**Московский государственный технический университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра «Системы обработки информации и управления»

**Отчёт по Лабораторной работе №8**

**“Создание простого веб приложения на Django”**

Отчёт

(вид документа)

Листы А4

(вид носителя)

10

(количество листов)

|  |  |
| --- | --- |
| Исполнитель: | Студент группы ИУ5-54Б  Савельев Алексей Александрович |

Москва - 2020

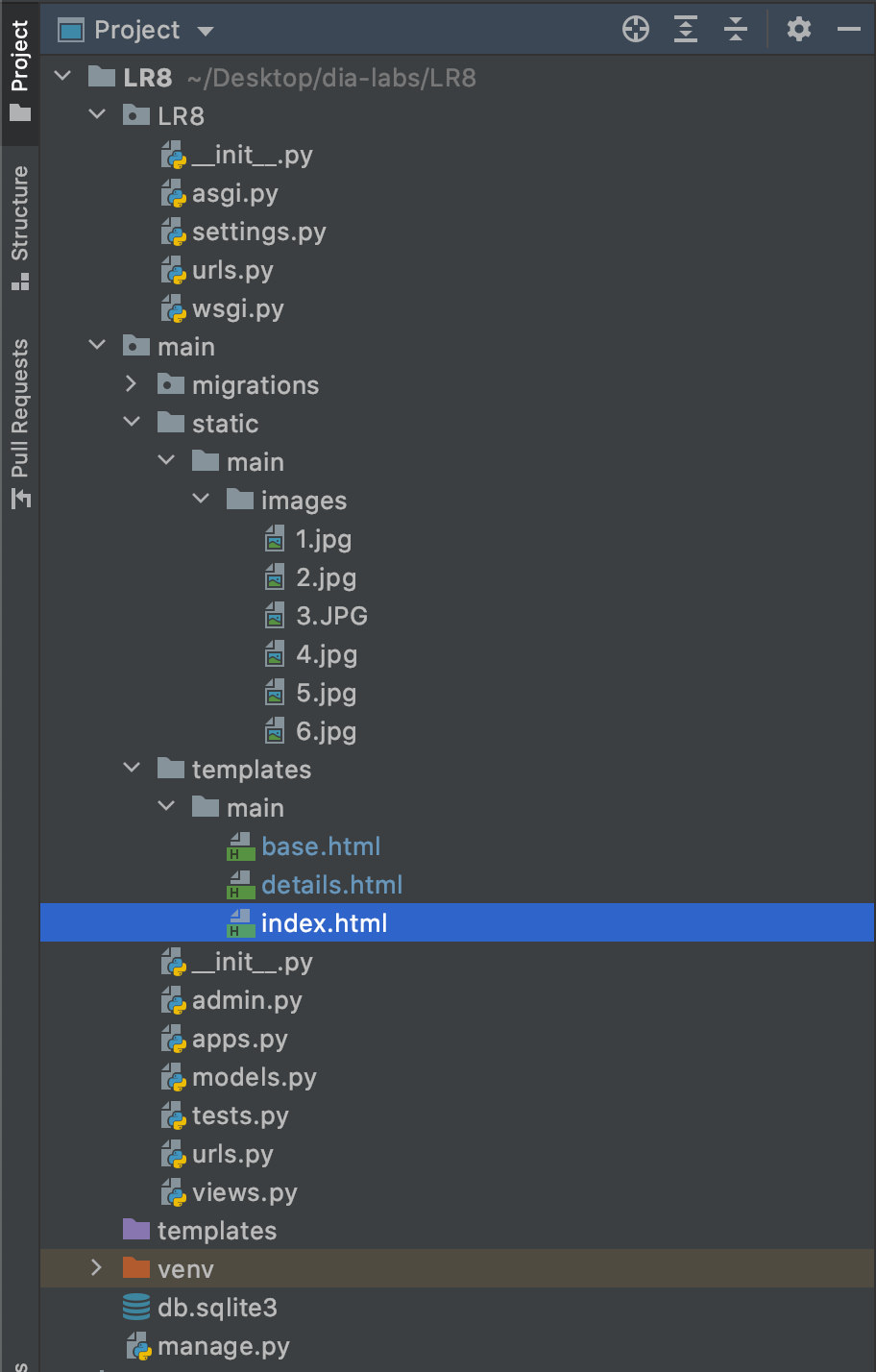
**Цель работы**: изучение возможностей создания прототипа веб-приложения с использованием фреймворка Django.

