Московский государственный технический университет имени Н.Э. Баумана

Кафедра ИУ5 «Системы обработки информации и управления»

Отчет по лабораторной работе №7

«Авторизация, работа с формами и Django Admin»

по дисциплине «Разработка Интернет-приложений»

Выполнил: студент
группы ИУ5-53
Иванников Александр

Москва, 2016

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.

- Создайте view, которая возвращает форму для регистрации. **Поля формы:** Логин Пароль, Повторный ввод пароля, Email, Фамилия, Имя
- Создайте view, которая возвращает форму для авторизации. **Поля формы:** Логин, Пароль
- 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
- При возникновении ошибок в момент отправки формы, введенные значения в полях ввода, кроме пароля, не должны исчезать.
- Переписать view регистрации с использованием Django Form, правила валидации удалить из view, использовать встроенный механизм валидации полей.
- Во view авторизации реализовать логин при POST запросе. При успешной авторизации должен происходить переход на страницу успешной авторизации.
- Страница успешной авторизации должна проверять, что пользователь авторизован. Иначе делать перенаправление на страницу авторизации.
- Реализовать view для выхода из аккаунта.
- Заменить проверку на авторизацию на декоратор login_required
- Добавить superuser’a через комманду manage.py
- Подключить django.contrib.admin и войти в панель администрирования.
- Зарегистрировать все свои модели в django.contrib.admin
- Для выбранной модели настроить страницу администрирования:
- Настроить вывод необходимых полей в списке Добавить фильтры, Добавить поиск, Добавить дополнительное поле в список

Исходный код

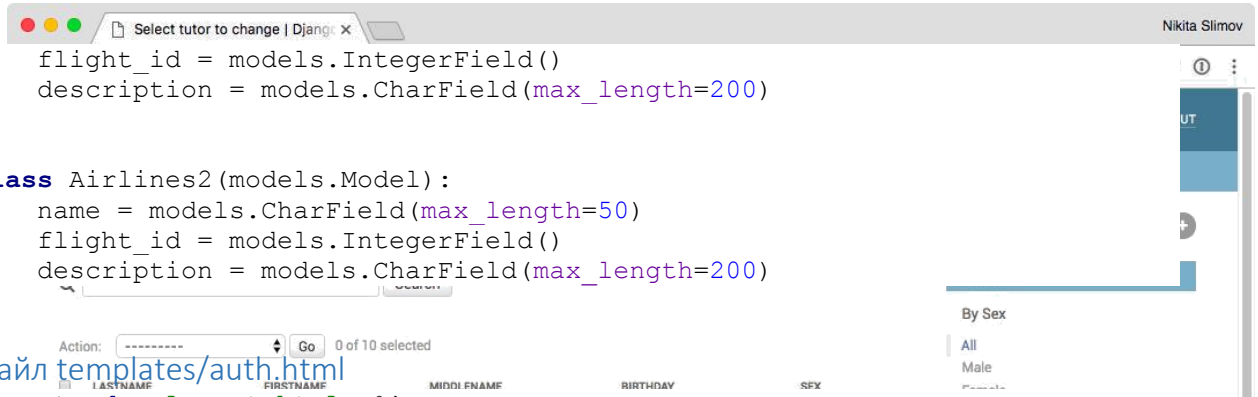
Файл lab/forms.py

```
from django import forms
from django.core.exceptions import ValidationError
from django.core.validators import validate_email
from django.contrib.auth.models import User


class LoginForm(forms.Form):
    login = forms.CharField(label='Login')
```

```
    password = forms.CharField(label='Password',
widget=forms.PasswordInput)


class SignupForm(forms.Form):
    login = forms.CharField(label='Login', min_length=5)
    email = forms.CharField(label='Email')
    first_name = forms.CharField(label='First name')
    last_name = forms.CharField(label='Last name')
    password = forms.CharField(label='Password', min_length=8,
widget=forms.PasswordInput)
    repeat_password = forms.CharField(label='Repeat password',
widget=forms.PasswordInput)

    def clean_login(self):
        login = self.cleaned_data['login']
        if User.objects.filter(username=login):
            raise ValidationError('Этот login уже занят')
        return login

    def clean_email(self):
        email = self.cleaned_data['email']
        validate_email( self.cleaned_data['email'])
        if User.objects.filter(email=email):
            raise ValidationError('Этот email уже зарегистрирован')
        return self.cleaned_data['email']

    def clean(self):
        cleaned_data = super(SignupForm, self).clean()
        if self.cleaned_data.get('password') and
self.cleaned_data.get('repeat_password'):
            if self.cleaned_data['password'] !=
self.cleaned_data['repeat_password']:
                raise ValidationError('Пароли не совпадают')
        return cleaned_data

    def save(self):
        user =
User.objects.create_user(username=self.cleaned_data['login'],
                                         email=self.cleaned_data['email'],

password=self.cleaned_data['password'],

first_name=self.cleaned_data['first_name'],

last_name=self.cleaned_data['last_name'],
                                         )
        return user
```

