">  
    <title>修改学生</title>  
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
    <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
   </head>  
   <body>  
    <div class="container mt-5">  
    <div class="title-container fade-in">  
    <h1 class="text-center mb-4"><i class="fas fa-user-edit"></i> 修改学生</h1>  
    </div>  
    <form method="post" class="card p-4 shadow">  
    <input type="hidden" name="stuId" value="{{ student.stuId }}">  
    <div class="mb-3">  
    <label for="stuName" class="form-label">学生姓名:</label>  
    <input type="text" name="stuName" class="form-control" value="{{ student.stuName }}" required>  
    </div>  
    <div class="mb-3">  
    <label for="stuClass" class="form-label">学生班级:</label>  
    <input type="text" name="stuClass" class="form-control" value="{{ student.stuClass }}" required>  
    </div>  
    <div class="mb-3">  
    <label for="stuIntro" class="form-label">学生简介:</label>  
    <textarea name="stuIntro" class="form-control" required>{{ student.stuIntro }}</textarea>  
    </div>  
    <button type="submit" class="btn btn-primary">保存</button>  
    </form>  
    </div>  
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
   </body>  
   </html>

**Dashboard代码：**

<!-- templates/dashboard.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>仪表盘</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
 <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">学生信息仪表盘 <i class="fas fa-tachometer-alt"></i></h1>  
  
 <!-- 学生总数 -->  
 <div class="card mb-4 shadow">  
 <div class="card-body">  
 <h5 class="card-title"><i class="fas fa-users"></i> 学生总数</h5>  
 <p class="card-text display-4">{{ total\_students }}</p>  
 </div>  
 </div>  
  
 <!-- 班级分布 -->  
 <div class="card mb-4 shadow">  
 <div class="card-body">  
 <h5 class="card-title"><i class="fas fa-chart-bar"></i> 班级分布</h5>  
 <canvas id="classDistributionChart" width="400" height="200"></canvas>  
 </div>  
 </div>  
  
 <!-- 最近添加的学生 -->  
 <div class="card mb-4 shadow">  
 <div class="card-body">  
 <h5 class="card-title"><i class="fas fa-user-plus"></i> 最近添加的学生</h5>  
 <table class="table table-bordered">  
 <thead>  
 <tr>  
 <th>学生编号</th>  
 <th>学生姓名</th>  
 <th>班级</th>  
 <th>简介</th>  
 </tr>  
 </thead>  
 <tbody>  
 {% for student in recent\_students %}  
 <tr>  
 <td>{{ student.stuId }}</td>  
 <td>{{ student.stuName }}</td>  
 <td>{{ student.stuClass }}</td>  
 <td>{{ student.stuIntro }}</td>  
 </tr>  
 {% endfor %}  
 </tbody>  
 </table>  
 </div>  
 </div>  
 </div>  
  
 <script>  
 // 班级分布图表  
 const *classDistributionCtx* = *document*.getElementById('classDistributionChart').getContext('2d');  
 const *classDistributionChart* = new Chart(*classDistributionCtx*, {  
 type: 'bar', // 柱状图  
 data: {  
 labels: {{ class\_names|safe }}, // 班级名称  
 datasets: [{  
 label: '学生数量',  
 data: {{ class\_counts|safe }}, // 每个班级的学生数量  
 backgroundColor: 'rgba(75, 192, 192, 0.2)',  
 borderColor: 'rgba(75, 192, 192, 1)',  
 borderWidth: 1  
 }]  
 },  
 options: {  
 scales: {  
 y: {  
 beginAtZero: true  
 }  
 }  
 }  
 });  
 </script>  
</body>  
</html>

