ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»

НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ «ШАГ В БУДУЩЕЕ, МОСКВА»

39741

регистрационный номер

Информатика и системы управления

Системы обработки информации и управления

СОЗДАНИЕ ВЕБ – САЙТА ДЛЯ ПОИСКА РЕЦЕПТОВ ПО ИМЕЮЩИМСЯ ИНГРЕДИЕНТАМ

Автор:
Зиновьев Константин Николаевич
ГБОУ Инженерная школа №1581, 10 И

Научный руководитель:
Веселовская Ольга Александровна
МГТУ им. Н.Э. Баумана, к.т.н., доцент

Москва – 2024

АННОТАЦИЯ

Приготовление пищи является одним из популярных занятий для современного человека. В настоящее время существует большое количество различных электронных ресурсов, позволяющих узнать рецепт и способ приготовления выбранного блюда.

Большинство таких веб-ресурсов позволяют пользователю ввести список продуктов, из которых он хочет приготовить то или иное блюдо, и предлагают подбор рецептов. Однако после анализа таких сервисов был выявлен их важный недостаток – предлагаемые рецепты содержат в себе дополнительные продукты, помимо введенных пользователем.

Такие ресурсы не всегда подходят пользователю, так как в реальной жизни возникают ситуации, когда дома есть только определенные продукты, и приобретение дополнительных не может быть возможно.

В данной работе разрабатывается веб-ресурс, который предлагает рецепты на основе только тех продуктов, которые выбраны пользователем. Кроме того, добавлен ряд функций, упрощающих работу пользователя: возможность ведения списка продуктов, списка покупок.

Цель работы: разработка информационной системы, состоящий из базы данных рецептов, графического веб-интерфейса для работы с ней и алгоритмического обеспечения для реализации заявленного функционала.

Методы и приемы: проектирование информационной системы и ее компонентов, прототипирование графического интерфейса, разработка базы данных на Postgresql, разработка веб-интерфейса с помощью фреймворка Django.

Результаты: разработана действующая информационная система, состоящая из базы данных и веб-интерфейса, которая подбирает список рецептов блюд, которые пользователь может приготовить.

Выводы: получен веб-ресурс, который удобен тем, что позволяет находить рецепты только из введенных ингредиентов, не показывая рецепты, в которых используются дополнительные продукты, которых нет у пользователя, что упрощает его повседневную жизнь.

СОДЕРЖАНИЕ

1. Введение

В наше время существует множество популярных веб-ресурсов, предлагающих подбор рецептов на основе выбранных пользователем продуктов. Однако, проведенный анализ выявил значительный недостаток подобных сервисов: множество предлагаемых рецептов содержат в себе дополнительные продукты, не указанные пользователем. Этот факт создает неудобства для пользователей, поскольку в реальной жизни часто возникают ситуации, когда доступны лишь определенные продукты, и приобретение дополнительных может быть невозможным или нецелесообразным.

С учетом указанных проблем, в данной работе проводится разработка веб-сервиса, который предлагает рецепты исключительно на основе продуктов, выбранных пользователем. Это позволит улучшить пользовательский опыт и повысить практичность использования сервиса. Кроме того, в рамках разработки веб-платформы внедряются дополнительные функции, направленные на упрощение пользовательского взаимодействия с сервисом, такие как возможность ведения списка имеющихся продуктов и планирование необходимых покупок.

Целью данного проекта является разработка информационной системы, включающей базу данных рецептов, графический веб-интерфейс для работы с ней и алгоритмическое обеспечение для реализации заявленного функционала. Создание интуитивно понятного и эффективного инструмента направлено на удовлетворение потребностей пользователей в быстром и удобном поиске рецептов на основе имеющихся у них продуктов. Предполагается, что разработанный веб-сервис станет полезным и популярным инструментом для всех, кто интересуется кулинарией и стремится к более удобному способу планирования приема пищи.

Задачи, направленные на достижение поставленной цели:

- Анализ существующих решений в данной предметной области;
- Изучение теоретического материала по созданию веб – сайтов, разработке баз данных и работе с ними;

- Разработка модели базы данных;
- Разработка веб-приложение для работы с базой данных, и поиска рецептов по введенным пользователем ингредиентам;
- Разработка функционала с помощью, которого пользователи смогут хранить в приложении список имеющихся у них продуктов и планировать список покупок

2. Сравнение применяемых методов исследования с известными

Для анализа существующих веб-сайтов, ориентированных на схожую тематику, было выбрано 10 сайтов, которые были показаны на первой странице поисковой системы «Яндекс» по запросу «Рецепты» и «Рецепты по ингредиентам». Из списка были удалены сайты, такие как «vk.com», «dzen.ru», «journal.tinkoff.ru», так как они не имеют нужного функционала для анализа, а лишь публикуют текстовые статьи на схожие с запросом темы. После выбора веб-сайтов, была создана таблица для их сравнения (таблица 1), в которой отмечалась возможность поиска рецептов по заданным ингредиентам («+» - данный функционал реализован на сайте, «+-» - функционал реализован с ограничениями, «-» - функционал не представлен на сайте). Также указывалось выдает ли веб-сайт рецепты, в которых используются дополнительные продукты, помимо запрашиваемых пользователем («+» - в рецептах содержатся посторонние продукты, «-» - рецепты содержат только продукты, введенные пользователем). Поле комментарий содержит информацию о дополнительных функциях, представленных на сайте.

Таблица 1. Сравнение веб-сайтов

Название сайта	**Есть ли возможность поиска по определенным продуктам**	**Выдает ли рецепты с использованием не введенных продуктов**	**Комментарий**
russianfood.com [1]	+	+	На сайте есть система регистрации пользователя. После регистрации можно оставлять

			свои записи и сообщения на сайте.
1000.menu [2]	+	+	На сайте есть возможность сохранения заметок после регистрации пользователя
eda.ru [3]	+	+	Сервис имеет интерактив в виде кулинарного журнала
povar.ru [4]	+-	+	На сайте присутствует поиск по ингредиентам, однако он происходит по фиксированному списку, который пользователь не может изменить
iamcook.ru [5]	+-	+	На сайте происходит поиск по фиксированному списку ингредиентов, в котором

			пользователь может не найти подходящий ингредиент
povarenok.ru [6]	+	+	Сайт имеет функцию "советы", что помогает пользователям, если у них появились трудности
food.ru [7]	-	+	Сайт не имеет функции поиска во введенным ингредиентам, однако из плюсов, сайт предоставляет возможность воспользоваться калькулятором калорий
kulinariya.online [8]	-	+	Сайт не имеет функции поиска во введенным ингредиентам, из плюсов можно выделить красивое

			оформление и понятный для пользователя интерфейс
say7.info [9]	-	+	Сайт не имеет функции поиска по введенным ингредиентам, а также не обладает каким-либо дополнительным функционалом
edimdoma.ru [10]	+	+	На сайте реализован проект "кулинарной школы", что помогает пользователем с большей точностью воспроизводить любимые рецепты

Исследование кулинарных сайтов позволяет сделать следующие общие выводы:

- Большинство изученных сайтов предоставляют возможность поиска рецептов по определенным продуктам, что облегчает пользовательский опыт при выборе блюд для приготовления;
- Некоторые сайты ограничивают функционал поиска по ингредиентам, предлагая лишь фиксированные списки, что может быть неудобным для пользователей с уникальными потребностями или предпочтениями;
- Отсутствие поиска по введенным ингредиентам на некоторых сайтах ограничивает их функциональность и удобство использования;
- Наличие дополнительных функций, таких как системы регистрации, сохранение заметок, кулинарные советы и калькуляторы калорий, может повысить привлекательность сайта для пользователей;
- Все сайты выдают рецепты с использованием не введенных ингредиентов, что усложняет выбор рецептов, которые пользователь может приготовить, так как ему самостоятельно приходиться отбирать только подходящие ему рецепты.

Таким образом разработка сайта, который прежде всего будет направлен на то, чтобы пользователь получал только корректный список рецептов, является важным шагом в повышении удобства использования кулинарных ресурсов в сети.

3. Представление авторского продукта

3.1. Выбор инструментов реализации

Для реализации проектной работы было принято решение использовать формат веб-сайта. Этот выбор обусловлен несколькими причинами. Во-первых, при использовании веб-сайта исключается необходимость выхода на маркеты приложений, что помогает в процессе разработки, ведь веб-приложения могут быть обновлены и модифицированы намного быстрее, чем мобильные приложения, так как изменения применяются на серверной стороне и не требуют загрузки обновлений на устройства пользователей через маркетплейсы. Во-вторых, веб-сайт доступен через браузер на любом устройстве с доступом в интернет, что расширяет потенциальную аудиторию пользователей. В-третьих, использование веб-сайта исключает необходимость загрузки и установки приложения на устройство, что упрощает процесс доступа к функционалу сервиса для конечного пользователя.

Для разработки сайта был выбран фреймворк Django. Данное решение обусловлено тем, что для использования фреймворка необходимо знание языка Python, который является самым популярным языком программирования в 2024 году (рисунок 1), который входит в стандартную программу обучения в школе [11]. Этот факт оказал значительное влияние на выбор инструмента для реализации проекта.

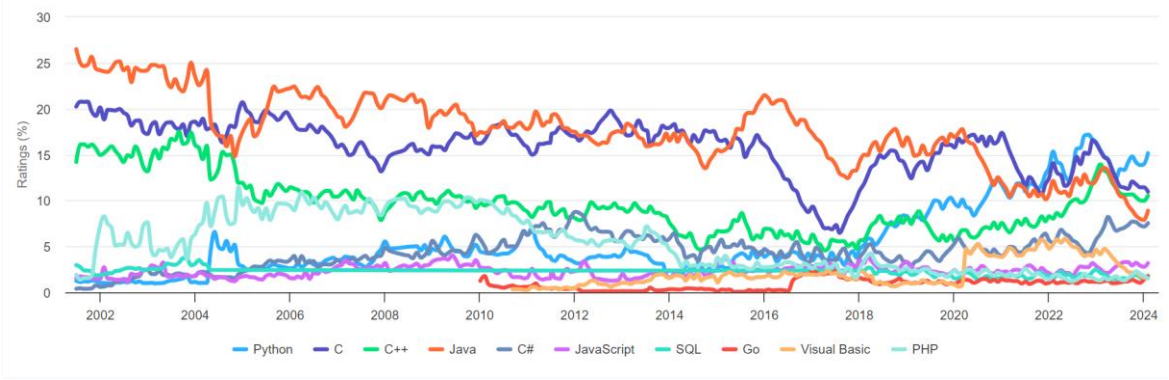


Рисунок 1 – TIOBE Programming Community Index

Также Django предоставляет широкий спектр встроенных функций, что упрощает процесс разработки, позволяя сосредоточиться на функциональности продукта. Важными встроенными функциями для данной проектной работы являются: ORM (Object-Relational Mapping) для работы с базой данных, система маршрутизации URL, встроенная аутентификация и авторизация, а также инструменты для обработки форм и административный интерфейс веб-сайта. Это делает Django мощным инструментом для быстрой и эффективной разработки веб-приложений. И последний, немаловажный факт – фреймворк имеет обширную документацию [12], что облегчает понимание функций и возможностей Django и способствует успешной реализации проекта.

3.2. Процесс разработки

В начале процесса разработки была создана база данных, в которой хранится информация о рецептах, их составляющих ингредиентах и методах приготовления. Для реализации этой задачи был выбран язык Postgresql, так как он является инструментом управления базами данных с открытым исходным кодом и свободным распространением.

Далее был проведен анализ удобства использования сервиса по поиску рецептов со стороны конечного пользователя. Этот анализ направлен на определение оптимальных способов предоставления информации и взаимодействия с интерфейсом. Основываясь на выводах этого анализа, было принято решение, что наиболее понятным интерфейсом для пользователя будет возможность ввода ингредиентов и выбора их из появляющегося списка. Такой подход позволит уменьшить количество ошибок, когда пользователи совершают опечатки в процессе ввода ингредиентов и в результате не могут найти подходящий рецепт. Для реализации данного функционала использовалась библиотека Select2 [13].

