Московский государственный технический университет им. Н.Э. Баумана Факультет «Информатика и системы управления»

ДИСЦИПЛИНА:

«Разработка интернет-приложений»

**Отчет по лабораторной работе №4**

**«Django. Шаблонизация»**

Выполнил: Студент 4 курса Факультет ГУИМЦ Группа ИУ5Ц-71Б Гаранин А.В. Преподаватель: Гапанюк Ю.Е.

Москва, 2019 г.

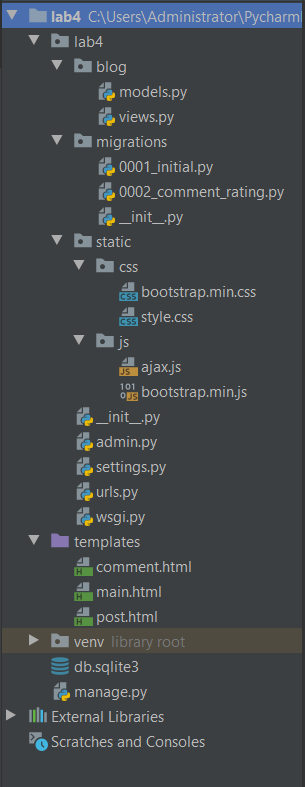
1. **Задание**

В этой ЛР вы создадите Django-проект, покажете пользователю статичную страницу, познакомитесь с конструкциями шаблонизаторов: переменные, теги, наследование шаблонов.

* Создать проект
* Реализовать view, в которых генерируются html-страницы
* В шаблонах должны быть использованы рассмотренные конструкции: переменные, вложенные значения, циклы, условия
* Все шаблоны должны расширять базовый шаблон
* Для элементов списка использовать тег include
* По нажатии на элемент списка должна открываться страница информации об элементе
* Для верстки необходимо использовать Bootstrap

1. **Выполнение задания и исходный код проекта**

Структура проекта:



models.py:

from django.db import models

class Post(models.Model):

post\_head = models.CharField(max\_length=70) post\_text = models.CharField(max\_length=255)

publication\_date = models.DateField('Date published') def str (self):

return self.post\_head

class Comment(models.Model):

id\_post = models.ForeignKey(Post, on\_delete=models.CASCADE) answer\_text = models.CharField(max\_length=70)

rating = models.IntegerField(default=0) def str (self):

return self.answer\_text

views.py:

from django.shortcuts import render, get\_object\_or\_404 from django.db.models import Count

from django.views import generic from .models import \*

class MainView(generic.ListView): template\_name = 'main.html' context\_object\_name = 'main\_postList'

def get\_queryset(self):

return Post.objects.annotate(num=Count('comment')).order\_by('- publication\_date')[:5]

class PostView(generic.DeleteView): model = Post

template\_name = 'post.html'

Использование наследования шаблонов, тегов и переменных (шаблон main.html):

<div class="justify-content-center w-50 mx-auto">

{% block body %}

{% if main\_postList %}

{% for post in main\_postList %}

<div class="card m-2 mb-4 mt-4">

<div class="card-body">

<h5 class="card-title">

<a href="/{{ post.id }}/">{{ post.post\_head }}</a>

</h5>

<p class="card-text">{{post.post\_text}}</p>

</div>

<div class="card-footer">

<span><i class="fas fa-comments"></i> {{ post.num }}</span>

</div>

</div>

{% endfor %}

{% else %}

<div class="alert alert-info" role="alert">

Доступных записей нет.

</div>

{% endif %}

{% endblock %}

</div>

Подключение статических файлов:

{% load staticfiles %}

<link rel="stylesheet" href="{% static 'css/bootstrap.min.css' %}" type="text/css">

<script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>

<script src="{% static 'js/bootstrap.min.js' %}"></script>

Использование include:

Шаблон post.html использует шаблон comment.html

<div class="card mt-4">

<div class="card-body">

<ul class="list-unstyled">

{% for comment in post.comment\_set.all %}

{% include 'comment.html' with comment=comment %}

{% empty %}

<div class="alert alert-info" role="alert">

Комментариев к этой записи пока нет.

</div>

{% endfor %}

</ul>

</div>

</div>