**Laravel** - это бесплатный  PHP фреймворк с открытым исходным кодом, созданный Тейлором Отвеллом для разработки веб-приложений по архитектурному шаблону MVC.

Можно сказать, что на появление Laravel повлияли остальные PHP фреймворки.

Он был создан как альтернатива фреймворку Codeigniter (о нем будет рассказано ниже), в котором было недостаточно полезных функций для разработки веб-приложений. В качестве основы Laravel выступают компоненты другого фреймворка - Symfony (о нем также, будет рассказано ниже).

Фреймворк Laravel очень популярен среди западных разработчиков веб-приложений.

С помощью менеджера пакетов Composer, фреймворк Laravel позволяет легко устанавливать и подключать различные компоненты для использования в веб-приложении.

Реализация шаблона ActiveRecord - Eloquent ORM, позволяет установить отношения между объектами базы данных веб-приложения и выстраивать удобные запросы для манипуляции данными.

Механизм автозагрузки классов позволяет не подключать вручную файлы через include и предотвращает загрузку не используемых компонентов.

Удобная система миграций помогает упростить развертывание и обновление веб-приложения.

В Laravel есть встроенная поддержка движка шаблонов Blade, с помощью которого  можно делать простые представления веб-приложения используя специальный синтаксис.

При создании приложения можно использовать Artisan - интерфейс командной строки для ввода встроенных команд, а также создания своих собственных.

В Laravel есть много полезных функций, позволяющих сделать процесс разработки веб-приложений быстрым, простым и качественным.

Из подобных функций можно отметить dd() - удобный аналог стандартной функции PHP var\_dump(). Функция выводит информацию переменной в более понятной форме, разделяя данные на дерево атрибутов и значений, в возможностью поиска и перехода по ним.