Затем был разработан алгоритм поиска рецептов, который опирается на сопоставление списка ингредиентов, введенного пользователем, с базой данных. Для этого создается автоматическое поле в таблице ингредиентов, необходимых для конкретного рецепта и в него вносится количество каждого продукта по

запросу пользователя. Если в данном поле есть хотя бы один элемент с количеством 0, то это означает, что данного ингредиента нет в запросе пользователя и такой рецепт выводится не будет. Этот алгоритм позволяет определить, какие рецепты подходят под доступные ингредиенты пользователя.

3.3. Результат разработки

В итоге проектной работы был разработан веб-сайт, полностью соответствующий заявленному функционалу, определенному на начальном этапе проекта. Сайт предоставляет пользователям удобный доступ к рецептам на основе введенных ими ингредиентов. Пользователи могут легко искать рецепты, используя доступные им продукты, без необходимости покупки дополнительных ингредиентов.

В процессе разработки были реализованы следующие функции:

- Поиск рецептов по ингредиентам: система позволяет пользователям вводить список имеющихся у них ингредиентов, на основе которого предлагаются подходящие рецепты. Ввод реализован в удобном для пользователя формате, с появлением подсказок. Пример: при вводе продуктов «Яйцо», «Молоко», «Мука» (рисунок 2), пользователю выводятся рецепты: «Оладушки», «Омлет» и «Вареные яйца» (рисунок 3). Но если пользователь ограничит список только двумя наименованиями: «Яйцо» и «Молоко», то система не покажет рецепт «Оладушки», так как он имеет дополнительный ингредиент, не введенный пользователем (рисунок 4).

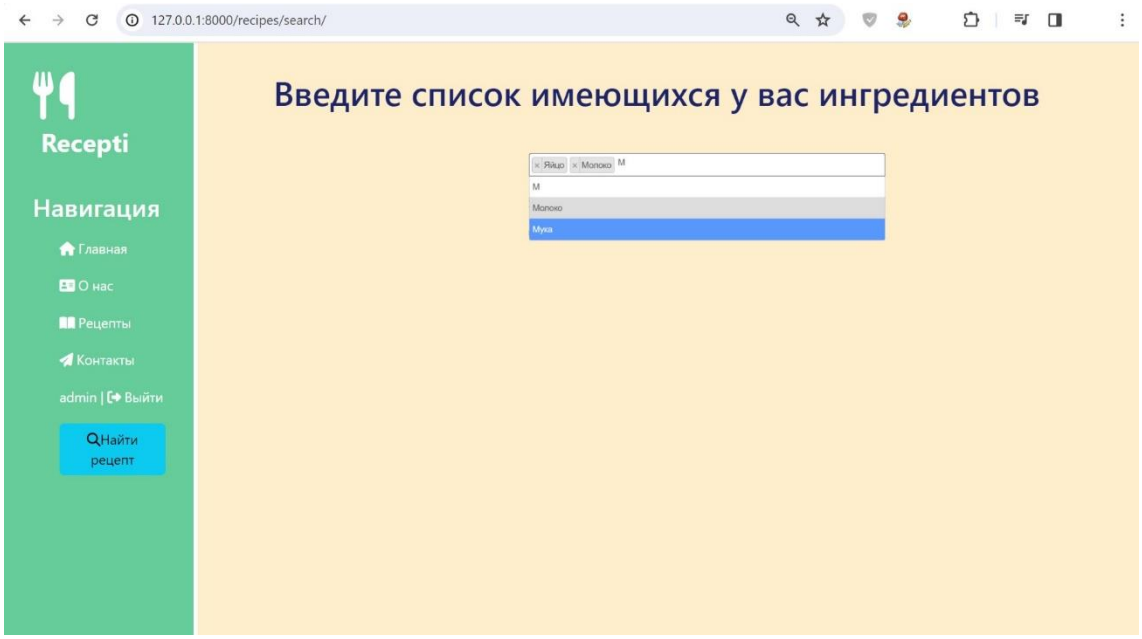


Рисунок 2 – Ввод ингредиентов

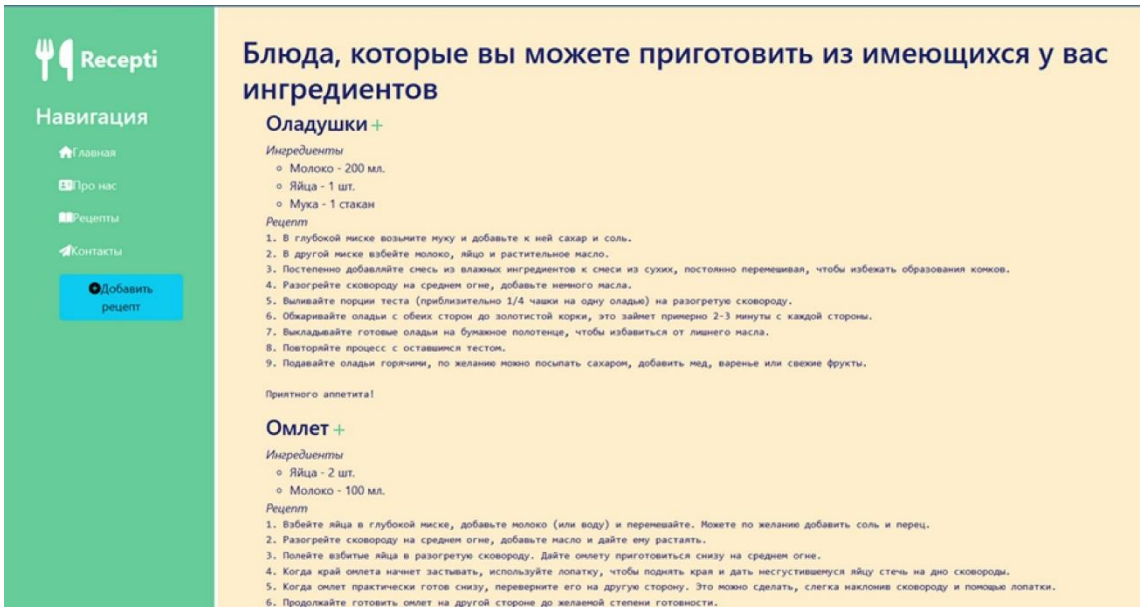


Рисунок 3 – Вывод рецептов при наличии в списке ингредиента «Мука»

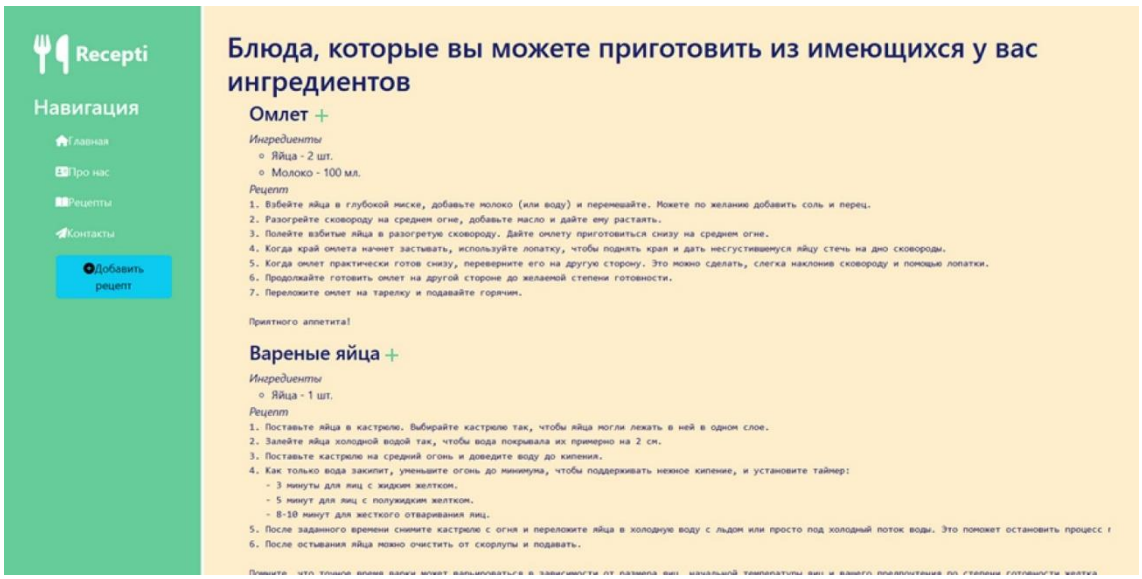


Рисунок 4 – Вывод рецептов при отсутствии в списке ингредиента «Мука»

- Регистрация и аутентификация пользователей: пользователи могут создать учетную запись на сайте и войти в систему для получения доступа к функции ведения личного списка продуктов (рисунок 5).

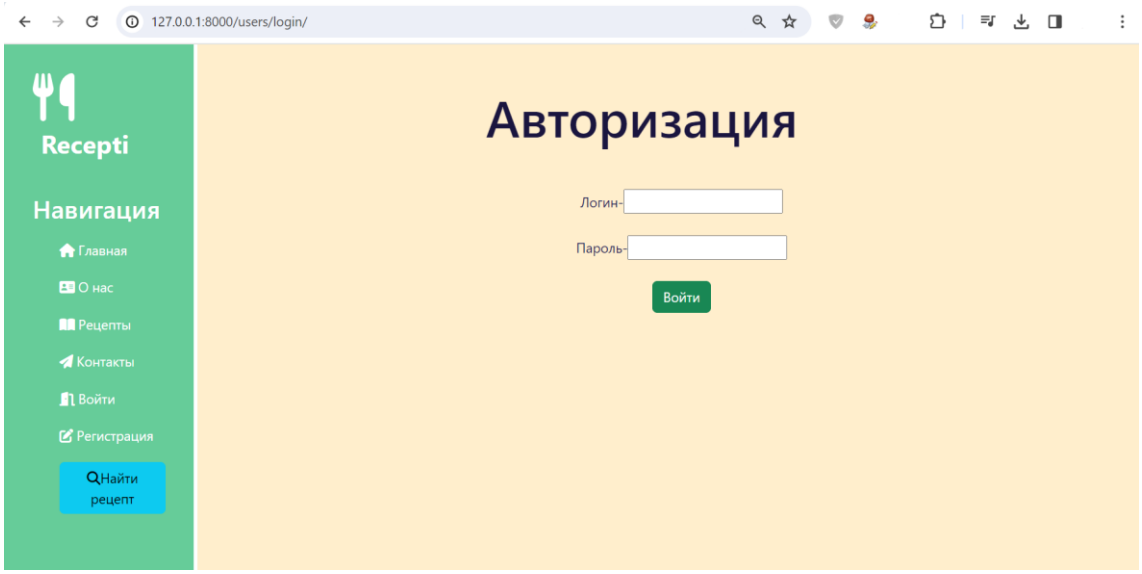


Рисунок 5 – Экран авторизации

- Срок годности продуктов: пользователи могут добавлять срок годности к ингредиентам, которые есть у них в списке продуктов. Эта функция позволяет сортировать рецепты, которые может приготовить пользователь, по важности. Первыми будут высвечиваться рецепты, в состав которых входят ингредиенты, у которых скоро закончится срок годности. Данный функционал поможет пользователям оптимизировать

свои траты на продукты и избежать больших объемов выброшенных продуктов.

- Планирование покупки продуктов: пользователи могут составлять список продуктов, которые им необходимо приобрести для приготовления выбранных рецептов. При выборе рецепта из списка, пользователь, может добавить его в личный кабинет, нажав «+». В личном кабинете пользователь может посмотреть список своих продуктов, который он добавил заранее, а также список к покупке, который формируется на основе выбранных рецептов и выводит все продукты, которые есть в рецептах, но отсутствуют в личном списке пользователя (рисунок 6).

