ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

Факультет «Инфокоммуникационных технологий»
Направление подготовки 09.03.03 «Мобильные и сетевые технологии»

ОТЧЕТ

Тема задания: Реализация простого сайта средствами DJANGO

Выполнил:
Студент: Иванова И.И., K33402
(ФИО, номер группы)

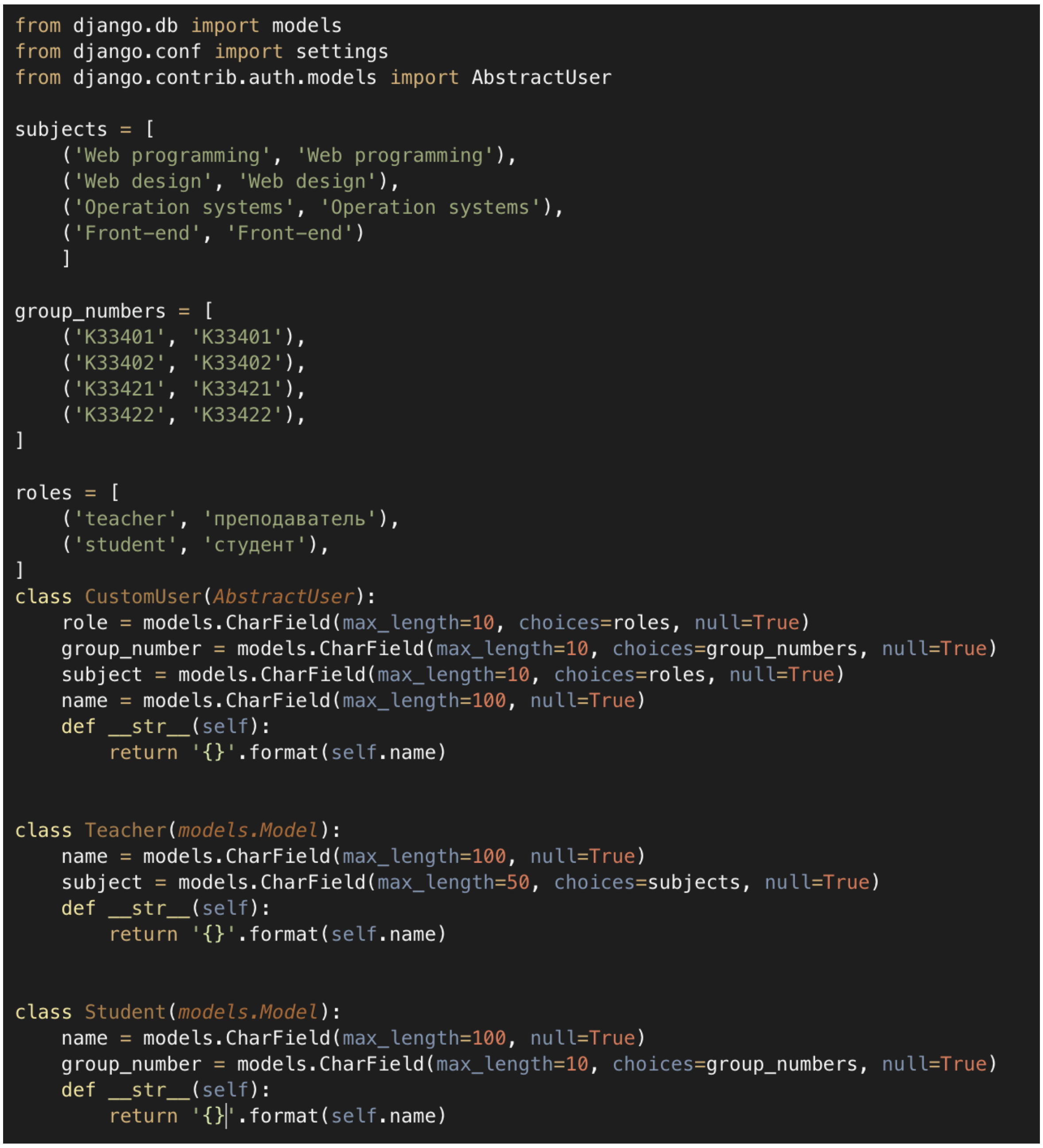
Проверил:
Преподаватель: Говоров А.И.
(ФИО)

Санкт-Петербург
2020

Цель: овладеть практическими навыками и умениями реализации web-сервисов средствами Django 2.2.

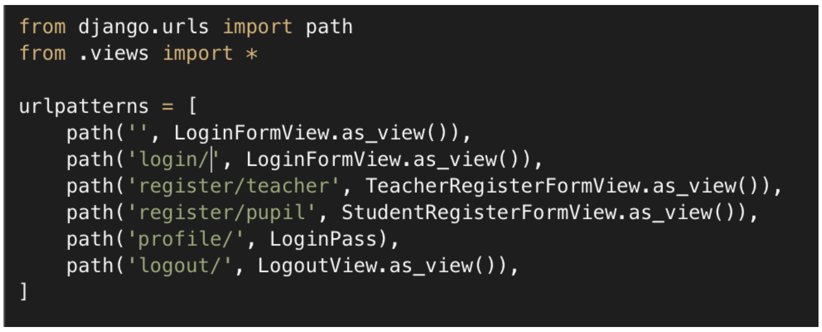
1) Добавим модели пользователя Teacher и Student и сделаем страницы регистрации и входа

