

MUHAMMAD AL-XORAZMIY
NOMIDAGI TATU FARG'ONA FILIALI
FERGANA BRANCH OF TUIT
NAMED AFTER MUHAMMAD AL-KHORAZMI

“AL-FARG‘ONIIY AVLODLARI”

ELEKTRON ILMIY JURNALI | ELECTRONIC SCIENTIFIC JOURNAL

TA'LIMDAGI ILMIY, OMMABOP VA ILMIY TADQIQOT ISHLARI



4-SON 1(4)
2023-YIL

TATU, FARG'ONA
O'ZBEKISTON



O'ZBEKISTON RESPUBLIKASI RAQAMLI TEXNOLOGIYALAR VAZIRLIGI

MUHAMMAD AL-XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
FARG'ONA FILIALI



Muassis: Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti Farg'ona filiali.

Chop etish tili: O'zbek, ingliz, rus. Jurnal texnika fanlariga ixtisoslashgan bo'lib, barcha shu sohadagi matematika, fizika, axborot texnologiyalari yo'nalishida maqolalar chop etib boradi.

Учредитель: Ферганский филиал Ташкентского университета информационных технологий имени Мухаммада ал-Хоразми.

Язык издания: узбекский, английский, русский. Журнал специализируется на технических науках и публикует статьи в области математики, физики и информационных технологий.

Founder: Fergana branch of the Tashkent University of Information Technologies named after Muhammad al-Khorazmi.

Language of publication: Uzbek, English, Russian. The magazine specializes in technical sciences and publishes articles in the field of mathematics, physics, and information technology.

2023 yil, Tom 1, №4
Vol.1, Iss.4, 2023 y

ELEKTRON ILMIY JURNALI

ELECTRONIC SCIENTIFIC JOURNAL

«Al-Farg'oniyl avlodlari» («The descendants of al-Fargani», «Potomki al-Fargani») O'zbekiston Respublikasi Prezidenti administratsiyasi huzuridagi Axborot va ommaviy kommunikatsiyalar agentligida 2022-yil 21 dekabrda 054493-son bilan ro'yxatdan o'tgan.

Jurnal OAK Rayosatining 2023-yil 30 sentabrdagi 343-sonli qarori bilan Texnika fanlari yo'nalishida milliy nashrlar ro'yxatiga kiritilgan.

Tahririyat manzili:
151100, Farg'ona sh.,
Aeroport ko'chasi 17-uy,
202A-xona
Tel: (+99899) 998-01-42
e-mail: info@al-fargoniy.uz

Qo'lyozmalar taqrizlanmaydi va qaytarilmaydi.

FARG'ONA - 2023 YIL

TAHRIR HAY'ATI

Maxkamov Baxtiyor Shuxratovich,

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti rektori, iqtisodiyot fanlari doktori, professor

Muxtarov Farrux Muhammadovich,

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti Farg'ona filiali direktori, texnika fanlari doktori

Arjannikov Andrey Vasilevich,

Rossiya Federatsiyasi Sibir davlat universiteti professori, fizika-matematika fanlari doktori

Satibayev Abdugani Djunosovich,

Qirg'iziston Respublikasi, Osh texnologiyalari universiteti, fizika-matematika fanlari doktori, professor

Rasulov Akbarali Maxamatovich,

Muhammad al-Xorazmiy nomidagi TATU Farg'ona filiali Axborot texnologiyalari kafedrasida professori, fizika-matematika fanlari doktori

Yakubov Maksadxon Sultaniyazovich,

Muhammad al-Xorazmiy nomidagi TATU «Axborot texnologiyalari» kafedrasida professori, t.f.d., professor, xalqaro axborotlashtirish fanlari Akademiyasi akademigi

G'ulomov Sherzod Rajaboyevich,

Muhammad al-Xorazmiy nomidagi TATU Kiberxavfsizlik fakulteti dekani, Ph.D., dotsent

G'aniyev Abduxalil Abdjalioviich,

Muhammad al-Xorazmiy nomidagi TATU Kiberxavfsizlik fakulteti, Axborot xavfsizligi kafedrasida t.f.n., dotsent

Zaynidinov Hakimjon Nasritdinovich,

Muhammad al-Xorazmiy nomidagi TATU Kompyuter injiniringi fakulteti, Sun'iy intellekt kafedrasida texnika fanlari doktori, professor

