



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ (ИУ)

КАФЕДРА _____ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ (ИУ5)

О Т Ч Е Т

по домашнему заданию

по дисциплине: Разработка интернет-приложений

на тему: веб-приложение на основе технологий Python, Django, JavaScript, MySQL

Студент ИУ5-53
(Группа)

(Подпись, дата)

А.С. Волков
(И.О.Фамилия)

Руководитель

(Подпись, дата)

Ю.Е. Гапанюк
(И.О.Фамилия)

2019 г.

1. Задание и порядок выполнения

В рамках домашнего задания студент разрабатывает веб-приложение на основе технологий Python, Django, JavaScript, MySQL.

Предметная область домашнего задания содержит субъект, отношение и объект. Необходимые атрибуты субъекта и объекта выбираются студентом самостоятельно. Предметная область (по вариантам) представлена в следующей таблице.

№	Субъект	Отношение	Объект
5	Пользователь	Отзыв	Книга

2. Исходные коды

2.1. views.py

```
from django.shortcuts import render, HttpResponseRedirect
from django.urls import reverse
from django.views.generic import View, ListView
from .form import *
from .models import *
from .registration import *
from django.contrib.auth.models import User
from django.contrib import auth
from django.contrib.auth import authenticate
from django.contrib.auth.decorators import login_required

def main(request):
    return render(request, 'main.html', locals())

def db(request):
    return render(request, 'db.html', locals())

def book_id(request, id):
    book = Book.objects.filter(id_book=id)
    book = book.first()
    review_filter = Issuance.objects.filter(book_name_id=id)
    review = list(review_filter)
    id = request.GET.get('id')
    if id:
        queryset = Book.objects.filter(id=id)
    return render(request, 'book_id.html', locals())

class ReaderView(ListView):
    model = User
    template_name = 'reader.html'
```

```

def books(request):
    form = BookForm(request.POST, request.FILES or None)
    if request.POST and form.is_valid():
        instance = form.save(commit=False)
        instance.save()
        return HttpResponseRedirect(reverse('book_id_url', args=(instance.pk,)))
    return render(request, 'books.html', locals())

def registration(request):
    errors = {'username': '', 'password': '', 'password2': '', 'email': '',
'firstname': '', 'surname': ''}
    error_flag = False
    if request.method == 'POST':
        username = request.POST.get('username')
        if not username:
            errors['username'] = 'Введите логин'
            error_flag = True
        elif len(username) < 5:
            errors['username'] = 'Логин должен превышать 5 символов'
            error_flag = True
        elif User.objects.filter(username=username).exists():
            errors['username'] = 'Такой логин уже существует'
            error_flag = True
        password = request.POST.get('password')
        if not password:
            errors['password'] = 'Введите пароль'
            error_flag = True
        elif len(password) < 8:
            errors['password'] = 'Длина пароля должна превышать 8 символов'
        password_repeat = request.POST.get('password2')
        if password != password_repeat:
            errors['password2'] = 'Пароли должны совпадать'
            error_flag = True
        email = request.POST.get('email')
        if not email:
            errors['email'] = 'Введите e-mail'
        firstname = request.POST.get('firstname')
        if not firstname:
            errors['firstname'] = 'Введите имя'
        surname = request.POST.get('surname')
        if not surname:
            errors['surname'] = 'Введите фамилию'
        if not error_flag:
            user = User.objects.create_user(username=username, password=password,
email=email, first_name=firstname,
last_name=surname)
            return HttpResponseRedirect('/login/')
    return render(request, 'registration.html', locals())

def login(request):
    error = ""
    username = None
    password = None
    if request.method == 'POST':
        username = request.POST.get('username')
        password = request.POST.get('password')

