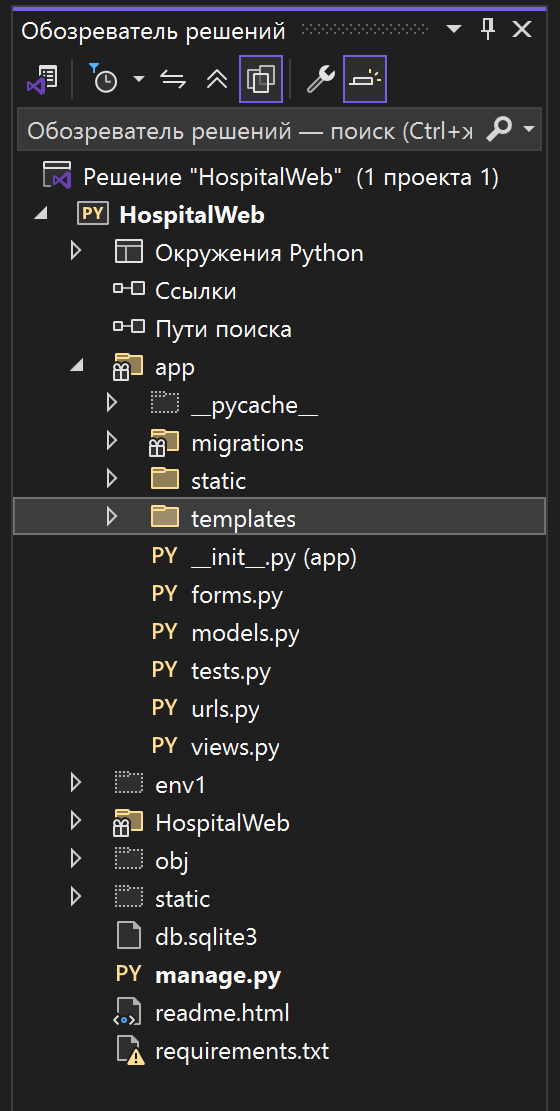
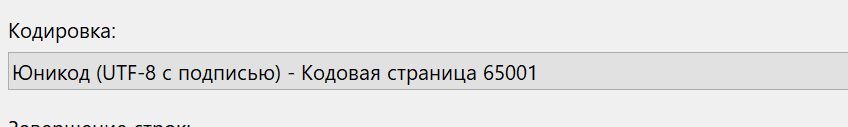
Проект HospitalWeb имеет следующую структуру:



Проект создается в Visual Studio 2017

Создано окружение env1 и активно.

Все файлы в проекте были пересохранены с кодировкой UTF-8 с подписью



Для запуска используется Яндекс Браузер.

Полный код программ:

HospitalWeb/settings.py

"""

Django settings for HospitalWeb project.

Based on 'django-admin startproject' using Django 2.1.2.

For more information on this file, see

https://docs.djangoproject.com/en/2.1/topics/settings/

For the full list of settings and their values, see

https://docs.djangoproject.com/en/2.1/ref/settings/

"""

import os

import posixpath

# Build paths inside the project like this: os.path.join(BASE\_DIR, ...)

BASE\_DIR = os.path.dirname(os.path.dirname(os.path.abspath(\_\_file\_\_)))

# Quick-start development settings - unsuitable for production

# See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!

SECRET\_KEY = '5bfe77d4-41e6-4f8d-9360-ca14c28de553'

# SECURITY WARNING: don't run with debug turned on in production!

DEBUG = True

ALLOWED\_HOSTS = []

# Application references

# https://docs.djangoproject.com/en/2.1/ref/settings/#std:setting-INSTALLED\_APPS

INSTALLED\_APPS = [

'app',

# Add your apps here to enable them

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

]

# Middleware framework

# https://docs.djangoproject.com/en/2.1/topics/http/middleware/

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',

]

ROOT\_URLCONF = 'HospitalWeb.urls'

