



LOGISTIC MANAGEMENT PLATFORM

Arno Willaert & Jonas Van Noten

LMP



Voorwoord

Beste,

Wij zijn Arno Willaert en Jonas Van Noten en wij zijn studenten aan Thomas More campus De Nayer en wij hebben voor project 2 een platform gemaakt voor communicatie tussen klant en personeel van een transportbedrijf. We zouden ook graag mevrouw De Greef bedanken voor haar steun tijdens dit project. Verder wens ik uw nog veel plezier met het lezen van ons verslag.

Informatie auteurs

Arno Willaert
arno.willaert@student.thomasmore.be
arno.willaert@gmail.com

Jonas Van Noten
jonas.vannoten@student.thomasmore.be
jonasvannoten@gmail.com

Samenvatting

Bij het zoeken naar een geschikt project hebben we gekeken naar onze nabije omgeving. De vader van Arno Willaert is de CEO van het middelgrote transportbedrijf Willaert-Van Boom. Daar kampen ze al geruime tijd met een probleem op de planning; transportopdrachten komen binnen via verschillende kanalen (e-mail, telefoon, fax) en in een heel spectrum aan formaten (excel, plaintext in mail, etc.). Dit zorgt ervoor dat het organiseren en het plannen van vrachten heel onoverzichtelijk wordt.

Wij zijn samen op zoek gegaan naar een oplossing voor dit probleem en uiteindelijk is dit project er uit gegroeid. Een eenduidige webinterface moet een helder communicatiekanaal voorzien tussen de klant en de planningsafdeling.

We maakten een multi-user webinterface waar beide partijen kunnen inloggen, ondersteund door een databank.

De klant is in staat om locaties voor het laden en lossen van vrachten op te slaan, net zoals de goederen die hij wenst te vervoeren. De opgeslagen gegevens kunnen vervolgens in een formulier ingeladen worden en voorzien worden van de nodige extra informatie, om zo een transportaanvraag in te dienen.

De planningsafdeling kan op zijn interface alle aanvragen bekijken en sorteren, en kan de ingediende aanvragen goedkeuren of afkeuren. Het is ook mogelijk voor de planning om details omtrent het transport te wijzigen.

Wanneer een opdracht succesvol is afgerond, kan de administratie de opdracht aanduiden als zijnde voltooid. De klant heeft zo een overzicht over welke opdrachten er nog zijn richting uit komen, en welke al zijn voltooid.

Inhoudstabel

Voorwoord	1
Informatie auteurs	2
Samenvatting.....	2
Taakverdeling per groepslid	4
Inhoud	5
1 Databank	5
2 Github	7
3 Gebruikers	7
4 Features	9
4.1 Mijn account.....	9
4.2 Locatie Template & Product Template.....	9
4.3 Transportopdracht aanmaken.....	11
4.4 Opdrachten.....	12
4.5 Geschiedenis.....	12
4.6 Order request	13
4.7 Admin page.....	13
5 Plugins	14
5.1 Tabellen in datatables	14
5.2 Datetimepicker plugin	14
6 Automatische vernieuwing pagina	15
7 Hashen wachtwoorden	15
8 Https encryptie op de web server.	16
9 Verificatie via mail	17
Conclusie	18
Bijlage	19

Taakverdeling per groepslid

Arno heeft het volgende gedaan:

- Locaties
- Producten
- Maak transportopdracht
- ER-diagram
- Databank in SQL
- Order request
- Datatables
- Datetime picker
- Poster
- Filmpje
- Debuggen van de code
- Javascript validatie
- Aanmaken logo
- Feedback van potentiële gebruiker

Jonas heeft het volgende gedaan:

- Mijn account
- Geschiedenis
- Opdrachten
- Automatische vernieuwing
- Mail verificatie
- Wachtwoorden hash
- Https protocol toegevoegd
- Admin page
- Login & logout
- Locaties en producten in laden in opdracht en concepten via ajax
- Debuggen van de code
- Aanmaken logo
- Security testen

