

project: ROCTracker

Een systeem die assessoren up-to-date houdt



Gemaakt door : Mazeyar Rezaei Ghavamabadi
Datum : 1/20/2017
Opleiding : Applicatie Ontwikkelaar
Dank aan : Laravel

Meer informatie:
• Deze opdracht is uitgevoerd in een periode van 12 weken.
• Deze opdracht is een eindopdracht van de opleiding.

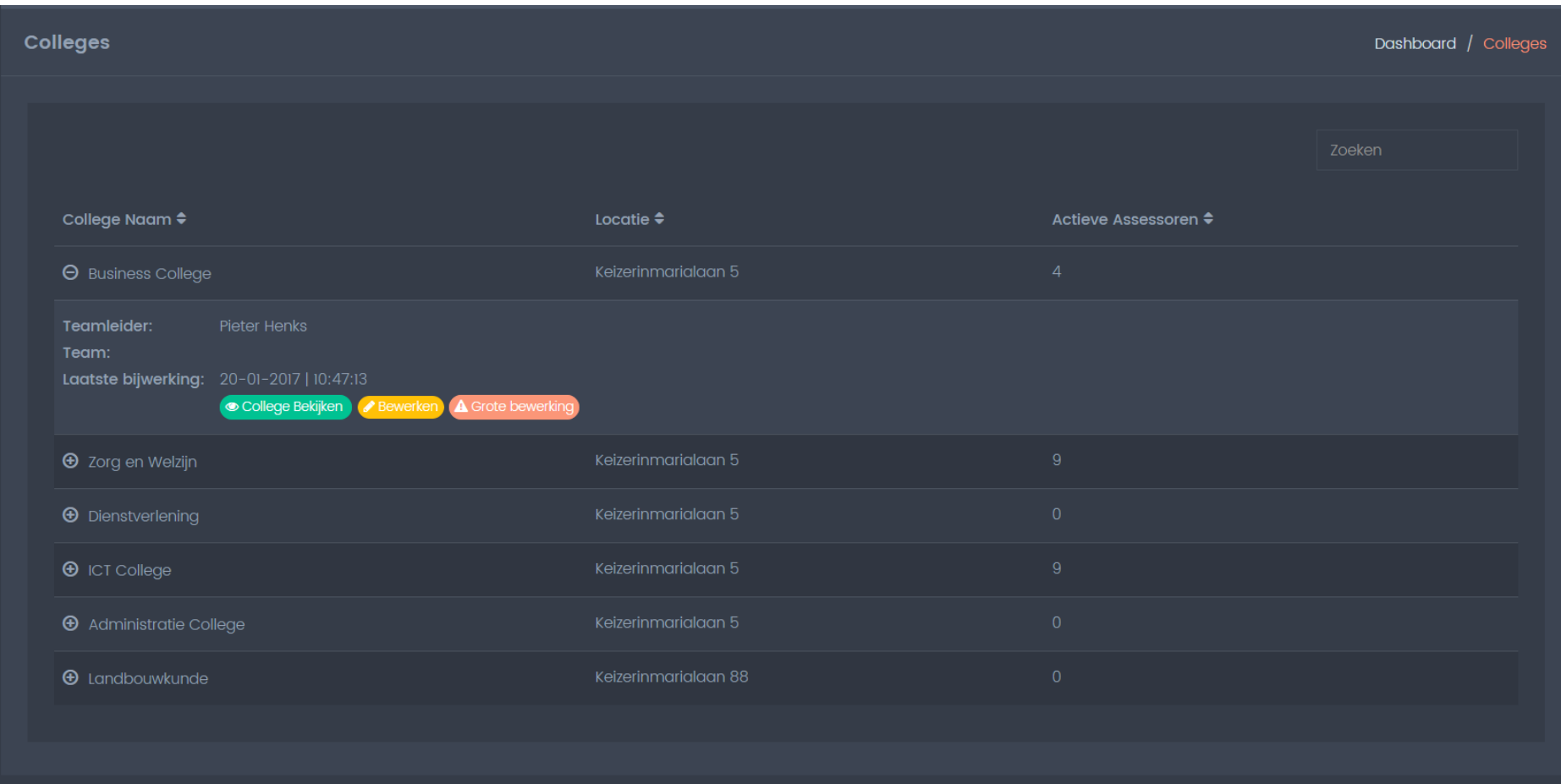
Doelen :
Een systeem ontwikkelen die de volgende taken uitvoert.
• Invoeren en bijhouden van colleges in het ROC Ter AA.
• Invoeren en bijhouden van Teamleiders in het ROC Ter AA .
• Overzicht krijgen van alle assessoren in het ROC Ter AA.
• Assessor data kunnen wijzigen op basis van de examen gegevens.

