

Защищено:

Гапанюк Ю.Е.

Демонстрация ЛР:

Гапанюк Ю.Е.

"__" _____ 2016 г.

"__" _____ 2016 г.

**Отчет по лабораторной работе № 4 по курсу
Разработка интернет приложений**

«Шаблонизация»

Вариант № <20>

ИСПОЛНИТЕЛЬ:

студент группы ИУ5-54

(подпись)

Савельева М. А.

"__" _____ 2016 г.

Задание лабораторной работы

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.

1. Создайте view, которая возвращает форму для регистрации.

Поля формы:

- Логин
- Пароль
- Повторный ввод пароля
- Email
- Фамилия
- Имя

2. Создайте view, которая возвращает форму для авторизации. Поля формы:

- Логин
- Пароль

3. При отправке формы регистрации во view проверять каждый параметр по правилам валидации, если валидация всех полей пройдена, то создавать пользователя и делать перенаправление на страницу логина, а ошибки, если они есть, выводить над формой.

Правила валидации:

- Логин не меньше 5 символов
- Пароль не меньше 8 символов
- Пароли должны совпадать
- Все поля должны быть заполнены
- Логин – уникален для каждого пользователя

4. При возникновении ошибок в момент отправки формы, введенные значения в полях ввода, кроме пароля, не должны исчезать.

5. Переписать view регистрации с использованием Django Form, правила валидации удалить из view, использовать встроенный механизм валидации полей.

6. Во view авторизации реализовать логин при POST запросе. При успешной авторизации должен происходить переход на страницу успешной авторизации.

7. Страница успешной авторизации должна проверять, что пользователь авторизован. Иначе делать перенаправление на страницу авторизации. 8. Реализовать view для выхода из аккаунта.

9. Заменить проверку на авторизацию на декоратор login_required
10. Добавить superuser'а через команду manage.py
11. Подключить django.contrib.admin и войти в панель администрирования.
12. Зарегистрировать все свои модели в django.contrib.admin
13. Для выбранной модели настроить страницу администрирования:
 - Настроить вывод необходимых полей в списке
 - Добавить фильтры
 - Добавить поиск
 - Добавить дополнительное поле в список

Текст программы

Файл admin.py:

```
from django.contrib import admin
from .models import ServiceModel, OrderModel, User

class ServiceAdmin(admin.ModelAdmin):
    list_display = ('service_type', 'service_price')
    search_fields = ('service_type', 'service_price')

admin.site.register(ServiceModel, ServiceAdmin)

class OrderAdmin(admin.ModelAdmin):
    def item_amount(self, obj):
        data = ServiceModel.objects.filter(ord_num=obj.id).all()
        i = len(data)
        return i
    list_display = ('order_num', 'order_date', 'item_amount')
    list_filter = ['order_date']

admin.site.register(OrderModel, OrderAdmin)

class UserAdmin(admin.ModelAdmin):
    list_display = ('username', 'first_name', 'last_name')

admin.site.register(User, UserAdmin)
```

Файл views.py:

```
from django.shortcuts import render, redirect
from django.contrib.auth import logout, authenticate
from django.contrib import auth
from django.contrib.auth.decorators import login_required
from my_LAB_7.forms import *
```

```
def signin(request):
    if request.method == 'POST':
        form = RegistrationForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('success/')
        return render(request, 'signin.html', {'form': form})
    else:
        form = RegistrationForm()
        return render(request, 'signin.html', {'form': form})

def signin_success(request):
    return render(request, 'signin_success.html')

def signup(request):
    errors = []
    if request.method == 'POST':
        username = request.POST.get('username')
        password = request.POST.get('password')
        if not username:
            errors.append('Введите логин!')
        elif not password:
            errors.append('Введите пароль!')
        else:
            user = authenticate(username=username, password=password)
            if user:
                auth.login(request, user)
                return redirect('success/')
            else:
                errors.append('Неправильный логин или пароль!')
        return render(request, 'signup.html', {'errors': errors, 'username':
username})
    return render(request, 'signup.html', {'errors': errors})

@login_required
def signup_success(request):
    return render(request, 'signup_success.html')

@login_required
def logout_view(request):
    return render(request, 'logout.html')
```

Файл forms.py:

