Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ярославский государственный технический университет»

Кафедра «информационные системы и технологии»

Отчет по практике защищен

с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_Малышев Д. В.

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023

**МИНИ-БЛОГ НА DJANGO**

Отчет о лабораторной работе по курсу «Web - технологии»

ЯГТУ 09.03.04-001 ЛР

|  |  |
| --- | --- |
|  | Отчет выполнил  студент группы ЦПИ-2  \_\_\_\_\_\_\_\_\_\_\_\_\_Юрьев М.С.  «30» декабря 2023 |

2023

Цель работы: разработать многостраничный веб-сайт с клиентской и серверной частью.

Над проектом работали студенты группы ЦПИ-21 Юрьев М. С. и Ярошевская К. Д.

Описание работы:

Был выбран фреймфорк Django для написания сайта.

Для начала были написаны методы, реализующие следующие функции:

* Создание записей на главной странице сайта
* Удаление записей с главной страницы
* Комментирование отдельных записей

После были написаны html страницы, к которым подключен файл css, а также прописаны пути адресации по сайту. Всего сайт имеет 5 страниц:

1. Главная страница, на которой отображаются все записи

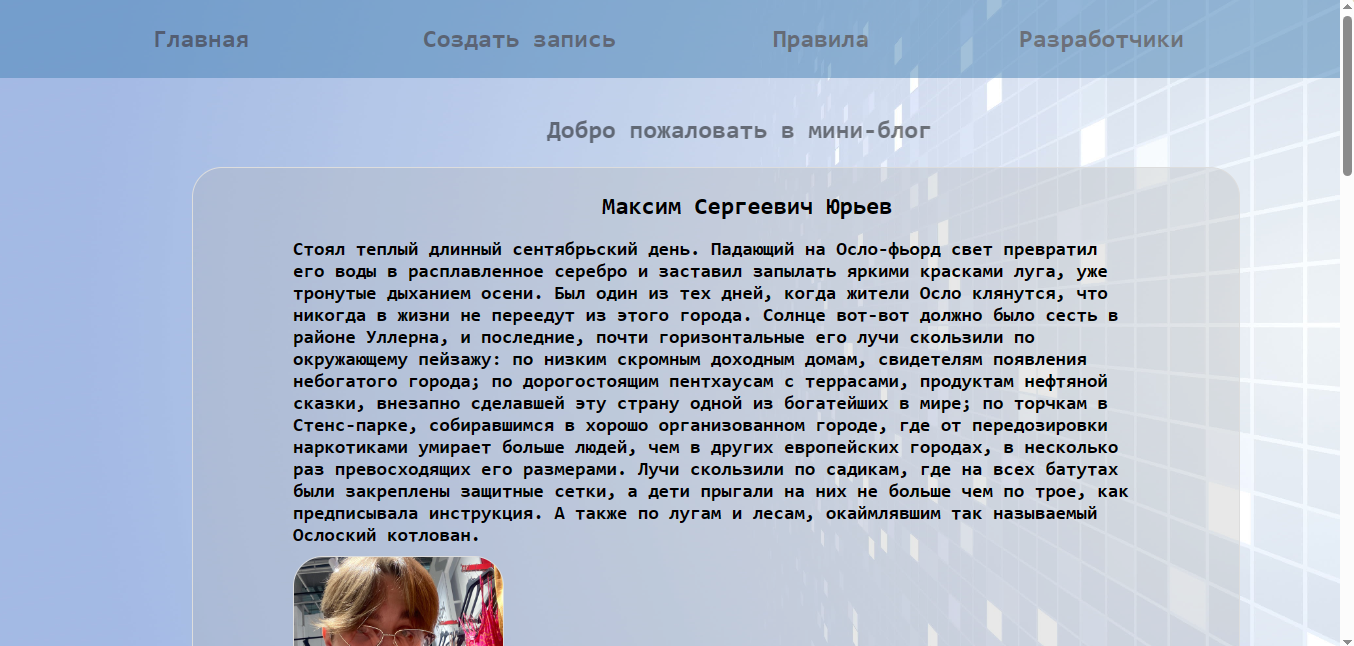


Рис. 1 – главная страница сайта

1. Страница создания новых записей

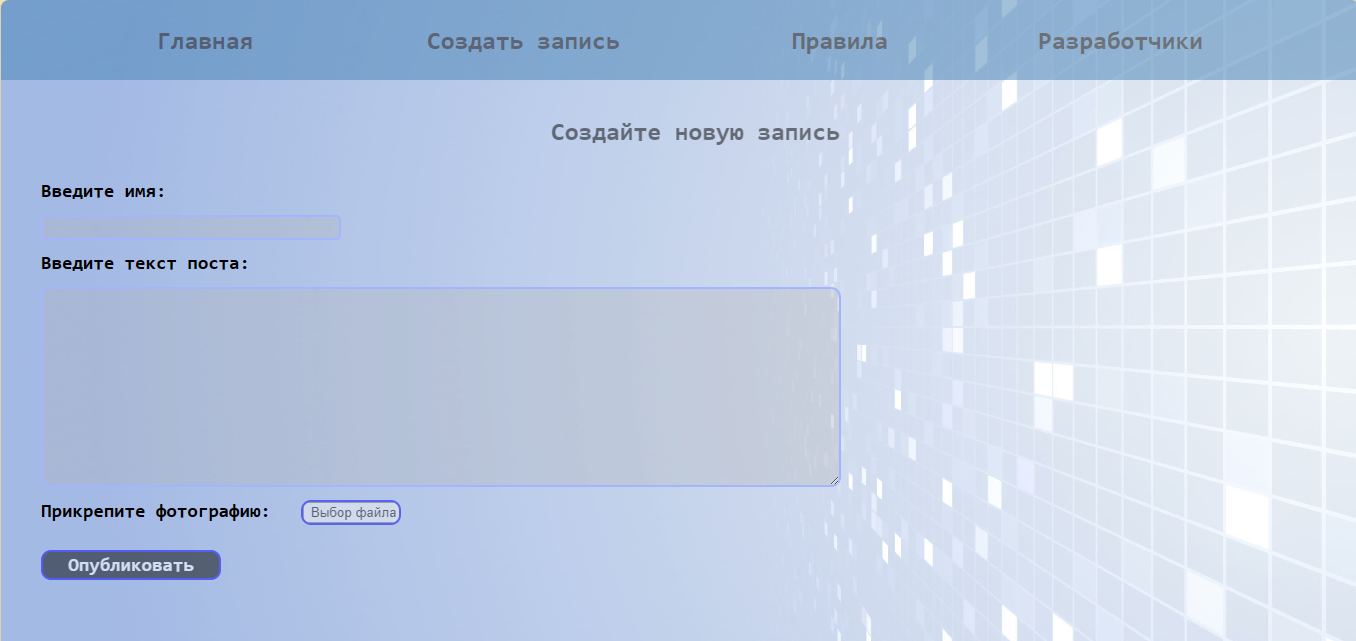


Рис. 2 – страница создания записей

1. Страница с правилами сайта

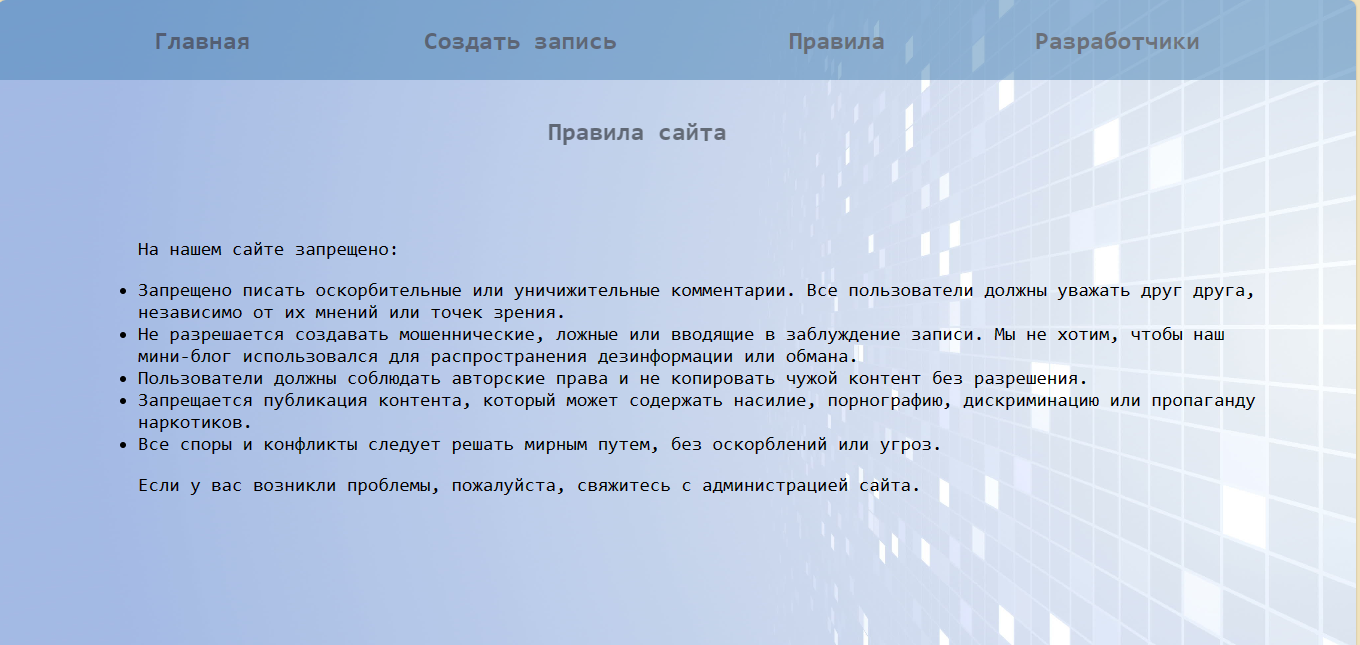


Рис. 3 – страница правил сайта

1. Страница с контактами разработчиков



Рис. 4 – страница с контактами разработчиков

1. Страница “деталей” поста, на которой реализованы комментарии. Переход на данную страницу идет по имени создателя записи.