Задание:

1. Создайте прототип веб-приложения с использованием фреймворка Django:
   * Создайте виртуальное окружение.
   * Установите в него Django.
   * Создайте проект и приложение Django.
2. Создайте представления и шаблоны (по желанию можно использовать модели), реализующие концепцию master/detail со следующей функциональностью:
   * На странице master в виде списка HTML выводится информация о трех объектах (например, о трех сортах мороженого). Каждая строка списка представляет собой гиперссылку, при нажатии на которую происходит переход к странице detail.
   * Страница detail содержит детальное описание объекта (сорта мороженого), фотографию, гиперссылку на master-страницу.
   * Фотография относится к статическому содержимому сайта.
   * Страница detail должна выводить данные с использованием таблицы HTML.
   * Шаблон страницы detail получает от представления данные о детальном объекте с использованием контекста.
   * **НЕОБЯЗАТЕЛЬНЫЙ ПУНКТ.** По желанию можно использовать верстку с применением Bootstrap (или аналогичного фреймворка), а также представления на основе классов (class-based views).

## Листинг программы

Файловая структура проекта



* settings.py

from pathlib import Path  
  
# Build paths inside the project like this: BASE\_DIR / 'subdir'.  
BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent  
  
  
# Quick-start development settings - unsuitable for production  
# See https://docs.djangoproject.com/en/3.1/howto/deployment/checklist/  
  
# SECURITY WARNING: keep the secret key used in production secret!  
SECRET\_KEY = '=)0+p\_@-%yh%9qf2\*j0eyja9-0qwxxtt#jjyuv6#7ve^q5\_$rz'  
  
# SECURITY WARNING: don't run with debug turned on in production!  
DEBUG = True  
  
ALLOWED\_HOSTS = []  
  
  
# Application definition  
  
INSTALLED\_APPS = [  
 'django.contrib.admin',  
 'django.contrib.auth',  
 'django.contrib.contenttypes',  
 'django.contrib.sessions',  
 'django.contrib.messages',  
 'django.contrib.staticfiles',  
 'main',  
]  
  
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 = 'LR8.urls'  
  
TEMPLATES = [  
 {  
 'BACKEND': 'django.template.backends.django.DjangoTemplates',  
 'DIRS': [BASE\_DIR / 'templates']  
 ,  
 '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 = 'LR8.wsgi.application'  
  
  
# Database  
# https://docs.djangoproject.com/en/3.1/ref/settings/#databases  
  
DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.sqlite3',  
 'NAME': BASE\_DIR / 'db.sqlite3',  
 }  
}  
  
  
# Password validation  
# https://docs.djangoproject.com/en/3.1/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/3.1/topics/i18n/  
  
LANGUAGE\_CODE = 'ru'  
  
TIME\_ZONE = 'UTC'  
  
USE\_I18N = True  
  
USE\_L10N = True  
  
USE\_TZ = True  
  
  
# Static files (CSS, JavaScript, Images)  
# https://docs.djangoproject.com/en/3.1/howto/static-files/  
  
STATIC\_URL = '/static/'

* urls.py

from django.contrib import admin  
from django.urls import path, include  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', include('main.urls'))  
]

* main/admin.py

from django.contrib import admin  
from .models import Phenomenon, Description  
  
  
admin.site.register(Phenomenon)  
admin.site.register(Description)

* main/models.py

from django.db import models  
  
class Phenomenon(models.Model):  
 name = models.CharField(max\_length=30)  
  
 def \_\_str\_\_(self):  
 return self.name  
  
class Description(models.Model):  
 phen\_id = models.ForeignKey(Phenomenon, on\_delete=models.CASCADE)  
 description = models.CharField(max\_length=1500)

