Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Санкт-Петербургский университет аэрокосмического приборостроения»

КАФЕДРА № 41

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доц., канд. техн. наук |  |  |  | Е.Л. Турнецкая |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ  Практическая работа № 4 |
| по курсу: ПРОГРАММНАЯ ИНЖЕНЕРИЯ |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | М021 |  | . |  | В. Я. Ломоносов  П. А. Носуленко |
|  |  |  | подпись, дата |  | инициалы, фамилия |

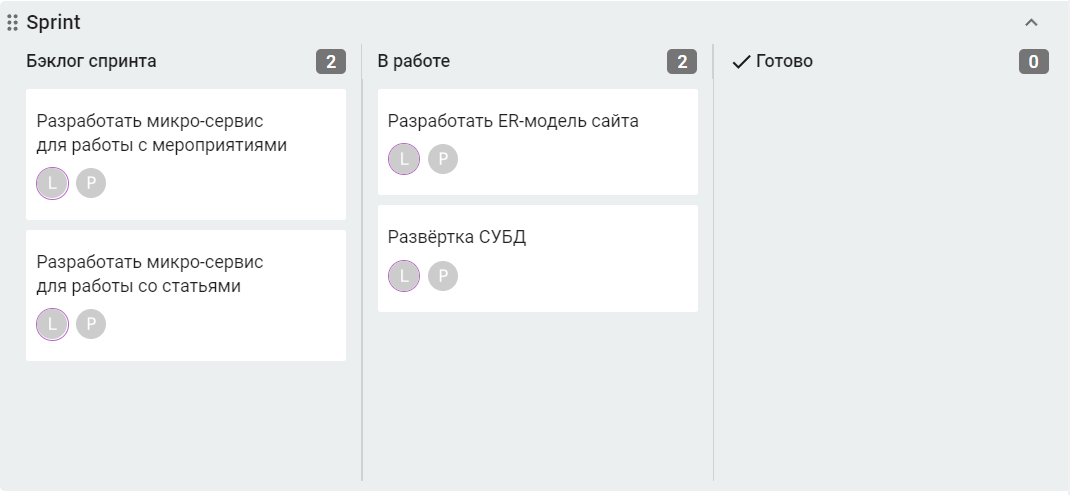
Санкт-Петербург 2023

**Цель работы:** Получение практических навыков разработки приложения, разработки архитектуры, применения паттернов программирования. Выполнить двухнедельные спринты согласно спланированным в Kaiten блокам.

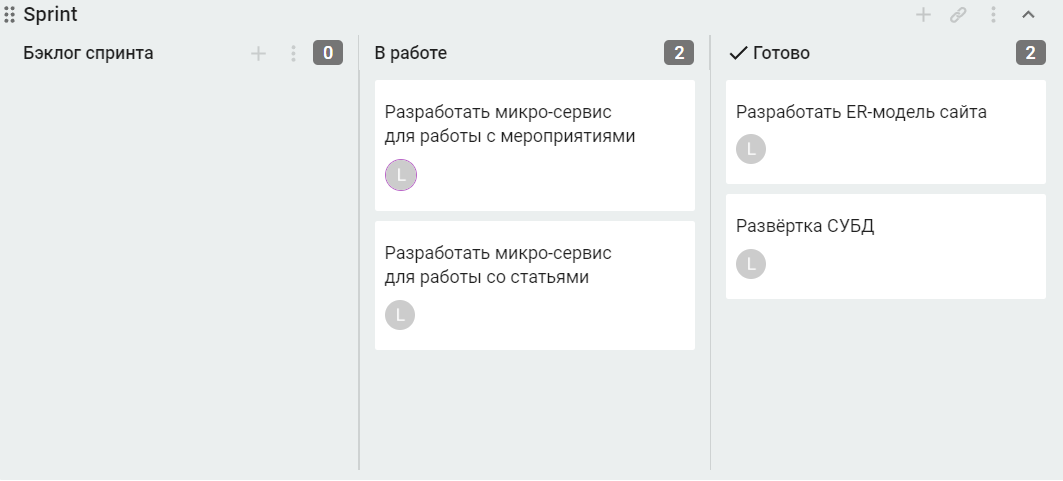
**Ход работы:**

Описанная в прошлых отчетах работа была выполнена в ходе четырёх двухнедельных спринтов:

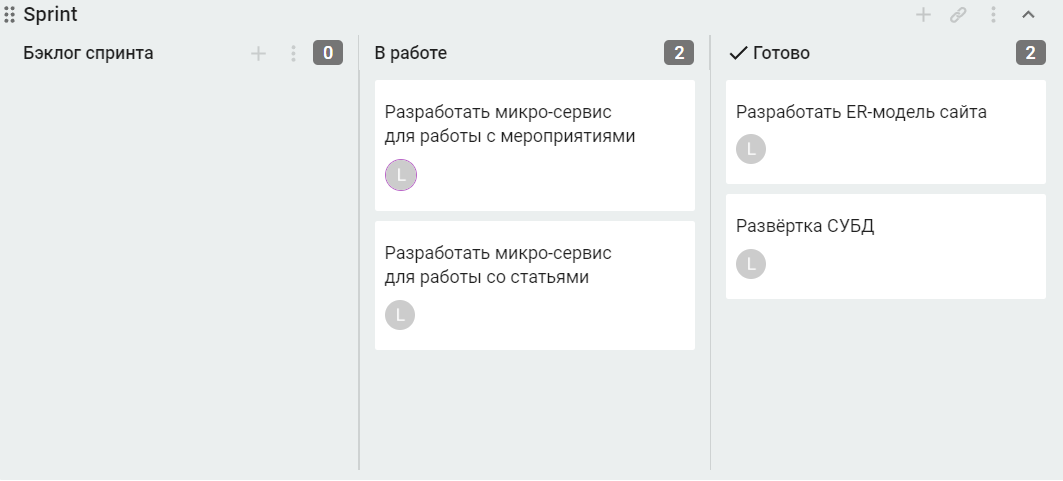
1. Во время первого были разработаны ЕR-модель, модели данных Django и развёрнута база данных PostgreSQL.