user\_register

<!-- templates/user\_register.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>普通用户注册</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">普通用户注册</h1>  
 <form method="post" class="card p-4 shadow">  
 {% csrf\_token %}  
 <div class="mb-3">  
 <label for="userName" class="form-label">用户名:</label>  
 <input type="text" name="userName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="userPassword" class="form-label">密码:</label>  
 <input type="password" name="userPassword" class="form-control" required>  
 </div>  
 <button type="submit" class="btn btn-primary">注册</button>  
 <p class="mt-3">已有账号？ <a href="{% url 'user\_login' %}">点击登录</a></p>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

user\_login

<!-- templates/user\_login.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>普通用户登录</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">普通用户登录 <i class="fas fa-sign-in-alt"></i></h1> <!-- 添加登录图标 -->  
 {% if error %}  
 <div class="alert alert-danger">  
 <i class="fas fa-exclamation-circle"></i> {{ error }} <!-- 添加错误提示图标 -->  
 </div>  
 {% endif %}  
 <form method="post" class="card p-4 shadow">  
 {% csrf\_token %}  
 <div class="mb-3">  
 <label for="userName" class="form-label">  
 <i class="fas fa-user"></i> 用户名: <!-- 添加用户名图标 -->  
 </label>  
 <input type="text" name="userName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="userPassword" class="form-label">  
 <i class="fas fa-lock"></i> 密码: <!-- 添加密码图标 -->  
 </label>  
 <input type="password" name="userPassword" class="form-control" required>  
 </div>  
 <button type="submit" class="btn btn-primary">  
 <i class="fas fa-sign-in-alt"></i> 登录 <!-- 添加登录按钮图标 -->  
 </button>  
 <p class="mt-3">  
 <i class="fas fa-user-plus"></i> 没有账号? <a href="{% url 'user\_register' %}">点击注册</a> <!-- 添加注册图标 -->  
 </p>  
 <p class="mt-3">  
 <i class="fas fa-user-shield"></i> 管理员登录? <a href="{% url 'admin\_login' %}">点击这里</a> <!-- 添加管理员登录图标 -->  
 </p>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

user\_list

<!-- templates/user\_list.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>普通用户列表</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">普通用户列表 <i class="fas fa-users"></i></h1> <!-- 添加用户列表图标 -->  
  
 <!-- 添加普通用户按钮 -->  
 <a href="{% url 'user\_create' %}" class="btn btn-success mb-3">  
 <i class="fas fa-user-plus"></i> 添加普通用户 <!-- 添加添加用户图标 -->  
 </a>  
  
 <!-- 用户列表表格 -->  
 <table class="table table-bordered table-striped shadow">  
 <thead class="table-dark">  
 <tr>  
 <th><i class="fas fa-id-card"></i> 用户ID</th> <!-- 添加用户ID图标 -->  
 <th><i class="fas fa-user"></i> 用户名</th> <!-- 添加用户名图标 -->  
 <th><i class="fas fa-cogs"></i> 操作</th> <!-- 添加操作图标 -->  
 </tr>  
 </thead>  
 <tbody>  
 {% for user in page\_obj %}  
 <tr>  
 <td>{{ user.id }}</td>  
 <td>{{ user.userName }}</td>  
 <td>  
 <!-- 修改按钮 -->  
 <a href="{% url 'user\_update' user.id %}" class="btn btn-sm btn-warning">  
 <i class="fas fa-edit"></i> 修改 <!-- 添加修改图标 -->  
 </a>  
 <!-- 删除按钮 -->  
 <a href="{% url 'user\_delete' user.id %}" class="btn btn-sm btn-danger">  
 <i class="fas fa-trash"></i> 删除 <!-- 添加删除图标 -->  
 </a>  
 </td>  
 </tr>  
 {% endfor %}  
 </tbody>  
 </table>  
  
