

**Bachelor in de Grafische en Digitale Media
Multimediaproductie New Media Development**

CONNECTED

OLIVIER DECOCK

2019-2020

CMS

BRIEFING	4
ANALYSE	5
PLANNING	6
DESIGN	7
DEVELOP	10
HANDLEIDING	12
DEPLOYMENT GUIDE	14

BRIEFING

Wat wil de opdrachtgever, de klant of het bedrijf?

De klant heeft weinig kaas gegeten van computers, maar beseft wel dat zijn manier van werken een stuk gemakkelijker zou kunnen gemaakt worden door het maken van een webapplicatie. Hij had zelf het volgende idee:

Een all-in-one webapplicatie waarbij hij de uren niet meer zelf hoeft in te vullen, maar reeds gebeurd is door het personeel / onderaannemers die hij uitstuurt.

Wie behoort tot het doelpubliek?

Alle werknemers en onderaannemers die werken voor bedrijf X alsook de werkgever en de klanten van bedrijf X.

Welke informatie is beschikbaar?

Wat we tijdens de briefing hebben te horen gekregen was hoe de klant nu te werk gaat en welke problemen dat met zich meebrengt. Er is een oplossing bedacht door de klant met de features die hij graag had gezien. Het is aan ons om dit in code om te zetten.

Wat is de timing?

De finale versie van het volledige project moet klaar zijn tegen zaterdag 22 augustus 2020 om 20u, het dossier moet worden ingediend op 20 augustus om 20u.

ANALYSE

Wat zullen we de klant aanbieden/wat gaan we doen

We zullen de klant aan de ene kant een all-in-one webapplicatie aanbieden waarbij hij de uren niet meer zelf hoeft in te vullen maar dit reeds gebeurd is door het personeel/onderaannemer die hij heeft uigestuurd.

Als 2e bieden we hem ook een backend versie aan waarbij hij zijn volledig klantenbestand alsook al zijn personeel kan toevoegen, bewerken of verwijderen. Alle berekeningen van facturen en mailing gebeurd vanaf nu automatisch op een daarvoor ingesteld tijdstip.

De manager kan op een gebruiksvriendelijke manier alles opvolgen en maakt zo komaf met de grote waarboel van papier en mails.

Functionele eisen

De backend moet in staat zijn om klanten toe te voegen, bewerken en verwijderen, dit geldt ook voor het toevoegen/bewerken/verwijderen van werknemers. Daarnaast moet de backend ook instaat zijn voor het aanmaken van facturen alsook het versturen van werkbonnen naar de klant.

De all-in-one webapplicatie moet kunnen communiceren met de backend om zo de uren van de werknemers/onderaannemers te kunnen loggen.

Er mag enkel van de functionaliteiten van de applicatie kunnen worden gebruik gemaakt als de user is ingelogd. Het beheren van velden in de backend kan enkel gebeuren door de super admin.

Niet functionele eisen

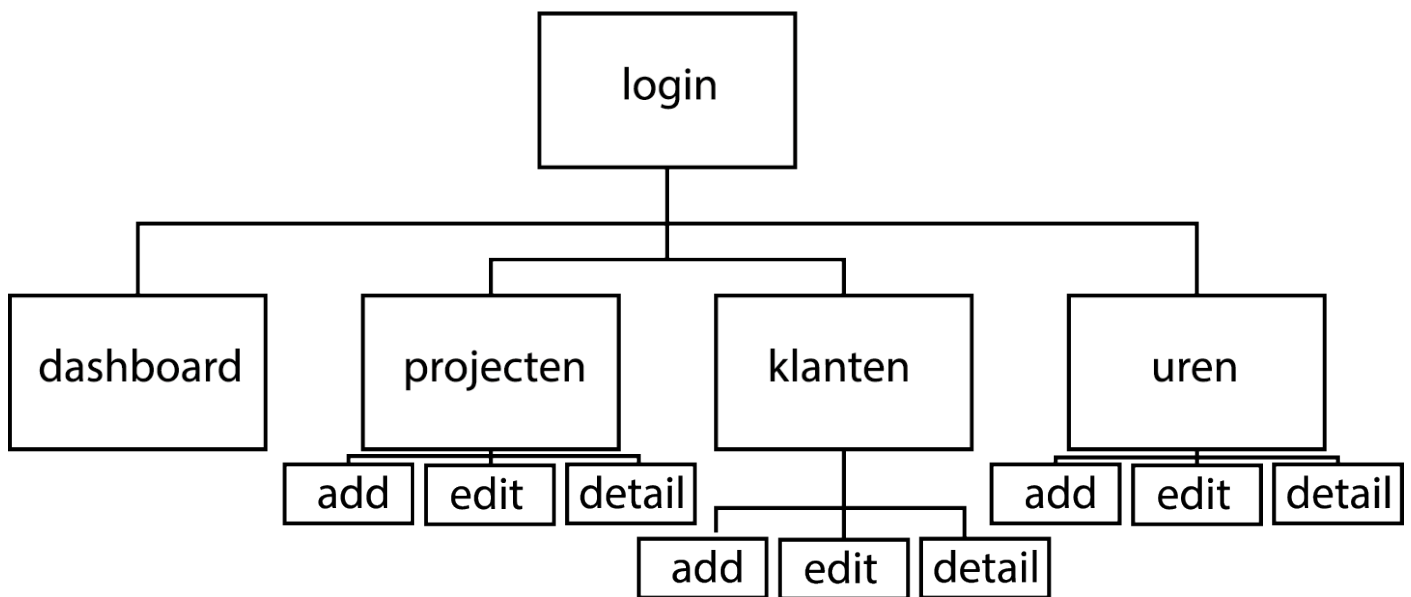
De gebruiksvriendelijkheid staat op kop, de laadtijd van de pagina's moet tot een minimum beperkt worden. Zoals eerder vermeld zijn beide projecten enkel toegankelijk als je daar de toestemming voor hebt, de beveiliging moet dus in orde zijn.