Рисунок 1 - Доска первого спринта

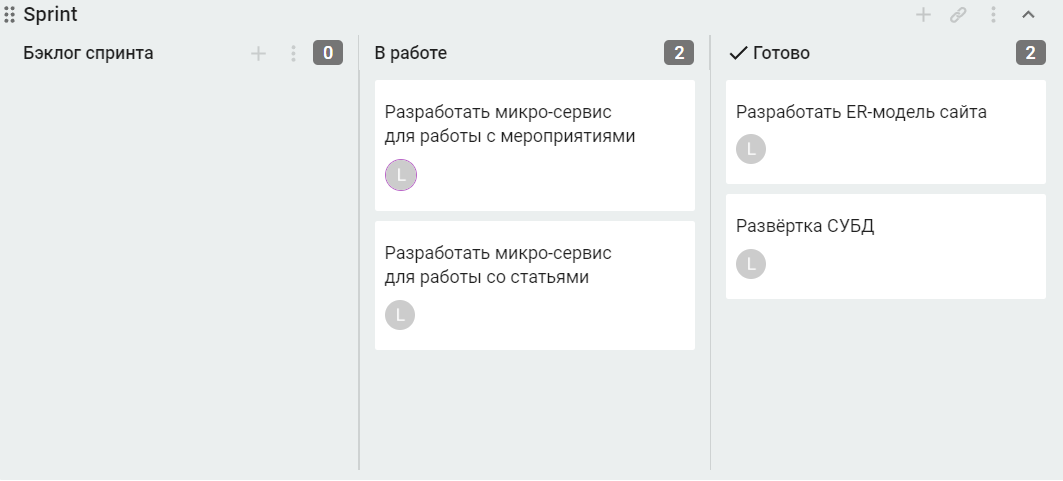
1. Во время второго были переработаны HTML-страницы в Django-шаблоны страниц.

Рисунок 2 - Доска второго спринта

1. Во время третьего был разработан модуль, отвечающий за маршрутизацию на сайте(прописаны url-адреса и связаны в пару url-адреса – view-модули).

Рисунок 3 - Доска третьего спринта

1. Во время четвёртого были разработаны модуль, отвечающий за отображение данных на странице с помощью шаблонов и моделей данных и работу с формами на них.

Рисунок 4 - Доска четвёртого спринта

В соответствии с методологией Scrum, данная часть разработки может быть изменена, если в том будет необходимость. Если такое будет проделано, то изменения будут описаны в последующих отчётах.

Разработка ER-модели сайта и развертка СУБД

Для хранения данных в веб-приложении, как правило, применются базы данных. И фреймворк Django уже по умолчанию предоставляет удобный функционал для работы с различными системами баз данных. По умолчанию Django в качестве базы данных использует SQLite, однако мы воспользуемся PostgreSQL.

Модели в Django описывают структуру используемых данных. Используемые в программе данные хранятся в базах данных, и с помощью моделей как раз осуществляется взаимодействие с базой данных. При создании приложения по умолчанию в его каталог добавляется файл models.py, который применяется для определения моделей. Код файла models.py будет представлен в приложении. В нашем случае потребуется создать 6 таблиц для работы, кроме того мы будем пользоваться уже встроенной в Django таблицей auth\_user, которая упрощает работу с пользователями.

Перечень таблиц:

1. аuth\_user – встроенная таблица пользователей, хранит в себе информацию о пользователях, их имена, пароли и т.д.
2. Articles\_article – таблица статей, хранит в себе названия статей, путь к файлу со статьей, аннотацию к статье, дату написания статьи, фотографию статьи.
3. Articles\_article\_user – таблица связывает таблицу со статьями и таблицу пользователей связью многие ко многим.
4. Article\_event – таблица событий, которая хранит название мероприятия, место проведения, точный адрес, дату начала, дату окончания, информацию о мероприятии, фотографию мероприятия.
5. Articles\_event\_user – таблица связывает таблицу с мероприятиями и таблицу пользователей связью многие ко многим, содержит также информацию о том является ли пользователь организатором или просто участником.
6. Articles\_student – таблица, которая связана с таблицей пользователей связью один к одному, служит расширением таблицы пользователей и содержит в себе номер телефона пользователя, учреждение, дополнительную информацию, фотографию пользователя.

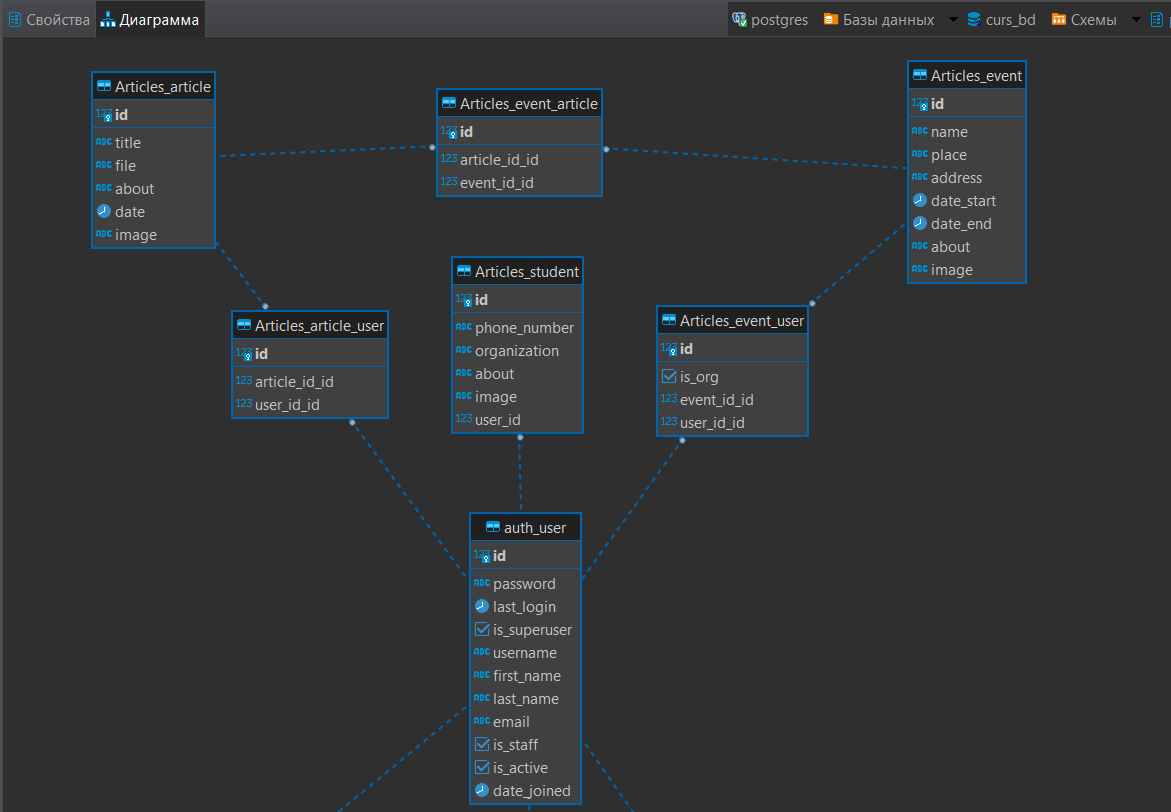


Рисунок 5 – Схема базы данных веб-приложения

Переработка HTML-страниц в Django-шаблоны страниц

Шаблоны (templates) отвечают за формирование внешнего вида приложения. Они предоставляют специальный синтаксис, который позволяет внедрять данные в код HTML. Одним из преимуществ шаблонов является то, что мы можем передать в них динамически из представлений различные данные. Для вывода данных в шаблоне могут использоваться различные способы.

