МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И

МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

«МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

СВЯЗИ И ИНФОРМАТИКИ»

Кафедра «Сети связи и системы коммутации»

Лабораторная работа №6

по дисциплине «Web-программирование»

Выполнил студент

группы БФИ1901

Кочеринский Н.В.

Проверил:

Москва, 2021

**Оглавление**

[**1 Задание на лабораторную работу** 3](#_Toc89170066)

[**2 Ход лабораторной работы.** 3](#_Toc89170067)

[2.1 Создание формы, шаблона и представления для регистрации 3](#_Toc89170068)

[2.2 Создание формы, шаблона и представления для авторизации 5](#_Toc89170069)

[**Список используемых источников** 10](#_Toc89170070)

# **1 Задание на лабораторную работу**

1. В директорию lab6 скопируйте ваш прошлый проект blog.
2. Создайте шаблон и настройте адрес для отображения формы регистрации;
3. Создайте представление, которое будет обрабатывать поступающие запросы и регистрировать новых пользователей. Не забудьте сделать проверку на то, что отправленные поля не являются пустыми, а введенное имя пользователя уникально;
4. Создайте стили, подключив CSS-файл к шаблону;
5. Добавьте в шапку страницы всех записей и страницы для определенных статей, ссылку на регистрацию в верхнем правом углу (стиль ссылки сделать такой же, как у ссылки “Все статьи” на собственных страницах постов в предыдущих работах).
6. Создайте шаблон и настройте адрес для отображения формы авторизации;
7. Создайте представление, которое будет обрабатывать поступающие запросы и авторизовывать зарегистрированных пользователей. Не забудьте сделать проверку на то, что отправленные поля не являются пустыми, а введенные имя пользователя и пароль соответствуют одному из зарегистрированных аккаунтов;
8. Создайте стили, подключив CSS-файл к шаблону;
9. Загрузите ваш проект на любой гит-репозиторий (GitHub, GitLab, Google Code, Bitbucket и т.п.).

# **2 Ход лабораторной работы.**

## 2.1 Создание формы, шаблона и представления для регистрации

Создаём новую директорию для лабораторной работы №6 и копируем в неё проект blog из прошлой лабораторной работы. Добавим в файл urls.py новые пути.

urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', views.archive, name='home'),  
 path('article/<int:article\_id>', views.get\_article, name='get\_article'),  
 path('article/new/', views.create\_post, name='create\_post'),  
 path('registration', views.registred, name='regPage'),  
 path('logIn', views.logIn, name='logIn'),  
 path('logOut', views.logoutFunc, name='logOut')  
]

Создадим в файле views.py новую функцию registred , которая должна будет отображать форму с регистрацией пользователя, и переводить его на форму авторизации в случае успешной регистрации.

def registred(request):  
 if request.method == "POST":  
 form = {  
 'username': request.POST["username"],  
 'email': request.POST["email"],  
 'password': request.POST['password']  
 }  
 if form["username"] and form['email'] and form['password']:  
 try:  
 User.objects.get(username=request.POST["username"])  
 form['errors'] = u"Пользователь с таким именем уже существует"  
 return render(request, 'registrationpage.html', {'form': form})  
 except User.DoesNotExist:  
 User.objects.create\_user(username=request.POST["username"],  
 email=request.POST["username"],  
 password=request.POST['password'])  
 return redirect('logIn')  
 else:  
 form['errors'] = u"Не все поля заполнены"  
 return render(request, 'registrationpage.html', {'form': form})  
 else:  
 return render(request, 'registrationpage.html', {})

Далее создадим файл registrationpage.html, программный код которого представлен на листинге 1.

Листинг 1 - registrationpage.html

<!DOCTYPE html>  
<html>  
 <head>  
 <title>Регистрация</title>  
 <link rel="stylesheet" href="{{ STATIC\_URL }}/static/article.css"/>  
 </head>  
 <body>  
 <div class="content">  
 <h1>Страница регистрации</h1>  
 <form method="POST">  
 {% csrf\_token %}  
 <input  
 type="text"  
 name="username"  
 placeholder="Имя пользователя"  
 value="{{ form.title }}"  
 />  
 <input  
 type="text"  
 name="email"  
 placeholder="email пользователя"  
 value="{{ form.title }}"  
 />  
 <input  
 type="password"  
 name="password"  
 placeholder="Пароль"  
 value="{{ form.title }}"  
 />  
 <input type="submit" value="Сохранить" />  
 </form>  
 {{ form.errors }}  
 </div>  
 </body>  
</html>