models.py



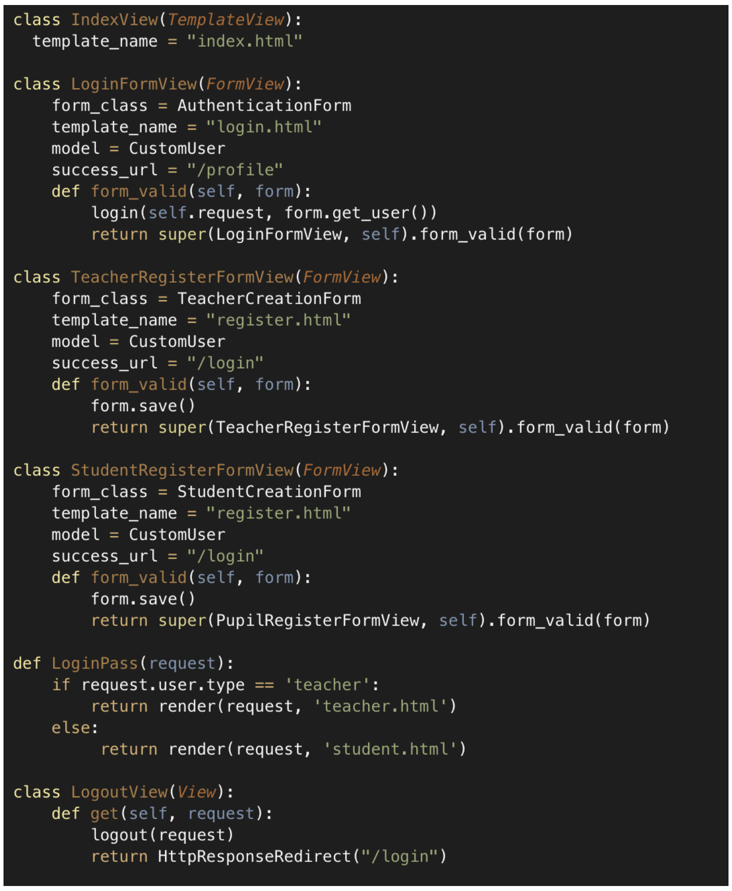
```python
from django.db import models
from django.conf import settings
from django.contrib.auth.models import AbstractUser

subjects = [
    ('Web programming', 'Web programming'),
    ('Web design', 'Web design'),
    ('Operation systems', 'Operation systems'),
    ('Front-end', 'Front-end')
    ]

group_numbers = [
    ('K33401', 'K33401'),
    ('K33402', 'K33402'),
    ('K33421', 'K33421'),
    ('K33422', 'K33422'),
]

roles = [
    ('teacher', 'преподаватель'),
    ('student', 'студент'),
]
class CustomUser(AbstractUser):
    role = models.CharField(max_length=10, choices=roles, null=True)
    group_number = models.CharField(max_length=10, choices=group_numbers, null=True)
    subject = models.CharField(max_length=10, choices=roles, null=True)
    name = models.CharField(max_length=100, null=True)
    def __str__(self):
        return '{}'.format(self.name)


class Teacher(models.Model):
    name = models.CharField(max_length=100, null=True)
    subject = models.CharField(max_length=50, choices=subjects, null=True)
    def __str__(self):
        return '{}'.format(self.name)


class Student(models.Model):
    name = models.CharField(max_length=100, null=True)
    group_number = models.CharField(max_length=10, choices=group_numbers, null=True)
    def __str__(self):
        return '{}'.format(self.name)
```

urls.py



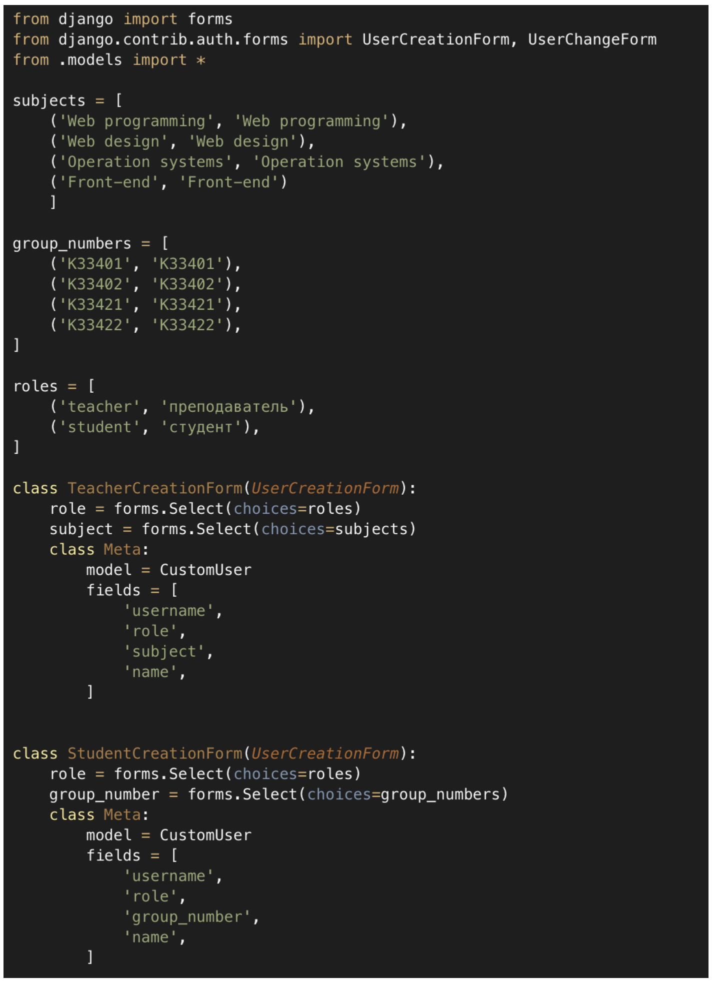
```python
from django.urls import path
from .views import *

urlpatterns = [
    path('', LoginFormView.as_view()),
    path('login/', LoginFormView.as_view()),
    path('register/teacher', TeacherRegisterFormView.as_view()),
    path('register/pupil', StudentRegisterFormView.as_view()),
    path('profile/', LoginPass),
    path('logout/', LogoutView.as_view()),
]
```

views.py



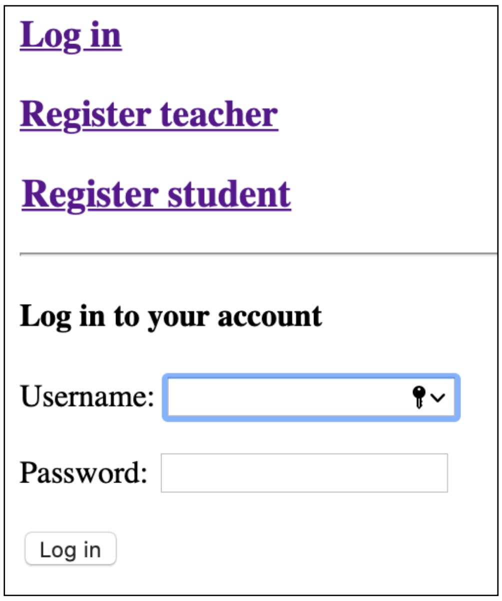
```python
class IndexView(TemplateView):
  template_name = "index.html"

class LoginFormView(FormView):
    form_class = AuthenticationForm
    template_name = "login.html"
    model = CustomUser
    success_url = "/profile"
    def form_valid(self, form):
        login(self.request, form.get_user())
        return super(LoginFormView, self).form_valid(form)

class TeacherRegisterFormView(FormView):
    form_class = TeacherCreationForm
    template_name = "register.html"
    model = CustomUser
    success_url = "/login"
    def form_valid(self, form):
        form.save()
        return super(TeacherRegisterFormView, self).form_valid(form)

class StudentRegisterFormView(FormView):
    form_class = StudentCreationForm
    template_name = "register.html"
    model = CustomUser
    success_url = "/login"
    def form_valid(self, form):
        form.save()
        return super(PupilRegisterFormView, self).form_valid(form)

def LoginPass(request):
    if request.user.type == 'teacher':
        return render(request, 'teacher.html')
    else:
         return render(request, 'student.html')

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

forms.py