Список созданных шаблонов:

1. add\_event.html – шаблон для создания нового мероприятия
2. article.html – шаблон для страницы конкретной статьи
3. article\_add.html – шаблон для загрузки новой статьи на сайт
4. article\_delete.html – шаблон для удаления статьи
5. article\_edit.html – шаблон для редактировании информации о статье
6. base.html – базовый шаблон, содержит информацию об общем стиле сайта
7. choose\_article.html – шаблон содержащий список статей пользователя
8. conference.html – шаблон информации о конкретном мероприятии
9. edit\_profile.html – шаблон редактирования профиля
10. event\_delete.html – шаблон удаления мероприятия
11. event\_edit.html – шаблон для редактирования мероприятия
12. event\_speakers.html – шаблон содержащий участников мероприятия
13. log\_in.html – шаблон для авторизации пользователя на сайте
14. log\_out.html – шаблон для выхода пользователя из учетной записи
15. main.html – шаблон для главной страницы сайта
16. menu.html – шаблон содержащий меню сайта
17. profile.html – шаблон профиля пользователя
18. registration.html – шаблон страницы регистрации
19. select\_article.html – шаблон страницы со списком статьей
20. select\_conference.html – шаблон страницы со списком мероприятий
21. take\_part.html – шаблон страницы для оправки статьи на мероприятие.

Код каждого из шаблонов представлен в приложении.

Разработка модуля, отвечающий за маршрутизацию на сайте

Центральным моментом любого веб-приложения является обработка запроса, который отправляет пользователь. В Django за обработку запроса отвечают представления или views. По сути представления представляют функции обработки, которые принимают данные запроса в виде объекта HttpRequest из пакета django.http и генерируют некоторый результат, который затем отправляется пользователю. Чтобы эти представления сопоставлялись с запросами, надо определить для них маршруты в проекте в файле urls.py.

URL-адреса прописанные в файле urls.py:

1. main – главная страница
2. log\_in – вход в учетную запись
3. log\_out – выход из учетной записи
4. register – регистрация пользователя
5. articles/article/<int:pk>/delete – страница удаление статьи
6. articles/article/<int:pk>/edit – страница редактирования статьи
7. articles/article/<int:pk>/download – страница загрузки статьи
8. articles/article/<int:pk>/ – страница просмотра информации о статье
9. articles/add – страница добавления статьи
10. articles – страница списка со статьями
11. events/event/<int:pk>/delete – страница удаления мероприятия
12. events/event/<int:pk>/edit – страница редактирования мероприятия
13. events/event/<int:pk>/speakers – страница статей зарегистрированных на мероприятии
14. events/event/<int:pk>/take\_part/<int:art> – страница добавления статьи для регистрации ее на мероприятии
15. events/event/<int:pk>/take\_part – страница для подачи заявки на участие на мероприятии
16. events/event/<int:pk>/ – страница просмотра информации о статье
17. events/add – страница добавления мероприятия
18. my\_events – страница со списком мероприятий, на которые пользователь уже подал статьи
19. events – страница со списком мероприятий
20. profile/edit – страница редактирования профиля
21. profile – страница профиля

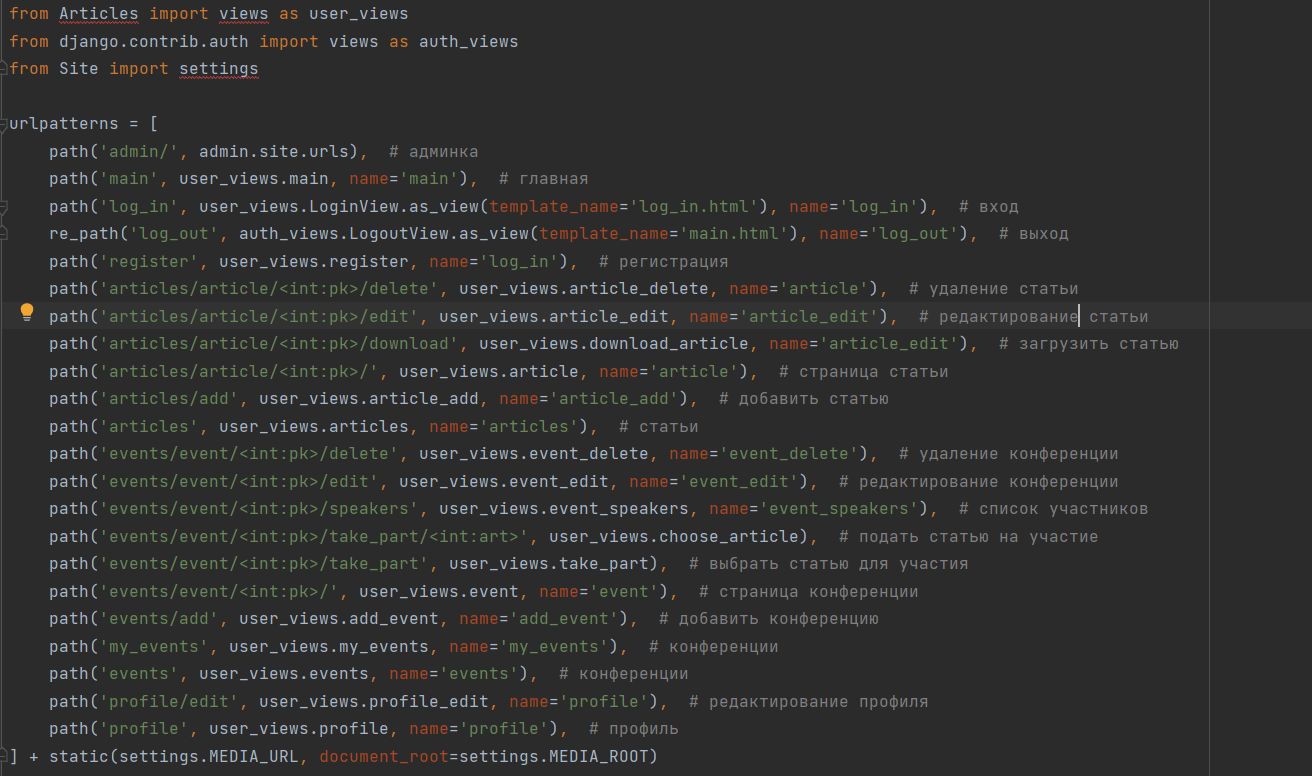


Рисунок 6 – файл urls.py

Полный код файла представлен в приложении.

