

Step 1: Follow laravel pusher setup docs first

Step 2: ekta view and controller create kora holo –

From web.php,

```
Route::get('/chat', "ChatController@showChat")->name('chat.show');
```

From ChatController.php,

```
<?php

namespace App\Http\Controllers;

use App\Events\MessageSent;
use Illuminate\Http\Request;

class ChatController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }

    public function showChat()
    {
        return view('chat.show');
    }

    // public function sendMessage(Request $request){
    //
    //
    //     broadcast(new MessageSent($request->user, $request->message));
    //
    //     return "broadcast";
    // }
}
```

View chat->show.blade.php,

```
@extends('layouts.app')

@push('styles')
    <style type="text/css">

    </style>
@endpush

@section('content')
    <div class="container-fluid">
        <div class="row justify-content-center">
            <div class="col-md-12">
                <div class="card">
                    <div class="card-header">{{ __('Chat') }}</div>

                    <div class="card-body">
                        <div class="row p-2">
                            <div class="col-10">
                                <div class="row">
                                    <div class="col-12 border rounded-lg p-3">
                                        <ul>
                                            id="messages"
                                            class="list-unstyled overflow-auto"
                                            style="height: 45vh"
                                        >
                                            {{-- <li>Test 1:
                                            Hello</li>--}}
                                            {{-- <li>Test 2:
                                            Hi There</li>--}}
                                        </ul>
                                    </div>
                                </div>
                            </div>
                        </div>

                        <form action="">
                            <div class="row py-3">
                                <div class="col-10">
                                    <input type="text" id="message" class="form-control">
                                </div>
                                <div class="col-2">
                                    <button id="send" type="submit" class="btn btn-primary btn-
block">Send</button>
                                </div>
                            </div>
                        </form>
                    </div>

                    <div class="col-2">
                        <p><strong>Online Now</strong></p>
                        <ul>
                            id="users"
                            class="list-unstyled overflow-auto text-info"
                            style="height: 45vh"
                        >
                            {{-- <li>Test 1</li>--}}
                            {{-- <li>Test 2</li>--}}
                        </ul>
                    </div>
                </div>
            </div>
        </div>
    </div>
@endsection
```

Then channels.php te channel route kora holo,

```
Broadcast::channel('chat', function ($user) {  
    // return $user != null;  
  
    if($user != null){  
        return ['id' => $user->id, 'name' => $user->name];  
    }  
});
```

Next e ekta event create kora holo **MessageSent** naam e,

```
<?php  
  
namespace App\Events;  
  
use App\User;  
use Illuminate\Broadcasting\Channel;  
use Illuminate\Broadcasting\InteractsWithSockets;  
use Illuminate\Broadcasting\PresenceChannel;  
use Illuminate\Broadcasting\PrivateChannel;  
use Illuminate\Contracts\Broadcasting\ShouldBroadcast;  
use Illuminate\Foundation\Events\Dispatchable;  
use Illuminate\Queue\SerializesModels;  
  
class MessageSent implements ShouldBroadcast  
{  
    use Dispatchable, InteractsWithSockets, SerializesModels;  
  
    public $user;  
    public $message;  
  
    /**  
     * Create a new event instance.  
     *  
     * @return void  
     */  
    public function __construct($user, $message)  
    {  
        $this->user = $user;  
        $this->message = $message;  
    }  
  
    /**  
     * Get the channels the event should broadcast on.  
     *  
     * @return \Illuminate\Broadcasting\Channel|array  
     */  
    public function broadcastOn()  
    {  
        \Log::debug($this->user . $this->message);  
        return new PresenceChannel('chat');  
    }  
}
```

Lastly script code add kora holo chat->show.blade.php te,

```
@push('scripts')
<script>
    const userElement = document.getElementById("users");
    const messageElement = document.getElementById("message");
    const sendData = document.getElementById("send");
    let messages = document.getElementById("messages");

    Echo.join('chat').here(users => {
        users.forEach((user, index) => {
            // console.log(users);
            let element = document.createElement("li");
            element.setAttribute('id', user.id);
            element.innerText = user.name;
            userElement.appendChild(element);
        });
    }).joining(user => {
        let element = document.createElement("li");
        element.setAttribute('id', user.id);
        element.innerText = user.name;
        userElement.appendChild(element);
    }).leaving(user => {
        const element = document.getElementById(user.id);
        element.parentNode.removeChild(element);
    }).listen('MessageSent', el => {

        let gg = document.createElement('li');
        gg.innerText = el.user + " : " + el.message;
        messages.appendChild(gg);
    });
</script>

<script>
    sendData.addEventListener('click', function (e) {
        e.preventDefault();

        window.axios.post("/chat/message", {
            user: '{{ auth()->user()->name }}',
            message: messageElement.value,
        }).then(el =>{
            // console.log(el);
        });

        messageElement.value = "";
    });
</script>
@endpush
```

