Laravel WebSocket Installation & Configuration

BeyondCode (https://beyondco.de/) package don't work with laravel 8

1 Laravel Installation.

> *Prerequisites*: It assumes you have `PHP, MySQL, Composer, Node.js, Npm` installed on your system.

*Run the commands marked in **bold dark-blue***

```
composer create-project --prefer-dist  laravel/laravel:^7.0 project_name
composer create-project --prefer-dist  laravel/laravel project_name
```

```
composer create-project --prefer-dist  laravel/laravel laravel-websocket
```

```
composer require beyondcode/laravel-websockets

php artisan vendor:publish --
provider="BeyondCode\LaravelWebSockets\WebSocketsServiceProvider" --
tag="migrations"

php artisan vendor:publish --provider="BeyondCode\LaravelWebSockets\WebSocketsServiceProvider" --
tag="migrations"

composer require --with-all-dependencies beyondcode/laravel-websockets
```

```
cd project_name
php artisan –V
Laravel Framework 7.30.4
```

```
git init
git add .
git commit –m "Comment"
```

```
git remote add origin https://github.com/SergeyHub/repository-name.git
git push -u origin master
```

Edit `env.` file

```
mysql -u root -p
create database db_name;
drop    database db_name;
exit
```

```
php artisan migrate

php artisan vendor:publish --provider="BeyondCode\LaravelWebSockets\WebSocketsServiceProvider" --
tag="config"
```

Edit env. file

```
PUSHER_APP_ID=anyId
PUSHER_APP_KEY=anyKey
PUSHER_APP_SECRET=anySecret
PUSHER_APP_CLUSTER=mt1
```

```
php artisan serve
Laravel development server started: http://127.0.0.1:8000

http://127.0.0.1:8000/laravel-websockets
```

E:\laravel7-websocket\config**websockets.php**

```
/*
 * This path will be used to register the necessary routes for the package.
 */
'path' => 'laravel-websockets',
//'path' => 'admin/websocket',
```

2. Basic Usage

Pusher Replacement

Requirements #

```
composer require pusher/pusher-php-server "~3.0"
```

Edit `env.` file

```
BROADCAST_DRIVER=pusher
```

Pusher Configuration #

When broadcasting events from your Laravel application to your WebSocket server, the default behavior is to send the event information to the official Pusher server. But since the Laravel WebSockets package comes with its own Pusher API implementation, we need to tell Laravel to send the events to our own server.

To do this, you should add the **host** and **port** configuration key to your **config/broadcasting.php** and add it to the **pusher section**. The default port of the Laravel WebSocket server is 6001.

Edit E:\laravel7-websocket\config**broadcasting.php**

```php
'pusher' => [
    'driver' => 'pusher',
    'key' => env('PUSHER_APP_KEY'),
    'secret' => env('PUSHER_APP_SECRET'),
    'app_id' => env('PUSHER_APP_ID'),
    'options' => [
        'cluster' => env('PUSHER_APP_CLUSTER'),
        'useTLS' => true,
        'host' => '127.0.0.1',
        'port' => 6001,
        'scheme' => 'http'
    ],
],
```

Usage with Laravel Echo #

```
npm install
npm install laravel-echo pusher-js
```

Edit E:\laravel7-websocket\resources\js**bootstrap.js**

```
import Echo from 'laravel-echo';

window.Pusher = require('pusher-js');

window.Echo = new Echo({
    broadcaster: 'pusher',
    key: process.env.MIX_PUSHER_APP_KEY,
    cluster: process.env.MIX_PUSHER_APP_CLUSTER,
    wsHost: window.location.hostname,
    wsPort: 6001,
    disableStats: true,
});
```

```
npm run watch
```

Run in the second terminal

```
php artisan websocket:serve
```

Run in the third terminal

```
php artisan serve
http://127.0.0.1:8000/laravel-websockets
click connect
```

Run in the next terminal

```
php artisan make:event WebsocketDemoEvent
```

Edit E:\laravel7-websocket\app\Events**WebsocketDemoEvent.php**

```php
class WebsocketDemoEvent implements ShouldBroadcast
{
    use Dispatchable, InteractsWithSockets, SerializesModels;

    public $event_data;
    /**
     * Create a new event instance.
     *
     * @return void
     */
    public function __construct($event_data)
    {
        $this -> event_data = $event_data;
    }

    /**
     * Get the channels the event should broadcast on.
     *
     * @return \Illuminate\Broadcasting\Channel|array
     */
    public function broadcastOn()
    {
        //return new PrivateChannel('channel-name');
        return new Channel('DemoChannel');
    }
}
```

Edit E:\laravel7-websocket\resources\js**bootstrap.js**

```js
window.Echo.channel('DemoChannel').listen('WebsocketDemoEvent' , (e) => {
    console.log(e);
});
```

E:\laravel7-websocket\routes**web.php**

```php
use Illuminate\Support\Facades\Route;
use App\Events\WebsocketDemoEvent;
/*
|--------------------------------------------------------------------------
| Web Routes
|--------------------------------------------------------------------------
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::get('/', function () {

    broadcast(new WebsocketDemoEvent('event_data'));

    return view('welcome');
});
```

Edit E:\laravel7-websocket\resources\views**welcome.blade.php**

```html
<meta name='csrf-token' content="{{ csrf_token() }}">
<body>
    <div id="app" class="flex-center position-ref full-height">

    <script src="js/app.js"></script>
</body>
</html>
```

```
\book>git init
\book>git add .
\book>git commit -m "Init"
\book>git remote add origin https://github.com/SergeyHub/Book.git
\book>git push -u origin master
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com': SergeyHub
Password for 'https://SergeyHub@github.com': my_password
```

```
composer require laravel/ui
composer require laravel/ui "^2.4"          7
```

```
php artisan ui bootstrap
php artisan ui bootstrap --auth

php artisan ui vue
php artisan ui vue --auth

npm i axios vform
```

```
npm install  && npm run dev
npm run watch

php artisan migrate
php artisan serve
```

```
sudo chmod -R 775 storage/framework
sudo chmod -R 775 ./storage/logs
chmod -R 775 bootstrap/cache

sudo service apache2 restart
```

sudo chmod -R 777 /var/www/html Never use 777. With 777 permissions, you are giving anyone with a connection full access to the files or directories with those permissions. They may alter them in any way they choose, including maliciously. Many account hacking incidents stem from 777 permissions.

the requested url was not found on this server. Laravel

```
1. edit the apache2 conf to allow for symlinks etc
sudo nano /etc/apache2/apache2.conf

2. Add this to the file

<Directory /var/www/html/>
  Options Indexes FollowSymLinks
  AllowOverride all
  Require all granted
</Directory>

3. Restart your apache2 server

sudo service apache2 restart
```