Bo'taboyev Muhammadjon To'ychiyevich,

Farg'ona politexnika instituti, Iqtisod fanlari doktori, professor

Abdullayev Abdujabbor,

Andijon mashinosozlik instituti, Iqtisod fanlari doktori, professor

Qo'ldashev Abbosjon Hakimovich,

O'zbekiston milliy universiteti huzuridagi Yarimo'tkazgichlar fizikasi va mikroelektronika ilmiy-tadqiqot instituti, texnika fanlari doktori, professor

Ergashev Sirojiddin Fayazovich,

Farg'ona politexnika instituti, elektronika va asbobsozlik kafedrasida professori, texnika fanlari doktori, professor

Qoraboyev Muhammadjon Qoraboevich,

Toshkent tibbiyot akademiyasi Farg'ona filiali fizika matematika fanlari doktori, professor, BMT ning maslahatchisi maqomidagi xalqaro axborotlashtirish akademiyasi akademigi

Polvonov Baxtiyor Zaylobiddinovich,

Muhammad al-Xorazmiy nomidagi TATU Farg'ona filiali Ilmiy ishlar va innovatsiyalar bo'yicha direktor o'rinbosari

Zulunov Ravshanbek Mamatovich,

Muhammad al-Xorazmiy nomidagi TATU Farg'ona filiali Dasturiy injiniring kafedrasida dotsenti, fizika-matematika fanlari nomzodi

Saliyev Nabijon,

O'zbekiston jismoniy tarbiya va sport universiteti Farg'ona filiali dotsenti

Abdullaev Temurbek Marufovich,

Muhammad al-Xorazmiy nomidagi TATU Axborot texnologiyalari kafedra mudiri, texnika fanlar bo'yicha falsafa doktori

Zokirov Sanjar Ikromjon o'g'li,

Muhammad al-Xorazmiy nomidagi TATU Farg'ona filiali Ilmiy tadqiqotlar, innovatsiyalar va ilmiy-pedagogik kadrlar tayyorlash bo'limi boshlig'i, fizika-matematika fanlari bo'yicha falsafa doktori

Jurnal quyidagi bazalarda indekslanadi:



Eslatma! Jurnal materiallari to'plamiga kiritilgan ilmiy maqolalardagi raqamlar, ma'lumotlar haqqoniyligiga va keltirilgan iqtiboslar to'g'riligiga mualliflar shaxsan javobgardirlar.