Разработка модуля, отвечающего за отображение данных на странице

Функция представления, или сокращенно view - это функция Python, которая принимает веб-запрос и возвращает веб-ответ. Этим ответом может быть HTML-содержимое веб-страницы, перенаправление, ошибка 404, XML-документ и т.д.. Само представление содержит любую произвольную логику, необходимую для возврата ответа. Этот код может находиться где угодно, лишь бы он был в пути Python. По умолчанию представления размещаются в приложении в файле views.py.

Представления прописанные в файле views.py:

1. main – функция главной страницы
2. event\_speakers – функция страницы участников мероприятия
3. download\_article – функция загрузки статьи
4. register – функция регистрации на сайте
5. article\_delete – функция удаления статьи
6. article\_edit – функция редактирования статьи
7. article – функция отображения статьи
8. take\_part – функция отображения страницы для выбора статьи, которую собираются отправить на мероприятие
9. choose\_article – функция страницы со статьей выбранной для отправки на мероприятие
10. article\_add – функция добавления статьи
11. articles – функция страницы
12. event\_delete – функция для удаления мероприятия
13. event\_edit – функция редактирования мероприятия
14. event – функция страницы мероприятия
15. add\_event – функция добавления мероприятия
16. events – функция страницы со списком мероприятий
17. my\_events – функция страницы со списком мероприятий, на которые пользователь отправил статью
18. profile\_edit – функция редактирования профиля
19. profile – функция страницы профиля пользователя
20. LoginUser – класс для авторизации пользователя

Полный код файла views.py представлен в приложении.

**Вывод:** в процессе выполнения 4-х двухнедельных спринтов были разработаны ER-модель сайта, развернута база данных PostgreSQL, были переработаны html-страницы в шаблоны, а также разработан модуль, отвечающий за маршрутизацию, и модуль, отвечающий за отображение данных на странице.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Введение в Django. URL – https://metanit.com/python/django/1.1.php
2. Документация Django 4.1 – https://django.fun/ru/docs/django/4.1
3. Django в примерах – https://pocoz.gitbooks.io/django-v-primerah/content/

Приложение

Листинг файла views.py:

from django.http import HttpResponseRedirect  
from django.shortcuts import render, redirect, get\_object\_or\_404  
from django.contrib import messages  
from django.contrib.auth.decorators import login\_required  
from django.urls import reverse\_lazy  
from .utils import DataMixin  
from django.contrib.auth.models import User  
from django.contrib.auth.views import LoginView  
from .models import Student  
from .forms import \*  
from .models import \*  
  
  
def main(request): # главная страница  
 return render(request, "main.html")  
  
  
def event\_speakers(request, pk):  
 event = get\_object\_or\_404(Event, pk=pk)  
 articles = Article.objects.filter(event=event)  
 context = {  
 'articles': articles,  
 }  
 return render(request, "event\_speakers.html", context)  
  
  
def download\_article(request):  
  
 return render(request, "")  
  
  
def register(request): # страница регистрации  
 if request.method == 'POST':  
 form = CreateUserForm(request.POST)  
  
 if form.is\_valid():  
 student = Student.objects.create(  
 phone\_number="Телефон", organization="Введите учреждение", about="",  
 user=form.save(), image="zoro.jpg"  
 ) # создание профиля для пользователя  
 messages.success(request, f'Вы успешно зарегистрировались. Теперь вы можете войти.')  
 return HttpResponseRedirect("log\_in") # перевод на страницу входа  
 else:  
 messages.success(request, f'Ошибка! Проверьте корректность введенных данных') # сообщение об ошибке  
 form = CreateUserForm()  
 context = {  
 'form': form  
 }  
 return render(request, "registration.html", context)  
  
  
def article\_delete(request, pk):  
 article = get\_object\_or\_404(Article, pk=pk)  
 if request.method == 'POST':  
 article.delete()  
 messages.success(request, f'Ваша статья успешно удалена.')  
 return redirect('/articles')  
 context = {  
 'article': article,  
 }  
 return render(request, "article\_delete.html", context)  
  
  
def article\_edit(request, pk):  
 article = get\_object\_or\_404(Article, pk=pk)  
 if request.method == 'POST':  
 form = ArticleEdit(request.POST, request.FILES, instance=article)  
 if form.is\_valid():  
 form.save()  
 messages.success(request, f'Ваша статья успешно обновлена.')  
 return HttpResponseRedirect("/articles")  
 else:  
 form = ArticleEdit(instance=article)  
 context = {  
 'form': form,  
 }  
 return render(request, "article\_edit.html", context)  
  
  
def article(request, pk): # страница информации о статье  
 article = get\_object\_or\_404(Article, pk=pk)  
 author = get\_object\_or\_404(Article\_User, article\_id=pk)  
 user = get\_object\_or\_404(User, username=author.user\_id)  
 context = {  
 'article': article,  
 'author': user  
 }  
 return render(request, "article.html", context)  
  
  
def take\_part(request, pk):  
 event = get\_object\_or\_404(Event, pk=pk)  
 articles = Article.objects.filter(user=request.user)  
 context = {  
 'event': event,  
 'articles': articles,  
 }  
 return render(request, "take\_part.html", context)  
  
  
def choose\_article(request, pk, art):  
 event = get\_object\_or\_404(Event, pk=pk)  
 article = get\_object\_or\_404(Article, pk=art)  
 if request.method == 'POST':  
 con = Event\_Article.objects.filter(event\_id=pk, article\_id=art)  
 if con.exists():  
 messages.success(request, f'Эта статья уже зарегистрированна. Выберите другую статью')  
 return redirect('/events/event/' + str(pk) + '/take\_part')  
 con = Event\_Article.objects.create(article\_id=article, event\_id=event)  
 messages.success(request, f'Ваша статья успешно зарегистрирована. Вас будут ожидать на мерорприятии')  
 return redirect('/events/event/' + str(pk))  
 context = {  
 'event': event,  
 'article': article,  
 }  
 return render(request, "choose\_article.html", context)  
  
  
def article\_add(request):  
 if request.method == 'POST':  
 form = ArticleEdit(request.POST, request.FILES)  
 if form.is\_valid():  
 obj = form.save(commit=False)  
 user = User.objects.filter(username=request.user)  
 obj.save()  
 obj.user.set(user)  
 obj.save()  
 messages.success(request, f'Ваша статья успешно добавлена.')  
 return redirect('/articles')  
 else:  
 messages.success(request, f'Ваша статья не была добавлена. Проверьте корректность введенных данных')  
 else:  
 form = ArticleEdit(instance=request.user)  
 context = {  
 'form': form,  
 }  
 return render(request, "article\_add.html", context)  
  
