



Московский государственный технический университет имени Н. Э. Баумана

Факультет «Информатика и системы управления»

Кафедра ИУ5

Отчёт по

Лабораторной работе №8

**«Создание прототипа веб-приложения с использованием фреймворка
Django»**

Выполнила:

Калашникова А.В.

Группа ИУ5-54Б

Москва

2020

1. Общее описание задания

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:

- a. 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.
- b. Страница detail содержит детальное описание объекта (сорта мороженого), фотографию, гиперссылку на master-страницу.
- c. Фотография относится к статическому содержимому сайта.
- d. Страница detail должна выводить данные с использованием таблицы HTML.
- e. Шаблон страницы detail получает от представления данные о детальном объекте с использованием контекста.
- f. **НЕОБЯЗАТЕЛЬНЫЙ ПУНКТ.** По желанию можно использовать верстку с применением Bootstrap (или аналогичного фреймворка), а также представления на основе классов (class-based views)

2. Текст программы

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 = 'a23&=zliwqtlxlyco)a-=rk0q(fzbqe3y@og+jkw2vhlgy*^(n'

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

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
```

```

    'polls.apps.PollsConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

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 = 'mysite.urls'

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 = 'mysite.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 = 'en-us'

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/'

```

mysite/urls.py

```

from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('polls/', include('polls.urls')),
    path('admin/', admin.site.urls),
]

```

polls/urls.py

```

from django.urls import path
from . import views

urlpatterns = [
    path('', views.index, name='index'),
    path('<int:food_id>', views.details, name='details'),
]

```

models.py

```

from django.db import models

class Food(models.Model):
    name = models.CharField(max_length=30)
    def __str__(self):
        return self.name

```

```
class Choice(models.Model):
    foodnum = models.ForeignKey(Food, on_delete=models.CASCADE)
    description = models.CharField(max_length=200)
```

views.py

```
from django.shortcuts import get_object_or_404, render

from .models import Food
from .models import Choice

def index(request):
    food_list = Food.objects.all()
    context = {'food_list': food_list}
    return render(request, 'polls/index.html', context)

def details(request, food_id):
    food_list = Food.objects.all()
    for i in food_list:
        if i.id == food_id:
            food = i
    description = get_object_or_404(Choice, pk=food_id)
    return render(request, 'polls/details.html', {'food': food,
'description': description})
```

polls/templates/polls/index.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">
    <title>Lab8</title>
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <link href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.0.0-
beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
giJf6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1"
crossorigin="anonymous">
</head>
<body class="d-flex h-100 text-center text-white" style="background-color:
#0c5460;">
<div class="cover-container d-flex w-100 h-100 p-3 mx-auto flex-column">
    <header class="mb-auto">
        <div class="container" style="background-color: black; height:
60pt;">
            <h3 style="text-align: center; margin: 0 auto;">Выберете кухню
мира</h3>
            <nav class="nav nav-masthead justify-content-center float-md-
end">
                <a class="nav-link active" aria-current="page" href="#">Ha
главную</a>
            </nav>
        </div>
    </header>

    <main>
        {% if food_list %}
        <ul class="list-group" style="width: 40%; margin: 0 auto;">
```

```

        {% for food in food_list %}
        <li class="list-group-item" style="margin-top: 10pt;" >
            <a class="nav-link list" href="/polls/{{food.id}}/">{{
food.name }}</a>
        </li>
        {% endfor %}
    </ul>
    {% else %}
    <p>Sorry, there's no food</p>
    {% endif %}
</main>
</div>
</body>
</html>

