

Poblacion 5 General Mariano Alvarez Cavite

SOURCE CODE



```
NotifyDue.php
                                      * The console command
                                 description.
<?php
                                      * @var string
namespace
App\Console\Commands;
                                      */
                                     protected $description =
                                 'Send Notification to user';
use App\Events\DueTask;
App\Http\Traits\InstrumentsTr
                                    /**
ait;
                                      * Execute the console
use App\Models\Notification;
                               command.
use App\Models\TaskAssign;
use Carbon\Carbon;
                                      * @return int
                                      */
use
Illuminate\Console\Command;
                                     public function handle()
                                     {
class NotifyDue extends
                                         $dateNow =
Command
                                 Carbon::now()->format('Y-m-
                                 d');
    use InstrumentsTrait;
    /**
                                         $dueTask =
                                 TaskAssign::where('due',
     * The name and signature
                                 $dateNow)
of the console command.
                                 >whereNull('notified')
     * @var string
                                             ->get(['id',
                                 'instrumentId', 'userId',
     */
                                 'accredId'])
    protected $signature =
                                             ->map(function
'notify:due';
                                 ($due) {
    /**
```



```
$parent =
                                                  unset ($due-
$this->getParents($due-
                                 >instrument, $due-
>instrumentId, true, null);
                                 >accreditation);
                $accred =
$due->accreditation;
                                                  return $due;
                                              });
if(count($parent) > 0){
                                         //echo $dueTask;
                     $due-
>area =
$parent[count($parent)-1]-
                                         foreach ($dueTask as
>only('title');
                                 $value) {
                     $due-
                                              $notif =
>parameter =
                                 Notification::create([
$parent[count($parent)-2]-
>only('title');
                                 'userId'=>$value->userId,
                }else{
                     $due-
                                 'action'=>'due',
>area = null;
                                                  'details' =>
                     $due-
>parameter = null;
                                                      'You have
                                 one due task today in
                }
                                 '.$value-
                                 >program['abbreviation'].' /
                                 '.$value->level['title'].
                $due->inst =
$due->instrument;
                                 '.$value->area['title'].' /
                $due->program
                                 '.$value-
= $accred->taggedPrograms-
                                 >parameter['title'].' /
>only('abbreviation');
                                 '.$value->inst['title']
                $due->level =
                                              1);
$accred->getLevelInstrument-
>only('title');
                                              $task =
                                 TaskAssign::find($value->id);
```



```
/**
            $task->notified =
true;
                                      * The name and signature
            $task->save();
                                of the console command.
                                      * @var string
DueTask::dispatch($notif);
                                     */
                                    protected
                                                $signature =
//Duedates::dispatch([$value-
                                'survey:open';
>userId, $notif]);
        }
                                    /**
    }
                                      * The console command
}
                                description.
ScheduledSurvey.php
                                     * @var string
                                     */
<?php
                                    protected $description =
                                 'Open scheduled accreditation
                                survey';
namespace
App\Console\Commands;
                                    /**
                                     * Execute the console
use App\Events\UpdateAccred;
                                command.