[illegible]

```

password_2 = forms.CharField(widget=forms.PasswordInput(attrs={'class': 'form-control', 'id': 'password_2'}),
                             min_length=8, label='Повторите пароль:')

def rem_password_2(self):
    if self.cleaned_data.get('password') != self.cleaned_data.get('password_2'):
        raise forms.ValidationError('Passwords does not match!')

def save(self):
    u = User()
    u.username = self.cleaned_data.get('username')
    u.password = make_password(self.cleaned_data.get('password'))
    u.first_name = self.cleaned_data.get('name')
    u.last_name = self.cleaned_data.get('surname')
    u.email = self.cleaned_data.get('email')
    u.is_staff = False
    u.is_active = True
    u.is_superuser = False
    u.save()

def rem_username(self):
    username = self.cleaned_data.get('username')
    try:
        u = User.objects.get(username=username)
        raise forms.ValidationError('This login is not available')
    except User.DoesNotExist:
        return username

```

Файл models.py:

```

from django.db import models
from django.contrib.auth.models import AbstractUser

class ServiceModel (models.Model):
    id = models.AutoField(primary_key=True)
    service_price = models.CharField(max_length=20)
    service_type = models.CharField(max_length=40)
    service_class_of_price = models.CharField(max_length=20)
    ord_num = models.ManyToManyField('OrderModel')

class User(AbstractUser):
    phone = models.CharField(max_length=15, default='')
    address = models.TextField(verbose_name='Адрес', default='')

    class Meta(AbstractUser.Meta):
        swappable = 'AUTH_USER_MODEL'

class OrderModel(models.Model):
    id = models.AutoField(primary_key=True)
    order_num = models.IntegerField(unique=True)
    order_date = models.DateField()
    user_id = models.ForeignKey(User)

    def __str__(self):
        return str(self.order_num)

```

Файл base.html:

```

{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <link rel="stylesheet" type="text/css" href="/static/bootstrap-3.3.5-dist/css/bootstrap.min.css">

```

```

<title>{% block title %}{% endblock %}</title>
</head>
<body>
<style type="text/css">
    .jumbotron
    {
        background: #42aaff;
        color: #fff;
    }
</style>
<div class="jumbotron">
    <div class="container">
        <em>{% block title-visible %}{% endblock %}</em>
    </div>
</div>
<div class="container-fluid">
    {% block body %}{% endblock %}
</div>
</body>
</html>

```

Файл logout.html:

```

{% extends 'base.html' %}

{% block title %}Logout{% endblock %}

{% block title-visible %}<h1>Выход</h1>{% endblock %}

{% block body %}<p class="text-center"><big>Вы успешно вышли из учётной записи! До
новой встречи!</big></p>{% endblock %}

```

Файл signin.html:

```

{% extends 'base.html' %}

{% block title %}Signin{% endblock %}

{% block title-visible %}<h1>Регистрация</h1>{% endblock %}

{% block body %}
    <form class="form-horizontal" action="" method="post">
        {% csrf_token %}
        <div class="form-group">
            {% if form.errors %}
                <div class="alert alert-danger">
                    {% for f in form %}
                        {% if f.errors %}
                            {% for el in f.errors %}
                                <p align="center"><strong>{{ label }}{{ el }}</strong></p>
                            {% endfor %}
                        {% endif %}
                    {% endfor %}
                </div>
            {% endif %}
        </div>

        <div class="form-group">
            <label class="col-xs-2 control-label">Логин:</label>
            <div class="col-xs-3 input-group-sm">
                {{ form.username }}
            </div>
        </div>

        <div class="form-group">
            <label class="col-xs-2 control-label">Имя:</label>
            <div class="col-xs-3 input-group-sm">
                {{ form.name }}
            </div>
        </div>
    </form>

```

```

</div>
<div class="form-group">
  <label for="enterSurname" class="col-xs-2 control-label">Фамилия:</label>
  <div class="col-xs-3 input-group-sm">
    {{ form.surname }}
  </div>
</div>
<div class="form-group">
  <label class="col-xs-2 control-label">Email:</label>
  <div class="col-xs-3 input-group-sm">
    {{ form.email }}
  </div>
</div>
<div class="form-group">
  <label for="inputPassword" class="col-xs-2 control-label">Пароль:</label>
  <div class="col-xs-3 input-group-sm">
    {{ form.password }}
  </div>
</div>
<div class="form-group">
  <label for="confirmPassword" class="col-xs-2 control-label">Повторите
  пароль:</label>
  <div class="col-xs-3 input-group-sm">
    {{ form.password_2 }}
  </div>
</div>

  <div class="form-group">
    <div class="col-xs-offset-2 col-xs-1">
      <button type="submit" class="btn btn-primary" style="float:
left">Зарегистрироваться</button>
    </div>
    <div class="col-xs-2">
      <a href="{% url 'signup' %}" class="btn btn-default" style="float:
right">Войти</a>
    </div>
  </div>
</form>
{% endblock %}