```python
from django import forms
from django.contrib.auth.forms import UserCreationForm, UserChangeForm
from .models import *

subjects = [
    ('Web programming', 'Web programming'),
    ('Web design', 'Web design'),
    ('Operation systems', 'Operation systems'),
    ('Front-end', 'Front-end')
    ]

group_numbers = [
    ('K33401', 'K33401'),
    ('K33402', 'K33402'),
    ('K33421', 'K33421'),
    ('K33422', 'K33422'),
]

roles = [
    ('teacher', 'преподаватель'),
    ('student', 'студент'),
]

class TeacherCreationForm(UserCreationForm):
    role = forms.Select(choices=roles)
    subject = forms.Select(choices=subjects)
    class Meta:
        model = CustomUser
        fields = [
            'username',
            'role',
            'subject',
            'name',
        ]


class StudentCreationForm(UserCreationForm):
    role = forms.Select(choices=roles)
    group_number = forms.Select(choices=group_numbers)
    class Meta:
        model = CustomUser
        fields = [
            'username',
            'role',
            'group_number',
            'name',
        ]
```

http://127.0.0.1:8000



Log in

Register teacher

Register student

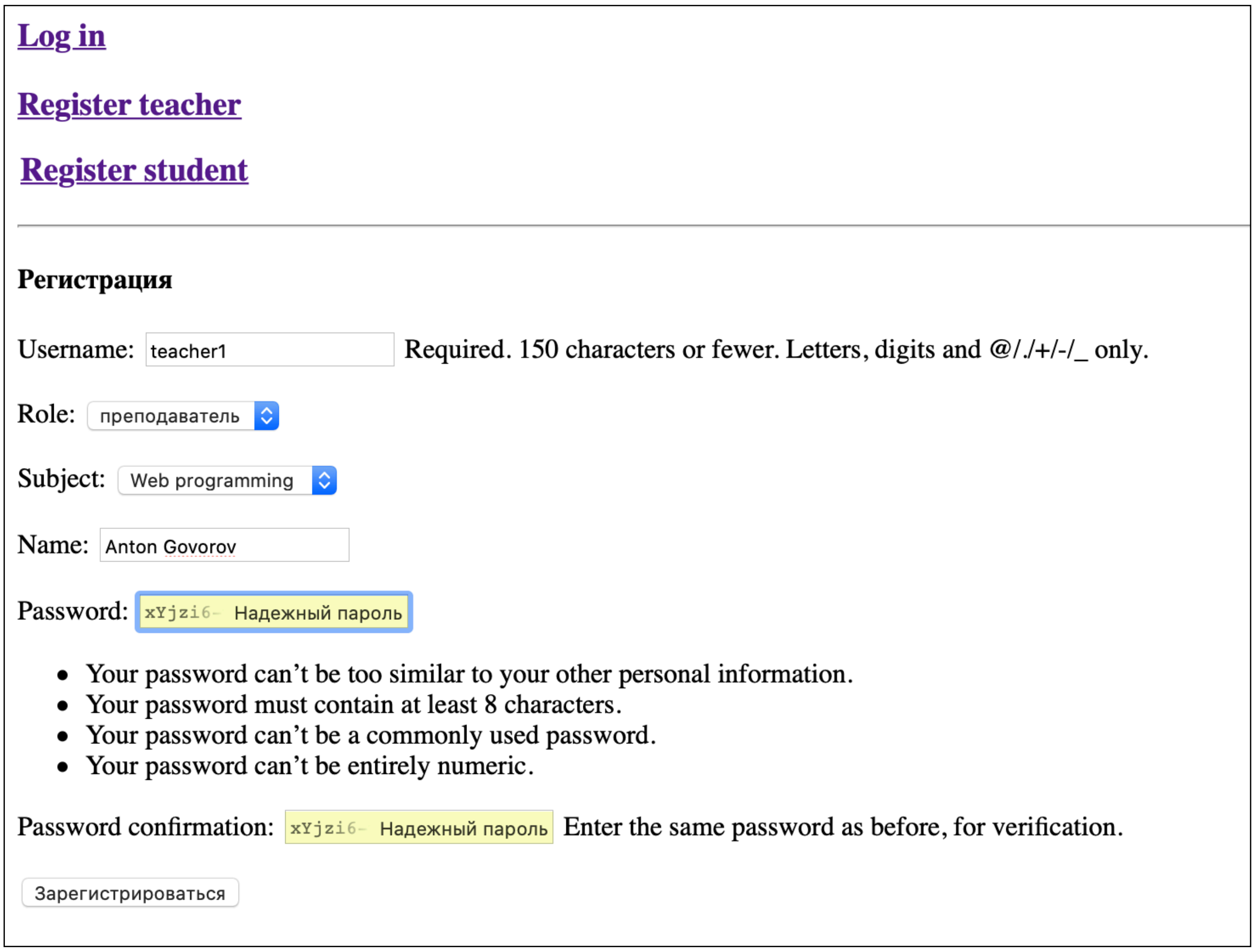
Log in to your account

Username:

Password:

Log in

http://127.0.0.1:8000/register/teacher



Log in