PLANNING

Vermits er tot 12 augustus aan de bachelorproef mag worden gewerkt zal ik 13 augustus beginnen aan dit project. Tegen 16 augustus moet de backend volledig klaar zijn vermits dit gelijkaardig is aan mijn stage verwacht ik ook dat dit aan een vlot tempo zal gaan. Voor het maken van het dossier en de assets plan ik 1 dag, de overige dagen zijn voor het maken van de react applicatie.

DESIGN

Sitemap Website



Wireflow Mobile Web App



Colors



#6D767E
table titles, tekst



#313A46
navigatie achtergrond



#F1F3FA
body achtergrond

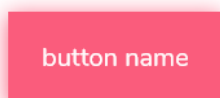


#FA5C7C
buttons, links



#4E5BF2
navigatie belangrijkste element

Buttons

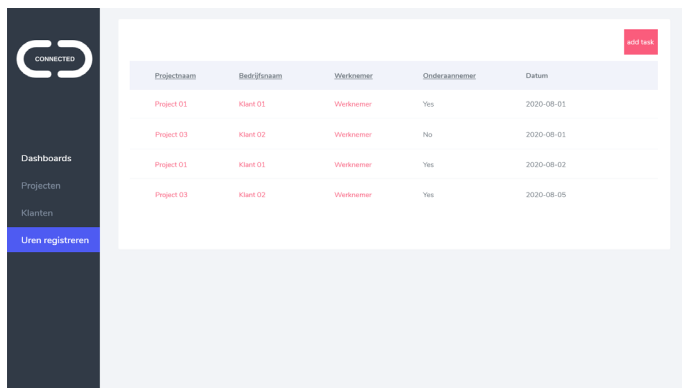


Fonts

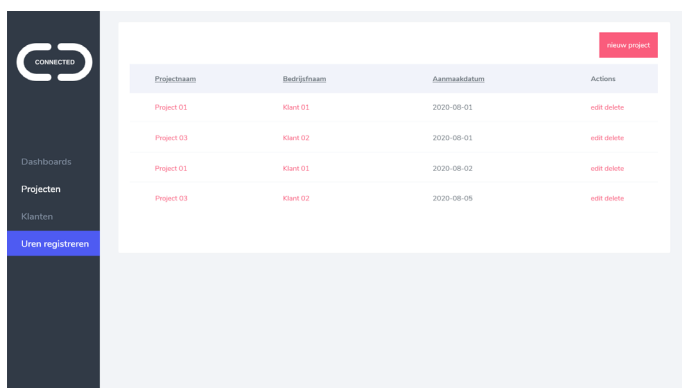
Nunito Bold
titles

Nunito Regular
lopende tekst, navigatie

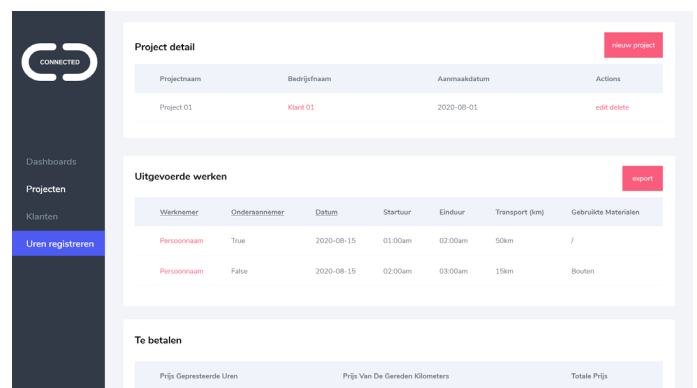
Visual Designs



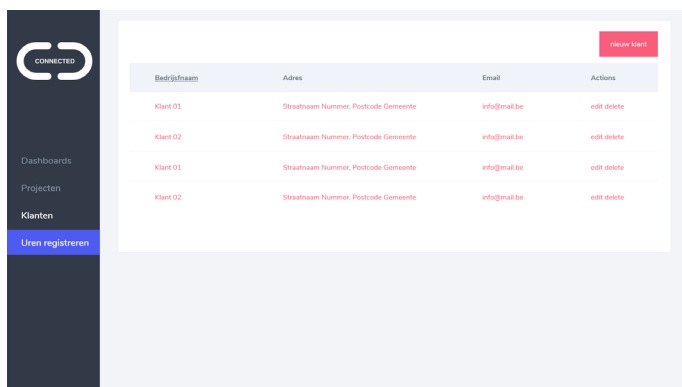
Dashboard



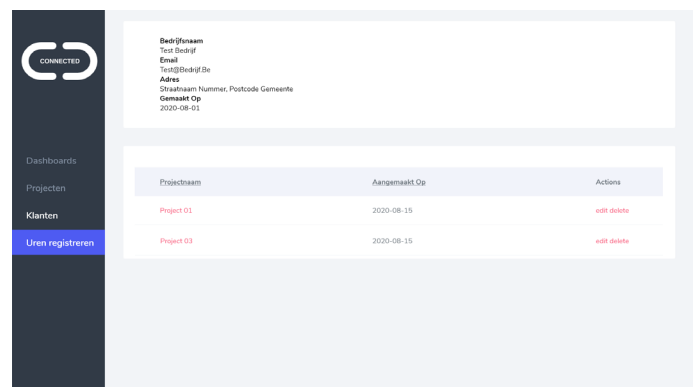
Projecten



Projecten detail



Klanten



Klanten detail

DEVELOP

Prijsberekening

```
$price = Views::getView( id: 'price');
$price->execute();
$price_result = $price->result;
$transportPrice = $price_result[0]->_entity->get('field_transportkosten')->getString();

$hours = Views::getView( id: 'project_tasks');
$hours->execute();
$hours_result = $hours->result;

$totalWorkTransport = 0;
$totalWorkPrice = 0;
$totalAmount = 0;

foreach($hours_result as $hour){
    $entity = $hour->_entity;
    $start = $entity->get('field_startuur')->getString();
    $eind = $entity->get('field_einduur')->getString();

    $salaryPrice = $price_result[0]->_entity->get('field_uurtarief')->getString();
    $transportPrice = $price_result[0]->_entity->get('field_transportkosten')->getString();

    $pauze = $entity->get('field_pauze')->getString();
    if($pauze !== null)
        $arr = explode( delimiter: ",", $pauze, limit: 2);
        $startPauze = $arr[0];
        $stopPauze = $arr[1];
        $pauzeDuur = $stopPauze-$startPauze;

    $totalWorkHours = ($eind - $start - $pauzeDuur) / 3600;