MUNDARIJA | ОГЛАВЛЕНИЕ | TABLE OF CONTENTS

Muxtarov Farrux Muhammadovich, TARMOQ TRAFIGI ANOMALIYALARINI IDENTIFIKATSIYA QILISHNING STATIK USULI	4-7
Daliyev Baxtiyor Sirojiddinovich, Abelning umumlashgan integral tenglamasini yechish uchun Sobolev fazosida optimal kvadratur formulalar	8-14
Umarov Shuxratjon Azizjonovich, KRIPTOBARDOSHLI KRIPTOGRAFIK TIZIMLAR VA ULARNING KLASSIFIKATSIYASI	15-21
Zulunov Ravshanbek Mamatovich, PYTHONDA NEYRON TARMOQNI QURISH VA BASHORAT QILISH	22-26
Djalilov Mamatisa Latibdjanovich, IKKI QATLAMLI NOELASTIK PLASTINKANING KO'NDALANG TEBRANISHI UMUMIY TENGLAMASINI TAHLIL QILISH	27-30
Erkin Uljaev, Azizjon Abdulkhamidov, Utkirjon Ubaydullayev, A Convolutional Neural Network For Classification Cotton Boll Opening Degree	31-36
Seytov Aybek Jumabayevich, Xusanov Azimjon Mamadaliyevich, Magistral kanallarda suv resurslarini boshqarish jarayonlarini modellashtirish algoritmini ishlab chiqish	37-43
Abdullayev Temurbek Marufjonovich, Algorithm of functioning of intellectual information-measuring system	44-49
Odinakhon Sadikovna Rayimjanova, Usmonali Umarovich Iskandarov, Reaserch of highly sensitive deformation semiconductor sensors based on AFV	50-53
S.S.Radjabov, G.R.Mirzayeva, A.O.Tillavoldiyev, J.A.Allayorov, BARG TASVIRI BO'YICHA MADANIY O'SIMLIKLARNING FITOSANITAR HOLATINI ANIQLASH ALGORITMLARI	54-59
Эргашев Отабек Мирзапулатович, Интеллектуальный оптоэлектронный прибор для учета и контроля расходом воды в открытых каналах	60-65
Xomidov Xushnudbek Rapiqjon o'g'li, Nurmatov Sardorbek Xasanboy o'g'li, Yo'ldashev Bilol Iqboljon o'g'li, O'lmasov Farrux Yorqinjon o'g'li, Konus setkali chang tozalovchi qurilma uchun chang namunalarning dispers tarkibi tahlili	66-69
Akhundjanov Umidjon Yunus ugli, VERIFICATION OF STATIC SIGNATURE USING CONVOLUTIONAL NEURAL NETWORK	70-74
Лазарева Марина Викторовна, Горовик Александр Альфредович, Цифровизация и цифровой менеджмент в современном управлении	75-81
D.X.Tojimatov, KIBERTAHDIDLARNI OLDINI OLIHDA KIBERRAZVEDKA AMALIYOTI VA UNING USTUVOR VAZIFALARI	82-85
Muxtarov Farrux Muhammadovich, Rasulov Akbarali Maxamatovich, Ibroximov Nodirbek Ikromjonovich, Kompyuter eksperimenti orqali kam atomli mis klasterlarining geometrik tuzilishini o'rganish	86-89
Umurzakova Dilnoza Maxamadjanovna, BOSHQARISH QONUNLARINI ADAPTATSIYALASH ALGORITMLARINI ISHLAB CHIQLASH	90-94
Muxamedieva Dildora Kabilovna, Muxtarov Farrux Muhammadovich, Sotvoldiev Dilshodbek Marifjonovich, JAMOAT TRANSPORTI MARSHRUTLARINI QURISH INTELLEKTUAL ALGORITMLARI	95-103
Нурдинова Разияхон Абдихаликовна, Перспективы применения элементов с аномальными фотовольтаическими напряжениями	104-108
Bozarov Baxromjon Pkhomovich, UCH O'LCHOVLI FAZODAGI SFERADAANIQLANGAN FUNKSIYALARNI TAQRIBIY INTEGRALLASH UCHUN OPTIMAL KUBATUR FORMULALAR	109-113
Улжаев Эркин, Худойбердиев Элёр Фахриддин угли, Нарзуллаев Шохрух Нурали угли, РАЗРАБОТКА КОНСТРУКЦИИ И ФУНКЦИОНАЛЬНОЙ СХЕМЫ ПОЛУЦИЛИНДРИЧЕСКОГО ЁМКОСТНОГО ПОТОЧНОГО ВЛАГОМЕРА	114-122
Mamirov Uktam Farkhodovich, Buronov Bunyod Mamurjon ugli, ALGORITHMS FOR FORMATION OF CONTROL EFFECTS IN CONDITIONS OF UNOBSERVABLE DISTURBANCES	123-127
Sharibayev Nosirjon Yusubjanovich, Jabborov Anvar Mansurjonovich, YURAK-QON TOMIR KASALLIKLARI DIAGNOSTIKASI UCHUN TEXNOLOGIYALAR, ALGORITMLAR VA VOSITALAR	128-136
Marina Lazareva, Estimating development time and complexity of programs	137-141
Asrayev Muhammadmullo, ONLINE HANDWRITING RECOGNITION	142-146
Norinov Muhammadyunus Usibjonovich, SPEKTR ZONALI TASVIRLARGA INTELLEKTUAL ISHLOV BERISH USULLARI TAHLILI	147-152
Xudoynazarov Umidjon Umarjon o'g'li, PARAMETRLI ALGEBRAGA ASOSLANGAN EL-GAMAL SHIFRLASH ALGORITMLARINI GOMOMORFIK XUSUSIYATINI TADQIQ ETISH	153-157
D.M.Okhunov, M.Okhunov, THE ERA OF THE DIGITAL ECONOMY IS AN ERA OF NEW OPPORTUNITIES AND PROSPECTS FOR BUSINESS DEVELOPMENT BASED ON CROWDSOURCING TECHNOLOGIES	158-165