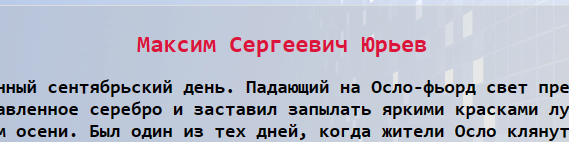
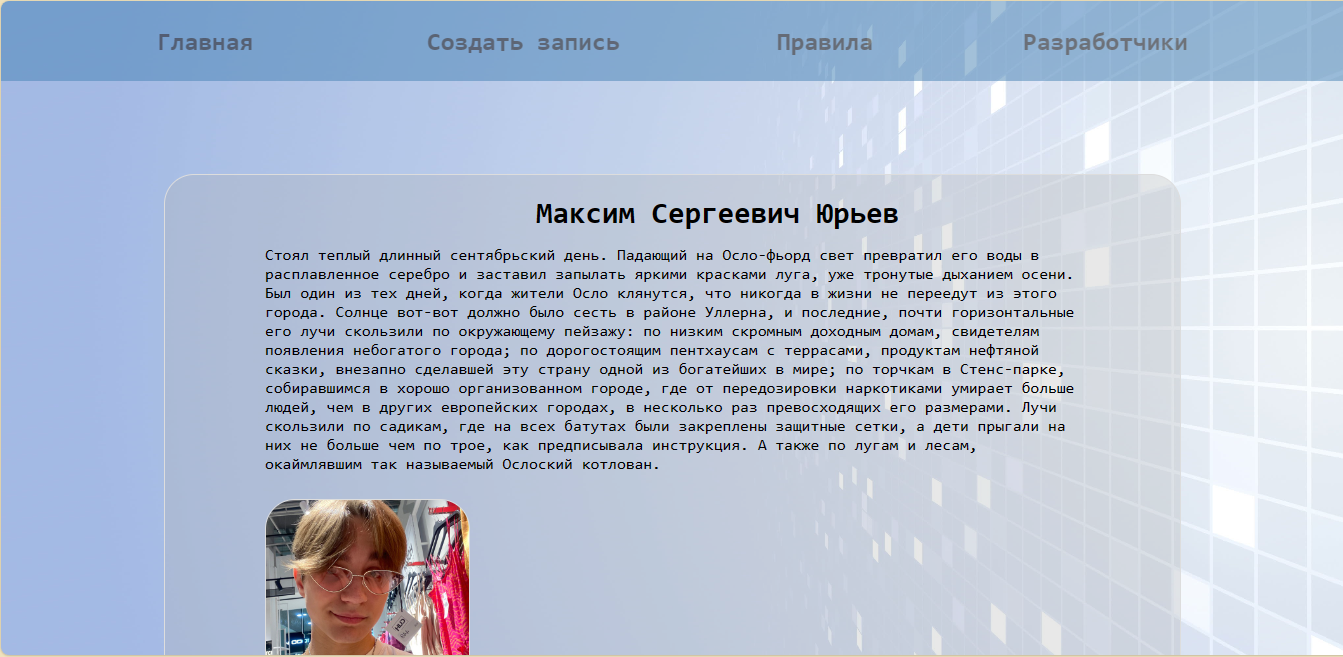
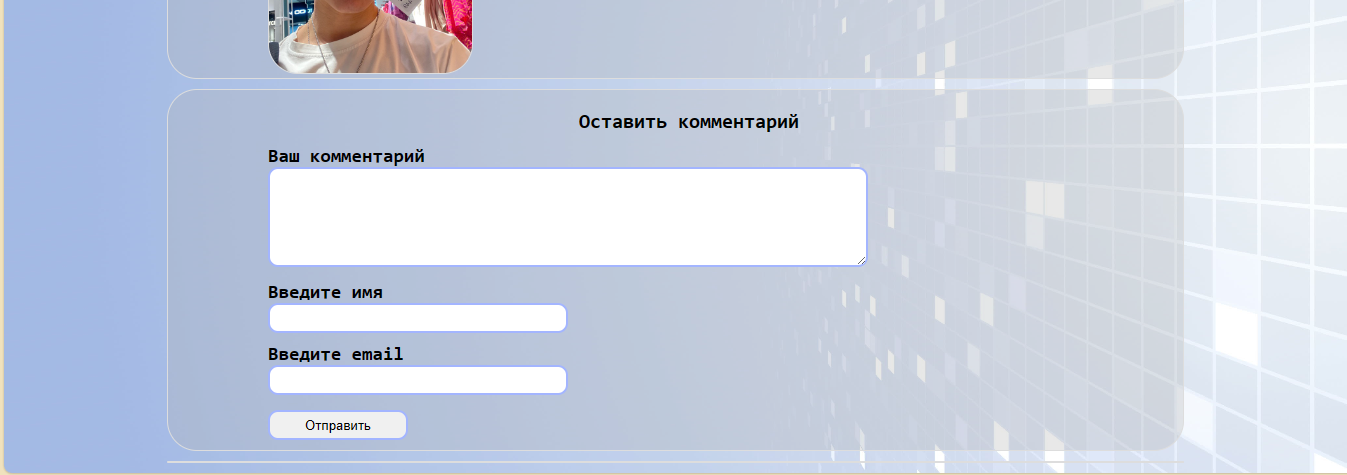


Рис. 5 – Ссылка на страницу с комментариями



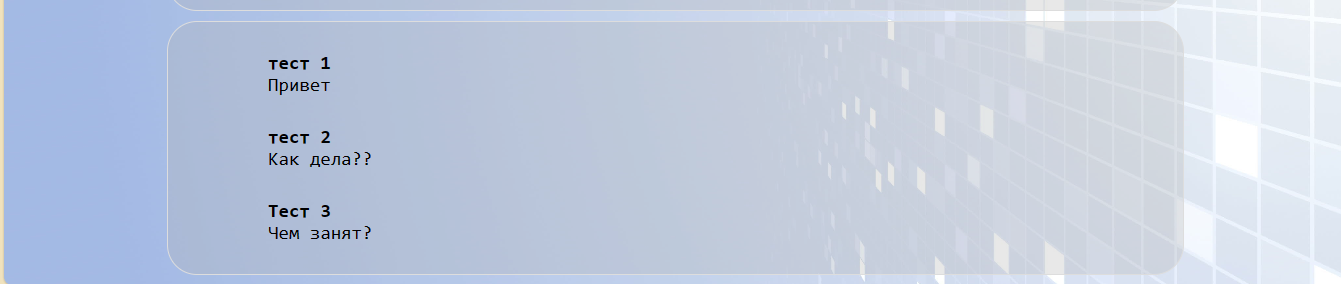


Рис. 6 – страница с комментариями

Листинг кода:

Модель БД для постов и комментариев:

from django.db import models  
class Post(models.Model): //МОДЕЛЬ ПОСТА  
 name = models.CharField(max\_length=20)  
 text\_data = models.CharField(max\_length=300)  
 photo = models.ImageField(upload\_to='photos')  
  
 def \_\_str\_\_(self):  
 return self.name  
  
class Comments(models.Model): //МОДЕЛЬ КОМЕНТАРИЕВ  
 email = models.EmailField()  
 yourname = models.CharField(max\_length=50)  
 text\_comment = models.CharField(max\_length=300)  
 post = models.ForeignKey(Post, verbose\_name='Публикация', on\_delete=models.CASCADE)  
  
 def \_\_str\_\_(self):  
 return f'{self.name},{self.post}'

Реализация методов создания постов, их удаления и комментирования, а также переадресация:

from django.shortcuts import render, HttpResponse, HttpResponseRedirect, redirect  
from django.http import HttpResponseNotFound  
from django.urls import reverse  
from .models import Post  
from .form import CommentsForm

//ОТОБРАЖЕНИЕ ГЛАВНОЙ СТРАНИЦЫ  
def index(request):  
 post = Post.objects.all()  
 return render(request, 'index.html', {'posts': post})

//МЕТОД СОЗДАНИЯ ЗАПИСИ  
def create(request):  
 if request.method == 'POST':  
 post = Post()  
 post.name = request.POST.get('name')  
 post.text\_data = request.POST.get('text\_data')  
 post.photo = request.FILES.get('photo')  
 post.save()  
 return HttpResponseRedirect(reverse('website:index'))  
 return render(request, 'create.html')

//МЕТОД УДАЛЕНИЯ ЗАПИСИ  
def delete(request, id):  
 try:  
 post = Post.objects.get(id = id)  
 post.delete()  
 return HttpResponseRedirect('/website/')  
 except:  
 return HttpResponseNotFound('<h2> Post not found </h2>')

//ПЕРЕАДРЕСАЦИЯ НА СТРАНИЦУ ДЕТАЛЕЙ КОНКРЕТНОГО ПОСТА  
def details(request, id):  
 post = Post.objects.get(id=id)  
 return render(request, 'details.html', {'post': post})

//МЕТОД ДОБАВЛЕНИЯ КОММЕНТАРИЯ  
def comm(request, id):  
 form = CommentsForm(request.POST)  
 if form.is\_valid():  
 form = form.save(commit=False)  
 form.post\_id = id  
 form.save()  
 return redirect(f'/{id}')  
  
//МЕТОД ПЕРЕАДРЕСАЦИИ НА СТРАНИЦУ С ПРАВИЛАМИ  
def rules(request):  
 return render(request, "rules.html", {'rules': rules})

//МЕТОД ПЕРЕАДРЕСАЦИИ НА СТРАНИЦУ С КОНТАКТАМИ  
def info(request):  
 return render(request, "info.html", {'info': info})

Создание url путей:

from django.urls import path  
from . import views  
  
app\_name = 'website'  
  
urlpatterns = [  
 path('', views.index, name='index'),//главная страница  
 path('create/', views.create, name='create'),//страница создания поста  
 path('delete/<int:id>/', views.delete, name='delete'),//метод удаления  
 path('rules/', views.rules, name='rules'),//страница с правилами  
 path('info/', views.info, name='info'),//страница с контактами  
 path('<int:id>/', views.details, name='details'),//страница с деталями  
 path('review/<int:id>/', views.comm, name='comments'),//метод создания комментария  
]

HTML и CSS не вставлены, т. к. они очень объемные. Практичнее будет посмотреть их на GitHub.

Ссылка на репозиторий GitHub:

<https://github.com/yurevms/web_project.git>

Проект находится в ветке master.

Вывод: создан сайт «Мини-блог на Django», который имеет следующие 5 страниц: Главная, Создать запись, Правила, Разработчики и страница с возможностью прокомментировать запись.