 <!-- 分页导航 -->  
 <nav aria-label="Page navigation">  
 <ul class="pagination justify-content-center">  
 {% if page\_obj.has\_previous %}  
 <li class="page-item">  
 <a class="page-link" href="?page=1">  
 <i class="fas fa-angle-double-left"></i> 首页 <!-- 添加首页图标 -->  
 </a>  
 </li>  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.previous\_page\_number }}">  
 <i class="fas fa-angle-left"></i> 上一页 <!-- 添加上一页图标 -->  
 </a>  
 </li>  
 {% endif %}  
 <li class="page-item disabled">  
 <span class="page-link">  
 第 {{ page\_obj.number }} 页，共 {{ page\_obj.paginator.num\_pages }} 页。  
 </span>  
 </li>  
 {% if page\_obj.has\_next %}  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.next\_page\_number }}">  
 下一页 <i class="fas fa-angle-right"></i> <!-- 添加下一页图标 -->  
 </a>  
 </li>  
 <li class="page-item">  
 <a class="page-link" href="?page={{ page\_obj.paginator.num\_pages }}">  
 末页 <i class="fas fa-angle-double-right"></i> <!-- 添加末页图标 -->  
 </a>  
 </li>  
 {% endif %}  
 </ul>  
 </nav>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

user\_create

<!-- templates/user\_create.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>创建普通用户</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">创建普通用户</h1>  
 <form method="post" class="card p-4 shadow">  
 {% csrf\_token %}  
 <div class="mb-3">  
 <label for="userName" class="form-label">用户名:</label>  
 <input type="text" name="userName" class="form-control" required>  
 </div>  
 <div class="mb-3">  
 <label for="userPassword" class="form-label">密码:</label>  
 <input type="password" name="userPassword" class="form-control" required>  
 </div>  
 <button type="submit" class="btn btn-primary">创建</button>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

user\_update

<!-- templates/user\_update.html -->  
{% load static %}  
<!DOCTYPE html>  
<html lang="zh-CN">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>修改普通用户</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">  
 <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet">  
 <link href="{% static 'css/styles.css' %}" rel="stylesheet">  
</head>  
<body>  
 <div class="container mt-5">  
 <h1 class="text-center mb-4">修改普通用户</h1>  
 <form method="post" class="card p-4 shadow">  
 {% csrf\_token %}  
 <input type="hidden" name="userId" value="{{ user.id }}">  
 <div class="mb-3">  
 <label for="userName" class="form-label">用户名:</label>  
 <input type="text" name="userName" class="form-control" value="{{ user.userName }}" required>  
 </div>  
 <div class="mb-3">  
 <label for="userPassword" class="form-label">密码:</label>  
 <input type="password" name="userPassword" class="form-control" value="{{ user.userPassword }}" required>  
 </div>  
 <button type="submit" class="btn btn-primary">保存</button>  
 </form>  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>  
</body>  
</html>

## Settings.py文件其他配置

**（1）App应用添加**

INSTALLED\_APPS = [  
 'django.contrib.admin',  
 'django.contrib.auth',  
 'django.contrib.contenttypes',  
 'django.contrib.sessions',  
 'django.contrib.messages',  
 'django.contrib.staticfiles',  
 'sms'  
]

**（2）注释掉csrf安全防护攻击**

MIDDLEWARE = [  
 'django.middleware.security.SecurityMiddleware',  
 'django.contrib.sessions.middleware.SessionMiddleware',  
 'django.middleware.common.CommonMiddleware',  
 #'django.middleware.csrf.CsrfViewMiddleware',  
 'django.contrib.auth.middleware.AuthenticationMiddleware',  
 'django.contrib.messages.middleware.MessageMiddleware',  
 'django.middleware.clickjacking.XFrameOptionsMiddleware',  
]

## 数据库迁移

python manage.py makemigrations

python manage.py migrate

## css文件创建(新建static/css/styles.css)

1. /\* 自定义CSS文件 - 增强Bootstrap样式并添加动画 \*/  
     
   /\* 1. 页面整体样式 \*/  
   body {  
    font-family: 'Arial', sans-serif;  
    background-color: #f8f9fa;  
    color: #333;  
    margin: 0;  
    padding: 0;  
   }  
     