MUNDARIJA | ОГЛАВЛЕНИЕ | TABLE OF CONTENTS

Солиев Бахромжон Набиджонович, Путеводитель по построению веб-API на Django - Шаг за шагом с Django REST framework — от моделей до проверки работоспособности	166-171
Sevinov Jasur Usmonovich, Boborayimov Okhunjon Khushmurod ogli, ALGORITHMS FOR SYNTHESIS OF ADAPTIVE CONTROL SYSTEMS WITH IMPLICIT REFERENCE MODELS BASED ON THE SPEED GRADIENT METHOD	172-176
Mamatov Narzullo Solidjonovich, Jalelova Malika Moyatdin qizi, Tojiboyeva Shaxzoda Xoldorjon qizi, Samijonov Boymirzo Narzullo o'g'li, SUN'IY YO'LDOSHDAN OLINGAN TASVIRDAGI DALA MAYDONI CHEGARALARINI ANIQLASH USULLARI	177-181
Обухов Вадим Анатольевич, Криптография на основе эллиптических кривых (ECC)	182-188
Turdimatov Mamirjon Mirzayevich, Sadirova Xursanoy Xusanboy qizi, AXBOROTNI HIMOYALASHDA CHETLAB O'TISHNING MUMKIN BO'LGAN EHTIMOLLIK XOLATINI BAHOLASH USULLARI	189-193
Musayev Xurshid Sharifjonovich, TRIKOTAJ MAHSULOTLARIDA NUQSONLI TO'QIMALARNING ANIQLASHNING MATEMATIK MODELI VA UNING ALGORITMLARI	194-196
Kodirov Ahkhmadkhon, Umarov Abdumukhtar, Rozaliyev Abdumalikjon, ANALYSIS OF FACIAL RECOGNITION ALGORITHMS IN THE PYTHON PROGRAMMING LANGUAGE	197-205
Suyumov Jorabek Yunusalievich, METHODOLOGICAL PROBLEMS OF QUALIMETRY IN CONDUCT OF PEDAGOGICAL EXPERIMENT-EXAMINATION	206-211
Хаджаев Саидакбар Исмоил угли, АКТУАЛЬНОСТЬ ПРОБЛЕМЫ ЗАЩИТЫ ИНФОРМАЦИОННЫХ СИСТЕМ МАЛОГО И СРЕДНЕГО БИЗНЕСА ОТ КИБЕРАТАК	212-217
M.M.Khalilov, Effect of Heat Treatment on the Photosensitivity of Polycrystalline PbTe Films AND PbS	218-221
Тажибаев Илхом Бахтиёрвич, ПОЛНОСТЬЮ ВОЛОКОННЫЙ СЕНСОР, ОСНОВАННЫЙ НА КОНСТРУКЦИИ ИЗ МАЛОМОДОВОГО ВОЛОКОННОГО СМЕЩЕНИЯ С КАСКАДНЫМ СОЕДИНЕНИЕМ ВОЛОКОННОЙ РЕШЕТКИ С БОЛЬШИМ ИНТЕРВАЛОМ, ИСПОЛЬЗУЕТСЯ ДЛЯ ОПРЕДЕЛЕНИЯ ИСКРИВЛЕНИЯ И ПРОВЕДЕНИЯ АКУСТИЧЕСКИХ ИЗМЕРЕНИЙ	222-225
Sharibaev Nosir Yusubjanovich, Djuraev Sherzod Sobirjanovich, To'xtasinov Davronbek Xoshimjon o'g'li, PRIORITIES IN DETERMINING ELECTRIC MOTOR VIBRATION WITH ADXL345 ACCELEROMETER SENSOR	226-230
Mukhammadjonov A.G., ANALYSIS OF AUTOMATION THROUGH SENSORS OF HEAT AND HUMIDITY OF DIFFERENT DIRECTIONS	231-236
Эрматова Зарина Кахрамоновна, АКТУАЛЬНОСТЬ ПРЕПОДАВАНИЯ ЯЗЫКА ПРОГРАММИРОВАНИЯ C++ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ	237-241
Saparbaev Rakhmon, ANALOG TO DIGITAL CONVERSION PROCESS BY MATLAB SIMULINK	242-245
Садикова М.А., Авазова Н.К., САМООБУЧЕНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА, БАЗОВЫЕ ПРИНЦИПЫ РАБОТЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НА ПРОСТОМ ПРИМЕРЕ	246-250
Abduhafizov Tohirjon Ubaydullo o'g'li, Abdurasulova Dilnoza Botirali kizi, DEVELOPMENT OF ALGORITHMS IN THE ANALYSIS OF DEMAND AND SUPPLY PROCESSES IN ECONOMIC SYSTEMS	251-256
Kayumov Ahror Muminjonovich, CREATING MATHEMATICAL MODELS TO IDENTIFY DEFECTS IN TEXTILE MACHINERY FABRIC	257-261
Mirzakarimov Baxtiyor Abdusalomovich, Xayitov Azizjon Mo'minjon o'g'li, BIOMETRIC METHODS SECURE COMPUTER DATA FROM UNAUTHORIZED ACCESS	262-266
Soliyev B., Odilov A., Abdurasulova Sh., Leveraging Python for Enhanced Excel Functionality: A Practical Exploration	267-271
Жураев Нурмахамад Маматович, Системы Электроснабжения Оборудования Предприятий Связи: Надежность и Эффективность	272-276
Rasulova Feruzaxon Xoshimjon qizi, Isroilov Sharobiddin Mahammadyusufovich, OLIY TA'LIM MUASSASALARIDA MUTAXASSISILIK FANLARINI O'QITISHDA MULTIMEDIALI MOBIL ILOVADANDAN FOYDALANISHNING STATISTIK TAHLILI	277-280
Muxtarov Farrux Muxammadovich, Toshpulatov Sherali Muxamadaliyevich, SUN'IY INTELLEKT YORDAMIDA IJTIMOYIY TARMOQ MONITORINGI TIZIMINI YARATISH, AFZALLIKLARI VA MUHIM JIXATLARI	281-285
Sadikova Munira Alisherovna, APPLICATION OF ARTIFICIAL INTELLIGENCE DEVICES IN MANUFACTURING	286-290
Mamatov Narzullo Solidjonovich, Ibroximov Sanjar Rustam o'g'li, Fayziyev Voxid Orzumurod o'g'li, Samijonov Abdurashid Narzullo o'g'li, SUN'IY INTELLEKT VOSITALARINI TA'LIMNI NAZORAT QILISH VA BAHOLASHDA QO'LLASH	291-297

