

Ресурсы:

- 1) Сравнительная таблица Django и Flask: <https://python-scripts.com/flask-vs-django>
- 2) Top 10 Python Web Frameworks to Learn in 2018: <https://hackernoon.com/top-10-python-web-frameworks-to-learn-in-2018-b2ebab969d1a>
- 3) Лучшие фреймворки для разработки на Python: <https://techrocks.ru/2018/11/20/best-python-frameworks/>

Паттерн Model-View-Controller (MVC) является крайне полезным при создании приложений со сложным графическим интерфейсом или поведением.

Как следует из названия, паттерн MVC включает в себя 3 компонента: Модель, Представление и Контроллер. Каждый из компонентов выполняет свою роль и является взаимозаменяемым. Это значит, что компоненты связаны друг с другом лишь некими четкими интерфейсами, за которыми может лежать любая реализация. Такой подход позволяет подменять и комбинировать различные компоненты, обеспечивая необходимую логику работы или внешний вид приложения.

Сравнительная таблица

Название	Ключевая разница	Как выбрать?	Другие сравнения
1. Django	Django предоставляет пакет « <i>все включено</i> »: у вас есть панель админа, интерфейсы баз данных, ORM, и структура каталогов для ваших приложений и проектов.	Django, если вас интересует конечный продукт. Особенно если вы работаете с прямолинейным приложением, таким как новостной сайт, магазин, блог, и если вам нужно, чтобы всегда один прозрачный способ выполнить ту или иную задачу.	Django это универсальный фреймворк. Он может использоваться при создании любых сайтов, поскольку умеет доставлять веб-контент в различных форматах (например, JSON, XML, RSS).
2. Flask	Flask предоставляет простоту, гибкость и аккуратность в работе, позволяя пользователю самому выбирать, как реализовать те или иные вещи.	Flask, если вы хотите получить больше опыта и возможности для обучения, или же если вам нужно больше контроля над используемыми компонентами (такие как базы данных, которые	Flask поставляется со специальным шаблоном, известным как Jinja. Но при желании вы можете выбрать любой другой движок шаблонов. Flask считается

		вам нужно использовать или взаимодействовать с ними).	открытым фреймворком. Вы можете использовать его для создания APIs, RESTful сервисов и endpoints.
--	--	---	---

Сведения о других фреймворках

1) **Pyramid** - это платформа веб-приложений на основе Python с открытым исходным кодом. Его главная цель - сделать как можно больше с минимальной сложностью.

Pyramid - это идеальный фреймворк для больших и сложных приложений, которые обычно рассчитаны на обслуживание многих людей. Однако его можно использовать и для простых приложений. Также это идеальный выбор для создания прототипов, которые будут использованы для разработки APIs.

Наиболее яркой особенностью Pyramid является способность хорошо работать как с маленькими, так и с большими приложениями. **Некоторые особенности:**

- Однофайловые приложения
- Генерация URL
- Расширяемая конфигурация
- Всеобъемлющие шаблоны и технические характеристики
- Гибкая аутентификация и авторизация
- Тестирование, поддержка и полная документация данных
- Функциональные декораторы

2) **CherryPy** - это минималистичный веб-фреймворк с открытым исходным кодом. Это делает создание веб-приложений Python ничем не отличается от создания любой другой объектно-ориентированной программы.

Фактически, веб-приложение на платформе CherryPy представляет собой автономное приложение Python, в которое встроен собственный многопоточный веб-сервер. Приложения CherryPy работают в любой операционной системе, которая поддерживает Python (Windows, macOS, Linux и т. Д.). Они могут быть развернуты везде, где вы можете запустить обычное приложение Python.

Этот фреймворк поставляется со специальным плагином, который помогает мониторить производительность приложения.

Некоторые из стандартных функций CherryPy включают в себя:

- HTTP / 1.1 совместимый с WSGI потоков объединенных веб - сервер
- Простота запуска нескольких HTTP-серверов одновременно
- Мощная система конфигурации
- Гибкая система плагинов
- Готовые инструменты
для кэширования , кодирования , сеансов , аутентификации , статического контента и т. Д.
- Встроенная поддержка тестирования
- Возможность работы на Python 2.7+, Python 3.1+, PyPy, Jython и Android

3) **Tornado** - это веб-фреймворк Python и библиотека асинхронных сетей. Он использует неблокирующий сетевой ввод-вывод и решает проблему C10k (это означает, что при правильной настройке он может обрабатывать более 10 000 одновременных подключений).

Это делает его отличным инструментом для создания приложений, требующих высокой производительности и десятков тысяч одновременно работающих пользователей.

Основные характеристики Tornado:

- Встроенная поддержка аутентификации пользователя
- Услуги в реальном времени
- Неблокирующий HTTP-клиент
- Реализация сторонних схем аутентификации и авторизации (Google OpenID / OAuth , Facebook Login, Yahoo BBAuth , FriendFeed OpenID / OAuth, Twitter OAuth)
- Поддержка перевода и локализации