Register teacher

Register student

Регистрация

Username: teacher1 Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Role: преподаватель

Subject: Web programming

Name: Anton Govorov

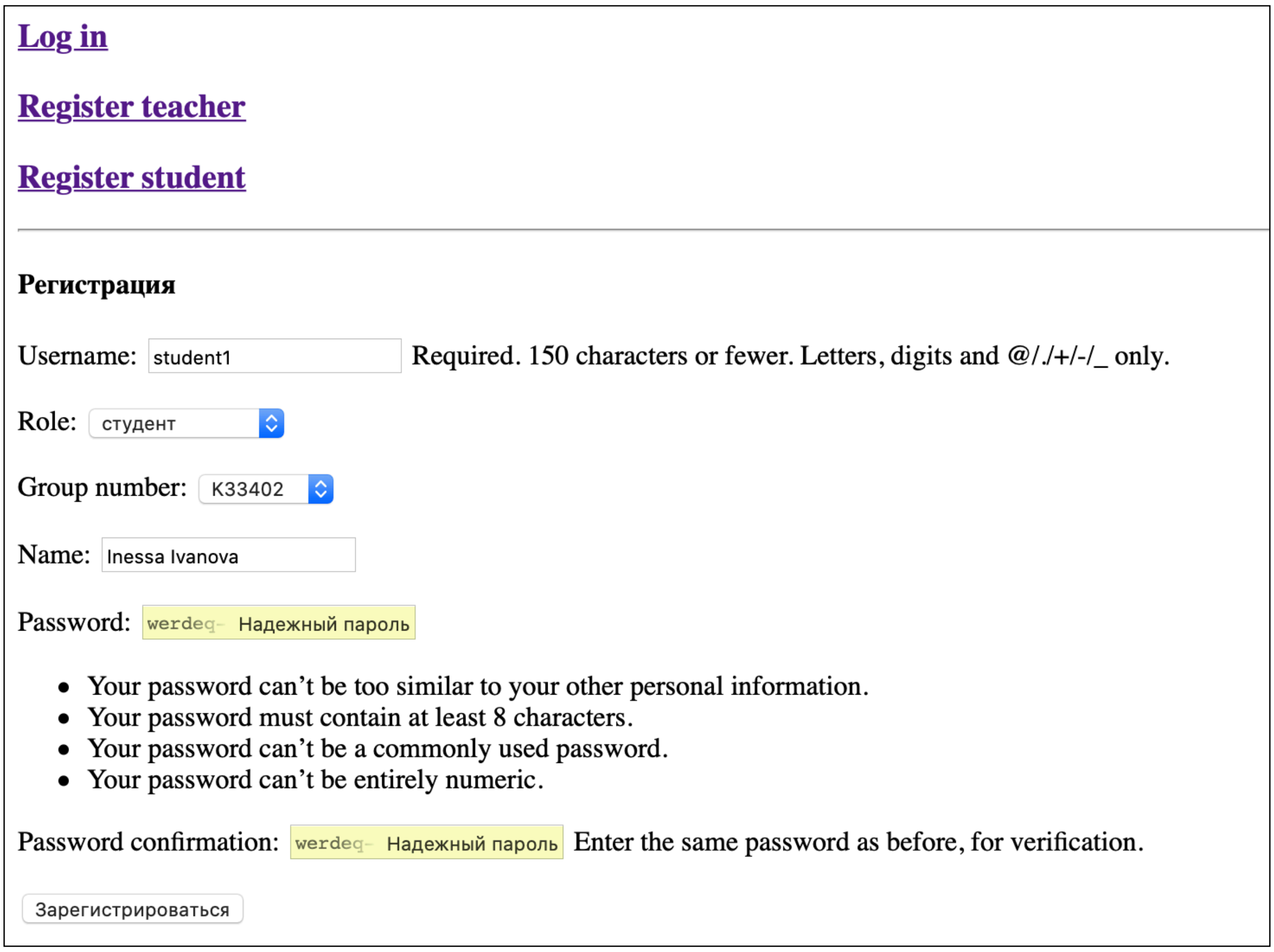
Password: xYjzi6- Надежный пароль

- Your password can’t be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can’t be a commonly used password.
- Your password can’t be entirely numeric.

Password confirmation: xYjzi6- Надежный пароль Enter the same password as before, for verification.

Зарегистрироваться

http://127.0.0.1:8000/register/student



Log in

Register teacher

Register student

Регистрация

Username: student1 Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Role: студент

Group number: K33402

Name: Inessa Ivanova

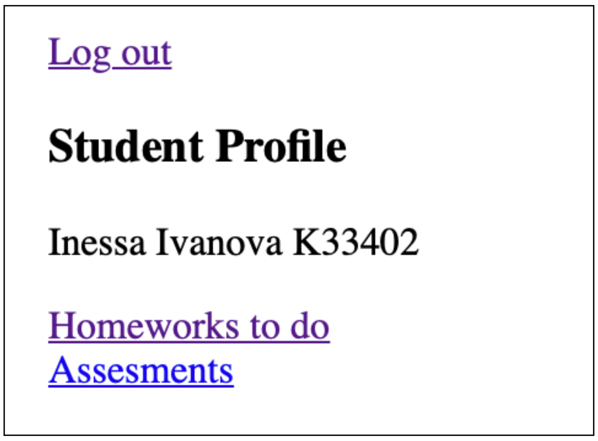
Password: werdeq Надежный пароль

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation: werdeq Надежный пароль Enter the same password as before, for verification.

Зарегистрироваться

http://127.0.0.1:8000/profile/

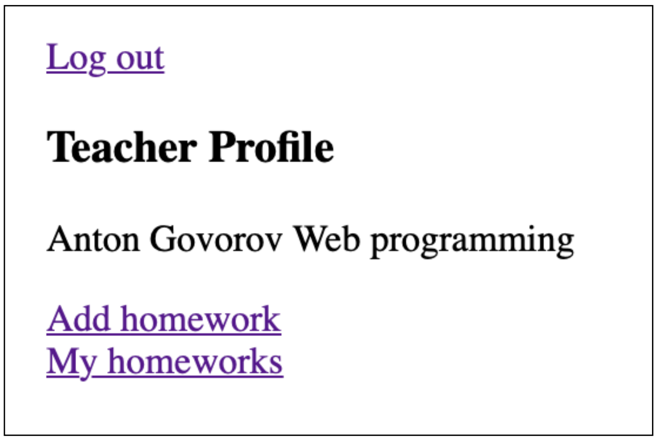


Log out

Student Profile

Inessa Ivanova K33402

Homeworks to do
Assesments



Log out

Teacher Profile

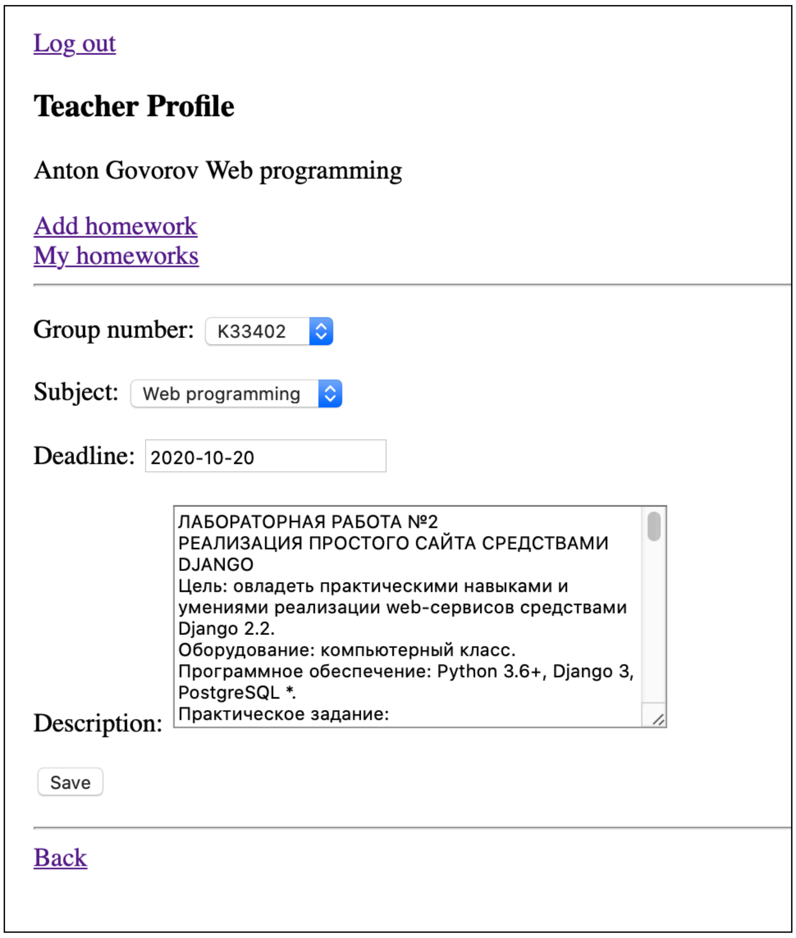
Anton Govorov Web programming

Add homework
My homeworks

2) Добавим следующие модели