    $pauzeDuur = 0;
    $transportHours = $entity->get('field_transport')->getString();

    $freelancer = $entity->get('field_freelancer')->getString();
    $check = $freelancer == 1;
    if ($check == true){
        $transportPrice = $entity->get('field_transportkost')->getString();
        $salaryPrice = $entity->get('field_prijs')->getString(); |
    }

    $totalWorkHoursPrice = $totalWorkHours * $salaryPrice;
    $totalWorkTransportPrice = $transportHours * $transportPrice;
    $totalWorkTransport = $totalWorkTransport + $totalWorkTransportPrice;
    $totalWorkPrice = $totalWorkPrice + $totalWorkHoursPrice;
}

$totalAmount = $totalAmount + $totalWorkPrice + $totalWorkTransport;
$totalAmount = round($totalAmount, precision: 2);
$totalWorkTransport = round($totalWorkTransport, precision: 2);
$totalWorkPrice = round($totalWorkPrice, precision: 2);

return [
    'work' => $totalWorkPrice,
    'transport' => $totalWorkTransport,
    'total' => $totalAmount,
];
```

view ophalen van prijs en uren, daaruit de juiste velden halen en berekenen.

```

butes.addClass(classes) }>>
content %}
content_attributes.addClass('content') }>>
class="views-element-container contextual-region block block-views block-views-blockproject-tasks-block-1" id="block-views-block-project-tasks-block-1">
Te betalen</h2>
class="contextual-region view view-project-tasks view-id-project_tasks view-display-id-block_1 js-view-dom-id-320720f234cbc2c278db7f9c982157c668019efb49de9cf86e65b4d8002ee771">
div class="view-content">
<table class="views-table views-view-table cols=8">
<thead>
<tr>
<th class="views-field" scope="col">Prijs gepresteerde uren</th>
<th class="views-field" scope="col">Prijs van de gereden kilometers</th>
<th class="views-field" scope="col">Totale prijs</th>
</tr>
</thead>
<tbody>
<tr class="views-field">
<td>{{ content.work }}</td>
<td>{{ content.transport }}</td>
<td>{{ content.total }}</td>
</tr>
</tbody>
</table>
div>
v>
k %}

```

De berekeningen worden gereturned naar bovenstaande custom block waar ze dan worden uitgelezen.