Путеводитель по построению веб-API на Django - Шаг за шагом с Django REST framework — от моделей до проверки работоспособности

Солиев Бахромжон Набиджонович,

Старший преподаватель Ферганского филиала
Ташкентского университета информационных
технологий имени Мухаммада аль-Хорезми
bahromjonsoliev@gmail.com

Аннотация. В данной публикации мы изучим применение Django REST framework для разработки веб-API в контексте фреймворка Django. Рассмотрим основные шаги по настройке проекта, созданию эндпоинтов, аутентификации пользователей, а также использованию инструментов для обработки и валидации данных.

Ключевые слова. Django REST framework, Web API, Django фреймворк, Веб-сервисы, Сериализация данных, Аутентификация, авторизация, URL-маршрутизация, Валидация данных, Документация API

Введение. Разрабатывая приложение или любой другой программный продукт, мы стремимся к оперативному и высококачественному процессу. Эти цели легче достигнуть, воспользовавшись опытом и достижениями других программистов, поскольку многие задачи, с которыми мы сталкиваемся, уже находятся в сфере чужого опыта. Сущность заключается в том, что применение фреймворка позволяет сосредоточиться на бизнес-задачах, минуя необходимость беспокоиться о технических аспектах там, где это возможно. Нужно создать кнопку на веб-сайте? Авторизовать пользователя или восстановить пароль? Сохранить данные пользователя из веб-формы в базу данных? Все уже готово — просто используйте!

Кроме того, важно понимать различия между фреймворком и библиотекой. Эти понятия схожи, но в обоих случаях в Python вам нужно установить соответствующий пакет, импортировать его и начать использовать. Основное различие заключается в том, что библиотеку вы интегрируете в проект там, где это необходимо и по своему усмотрению, в то время как фреймворк сам определяет подход к разработке приложения. То есть, фреймворк не только предоставляет удобные инструменты разработки в виде вспомогательных функций и классов, но и устанавливает архитектуру проекта, создает

структуру кода и, короче говоря, определяет путь, по которому будет развиваться ваше приложение.

Литературный обзор и методология. На настоящий момент наиболее эффективным инструментом для разработки веб-приложений на языке программирования Python является фреймворк Django. Django можно охарактеризовать как MVC-фреймворк, так как он обеспечивает взаимодействие между пользователем и системой через следующие компоненты:

- Модель (хранение данных пользователя)
- Представление (отображение данных пользователя)
- Контроллер (обработка изменений данных от пользователя).

В контексте Django, данная терминология может звучать немного иначе, но суть остается неизменной. Разработка Django началась в 2003 году под руководством программистов Адриана Холовати и Саймона Уиллисона, а первый публичный релиз состоялся в 2005 году. Функционал фреймворка тщательно соответствовал требованиям веб-разработки того времени и продолжает активно развиваться.

Несмотря на то, что Django считается обширным и многофункциональным фреймворком, сам по себе он не всегда способен



удовлетворить все потребности веб-разработчиков. За годы существования фреймворка было создано множество сторонних библиотек и дополнительных фреймворков, которые расширяют его возможности. Эти дополнения упрощают выполнение различных задач, таких как аутентификация через социальные сети, кеширование данных, облачное хранение файлов и многое другое. Некоторые из этих расширений впоследствии интегрированы в проект Django, например, библиотека South, отвечающая за управление миграциями базы данных. Тем не менее, большинство дополнений остаются автономными пакетами, включая Django REST Framework, который предоставляет средства для создания веб-API на основе Django.