  
@login\_required  
def articles(request):  
 article = Article.objects.filter(user=request.user)  
 context = {  
 'articles': article,  
 }  
 return render(request, "select\_article.html", context)  
  
  
def event\_delete(request, pk):  
 event = get\_object\_or\_404(Event, pk=pk)  
 if request.method == 'POST':  
 event.delete()  
 messages.success(request, f'Ваше мероприятие успешно удалено')  
 return redirect('/events')  
 context = {  
 'event': event,  
 }  
 return render(request, "event\_delete.html", context)  
  
  
def event\_edit(request, pk):  
 event = get\_object\_or\_404(Event, pk=pk)  
 if request.method == 'POST':  
 form = EventEdit(request.POST, request.FILES, instance=event)  
 if form.is\_valid():  
 form.save()  
 messages.success(request, f'Вашe мероприятие успешно обновлена.')  
 return HttpResponseRedirect("/events")  
 else:  
 form = EventEdit(instance=event)  
 context = {  
 'form': form,  
 }  
 return render(request, "event\_edit.html", context)  
  
  
def event(request, pk): # страница информации о конференции  
 event = get\_object\_or\_404(Event, pk=pk)  
 orgs = User.objects.filter(event\_user\_\_event\_id\_id=pk, event\_user\_\_is\_org="True")  
 try:  
 org = Event\_User.objects.get(event\_id=pk, user\_id=request.user)  
 context = {  
 'event': event,  
 'orgs': orgs,  
 'org': org,  
 }  
 except Exception:  
 context = {  
 'event': event,  
 'orgs': orgs,  
 }  
 return render(request, "conference.html", context)  
  
  
def add\_event(request):  
 if request.method == 'POST':  
 form = EventEdit(request.POST, request.FILES)  
 if form.is\_valid():  
 obj = form.save(commit=False)  
 user = User.objects.filter(username=request.user)  
 obj.save()  
 obj.user.set(user, through\_defaults={"is\_org": True})  
 obj.save()  
 messages.success(request, f'Ваше мероприятие успешно добавлено.')  
 return redirect('/events')  
 else:  
 messages.success(request, f'Ваше мероприятие не было добавлено. Проверьте корректность введенных данных')  
 else:  
 form = EventEdit(instance=request.user)  
 context = {  
 'form': form,  
 }  
 return render(request, "add\_event.html", context)  
  
  
def events(request):  
 events = Event.objects.all()  
 context = {  
 'events': events,  
 }  
 return render(request, "select\_conference.html", context)  
  
  
def my\_events(request):  
 events = Event.objects.filter(user=request.user)  
 context = {  
 'events': events  
 }  
 return render(request, "select\_conference.html", context)  
  
  
def profile\_edit(request):  
 if request.method == 'POST':  
 u\_form = UserUpdateForm(request.POST, instance=request.user)  
 p\_form = StudentUpdateForm(request.POST,  
 request.FILES,  
 instance=request.user.student)  
 if u\_form.is\_valid() and p\_form.is\_valid():  
 u\_form.save()  
 p\_form.save()  
 messages.success(request, f'Ваш профиль успешно обновлен.') # сообщение об успехе  
 return redirect('profile') # перевод на страницу профиля  
 else:  
 u\_form = UserUpdateForm(instance=request.user)  
 p\_form = StudentUpdateForm(instance=request.user.student)  
  
 context = {  
 'u\_form': u\_form,  
 'p\_form': p\_form  
 }  
 return render(request, "edit\_profile.html", context)  
  
  
@login\_required  
def profile(request): # страница профиля  
 stud = Student.objects.get(user=request.user)  
 if stud.phone\_number == "Телефон": # перевод на страницу редактирования профиля  
 return HttpResponseRedirect("/profile/edit")  
 return render(request, "profile.html")  
  
  
class LoginUser(DataMixin, LoginView): # класс для авторизации пользователя  
 form\_class = LoginForm  
 template\_name = 'log\_in.html'  
  
 def get\_context\_data(self, \*, object\_list=None, \*\*kwargs):  
 context = super().get\_context\_data(\*\*kwargs)  
 c\_def = self.get\_user\_context(title="Вход")  
 return dict(list(context.items()) + list(c\_def.items()))  
  
 def get\_success\_url(self):  
 return reverse\_lazy('profile')

Листинг файла urls.py

from django.conf.urls.static import static  
from django.contrib import admin  
from django.urls import path, re\_path  
from Articles import views as user\_views  
from django.contrib.auth import views as auth\_views  
from Site import settings  
  
urlpatterns = [  
 path('admin/', admin.site.urls), # админка  
 path('main', user\_views.main, name='main'), # главная  
 path('log\_in', user\_views.LoginView.as\_view(template\_name='log\_in.html'), name='log\_in'), # вход  
 re\_path('log\_out', auth\_views.LogoutView.as\_view(template\_name='main.html'), name='log\_out'), # выход  
 path('register', user\_views.register, name='log\_in'), # регистрация  
 path('articles/article/<int:pk>/delete', user\_views.article\_delete, name='article'), # удаление статьи  
 path('articles/article/<int:pk>/edit', user\_views.article\_edit, name='article\_edit'), # редактирование статьи  
 path('articles/article/<int:pk>/download', user\_views.download\_article, name='article\_edit'), # загрузить статью  
 path('articles/article/<int:pk>/', user\_views.article, name='article'), # страница статьи  
 path('articles/add', user\_views.article\_add, name='article\_add'), # добавить статью  
 path('articles', user\_views.articles, name='articles'), # статьи  
 path('events/event/<int:pk>/delete', user\_views.event\_delete, name='event\_delete'), # удаление конференции  
 path('events/event/<int:pk>/edit', user\_views.event\_edit, name='event\_edit'), # редактирование конференции  
 path('events/event/<int:pk>/speakers', user\_views.event\_speakers, name='event\_speakers'), # список участников  
 path('events/event/<int:pk>/take\_part/<int:art>', user\_views.choose\_article), # подать статью на участие  
 path('events/event/<int:pk>/take\_part', user\_views.take\_part), # выбрать статью для участия  
 path('events/event/<int:pk>/', user\_views.event, name='event'), # страница конференции  
 path('events/add', user\_views.add\_event, name='add\_event'), # добавить конференцию  
 path('my\_events', user\_views.my\_events, name='my\_events'), # конференции  
 path('events', user\_views.events, name='events'), # конференции  
 path('profile/edit', user\_views.profile\_edit, name='profile'), # редактирование профиля  
 path('profile', user\_views.profile, name='profile'), # профиль  
] + static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