```

polls/templates/polls/details.html

```

<!DOCTYPE html>
<html>
<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>Lab8</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-
beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
giJf6kkoqN0Q0vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHYTfAAsrekwKmP1"
crossorigin="anonymous">
    {% load static %}
</head>
<body class="d-flex h-100 text-center text-white bg-dark">
<div class="cover-container d-flex w-100 h-100 p-3 mx-auto flex-column">
    <header class="mb-auto">
        <div class="container" style="background-color: black; height:
60pt;">
            <h3 style="text-align: center; margin: 0 auto;">Выбранная
кухня</h3>
            <nav class="nav nav-masthead justify-content-center float-md-
end">
                <a class="nav-link active" aria-current="page"
href="/polls/">На главную</a>
            </nav>
        </div>
    </header>
    <main class="container-fluid">
        <h1 class="text-center text-light">{{ food.name }}</h1>
        <table class="table table-dark table-borderless">
            <tr>
                <td class="text-light td-text-short" style="text-align:
justify">{{ description.description }}</td>

                {% if food.id == 1 %}
                

                {% endif %}

                {% if food.id == 2 %}
                

```

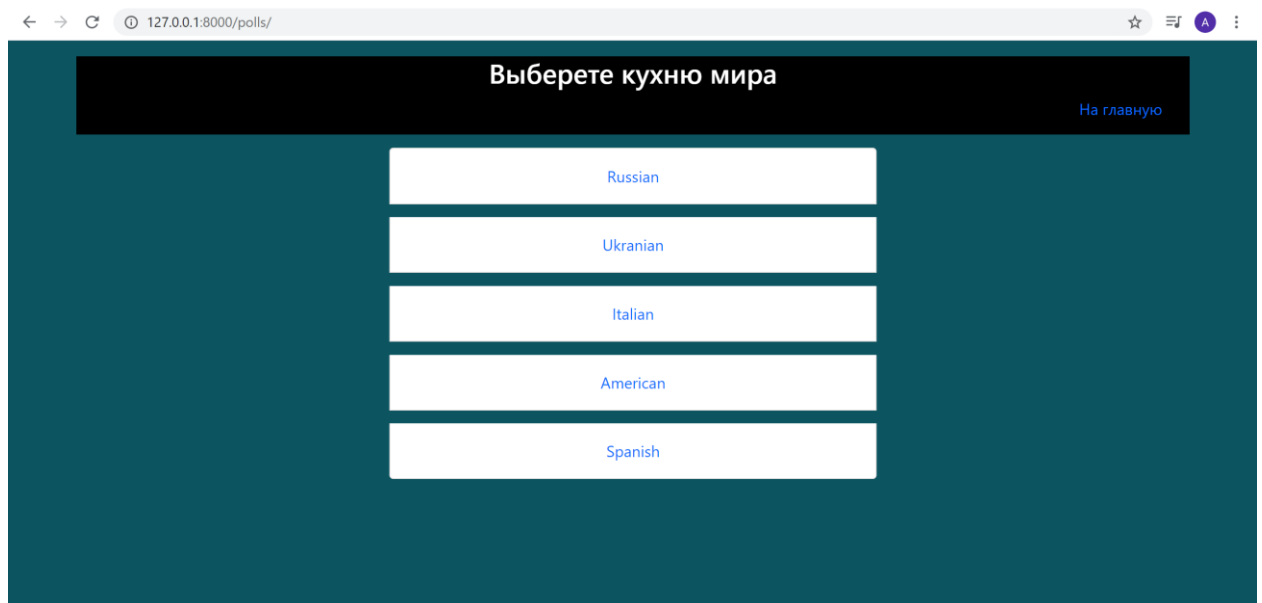
```

        {% endif %}
        {% if food.id == 3 %}
        
        {% endif %}
        {% if food.id == 4 %}
        
        {% endif %}
        {% if food.id == 5 %}
        
        {% endif %}

    </tr>
</table>
</main>
</div>
</body>
</html>

```

3. Экранные формы



Выбранная кухня

[На главную](#)

Russian



Традиционным русским блюдом (особенно любимым всеми на новый год) является запеченный гусь. Рождественского гуся чаще всего набивают яблоками, пищевыми каштанами, луком или черносливом.

Выбранная кухня

[На главную](#)

Italian



Какая у Вас первая ассоциация приходит в голову, когда Вы слышите слово Италия? У многих это паста! Блюдо находится в обиходе у всех слоев населения, от мала до велика.