```python
class Homework(models.Model):
    group_number = models.CharField(max_length=6, choices=group_numbers)
    subject = models.CharField(max_length=50, choices=subjects)
    teacher = models.ForeignKey(Teacher, on_delete=models.CASCADE)
    deadline = models.DateField()
    description = models.TextField(max_length=10000)

    def __str__(self):
        return '{} {}'.format(self.group_number, self.subject)


class Assignment(models.Model):
    student = models.ForeignKey(Student, on_delete=models.CASCADE)
    homework = models.ForeignKey(Homework, on_delete=models.CASCADE)
    solution = models.TextField(max_length=10000)

    def __str__(self):
        return '{} - {}'.format(self.student, self.homework.subject)


class Assessment(models.Model):
    student = models.ForeignKey(Student, on_delete=models.CASCADE)
    homework = models.ForeignKey(Homework, on_delete=models.CASCADE)
    grade = models.CharField(max_length=2)

    def __str__(self):
        return '{} - {}'.format(self.student, self.homework.subject)
```

3) Добавим возможность добавить ДЗ


Log out

Teacher Profile

Anton Govorov Web programming

Add homework
My homeworks

Group number: K33402

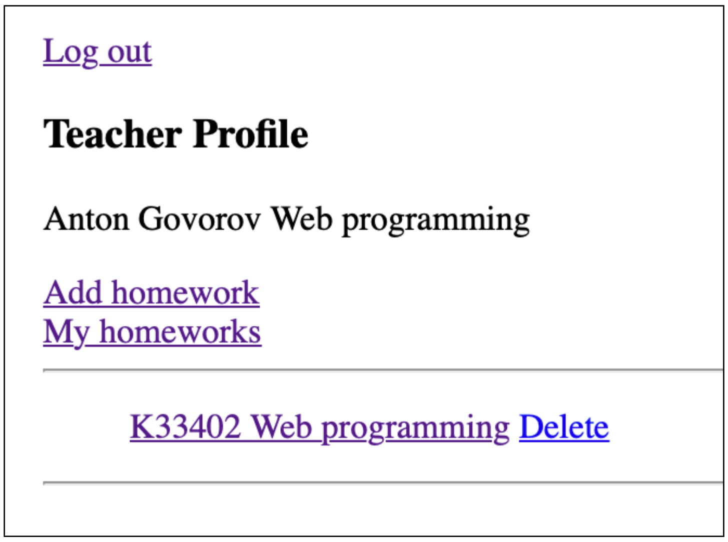
Subject: Web programming

Deadline: 2020-10-20

Description:
ЛАБОРАТОРНАЯ РАБОТА №2
РЕАЛИЗАЦИЯ ПРОСТОГО САЙТА СРЕДСТВАМИ DJANGO
Цель: овладеть практическими навыками и умениями реализации web-сервисов средствами Django 2.2.
Оборудование: компьютерный класс.
Программное обеспечение: Python 3.6+, Django 3, PostgreSQL *.
Практическое задание:

Save

Back


Log out

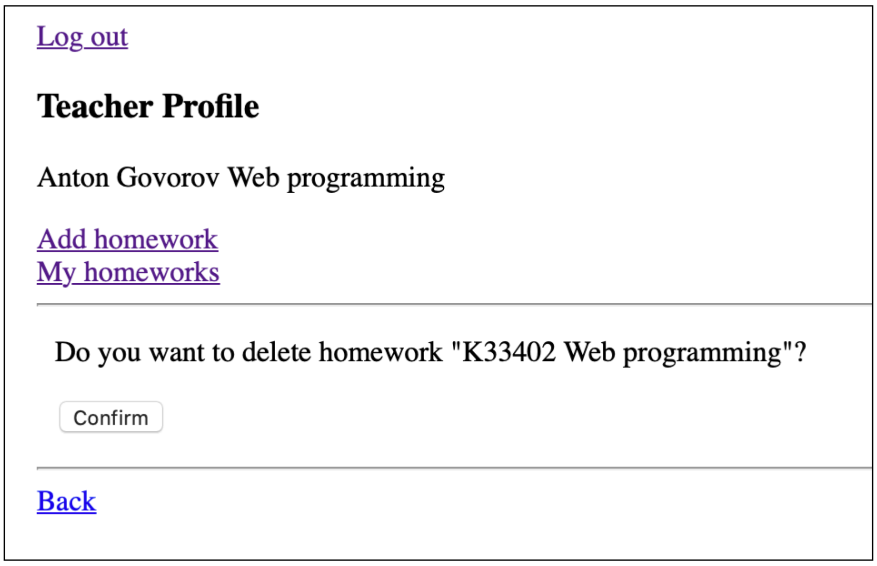
Teacher Profile

Anton Govorov Web programming

Add homework
My homeworks

K33402 Web programming Delete

4) Добавим возможность удалить ДЗ


Log out

Teacher Profile

Anton Govorov Web programming

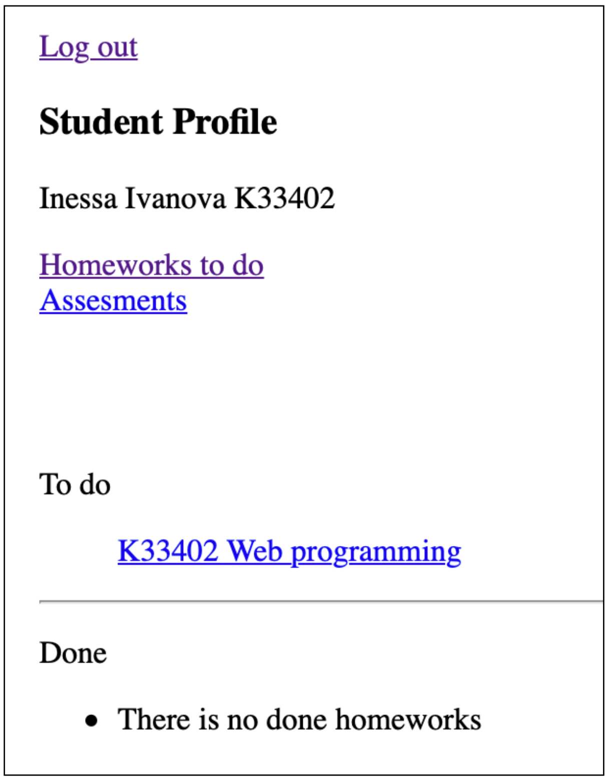
Add homework