* main/urls.py

from django.urls import path  
from . import views  
  
  
urlpatterns = [  
 path('', views.index, name='home'),  
 path('<int:phen\_id>/', views.details, name='details'),  
]

* main/views.py

from django.shortcuts import get\_object\_or\_404, render  
  
from .models import Phenomenon  
from .models import Description  
  
def index(request):  
 phen\_list = Phenomenon.objects.all()  
 description = []  
 for i in phen\_list:  
 description.append(get\_object\_or\_404(Description, pk=i.id))  
 context = {'title': 'Природные явления','phen\_list': phen\_list, 'disc':description}  
 return render(request, 'main/index.html', context)  
  
  
def details(request, phen\_id):  
 phen\_list = Phenomenon.objects.all()  
 for i in phen\_list:  
 if i.id == phen\_id:  
 phen = i  
 description = get\_object\_or\_404(Description, pk=phen\_id)  
 return render(request, 'main/details.html', {'title': 'Детали', 'pnen': phen, 'description': description})

* main/templates/main/base.html

<!doctype html>  
<html lang=ru>  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport"  
 content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>{% block title %} {% endblock %}</title>  
 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">  
 <style>  
 {% block styles %} {% endblock %}  
 </style>  
</head>  
<body>  
  
 <div class="d-flex flex-column flex-md-row align-items-center p-3 px-md-4 mb-3 bg-white border-bottom shadow-sm">  
 <p class="h5 my-0 me-md-auto fw-normal">Лабораторная работа №8</p>  
 <nav class="my-2 my-md-0 me-md-3">  
 <a class="p-2 text-dark" href="{% url 'home' %}">Гланая</a>  
 </nav>  
 </div>  
 <main class="px-3">  
 <div class="container">  
 {% block content %} {% endblock %}  
 </div>  
 </main>  
</body>  
</html>