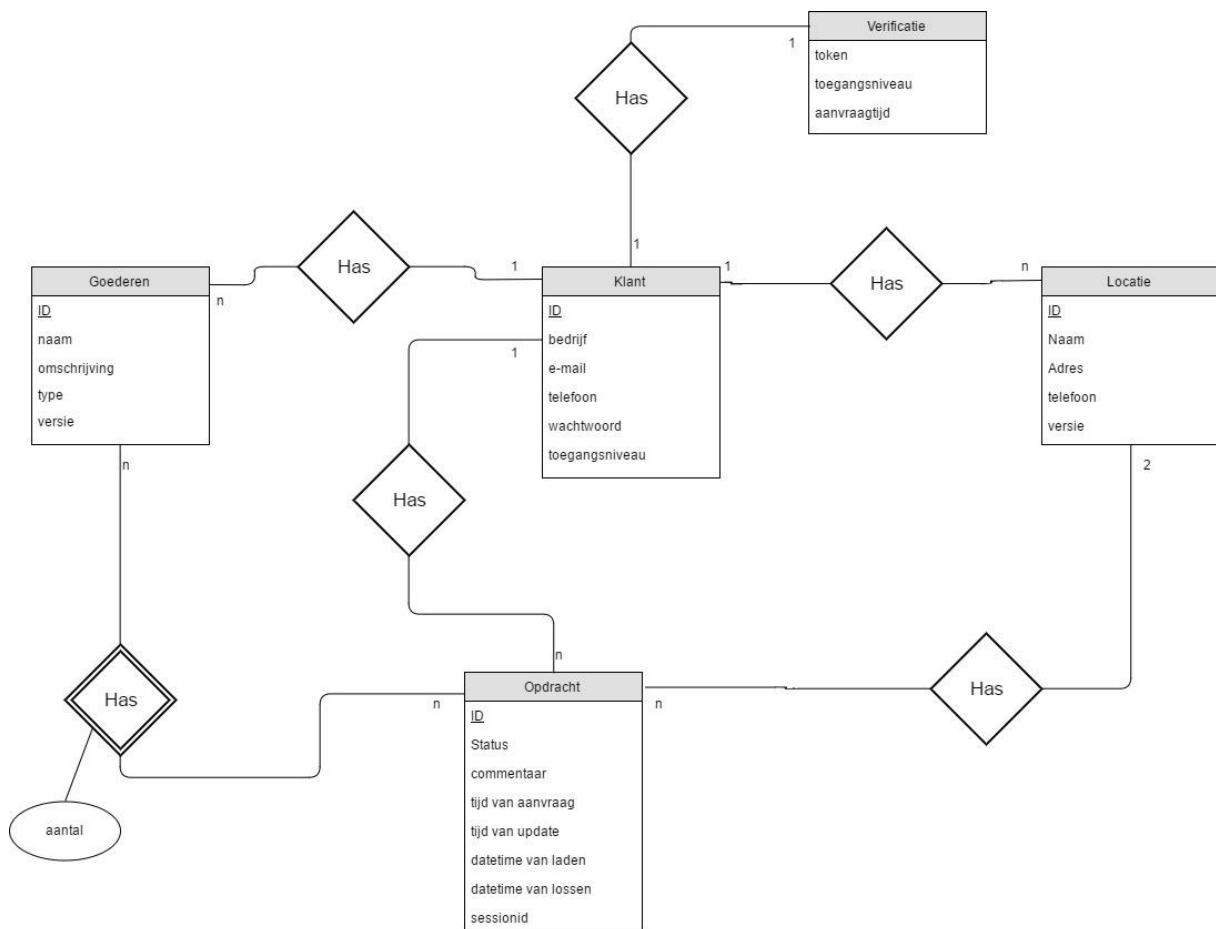
Inhoud

1 Databank

In dit onderdeel gaan we de structuur van de databank bespreken, zowel het ERD als de sql code. Ik ga niet overal in detail treden, maar slechts de minder voor de hand liggende relaties, of datatypes bespreken.

Om plaats te besparen ga ik verwijzen naar de create_table.sql in de bijlagen.

Entity Relationship Diagram.



Hierboven ziet u het ER diagram van onze databank, met centraal de tabel klant. De naam is een beetje een misnomer, want in deze tabel worden de gegevens bijgehouden over alle gebruikersaccounts, dus niet enkel klanten van het transportbedrijf. De waarde van toegangsniveau is een getal van 0 tot en met 4. Deze waarden geven aan welke bevoegdheden een gebruiker heeft. 0 staat voor inactief/gebanned/verwijderd, 1 staat voor klant, 2 voor personeel, 3 voor admin, 4 voor superuser.

De rede waarom we verwijderde klanten niet regelrecht verwijderen uit de databank is omdat wanneer we dit doen, we problemen krijgen met de foreign keys uit andere tabellen die verwijzen naar een klant id. De transporteur zou dan ook niet meer in zijn geschiedenis de opdrachten van de verwijderde user kunnen opvragen. Om dit te vermijden zetten we de toegang van de gebruiker op 0. Een gebruiker met toegang 0 kan overigens niet meer inloggen.

Een admin heeft in deze versie enkel de rechten om gebruikers aan te maken, te beheren en te verwijderen.

Bij goederen wil ik de attributen type en versie toelichten.

Het type goederen wijst op de verpakking van de goederen. Op deze moment zijn er slechts 2 opties, doos en (euro)pallet. Dit lijkt vrij gelimiteerd, maar dit is afgestemd op de noden van onze opdrachtgever.

Versie wordt relevant wanneer de klant een product gaat aanpassen. Wanneer een klant een product aanpast gebeurt het volgende; er word een nieuw record aangemaakt met een nieuwe id, met de waarde die de klant heeft opgegeven en bij versie waarde 1, en bij de oude record word de versie op 0 gezet, om aan te duiden dat deze niet meer relevant is.

Wanneer de klant een product wil inladen worden enkel goederen weergegeven met versie waarde 1. Maar, en dit is waarom we de oude bijhouden, wanneer de klant of transportbedrijf een oude opdracht inkijken, wordt nog steeds het oude product weergegeven, aangezien de goederen via hun ID verbonden zijn met de opdracht.