Gebruikte materialen en gemaakte kosten:
• MAMP PRO
• PHP Storm 2016.3
• Github
• Git Bash
• Laravel Framework

Projectbeschrijving: “*ROCTracker*” is een systeem die het papier werk van assessoren zal automatiseren. Het systeem zal ervoor zorgen dat de volgende benoemde informatie in een systeem zal worden weergegeven en bijgehouden zal worden.

- Colleges
- Assessoren
- Teamleiders

Het systeem zal alleen toegankelijk zijn voor administrators, deze gebruikers kunnen worden toegevoegd door andere administrators. Elke **grote** wijziging die door deze gebruikers plaatsvind zal worden opgeslagen onder de gebruiker zijn naam. In het systeem kunnen ook **kleine** wijzigingen worden gemaakt deze is bedoeld voor correcties en veranderingen die niet opmerkelijk zijn.



Op deze afbeelding word de college pagina weergegeven. Hier is een overzicht te zien over alle colleges in het ROC Ter AA. Deze layout word gebruikt voor alle colleges binnen het systeem.

Wanneer de gebruiker er voor kiest om een grote wijziging te maken in het college word dit bijgehouden in het systeem.

Wanneer er een grote wijziging plaatsvind voor een college moet er ook een wijziging

De wijziging die word uitgevoerd zal door een switch heen gaan. Deze switch zal verder kijken wat er moet gebeuren met de wijziging, doormiddel van de volgende stappen.

1. Eerst zal de wijziging moeten worden gecheckt op de inputs.
2. Dan zal er moeten worden gecheckt of alle inputs correct zijn ingevuld
3. Wanneer alle checks zijn voltooid zal de er gekeken worden wat er moet gebeuren met de data, die aan deze wijziging is gelinkte.
4. Dan zal de wijziging worden doorgevoerd en zal er in de log een node worden gemaakt met wat de

```
use Illuminate\Http\Request;

class CollegeController extends Controller
{
    public function postChangeCollege (Request $request, $id) {
        switch ($request->assessor_option) {
            case 'selection':
                $validator = Validator::make($request->all(), [
                    // Validation rules

                ]);

                if ($validator->fails()) {
                    // Validation failed, return response
                }

                $college = College::find($id);
                $name = $college->name;
                $location = $college->location;

                $name = ($name != $request->name) ? true : false;
                $location = ($location != $request->location) ? true : false;
                $change_both = ($name==true && $location==true) ? true : false;
                if ($name == false && $location == false) {
                    // No changes, return response
                }

                $message = null;
                $old_college['name'] = $college->name;
                $old_college['location'] = $college->location;
                if ($change_both==true) {
                    $message = 'College naam en locatie is veranderd';
                } elseif ($name==true) {
                    $message = 'College naam is veranderd';
                } elseif ($location==true) {
                    $message = 'College locatie is veranderd';
                }
                $college->save();
                Log::info('CollegeLog($id, $message)');

                $assigned_assessors = Assessors::where('fk_college', '=', $id)->get();
                if (!empty($assigned_assessors)) {
                    foreach ($assigned_assessors as $assessor) {
                        if ($change_both == true) {
                            $message = 'College naam van <strong>' . $old_college['name'] . '</strong> naar <strong>' . $request->name . '</strong>' . ' en Locatie van <strong>' . $old_college['location'] . '</strong> naar <strong>' . $request->location . '</strong>';
                            Log::info('AssessorLog($assessor->id, $message)');
                        } elseif ($name == true) {
                            $message = 'College Locatie van <strong>' . $old_college['location'] . '</strong> naar <strong>' . $request->location . '</strong>';
                            Log::info('AssessorLog($assessor->id, $message)');
                        }
                    }
                }

                return redirect()->route('change_college_assessors', array($id, $request->name, $request->location));
            }
        }
    }
}
```