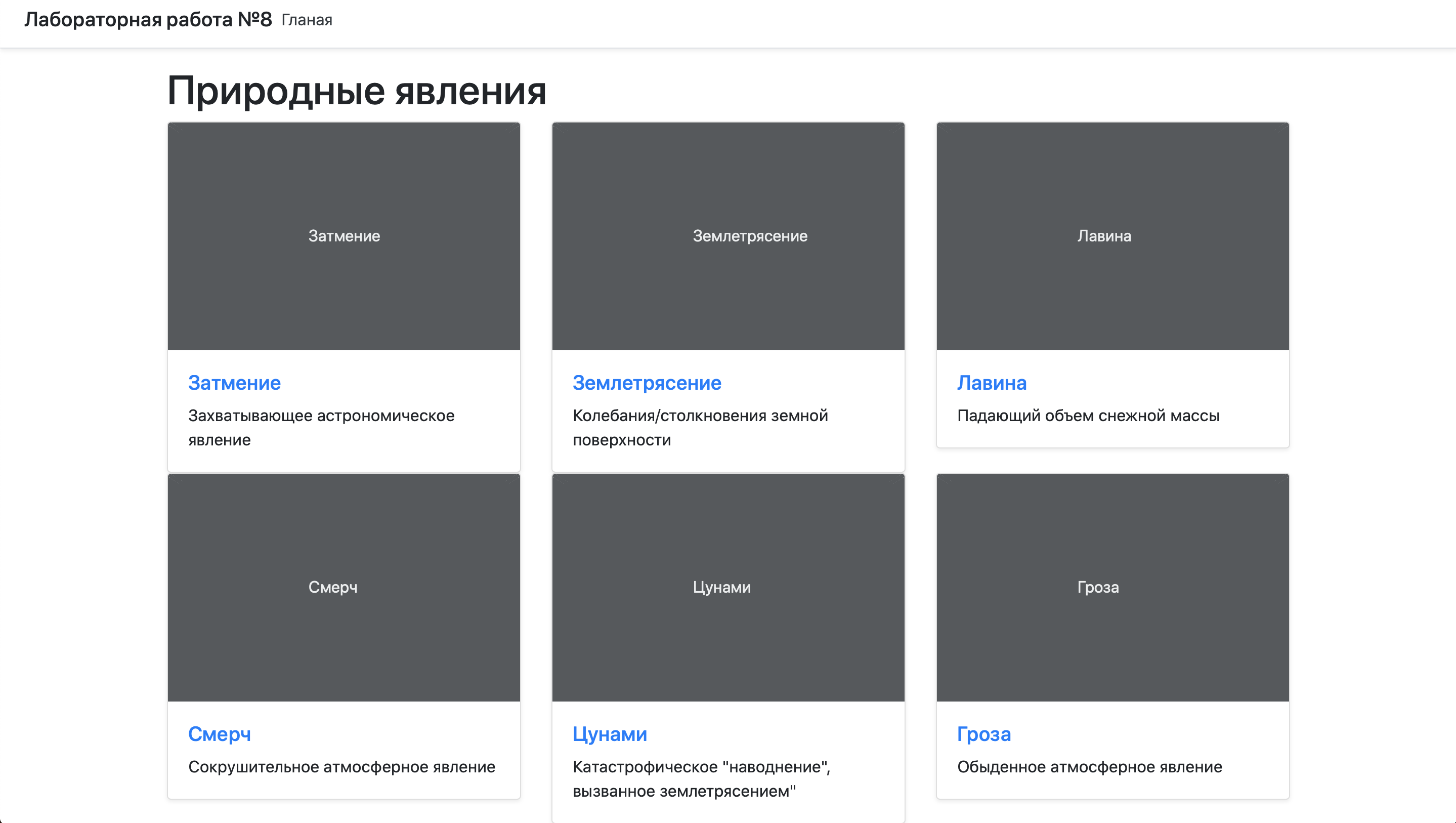
* main/templates/main/index.html

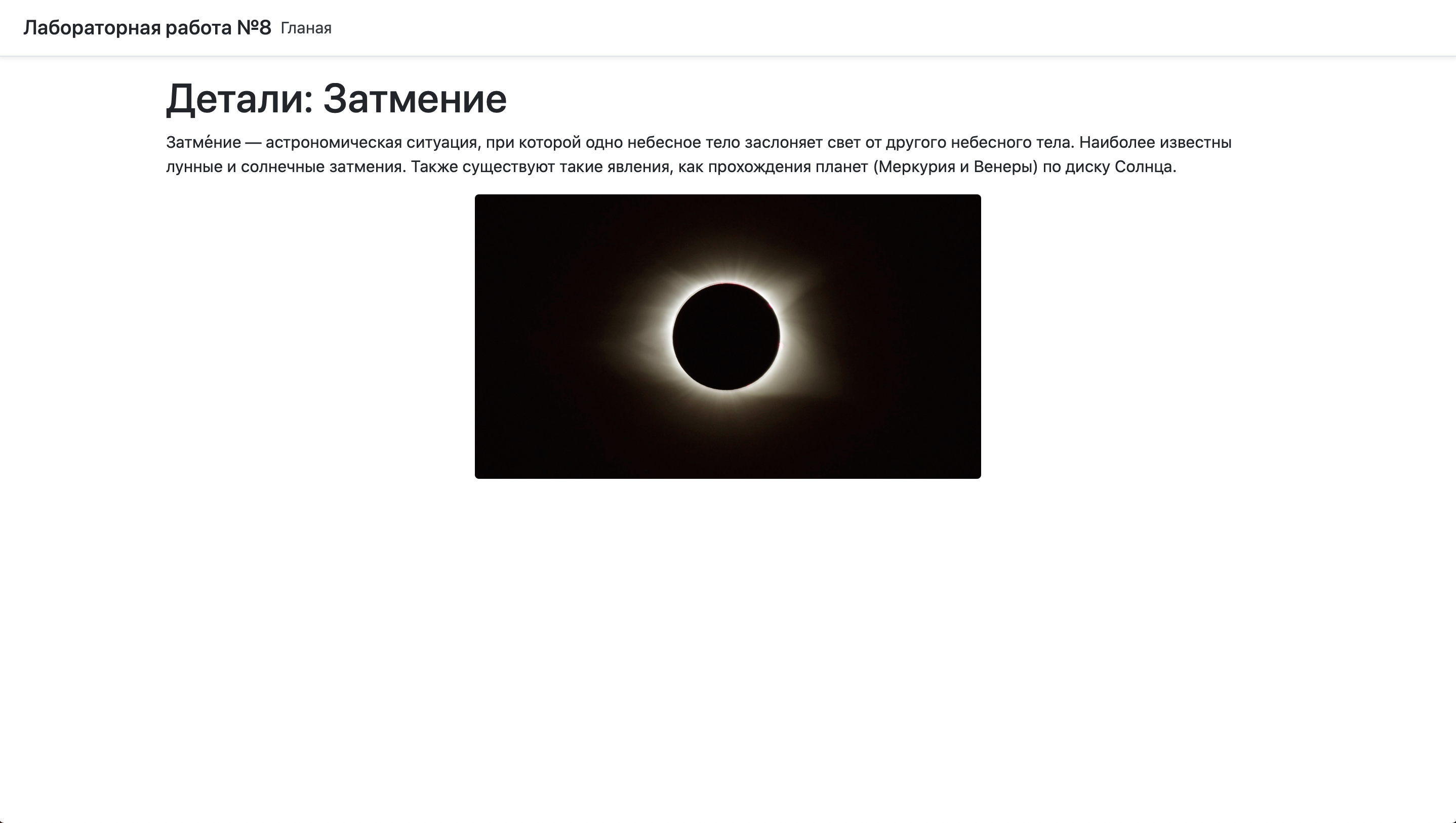
{% extends 'main/base.html'%}  
  
{% block title %}  
 {{ title }}  
{% endblock %}  
{% block content %}  
 <h1>{{ title }}</h1>  
 <div class="row row-cols-1 row-cols-sm-2 row-cols-md-3 g-3">  
 {% for phen in phen\_list %}  
  
 <div class="col">  
 <div class="card shadow-sm">  
 <svg class="bd-placeholder-img card-img-top" width="100%" height="225" xmlns="http://www.w3.org/2000/svg" role="img" aria-label="Placeholder: Thumbnail" preserveAspectRatio="xMidYMid slice" focusable="false"><title>Placeholder</title><rect width="100%" height="100%" fill="#55595c"></rect><text x="40%" y="50%" fill="#eceeef" dy=".3em">  
 {{ phen.name }}</text></svg>  
  
 <div class="card-body">  
 <a href="{% url 'details' phen.id %}"><h5>{{ phen.name }}</h5></a>  
 {% if phen.id == 1 %}  
 <p class="card-text">Захватывающее астрономическое явление</p>  
 {% endif %}  
 {% if phen.id == 2 %}  
 <p class="card-text">Колебания/столкновения земной поверхности</p>  
 {% endif %}  
 {% if phen.id == 3 %}  
 <p class="card-text">Падающий объем снежной массы</p>  
 {% endif %}  
 {% if phen.id == 4 %}  
 <p class="card-text">Сокрушительное атмосферное явление</p>  
 {% endif %}  
 {% if phen.id == 5 %}  
 <p class="card-text">Катастрофическое "наводнение", вызванное землетрясением"</p>  
 {% endif %}  
 {% if phen.id == 6 %}  
 <p class="card-text">Обыденное атмосферное явление</p>  
 {% endif %}  
  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
  