Файл lab/models.py

```
from django.db import models


class User2(models.Model):
    name = models.CharField(max_length=50)
    description = models.CharField(max_length=200)


class Airlines(models.Model):
    name = models.CharField(max_length=50)
```

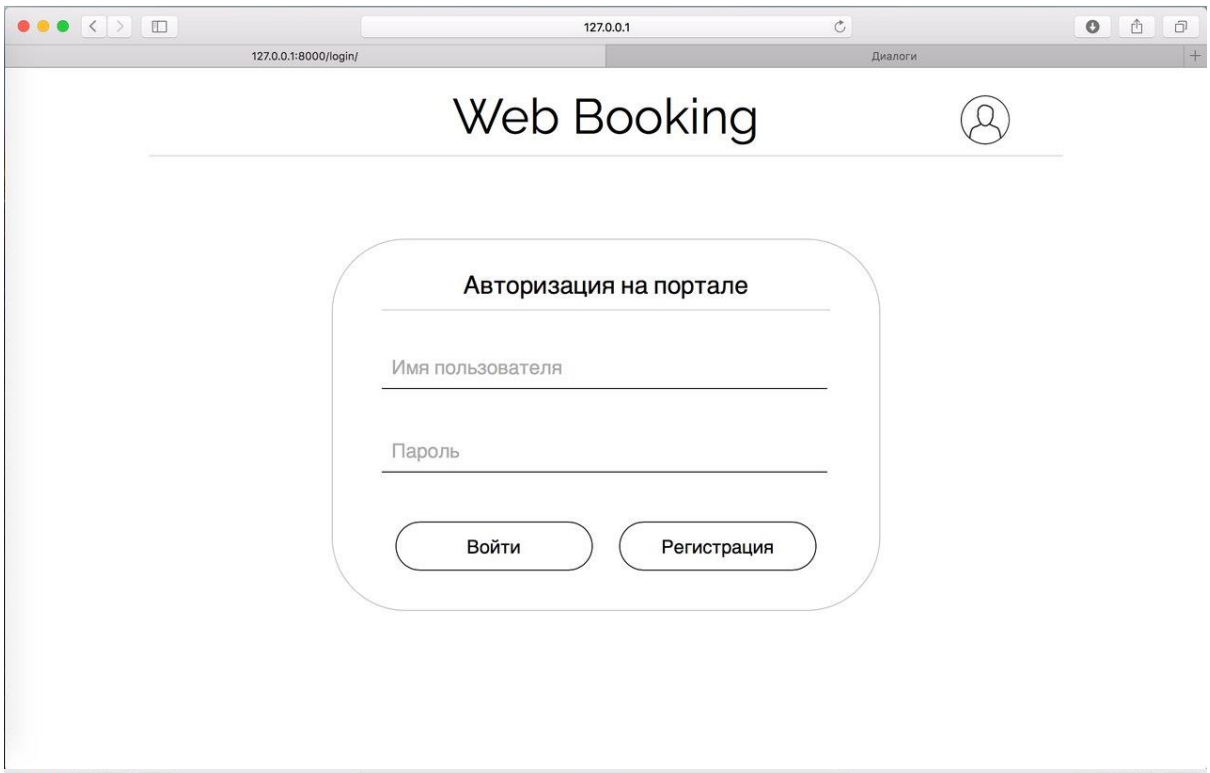


