

Web–программирование на языке Python. Фреймворки Django, Flask.

Васильев П. А.

*Васильев Петр Алексеевич / Vasilev Petr Alexeevich - студент,
кафедра информационных технологий,
институт математики и информатики
Северо-Восточный федеральный университет, г. Якутск*

Аннотация: в статье предлагается основы веб–программирования, инструменты разработки, их сравнение и различие.

Ключевые слова: Web–программирование, Python, Django, Flask.

В настоящее время с развитием сетевых технологий глобальная сеть – Internet стала использоваться в очень многих областях нашей жизни. Всё больше и больше людей пользуются интернетом – посещают разнообразные сайты и пользуются приложениями в смартфонах. В связи с этим появилось множество технологий, которые предоставляют возможность создания и разработки сайтов и Web–приложений.

Одним из самых популярных технологий является Python [1]. Python – это скриптовый язык программирования, который используется в разных областях программирования, в таких как искусственный интеллект, игры, микроконтроллеры, и, конечно же, Web–программирование. Но сам по себе Python не используется. Используются его технологии, библиотеки и фреймворки. Фреймворк - это программная платформа, определяющая структуру программной системы, программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. Фреймворки предназначены для решения проблемы однотипности задач при каждой разработке - способ аутентифицировать пользователей (вход, выход, регистрация), панель управления сайтом, формы, инструменты для загрузки файлов и т. д. Python имеет два самых популярных фреймворков – Django и Flask.

Django – самый популярный фреймворк языка python с широкой аудиторией пользователей – разработчиков, что значит достаточно большое количество готовых решений. Каждый год выпускаются новые версии фреймворка, в настоящий момент 1.9. Django имеет ряд преимуществ над другими фреймворками, это:

- Понятная документация - множество примеров, объяснений и, самое главное — открытый исходный код, который очень хорошо написан.
- Встроенный ORM (Object-relational mapper) – существует множество более гибких и мощных библиотек, но Django ORM справляется со своей задачей очень даже хорошо. Большой плюс этого ORM – это полное отсутствие SQL – синтаксиса. Например, вместо:
SELECT * FROM «Users» WHERE id = 2
Используется:
Users.objects.get(id=2).
- Автоматически генерируемая панель администратора – это является одним из самых главных преимуществ фреймворка. Функциональность позволяет значительно сократить время на написание нужного интерфейса администратора, а также позволяет сразу управлять базой данных.
- Поддержка Model-Template-View. Похож с классическим MVC и позволяет хорошо отделять бизнес–логику от дизайна. Очень простая и удобная технология для вывода шаблонов на браузер.
- Высокая скорость работы. Несмотря на то, что язык программирования это python, Django имеет встроенные средства кеширования страниц и распределения нагрузки.

В целом в фреймворке Django всё строго распределено. Методы могут храниться только в одном файле, модели в другом, формы в другом и т. п. При разработке на нём программисту не требуется думать об архитектуре проекта, всё заранее продумано разработчиками фреймворка. Это для кого-то может показаться достоинством, для кого-то, наоборот, недостатком.

Flask – микро-фреймворк языка python. Что подразумевается под «микро»? На первый взгляд на Flask сразу становится понятным, что это очень простой фреймворк, у которого отсутствуют какие–нибудь лишние вещи. Отсутствует готовый интерфейс администратора, отсутствует шаблонизатор встроенного ORM. То есть Flask – это только ядро для начала Web–программирования на python. Разработчику требуется всё самому подключить в проект. Многие скажут, что это недостаток, но Flask на то и микро-фреймворк. Он дает полную свободу действий для разработчика. Программист сам может выбирать, что использовать и как его использовать. Ему потребуется продумать архитектуру, которая будет предоставлять требуемый функционал. В проекте не будет никаких лишних элементов, библиотек и т. п. А это значит, что проекты на Flask – быстрые и расширяемые. Одним из преимуществ является то, что приложение может быть написано на одном файле, что может быть очень полезным для маленьких и простых проектов.

В заключение были рассмотрены два фреймворка Django и Flask. Они вдвоем отлично справляются со своими задачами, но у них своя философия и подход к разработке приложений. Django – более объемный фреймворк с уже готовыми решениями, Flask – простой, маленький фреймворк, но расширяемый. На мой взгляд Flask должны изучать уже более продвинутые программисты, которые имеют опыт в разработке,

потому что начинающему программисту будет трудно освоиться во Flask. Django – более строго продуманный фреймворк, и он отлично подойдет для начинающих разработчиков.

Литература

1. *Лутц М.* Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. 1280 с.