   .container {  
    max-width: 800px;  
    margin: 0 auto;  
    padding: 20px;  
   }  
     
   /\* 2. 标题容器样式 \*/  
   .title-container {  
    background: linear-gradient(135deg, #007bff, #00bfff, #007bff);  
    background-size: 200% 200%;  
    padding: 20px;  
    border-radius: 10px;  
    margin-bottom: 30px;  
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);  
    text-align: center;  
    animation: gradientBackground 5s ease infinite;  
   }  
     
   /\* 标题文字样式 \*/  
   .title-container h1 {  
    color: #fff;  
    font-size: 2.5rem;  
    font-weight: bold;  
    margin: 0;  
    animation: textGlow 2s infinite alternate;  
   }  
     
   /\* 文字发光动画 \*/  
   @keyframes textGlow {  
    from {  
    text-shadow: 0 0 10px rgba(255, 255, 255, 0.8), 0 0 20px rgba(255, 255, 255, 0.8);  
    }  
    to {  
    text-shadow: 0 0 20px rgba(255, 255, 255, 1), 0 0 40px rgba(255, 255, 255, 1);  
    }  
   }  
     
   /\* 背景渐变动画 \*/  
   @keyframes gradientBackground {  
    0% {  
    background-position: 0% 50%;  
    }  
    50% {  
    background-position: 100% 50%;  
    }  
    100% {  
    background-position: 0% 50%;  
    }  
   }  
     
   /\* 3. 卡片样式增强 \*/  
   .card {  
    border: none;  
    border-radius: 10px;  
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
    transition: transform 0.3s ease, box-shadow 0.3s ease;  
   }  
     
   .card:hover {  
    transform: translateY(-5px);  
    box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);  
   }  
     
   /\* 4. 按钮样式增强 \*/  
   .btn {  
    transition: background-color 0.3s ease, transform 0.3s ease;  
   }  
     
   .btn:hover {  
    transform: scale(1.05);  
   }  
     
   /\* 5. 表单输入框样式增强 \*/  
   .form-control {  
    border-radius: 5px;  
    border: 1px solid #ddd;  
    padding: 10px;  
    transition: border-color 0.3s ease, box-shadow 0.3s ease;  
   }  
     
   .form-control:focus {  
    border-color: #80bdff;  
    box-shadow: 0 0 5px rgba(128, 189, 255, 0.5);  
   }  
     
   /\* 6. 表格样式增强 \*/  
   .table {  
    border-radius: 10px;  
    overflow: hidden;  
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
   }  
     
   .table thead th {  
    background-color: #343a40;  
    color: #fff;  
    border: none;  
   }  
     
   .table tbody tr {  
    transition: background-color 0.3s ease;  
   }  
     
   .table tbody tr:hover {  
    background-color: #f1f1f1;  
   }  
     
   /\* 7. 分页样式增强 \*/  
   .pagination .page-item .page-link {  
    border: none;  
    color: #333;  
    transition: background-color 0.3s ease, color 0.3s ease;  
   }  
     
   .pagination .page-item.active .page-link {  
    background-color: #007bff;  
    color: #fff;  
   }  
     
   .pagination .page-item .page-link:hover {  
    background-color: #007bff;  
    color: #fff;  
   }  
     
   /\* 8. 动画效果 \*/  
   @keyframes fadeIn {  
    from {  
    opacity: 0;  
    transform: translateY(-20px);  
    }  
    to {  
    opacity: 1;  
    transform: translateY(0);  
    }  
   }  
     
   .fade-in {  
    animation: fadeIn 1s ease-out;  
   }  
     
   /\* 9. 响应式设计 \*/  
   @media (max-width: 768px) {  
    .container {  
    padding: 10px;  
    }  
     
    .card {  
    padding: 15px;  
    }  
     
    .btn {  
    width: 100%;  
    margin-bottom: 10px;  
    }  
   }