## 2.2 Создание формы, шаблона и представления для авторизации

Затем добавим функцию logIn, которая будет выполнять авторизацию пользователя.

def logIn(request):  
 if request.method == "POST":  
 form = {  
 'username': request.POST["username"],  
 'password': request.POST['password']  
 }  
 if form["username"] and form["password"]:  
 user = authenticate(request, username=request.POST["username"], password=request.POST["password"])  
 if user is not None:  
 login(request, user)  
 return redirect('home')  
 else:  
 form['errors'] = u"Введеный пользователь не существует"  
 return render(request, 'singInPage.html', {'form': form})  
 else:  
 form['errors'] = u"Не все поля заполнены"  
 return render(request, 'singInPage.html', {'form': form})  
 else:  
 return render(request, 'singInPage.html', {})

Далее создадим файл signinpage.html, программный код которого представлен на листинге 2.

Листинг 2 – singinpage.html

<!DOCTYPE html>  
<html>  
 <head>  
 <title>Авторизация</title>  
 <link rel="stylesheet" href="{{ STATIC\_URL }}/static/article.css" />  
 </head>  
 <body>  
 <div class="content">  
 <h1>Страница Авторизации</h1>  
 <form method="POST">  
 {% csrf\_token %}  
 <input  
 type="text"  
 name="username"  
 placeholder="Имя пользователя"  
 value="{{ form.title }}"  
 />  
 <input  
 type="password"  
 name="password"  
 placeholder="Пароль"  
 value="{{ form.title }}"  
 />  
 <input type="submit" value="Сохранить" />  
 </form>  
 {{ form.errors }}  
 </div>  
 </body>

На листинге 3 представлен программный код файла article.css со стилями для страниц.

Листинг 3 - article.css

body{  
 background: #1abc9c;  
 font-family: Tahoma, Arial, sans-serif;  
 color: #ffffff  
}  
  
img {  
 display: block;  
 width: 318px;  
 margin-left: auto;  
 margin-right: auto;  
}  
  
.archive {  
 width: 960px;  
 margin-left: auto;  
 margin-right: auto;  
}  
  
.post-title a {  
 color: #ffffff;  
}  
  
.article-author {  
 width: 50%;  
 float: left;  
}  
  
.article-created-date {  
 text-align: right;  
}  
  
.article-image{  
 display: block;  
 width: 318px;  
 margin-left: 0;  
}  
  
  
.article-border p{  
 text-align: right;  
}  
  
.article-text{  
 width: 960px;  
 text-align: justify;  
}  
.article-created-data{  
 text-align: right;  
}  
  
.content{  
 text-align: center;  
 padding-top: 70px;  
}  
  
input[name="title"]{  
 padding: 5px;  
 margin-bottom: 10px;  
 border: 1px solid #888;  
 outline: none;  
 -moz-appearance: none;  
 width: 200px;  
 text-align: center;  
 border-radius: 40px;  
}  
  
textarea[name="text"]{  
 padding: 25px;  
 margin-bottom: 10px;  
 border: 1px solid #888;  
 outline: none;  
 -moz-appearance: none;  
 width: 650px;  
 height: 350px;  
 resize: none;  
 border-radius: 40px;  
 scrollbar-width: thin;  
}  
  
.create{  
 display: flex;  
 flex-direction: column;  
 align-items: center;  
}  
  
.save\_button{  
 padding: 10px;  
 width: 150px;  
 background-color: white;  
 border: none;  
 border-radius: 40px;  
 color: #1abc9c;  
 font-weight: bold;  
 letter-spacing: 0.06em;  
 margin-top: 10px;  
}  
  
.save\_button:hover{  
 color: white;  
 background-color: #1abc9c;  
 box-shadow: 1px 1px 10px 10px;  
 transition-duration: 0.3s;  
}  
  