Листинг файла models.py

import datetime  
from django.utils import timezone  
from django.db import models  
from django.contrib.auth.models import User  
from django.urls import reverse  
  
  
class Student(models.Model): # модель профиля  
 phone\_number = models.CharField(max\_length=12, verbose\_name="Контактный телефон", null=True, blank=True)  
 organization = models.CharField(max\_length=50, verbose\_name="Учреждение", null=True, blank=True)  
 about = models.TextField(verbose\_name="О себе", null=True, blank=True)  
 image = models.ImageField(upload\_to="photos/", verbose\_name="Фото", null=True, blank=True)  
 user = models.OneToOneField(User, on\_delete=models.CASCADE)  
  
 def \_\_str\_\_(self):  
 return f'{self.user.username} Student'  
  
  
class Event(models.Model): # модель конференции  
 name = models.CharField(max\_length=50)  
 place = models.CharField(max\_length=50)  
 address = models.TextField(null=True, blank=True)  
 date\_start = models.DateField()  
 date\_end = models.DateField()  
 about = models.TextField()  
 image = models.ImageField(upload\_to="photos/events", verbose\_name="Фото", null=True, blank=True)  
 user = models.ManyToManyField(User, through="Event\_User")  
  
  
class Article(models.Model): # модель статьи  
 title = models.CharField(max\_length=100)  
 file = models.FileField(upload\_to='articles/')  
 about = models.TextField(null=True, blank=True)  
 date = models.DateField(default=timezone.now())  
 image = models.ImageField(upload\_to="photos/articles", verbose\_name="Фото", null=True, blank=True)  
 user = models.ManyToManyField(User, through="Article\_User")  
 event = models.ManyToManyField(Event, through="Event\_Article")  
  
  
class Article\_User(models.Model): # статья\_пользователь  
 article\_id = models.ForeignKey(Article, on\_delete=models.CASCADE)  
 user\_id = models.ForeignKey(User, on\_delete=models.CASCADE)  
  
  
class Event\_User(models.Model): # конференция\_пользователь  
 event\_id = models.ForeignKey(Event, on\_delete=models.CASCADE)  
 user\_id = models.ForeignKey(User, on\_delete=models.CASCADE)  
 is\_org = models.BooleanField()  
  
  
class Event\_Article(models.Model): # конференция\_статья  
 event\_id = models.ForeignKey(Event, on\_delete=models.CASCADE)  
 article\_id = models.ForeignKey(Article, on\_delete=models.CASCADE)

Листинг файла add\_event.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Добавить мероприятие{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Введите данные мероприятия</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST" enctype="multipart/form-data">  
 <div class="form-error">{{ form.non\_field\_errors }}</div>  
 <fieldset class="form-group">  
 {% csrf\_token %}  
 {{ form|crispy }}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Добавить">  
 </div>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла article.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Статья{% endblock title%}  
{% block content %}  
{% block info %}  
<div class="info">  
 <div class="left-column">  
 <div class="div-name">  
 <div class="div-h2">  
 <h2>{{article.title}}</h2>  
 </div>  
 </div>  
 <div class="education">  
 <div class="div-h3">  
 <h3>Автор статьи</h3>  
 </div>  
 <div class="div-p">  
 <p>  
 <a>{{author.first\_name}} {{author.last\_name}}</a>  
 </p>  
 </div>  
 <div class="div-h3">  
 <h3>Почта</h3>  
 </div>  
 <div class="div-p">  
 <p>  
 <a>{{author.email}}</a>  
 </p>  
 </div>  
 </div>  
 <div class="organization">  
 <div class="div-h3">  
 <h3>Дата и время публикации</h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.date}}</p>  
 </div>  
 </div>  
 <div class="button-block">  
 <a href='/articles/article/{{article.id}}/edit' class="button">Редактировать</a>  
 </div>  
 <div class="button-block">  
 <a href='/articles/article/{{article.id}}/delete' class="button">Удалить статью</a>  
 </div>  
 </div>  
 <div class="right-column">  
 <div class="profile-image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
 <div class="dop-info">  
 <div class="div-a blue-text">  
 <a href="">Скачать статью</a>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock info %}  
{% block about %}  
<div class="about-block">  
 <div class="div-h3">  
 <h3>О статье</h3>  
 <p>{{article.about}}</p>  
 </div>  
</div>  
{% endblock about %}  
{% endblock content%}

Листинг файла article\_add.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Добавить статью{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Введите данные статьи</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST" enctype="multipart/form-data">  
 <div class="form-error">{{ form.non\_field\_errors }}</div>  
 <fieldset class="form-group">  
 {% csrf\_token %}  
 {{ form|crispy }}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Добавить">  
 </div>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла article\_delete.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Удалить статью{% endblock title%}  
  
{% block content %}  
<h3 class="u-text u-text-default u-text-1">Вы уверены что хотите удалить следующую статью:</h3>  
<h4>{{ article.title }}</h4>  
<div class="image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
<form method="POST" enctype="multipart/form-data">  
 {% csrf\_token %}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Удалить">  
 </div>  
 </div>  
</form>  
<a href="/articles" class="u-border-none u-btn u-btn-round u-button-style u-grey-90 u-hover-grey-75 u-radius-50 u-btn-1">Отмена</a>  
{% endblock content %}

Листинг файла article\_edit.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Редактировать статью{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Отредактируйте данные статьи</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST" enctype="multipart/form-data">  
 <div class="form-error">{{ form.non\_field\_errors }}</div>  
 <fieldset class="form-group">  
 {% csrf\_token %}  
 {{ form|crispy }}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Редактировать">  
 </div>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла base.html

<!DOCTYPE html>  
{% load static %}  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>{% block title %}Title{% endblock title%}</title>  
 <link rel="stylesheet" href="{% static 'css/style.css' %}" />  
</head>  
<body>  
<div class="page">  
 {% include "menu.html"%}  
{% if messages %}  
 <ul>  
 {% for message in messages %}  
 <li>{{ message }}</li>  
 {% endfor %}  
 </ul>  
 {% endif %}  
  
 {% block content %}{% endblock content %}  
</div>  
</body>  
</html>

Листинг файла choose\_article.html

{% extends "base.html" %}  
{% load crispy\_forms\_tags %}  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Статья{% endblock title%}  
{% block content %}  
{% block info %}  
<div class="info">  
 <div class="left-column">  
 <div class="div-name">  
 <div class="div-h2">  
 <h2>{{article.title}}</h2>  
 </div>  
 </div>  
 <div class="education">  
 <div class="div-h3">  
 <h3>Автор статьи</h3>  
 </div>  
 <div class="div-p">  
 <p>  
 <a href="">{{author}}</a>  
 </p>  
 </div>  
 </div>  
 <div class="organization">  
 <div class="div-h3">  
 <h3>Дата и время публикации</h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.date}}</p>  
 </div>  
 </div>  
 <form method="POST" enctype="multipart/form-data">  
 {% csrf\_token %}  
 <div class="button-block">  
 <input type="submit" value="Отправить эту статью">  
 </div>  
 </form>  
 </div>  
 <div class="right-column">  
 <div class="profile-image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
 <div class="dop-info">  
 <div class="div-a blue-text">  
 <a href="">Скачать статью</a>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock info %}  
{% block about %}  
<div class="about-block">  
 <div class="div-h3">  
 <h3>О статье</h3>  
 <p>{{article.about}}</p>  
 </div>  
</div>  
{% endblock about %}  
{% endblock content%}