Dit is analoog met het attribuut versie in de tabel locatie.

De goederen zijn met een many-to-many relatie verbonden met opdrachten, dit heeft tot gevolg dat er een extra table nodig is om deze succesvol te verbinden. Deze table heet opdracht_has_goederen en vindt u in de bijlage in onze create_table.sql. Deze linkt de id's van goederen samen met een aantal aan de id's van opdrachten.

De relatie tussen opdrachten en locaties is ook wel opmerkelijk te noemen, hier gebruiken we een 2-to-many relatie, die toch wat ongebruikelijk is, maar in deze context werkt ze perfect. De 2-to-many wordt hier gebruikt omdat er in elke opdracht nood is aan 2 locaties: een voor het lossen en een voor het laden. Deze worden vertegenwoordigd in de sql file met los_id en laad_id.

Verder in de tabel opdracht ziet u ook session id staan. Dit was nodig omdat wanneer de opdracht aangemaakt wordt, er niet meteen goederen aan gekoppeld worden. Omdat we het id van de net gecreëerde record niet kennen, stoppen we er de session id bij als unieke key.

Wanneer het zo ver komt dat er producten bij de opdracht worden toegevoegd, word de opdracht ID geselecteerd waarvan de session id overeenkomt met dat van de gebruiker en die het recentst is aangemaakt.

2 Github

Een uitdaging die ontstaat door met meerdere mensen aan een software project te werken is versie-controle. Wanneer je met beide aan het zelfde stuk code schrijft, kan het geregeld lastig worden overzicht te behouden wat er verandert is, en waarom wanneer je de bestanden weer wilt samenvoegen. Om dit op te lossen besloten we gebruik te maken van GitHub, een web-based version control repository.

Dit laat ons toe om elk een aparte branch (een vertakking) te maken van ons project, waar we elk zonder zorgen aan onze code kunnen werken. Wanneer het tijd is om de code samen te voegen kunnen we gebruik maken van de merge functie in GitHub, dat duidelijk weergeeft wat de verschillen tussen onze versies en waar mogelijk zelfs automatisch de code samenvoegt.

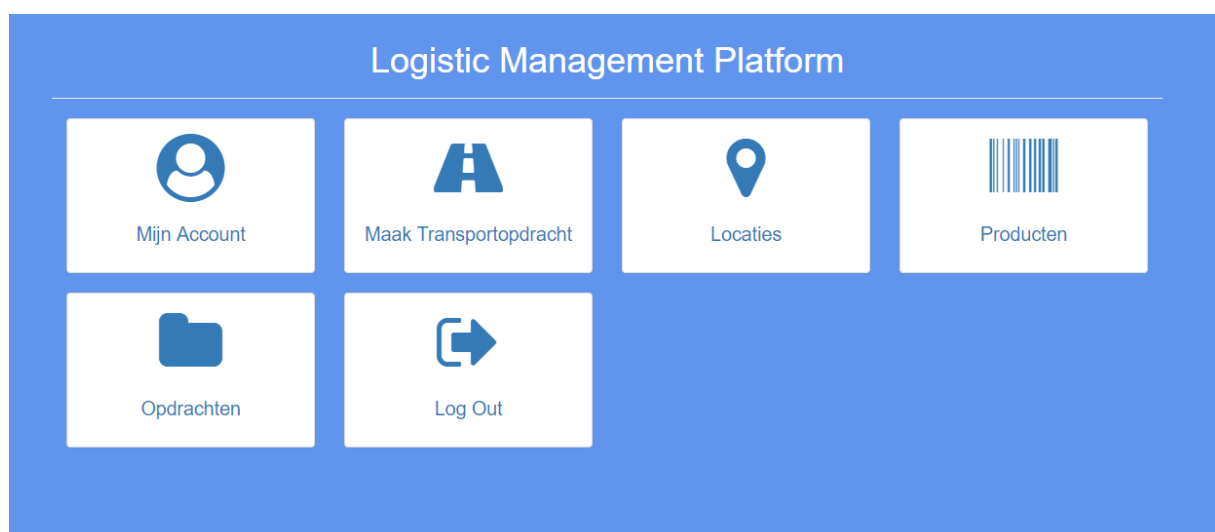
We hebben helemaal in het begin wel wat problemen gehad met GitHub, waar merges incorrect werden uitgevoerd, maar dat was volledig te wijten aan het gebrek aan ervaring met GitHub.

GitHub lost ook een ander probleem op. Als je code schrijft en aanpast gebeurt het wel vaker dat wanneer je een werkend stuk code aanpast, het stopt met werken zoals het hoort. En wat je ook probeert, je kunt het maar niet terug aan de draad krijgen. Dan is het hopen dat je een aparte kopie hebt met de werkende code, of je bent de sigaar. Met GitHub heb je die miserie niet voor. Het is meestal eenvoudig om uw branch enkele versies terug te draaien naar de werkende code. Dit zijn weer heel wat kopzorgen uitgespaard.