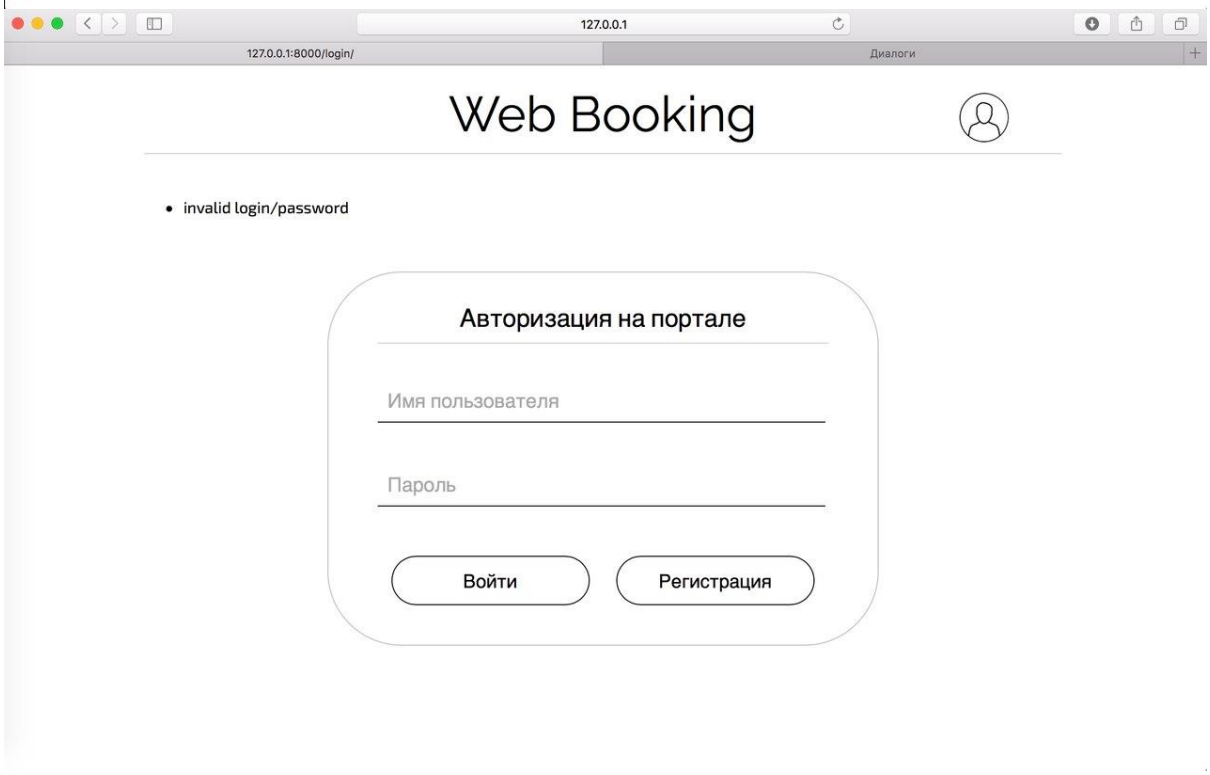

```
    flight_id = models.IntegerField()
    description = models.CharField(max_length=200)


class Airlines2(models.Model):
    name = models.CharField(max_length=50)
    flight_id = models.IntegerField()
    description = models.CharField(max_length=200)
```