# Template configuration

# https://docs.djangoproject.com/en/2.1/topics/templates/

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [],

'APP\_DIRS': True,

'OPTIONS': {

'context\_processors': [

'django.template.context\_processors.debug',

'django.template.context\_processors.request',

'django.contrib.auth.context\_processors.auth',

'django.contrib.messages.context\_processors.messages',

],

},

},

]

WSGI\_APPLICATION = 'HospitalWeb.wsgi.application'

# Database

# https://docs.djangoproject.com/en/2.1/ref/settings/#databases

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.sqlite3',

'NAME': os.path.join(BASE\_DIR, 'db.sqlite3'),

}

}

# Password validation

# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH\_PASSWORD\_VALIDATORS = [

{

'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',

},

{

'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',

},

]

# Internationalization

# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE\_CODE = 'ru-ru'

TIME\_ZONE = 'UTC'

USE\_I18N = True

USE\_L10N = True

USE\_TZ = True

# Static files (CSS, JavaScript, Images)

# https://docs.djangoproject.com/en/2.1/howto/static-files/

STATIC\_URL = '/static/'

STATIC\_ROOT = posixpath.join(\*(BASE\_DIR.split(os.path.sep) + ['static']))

LOGIN\_REDIRECT\_URL = '/'

LOGOUT\_REDIRECT\_URL = '/'

DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'

HospitalWeb/urls.py

# HospitalWeb/urls.py

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

path('admin/', admin.site.urls),

path('', include('app.urls')), # корень → ваше приложение

]

HospitalWeb/wsgi.py

"""

WSGI config for HospitalWeb project.

This module contains the WSGI application used by Django's development server

and any production WSGI deployments. It should expose a module-level variable

named ``application``. Django's ``runserver`` and ``runfcgi`` commands discover

this application via the ``WSGI\_APPLICATION`` setting.

Usually you will have the standard Django WSGI application here, but it also

might make sense to replace the whole Django WSGI application with a custom one

that later delegates to the Django one. For example, you could introduce WSGI

middleware here, or combine a Django application with an application of another

framework.

For more information, visit

https://docs.djangoproject.com/en/2.1/howto/deployment/wsgi/

"""

import os

from django.core.wsgi import get\_wsgi\_application

os.environ.setdefault(

'DJANGO\_SETTINGS\_MODULE',

'HospitalWeb.settings')

# This application object is used by any WSGI server configured to use this

# file. This includes Django's development server, if the WSGI\_APPLICATION

# setting points here.

application = get\_wsgi\_application()

app/forms.py

# app/forms.py

from django import forms

from django.contrib.auth.forms import AuthenticationForm

from django.forms import ModelForm

from .models import Blog

class FeedbackForm(forms.Form):

name = forms.CharField(

label="Ваше имя",

min\_length=2,

max\_length=50,

)

email = forms.EmailField(

label="Email",

min\_length=5,

)

rating = forms.ChoiceField(

label="Оцените сайт",

choices=[

('5', 'Отлично'),

('4', 'Хорошо'),

('3', 'Удовлетворительно'),

('2', 'Плохо'),

('1', 'Ужасно'),

],

widget=forms.RadioSelect,

initial='5',

)

favorite\_section = forms.ChoiceField(

label="Любимый раздел",

choices=[

('home', 'Главная'),

('about', 'О нас'),

('services', 'Услуги'),

('contacts', 'Контакты'),

],

)

subscribe = forms.BooleanField(

label="Хотите получать новости?",

required=False,

)

comments = forms.CharField(

label="Ваш комментарий",

widget=forms.Textarea(attrs={'rows': 4, 'cols': 40}),

required=False,

max\_length=500,

)

class LoginForm(AuthenticationForm):

username = forms.CharField(label='Логин', max\_length=254)

password = forms.CharField(label='Пароль', widget=forms.PasswordInput)

class BlogForm(ModelForm):

class Meta:

model = Blog

fields = ['title', 'summary', 'content']

widgets = {

'summary': forms.Textarea(attrs={'rows':3}),

'content': forms.Textarea(attrs={'rows':8}),

}

app/models.py

from django.db import models

from django.contrib import admin

from datetime import datetime

from django.urls import reverse

# импортим User для связи «многие к одной»

from django.contrib.auth.models import User

class Blog(models.Model):

title = models.CharField('Заголовок', max\_length=200)

summary = models.TextField('Краткое содержание', max\_length=500)

content = models.TextField('Полное содержание')

posted = models.DateTimeField('Опубликована', default=datetime.now, db\_index=True)