3 Gebruikers

Op ons platform hebben we gekozen om gebruikers verschillende toegangen te geven of ontkennen aan de features van het platform. We hebben de gebruikers opgedeeld in volgende categorieën : klant , administratie, beheerder, superuser, non-actief. Als de gebruiker op non-actief staat wilt dit zeggen dat deze niet kan inloggen. Op de home interface zijn verschillende features te zien en deze verschillen dus voor ieder van deze categorieën. (zie foto's hieronder)

De klanten kunnen locaties en producten aanmaken die ze dan later in een opdracht kunnen laden. Ze kunnen dan via opdrachten deze bezien en allerlei gepaste zaken op toepassen.



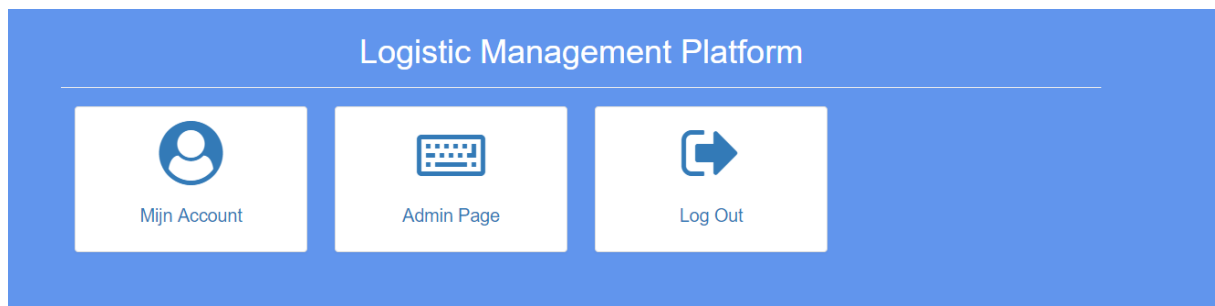
Afbeelding gebruiker 'klant'

De administratie krijgt dan de mogelijkheid om deze opdrachten te behandelen en ook krijgen ze toegang tot alle opdrachten die in het archief zitten.



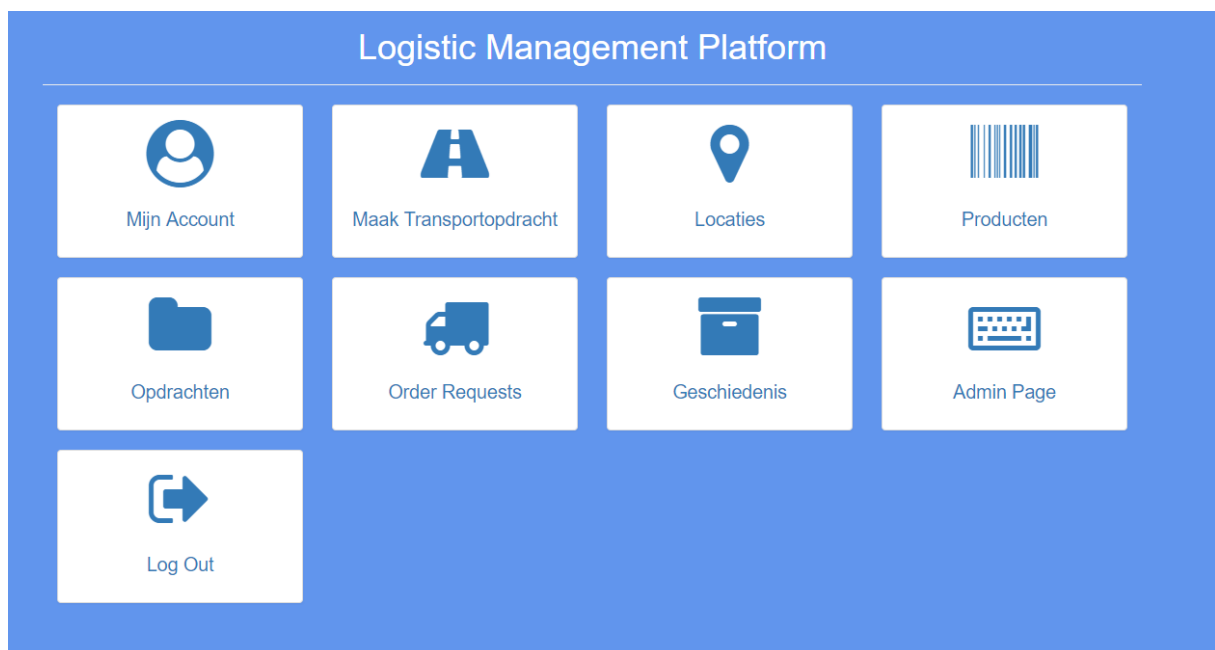
Afbeelding gebruiker 'administratie'

De beheerder kan accounts registreren en wijzigen of verwijderen.



Afbeelding gebruiker 'beheerder'

De superuser heeft toegang tot alles en kan dus zelf opdrachten aanmaken, goedkeuren en wijzigen. Vervolgens kan deze ook nog in de admin page alles wat de beheerder kan. Een superuser staat niet in de tabel met gebruikers en kan dus niet verwijderd worden door een beheerder. Een superuser kan wel zijn eigen verwijderen.