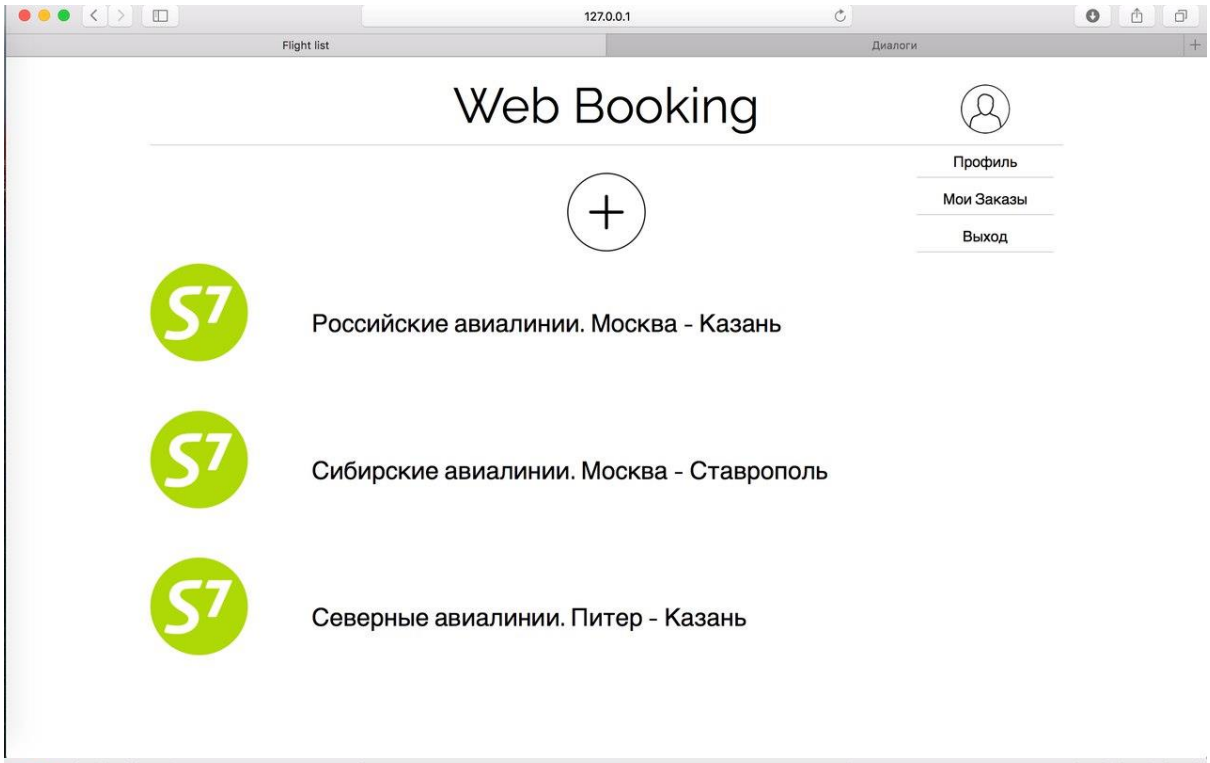
Файл templates/auth.html

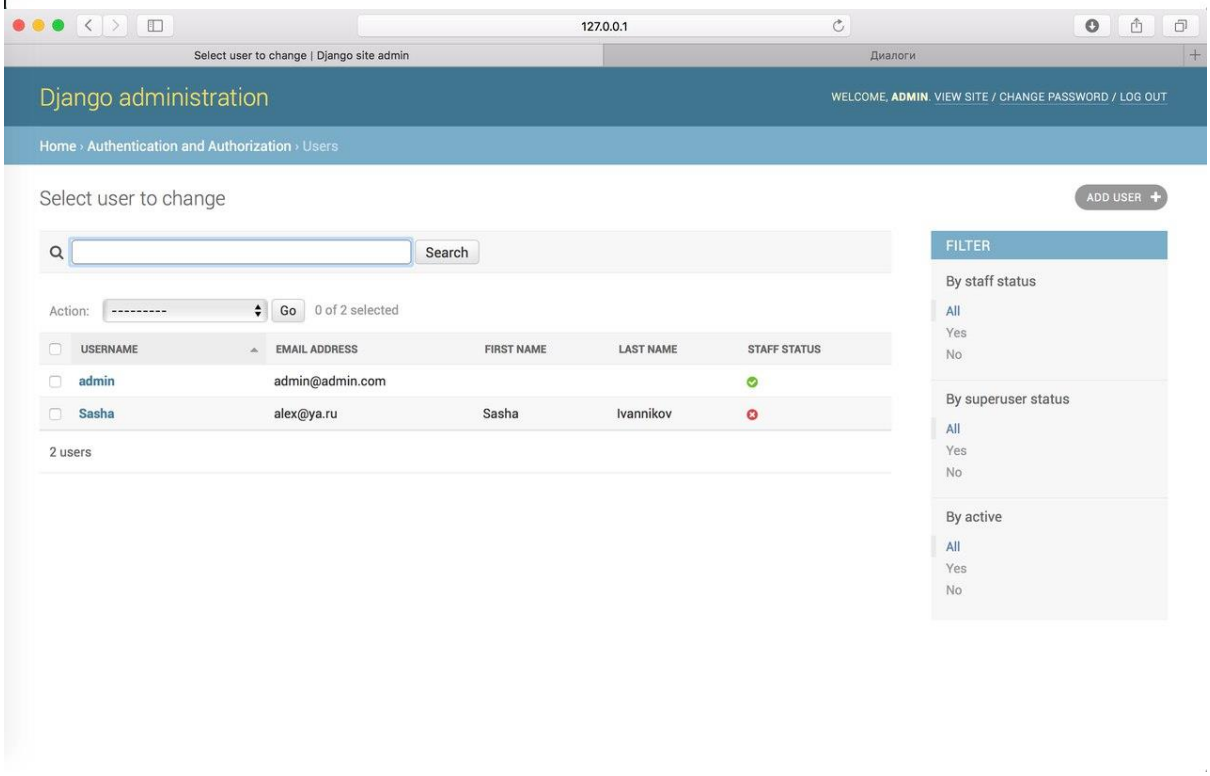
```
{% extends 'layout.html' %}
{% block body %}
    {{ form.non_field_errors }}
            <form action="/login/" method="post" class="sign">
              {% csrf_token %}

                <fieldset class="account-info">
                    <div class="sign-legend">Авторизация на портале</div>

                    <label>
                      <!  Имя пользователя !>
                        <input type="text" name="login" placeholder="Имя
пользователя">
                    </label>
                    <label>
                       <! Пароль !>
                        <input type="password" name="password"
placeholder="Пароль">
                    </label>

                    <input class="btn" type="submit" name="submit"
value="Войти">
                    <input class="btn" type="submit" name="submit"
value="Регистрация" disabled>
                </fieldset>
            </form>
{% endblock %}
```

Результат


127.0.0.1
127.0.0.1:8000/login/
Диалоги
Web Booking
Авторизация на портале
Имя пользователя
Пароль
Войти
Регистрация


127.0.0.1
127.0.0.1:8000/login/
Диалоги
Web Booking
invalid login/password
Авторизация на портале
Имя пользователя
Пароль
Войти
Регистрация


127.0.0.1
Flight list
Диалоги
Web Booking
Профиль
Мои Заказы
Выход
Российские авиалинии. Москва - Казань
Сибирские авиалинии. Москва - Ставрополь
Северные авиалинии. Питер - Казань


127.0.0.1
Select user to change | Django site admin
Диалоги
Django administration
WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Authentication and Authorization › Users
Select user to change
ADD USER
Search
Action: --------- Go 0 of 2 selected
USERNAME
EMAIL ADDRESS
FIRST NAME
LAST NAME
STAFF STATUS
admin
admin@admin.com
Sasha
alex@ya.ru
Sasha
Ivannikov
2 users
FILTER
By staff status
All
Yes
No
By superuser status
All
Yes
No
By active
All
Yes
No