

Step 0 : setup docx ta follow kore nite hobe.

Step 1 : Layout design and configuration setup hubohu “**Creating a Chat Room Using Realtime Messages with Laravel Echo**” ke follow korte hobe.

Step 2: from **chat->show.blade.php** te ekta attribute create korte hobe click method ke trigger korar jonno,

```
element.setAttribute('onclick', 'greetUser("' + user.id + '")');
```

```
Echo.join('chat').here(users => {
    users.forEach((user, index) => {
        // console.log(users);
        let element = document.createElement("li");
        element.setAttribute('id', user.id);
        element.setAttribute('onclick', 'greetUser("' + user.id
+ '")');

        element.innerText = user.name;
        userElement.appendChild(element);

    });
}).joining(user => {
    let element = document.createElement("li");
    element.setAttribute('id', user.id);
    element.setAttribute('onclick', 'greetUser("' + user.id +
'")');

    element.innerText = user.name;
    userElement.appendChild(element);
})
```

Then niche ekta script banate hobe controller e data pathanor jonno,

```
<script>
    function greetUser(id) {
        window.axios.post('/chat/greet/' + id).then(
            el => {
                // console.log(el);
            }
        );
    }
</script>
```

Ekta **event** banano holo **GreetingSent** 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 GreetingSent implements ShouldBroadcast
{
    use Dispatchable, InteractsWithSockets, SerializesModels;

    private $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->message);
    return new PrivateChannel("chat.greet.{ $this->user}");
}
}

```

Then ChatController ar web.php er route thik kore nite hobe,

From **web.php**,

```

// ----- Private Chat room
-----

Route::post('/chat/greet/{user}',
"ChatController@greetReceived")->name('chat.greet');

// ----- Private Chat room
-----

```

And from **channels.php**,

```

Broadcast::channel('chat.greet.{receiver}', function ($user, $receiver) {
    return (int) $user->id === (int) $receiver;
});

```

And method from **ChatController**,

```

public function greetReceived(Request $request, User $user){

    broadcast(new GreetingSent($user->id,"{$request->user-> name}
greeted you"));
    broadcast(new GreetingSent($request->user->id,"You Greeted
{$user->name}"));

    return "Greeting {$user->name}";
}

```

Ekhn lastly ei event ta ke jevabe fetch kore kaj korte hobe,

```
<script>
    Echo.private("chat.greet.{ { auth()->user()->id
}}").listen("GreetingSent", el => {
        let gg = document.createElement('li');
        gg.setAttribute("style", "color : green");
        gg.innerText = el.message + " greeted you";
        messages.appendChild(gg);
    });
</script>
```

***** Description *****

1. Private channel and specific vabe kauke pathanor jonno **\$receiver** add kora hoise jekhane condition ta hocche jodi **\$user->id** and **\$receiver** jodi match kore tahole oi event trigger korbe otherwise trigger korbe na
2. **GreetingSent** event e **PrivateChannel** kore pathano hoise cause data privately jabe.

```
public function broadcastOn()
{
    \Log::debug($this->message);
    return new PrivateChannel("chat.greet.{ $this->user }");
}
```