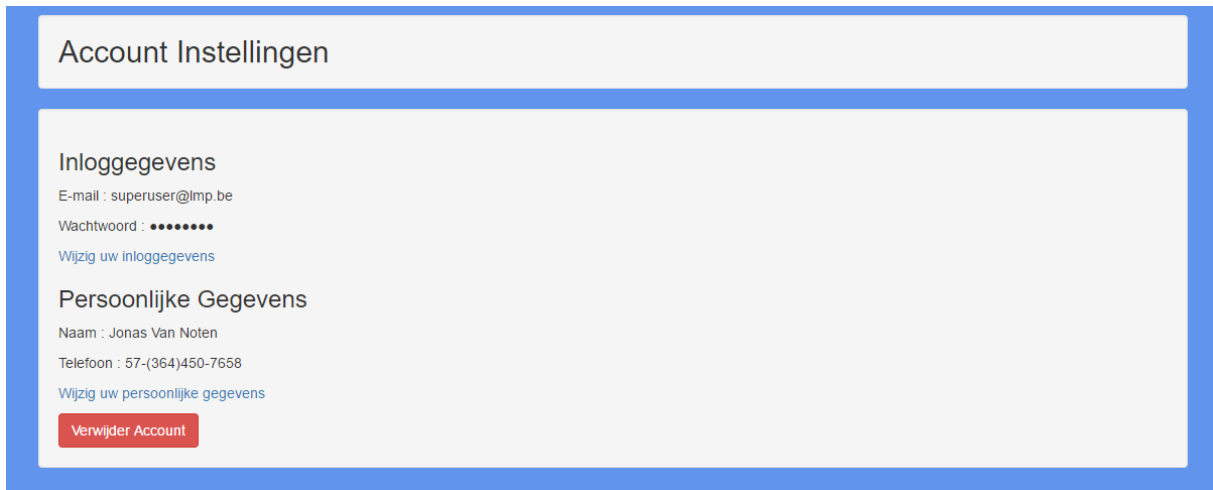


Afbeelding gebruiker 'superuser'

4 Features

4.1 Mijn account

Hier kan de gebruiker belangrijke gegevens aanpassen zoals de E-mail, naam, telefoon. Maar ook het wachtwoord kan hier gewijzigd worden. Als de gebruiker zijn account wilt verwijderen kan dat ook hier.



Account Instellingen

Inloggegevens

E-mail : superuser@imp.be

Wachtwoord : ••••••••

[Wijzig uw inloggegevens](#)

Persoonlijke Gegevens

Naam : Jonas Van Noten

Telefoon : 57-(364)450-7658

[Wijzig uw persoonlijke gegevens](#)

[Verwijder Account](#)

Afbeelding 'mijn account'

4.2 Locatie Template & Product Template

Hier kan de gebruiker aan template aanmaken of wijzigen of verwijderen. Deze gegevens worden bijgehouden in de databank. In de databank kunnen locaties en producten ook als foreign key in de opdrachten steken en dus mogen we geen locaties of producten verwijderen die al gebruikt zijn. Om dit op te lossen hebben we een extra kolom in de databank toegevoegd zodat er onderscheid kan gemaakt worden tussen de oude versies en de huidige versie. Als de locatie of product nog niet gebruikt is kan deze wel normaal aangepast worden of verwijderd.

Locatie Template

Naam

Straat

Nummer

Stad

Postcode

Land

Telefoon

Show **10** entries

Search:

Naam	Straat	Nr	Stad	Postcode	Land	Telefoon	Edit	Del
Baker Jan	Stationstraat	67	Oostende	6769	BE	+3265465499		
Haven Rotterdam	Havenweg	9	Rotterdam	5790	NL	+3176543567		
Slager Bart	Slagerstraat	7	Leuven	3000	BE	+3284654654		

Showing 1 to 3 of 3 entries

Previous
1
Next

Afbeelding van locatie template

Product Template

Naam

Type product

Omschrijving of opmerkingen

Show **10** entries

Search:

Naam	Type	Omschrijving	Edit	Del
Banaan	doos	Doos met banaanen		
Cornflakes	palet	Een palet met lekkere ontbijtgranen.		
Honden eten	doos	Hondensnoepjes		
Ijsjes	palet	Waterijsjes, dus vervoeren in een gekoelde omgeving.		