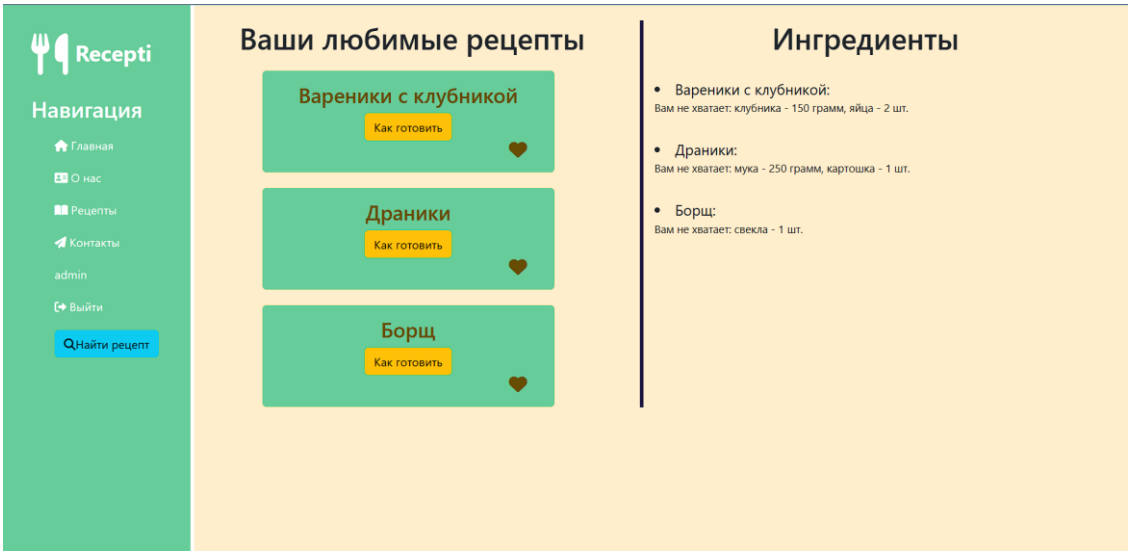


Рисунок 6 – Личный кабинет с добавленными рецептами и список покупок

4. Выводы и перспективы улучшения продукта

В результате выполнения проектной работы была найдена проблема и проанализирована предметная область. Это позволило прийти к решению, о создании веб-сайта. Был разработан веб-сайт, который эффективно решает поставленные задачи, обеспечивая пользователям доступ к широкому выбору рецептов на основе имеющихся у них продуктов.

В перспективе веб-сайт можно улучшить путем добавления оптимизации для работы на различных устройствах, что позволит увеличить количество пользователей. Данный функционал упростит использование веб-ресурса пользователю, так как ему будет удобно пользоваться любым из доступных устройств.

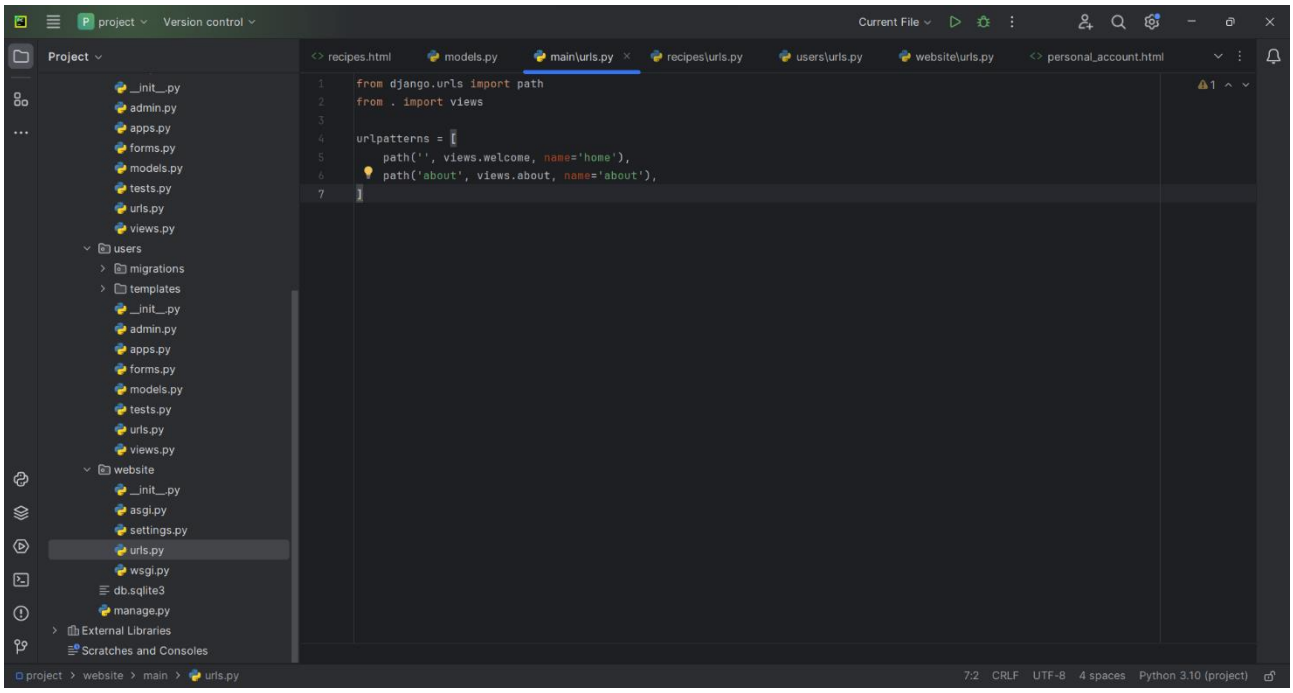
5. Список используемой литературы

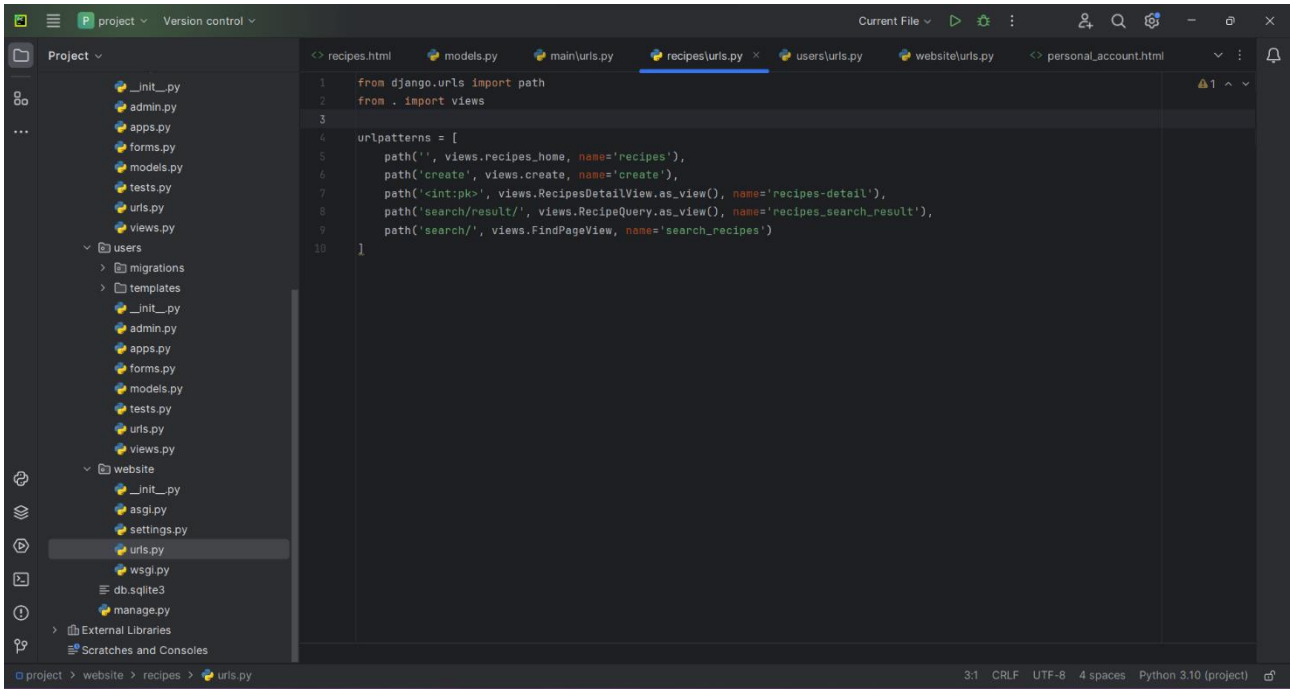
Электронные ресурсы

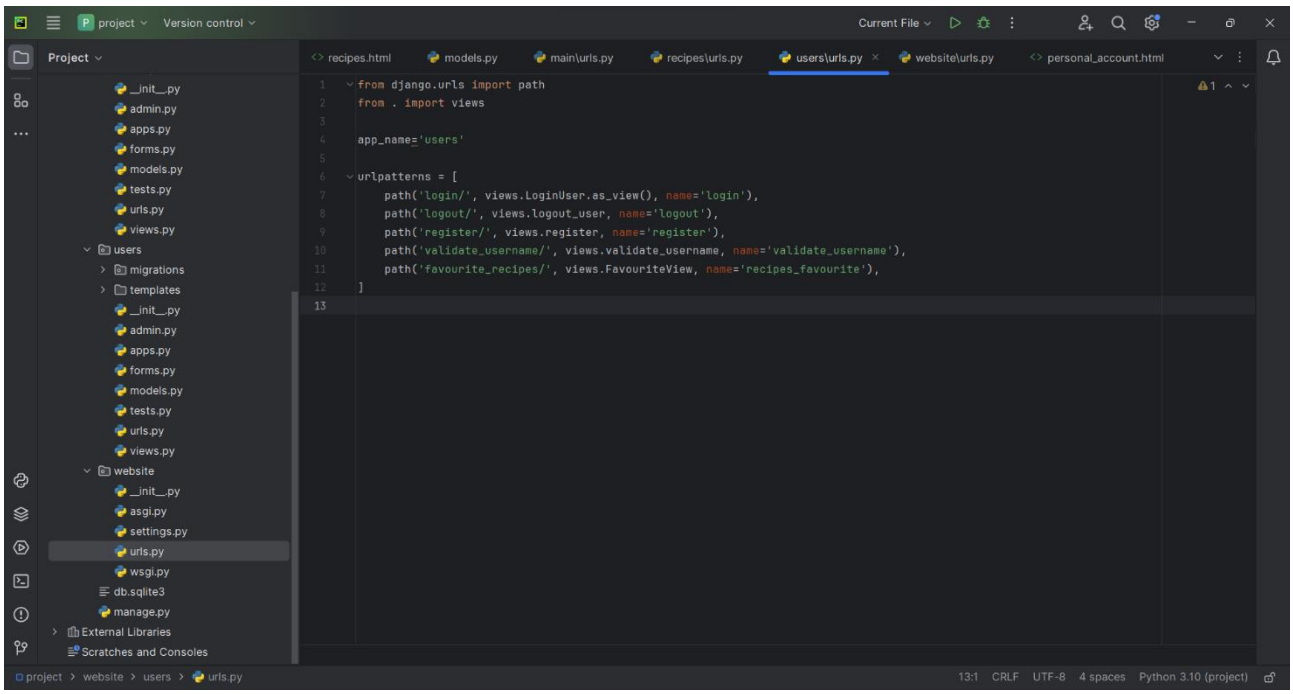
1. RussianFood.com [Электронный ресурс] // URL: https://www.russianfood.com/. Режим доступа: свободный. (Дата обращения 10.02.2024)
2. 1000.menu [Электронный ресурс] // URL: https://1000.menu/?ysclid=lthnyvqhj1565559931. Режим доступа: свободный. (Дата обращения 10.02.2024)
3. eda.ru [Электронный ресурс] // URL: https://eda.ru/. Режим доступа: свободный. (Дата обращения 10.02.2024)
4. povar.ru [Электронный ресурс] // URL: https://povar.ru/?ysclid=ltho71lgp0289007872. Режим доступа: свободный. (Дата обращения 10.02.2024)
5. iamcook.ru [Электронный ресурс] // URL: https://www.iamcook.ru/?ysclid=ltho9edgw4548590158. Режим доступа: свободный. (Дата обращения 10.02.2024)
6. povarenok.ru [Электронный ресурс] // URL: https://www.povarenok.ru/. Режим доступа: свободный. (Дата обращения 10.02.2024)
7. food.ru [Электронный ресурс] // URL: https://food.ru/?ysclid=lthocdo7d2178349168. Режим доступа: свободный. (Дата обращения 10.02.2024)
8. kulinariya.online [Электронный ресурс] // URL: https://kulinariya.online/?ysclid=lthodvhp9257892507. Режим доступа: свободный. (Дата обращения 10.02.2024)
9. say7.info [Электронный ресурс] // URL: https://www.say7.info/?ysclid=lthof5nlvc750873397. Режим доступа: свободный. (Дата обращения 10.02.2024)
10. edimdoma.ru [Электронный ресурс] // URL: https://www.edimdoma.ru/. Режим доступа: свободный. (Дата обращения 10.02.2024)
11. TIOBE Index [Электронный ресурс] // TIOBE Index for February 2024. URL: https://www.tiobe.com/tiobe-index. Режим доступа: свободный. (Дата обращения: 10.02.2024)
12. Django [Электронный ресурс] // Django documentation. URL: https://docs.djangoproject.com/en/5.0/. Режим доступа: свободный. (Дата обращения: 11.02.2024)
13. Select2 [Электронный ресурс] // URL: https://select2.org/. Режим доступа: свободный. (Дата обращения: 11.02.2024)