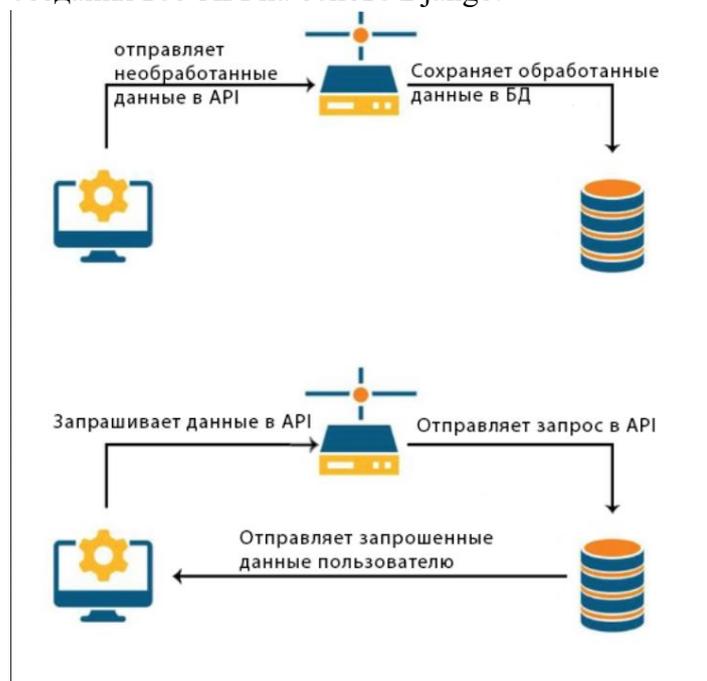


Рис 1. Связь между пользователем и сервером

На сегодняшний момент архитектура Интернета оперирует по принципу взаимодействия Клиента и Сервера. Клиент инициирует запрос, а Сервер предоставляет ответ (рис 1). В случае общения двух Серверов, принято условно называть тот, который отправляет запрос и ожидает ответа, Клиентом, а тот, который принимает запрос и предоставляет ответ, Сервером. Взаимодействие между браузерами и веб-сайтами, где браузеры действуют в роли Клиента, а веб-сайты в роли

Сервера, традиционно осуществлялось с использованием html-рендеринга – именно таким образом работал Django в своем начале. Для получения данных с веб-сайта браузер направляет GET-запрос к Серверу, тот формирует ответ в виде html-страницы и передает ее браузеру. Однако, как браузер может передавать данные Серверу?

В html-странице Сервер внедряет все необходимые веб-формы, заполнив которые, пользователь может передать свои данные обратно на сервер. Когда вы вводите свои данные в форму на сайте, браузер отправляет Серверу POST-запрос, содержащий ваши данные. Сервер обрабатывает их и записывает в базу данных.

Все функционировало успешно, но уже в середине 2000-х этот подход перестал соответствовать растущим требованиям веб-разработки. Появление мобильных приложений и различных гаджетов с доступом в интернет сделало стандартный метод html-рендеринга на сервере непригодным, так как каждому клиенту нужно было отображать данные по-разному. Взаимодействие между серверами стало более интенсивным, и формат html перестал быть оптимальным. Для решения этих задач был предложен другой метод обмена данными — Web API. Суть этого метода заключается в том, что Сервер передает Клиенту не html-страницу, а сами данные, не заботясь о том, как они будут представлены. Популярными форматами передачи данных стали XML и JSON. Таким образом, Сервер освобождается от обязанности отображения данных. Некоторое время разработчикам веб-приложений на сервере приходилось поддерживать оба метода одновременно: html-рендеринг для браузеров и использование Web API для мобильных приложений и взаимодействия с другими серверами. Очевидно, это требовало двойного труда от разработчиков, но в начале 2010-х ситуация стала меняться в пользу Web API. Помогли в этом стремительное развитие инструментов на JavaScript и появление различных веб-фреймворков, включая тот, о котором идет речь в данной статье.

Браузерные приложения научились самостоятельно отрисовывать веб-страницы,



получая чистые данные с сервера. Веб-приложения на сервере научились быстро и легко создавать API. Это привело к явному разделению на Backend и Frontend разработку: разработчики, поддерживающие приложение на сервере, и те, кто создает клиентские приложения для браузеров. Web API стал универсальным методом взаимодействия для Сервера и всех его клиентов, включая браузеры, мобильные приложения и другие сервера. Разумеется, это содействовало развитию стандартов в области взаимодействия между системами. Для обеспечения эффективного обмена данными Клиента и Сервера необходимо было определить общие правила – и в начале 2010-х концепция REST стала таким стандартом.

Django REST framework (DRF) - это мощный фреймворк для создания веб-API на основе Django. Он предоставляет набор инструментов для быстрого и удобного создания RESTful API веб-приложений с использованием языка программирования Python.

Вот некоторые ключевые особенности Django REST framework:

Сериализация данных: DRF предоставляет мощные инструменты для преобразования объектов Django в JSON-подобные данные и обратно. Это называется сериализацией.

Аутентификация и авторизация: DRF включает в себя готовые средства для аутентификации пользователей и управления их доступом к различным частям вашего API.

Просмотры (Views): DRF предоставляет множество предварительно настроенных просмотров для обработки различных типов запросов. Они организованы вокруг классов, что делает их мощными и гибкими.

URL-маршрутизация: DRF предоставляет инструменты для определения URL-маршрутов для ваших эндпоинтов API.

Поддержка разных форматов: DRF поддерживает различные форматы данных, такие как JSON, XML, YAML и другие.

Поддержка пагинации и фильтрации: Вы можете легко настроить пагинацию для ограничения количества возвращаемых результатов, а также фильтрацию для запросов.

Валидация данных: DRF предоставляет инструменты для валидации входных данных перед их обработкой.

Расширяемость: Вы можете легко расширять функциональность DRF, создавая свои собственные классы сериализаторов, просмотров и других компонентов.

Поддержка аутентификации через сторонние сервисы (OAuth, JWT и др.): DRF интегрируется с различными методами аутентификации для обеспечения безопасности вашего API.

Документация: DRF поставляется с автоматической интерактивной документацией API, которая облегчает понимание и использование вашего API.

Результаты. Давайте рассмотрим Django Rest Framework на конкретных примерах ниже.

Шаг 1: Установка Django REST framework

Первым этапом следует произвести установку Django REST framework. Процедуру установки можно выполнить при помощи инструмента pip:

```
pip install djangorestframework
```

Шаг 2: Инициализация Django-проекта

Для начала создания проекта Django, выполните следующую команду:

```
django-admin startproject myproject
```

Шаг 3: Создание приложения Django

Создайте приложение Django с помощью команды:

```
python manage.py startapp my_app
```

Шаг 4: Настройка Django REST framework

```
INSTALLED_APPS = [  
...  
'rest_framework',  
'my_app',  
]
```



Надо добавить REST framework middleware в MIDDLEWARE в файле settings.py:

```
MIDDLEWARE = [  
    ...  
  
    'rest_framework.middleware.AuthenticationMiddleware',  
  
    'rest_framework.middleware.AuthorizationMiddleware',  
]
```

Шаг 5: Определение модели в Django
Создайте модель Django, описав ее структуру в файле models.py вашего приложения:

```
from django.db import models  
  
class Product(models.Model):  
    pr_name = models.CharField(max_length=100)  
    pr_description = models.TextField()  
    pr_price = models.DecimalField(max_digits=10, decimal_places=2)  
    pr_created_at = models.DateTimeField(auto_now_add=True)  
    pr_updated_at = models.DateTimeField(auto_now=True)  
  
    def __str__(self):  
        return self.name
```

Шаг 6: Определение сериализатора в Django REST framework

В Django REST framework определение сериализатора происходит путем создания класса, который наследуется от одного из классов-сериализаторов, предоставляемых фреймворком. Основные классы-сериализаторы включают serializers.Serializer и serializers.ModelSerializer.

1. serializers.Serializer:

Этот класс является базовым для создания сериализаторов. Вам придется определить поля

(fields), которые вы хотите включить в сериализацию данных. Это может включать в себя различные типы полей, такие как CharField, IntegerField и другие.

Пример:

```
from rest_framework import serializers  
  
class MySerializer(serializers.Serializer):  
    field1 = serializers.CharField()  
    field2 = serializers.IntegerField()
```

2. serializers.ModelSerializer:

Этот класс упрощает создание сериализатора для моделей Django. Он автоматически создает поля сериализатора, основываясь на полях модели, что делает процесс создания сериализатора для модели более компактным.

Пример:

```
from rest_framework import serializers  
from .models import MyModel  
  
class  
MyModelSerializer(serializers.ModelSerializer):  
    class Meta:  
        model = MyModel  
        fields = '__all__'
```

После определения сериализатора, он может быть использован в представлениях (views) для преобразования данных модели в формат, удобный для передачи через веб-API, и наоборот. Сериализаторы также обеспечивают валидацию данных при создании или обновлении ресурсов через API.

Теперь создадим сериализатор в Django REST framework, описав его в файле serializers.py в приложения:

```
from rest_framework import serializers  
from my_app.models import Product
```



class

ProductSerializer(serializers.ModelSerializer):

class Meta:

model = Product

```
fields = ['id', 'pr_name', 'pr_description', 'pr_price', 'pr_created_at', 'pr_updated_at']
```

Шаг 7: Определение представления в Django REST framework

Создайте представление в Django REST framework, описав его в файле `views.py` вашего приложения:

```
from rest_framework import generics  
from my_app.models import Product  
from my_app.serializers import
```

ProductSerializer

class

ProductList(generics.ListCreateAPIView):

queryset = Product.objects.all()

serializer_class = ProductSerializer

class

ProductDetail(generics.RetrieveUpdateDestroyAPIView):

queryset = Product.objects.all()

serializer_class = Product

Шаг 8: Настройка маршрутов Django REST framework

Определите маршруты Django REST framework в файле `urls.py` вашего приложения:

```
from django.urls import path  
from my_app.views import ProductList,
```

ProductDetail

urlpatterns = [

path('products/', ProductList.as_view(),

name='product-list'),

path('products/<int:pk>',

ProductDetail.as_view(), name='product-detail'),

]

Шаг 9: Запуск приложения Django

Запустите приложение Django с помощью команды:

```
python manage.py runserver
```

Шаг 10: Проверка работоспособности Web API

Запустите веб-браузер и перейдите по адресу `http://127.0.0.1:8000/products/`. Здесь вы должны увидеть список всех доступных продуктов.

Для добавления нового продукта выполните POST-запрос на `http://127.0.0.1:8000/products/` с передачей данных в формате JSON:

```
{  
  "pr_name": "Product 1",  
  "pr_description": "Description for Product 1",  
  "pr_price": 10.99  
}
```

Чтобы получить детали конкретного продукта, отправьте GET-запрос на `http://127.0.0.1:8000/products/1/`, где 1 - идентификатор продукта.

Чтобы обновить продукт, отправьте PUT-запрос на `http://127.0.0.1:8000/products/1/` с данными в формате JSON:

```
{  
  "pr_name": "Updated Product 1",  
  "pr_description": "Updated Description for Product 1",  
  "pr_price": 12.99  
}
```

Чтобы удалить продукт, отправьте DELETE-запрос на `http://127.0.0.1:8000/products/1/`.

Заключение. В данной статье мы изучили процесс использования Django REST framework для формирования веб-API в контексте Django. Фреймворк предоставляет удобные инструменты для разработки RESTful API, включая возможности по аутентификации, авторизации, сериализации,



валидации и многие другие функции. Благодаря Django REST framework вы можете эффективно создавать мощные и гибкие веб-API на основе Django.

Использованная литература

1. Soliev B. N., kizi Abdurasulova D. B., Yakubov M. S. USING GINJA TEMPLATES TO CREATE E-COMMERCE PLATFORMS //Publishing House “Baltija Publishing”. – 2023.
2. Elevating E-Commerce in Uzbekistan with Python. (2023). Journal of Technical Research and Development, 1(1), 43-45.
<https://jtrd.mcdir.me/index.php/jtrd/article/view/2>
3. Navigating the E-Commerce Landscape in Uzbekistan with Python. (2023). Journal of Technical Research and Development, 1(1), 46-50.
<https://jtrd.mcdir.me/index.php/jtrd/article/view/1>
4. Uzbekistan’s Digital Market: Python’s E-Commerce Impact. (2023). Journal of Technical Research and Development, 1(1), 58-61.
<https://jtrd.mcdir.me/index.php/jtrd/article/view/5>
5. Python’s Role in Revolutionizing E-Commerce in Uzbekistan. (2023). Journal of Technical Research and Development, 1(1), 51-54.
<https://jtrd.mcdir.me/index.php/jtrd/article/view/4>
6. Nabijonovich S. B. EMPOWERING VIDEO ANALYTICS WITH AI-DRIVEN TEXT RECOGNITION IN PYTHON FOR STREAMLINED INSIGHTS //Galaxy International Interdisciplinary Research Journal. – 2023. – Т. 11. – №. 11. – С. 25-30.
7. <https://letslearnabout.net/blog/what-is-django-rest-framework-and-why-you-should-learn-it/>
8. <https://mkdev.me/ru/posts/chto-takoe-django-rest-framework>