```

```

        user = authenticate(username=username, password=password)
        if user:
            auth.login(request, user)
            return HttpResponseRedirect('/success/')
        else:
            error = "Попробуй ещё раз"
        return render(request, 'login.html', locals())

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

def logout(request):
    auth.logout(request)
    return HttpResponseRedirect('/main/')

def registration2(request):
    form = RegistrationForm(request.POST or None)
    if request.method == 'POST':
        if form.is_valid():
            user = User.objects.create_user(username=request.POST.get('username'),
                                             email=request.POST.get('email'),
                                             password=request.POST.get('password'),
                                             first_name=request.POST.get('firstname'),
                                             last_name=request.POST.get('surname'))
            return HttpResponseRedirect('/login/')
    return render(request, 'registration2.html', {'form': form})

def ListBookView(request):
    book_model = Book.objects.all()
    book_list = list(book_model)
    # template_name = 'list_books.html'
    form = BookForm(request.POST, request.FILES or None)
    if request.POST and form.is_valid():
        instance = form.save(commit=False)
        instance.save()
        return HttpResponseRedirect(reverse('book_id_url', args=(instance.pk,)))
    return render(request, 'list_books.html', locals())

def review_add(request):
    form = ReviewForm(request.POST, request.FILES or None)
    if request.POST and form.is_valid():
        instance = form.save(commit=False)
        instance.save()
        return HttpResponseRedirect(reverse('book_id_url', args=(instance.pk,)))
    return render(request, 'books.html', locals())

```

2.2. models.py

```

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

class Book(models.Model):
    id_book = models.AutoField(primary_key=True)
    name_book = models.CharField(max_length=100, verbose_name='Название книги')

```

```

        genre_book = models.CharField(max_length=50, default='Жанр неизвестен',
verbose_name='Жанр')
        photo = models.ImageField(null=True, blank=True, default='no-image.png',
verbose_name='Фото')
        writer_book = models.CharField(max_length=100, verbose_name='Автор')
        year_book = models.IntegerField(verbose_name='Год создания')
        users = models.ManyToManyField(User, through='Issuance',
through_fields=('book_name', 'username'))
        # reviews = models.ForeignKey(Issuance, on_delete=models.CASCADE)
        def str(self):
            return '%s, %s' % (str(self.id_book), str(self.name_book))

        def get_users(self):
            return [{'name': user.username} for user in self.users.all()]

        class Meta:
            verbose_name_plural = "Книги"
            verbose_name = "Название книги"

class Issuance(models.Model):
    book_name = models.ForeignKey(Book, on_delete=models.CASCADE,
verbose_name='Название книги')
    username = models.ForeignKey(User, on_delete=models.CASCADE)
    review = models.CharField(max_length=255, default = 'null', verbose_name='Отзыв')
    def str(self):
        return 'username:%s, book_name:%s', 'review:%s' %
(str(self.username.username), str(self.book_name.name_book), str(self.review))
    # def get_reviews(book):
    #     return Issuance.objects.filter(book_name_id = Book.id_book)
    class Meta:
        verbose_name_plural = "Отзывы"
        verbose_name = "Название книги"
        verbose_name = "Отзыв"

```

2.3. base.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Форум читателей</title>
    {% load static %}
    <link rel="stylesheet" href="{% static 'Bootstrap/css/bootstrap.css'
%}?20200111">
    <link rel="stylesheet" href="{% static 'Bootstrap/css/font-awesome.min.css'
%}?20200111">
    <link type="text/css" rel="stylesheet" href="{% static 'Bootstrap/css/main.css'
%}?20200111">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
    <script type="text/javascript" src="{% static 'Bootstrap/js/bootstrap.min.js'
%}"></script>
    <script type="text/javascript" src="{% static 'Bootstrap/js/jquery-3.2.1.min.js'
%}"></script>
    <script type="text/javascript" src="{% static
'Bootstrap/js/alight_0.12.last.min.js' %}"></script>
</head>
<body>
    <div class="navbar navbar-inverse navbar-fixed-top">

```

```

<div class="container">
  <div class="navbar-header">
    <a class="navbar-brand" href="{% url 'main_url' %}">Форум
читателей</a>
  </div>
  <div class="navbar-collapse collapse">
    <ul class="nav navbar-nav navbar-right">
      {% if request.user.is_authenticated %}
        {% if request.user.is_staff %}
          <li><a href="/admin/">Администрирование </a></li>
            {% endif %}
        {% endif %}
        <li><a href="{% url 'list_books_url' %}">Книги</a></li>
        <li><a href="{% url 'reader_url' %}">Читатели</a></li>
        <li>
          <div class="dropdown">
            <button class="dropdown-toggle" data-toggle="dropdown">
              