# Новое поле — связь с моделью User

author = models.ForeignKey(

User,

null=True,

blank=True,

on\_delete=models.SET\_NULL,

verbose\_name="Автор"

)

def \_\_str\_\_(self):

return self.title

def get\_absolute\_url(self):

return reverse('app:blog\_detail', args=[str(self.id)])

class Meta:

verbose\_name = 'Статья блога'

verbose\_name\_plural = 'Статьи блога'

ordering = ['-posted']

# Регистрация в админке — здесь ничего не меняется

admin.site.register(Blog)

app/tests.py

"""

This file demonstrates writing tests using the unittest module. These will pass

when you run "manage.py test".

"""

import django

from django.test import TestCase

# TODO: Configure your database in settings.py and sync before running tests.

class ViewTest(TestCase):

"""Tests for the application views."""

if django.VERSION[:2] >= (1, 7):

# Django 1.7 requires an explicit setup() when running tests in PTVS

@classmethod

def setUpClass(cls):

super(ViewTest, cls).setUpClass()

django.setup()

def test\_home(self):

"""Tests the home page."""

response = self.client.get('/')

self.assertContains(response, 'Home Page', 1, 200)

def test\_contact(self):

"""Tests the contact page."""

response = self.client.get('/contact')

self.assertContains(response, 'Contact', 3, 200)

def test\_about(self):

"""Tests the about page."""

response = self.client.get('/about')

self.assertContains(response, 'About', 3, 200)

app/urls.py

from django.urls import path

from . import views

from django.contrib.auth import views as auth\_views

from .forms import LoginForm

app\_name = 'app'

urlpatterns = [

path('', views.index, name='index'),

path('about/', views.about, name='about'),

path('contact/', views.contact, name='contact'),

path('feedback/', views.pool, name='feedback'),

path('registration/', views.registration, name='registration'),

path('login/', auth\_views.LoginView.as\_view(

template\_name = 'app/login.html',

authentication\_form = LoginForm

),

name='login'),

path('logout/', auth\_views.LogoutView.as\_view(), name='logout'),

# Пути для блога

path('blog/', views.blog\_list, name='blog\_list'),

path('blog/add/', views.blog\_create, name='blog\_create'), # ← новинка

path('blog/<int:pk>/', views.blog\_detail, name='blog\_detail'),

]

app/views.py

from datetime import datetime

from django.shortcuts import render, redirect, get\_object\_or\_404

from django.http import HttpRequest

from django.contrib.auth.forms import UserCreationForm

from django.contrib import messages

from django.contrib.auth.decorators import user\_passes\_test

from .forms import FeedbackForm, BlogForm

from .models import Blog

def index(request):

assert isinstance(request, HttpRequest)

return render(request,'app/index.html',{

'title': 'Главная',

'message': 'Здесь рождается счастье!',

'year': datetime.now().year,

})

def about(request):

assert isinstance(request, HttpRequest)

return render(request,'app/about.html',{

'title': 'О нас',

'message': 'Описание нашего перинатального центра, его миссии и ценностей.',

'year': datetime.now().year,

})

def contact(request):

assert isinstance(request, HttpRequest)

return render(request,'app/contact.html',{

'title': 'Контакты',

'message': 'Если у вас есть вопросы — свяжитесь с нами любым удобным способом.',

'year': datetime.now().year,

})

def pool(request):

assert isinstance(request, HttpRequest)

data = None

if request.method == 'POST':

form = FeedbackForm(request.POST)

if form.is\_valid():

cd = form.cleaned\_data

data = {

'name': cd['name'],

'email': cd['email'],

'rating': dict(form.fields['rating'].choices)[cd['rating']],

'favorite\_section': dict(form.fields['favorite\_section'].choices)[cd['favorite\_section']],

