МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

ФАКУЛЬТЕТ ИНФОКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

ОТЧЕТ

о лабораторной работе №2

«Реализация простого сайта на django»

по дисциплине

«Web программирование»

Выполнила Голуб А.Л. группа К33421

Проверил Говоров А. И.

Задание: табло отображения информации об авиаперелетах

Хранится информация о номере рейса, авиакомпании, отлете, прилете, типе (прилет, отлет), номере гейта.

Необходимо реализовать следующий функционал:

- Регистрация новых пользователей.
- Просмотр и резервирование мест на рейсах. Пользователь должен иметь возможность редактирования и удаления своих резервирований.
- Администратор должен иметь возможность зарегистрировать на рейс пассажира и вписать в систему номер его билета средствами Django-admin.
- В клиентской части должна формироваться таблица, отображающая всех пассажиров рейса.
- Написание отзывов к рейсам. При добавлении комментариев, должны
- сохраняться дата рейса, текст комментария, рейтинг (1-10), информация о комментаторе.

Модель данных (models.py)

```
# user - пользователь / пассажир
class User(AbstractUser):
   username = models.CharField(max length=30, unique=True)
   password = models.CharField(max length=30)
   last name = models.CharField(max length=30)
   first name = models.CharField(max length=30)
   passport = models.CharField(max length=10, blank=True, null=True)
   def str (self):
        if self.is superuser:
           return 'superuser'
        return self.last name + ' ' + self.first name
# flight - рейс
class Flight(models.Model):
   passengers = models.ManyToManyField(settings.AUTH USER MODEL,
through='Reservation')
   number = models.CharField(max length=30)
   airline = models.CharField(max length=30)
   departure = models.DateField()
   arrival = models.DateField()
   gate = models.CharField(max length=10)
   DEPARTURE = 'Departure'
   ARRIVAL = 'Arrival'
    type = models.CharField(
       max length=30,
       choices=[(DEPARTURE, 'Departure'), (ARRIVAL, 'Arrival')]
    def __str__(self):
        return self.airline + ' ' + self.number
```

```
# reservation - резервирование
class Reservation (models.Model):
    passenger = models.ForeignKey(settings.AUTH USER MODEL,
on delete=models.CASCADE)
    flight = models.ForeignKey(Flight, on delete=models.CASCADE)
    seat = models.CharField(max length=15)
    ticket = models.CharField(max length=15, blank=True, null=True)
   def __str__(self):
       return 'Резервирование'
# review - отзыв
class Review(models.Model):
    author = models.ForeignKey(settings.AUTH USER MODEL,
on delete=models.SET NULL, null=True)
    flight = models.ForeignKey(Flight, on delete=models.SET NULL, null=True)
    rating = models.IntegerField([(i, i) for i in range(1, 10)])
    comment = models.CharField(max length=400, blank=True, null=True)
```

Контроллеры (views.py)

```
# create user
class UserCreateView(CreateView):
   model = User
    fields = ['username', 'password', 'last_name', 'first_name', 'passport']
    success url = '/success'
# view user info
class UserRetrieveView(DetailView):
   model = User
# create reservation
class ReservationCreateView(CreateView):
   model = Reservation
    form class = ReservationForm
    success url = '/success'
# view reservation
class ReservationRetrieveView(DetailView):
   model = Reservation
# update reservation data
class ReservationUpdateView(UpdateView):
   model = Reservation
    form class = ReservationForm
    success_url = '/success'
# delete reservation
class ReservationDeleteView(DeleteView):
   model = Reservation
   success url = '/success'
```

```
# list passengers on specified flight
class PassengerList(ListView):
    model = Reservation
    template_name = 'passenger_list_view.html'

# write review
class ReviewCreateView(CreateView):
    model = Review
    form_class = ReviewForm
    success_url = '/success'

# success
def success(request):
    return render(request, 'success.html')
```

Маршрутизация (urls.py)

```
'user/register/' # create user
'user/<int:pk>/' # view user info

'reservation/create' # create reservation
'reservation/<int:pk>' # view reservation info
'reservation/edit/<int:pk>' # edit reservation
'reservation/delete/<int:pk>' # delete reservation
'flight/<int:pk>' # view list of passengers on a specified flight
'review' # write review

path('success', success) # success url
```

Примеры работы программы

Регистрация пользователя

← → C ♠ ① localhost:8000/user/register/
Username: lida
Password: lida
Last name: Григорьева
First name: Лидия
Passport: 1212121212
Submit

Просмотр информации о пользователе

← → C ↑ ① localhost:8000/user/6/

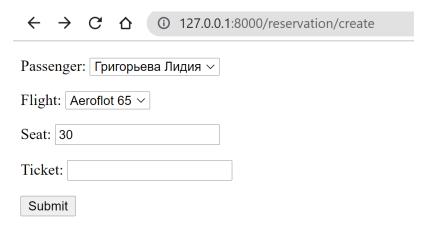
Григорьева Лидия

Фамилия: Григорьева

Имя: Лидия

Паспортные данные: 1212121212

Создание резервирования



Просмотр резервирования



Резервирование

Пассажир: Григорьева Лидия

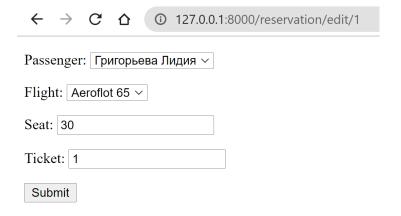
Рейс: 65

Авиакомпания: Aeroflot Время вылета: Nov. 6, 2021 Время прилета: Nov. 6, 2021

Тип: Arrival Гейт: 4 Место: 30

Номер билета: 1

Изменение резервирования



Отмена резервирования



Вы уверены, что хотите отменить резервацию?

Пассажир: Григорьева Лидия

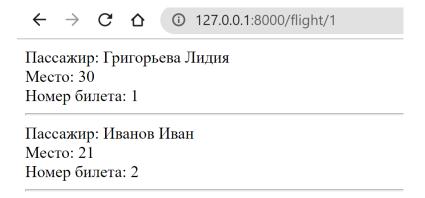
Рейс: 65

Авиакомпания: Aeroflot Время вылета: Nov. 6, 2021 Время прилета: Nov. 6, 2021

Тип: Arrival Гейт: 4 Место: **30** Номер билета: **1**

Confirm

Список зарегистрированных на рейс



Оставить отзыв

← → C ↑ (1) 127.0.0.1:8000/review
Author: Григорьева Лидия У
Flight: Aeroflot 65 V
Rating: 6
Comment: Еда невкусная(
Submit