



*Republic of the Philippines*  
**EULOGIO "AMANG" RODRIGUEZ**  
**INSTITUTE OF SCIENCE AND TECHNOLOGY**  
**CAVITE CAMPUS**

*Poblacion 5 General Mariano Alvarez Cavite*

**SOURCE CODE**



Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```
NotifyDue.php

<?php

namespace
App\Console\Commands;

use App\Events\DueTask;

use
App\Http\Traits\InstrumentsTr
ait;

use App\Models\Notification;
use App\Models\TaskAssign;
use Carbon\Carbon;

use
Illuminate\Console\Command;

class NotifyDue extends
Command
{
    use InstrumentsTrait;

    /**
     * The name and signature
     of the console command.
     *
     * @var string
     */
    protected $signature =
'notify:due';

    /**
     * The console command
     description.
     *
     * @var string
     */
    protected $description =
'Send Notification to user';

    /**
     * Execute the console
     command.
     *
     * @return int
     */
    public function handle()
    {
        $dateNow =
Carbon::now()->format('Y-m-
d');

        $dueTask =
TaskAssign::where('due',
$dateNow)
->whereNull('notified')
->get(['id',
'instrumentId', 'userId',
'accredId'])
->map(function
($due) {
```



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

```

        $parent =
$this->getParents($due-
>instrumentId, true, null);

        $accred =
$due->accreditation;

        unset($due-
>instrument, $due-
>accreditation);

        return $due;

    });

if(count($parent) > 0){

        $due-
//echo $dueTask;

>area =
$parent[count($parent)-1]-
>only('title');

        foreach ($dueTask as
$value) {

            $due-
$notif =
Notification::create([

                >parameter =
$parent[count($parent)-2]-
>only('title');

            }else{

                'userId'=>$value->userId,

                $due-
'actionDate'=>'due',

                >area = null;

                $due-
'details' =>

                >parameter = null;

                'You have
one due task today in
'.'.$value-
>program['abbreviation'].' /
'.'.$value->level['title'].

                $due->inst =
$due->instrument;

                ' /

                $due->program
'.'.$value->area['title'].' /

                = $accred->taggedPrograms-
'.'.$value-
>only('abbreviation');

                >parameter['title'].' /

                $due->level =
'.'.$value->inst['title']

                $accred->getLevelInstrument-
]);

                >only('title');

                $task =
TaskAssign::find($value->id);

```



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

```

    $task->notified = /**
true;
    $task->save();
    * The name and signature
    * of the console command.
    *
    * @var string
    */
    DueTask::dispatch($notif);

//Duedates::dispatch([$value-
>userId, $notif]);
    }
    }
}

ScheduledSurvey.php

<?php
namespace
App\Console\Commands;

use App\Events\UpdateAccred;
use App\Models\Accreditation;
use App\Models\User;
use Carbon\Carbon;
use
Illuminate\Console\Command;

class ScheduledSurvey extends Command
{
    $dateNow
Carbon::now() -
>timezone('Asia/Manila') -
>format('Y-m-d');
    /**
    * The console command
    * description.
    *
    * @var string
    */
    protected $description =
'Open scheduled accreditation
survey';

    /**
    * Execute the console
    * command.
    *
    * @return int
    */
    public function handle()
    {

```



Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```

Accreditation::where('date_se
lf_survey', $dateNow)
-
>whereNull('survey')
-
>whereNull('status')
-
>whereNull('selfsurvey')
-
>update(['survey' => 2]);
-
Accreditation::where('date_ac
tual_survey', $dateNow)
-
>whereNull('survey')
-
>update(['survey' => 1,
'restrict' => true]);
-
    foreach ([1, 6] as
$value) {
UpdateAccred::dispatch(collec
t([
    'auth' =>
$value,
    'self' =>
Accreditation::where('date_se
lf_survey', $dateNow)
-
>where('survey', 2)
-
>whereNull('status')
-
>whereNull('selfsurvey')
-
>get(['id',
'date_actual_survey',
'date_self_survey',
'instrumentId', 'programId',
'restrict', 'selfsurvey',
'survey'])
-
>map(function ($accred) {
return collect([
'id' => $accred->id,
'date_actual_survey' =>
$accred->date_actual_survey,
'date_self_survey' =>
$accred->date_self_survey,
'instrumentId' => $accred-
>instrumentId,
'programId' => $accred-
>programId,
'restrict' => $accred-
>restrict,
'selfsurvey' => $accred-
>selfsurvey,

```



Republic of the Philippines  
**EULOGIO "AMANG" RODRIGUEZ**  
**INSTITUTE OF SCIENCE AND TECHNOLOGY**  
**CAVITE CAMPUS**