'subscribe': 'Да' if cd['subscribe'] else 'Нет',

'comments': cd['comments'],

}

form = None

else:

form = FeedbackForm()

return render(request,'app/pool.html',{

'title':'Обратная связь',

'year': datetime.now().year,

'form': form,

'data': data,

})

def registration(request):

assert isinstance(request, HttpRequest)

if request.method == 'POST':

form = UserCreationForm(request.POST)

if form.is\_valid():

form.save()

messages.success(request, 'Регистрация прошла успешно! Теперь войдите, пожалуйста.')

return redirect('app:login')

else:

messages.error(request, 'Пожалуйста, исправьте ошибки в форме.')

else:

form = UserCreationForm()

return render(request,'app/registration.html',{

'regform': form,

'year': datetime.now().year,

})

def blog\_list(request):

posts = Blog.objects.all()

return render(request, 'app/blog\_list.html', {

'title': 'Блог',

'posts': posts,

'year': datetime.now().year,

})

@user\_passes\_test(lambda u: u.is\_superuser) # ← только admin

def blog\_create(request):

"""Создание новой статьи блога только для суперпользователя."""

if request.method == 'POST':

form = BlogForm(request.POST)

if form.is\_valid():

post = form.save(commit=False)

post.author = request.user

post.save()

return redirect(post.get\_absolute\_url())

else:

form = BlogForm()

return render(request, 'app/blog\_form.html', {

'title': 'Добавить статью',

'form': form,

'year': datetime.now().year,

})

def blog\_detail(request, pk):

post = get\_object\_or\_404(Blog, pk=pk)

return render(request, 'app/blog\_detail.html', {

'title': post.title,

'post': post,

'year': datetime.now().year,

})

app/templates/app/about.html

{% extends "app/layout.html" %}

{% block content %}

<h2>**{{** title **}}**.</h2>

<h3>**{{** message **}}**</h3>

<p>Use this area to provide additional information.</p>

{% endblock %}

app/templates/app/blog\_detail.html

{% extends 'app/layout.html' %}

{# app/templates/app/blog\_detail.html #}

{% block title %}**{{** title **}}**{% endblock %}

{% block content %}

<article class="mt-4">

<h2>**{{** post.title **}}**</h2>

<p class="text-muted">

Опубликована: **{{** post.posted|date:"d.m.Y H:i" **}}**

{% if post.author %}

&nbsp;—&nbsp;Автор: **{{** post.author.get\_full\_name|default:post.author.username **}}**

{% endif %}

</p>

<div class="mt-3">

**{{** post.content|linebreaks **}}**

</div>

<a href="{% url 'app:blog\_list' %}" class="btn btn-link mt-3">← Назад к списку</a>

</article>

{% endblock %}

app/templates/app/blog\_form.html

{% extends 'app/layout.html' %}

{% block title %}**{{** title **}}**{% endblock %}

{% block content %}

<h2>**{{** title **}}**</h2>

<form method="post" class="mt-3">

{% csrf\_token %}

**{{** form.as\_p **}}**

<button type="submit" class="btn btn-primary">Сохранить</button>

<a href="{% url 'app:blog\_list' %}" class="btn btn-link">Отмена</a>

</form>

{% endblock %}

app/templates/app/blog\_list.html

{% extends 'app/layout.html' %}

{% load static %}

{% block title %}Блог{% endblock %}

{% block content %}

<div class="d-flex justify-content-between align-items-center mb-3">

<h2>Блог</h2>

{% if user.is\_superuser %}

<a href="{% url 'app:blog\_create' %}"

class="btn btn-pink">

Создать статью

</a>

{% endif %}

</div>

<div class="list-group">

{% for post in posts %}

<a class="list-group-item list-group-item-action" href="**{{** post.get\_absolute\_url **}}**">

