PHP — наиболее популярный в мире серверный скриптовый язык. Он прошел большой путь развития от небольших, встраиваемых в код статических HTML страниц, сниппетов, до современного языка, на котором разрабатывается большинство современных динамических сайтов. Сложность и функциональность современных сайтов только возрастает, и ни у кого нет желания писать весь необходимый код с нуля. Программистам необходимо разрабатывать сложные сайты и веб-приложения, а это обычно занимает очень много времени. Чтобы облегчить процесс разработки программистам, начали создаваться фреймворки.

Laravel — это фреймворк для web-приложений с выразительным и элегантным синтаксисом. Он позволит упростить решение основных наболевших задач, таких как аутентификация, маршрутизация, сессии и кэширование. Laravel — это попытка объединить всё самое лучшее, что есть в других PHP фреймворках. Основные преимущества Laravel:

большая экосистема с мгновенным разворачиванием своей платформы. Официальный сайт предоставляет множество мануалов и информации для ознакомления;

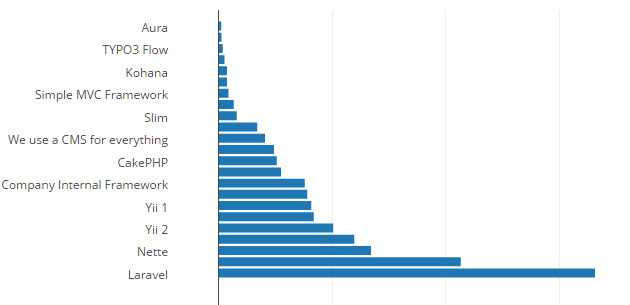
документация Laravel близка к совершенству;

у Laravel есть свой движок для шаблонов Blade, «красивый» синтаксис языка, который способствует решению всех необходимых задач, таких как аутентификация, сессии, кэширование и маршрутизация RESTful.

Несмотря на свою молодость (первый релиз вышел в 2011 году), это уже совершенно зрелый продукт, и, согласно опросу, проведенному порталом SitePoint, он занимает первое место по популярности среди разработчиков на PHP.

### Популярность фреймворка Laravel

Для того чтобы оценить популярность фреймворка, предлагаю обратиться к нескольким источникам статистики использования. Статистика популярности от Sitepoin за 2015 год.



### Как видно из графиков и статистики от SitePoint, а также статистике Google Trends, фреймворк Laravel занимает уверенное первое место и постоянно набирает популярность, не сбавляя обороты. Что, в общем, очень хорошо, учитывая, что это общемировая статистика.

### Возможности Laravel

**Пакеты** — позволяют создавать и подключать модули в формате Composer к приложению на Laravel. Многие дополнительные возможности уже доступны в виде таких модулей.

**Eloquent ORM** — реализация шаблона проектирования ActiveRecord на PHP. Позволяет строго определить отношения между объектами базы данных. Стандартный для Laravel построитель запросов Fluent поддерживается ядром Eloquent.

**Логика приложения** — часть разрабатываемого приложения, объявленная либо при помощи контроллеров, либо маршрутов.  
Обратная маршрутизация связывает между собой генерируемые приложением ссылки и маршруты, позволяя изменять последние с автоматическим обновлением связанных ссылок. При создании ссылок с помощью именованных маршрутов Laravel автоматически генерирует конечные URL.

R**EST-контроллеры** — дополнительный слой для разделения логики обработки GET- и POST-запросов HTTP.

**Автозагрузка классов** — механизм автоматической загрузки классов PHP без необходимости подключать файлы их определений в include. Загрузка по требованию предотвращает загрузку ненужных компонентов; загружаются только те из них, которые действительно используются.

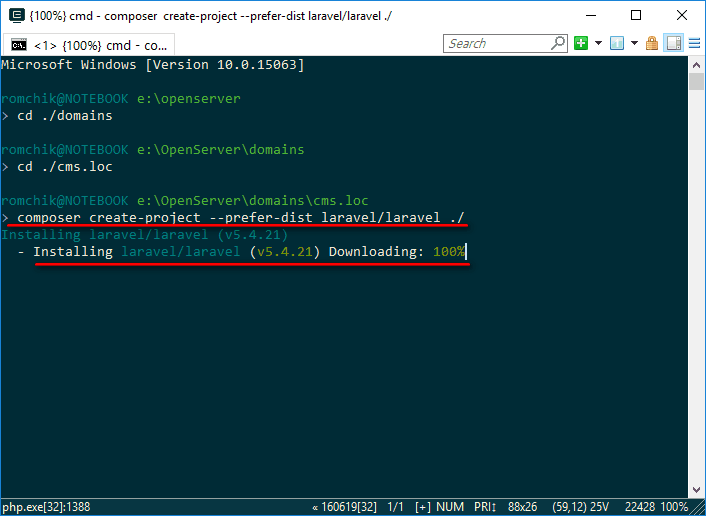
**Составители представлений** — блоки кода, которые выполняются при генерации представления (шаблона).  
Инверсия управления — позволяет получать экземпляры объектов по принципу обратного управления. Также может использоваться для создания и получения объектов-одиночек.

**Миграции** — система управления версиями для баз данных. Позволяет связывать изменения в коде приложения с изменениями, которые требуется внести в структуру БД, что упрощает развёртывание и обновление приложения.

**Модульное тестирование (юнит-тесты)** — играет очень большую роль в Laravel, который сам по себе содержит большое число тестов для предотвращения ошибок.

**Страничный вывод** — упрощает генерацию страниц, заменяя различные способы решения этой задачи единым механизмом, встроенным в Laravel.

**Установка Laragon**



**Laravel. Создание проекта**

