1. AssignTicket Model

public static function checkAssign($id){

$assigns = AssignTicket::where('ticket\_id',$id)->get();

$ids = '';

foreach($assigns as $assign){

if($assign->user->department\_id == Auth::user()->department\_id && $assign->user->level == Auth::user()->level){

$ids = $assign->user->id;

}

}

return $ids;

}

1. Index.blade.php of ticket

@if(\App\Model\AssignTicket::checkAssign($ticket->id))

Online 147

1. TicketController@show

public function show($id)

{

$this->readComment($id); // to read a coment by a valid user.

$assignTickets = AssignTicket::where('ticket\_id', $id)->with('user','user\_assigned\_by')->get();

$assignedUser = AssignTicket::where('ticket\_id',$id)->get();

foreach($assignedUser as $assign){

if($assign->user->department\_id == Auth::user()->department\_id && $assign->user->level == Auth::user()->level){

$user = TRUE;

$assign->status = 1;

$assign->save();

}else{

$user = FALSE;

}

}

$departments = Department::all();

$ticket = Ticket::find($id);

$responses = Response::where('ticket\_id',$id)->get();

return view('ticket.view',compact('ticket', 'responses', 'departments', 'user', 'assignTickets'));

}

1. Aja in create for deadline create.blade.php in line 351

$('select#issue').change(function(){

var issueId = $(this).children("option:selected").val();

console.log(issueId)

$.ajax({

url: "{{url('/getCriticalityByIssue')}}",

data:{

\_token: CSRF\_TOKEN,

issue\_id: issueId,

},

dataType: 'JSON',

success:function(data){

var issue = data.msg;

console.log(issue)

$('#criticality').val(issue.criticality)

var today = new Date()

today.toDateString()

today.setHours(today.getHours() + issue.hour);

var day = today.getDate()

var month = today.getMonth()+1

var year = today.getFullYear()

var hour = today.getHours()

var minute = today.getMinutes()

var second = today.getSeconds()

var date = year+'-'+month+'-'+day+' '+hour+':'+minute+':'+second;

console.log(date)

$('#datepicker').val(date)

}

})

});

1. TicketController overrite function by below given

public function getCriticalityByIssue(Request $request)

{

$data = Issue::where('id', $request->issue\_id)->first();

$response = array(

'status' => 'success',

'msg' => $data

);

return response()->json($response);

}