Листинг файла conference.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Мероприятия{% endblock title%}  
{% block content %}  
{% block info %}  
<div class="info">  
 <div class="left-column">  
 <div class="div-name">  
 <div class="div-h2">  
 <h2>{{event.name}}</h2>  
 </div>  
 </div>  
 <div class="education">  
 <div class="div-h3">  
 <h3>Место проведения</h3>  
 </div>  
 <div class="div-p">  
 <p>{{event.place}}</p>  
 </div>  
 </div>  
 <div class="organization">  
 <div class="div-h3">  
 <h3>Дата проведения</h3>  
 </div>  
 <div class="div-p">  
 <p>{{event.date\_start}}</p>  
 </div>  
 </div>  
 <div class="button-block">  
 <a href='/events/event/{{event.id}}/take\_part' class="button">Подать заявку</a>  
 </div>  
 {% if org.is\_org %}  
 <div class="button-block">  
 <a href='/events/event/{{event.id}}/edit' class="button">Редактировать</a>  
 </div>  
 <div class="button-block">  
 <a href='/events/event/{{event.id}}/delete' class="button">Удалить</a>  
 </div>  
 {% endif %}  
 </div>  
 <div class="right-column">  
 <div class="profile-image-block">  
 <div class="image">  
 <img src="{{event.image.url}}">  
 </div>  
 </div>  
 <div class="dop-info">  
 <div class="div-a blue-text">  
 <a href="speakers">Список участников</a>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock info %}  
{% block about %}  
<div class="about-block">  
 <div class="div-h3">  
 <h3>О мероприятии</h3>  
  
 <p>{{event.about}}</p>  
 </div>  
</div>  
{% endblock about %}  
{% block contact %}  
<div class="contact-block">  
 <div class="div-h3">  
 <h3>Контакты</h3>  
 </div>  
 <div class="contact-info">  
 <div class="div-h3">  
 <h3>Организаторы</h3>  
 </div>  
 {% for org in orgs%}  
 <div class="organizator">  
 <div class="div-img">  
 <img src="{{ org.student.image.url }}">  
 </div>  
 <div class="div-p">  
 <p>  
 <a>{{org.first\_name}} {{org.last\_name}}</a>  
 </p>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
 </div>  
</div>  
{% endblock contact %}  
{% endblock content%}

Листинг файла edit\_profile.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Редактирование профиля{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Введите новые данные профиля</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST">  
 {% csrf\_token %}  
 <div class="form-error">{{ form.non\_field\_errors }}</div>  
 <fieldset class="form-group">  
 <legend>Ваш профиль</legend>  
 {{ u\_form|crispy }}  
 {{ p\_form|crispy }}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Обновить">  
 </div>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла event\_delete.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Удалить мероприятие{% endblock title%}  
  
{% block content %}  
<h3 class="u-text u-text-default u-text-1">Вы уверены что хотите удалить следующее мероприятие:</h3>  
<h4>{{ event.name }}</h4>  
<div class="image-block">  
 <div class="image">  
 <img src="{{event.image.url}}">  
 </div>  
 </div>  
<form method="POST" enctype="multipart/form-data">  
 {% csrf\_token %}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Удалить">  
 </div>  
 </div>  
</form>  
<a href="/events" class="u-border-none u-btn u-btn-round u-button-style u-grey-90 u-hover-grey-75 u-radius-50 u-btn-1">Отмена</a>  
{% endblock content %}

Листинг файла event\_edit.html

{% extends "base.html" %}  
{% load static %}  
{% load crispy\_forms\_tags %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Редактировать Мероприятие{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Отредактируйте данные мероприятия</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST" enctype="multipart/form-data">  
 <div class="form-error">{{ form.non\_field\_errors }}</div>  
 <fieldset class="form-group">  
 {% csrf\_token %}  
 {{ form|crispy }}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Редактировать">  
 </div>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла event\_speakers.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Публикации{% endblock title%}  
  
{% block content %}  
  
{% block search %}  
<div class="search-section">  
 <div class="search-block">  
 <input type="text" name="search-conference" placeholder="Я хочу найти статью">  
 </div>  
</div>  
{% endblock search%}  
  
{% block view %}  
<div class="view-section">  
 <div class="view-grid">  
 {% for article in articles %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3><a href='/articles/article/{{article.id}}' style="color:black">{{article.title}}</a></h3>  
 </div>  
 <div class="div-h3">  
 <h3><a style="color:black">{{author.user.username}}</a></h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock view %}  
  
{% endblock content %}

Листинг файла log\_in.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Вход{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Вход</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST">  
 {% csrf\_token %}  
 {{form}}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Войти">  
 </div>  
 </div>  
 </form>  
 </div>  
 </div>  
 <div class="info-block">  
 <div class="div-p">  
 <p>Нет аккаунта?</p>  
 </div>  
 <div class="div-a">  
 <a href="register">Зарегистрироваться</a>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла log\_out.html

{% extends "base.html" %}  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Выход{% endblock title %}  
  
{% block content %}  
<h2>Вы вышли из своей учетной записи</h2>  
 <div class="border-top pt-3">  
 <a href="/main">На главную</a>  
 </div>  
{% endblock content %}

Листинг файла main.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Главная{% endblock title%}  
  
{% block content %}  
  
{% block welcome-banner %}  
<div class="welcome-section">  
 <div class="table">  
 <div class="welcome-content">  
 <div class="text">  
 <h2>{{welcome\_h2}}</h2>  
 <p>{{welcome\_about}}</p>  
 </div>  
 <div class="button">  
 <a href="">Подробнее</a>  
 </div>  
 </div>  
 <div class="welcome-banner">  
 <img src="{% static 'image/welcome-banner.svg' %}">  
 </div>  
 </div>  
</div>  
{% endblock welcome-banner %}  
  
{% block 3-last-conference%}  
<div class="board-card-section">  
 <div class="div-header">  
 <div class="div-h2">  
 <h2>{{conference\_h2}}</h2>  
 </div>  
 <div class="div-a">  
 <a>{{conference\_a}}</a>  
 </div>  
 </div>  
 <div class="board">  
 {% for event in events %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{% static 'image/image.svg' %}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3>{{event.name}}</h3>  
 </div>  
 <div class="div-p">  
 <p>{{event.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock 3-last-conference %}  
  