            </button>
            <div class="dropdown-content">
              {% if request.user.is_authenticated %}
                <span>Здравствуй, {{ request.user.username
}}</span>
                <a href="{% url 'logout_url' %}">Выйти</a>
              {% else %}
                <a href="{% url 'login_url' %}">Вход</a>
                <a href="{% url 'registration2_url'
%}">Регистрация</a>
              {% endif %}
            </div>
          </div>
        </li>
      </ul>
    </div>
  </div>
</div>
{% block body1 %}{% endblock %}
</body>
</html>

```

2.4. list_books.html

```

{% extends 'base.html' %}
{% block body1 %}
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  <div id="myModal" class="modal fade" role="dialog">
    <div class="modal-dialog">
      <div class="modal-dialog" style="width: 400px">
        <div class="modal-content">
          <div class="modal-header">
            <button type="button" class="close" data-dismiss="modal"
aria-hidden="true">×</button>
            <h2 class="modal-title" style="font-size:
30px">Добавление новой книги</h2>
          </div>

```

```

<div class="modal-body">
  <form method="POST" enctype="multipart/form-data">
    <div class="add-form container">
      {% csrf_token %}
      <div class="course_add">
        <label for="id_name_book">Название
КНИГИ:</label>

        <div class="data-and-error">
          <input type="text" name="name_book"
maxlength="100" required
          id="id_name_book"/>
        </div>
      </div>
      <div class="course_add">
        <label for="id_genre_book">Жанр:</label>
        <div class="data-and-error">
          <input type="text" name="genre_book"
maxlength="100" required
          id="id_genre_book"/>
        </div>
      </div>
      <div class="course_add">
        <label for="id_photo">Фото:</label>
        <div class="data-and-error">
          <input type="file" name="photo"
id="id_photo"/>
        </div>
      </div>
      <div class="course_add">
        <label for="id_writer_book">Автор:</label>
        <div class="data-and-error">
          <input type="text" name="writer_book"
maxlength="100" required
          id="id_genre_book"/>
        </div>
      </div>
      <div class="course_add">
        <label for="id_year_book">Год
написания:</label>

        <div class="data-and-error">
          <input type="text" name="year_book"
maxlength="100" required
          id="id_year_book"/>
        </div>
      </div>
      <br><button type="submit" id="btn_add"
style="margin-left: 5%">Добавить книгу</button>
      <br>
    </div>
  </form>
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-default" data-
dismiss="modal">Заккрыть</button>
</div>
</div>
</div>
</div>
<div id="header1">

```

```

<div class="container">
  <div class="row centered">
    <div class="col-lg-8 col-lg-offset-2">
      <h1 style="font-size: 60px">Книги</h1>
      <div al-app al-init="count=5" @scroll.debounce-
50="($element.scrollTop > $element.scrollHeight - $element.clientHeight -
150)?count++:0" class="body">
        <div al-repeat="n in count">
          {% for Book in book_list %}
            <div class="infinite-container">
              <div class="infinite-item">
                <tr>
                  <td> <a href="{% url 'book_id_url'
Book.id_book%}">{{Book.name_book}} </a></td>
                  {% if Book.photo %}
                    <td><a href="{% url 'book_id_url' Book.id_book%}"></a></td>
                  {% endif %}
                </tr>
              </div>
            </div>
          {% endfor %}
        </div>
      </div>
    </div>
    <button type="button" class="btn ui-button-icon-primary" data-
toggle="modal" data-target="#myModal">Добавить книгу</button>
  </div>
</div>
{% endblock %}

```

2.5. book_id.html

```

{% extends 'base.html' %}

{% block body1 %}
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  <div id="myModal" class="modal fade" role="dialog">
    <div class="modal-dialog">
      <div class="modal-dialog" style="width: 400px">
        <div class="modal-content">
          <div class="modal-header">
            <button type="button" class="close" data-dismiss="modal"
aria-hidden="true">x</button>
            <h2 class="modal-title" style="font-size:
30px">Добавление нового отзыва</h2>
          </div>
          <div class="modal-body">
            <form method="POST" enctype="multipart/form-data">
              <div class="add-form container">
                {% csrf_token %}
                <div class="course_add">
                  <label for="book_name">Название
книги:</label>
                  <div class="data-and-error">
                    {{book.name_book}}

```