.linkButton{  
padding-bottom: 10px;  
}  
  
  
.link {  
 color: white;  
 font-weight: bold;  
}  
  
.buttons{  
 list-style-type: none;  
 margin-left: 422px;  
}

На рисунках 1-2 представлены страницы авторизации и регистрации

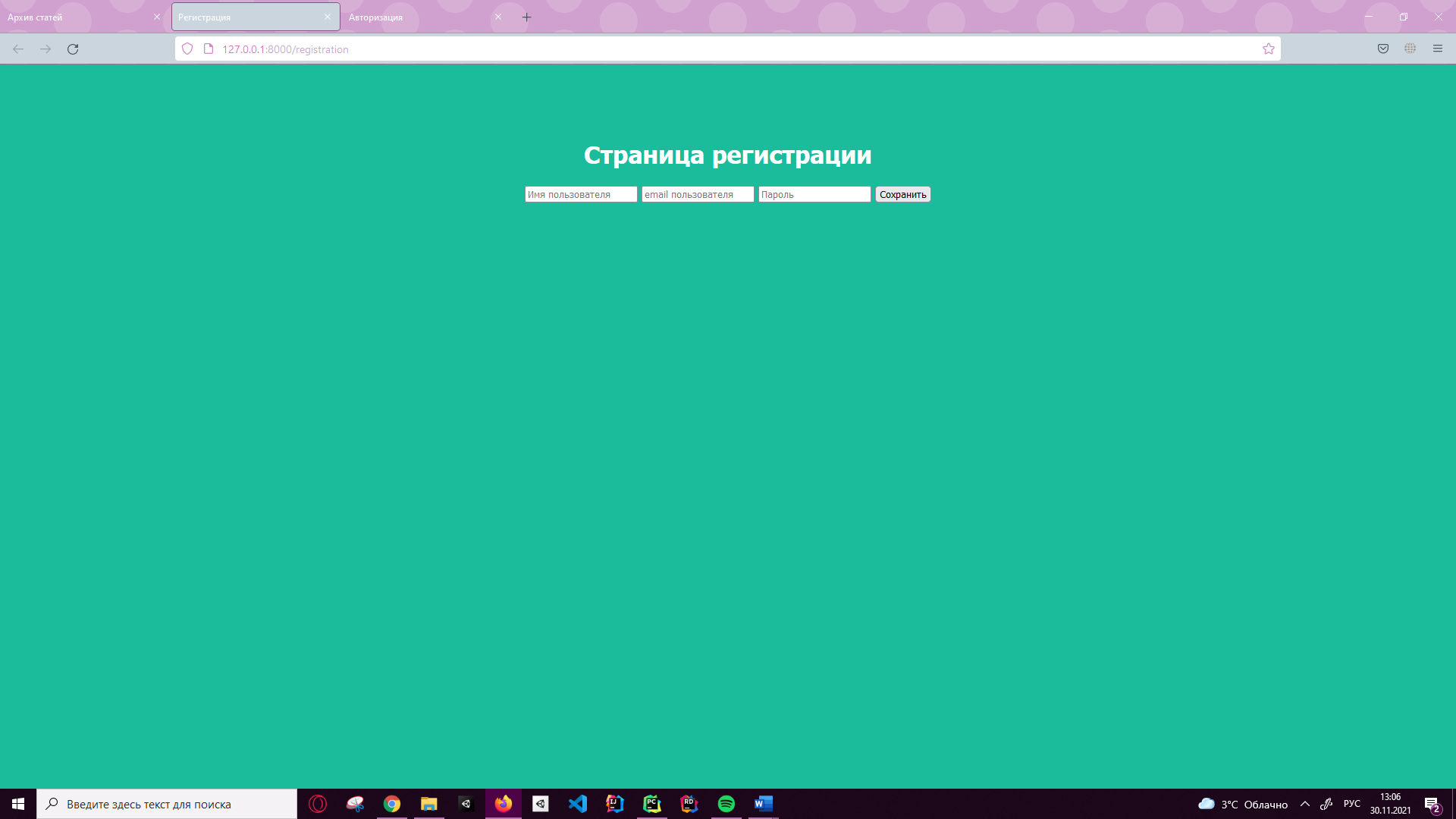


Рисунок 1 – Страница регистрации

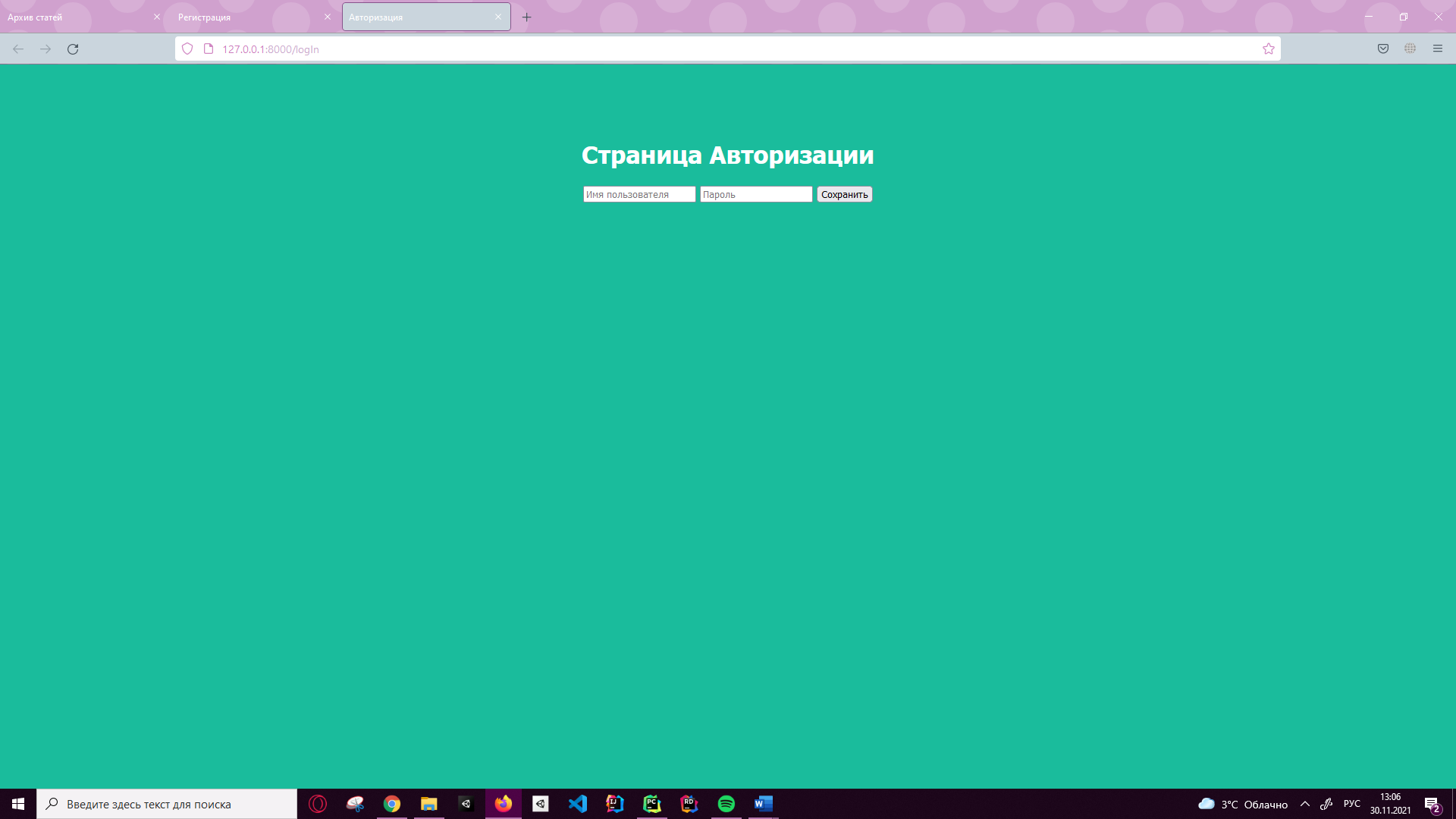


Рисунок 2 – Страница авторизации

Завершением лабораторной работы служит загрузка проекта на удаленный репозиторий.

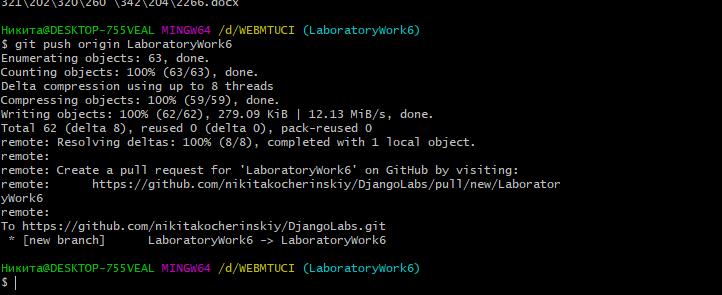


Рисунок 3 – Загрузка проекта

**Список используемых источников**

1 ГОСТ 7.32-2017 СИБИД. Отчет о научно-исследовательской работе. Структура и правила оформления.

2 ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание.