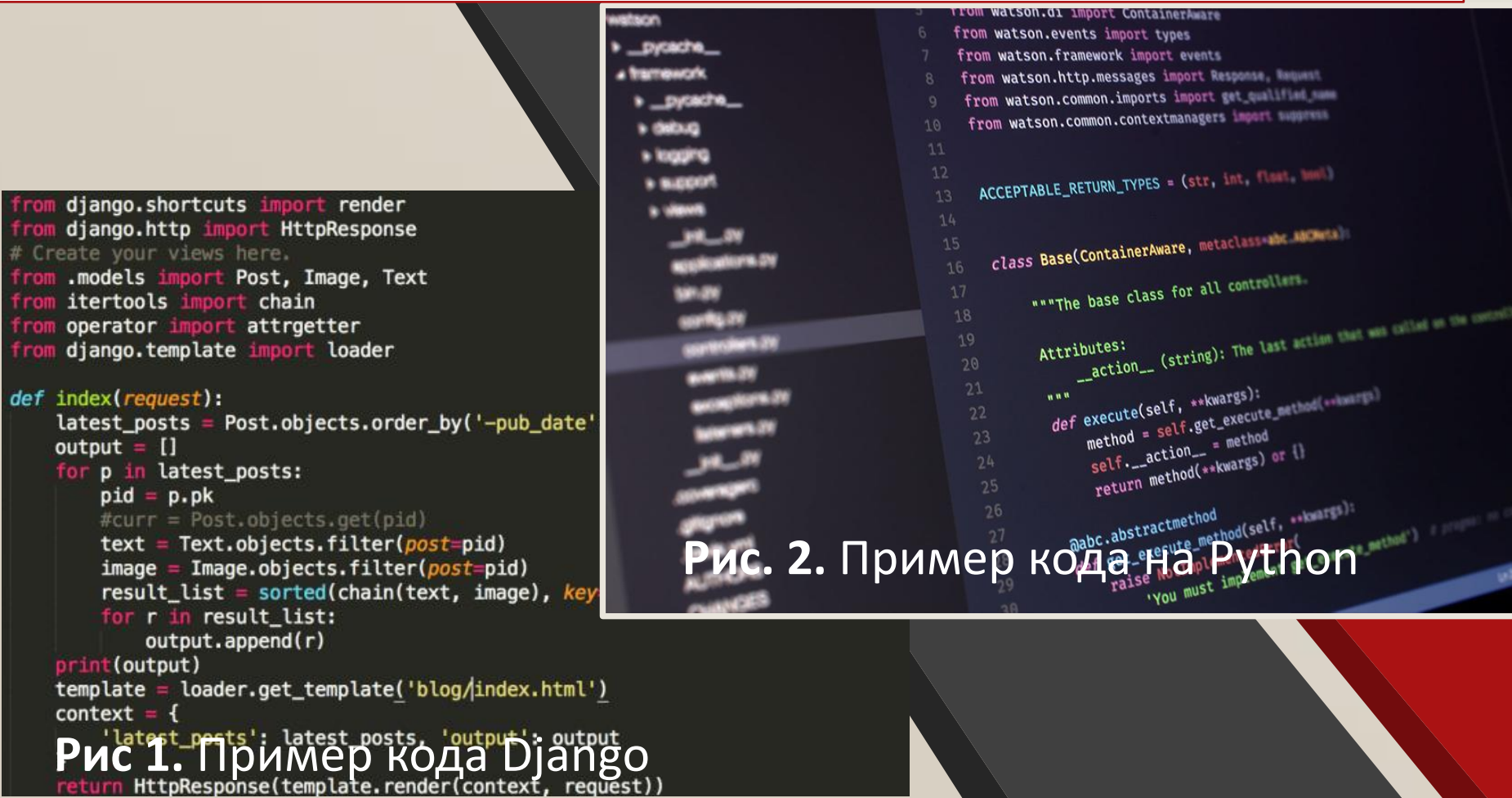


Аннотация

Python — это объектно-ориентированный, интерпретируемый и интерактивный язык программирования. Ещё одно следствие лидирующей позиции языка — рост популярности фреймворков на его основе. Фреймворки существуют для того, чтобы упрощать жизнь программисту. Они помогают экономить время, потому что благодаря им разработчик фокусируется на логике работы приложения, а не на рутинных задачах.

Введение

С развитием технологий и увеличением объема данных, необходимость в эффективной и быстрой веб-разработке становится все более актуальной. Существует множество Python-фреймворков, таких как Django, Flask и FastAPI, каждый из которых предлагает уникальные возможности и подходы к разработке веб-приложений. Несмотря на обширные исследования, недостаточно внимания уделено сравнительному анализу различных фреймворков в контексте специфических задач. Цель данного исследования заключается в сравнительном анализе популярных Python-фреймворков для веб-разработки с акцентом на их преимущества и недостатки в различных сценариях использования. Данное исследование имеет важное значение для разработчиков, выбирающих инструменты для создания веб-приложений



Методы и материалы

В исследовании проводился сравнительный анализ трех популярных Python-фреймворков (Django, Flask и FastAPI) с целью оценки их производительности, удобства использования и безопасности в различных сценариях веб-разработки. В качестве образцов были выбраны три фреймворка: Django, Flask и FastAPI. Исследование проводилось в условиях локальной среды разработки на стандартных машинах с установленными операционными системами Windows и Linux. Выбор фреймворков был основан на их популярности и распространенности в сообществе разработчиков. Каждому фреймворку было назначено одно приложение с одинаковыми функциональными требованиями

Результаты и обсуждение

FastAPI обеспечивает наилучшие показатели производительности и масштабируемости благодаря асинхронной обработке, тогда как Flask предлагает простоту и легкость в разработке для небольших проектов. Django, хотя и более комплексный, подходит для крупных приложений с множеством встроенных функций и хорошей безопасностью.