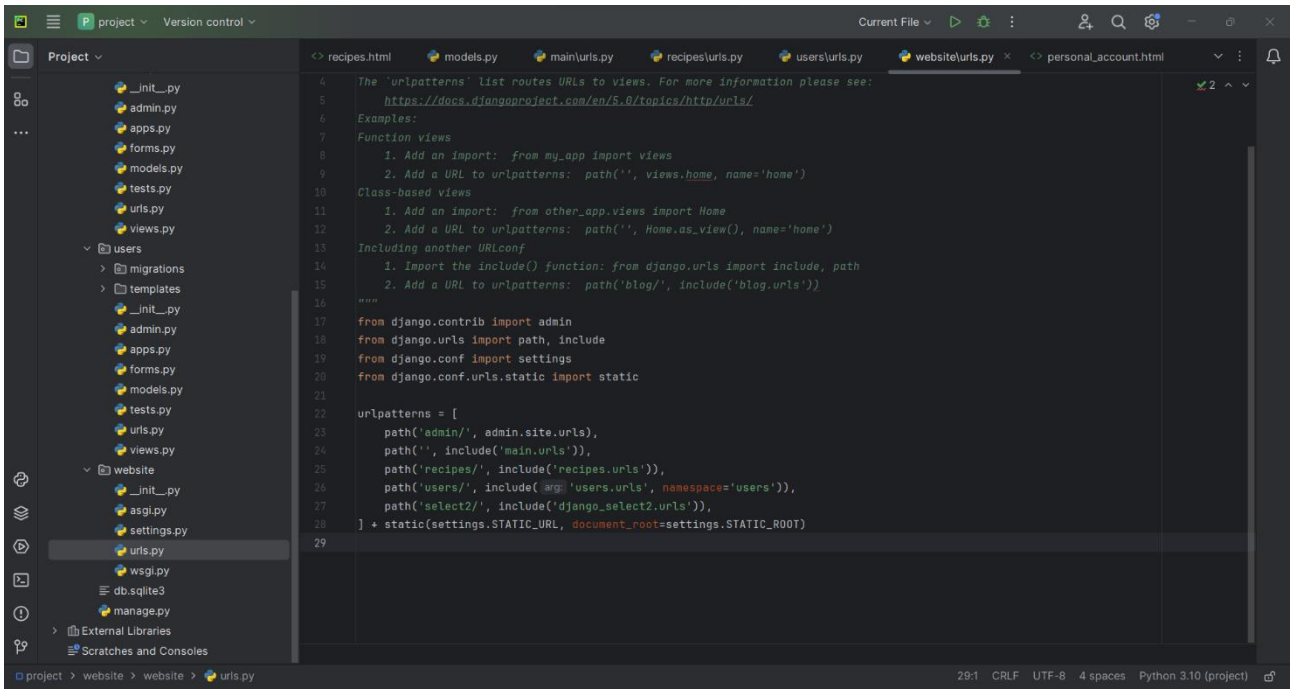
Приложение

Файлы urls.py

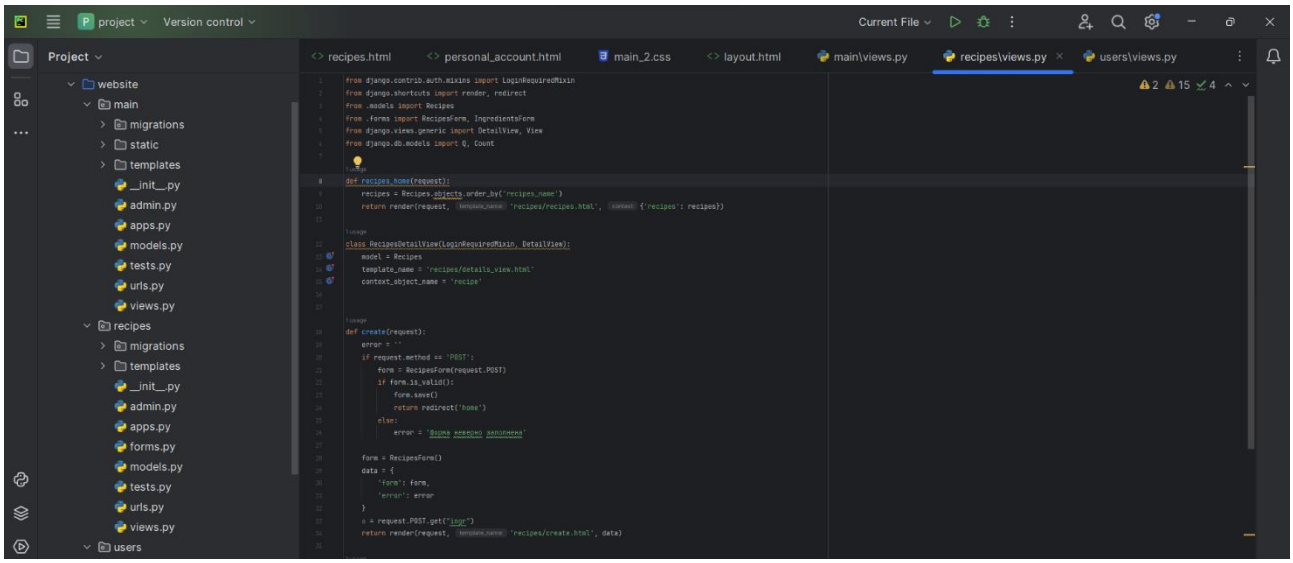


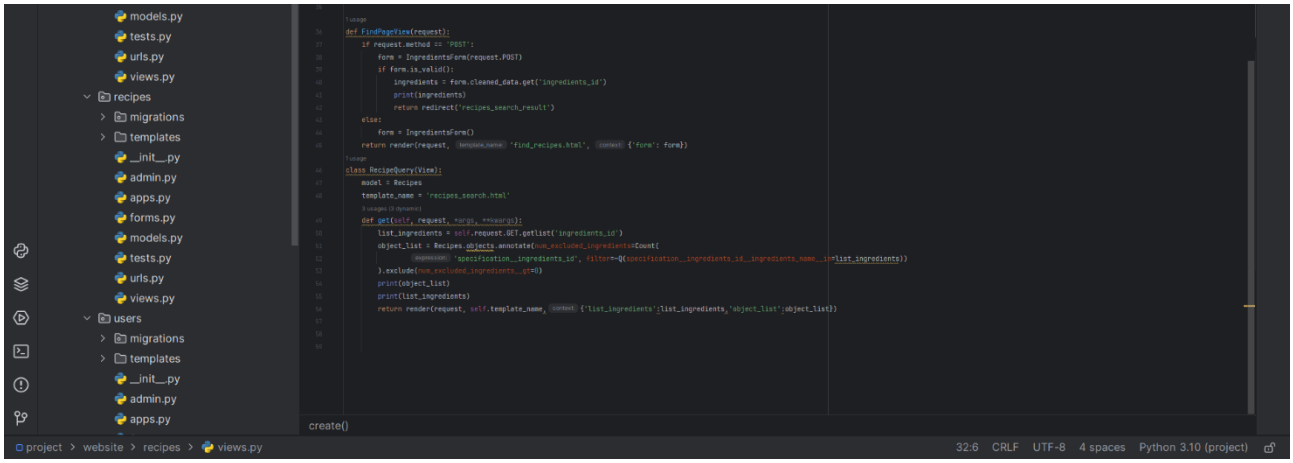


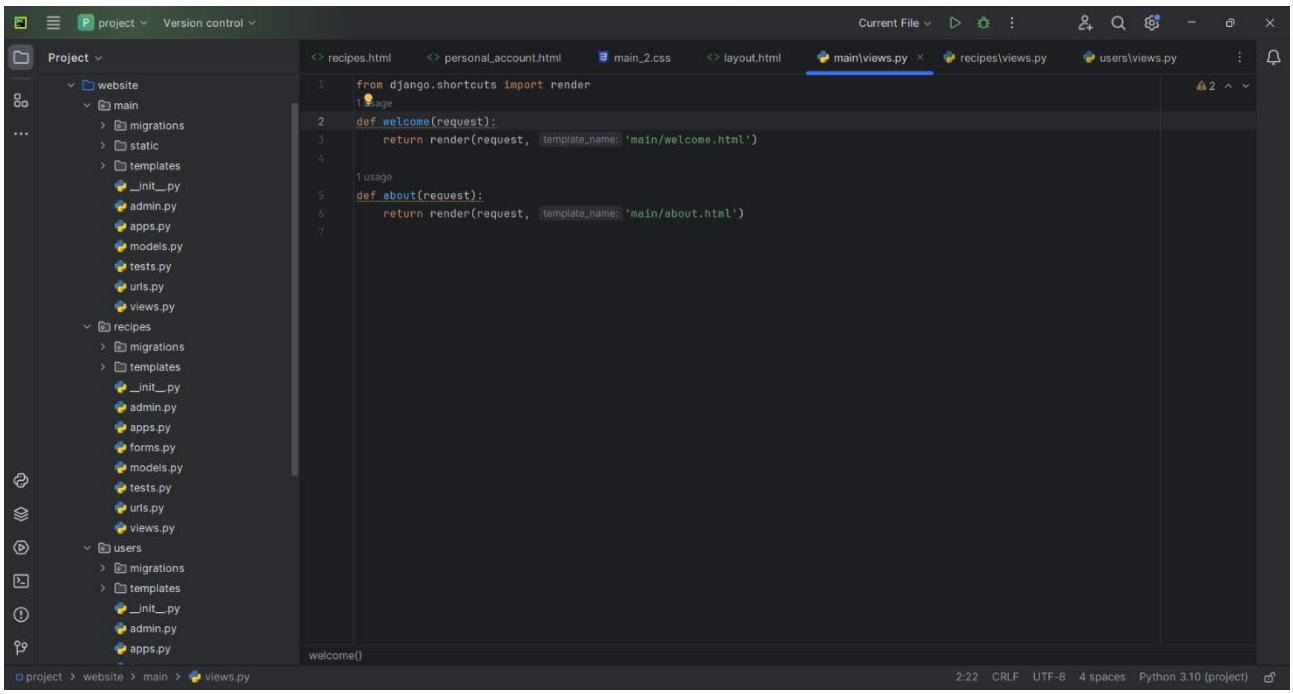

project
Version control
Current File
Project
recipes.html
models.py
main\urls.py
recipes\urls.py
users\urls.py
website\urls.py
personal_account.html
from django.urls import path
from . import views
app_name='users'
urlpatterns = [
path('login/', views.LoginUser.as_view(), name='login'),
path('logout/', views.logout_user, name='logout'),
path('register/', views.register, name='register'),
path('validate_username/', views.validate_username, name='validate_username'),
path('favourite_recipes/', views.FavouriteView, name='recipes_favourite'),
]
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
website
asgi.py
settings.py
wsgi.py
db.sqlite3
manage.py
External Libraries
Scratches and Consoles
project › website › users › urls.py
13:1 CRLF UTF-8 4 spaces Python 3.10 (project)