```

        </div>
    </div>
    <div class="course_add">
        <label for="username">Пользователь:</label>
        <div class="data-and-error">
            {{ request.user.username }}
        </div>
    </div>
    <div class="course_add">
        <label for="review">Отзыв:</label>
        <div class="data-and-error">
            <input type="text" name="review"
maxlength="255"/>
        </div>
    </div>
    <br><button type="submit" id="btn_add"
style="margin-left: 5%">Добавить отзыв</button>
    <br>
</div>
</form>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-default" data-
dismiss="modal">Закрыть</button>
</div>
</div>
</div>
</div>
<div id="header1">
    <div class="container">
        <div class="row centered">
            <div class="col-lg-8 col-lg-offset-2">
                <h1 style="font-size: 60px"><br>{{ book.name_book }}<br></h1>
                {% if book.photo %}
                    <td></td>
                    {% endif %}
            <table class="table">
                <thead class="thead">
                    <tr href="{% url 'list_books_url' %}">
                        <td>Название</td>
                        <td>Жанр</td>
                        <td>Автор</td>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>{{book.name_book}}</td>
                        <td>{{book.genre_book}}</td>
                        <td>{{book.writer_book}}</td>
                    </tr>
                </tbody>
            </table>
            <table class="table table-hover table-inverse">
                <tr>
                    <td>Читатель</td>
                    <td>Отзыв</td>
                </tr>

```

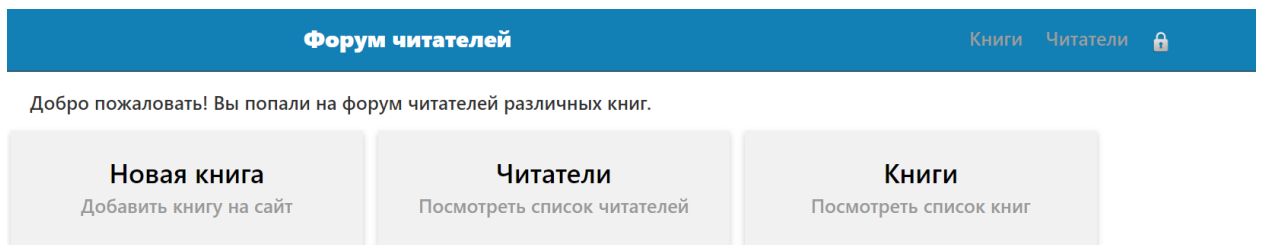
```

        {% for user in review %}
        <tr>
        <td>{{ user.username }}</td>
        <td>{{ user.review }}</td>
        </tr>
        {% endfor %}
    </table>
</div>
<button type="button" class="btn ui-button-icon-primary" data-
toggle="modal" data-target="#myModal">Добавить отзыв</button>
</div>
</div>
{% endblock %}