My homeworks

Do you want to delete homework "K33402 Web programming"?

Confirm

Back

5) В профиль студента добавим список ДЗ


Log out

Student Profile

Inessa Ivanova K33402

Homeworks to do
Assesments

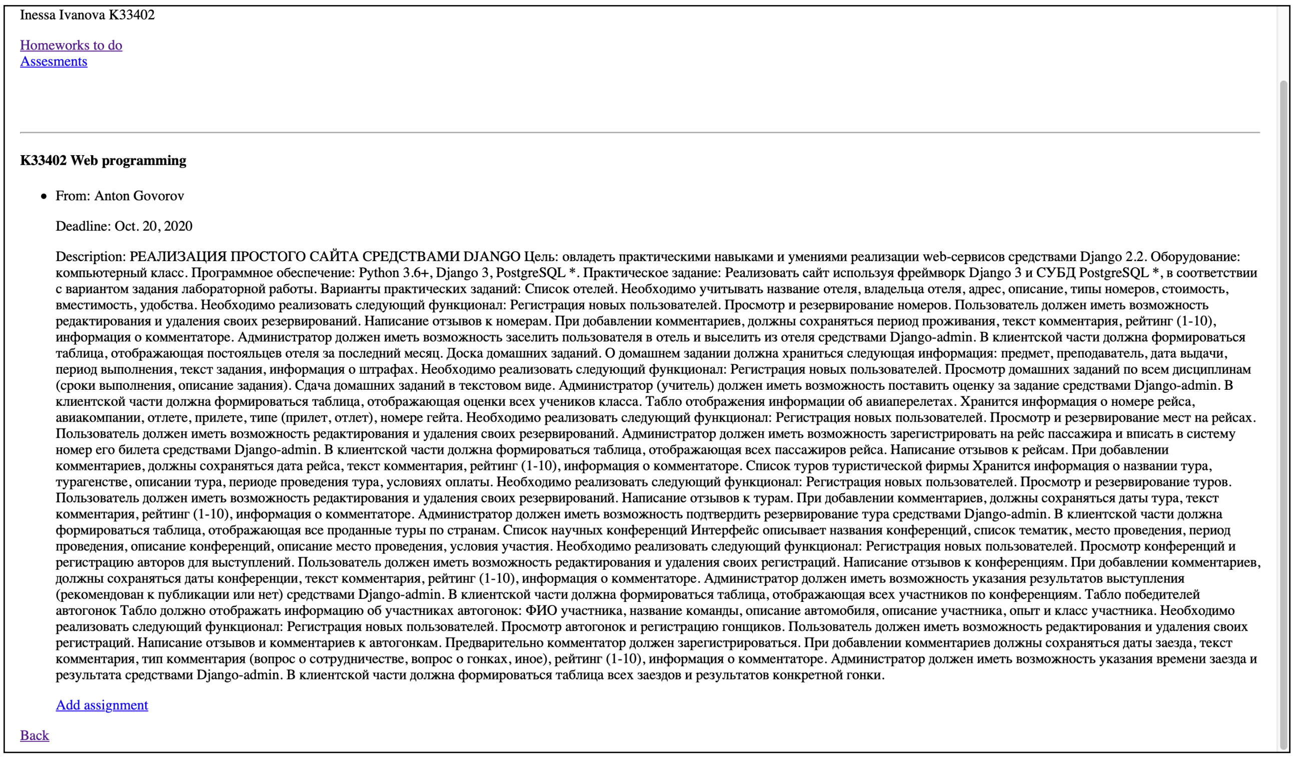
To do

K33402 Web programming

Done

- There is no done homeworks

6) При нажатии на ДЗ видим подробную информацию и возможность добавить ответ



Inessa Ivanova K33402

Homeworks to do
Assesments

K33402 Web programming

- From: Anton Govorov

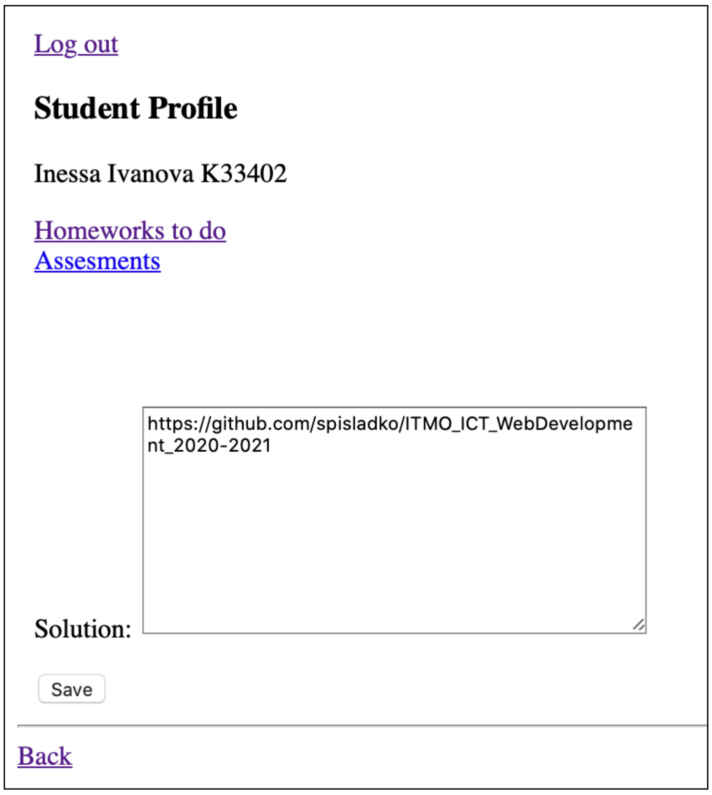
  Deadline: Oct. 20, 2020

  Description: РЕАЛИЗАЦИЯ ПРОСТОГО САЙТА СРЕДСТВАМИ DJANGO Цель: овладеть практическими навыками и умениями реализации web-сервисов средствами Django 2.2. Оборудование: компьютерный класс. Программное обеспечение: Python 3.6+, Django 3, PostgreSQL *. Практическое задание: Реализовать сайт используя фреймворк Django 3 и СУБД PostgreSQL *, в соответствии с вариантом задания лабораторной работы. Варианты практических заданий: Список отелей. Необходимо учитывать название отеля, владельца отеля, адрес, описание, типы номеров, стоимость, вместимость, удобства. Необходимо реализовать следующий функционал: Регистрация новых пользователей. Просмотр и резервирование номеров. Пользователь должен иметь возможность редактирования и удаления своих резервирований. Написание отзывов к номерам. При добавлении комментариев, должны сохраняться период проживания, текст комментария, рейтинг (1-10), информация о комментаторе. Администратор должен иметь возможность заселить пользователя в отель и выселить из отеля средствами Django-admin. В клиентской части должна формироваться таблица, отображающая постояльцев отеля за последний месяц. Доска домашних заданий. О домашнем задании должна храниться следующая информация: предмет, преподаватель, дата выдачи, период выполнения, текст задания, информация о штрафах. Необходимо реализовать