<h5>**{{** post.title **}}**</h5>

<small class="text-muted">

**{{** post.posted|date:"d.m.Y H:i" **}}**

{% if post.author %}

— автор: **{{** post.author.username **}}**

{% endif %}

</small>

<p>**{{** post.summary **}}**</p>

</a>

{% empty %}

<p>Статей пока нет.</p>

{% endfor %}

</div>

{% endblock %}

app/templates/app/contact.html

{% extends "app/layout.html" %}

{% block content %}

<h2>**{{** title **}}**.</h2>

<h3>**{{** message **}}**</h3>

<address>

One Microsoft Way<br />

Redmond, WA 98052-6399<br />

<abbr title="Phone">P:</abbr>

425.555.0100

</address>

<address>

<strong>Support:</strong> <a href="mailto:Support@example.com">Support@example.com</a><br />

<strong>Marketing:</strong> <a href="mailto:Marketing@example.com">Marketing@example.com</a>

</address>

{% endblock %}

app/templates/app/index.html

{% extends 'app/layout.html' %}

{% load static %}

{% block title %}**{{** title **}}**{% endblock %}

{% block content %}

<div id="mainCarousel" class="carousel slide" data-bs-ride="carousel">

<div class="carousel-indicators">

<button type="button" data-bs-target="#mainCarousel" data-bs-slide-to="0" class="active"></button>

<button type="button" data-bs-target="#mainCarousel" data-bs-slide-to="1"></button>

<button type="button" data-bs-target="#mainCarousel" data-bs-slide-to="2"></button>

</div>

<div class="carousel-inner">

<div class="carousel-item active">

<img src="{% static 'app/content/centr1.png' %}" class="d-block w-100" alt="Фон 1">

</div>

<div class="carousel-item">

<img src="{% static 'app/content/centr2.png' %}" class="d-block w-100" alt="Фон 2">

</div>

<div class="carousel-item">

<img src="{% static 'app/content/woman.png' %}" class="d-block w-100" alt="Фон 3">

</div>

</div>

<button class="carousel-control-prev" type="button" data-bs-target="#mainCarousel" data-bs-slide="prev">

<span class="carousel-control-prev-icon"></span>

</button>

<button class="carousel-control-next" type="button" data-bs-target="#mainCarousel" data-bs-slide="next">

<span class="carousel-control-next-icon"></span>

</button>

<div class="carousel-caption d-flex justify-content-center align-items-center">

<div class="promo-text text-center">

<h1>Здесь рождается счастье!</h1>

<a href="{% url 'app:about' %}" class="btn btn-pink btn-lg my-3">

Узнать больше

</a>

<p>Мы — семейно-ориентированный роддом. Если вам нужна поддержка родных и близких, то мы только «За»!</p>

</div>

</div>

</div>

<section class="services container mt-5">

<div class="row">

<div class="col-md-4 text-center mb-4">

<h3>Комплексная медицинская помощь</h3>

<p>Планирование и ведение беременности, роды, выхаживание и наблюдение за новорождёнными детьми.</p>

<a href="{% url 'app:blog\_list' %}" class="btn btn-outline-primary">Подробнее</a>

</div>

<div class="col-md-4 text-center mb-4">

<h3>Консультативно-диагностический центр</h3>

<p>В КДЦ женщины могут пройти необходимые обследования и получить консультации профильных специалистов.</p>

<a href="{% url 'app:blog\_list' %}" class="btn btn-outline-primary">Подробнее</a>

</div>

<div class="col-md-4 text-center mb-4">

<h3>Экстренная помощь беременным и детям</h3>

<p>На базе центра организованы отделения реанимации и интенсивной терапии взрослых, новорождённых и грудных детей.</p>

<a href="{% url 'app:blog\_list' %}" class="btn btn-outline-primary">Подробнее</a>

</div>

</div>

</section>

{% endblock %}

app/templates/app/layout.html

{% load static %}

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="utf-8">

<title>{% block title %}**{{** title **}}**{% endblock %}</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<!-- Bootstrap CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"

rel="stylesheet">

<!-- Ваши стили -->

<link rel="stylesheet" href="{% static 'app/content/site.css' %}">

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light fixed-top">

<div class="container">

<!-- Логотип -->

<a class="navbar-brand" href="{% url 'app:index' %}">

<img src="{% static 'app/content/logo.png' %}" height="40" alt="Логотип">

</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#nav">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="nav">