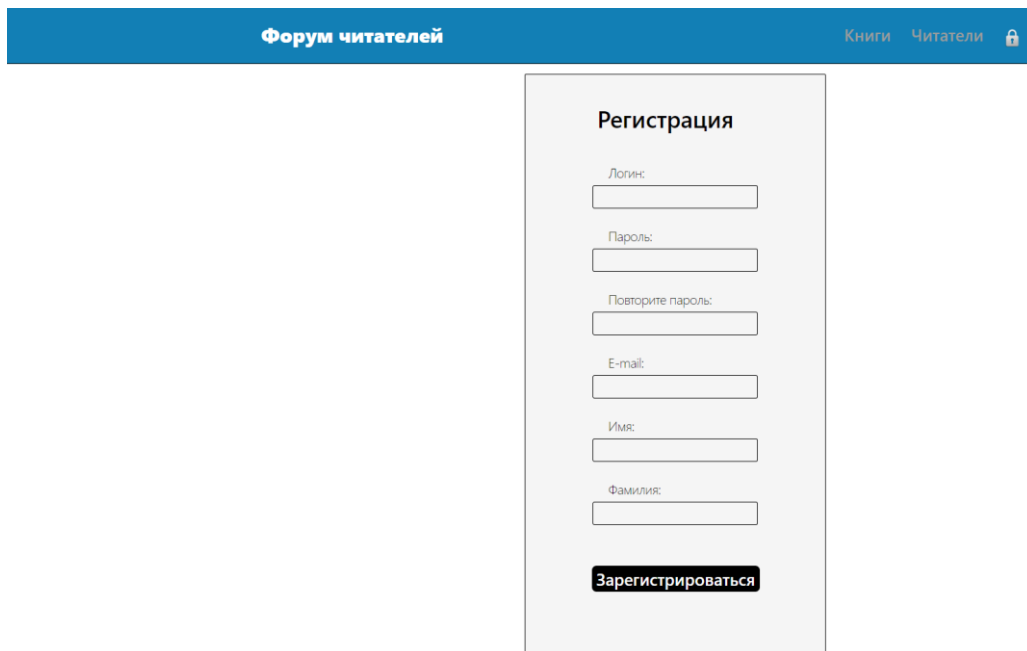
```

3. Скриншоты

3.1. Главная страница



3.2. Регистрация



3.3. Вход

Авторизация

Логин

Пароль

Войти

Форум читателей

Администрирование Книги Читатели

Добро пожаловать! Вы попали на форум читателей различных книг.

Здравствуйте, superadmin

Выйти

Новая книга

Добавить книгу на сайт

Читатели

Посмотреть список читателей

Книги

Посмотреть список книг

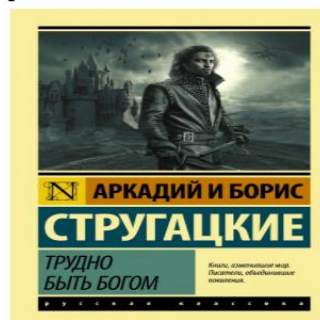
3.4. Отзыв

Форум читателей

Администрирование Книги Читатели

Добавить отзыв

Трудно быть богом



Название	Жанр	Автор
Трудно быть богом	Научная фантастика	А. и Б. Стругацкие
Читатель	Отзыв	
artyom	Очень вдохновляет!	
superadmin	Перечитываю постоянно!	
reader1	Вдохновляет! Bay!	

3.5. Администрирование

Django Администрация

WELCOME, SUPERADMIN | VIEW SITE / CHANGE PASSWORD / LOG OUT

Администрирование

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Change

Users

+ Add

Change

MY_APP

Книги

+ Add

Change

Отзывы

+ Add

Change

Recent actions

My actions

+ Issuance object (13)

Отзыв

+ Issuance object (12)

Отзыв

+ Issuance object (11)

Отзыв

+ Issuance object (10)

Отзыв

+ Issuance object (9)

Отзыв

+ Book object (1)

Название книги

+ Book object (5)

Название книги

+ Book object (4)

Название книги

+ Book object (3)

Название книги

+ Book object (2)

Название книги

Home | My App | Книги

Select Название книги to change

ADD НАЗВАНИЕ КНИГИ +

Q |

Search

Action:

0 of 5 selected

ID BOOK	НАЗВАНИЕ КНИГИ	ЖАНР	АВТОР
10	Полдень XXII век	Научная фантастика	А. и Б. Стругацкие
9	Справочник по БД	Научная литература	Резников Г.И.
8	Пикник на обочине	Научная фантастика	А. и Б. Стругацкие
7	Сказки	Рассказ	Волков
6	Трудно быть богом	Научная фантастика	А. и Б. Стругацкие

5 Книги

FILTER

By id book

All

6

7

8

9

10

By Название книги

All

Пикник на обочине

Полдень XXII век

Сказки

Справочник по БД

Трудно быть богом

By Жанр

All

Научная литература

Научная фантастика

Рассказ

By Автор

All

А. и Б. Стругацкие

Волков

Резников Г.И.

Home | My App | Отзывы | Add Отзыв

Add Отзыв

Название книги:

Book object (8)

+ +

Username:

artyom

+ +

Отзыв:

WOW

Select user to change

Q |

Search

Action:

0 of 3 selected

USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
artyom	artyom@gmail.com	Артём	Волков	0
reader1	reader@read.com	Иван	Иванов	0
superadmin	admin@admin.com			1

3 users