use App\Models\Accreditation;
use App\Models\User;
                                     * @return int
use Carbon\Carbon;
                                     */
Illuminate\Console\Command;
                                    public function handle()
                                        $dateNow
class ScheduledSurvey extends
Command
                                Carbon::now()-
                                >timezone('Asia/Manila')-
{
                                >format('Y-m-d');
```



```
>whereNull('status')
Accreditation::where('date se
lf survey', $dateNow)
                               >whereNull('selfsurvey')
>whereNull('survey')
                               >get(['id',
                               'date actual survey',
                               'date_self_survey',
>whereNull('status')
                               'instrumentId', 'programId',
                               'restrict', 'selfsurvey',
                               'survey'])
>whereNull('selfsurvey')
                               >map(function ($accred) {
>update(['survey' => 2]);
                               return collect([
Accreditation::where('date ac
                               'id' => $accred->id,
tual survey', $dateNow)
                               'date actual survey'
>whereNull('survey')
                               $accred->date actual survey,
>update(['survey' => 1,
                               'date self survey'
                                                         =>
'restrict' => true]);
                               $accred->date self survey,
       foreach ([1, 6] as
                               'instrumentId' =>
                                                   $accred-
$value) {
                               >instrumentId,
UpdateAccred::dispatch(collec
                               'programId'
                                                   $accred-
                                             =>
t([
                               >programId,
               'auth'
                          =>
$value,
                                             =>
                                                   $accred-
                               'restrict'
                               >restrict,
               'self'
                          =>
Accreditation::where('date se
lf survey', $dateNow)
                              'selfsurvey' =>
                                                   $accred-
                               >selfsurvey,
>where('survey', 2)
```



```
'survey' => $accred->survey,
                                 'date self survey'
                                 $accred->date self survey,
'program'
                     $accred-
              =>
>taggedPrograms-
                                 'instrumentId'
                                                 =>
                                                      $accred-
>only('abbreviation')['abbrev
                                >instrumentId,
iation'],
                                 'programId'
                                                      $accred-
                                                =>
'title'
                     $accred-
                                >programId,
>getLevelInstrument-
>only('title')['title']
                                 'restrict'
                                               =>
                                                      $accred-
                        ]);
                                >restrict,
                    }),
                                 'selfsurvey'
                                                      $accred-
                                                =>
                'actual'
                           =>
                                >selfsurvey,
Accreditation::where('date ac
tual survey', $dateNow)
                                 'survey' => $accred->survey,
>where('survey', 1)
                                 'program'
                                                      $accred-
                                >taggedPrograms-
>whereNull('status')
                                >only('abbreviation')['abbrev
                                iation'],
>get(['id',
'date actual survey',
                                'title'
                                                      $accred-
'date self survey',
                                >getLevelInstrument-
'instrumentId', 'programId',
                                >only('title')['title']
'restrict',
                'selfsurvey',
                                                         1);
'survey'])
                                                     }),
>map(function ($accred) {
                                             1));
                                         }
return collect([
                                     }
'id' => $accred->id,
                                 }
'date actual survey'
                                Kernel.php
$accred->date actual survey,
```



```
$schedule-
<?php
                                >command('survey:open')-
                                >dailyAt('06:00');
namespace App\Console;
                                    }
use
                                    /**
Illuminate\Console\Scheduling
                                     * Register the commands
\Schedule;
                                for the application.
use
Illuminate\Foundation\Console
\Kernel as ConsoleKernel;
                                     * @return void
                                     */
class
          Kernel
                      extends
                                    protected
                                                     function
ConsoleKernel
                                commands()
                                    {
    /**
                                        $this-
             Define
                          the
                                >load( DIR .'/Commands');
application's
                      command
schedule.
                                        require
                                base path('routes/console.php
                       @param
                                ');
\Illuminate\Console\Schedulin
                                    }
g\Schedule $schedule
                                }
     * @return void
                                Attachment.php
     */
                                <?php
    protected
                     function
schedule(Schedule)
    {
                                namespace App\Events;
        $schedule-
>command('notify:due')-
                                use
>hourly();
                                Illuminate\Broadcasting\Chann
                                el;
```



```
{
Illuminate\Broadcasting\Inter
                                         $this->attachment
actsWithSockets;
                                 $attachment;
use
                                     }
Illuminate\Broadcasting\Prese
nceChannel;
use
                                     /**
Illuminate\Broadcasting\Priva
                                      * Get the channels the
teChannel;
                                 event should broadcast on.
use
Illuminate\Contracts\Broadcas
ting\ShouldBroadcast;
                                                        @return
                                 \Illuminate\Broadcasting\Chan
use
                                 nel|array
Illuminate\Foundation\Events\
Dispatchable;
                                      */
use
                                     public
                                                       function
Illuminate\Queue\SerializesMo
                                 broadcastOn()
dels;
Illuminate\Support\Facades\Au
                                         return
                                                            new
                                 PrivateChannel('attach.'.Auth
th;
                                 ::id());
                                     }
class Attachment
                   implements
ShouldBroadcast
                                 }
                                 DocumentComment.php
                Dispatchable,
                                 <?php
InteractsWithSockets,
SerializesModels;
                                 namespace App\Events;
                                 use
    public $attachment;
                                 Illuminate\Broadcasting\Chann
                                 el;
    public
                     function
  construct($attachment)
```



```
Illuminate\Broadcasting\Inter
                                     /**
actsWithSockets;
                                      * Get the channels the
use
                                 event should broadcast on.