<ul class="navbar-nav me-auto">

<li class="nav-item">

<a class="nav-link {% if request.resolver\_match.url\_name == 'index' %}active{% endif %}"

href="{% url 'app:index' %}">Главная</a>

</li>

<li class="nav-item">

<a class="nav-link {% if request.resolver\_match.url\_name == 'about' %}active{% endif %}"

href="{% url 'app:about' %}">О нас</a>

</li>

<li class="nav-item">

<a class="nav-link

{% if request.resolver\_match.url\_name == 'blog\_list' or request.resolver\_match.url\_name == 'blog\_detail' %}

active

{% endif %}"

href="{% url 'app:blog\_list' %}">Блог</a>

</li>

<li class="nav-item">

<a class="nav-link {% if request.resolver\_match.url\_name == 'contact' %}active{% endif %}"

href="{% url 'app:contact' %}">Контакты</a>

</li>

<li class="nav-item">

<a class="nav-link {% if request.resolver\_match.url\_name == 'feedback' %}active{% endif %}"

href="{% url 'app:feedback' %}">Обратная связь</a>

</li>

</ul>

{% if user.is\_authenticated %}

<span class="navbar-text me-3">

Добро пожаловать, **{{** user.username **}}**!

</span>

<form method="post" action="{% url 'app:logout' %}" style="display:inline;">

{% csrf\_token %}

<button type="submit" class="btn btn-link nav-link px-2">Выйти</button>

</form>

{% else %}

<ul class="navbar-nav">

<li class="nav-item"><a class="nav-link" href="{% url 'app:login' %}">Вход</a></li>

<li class="nav-item"><a class="nav-link" href="{% url 'app:registration' %}">Регистрация</a></li>

</ul>

{% endif %}

</div>

</div>

</nav>

<div class="container-fluid" style="padding-top:80px;">

{% if messages %}

{% for message in messages %}

<div class="alert alert-**{{** message.tags **}}** mt-3">**{{** message **}}**</div>

{% endfor %}

{% endif %}

{% block content %}{% endblock %}

</div>

<!-- Футер -->

<footer class="bg-light text-center text-muted py-3 mt-5">

<div class="container">

<hr>

<p class="mb-1">© **{{** year **}}** — Перинатальный центр им. В. В. Панкратова</p>

<p class="small mb-0">Все права защищены.</p>

</div>

</footer>

<!-- Bootstrap JS -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

app/templates/app/login.html

{% extends 'app/layout.html' %}

{% block title %}Вход{% endblock %}

{% block content %}

<h2>Вход на сайт</h2>

<form method="post" class="mt-3">

{% csrf\_token %}

<div class="mb-3">

**{{** form.username.label\_tag **}}**<br>**{{** form.username **}}**

{% for err in form.username.errors %}<div class="text-danger">**{{** err **}}**</div>{% endfor %}

</div>

<div class="mb-3">

**{{** form.password.label\_tag **}}**<br>**{{** form.password **}}**

{% for err in form.password.errors %}<div class="text-danger">**{{** err **}}**</div>{% endfor %}

</div>

<button type="submit" class="btn btn-primary">Войти</button>

</form>

{% endblock %}

app/templates/app/loginpartial.html

{% if user.is\_authenticated %}

<form id="logoutForm" action="/logout/" method="post" class="navbar-right">

{% csrf\_token %}

<ul class="nav navbar-nav navbar-right">

<li><span class="navbar-brand">Hello **{{** user.username **}}**!</span></li>

<li><a href="javascript:document.getElementById('logoutForm').submit()">Log off</a></li>

</ul>

</form>

{% else %}

<ul class="nav navbar-nav navbar-right">

<li><a href="{% url 'login' %}">Log in</a></li>

</ul>

{% endif %}

app/templates/app/pool.html

{% extends 'app/layout.html' %}

{% block title %}**{{** title **}}**{% endblock %}

{% block content %}

<h2>**{{** title **}}**</h2>

{% if form %}

<form method="post" class="mt-4">

{% csrf\_token %}

<div class="mb-3">

**{{** form.name.label\_tag **}}**<br>

**{{** form.name **}}**

{% for err in form.name.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="mb-3">