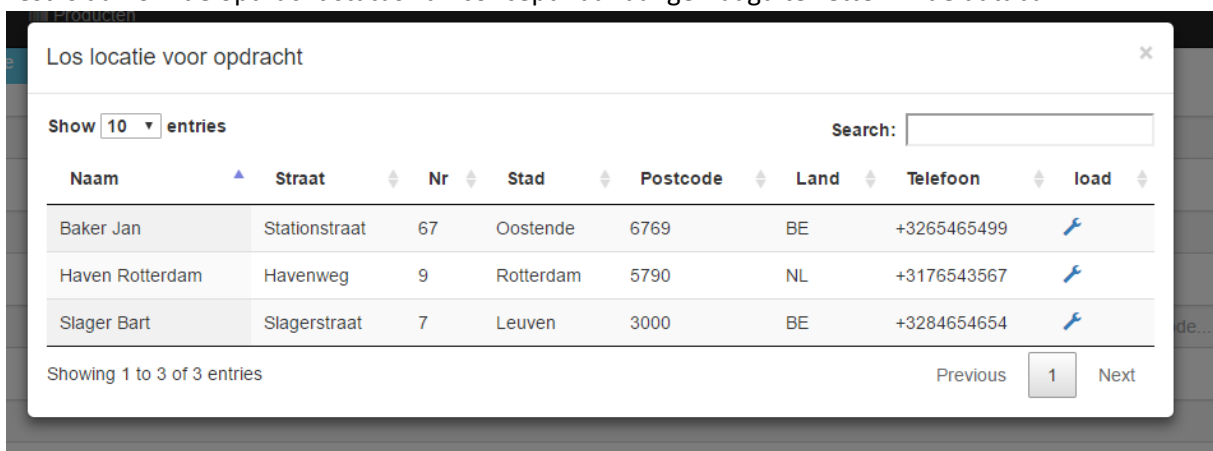
Showing 1 to 4 of 4 entries

Previous
1
Next

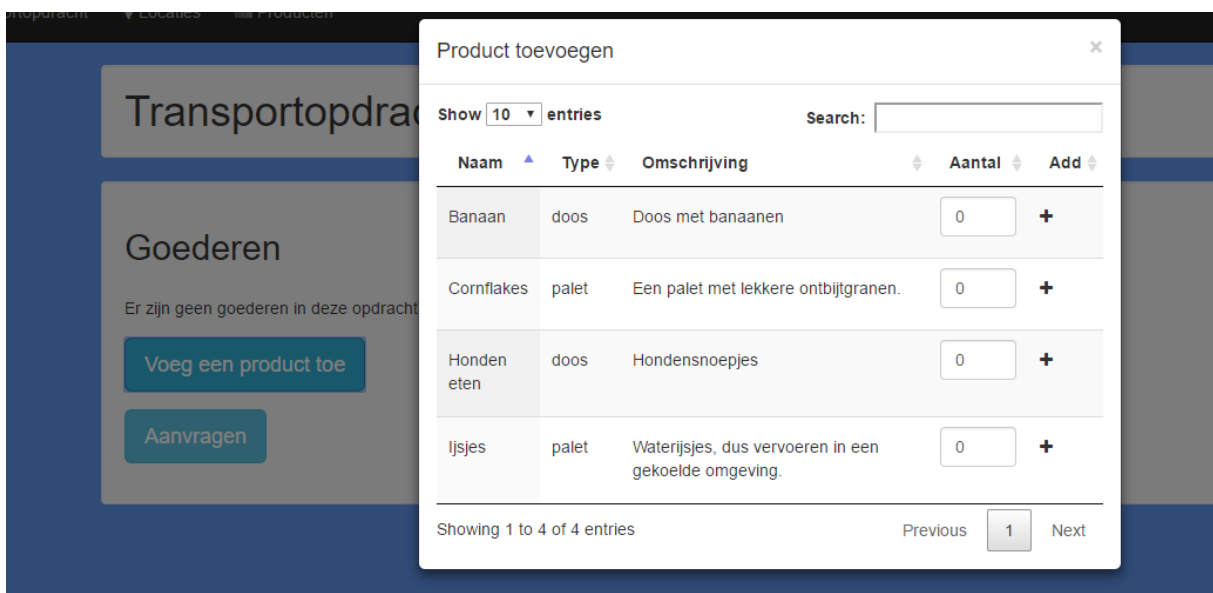
Afbeelding van product template

4.3 Transportopdracht aanmaken

Op deze pagina (zie bijlage 4.3.1) kan je de opdracht die je wilt uitvoeren aanmaken. Dit doe je door de locaties die je hebt aangemaakt in te laden. Verder wordt er nog informatie gevraagd zoals opmerkingen, los- en laad datums. Als de gebruiker op volgende klikt wordt er een validatie uitgevoerd zoals het nazien van de los en laad locaties, deze moeten verschillend zijn. Verder wordt gecontroleerd of dat de laad datum voor de los datum valt. Als dat allemaal in orde is wordt er in onze databank een opdracht aangemaakt met als status concept. In deze opdracht zitten dus nog geen producten. Als de klant bij deze stap de browser sluit of op een andere manier de pagina verlaat is zijn informatie dus opgeslagen. De locaties inladen wordt gedaan door een ajax call. Dan op de volgende pagina worden de producten toegevoegd met een waarde van het aantal er bij. De producten worden ook via een ajax call in de databank toegevoegd. De gebruiker kan ook producten verwijderen en het aantal aanpassen. Na deze stap kan de opdracht aangevraagd worden al wat nog rest is dan om de opdracht status van concept naar aangevraagd te zetten in de databank.



Afbeelding van het laden van een los of laad locatie.



Afbeelding van het toevoegen van de producten.