```

Файл signin_success:

```

{% extends 'base.html' %}

{% block title %}Success{% endblock %}

{% block title-visible %}<h1>Завершение регистрации</h1>{% endblock %}

{% block body %}<p class="text-center"><big>Вы успешно зарегистрированы!</big></p>{%
endblock %}

```

Файл signup.html:

```

{% extends 'base.html' %}

{% block title %}Signup{% endblock %}

{% block title-visible %}<h1>Авторизация</h1>{% endblock %}

{% block body %}
  <form class="form-horizontal" action="" method="post">
    {% csrf_token %}
    <div class="form-group">
      {% for el in errors %}
        <div class="alert alert-danger" role="alert">
          <strong>{{ el }}</strong>
        </div>
      {% endfor %}
    </div>
  </form>

```

```

<div class="form-group">
  <label for="id_username" class="control-label col-sm-2">Логин:</label>
  <div class="col-sm-3">
    <input class="form-control" type="text" name="username" id="id_username"
value="{{ username }}">
  </div>
</div>
<div class="form-group">
  <label for="id_password" class="control-label col-sm-2">Пароль:</label>
  <div class="col-sm-3">
    <input class="form-control" type="password" name="password"
id="id_password">
  </div>
</div>

<div class="form-group">
  <div class="col-xs-offset-2 col-xs-1">
    <button type="submit" class="btn btn-primary" style="float:
left">Войти</button>
  </div>
  <div class="col-xs-2">
    <a href="{% url 'signin' %}" class="btn btn-default" style="float:
right">Зарегистрироваться</a>
  </div>
</div>
</form>
{% endblock %}

```

Файл signup_success:

```

{% extends 'base.html' %}

{% block title %}Success{% endblock %}

{% block title-visible %}<h1>Завершение авторизации</h1>{% endblock %}

{% block body %}
  <p class="text-center">
    <big>Вы успешно авторизированы!</big>
  </p>
  <div class="text-center">
    <a href="{% url 'logout' %}" class="btn btn-primary">Выйти</a>
  </div>
{% endblock %}