следующий функционал: Регистрация новых пользователей. Просмотр домашних заданий по всем дисциплинам (сроки выполнения, описание задания). Сдача домашних заданий в текстовом виде. Администратор (учитель) должен иметь возможность поставить оценку за задание средствами Django-admin. В клиентской части должна формироваться таблица, отображающая оценки всех учеников класса. 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. Список туров туристической фирмы Хранится информация о названии тура, турагенстве, описании тура, периоде проведения тура, условиях оплаты. Необходимо реализовать следующий функционал: Регистрация новых пользователей. Просмотр и резервирование туров. Пользователь должен иметь возможность редактирования и удаления своих резервирований. Написание отзывов к турам. При добавлении комментариев, должны сохраняться даты тура, текст комментария, рейтинг (1-10), информация о комментаторе. Администратор должен иметь возможность подтвердить резервирование тура средствами Django-admin. В клиентской части должна формироваться таблица, отображающая все проданные туры по странам. Список научных конференций Интерфейс описывает названия конференций, список тематик, место проведения, период проведения, описание конференций, описание место проведения, условия участия. Необходимо реализовать следующий функционал: Регистрация новых пользователей. Просмотр конференций и регистрацию авторов для выступлений. Пользователь должен иметь возможность редактирования и удаления своих регистраций. Написание отзывов к конференциям. При добавлении комментариев, должны сохраняться даты конференции, текст комментария, рейтинг (1-10), информация о комментаторе. Администратор должен иметь возможность указания результатов выступления (рекомендован к публикации или нет) средствами Django-admin. В клиентской части должна формироваться таблица, отображающая всех участников по