4.4 Opdrachten

Hier kan de klant zijn opdrachten bekijken, de opdracht heeft een status en hier zie je alle relevante statussen voor de klant. De eerste is goedgekeurd zo weet de klant welke opdrachten zijn goedgekeurd en dus doorgaan. De volgende is aangevraagd dit zijn de opdrachten die nog niet beantwoord zijn door het personeel van het transportbedrijf. Dan is er ook nog de status gewijzigd en dit betekent dat er iets gewijzigd is door het personeel en ook waarom. Dan krijgt de klant de keuze om hiermee akkoord te gaan of om de opdracht te stoppen. Daarna is er nog de afgekeurde opdrachten van de klant. De gegevens worden weergegeven in tabellen en er kan meer informatie worden opgevraagd waar de klant dan afhankelijk van de opdracht nog verder kan gaan.

Deze pagina is opgebouwd uit vier tabellen maar er is altijd maar één zichtbaar, dit komt door een javascript functie die de er voor zorgt dat bij een klik op een tab of bij een refresh de juiste tabellen niet meer zichtbaar zijn en de correcte tabel wel zichtbaar wordt (zie bijlage 4.4.1). Op deze pagina zit ook een automatisch vernieuwing (zie hoofdstuk 6).

Goedkeuringen1

Aanvragen1

Wijzigingen0

Afkeuringen1

Show10▼entries

Search:

Aangepast om	Laad Adres	Los Adres	Datum Laden	Datum Lossen	Meer Info
2017-05-10 16:33:34	Tiensesteenweg 89, Gent	Station 67b, Oostende	2017-06-15 20:00:00	2017-06-11 20:42:00	

Showing 1 to 1 of 1 entries

Previous

1

Next


Afbeelding van opdrachten

4.5 Geschiedenis

Dit is een archief van alle opdrachten die als status voltooid hebben. Dit wil zeggen dat de opdracht is doorgegaan en is uitgevoerd zonder complicaties. Deze informatie wordt in een tabel weergegeven. Ook hier kan je weer meer informatie opvragen via een klik op het mapje. Als klant krijg je hier alleen de opdrachten die van jou zijn te zien en als administratie krijg je hier alle opdrachten van al de gebruikers te zien.

Show 10 entries

Search:

Naam klant	Bedrijf	Laad Adres	Los Adres	Datum Laden	Datum Lossen	Meer info
Jonas Van Noten	Photobean	Rotterdam	Leuven	2017-05-13 20:40:00	2017-05-31 20:40:00	

Showing 1 to 1 of 1 entries

Previous

1

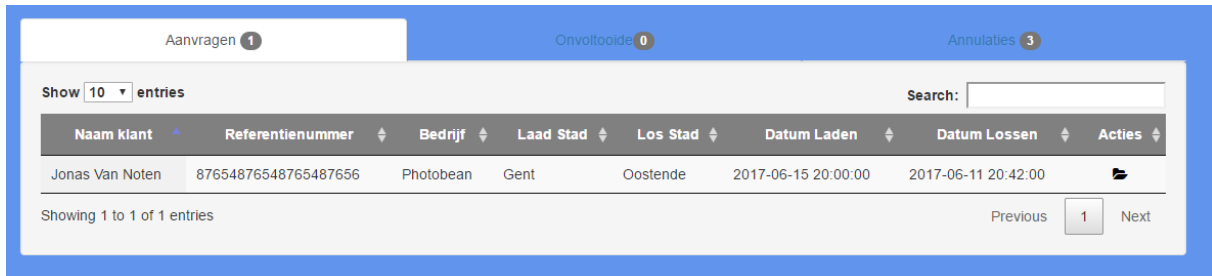
Next

Afbeelding van geschiedenis

4.6 Order request

Hier krijgt de administratie de binnengekomen opdrachten te zien en kunnen ze deze goedkeuren, afkeuren of aanpassen. Ook de onvoltooide opdrachten staan hier, dat zijn de opdrachten die goedgekeurd zijn maar nog niet uitgevoerd. Dan als laatste zijn er nog de geannuleerde opdrachten.

Als een opdracht aangepast wordt komt deze terug bij de klant terecht die dan akkoord kan gaan met de aanpassing of die de opdracht annuleert.