```

Изменения в файле settings.py:

```

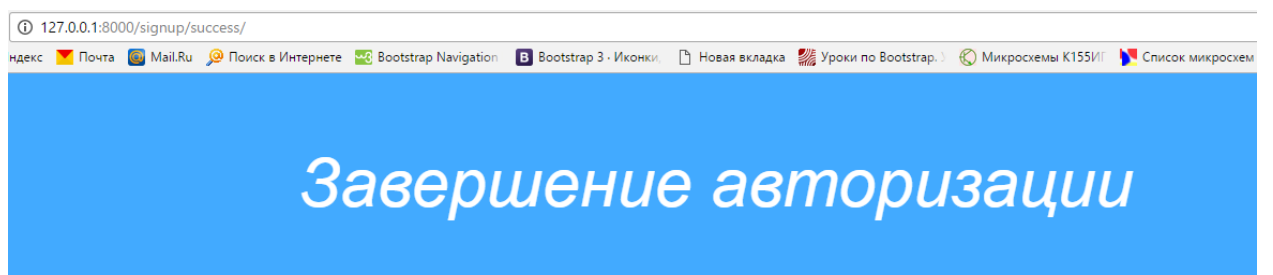
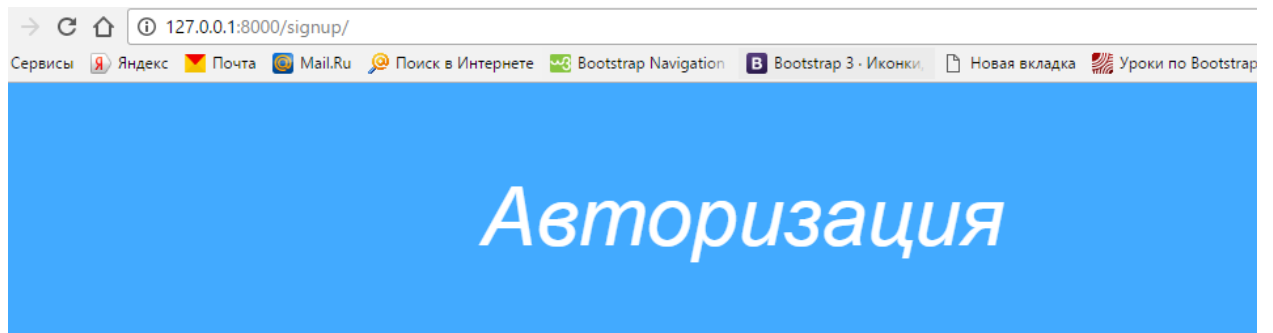
DATABASES = {
  'default': {
    'ENGINE': 'django.db.backends.mysql',
    'NAME': 'service_log',
    'USER': 'dbuser',
    'PASSWORD': '123',
    'HOST': 'localhost',
    'PORT': 3306,
    'OPTIONS': {'charset': 'utf8'},
    'TEST_CHARSET': 'utf8',
  }
}

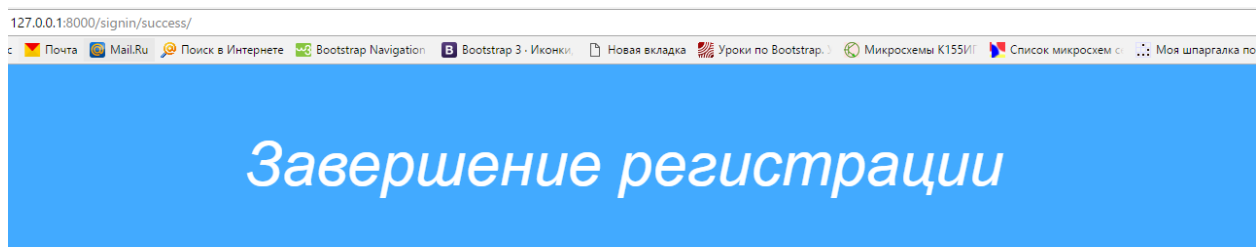
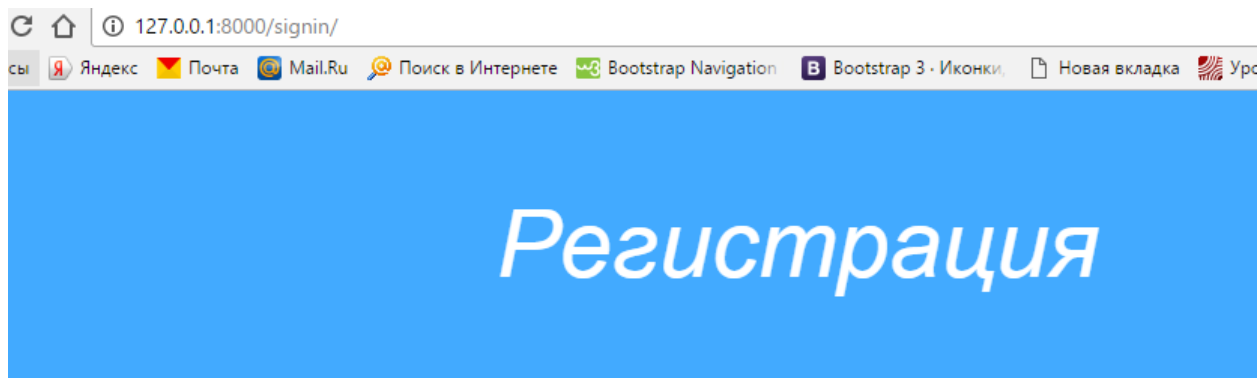
```

```
LOGIN_URL = 'signup'
```

```
AUTH_USER_MODEL = 'my_LAB_7.User'
```


Результаты работы программы





127.0.0.1:8000/logout/

Индекс Почта Mail.Ru Поиск в Интернете Bootstrap Navigation Bootstrap 3 - Иконки Новая вкладка Уроки по Bootstrap Микросхемы K155ИИ Список микросхем с

Выход

Вы успешно вышли из учётной записи! До новой встречи!