  
{% endblock %}

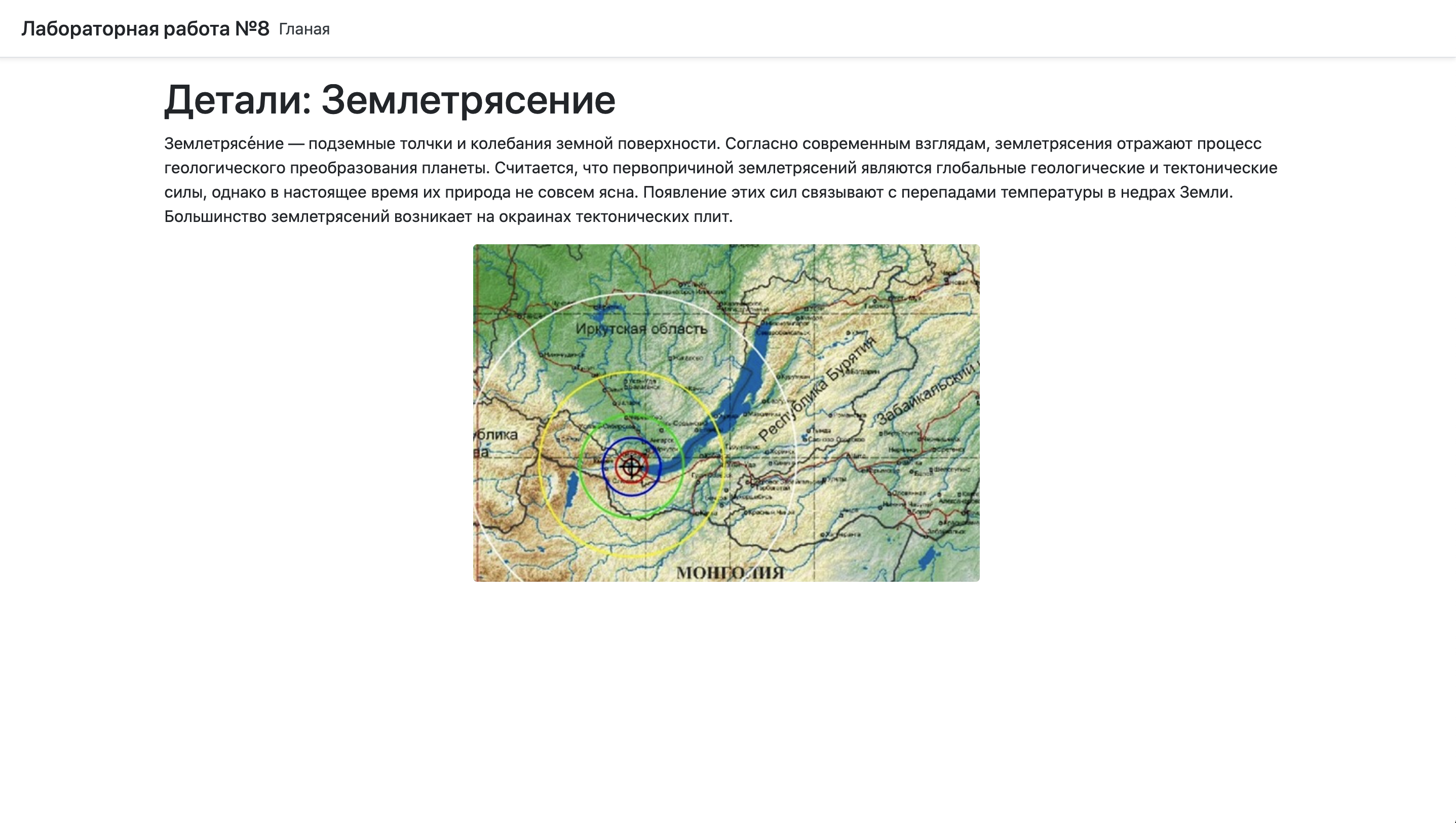
* main/templates/main/details.html

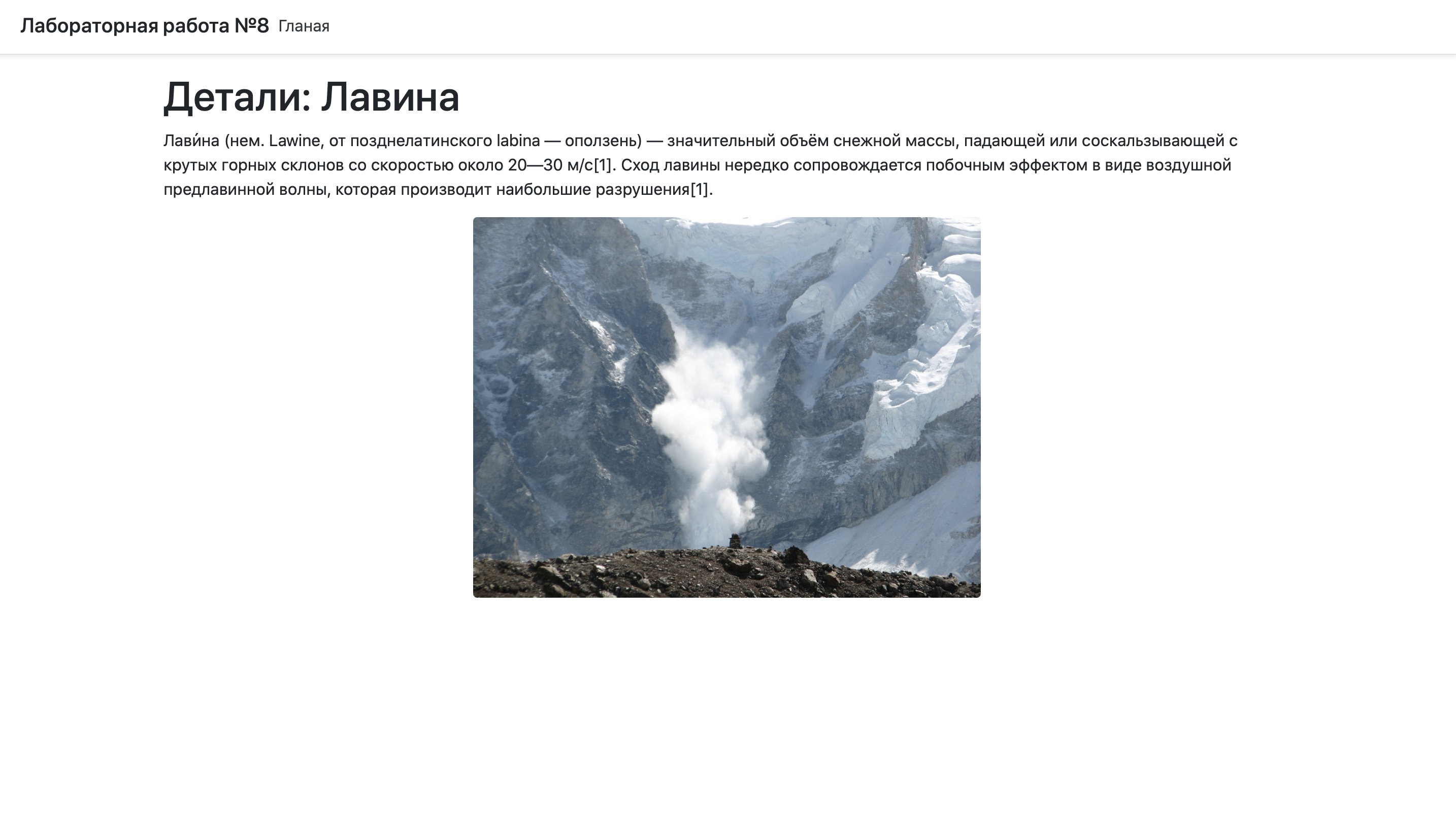
{% extends 'main/base.html'%}  
  
{% block title %}  
 {{ title }}  
{% endblock %}  
  
{% block styles %}  
 .center {  
 display: block;  
 margin-left: auto;  
 margin-right: auto;  
 width: 50%;  
 }  
{% endblock %}  
  
{% block content %}  
 <h1>Детали: {{ pnen.name }}</h1>  
 <p class="card-text">{{ description.description }}</p>  
 {% load static %}  
 {% if pnen.id == 1 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/1.jpg' %}" alt="My image">  
 {% endif %}  
 {% if pnen.id == 2 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/2.jpg' %}" alt="My image">  
 {% endif %}  
 {% if pnen.id == 3 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/3.jpg' %}" alt="My image">  
 {% endif %}  
 {% if pnen.id == 4 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/4.jpg' %}" alt="My image">  
 {% endif %}  
 {% if pnen.id == 5 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/5.jpg' %}" alt="My image">  
 {% endif %}  
 {% if pnen.id == 6 %}  
 <img class="center rounded img-fluid" style="max-width: 500px; height: auto;" src="{% static 'main/images/6.jpg' %}" alt="My image">  
 {% endif %}  
{% endblock %}

# Результат









Ну и остальные три страницы аналогичны продемонстрированным выше.