Te betalen

Prijs gepresteerde uren	Prijs van de gereden kilometers	Totale prijs
€390	€512.5	€902.5

Bovenstaande schermafbeeldingen is het resultaat van de 2 voorgaande codesnippets.

HANDLEIDING

Drupal

1. Surf naar het domein, indien u nog niet bent ingelogd krijgt u een login formulier te zien, bij het correct invullen en submitten van dit formulier wordt u naar het dashboard gebracht.
- 2.
3. Op het dashboard krijgt u een tabel te zien met daarin de laatst toegevoegde taken van uw werknemers, als admin kan u ook nog zelf taken toevoegen door op de rode knop rechtsboven 'nieuwe taak' te klikken. Om de details te zien van de klant of van het project van de betreffende taak klikt u gewoon op de naam van wat u wenst te zien.
- 4.
5. Een 2e pagina waar u toegang tot heeft is de projecten pagina, hierop ziet u welke projecten er voor welke klanten lopende zijn. Deze projecten hebben een detailpagina, deze opent u door op de naam van het project te klikken. Als u een nieuw project toe te voegen dan klikt u op de knop 'nieuw project' en dient u het formulier correct in te vullen en te submitten. Iedere taak kan ook bewerkt en verwijderd worden door op de knoppen 'bewerk' of 'verwijder' te drukken van de betreffende taak.
- 6.
7. De 3e pagina die u kan openen is de klantenpagina, hierop ziet u een tabel met daarin uw klanten. Per pagina staan er maximum 50 klanten, u kunt navigeren tussen de pagina's door onderaan gebruik te maken van de pijltjes. Iedere klant kan bewerkt of verwijderd worden en heeft net zoals bij projecten een detailpagina, deze kunt u openen door te klikken op de naam van de klant. Een nieuwe klant kan worden toegevoegd door op de knop 'nieuwe klant' te drukken. Hierbij krijgt u een formulier te zien, vul dit correct in en de klant wordt voor u aangemaakt.
- 8.
9. De laatste pagina waar u toegang tot heeft via de navigatie is 'uren registreren', indien u hierop klikt wordt u naar een formulier gebracht waarbij u een taak kunt registreren. Bij het correct invullen van dit formulier zal deze taak bij het juiste project worden toegevoegd.

All-in-one Web App

Als u een werknemer/onderaannemer bent van bedrijf x dan zal iemand binnen het bedrijf een account voor u moeten aanmaken en de logingegevens aan u doorgeven.

Indien u de gegevens heeft kan u surfen naar het domein waarbij u, indien u nog niet bent ingelogd, op het login scherm komt. Hier voert u de gebruikersnaam en het wachtwoord in die u heeft doorgekregen.

Indien u bent ingelogd krijgt u een formulier te zien waarbij u de mogelijkheid heeft om een nieuwe taak toe te voegen. Vul de velden correct in, hierna duwt u op de button onderaan en de taak wordt gelogd.

Als u zich wenst uit te loggen kan dit door op de button 'logout' rechts bovenaan te klikken.

DEPLOYMENT GUIDE

Drupal

1. Download, en unzip indien nodig, de bronbestanden van volgende link https://github.com/Olivier070699/cms_connected
2. Open MAMP
3. Ga in MAMP naar preference/Web Server
4. Navigeer bij 'document root:' naar het path van de locatie waar de gedownloadde versie van de repo zich bevindt ga hierbij nog 1 stap verder en open 'backend_drupal/web'.
5. Als u nu in uw browser naar localhost gaat zal zien dat de site niet kan openen
6. Ga terug naar de terminal en navigeer naar het drupal bestand
7. voer het commando 'composer install' uit
8. Daarna zou je moeten kunnen op de browser naar localhost kunnen surfen en zou deze moeten werken.
9. Je zal een wizard moeten invullen. Daar zal je alle gegevens moeten invullen. Zorg dat de benaming die je gegeven hebt aan jouw database hetzelfde is als degene die je invult in de wizard

React

10. Laat de MAMP-server aanstaan
11. Open de terminal en navigeer naar de app.
12. Zorg dat npm of yarn geïnstalleerd is en type in de terminal: npm install of yarn install
13. Als dat gedaan is geef je in: npm start of yarn start