



Московский государственный технический университет
им. Н.Э. Баумана
(МГТУ им. Н.Э. Баумана)
Радиотехнический факультет (РТ)

Отчёт по лабораторной работе № 6

По дисциплине

«Разработка интернет-приложений»

Проверила:

Преподаватель кафедры ИУ-5

Гапанюк Ю.Е.

Подпись: _____

«___» _____ 201_ г.

Выполнил:

студент группы РТ5-51Б

Ануров Н.С.

Подпись: _____

«___» _____ 201_ г.

Москва, 2019

1. Задание

Основная цель данной лабораторной работы – научиться обрабатывать веб-формы на стороне приложения, освоить инструменты, которые предоставляет Django, по работе с формами. Также в этой лабораторной работе вы освоите инструменты Django по работе с авторизацией и реализуете простейшую авторизацию. Напоследок, вы познакомитесь с инструментом администрирования Django – как в несколько строчек кода сделать панель администратора сайта.

2. Выполнение задания и исходный код проекта

Создайте view, которая возвращает форму для регистрации:

Действие: Выбрано 0 объектов из 2

ИМЯ ПОЛЬЗОВАТЕЛЯ	АДРЕС ЭЛЕКТРОННОЙ ПОЧТЫ	ИМЯ	ФАМИЛИЯ	СТАТУС ПЕРСОНАЛА
admin	admin@gmail.com			✓
Nikita	anurov@icloud.com	Никита	Ануров	✗

2 пользователи

← → ↻ 127.0.0.1:8000/registration/

Имя пользователя:

Пароль:

Подтверждение пароля: Для подтверждения введите, пожалуйста, пароль ещё раз.

Имя:

Фамилия:

Почта:

```
class RegistrationForm(FormView):
    form_class = RegistrationForm
    success_url = "/login/"
    template_name = "registration.html"

    def form_valid(self, form):
        form.save()
        return super(RegistrationForm, self).form_valid(form)

    def form_invalid(self, form):
        return super(RegistrationForm, self).form_invalid(form)
```

Создайте view, которая возвращает форму для авторизации:

← → ↻ ⓘ 127.0.0.1:8000/login/

Имя пользователя:

Пароль:

```
class LoginForm(FormView):
    form_class = LoginForm
    success_url = "/profile/"
    template_name = "login.html"

    def form_valid(self, form):
        self.user = form.get_user()
        login(self.request, self.user)
        return super(LoginForm, self).form_valid(form)
```

Декоратор @login_required и форма Logout:

```
@login_required(login_url='/login/')
def user_profile(request):
    return render(request, 'profile.html')

class LogoutForm(View):
    def get(self, request):
        logout(request)
        return HttpResponseRedirect("/")
```

← → ↻ ⓘ 127.0.0.1:8000/profile/

ТЫ ВОШЕЛ В АККАУНТ

[Выйди](#)

Добавить superuser'а через команду manage.py

```
createsuperuser
->Username: admin
->Email address: admin@gmail.com
->Password: ****
->Password (again): ****
->Superuser created successfully.
```

<input type="checkbox"/>	ИМЯ ПОЛЬЗОВАТЕЛЯ	АДРЕС ЭЛЕКТРОННОЙ ПОЧТЫ	ИМЯ	ФАМИЛИЯ	СТАТУС ПЕРСОНАЛА
<input type="checkbox"/>	admin	admin@gmail.com			✓

Настройка админ-панели

Администрирование сайта

LAB4		
Comments	+ Добавить	✎ Изменить
Posts	+ Добавить	✎ Изменить
ПОЛЬЗОВАТЕЛИ И ГРУППЫ		
Группы	+ Добавить	✎ Изменить
Пользователи	+ Добавить	✎ Изменить

```
from django.contrib import admin
from .blog.models import *

class PostAdmin(admin.ModelAdmin):
    list_display = ('post_text', 'publication_date')
    list_filter = ['publication_date']
    search_fields = ['post_head']

admin.site.register(Post, PostAdmin)
admin.site.register(Comment)
```