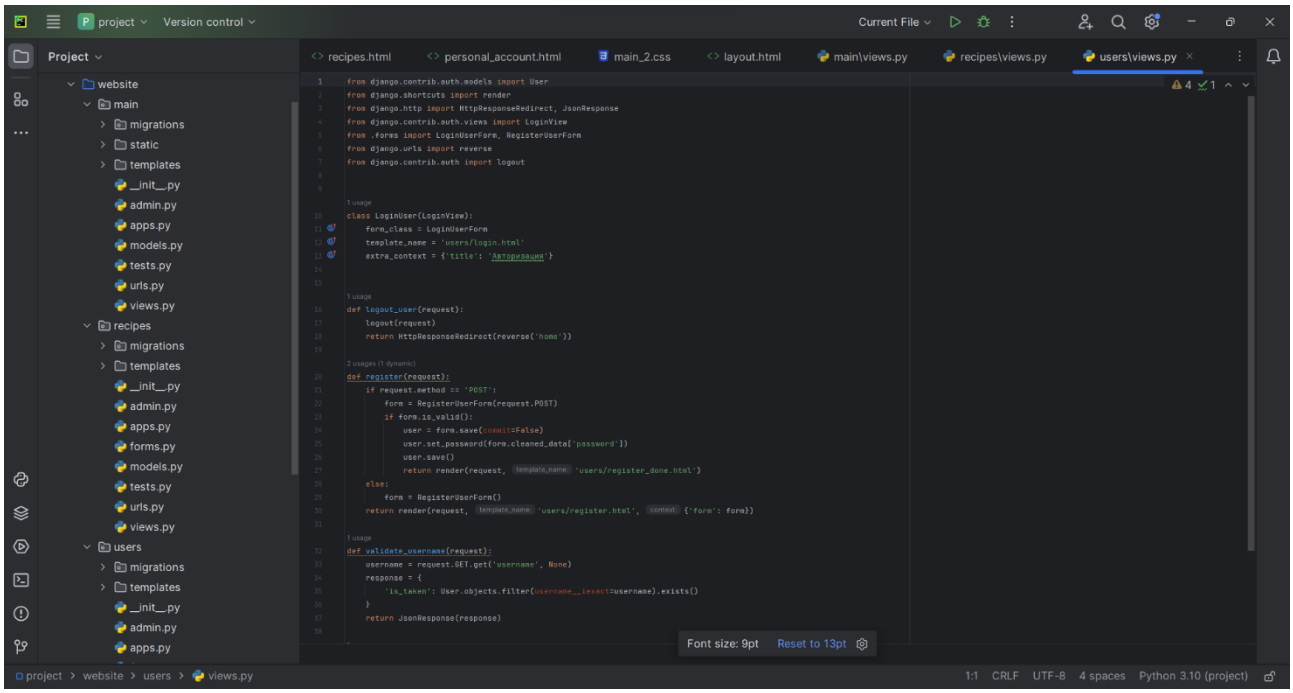

project
Version control
Current File
Project
recipes.html
models.py
main\urls.py
recipes\urls.py
users\urls.py
website\urls.py
personal_account.html
The `urlpatterns` list routes URLs to views. For more information please see:
https://docs.djangoproject.com/en/5.0/topics/http/urls/
Examples:
Function views
1. Add an import: from my_app import views
2. Add a URL to urlpatterns: path('', views.home, name='home')
Class-based views
1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
Including another URLconf
1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include
from django.conf import settings
from django.conf.urls.static import static
urlpatterns = [
path('admin/', admin.site.urls),
path('', include('main.urls')),
path('recipes/', include('recipes.urls')),
path('users/', include(arg: 'users.urls', namespace='users')),
path('select2/', include('django_select2.urls')),
] + static(settings.STATIC_URL, document_root=settings.STATIC_ROOT)
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
website
asgi.py
settings.py
wsgi.py
db.sqlite3
manage.py
External Libraries
Scratches and Consoles
project › website › website › urls.py
29:1 CRLF UTF-8 4 spaces Python 3.10 (project)

Файлы views.py

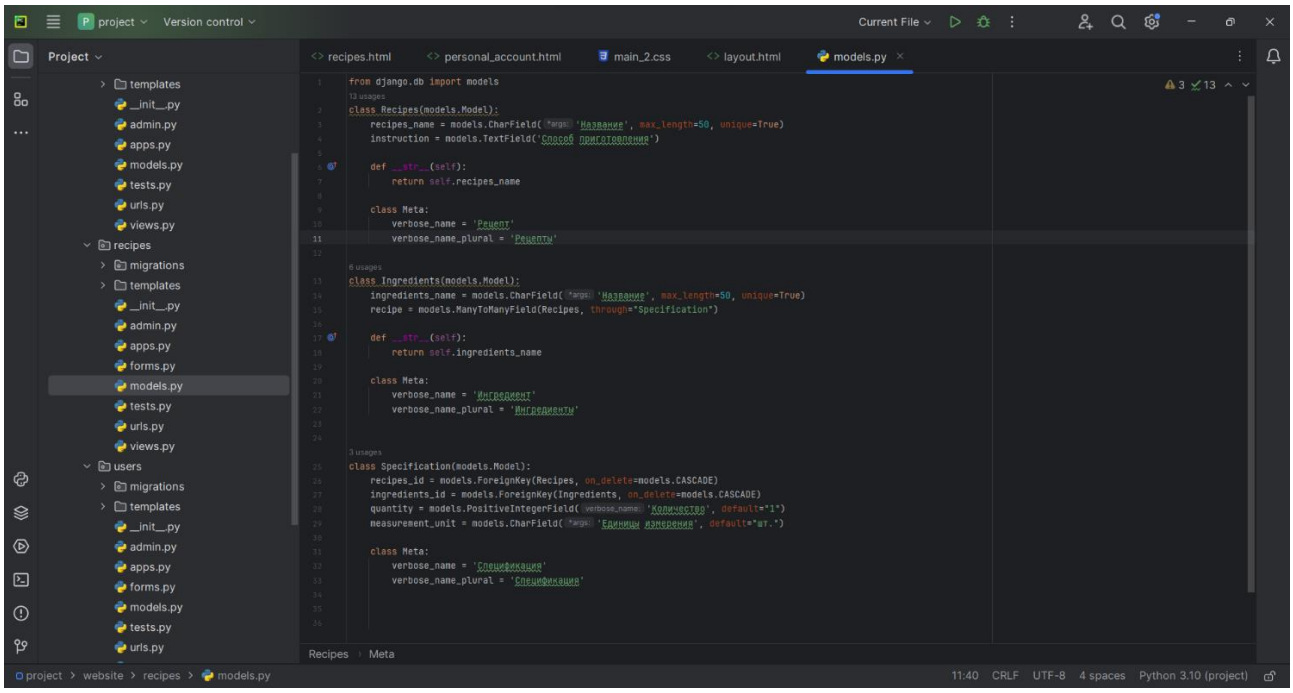




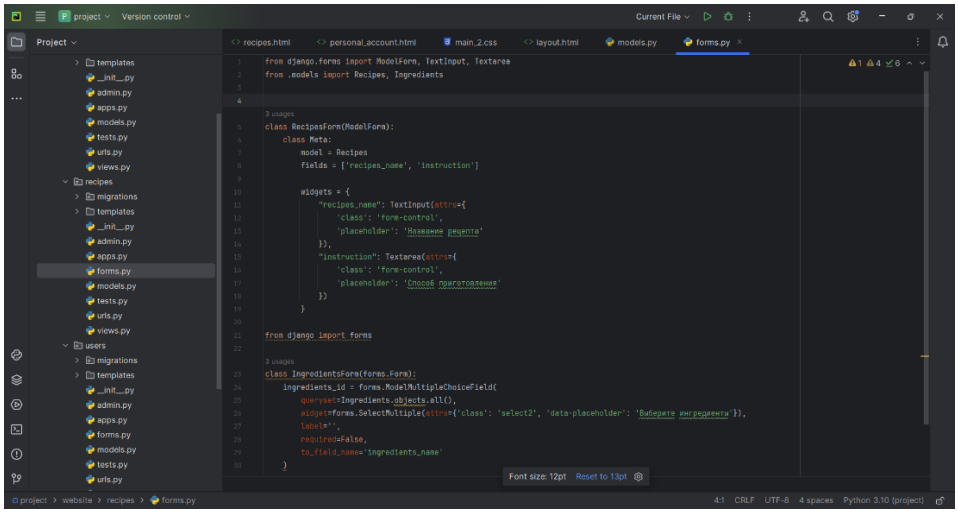

project
Version control
Current File
Project
website
main
migrations
static
templates
__init__.py
admin.py
apps.py
models.py
tests.py
urls.py
views.py
recipes
migrations
templates
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
__init__.py
admin.py
apps.py
recipes.html
personal_account.html
main_2.css
layout.html
main\views.py
recipes\views.py
users\views.py
from django.shortcuts import render
1 usage
def welcome(request):
return render(request, template_name: 'main/welcome.html')
1 usage
def about(request):
return render(request, template_name: 'main/about.html')
welcome()
project › website › main › views.py
2:22 CRLF UTF-8 4 spaces Python 3.10 (project)

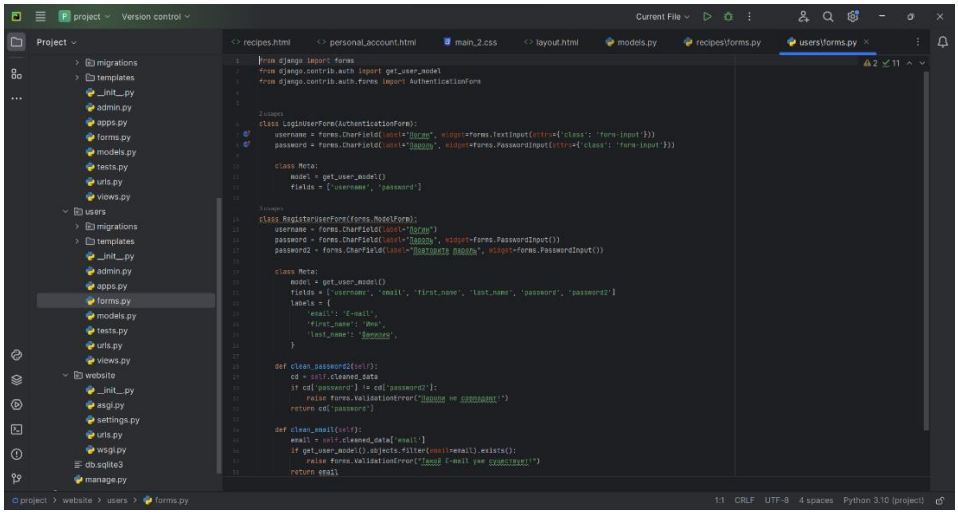

project
Version control
Current File
Project
website
main
migrations
static
templates
__init__.py
admin.py
apps.py
models.py
tests.py
urls.py
views.py
recipes
migrations
templates
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
__init__.py
admin.py
apps.py
recipes.html
personal_account.html
main_2.css
layout.html
main\views.py
recipes\views.py
users\views.py
from django.contrib.auth.models import User
from django.shortcuts import render
from django.http import HttpResponseRedirect, JsonResponse
from django.contrib.auth.views import LoginView
from .forms import LoginUserForm, RegisterUserForm
from django.urls import reverse
from django.contrib.auth import logout
1 usage
class LoginUser(LoginView):
form_class = LoginUserForm
template_name = 'users/login.html'
extra_context = {'title': 'Авторизация'}
1 usage
def logout_user(request):
logout(request)
return HttpResponseRedirect(reverse('home'))
2 usages (1 dynamic)
def register(request):
if request.method == 'POST':
form = RegisterUserForm(request.POST)
if form.is_valid():
user = form.save(commit=False)
user.set_password(form.cleaned_data['password'])
user.save()
return render(request, template_name: 'users/register_done.html')
else:
form = RegisterUserForm()
return render(request, template_name: 'users/register.html', context: {'form': form})
1 usage
def validate_username(request):
username = request.GET.get('username', None)
response = {
'is_taken': User.objects.filter(username__iexact=username).exists()
}
return JsonResponse(response)
Font size: 9pt
Reset to 13pt
project › website › users › views.py
1:1 CRLF UTF-8 4 spaces Python 3.10 (project)