Illuminate\Broadcasting\Prese
nceChannel;
use
                                                       @return
Illuminate\Broadcasting\Priva
                                 \Illuminate\Broadcasting\Chan
teChannel;
                                 nel|array
use
                                      */
Illuminate\Contracts\Broadcas
                                     public
                                                      function
ting\ShouldBroadcast;
                                 broadcastOn()
use
                                     {
Illuminate\Foundation\Events\
Dispatchable;
                                         return
                                 PrivateChannel('docuComment.'
use
                                 .$this->comment['userId']);
Illuminate\Queue\SerializesMo
dels;
                                     }
                                 }
class
              DocumentComment
implements ShouldBroadcast
                                 DocumentReview.php
                Dispatchable,
    use
InteractsWithSockets,
SerializesModels;
                                 <?php
    public $comment;
                                 namespace App\Events;
                     function
    public
                                 use
 construct($comment)
                                 Illuminate\Broadcasting\Chann
                                 el;
        $this->comment
                                 use
                                 Illuminate\Broadcasting\Inter
$comment;
                                 actsWithSockets;
```



```
* Get the channels the
Illuminate\Broadcasting\Prese
                                 event should broadcast on.
nceChannel;
use
                                                        @return
Illuminate\Broadcasting\Priva
                                 \Illuminate\Broadcasting\Chan
teChannel;
                                 nel|array
use
                                      */
Illuminate\Contracts\Broadcas
ting\ShouldBroadcast;
                                     public
                                                       function
                                broadcastOn()
use
Illuminate\Foundation\Events\
                                     {
Dispatchable;
                                         return
use
                                 PrivateChannel('review.'.$thi
Illuminate\Queue\SerializesMo
                                 s->review['userId']);
dels;
                                     }
                                 }
class
               DocumentReview
implements ShouldBroadcast
{
                                 DocumentUpload.php
                Dispatchable,
    use
InteractsWithSockets,
SerializesModels;
                                 <?php
    public $review;
                                 namespace App\Events;
    public
                     function
 construct($review)
    {
                                 use
                                 Illuminate\Broadcasting\Chann
        $this->review
                                 el;
$review;
                                 use
    }
                                 Illuminate\Broadcasting\Inter
                                 actsWithSockets;
    /**
```



```
* Get the channels the
Illuminate\Broadcasting\Prese
                                 event should broadcast on.
nceChannel;
use
                                                        @return
Illuminate\Broadcasting\Priva
                                 \Illuminate\Broadcasting\Chan
teChannel;
                                 nel|array
use
                                      */
Illuminate\Contracts\Broadcas
ting\ShouldBroadcast;
                                     public
                                                      function
                                 broadcastOn()
use
Illuminate\Foundation\Events\
                                     {
Dispatchable;
                                         return
use
                                 PrivateChannel('docupload.'.$
Illuminate\Queue\SerializesMo
                                 this->upload['userId']);
dels;
                                     }
                                 }
               DocumentUpload
class
implements ShouldBroadcast
{
                Dispatchable,
    use
InteractsWithSockets,
SerializesModels;
    public $upload;
    public
                     function
 construct($upload)
    {
        $this->upload
$upload;
    }
    /**
```

Republic of the Philippines EULOGIO "AMANG" RODRIGUEZ INSTITUTE OF SCIENCE AND TECHNOLOGY CAVITE CAMPUS Poblacion 5 General Mariano Alvarez Cavite	

Republic of the Philippines EULOGIO "AMANG" RODRIGUEZ INSTITUTE OF SCIENCE AND TECHNOLOGY CAVITE CAMPUS Poblacion 5 General Mariano Alvarez Cavite	