Ekhane jei channel ta use kora hoise sheta hocche,

1. **PresenceChannel**
2. **Echo.join()** hocche PresenceChannel er ekta part jekhane **.here()** hocche jokhon keu direct page e access ney server e and data hocche direct ekhan theke fetch hoy

```
Broadcast::channel('chat', function ($user) {  
    // return $user != null;  
  
    if($user != null){  
        return ['id' => $user->id, 'name' => $user->name];  
    }  
});
```

3. **joining()** hocche after join kora, kono array expect kore na
4. **leaving()** hocche Jodi keu leave ney server theke
5. **listen()** hocche kono Broadcast Event ke listen kora.

Full show.blade.php code,

```
@extends('layouts.app')  
  
@push('styles')  
    <style type="text/css">  
  
    </style>  
@endpush  
  
@section('content')  
    <div class="container-fluid">  
        <div class="row justify-content-center">  
            <div class="col-md-12">  
                <div class="card">  
                    <div class="card-header">{{ __('Chat') }}</div>  
  
                    <div class="card-body">  
                        <div class="row p-2">  
                            <div class="col-10">  
                                <div class="row">  
                                    <div class="col-12 border rounded-lg p-3">  
                                        <ul  
                                            id="messages"  
                                            class="list-unstyled overflow-auto"  
                                            style="height: 45vh"  
                                        >  
                                            {{-- <li>Test 1: Hello</li>--}}  
                                            {{-- <li>Test 2: Hi There</li>--}}  
                                        </ul>  
                                    </div>  
                                </div>  
                            </div>  
  
                            <div class="">  
                                <div class="row py-3">  
                                    <div class="col-10">  
                                        <input type="text" id="message" class="form-control">  
                                    </div>  
                                    <div class="col-2">  
                                        <button id="send" type="submit" class="btn btn-primary btn-block">Send</button>  
                                    </div>  
                                </div>  
                            </div>  
                        </div>  
                    </div>  
                </div>  
            </div>  
        </div>  
    </section>
```

```

        </div>
      </form>
    </div>

    <div class="col-2">
      <p><strong>Online Now</strong></p>
      <ul
        id="users"
        class="list-unstyled overflow-auto text-info"
        style="height: 45vh"
      >
        {{-- <li>Test 1</li>--}}
        {{-- <li>Test 2</li>--}}
      </ul>
    </div>

  </div>
</div>
</div>
</div>
</div>
@endsection

@push('scripts')
<script>
  const userElement = document.getElementById("users");
  const messageElement = document.getElementById("message");
  const sendData = document.getElementById("send");
  let messages = document.getElementById("messages");

  Echo.join('chat').here(users => {
    users.forEach((user, index) => {
      // console.log(users);
      let element = document.createElement("li");
      element.setAttribute('id', user.id);
      element.innerText = user.name;
      userElement.appendChild(element);
    });
  }).joining(user => {
    let element = document.createElement("li");
    element.setAttribute('id', user.id);
    element.innerText = user.name;
    userElement.appendChild(element);
  }).leaving(user => {
    const element = document.getElementById(user.id);
    element.parentNode.removeChild(element);
  }).listen('MessageSent', el => {

    let gg = document.createElement('li');
    gg.innerText = el.user + " : " + el.message;
    messages.appendChild(gg);
  });
</script>

<script>
  sendData.addEventListener('click', function (e) {
    e.preventDefault();

    window.axios.post("/chat/message", {
      user: '{{ auth()->user()->name }}',
      message: messageElement.value,
    }).then(el => {
      // console.log(el);
    });

    messageElement.value = "";
  });
</script>
@endpush

```

Used parts for this project,

1. ChatController
2. MessageSent event
3. Channels.php
4. /chat, /chat/message route from web.php