**{{** form.email.label\_tag **}}**<br>

**{{** form.email **}}**

{% for err in form.email.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="mb-3">

**{{** form.rating.label\_tag **}}**<br>

**{{** form.rating **}}**

{% for err in form.rating.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="mb-3">

**{{** form.favorite\_section.label\_tag **}}**<br>

**{{** form.favorite\_section **}}**

{% for err in form.favorite\_section.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="form-check mb-3">

**{{** form.subscribe **}}** **{{** form.subscribe.label\_tag **}}**

{% for err in form.subscribe.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="mb-3">

**{{** form.comments.label\_tag **}}**<br>

**{{** form.comments **}}**

{% for err in form.comments.errors %}

<div class="text-danger">**{{** err **}}**</div>

{% endfor %}

</div>

<div class="form-group">

<button type="submit" class="btn btn-primary">Отправить отзыв</button>

</div>

</form>

{% else %}

<div class="alert alert-success mt-4">

<h4>Спасибо за ваш отзыв!</h4>

<ul class="list-unstyled">

<li><strong>Имя:</strong> **{{** data.name **}}**</li>

<li><strong>Email:</strong> **{{** data.email **}}**</li>

<li><strong>Оценка сайта:</strong> **{{** data.rating **}}**</li>

<li><strong>Любимый раздел:</strong> **{{** data.favorite\_section **}}**</li>

<li><strong>Подписка на новости:</strong> **{{** data.subscribe **}}**</li>

<li><strong>Комментарий:</strong> **{{** data.comments|linebreaks **}}**</li>

</ul>

</div>

{% endif %}

{% endblock %}

app/templates/app/registration.html

{% extends 'app/layout.html' %}

{% block title %}Регистрация{% endblock %}

{% block content %}

<h2>Регистрация пользователя</h2>

<form method="post" class="mt-3">

{% csrf\_token %}

**{{** regform.as\_p **}}**

<button type="submit" class="btn btn-primary">Зарегистрировать</button>

</form>

{% endblock %}

app/static/app/content/site.css

/\* Активная вкладка \*/

.nav-link.active {

background-color: #f5ccec;

border-radius: .25rem;

}

/\* Анимация по ховеру \*/

.nav-link {

transition: background-color 0.3s ease;

}

.nav-link:hover {

background-color: #f5ccec;

border-radius: .25rem;

}

/\* Промо-блок поверх карусели \*/

.carousel-caption {

top: 0;

bottom: 0;

padding: 0;

}

.promo-text {

background: rgba(128, 128, 128, 0.5); /\* Полупрозрачный серый фон \*/

border-radius: 20px; /\* Скруглённые углы \*/

padding: 2rem; /\* Внутренние отступы \*/

max-width: 800px; /\* Максимальная ширина \*/

margin: auto; /\* Центрирование по горизонтали \*/

}

.promo-text h1 {

color: #fff;

font-size: 3rem;

margin-bottom: 1rem;

}

.promo-text p {

color: #fff;

font-size: 1.25rem;

}

/\* Розовые кнопки \*/

.btn-pink {

background-color: #e91e63;

border-color: #e91e63;

color: #fff;

}

.btn-pink:hover,

.btn-pink:focus {

background-color: #d81b60;

border-color: #d81b60;

color: #fff;

}

/\* Карусель — заголовок/кнопка/текст по центру \*/

.carousel-caption {

top: 50%;

transform: translateY(-50%);

}

.carousel-caption h1 {

font-size: 3rem;

color: white;

text-shadow: 0 0 10px rgba(0,0,0,0.5);

}

.carousel-caption p {

font-size: 1.25rem;

color: white;

text-shadow: 0 0 8px rgba(0,0,0,0.5);

}

.carousel-caption .btn {

font-size: 1.1rem;

}

/\* Раздел услуг — убрали иконки и добавили кнопки \*/

.service-card {

text-align: center;

margin-bottom: 2rem;

}

.service-card img {

display: none;

}

.service-card .btn-details {

margin-top: 1rem;

}