Модель базы данных

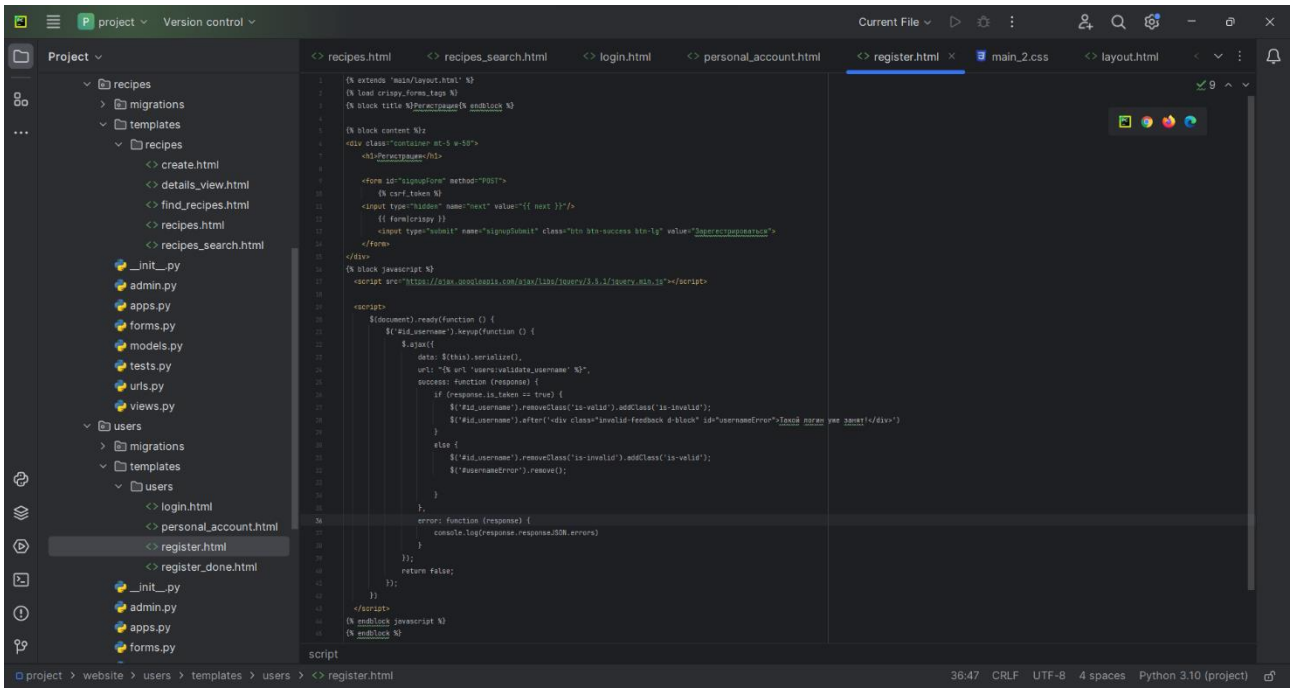


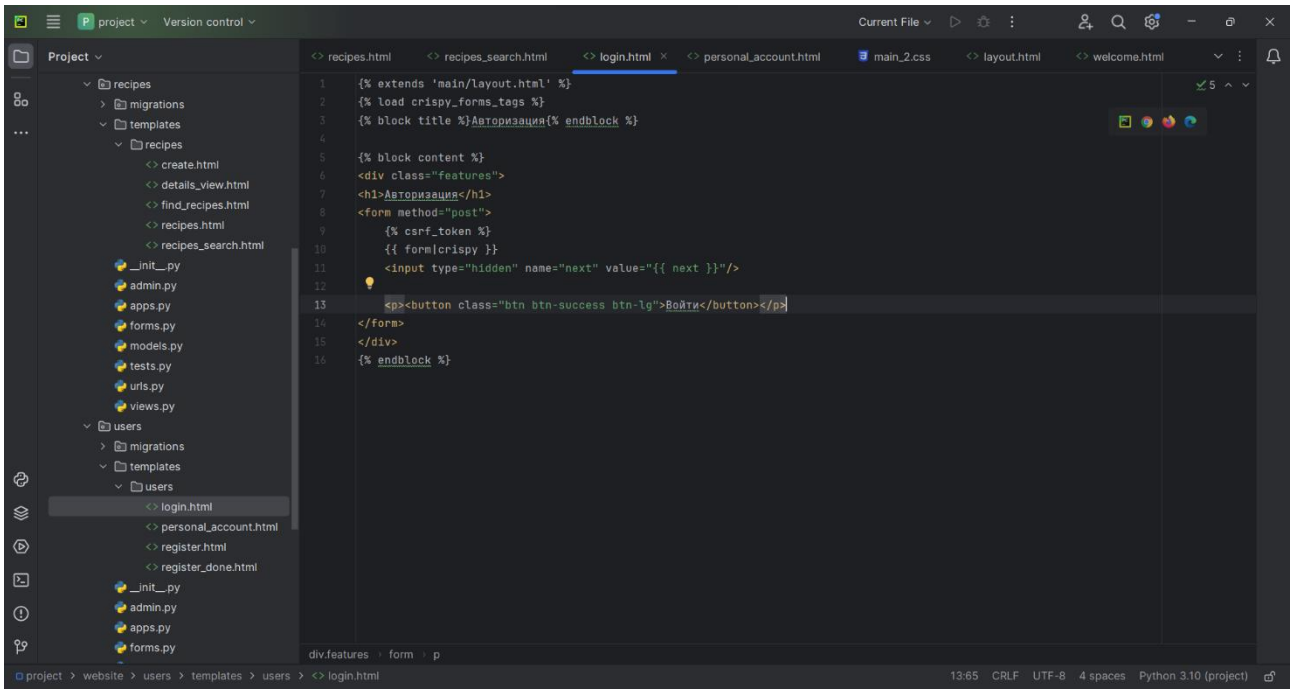
Файлы forms.py





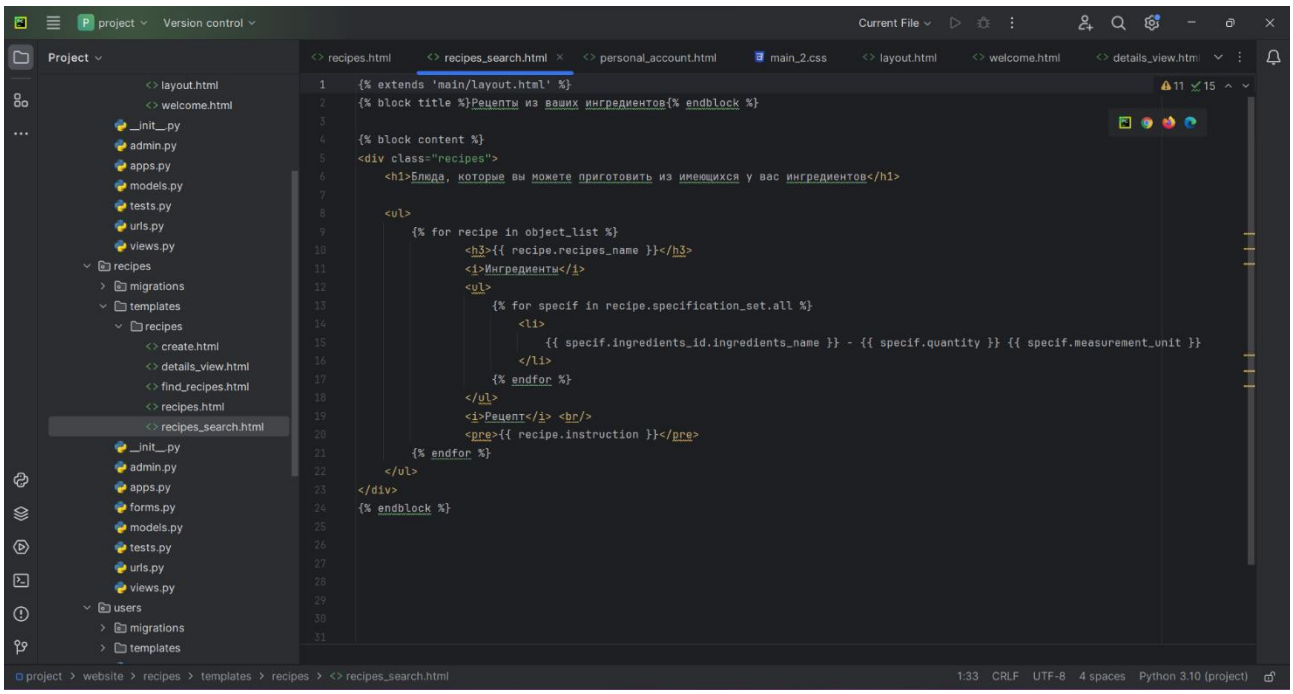
HTML файлы


project
Version control
Current File
Project
recipes.html
recipes_search.html
login.html
personal_account.html
register.html
main_2.css
layout.html
recipes
migrations
templates
recipes
create.html
details_view.html
find_recipes.html
recipes.html
recipes_search.html
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
users
login.html
personal_account.html
register.html
register_done.html
__init__.py
admin.py
apps.py
forms.py
script
project › website › users › templates › users › register.html
36:47 CRLF UTF-8 4 spaces Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
recipes_search.html
login.html
personal_account.html
main_2.css
layout.html
welcome.html
recipes
migrations
templates
recipes
create.html
details_view.html
find_recipes.html
recipes.html
recipes_search.html
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
users
migrations
templates
users
login.html
personal_account.html
register.html
register_done.html
__init__.py
admin.py
apps.py
forms.py
{% extends 'main/layout.html' %}
{% load crispy_forms_tags %}
{% block title %}Авторизация{% endblock %}
{% block content %}

<h1>Авторизация</h1>
<form method="post">
{% csrf_token %}
{{ form|crispy }}
<input type="hidden" name="next" value="{{ next }}"/>
<button class="btn btn-success btn-lg">Войти</button>
</form>

{% endblock %}
div.features › form › p
project › website › users › templates › users › login.html
13:65 CRLF UTF-8 4 spaces Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
recipes_search.html
personal_account.html
main_2.css
layout.html
welcome.html
details_view.html
{% extends 'main/layout.html' %}
{% block title %}Рецепты из ваших ингредиентов{% endblock %}
{% block content %}