Таблица 1. Сравнение Фреймворков по показателям

Показатель	Django	Flask	FastAPI
Время отклика (мс)	200-400	100-300	50-100
Максимальная нагрузка(req/sec)	100-300	200-400	500-1000
Уровень безопасности(баллы)	85-90	70-80	90-95
Время разработки(часы)	40-80	20-40	20-40
Простота использования(баллы)	70-80	85-95	80-90

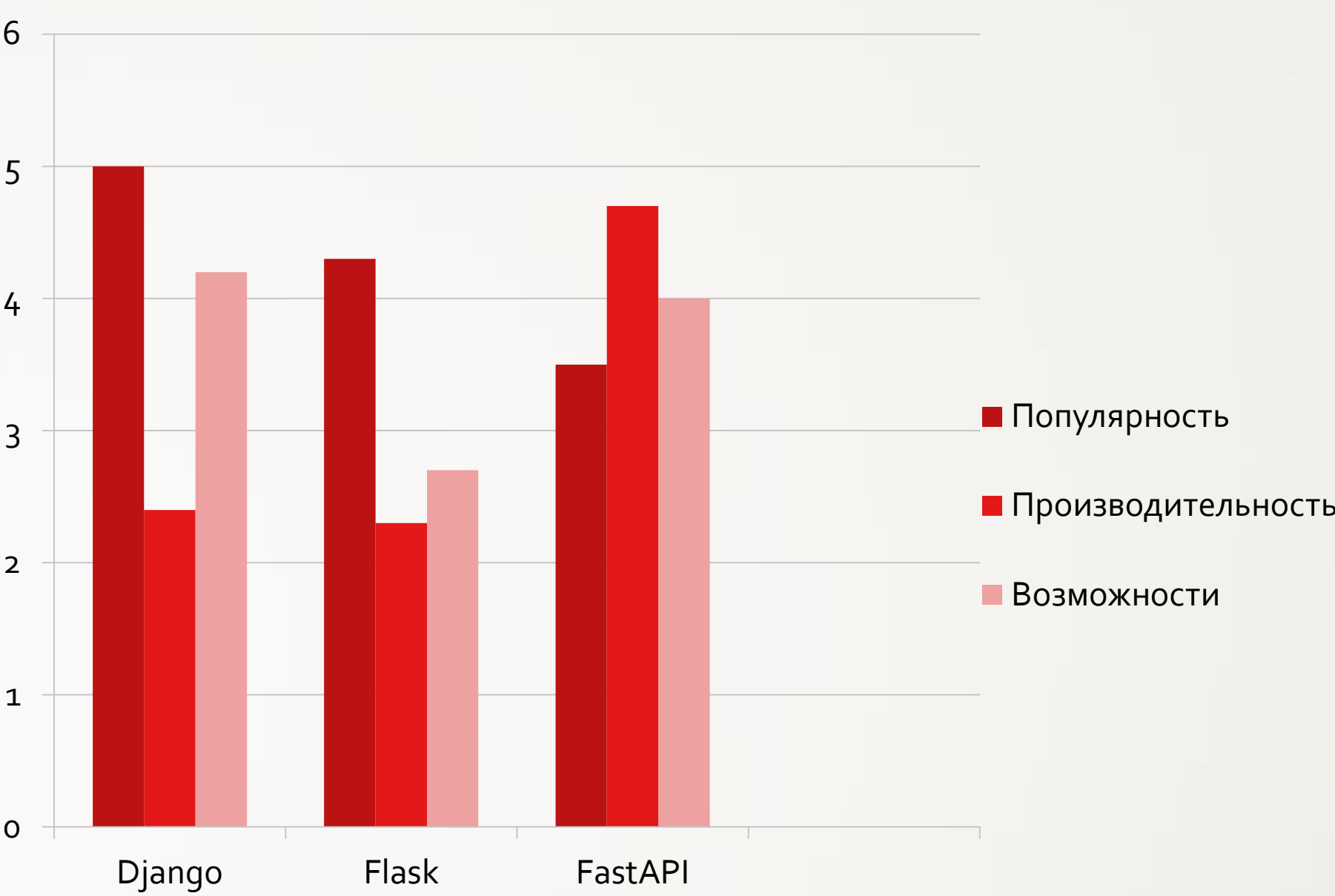


Диаграмма 1. Кагории Фреймворков

Заключение

Популярность может меняться со временем, но на данный момент Django и Flask остаются более известными, в то время как FastAPI быстро завоевывает свою нишу, особенно в контексте современных требований к производительности и асинхронности.

Библиографический список

- REST APIs with Django: Build powerful web APIs with Python and Django — William S. Vincent
- Flask by example, Gareth Dwyer 2016
- FastAPI: веб-разработка на Python | Любанович Билл

Владислав Гайтан
МГУ имени А.И. Куинджи

Email: vlad.gaytan1@mail.ru
Вебсайт: <https://mgumariupol.ru/>
Телефон: +79497220544