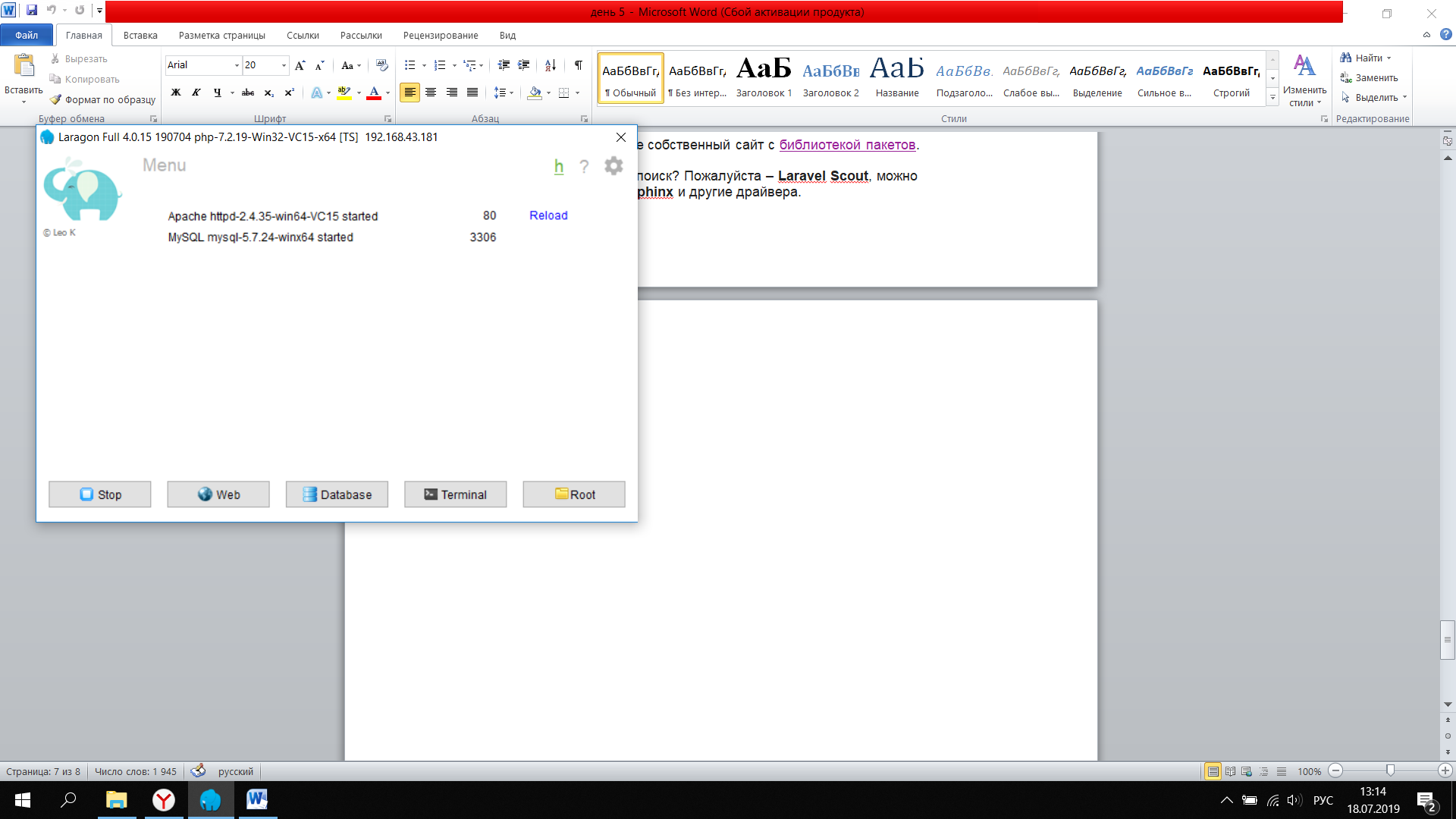
### Плюсы

* Достаточно неплохая и понятная документация.
* Вокруг фреймворка создана мощная экосистема. Различные курсы, конференции, обучающие материалы позволяют собрать вокруг фреймворка большое количество разработчиков и спонсоров, которые заинтересованы в развитии инструмента и принимают в этом участие. Да, здесь чувствуется запах маркетинга, и неплохой.
* Одним из самых очевидных плюсов Laravel, является гибкая система маршрутизации, позволяющая составить самые разные проверки маршрута веб-приложения. Вы можете выделить маршруты в специальные группы, использовать пространство имен, указать параметры маршрута, использовать регулярные выражения, настроить поддоменную маршрутизацию и многое другое.
* В Laravel много синтаксического сахара. Синтаксис API фреймворка достаточно простой и понятный. Здесь нет длинных и сложных конструкций, а только краткие и продуманные названия функций.
* Laravel содержит удобный механизм обработки ошибок и исключений.
* Фреймворк включает в себя встроенные механизмы аутентификации и авторизации пользователей, которую можно перенастроить под свои потребности.
* Laravel предоставляет из коробки механизмы для кэширования веб-приложения с помощью Memcached и Redis. Кроме этого есть удобные функции для использования простого файлового кэширования данных.
* Laravel предоставляет чистый и простой API поверх популярной библиотеки SwiftMailer с драйверами для SMTP, Mailgun, SparkPost, Amazon SES и sendmail, чтобы сделать отправку почты через локальную или облачную службу по выбору. В том числе есть механизм для построения очередей отправки почты.
* Laravel Cashier обеспечивает выразительный, свободный интерфейс к сервисам биллинга по подписке Stripe и Braintree.

### Минусы

* Для русскоязычных разработчиков, без знания английского языка или его слабым знанием, к минусам фреймворка можно отнести довольно небольшое количество статей, примеров кода, переводов официальной документации. Для тех, кто знает английский на уровне чтения технической документации, данный минус можно опустить.
* Синтаксический сахар в Laravel как плюс, так может быть и минусом. Очень легко привыкнуть к нему и позабыть, как пишутся чистые запросы и функции.
* Нарушение обратная совместимости между версиями фреймворка.
* Не логичное расположение каталогов и файлов. Например, по умолчанию в прямо в каталоге /app расположена модель User.php, которую логичней было бы расположить в каталоге /app/Models. Каталог resources с файлами представления размещен в корне приложения, хотя логичней будет его разместить в /app/resources.

**Установка Laragon**



**Laravel. Создание проекта**