<h1>Блюда, которые вы можете приготовить из имеющихся у вас ингредиентов</h1>

{% for recipe in object_list %}
<h3>{{ recipe.recipes_name }}</h3>
Ингредиенты

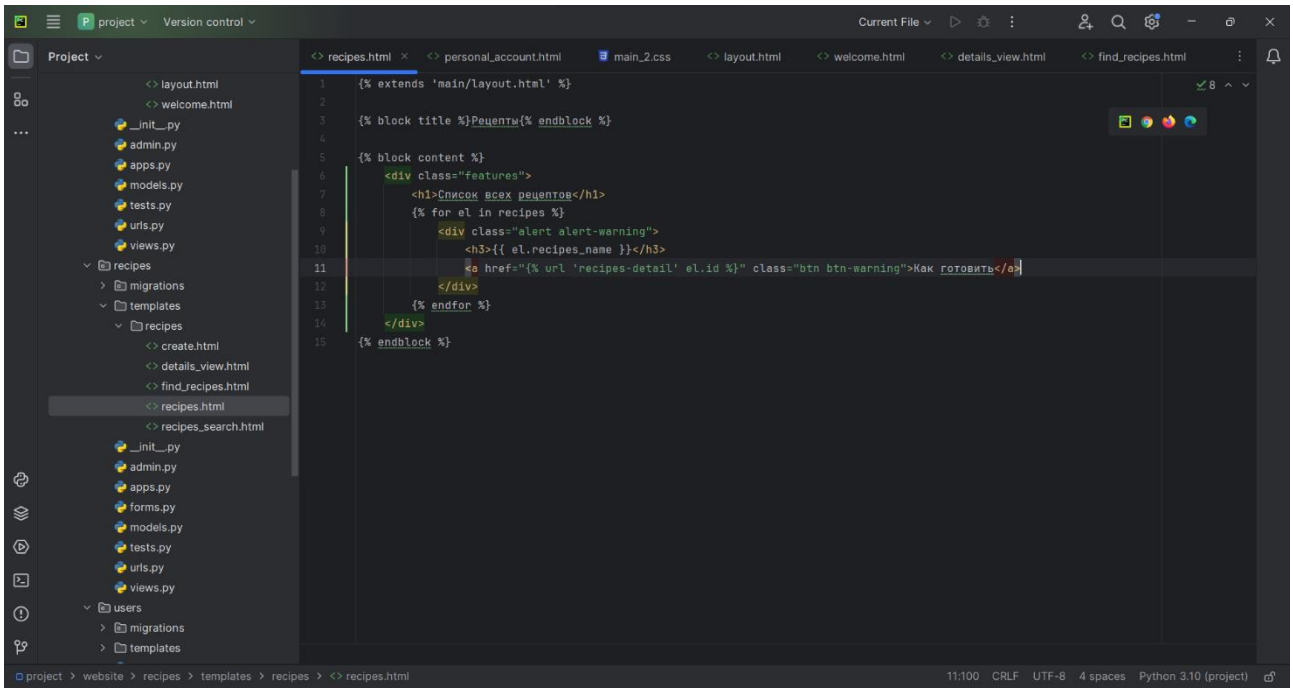
{% for specif in recipe.specification_set.all %}

{{ specif.ingredients_id.ingredients_name }} - {{ specif.quantity }} {{ specif.measurement_unit }}

{% endfor %}

Рецепт
<pre>{{ recipe.instruction }}</pre>
{% endfor %}

{% endblock %}
project > website > recipes > templates > recipes > recipes_search.html
1:33 CRLF UTF-8 4 spaces Python 3.10 (project)

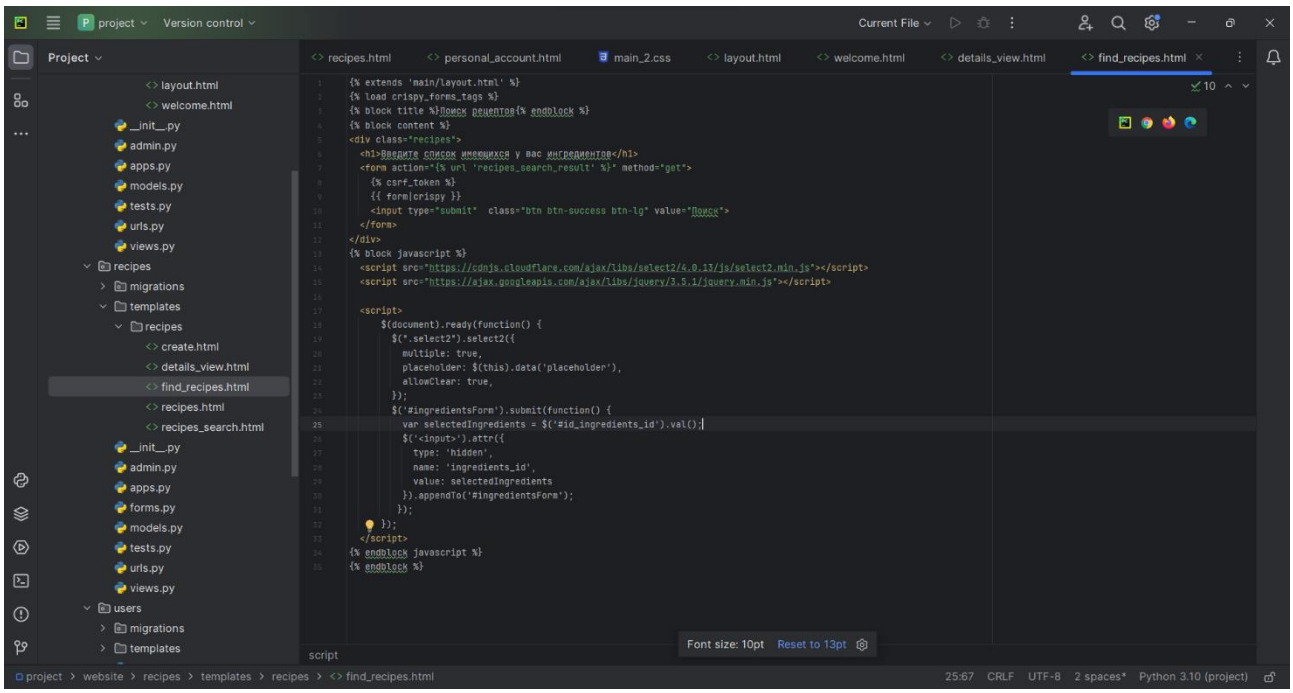

project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
welcome.html
details_view.html
find_recipes.html
{% extends 'main/layout.html' %}
{% block title %}Рецепты{% endblock %}
{% block content %}

<h1>Список всех рецептов</h1>
{% for el in recipes %}

<h3>{{ el.recipes_name }}</h3>
Как готовить

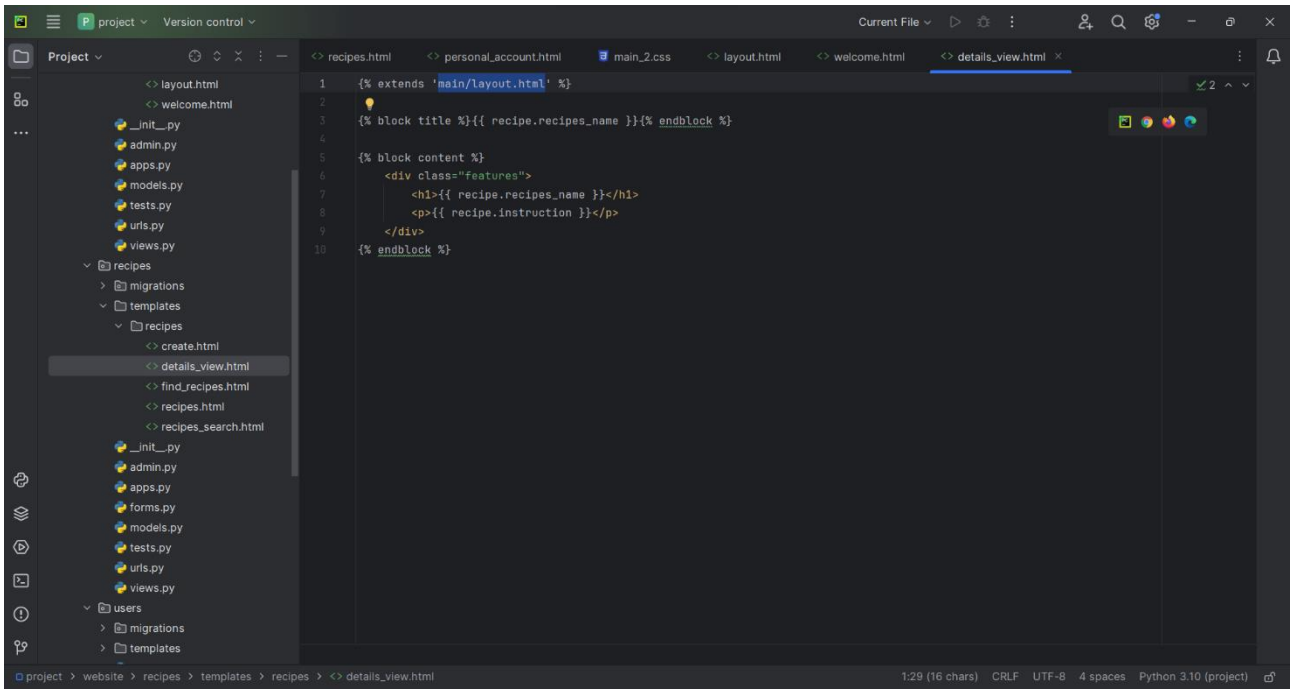
{% endfor %}

{% endblock %}
project > website > recipes > templates > recipes > recipes.html
11:100 CRLF UTF-8 4 spaces Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
welcome.html
details_view.html
find_recipes.html
{% extends 'main/layout.html' %}
{% load crispy_forms_tags %}
{% block title %}Поиск рецептов{% endblock %}
{% block content %}

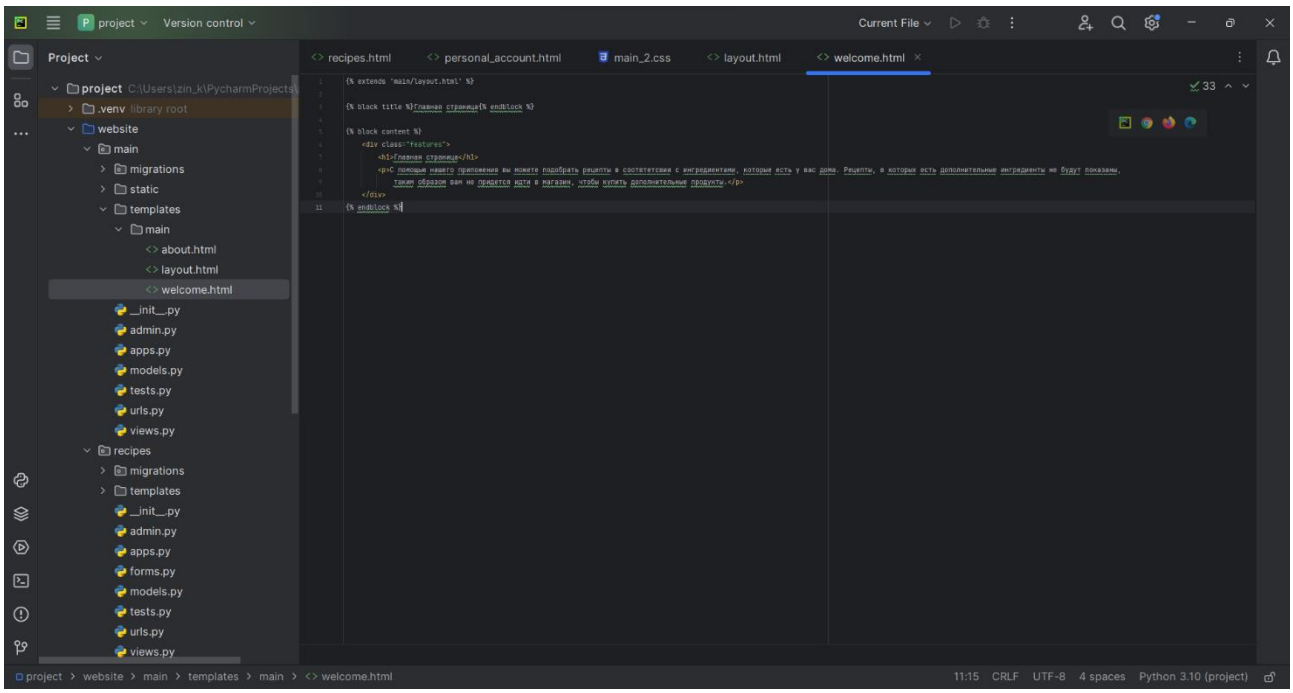
<h1>Введите список имеющихся у вас ингредиентов</h1>
<form action="{% url 'recipes_search_result' %}" method="get">
{% csrf_token %}
{{ form|crispy }}
<input type="submit" class="btn btn-success btn-lg" value="Поиск">
</form>