{% block 3-last-article %}  
<div class="board-card-section">  
 <div class="div-header">  
 <div class="div-h2">  
 <h2>{{article\_h2}}</h2>  
 </div>  
 <div class="div-a">  
 <a>{{article\_a}}</a>  
 </div>  
 </div>  
 <div class="board">  
 {% for article in articles %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{% static 'image/image.svg' %}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3>{{article.title}}</h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock 3-last-article %}  
  
{% endblock content %}

Листинг файла menu.html

{% load static %}  
<div class="menu">  
 <div class="logo">  
 <img src="{% static 'image/guap.svg' %}">  
 </div>  
 <div class="nav-block">  
 <div class="div-a">  
 <a href="/profile">Профиль</a>  
 </div>  
 <div class="div-a">  
 <a href="/events">Мероприятия</a>  
 </div>  
 <div class="div-a">  
 <a href="/articles">Публикации</a>  
 </div>  
 </div>  
 <div class="enter-nav">  
 <div class="enter-img">  
 <img src="{% static 'image/enter-vector.svg' %}">  
 </div>  
 <div class="div-a">  
 {% if user.is\_authenticated %}  
 <a href="/log\_out" style="color:black">Выход</a>  
 {% else %}  
 <a href="/log\_in" style="color:black">Вход</a>  
 {% endif %}  
 </div>  
 </div>  
</div>

Листинг файла profile.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}{{user.first\_name}} {{user.last\_name}}{% endblock title%}  
  
{% block content %}  
{% block info %}  
<div class="info">  
 <div class="left-column">  
 <div class="div-name">  
 <div class="div-h2">  
 <h2>{{user.first\_name}} {{user.last\_name}}</h2>  
 </div>  
 </div>  
 <div class="organization">  
 <div class="div-h3">  
 <h3>Организация</h3>  
 </div>  
 <div class="div-p">  
 <p>{{user.student.organization}}</p>  
 </div>  
 </div>  
 </div>  
 <div class="right-column">  
 <div class="profile-image-block">  
 <div class="image">  
 <img src="{{ user.student.image.url }}">  
 </div>  
 </div>  
 <div class="dop-info">  
 <div class="div-p email">  
 <p>Email: </p>  
 </div>  
 <div class="div-p blue-text">  
 <p>{{user.email}}</p>  
 </div>  
 <div class="div-p">  
 <p>Номер телефона: </p>  
 </div>  
 <div class="div-p blue-text">  
 <p>{{user.student.phone\_number}}</p>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock info %}  
  
{% block 3-last-conference%}  
<div class="board-card-section">  
 <div class="div-header">  
 <div class="div-h2">  
 <h2>{{conference\_h2}}</h2>  
 </div>  
 <div class="div-a">  
 <a>{{conference\_a}}</a>  
 </div>  
 </div>  
 <div class="board">  
 {% for event in events %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{% static 'image/image.svg' %}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3>{{event.name}}</h3>  
 </div>  
 <div class="div-p">  
 <p>{{event.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock 3-last-conference %}  
  
{% block 3-last-article %}  
<div class="board-card-section">  
 <div class="div-header">  
 <div class="div-h2">  
 <h2>{{article\_h2}}</h2>  
 </div>  
 <div class="div-a">  
 <a>{{article\_a}}</a>  
 </div>  
 </div>  
 <div class="board">  
 {% for article in articles %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{% static 'image/image.svg' %}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3>{{article.title}}</h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock 3-last-article %}  
{% endblock content %}

Листинг файла registration.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Мероприятия{% endblock title%}  
  
{% block content %}  
<div class="dialog-section">  
 <div class="enter-section">  
 <div class="enter-block">  
 <div class="div-h2">  
 <h2>Регистрация</h2>  
 </div>  
 <div class="form-block">  
 <form method="POST">  
 {% csrf\_token %}  
 {{form}}  
 <div class="button-block">  
 <div class="button">  
 <input type="submit" value="Зарегистрироваться">  
 </div>  
 </div>  
 </form>  
 </div>  
 </div>  
 </div>  
</div>  
{% endblock content %}

Листинг файла select\_article.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Публикации{% endblock title%}  
  
{% block content %}  
  
{% block search %}  
<div class="search-section">  
 <div class="search-block">  
 <input type="text" name="search-conference" placeholder="Я хочу найти статью">  
 </div>  
</div>  
{% endblock search%}  
  
{% block view %}  
<div class="view-section">  
 <div class="button-block">  
 <div class="button">  
 <a href="/articles/add">Добавить статью</a>  
 </div>  
 </div>  
 <div class="view-grid">  
 {% for article in articles %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3><a href='/articles/article/{{article.id}}' style="color:black">{{article.title}}</a></h3>  
 </div>  
 <div class="div-p">  
 <p>{{article.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock view %}  
  
{% endblock content %}

Листинг файла select\_conference.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Мероприятия{% endblock title%}  
  
{% block content %}  
  
{% block search %}  
<div class="search-section">  
 <div class="search-block">  
 <input type="text" name="search-conference" placeholder="Я хочу найти мероприятие">  
 </div>  
</div>  
{% endblock search%}  
  
{% block view %}  
 <div class="button-block">  
 <a href='/events/add' class="button">Добавить мероприятие</a>  
 </div>  
<div class="view-section">  
 <div class="view-grid">  
 {% for event in events %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{{event.image.url}}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3><a href='/events/event/{{event.id}}/' style="color:black">{{event.name}}</a></h3>  
 </div>  
 <div class="div-p">  
 <p>{{event.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock view %}  
  
{% endblock content %}

Листинг файла take\_part.html

{% extends "base.html" %}  
  
{% load static %}  
<link rel="stylesheet" href="{% static 'css/style.css' %}" />  
  
{% block title %}Публикации{% endblock title%}  
  
{% block content %}  
  
{% block search %}  
<div class="search-section">  
 <div class="search-block">  
 <input type="text" name="search-conference" placeholder="Я хочу найти статью">  
 </div>  
</div>  
{% endblock search%}  
  
{% block view %}  
<div class="view-section">  
 <div class="view-grid">  
 {% for article in articles %}  
 <div class="card">  
 <div class="image-block">  
 <div class="image">  
 <img src="{{article.image.url}}">  
 </div>  
 </div>  
 <div class="content">  
 <div class="div-h3">  
 <h3><a href='/articles/article/{{article.id}}' style="color:black">{{article.title}}</a></h3>  
 </div>  
 <div class="button-block">  
 <div class="button">  
 <a href="take\_part/{{article.id}}">Добавить эту статью</a>  
 </div>  
 </div>  
 <div class="div-p">  
 <p>{{article.about}}</p>  
 </div>  
 </div>  
 </div>  
 {% endfor %}  
 </div>  
</div>  
{% endblock view %}  
  
{% endblock content %}