конференциям. Табло победителей автогонок Табло должно отображать информацию об участниках автогонок: ФИО участника, название команды, описание автомобиля, описание участника, опыт и класс участника. Необходимо реализовать следующий функционал: Регистрация новых пользователей. Просмотр автогонок и регистрацию гонщиков. Пользователь должен иметь возможность редактирования и удаления своих регистраций. Написание отзывов и комментариев к автогонкам. Предварительно комментатор должен зарегистрироваться. При добавлении комментариев должны сохраняться даты заезда, текст комментария, тип комментария (вопрос о сотрудничестве, вопрос о гонках, иное), рейтинг (1-10), информация о комментаторе. Администратор должен иметь возможность указания времени заезда и результата средствами Django-admin. В клиентской части должна формироваться таблица всех заездов и результатов конкретной гонки.

  Add assignment

Back

7) Добавление ответа



Log out

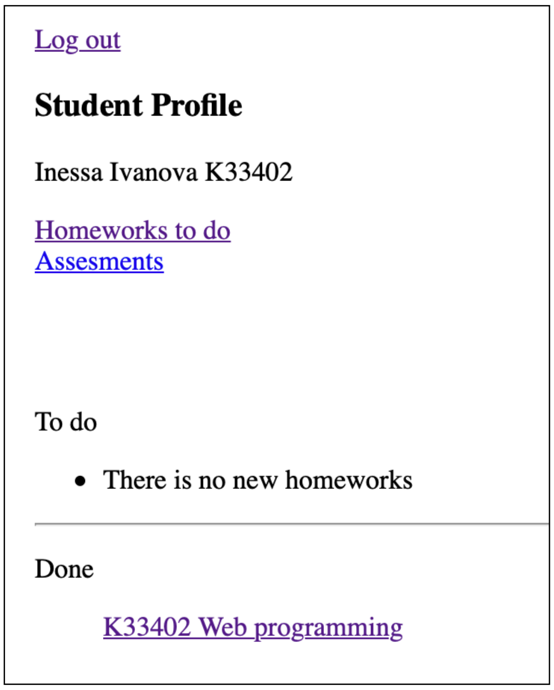
Student Profile

Inessa Ivanova K33402

Homeworks to do
Assesments

Solution: https://github.com/spisladko/ITMO_ICT_WebDevelopment_2020-2021

Back


Log out

Student Profile

Inessa Ivanova K33402

Homeworks to do
Assesments

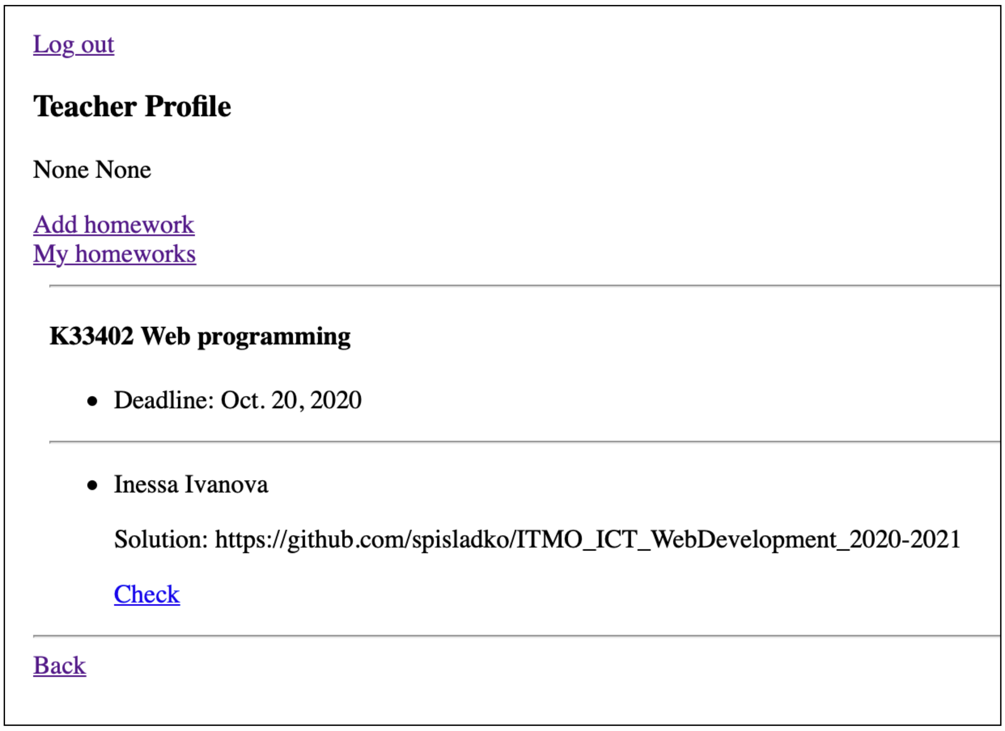
To do

- There is no new homeworks

Done

K33402 Web programming

8) Преподаватель видит решение


Log out

Teacher Profile

None None

Add homework
My homeworks

K33402 Web programming

- Deadline: Oct. 20, 2020

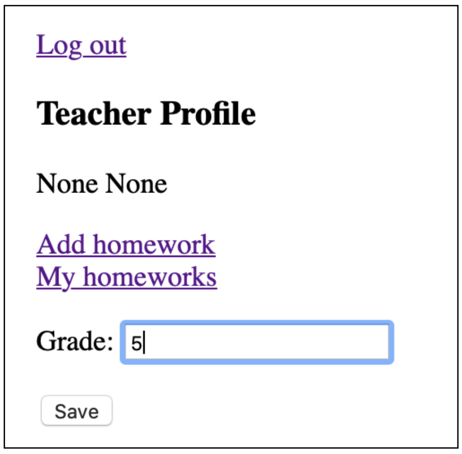
- Inessa Ivanova

  Solution: https://github.com/spisladko/ITMO_ICT_WebDevelopment_2020-2021

  Check

Back

9) И может его оценить


Log out

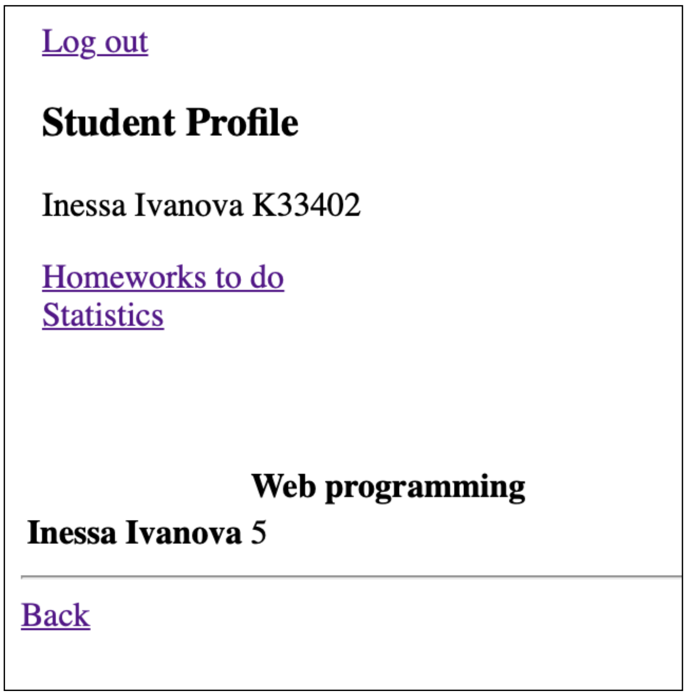
Teacher Profile

None None

Add homework
My homeworks

Grade: 5

10) Статистика оценок



Log out

Student Profile

Inessa Ivanova K33402

Homeworks to do
Statistics

Web programming

Inessa Ivanova 5

Back