Poblacion 5 General Mariano Alvarez Cavite

```

'survey' => $accred->survey,
'date_self_survey' =>
    $accred->date_self_survey,

'program' => $accred-
>taggedPrograms-
>only('abbreviation')['abbrev
iation'],

'title' => $accred-
>getLevelInstrument-
>only('title')['title']

    ]);

    }),

        'actual' =>
Accreditation::where('date_ac
tual_survey', $dateNow)

        -
>where('survey', 1)

        -
>whereNull('status')

        -
>get(['id',
'date_actual_survey',
'date_self_survey',
'instrumentId', 'programId',
'restrict', 'selfsurvey',
'survey'])

        -
    }),

>map(function ($accred) {

    }));

return collect([

    }

'id' => $accred->id,

    }

'date_actual_survey' =>
$accred->date actual survey,
Kernel.php

```



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

```
<?php
namespace App\Console;

use Illuminate\Console\Scheduling\Schedule;
use Illuminate\Foundation\Console\Kernel as ConsoleKernel;

class Kernel extends ConsoleKernel
{
    /**
     * Define the application's command schedule.
     *
     * @param \Illuminate\Console\Scheduling\Schedule $schedule
     * @return void
     */
    protected function schedule(Schedule $schedule)
    {
        $schedule-
        >command('survey:open') -
        >dailyAt('06:00');
    }

    /**
     * Register the commands for the application.
     *
     * @return void
     */
    protected function commands()
    {
        $this-
        >load(__DIR__.'/Commands');

        require base_path('routes/console.php');
    }
}

Attachment.php

<?php
namespace App\Events;

use Illuminate\Broadcasting\Channel;
```



Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```
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;
use Illuminate\Support\Facades\Auth;

class Attachment implements ShouldBroadcast
{
    use Dispatchable, InteractsWithSockets, SerializesModels;

    public $attachment;

    public function __construct($attachment)
```

```
{
    $this->attachment = $attachment;
}

/**
 * Get the channels the event should broadcast on.
 *
 * @return array
 */
public function broadcastOn()
{
    return new PrivateChannel('attach.'.Auth::id());
}
}

DocumentComment.php
<?php
namespace App\Events;

use Illuminate\Broadcasting\Channel;
```





Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```
use
Illuminate\Broadcasting\Inter
actsWithSockets;

use
Illuminate\Broadcasting\Prese
nceChannel;

use
Illuminate\Broadcasting\Priva
teChannel;

use
Illuminate\Contracts\Broadcas
ting\ShouldBroadcast;

use
Illuminate\Foundation\Events\
Dispatchable;

use
Illuminate\Queue\SerializesMo
dels;

class      DocumentComment
implements ShouldBroadcast
{
    use      Dispatchable,
    InteractsWithSockets,
    SerializesModels;

    public $comment;

    public      function
__construct($comment)
    {
        $this->comment      =
$comment;
    }
}
```

```
/**
 * Get the channels the
event should broadcast on.
 *
 * @return
Illuminate\Broadcasting\Chan
nel|array
 */
public      function
broadcastOn()
{
    return      new
PrivateChannel('docuComment.'
.$this->comment['userId']);
}
}

DocumentReview.php

<?php

namespace App\Events;

use
Illuminate\Broadcasting\Chann
el;

use
Illuminate\Broadcasting\Inter
actsWithSockets;
```



Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```
use Illuminate\Broadcasting\PresenceChannel;

use Illuminate\Broadcasting\PrivateChannel;

use Illuminate\Contracts\Broadcasting\ShouldBroadcast;

use Illuminate\Foundation\Events\Dispatchable;

use Illuminate\Queue\SerializesModels;

class DocumentReview implements ShouldBroadcast
{
    use Dispatchable, InteractsWithSockets, SerializesModels;

    public $review;

    public function __construct($review)
    {
        $this->review = $review;
    }

    /**
     * Get the channels the event should broadcast on.
     *
     * @return array
     */
    public function broadcastOn()
    {
        return new PrivateChannel('review.'. $this->review['userId']);
    }
}

DocumentUpload.php

<?php

namespace App\Events;

use Illuminate\Broadcasting\Channel;
use Illuminate\Broadcasting\InteractsWithSockets;
```



Republic of the Philippines  
EULOGIO "AMANG" RODRIGUEZ  
INSTITUTE OF SCIENCE AND TECHNOLOGY  
CAVITE CAMPUS

Poblacion 5 General Mariano Alvarez Cavite

```
use Illuminate\Broadcasting\PresenceChannel;

use Illuminate\Broadcasting\PrivateChannel;

use Illuminate\Contracts\Broadcasting\ShouldBroadcast;

use Illuminate\Foundation\Events\Dispatchable;

use Illuminate\Queue\SerializesModels;

class DocumentUpload implements ShouldBroadcast
{
    use Dispatchable,
        InteractsWithSockets,
        SerializesModels;

    public $upload;

    public function __construct($upload)
    {
        $this->upload = $upload;
    }

    /**
     * Get the channels the event should broadcast on.
     *
     * @return array
     */
    public function broadcastOn()
    {
        return new PrivateChannel('docupload.'.$this->upload['userId']);
    }
}
```



*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*



*Republic of the Philippines*  
**EULOGIO "AMANG" RODRIGUEZ**  
**INSTITUTE OF SCIENCE AND TECHNOLOGY**  
**CAVITE CAMPUS**

*Poblacion 5 General Mariano Alvarez Cavite*