{% block javascript %}
<script src="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.13/js/select2.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script>
$(document).ready(function() {
$(".select2").select2({
multiple: true,
placeholder: $(this).data('placeholder'),
allowClear: true,
});
$('#ingredientsForm').submit(function() {
var selectedIngredients = $('#id_ingredients_id').val();
$('<input>').attr({
type: 'hidden',
name: 'ingredients_id',
value: selectedIngredients
}).appendTo('#ingredientsForm');
});
});
</script>
{% endblock javascript %}
{% endblock %}
Font size: 10pt
Reset to 13pt
script
project › website › recipes › templates › recipes › find_recipes.html
25:67
CRLF
UTF-8
2 spaces*
Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
welcome.html
details_view.html
{% extends 'main/layout.html' %}
{% block title %}{{ recipe.recipes_name }}{% endblock %}
{% block content %}

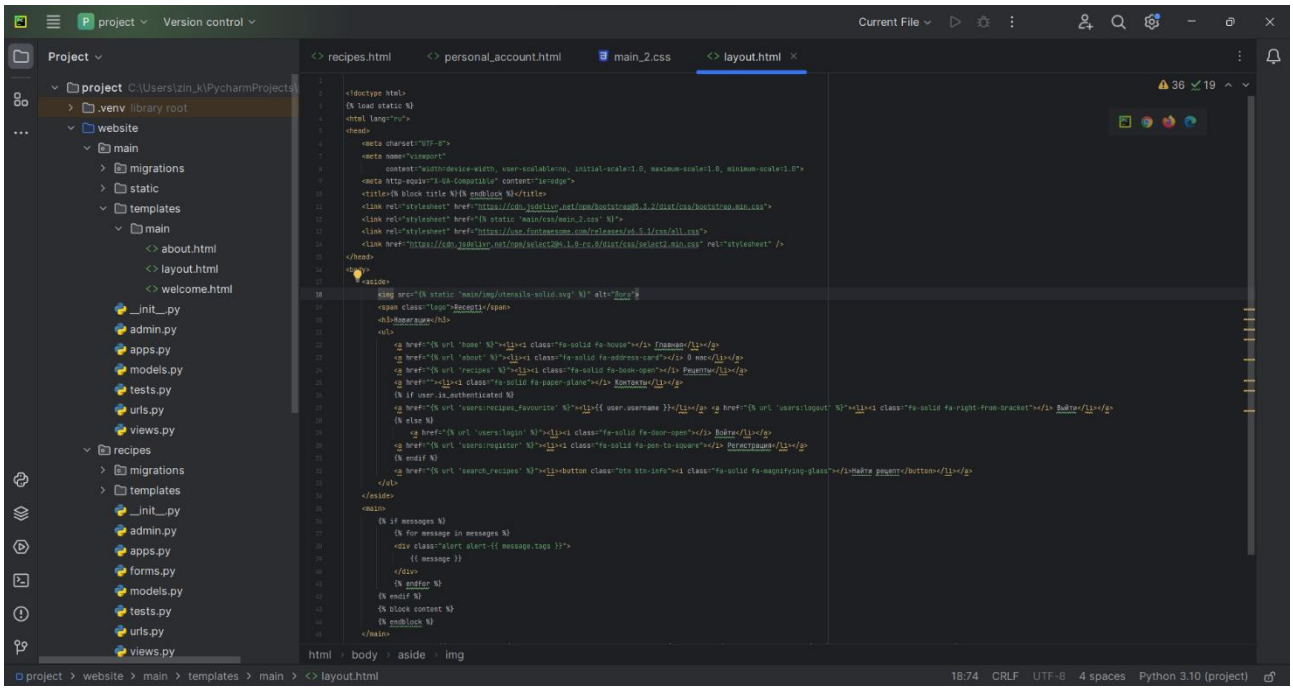
<h1>{{ recipe.recipes_name }}</h1>
{{ recipe.instruction }}

{% endblock %}
project › website › recipes › templates › recipes › details_view.html
1:29 (16 chars)
CRLF
UTF-8
4 spaces
Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
welcome.html
project C:\Users\zin_k\PycharmProjects\
.venv library root
website
main
migrations
static
templates
main
about.html
layout.html
welcome.html
__init__.py
admin.py
apps.py
models.py
tests.py
urls.py
views.py
recipes
migrations
templates
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
{% extends 'main/layout.html' %}
{% block title %}Главная страница{% endblock %}
{% block content %}

<h1>Главная страница</h1>
С помощью нашего приложения вы можете подобрать рецепты в соответствии с ингредиентами, которые есть у вас дома. Рецепты, в которых есть дополнительные ингредиенты не будут показаны,
таким образом вам не придется идти в магазин, чтобы купить дополнительные продукты.

{% endblock %}
33
project > website > main > templates > main > welcome.html
11:15 CRLF UTF-8 4 spaces Python 3.10 (project)

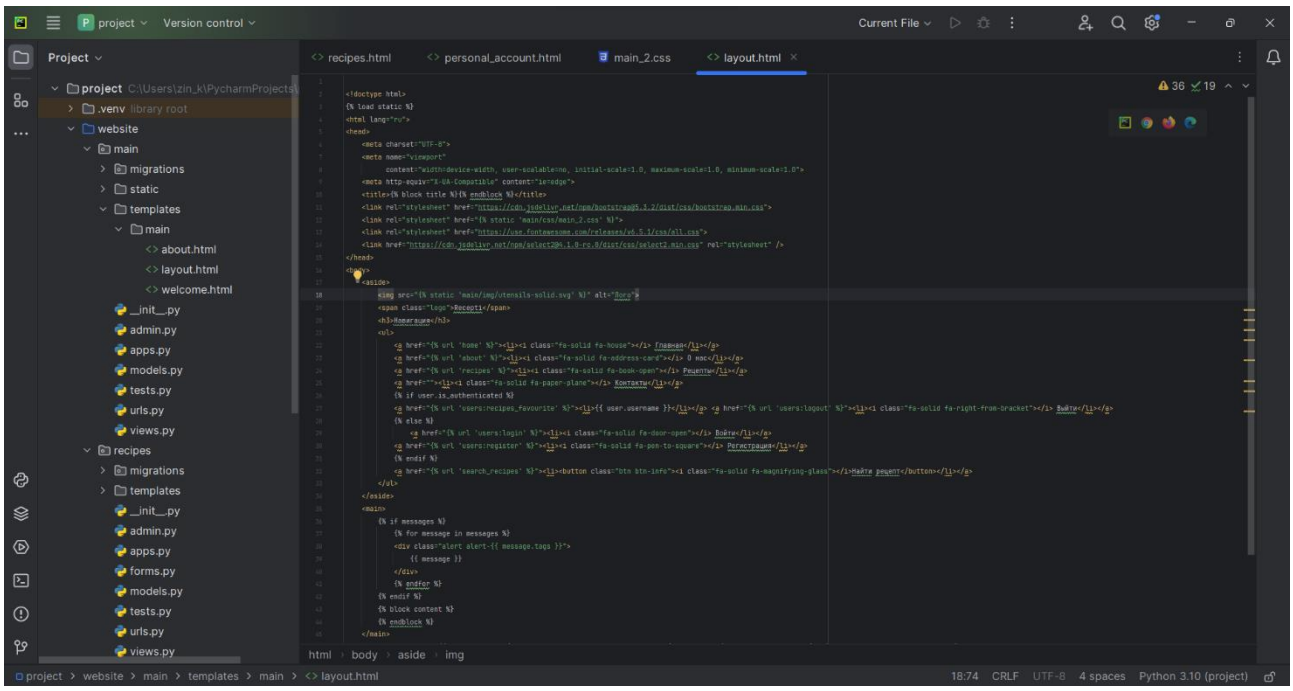

project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
project C:\Users\zin_k\PycharmProjects\
.venv library root
website
main
migrations
static
templates
main
about.html
layout.html
welcome.html
__init__.py
admin.py
apps.py
models.py
tests.py
urls.py
views.py
recipes
migrations
templates
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
<!doctype html>
{% load static %}
<html lang="ru">
<head>
<meta charset="UTF-8">
<meta name="viewport"
content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>{% block title %}{% endblock %}</title>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css">
<link rel="stylesheet" href="{% static 'main/css/main_2.css' %}">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v6.5.1/css/all.css">
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
</head>
<body>
<aside>
<img src="{% static 'main/img/utensils-solid.svg' %}" alt="Logo">
Рецепты
<h3>Навигация</h3>

 Главная
 О нас
 Рецепты
 Контакты
{% if user.is_authenticated %}
{{ user.username }} Выйти
{% else %}
 Войти
 Регистрация
{% endif %}
<button class="btn btn-info">Найти рецепт</button>

</aside>
<main>
{% if messages %}
{% for message in messages %}

{{ message }}

{% endfor %}
{% endif %}
{% block content %}
{% endblock %}
</main>
36 19
html > body > aside > img
project > website > main > templates > main > layout.html
18:74 CRLF UTF-8 4 spaces Python 3.10 (project)


project
Version control
Current File
Project
recipes.html
personal_account.html
main_2.css
layout.html
project C:\Users\zin_k\PycharmProjects\
.venv library root
website
main
migrations
static
templates
main
about.html
layout.html
welcome.html
__init__.py
admin.py
apps.py
models.py
tests.py
urls.py
views.py
recipes
migrations
templates
__init__.py
admin.py
apps.py
forms.py
models.py
tests.py
urls.py
views.py
<!doctype html>
{% load static %}
<html lang="ru">
<head>
<meta charset="UTF-8">
<meta name="viewport"
content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>{% block title %}{% endblock %}</title>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css">
<link rel="stylesheet" href="{% static 'main/css/main_2.css' %}">
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v6.5.1/css/all.css">
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
</head>
<body>
<aside>
<img src="{% static 'main/img/utensils-solid.svg' %}" alt="Logo">
Рецепты
<h3>Навигация</h3>

 Главная
 О нас
 Рецепты
 Контакты
{% if user.is_authenticated %}
{{ user.username }} Выйти
{% else %}
 Войти
 Регистрация
{% endif %}
<button class="btn btn-info">Найти рецепт</button>

</aside>
<main>
{% if messages %}
{% for message in messages %}

{{ message }}

{% endfor %}
{% endif %}
{% block content %}
{% endblock %}
</main>
html
body
aside
img
project
website
main
templates
main
layout.html
18:74
CRLF
UTF-8
4 spaces
Python 3.10 (project)

apps.py
forms.py
models.py
tests.py
urls.py
views.py
<script src="https://code.jquery.com/jquery-3.7.1.min.js" integrity="sha256-/JqT3SQfawRcv/BIHPThkBvs0OEvtFFmqPF/lYI/Cxo=" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
</body>
</html>
html
body
aside
img
project
website
main
templates
main
layout.html
18:74
CRLF
UTF-8
4 spaces
Python 3.10 (project)