

## СБОРКА БАЗОВОГО САЙТА (ПРИЛОЖЕНИЯ) НА LARAVEL FRAMEWORK 7.X

Данный материал содержит инструкцию по сборке базового сайта (приложения) на **Laravel Framework 7.x** под операционной системой Windows с установленным **Open Server**. Open Server Panel — это портативная серверная платформа и программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий.

Скачать Open Server можно здесь <https://ospanel.io/>. Установка его абсолютно интуитивно понятна.

Весь материал, касающийся именно базовой настройки Laravel, так же справедлив и для таких операционных систем, как Mac OS и Linux, естественно с учётом их особенностей.

### I. Установка Composer на OpenServer

1) переходим к текущему модулю PHP, например:

D:\OSPanel\modules\php\PHP\_7.3-x64

2) запускаем:

```
php -r "readfile('https://getcomposer.org/installer');" | php
```

или Composer-Setup.exe, указывая D:\OSPanel\modules\php\PHP\_7.3-x64

3) Проверяем какая версия стоит командой:

```
php composer.phar -V
```

4) Создаём .bat файл и в дальнейшем можно будет работать без указания php composer.phar, просто указывая composer:

```
echo @php "%~dp0composer.phar" %*>composer.bat
```

### II. Установка Laravel

1. Запускаем консоль OpenServer
2. Переходим в папку \OSPanel\domains
3. Запускаем команду  
**composer create-project --prefer-dist laravel/laravel laravel.empty.blog**  
laravel.empty.blog – каталог нашего проекта
4. Устанавливаем Node.JS <https://nodejs.org/en/>
5. Запускаем OpenServer, и идем в настройки. Во вкладке «Сервер» в поле «Настройка использования переменной Path» выставляем «Свой Path + userdata/config/path.txt». Создаем файл в папке \OpenServer\userdata\config\ под именем **path.txt** следующего содержания:  
**C:\Program Files\nodejs\**

6. Перезапускаем OpenServer
7. Проверяем **npm -i**
8. Запускаем консоль, переходим в паку проекта **\OSPanel\domains\laravel.empty.blog**
9. Запускаем установку **npm install**
10. В OpenServer идем в настройки. Во вкладке «Домены» добавляем папку **\OSPanel\domains\laravel.empty.blog\public** -> имя домена **laravel.empty.blog**
11. Проверяем в браузере <http://laravel.empty.blog>
12. В OpenServer идём в «Дополнительно» phpmyadmin (логин root без пароля)
13. Создаём базу **laravel.empty.bl** utf8mb4\_general\_ci
14. Редактируем файл **.env** в **\OSPanel\domains\laravel.empty.blog\**

```
APP_NAME=Laravel.Empty.Blog
APP_URL=http://laravel.empty.blog
DB_DATABASE=laravel.empty.bl
```

Можно настроить smtp (на примере Яндекс):

```
MAIL_DRIVER=smtp
MAIL_HOST=smtp.yandex.ru
MAIL_PORT=587
MAIL_USERNAME=ВАШ_ЛОГИН
MAIL_PASSWORD=ВАШ_ПАРОЛЬ
MAIL_ENCRYPTION=tls
MAIL_FROM_ADDRESS=ВАШ_EMAIL
MAIL_FROM_NAME="${APP_NAME}"
```

15. В консоли в папке **\OSPanel\domains\laravel.empty.blog\** запускаем команду **php artisan config:cache**

16. В **:\OSPanel\domains\laravel.empty.blog\resources\views\welcome.blade.php** меняем:

```
<title>Laravel</title>
на <title>{{ config('app.name', 'Laravel') }}</title>
Надпись Laravel в <div class="title m-b-md">
на {{ config('app.name', 'Laravel') }} ;)
```

### III. Запуск сайта (приложения) на Laravel.

17. Проверяем в браузере <http://laravel.empty.blog>

Мы должны увидеть: **Laravel.Empty.Blog ;)**

18. Открываем папку нашего проекта в PHPStorm или в другой IDE

19. Версию установленного Laravel можно проверить командой (в консоли в папке проекта):

```
php artisan -V
```

В нашем случае это **Laravel Framework 7.6.2**

20. Список всех команд: **php artisan**

21. Модифицируем файл **app/Providers/AppServiceProvider.php**

Без этих изменений может возникать ошибка:

Syntax error or access violation: **1071 Specified key was too long**; max key length is 767 bytes

Дело в том, что в Laravel 5.4 изменилась кодировка по умолчанию для базы данных (теперь это utf8mb4 для поддержки emoji). Ошибка Specified key was too long error проявляется только для MySQL ниже v5.7.7 и в старых версиях MariaDB.

Итак, в **app/Providers/AppServiceProvider.php** добавляем:

- 1) Добавьте строку в блоке use: **use Illuminate\Support\Facades\Schema;**
- 2) Добавьте в метод boot строку: **Schema::defaultStringLength(191);**

22. Запускаем миграции: **php artisan migrate**

Вариант сообщений при успешном выполнении миграций:

Migration table created successfully.

Migrating: 2014\_10\_12\_000000\_create\_users\_table

Migrated: 2014\_10\_12\_000000\_create\_users\_table (0.44 seconds)

Migrating: 2019\_08\_19\_000000\_create\_failed\_jobs\_table

Migrated: 2019\_08\_19\_000000\_create\_failed\_jobs\_table (0.24 seconds)

В базе данных **laravel.empty.bl** создались таблицы **failed\_jobs** и **users**

23. Добавляем механизм авторизации пользователей

Каркас фронтенда, обычно поставляемый с предыдущими версиями Laravel, был перенесен в отдельный пакет **laravel/ui**. Это позволяет создавать и версионировать пользовательские интерфейсы отдельно от основного фреймворка. В результате этого изменения в дефолтном каркасе фреймворка не будет Bootstrap и Vue.

Также была вынесена из фреймворка команда **make:auth**. Чтобы восстановить традиционный каркас Vue/Bootstrap, имеющийся в предыдущих версиях Laravel, вы можете установить пакет **laravel/ui** и использовать **artisan** команду **ui** для установки фронтенд каркаса.

Устанавливаем пакет: **composer require laravel/ui**

Устанавливаем фронтенд: **php artisan ui vue --auth**

Загружаем пакеты фронтенда: **npm install**

Запускаем нашу сборку в режиме разработчика: **npm run dev**

В результате последней команды будут скомпелированы файлы стилей и скриптов  
**/css/app.css**  
**/js/app.js**

## 24. Процесс регистрации.

Если всё сделано правильно, то перейдя на наш сайт <http://laravel.empty.blog> мы увидим в верхнем углу 2 пункта меню Login и Register.

Производим регистрацию нового пользователя – вводим имя, e-mail, пароль и подтверждение пароля. После успешной регистрации вы должны увидеть надпись You are logged in!

Это значит, что регистрация прошла успешно и Вы уже авторизованы на сайте!

Поздравляем! Базовая сборка сайта на **Laravel 7** произведена, и теперь Ваш сайт полностью готов к дальнейшему развитию!

## 25. Git

Весь код к данному материалу находится в общедоступном репозитории [https://github.com/Best-ITPro/LaravelEmptyBlog\\_Blank.git](https://github.com/Best-ITPro/LaravelEmptyBlog_Blank.git)

Вы можете скачать и использовать его для создания Ваших приложений.

В дальнейшем будет опубликована более расширенная версия сайта (приложения), которая будет содержать механизм публикации статей в различных категориях.

С уважением,  
**Команда Best IT Pro**  
<http://best-itpro.ru>