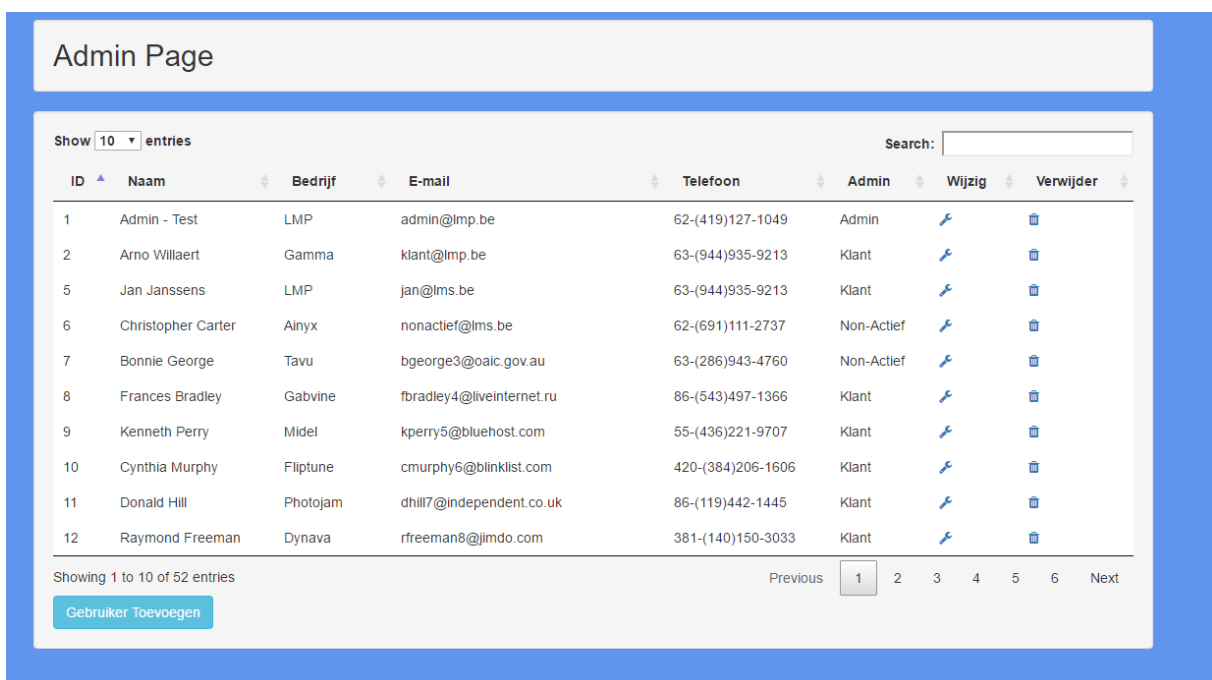


Naam klant	Referentienummer	Bedrijf	Laad Stad	Los Stad	Datum Laden	Datum Lossen	Acties
Jonas Van Noten	87654876548765487656	Photobean	Gent	Oostende	2017-06-15 20:00:00	2017-06-11 20:42:00	

Afbeelding van order request

4.7 Admin page

Deze pagina is voor de beheerders en hierop kunnen ze alle gebruikers zien (behalve de superusers) en deze verwijderen of wijzigen. Je kan hier ook de rechten van een gebruiker aanpassen. Bij het aanmaken van een nieuwe gebruiker worden de volgende zaken gevraagd: naam, bedrijf, e-mail, telefoon en bevoegdheid. De gebruiker krijgt dan een mail met de verificatielink en een wachtwoord dat dan gewijzigd kan worden.



ID	Naam	Bedrijf	E-mail	Telefoon	Admin	Wijzig	Verwijder
1	Admin - Test	LMP	admin@lmp.be	62-(419)127-1049	Admin		
2	Arno Willaert	Gamma	klant@lmp.be	63-(944)935-9213	Klant		
5	Jan Janssens	LMP	jan@lms.be	63-(944)935-9213	Klant		
6	Christopher Carter	Alnyx	nonactief@lms.be	62-(691)111-2737	Non-Actief		
7	Bonnie George	Tavu	bgeorge3@oaic.gov.au	63-(286)943-4760	Non-Actief		
8	Frances Bradley	Gabvine	fbradley4@liveinternet.ru	86-(543)497-1366	Klant		
9	Kenneth Perry	Midel	kperry5@bluehost.com	55-(436)221-9707	Klant		
10	Cynthia Murphy	Fliptune	cmurphy6@blinklist.com	420-(384)206-1606	Klant		
11	Donald Hill	Photojam	dhill7@independent.co.uk	86-(119)442-1445	Klant		
12	Raymond Freeman	Dynava	rfreeman8@jimdo.com	381-(140)150-3033	Klant		

Afbeelding van admin page.

5 Plugins

5.1 Tabellen in datatables

Om alle gegevens uit onze databank ordelijk te kunnen weergeven hebben we de jquery plugin datatables gebruikt. Dit laat ons toe om overzichtelijke tabellen weer te geven met een order en sorteer filter. Als de tabel te lang word zal deze plugin ook ervoor zorgen dat er meerdere pagina's zijn om in te zoeken. Om deze plugin te laten werken hebben we op elke pagina waar nodig de juiste .css en .js files in de header gezet die nodig zijn. Verder moeten de tabellen geladen worden in de document.ready functie. Elke tabel heeft een eigen id en dan doet de plugin de rest.

5.2 Datetimepicker plugin

Voor alle instanties waar het nodig is een tijd en een datum in te geven maken we gebruik van een jquery plugin met de originele naam Bootstrap 3 datetimepicker v4. De volledige documentatie is te vinden op de volgende website: <http://eonasdan.github.io/bootstrap-datetimepicker/>

We gaan hier weer niet in detail, maar we gaan beknopt de requirements opnoemen en toelichten hoe we de plugin oproepen.

Om de plugin te kunnen gebruiken heb je jquery.js nodig, moment.js om parameters mee te kunnen geven en eventuele localisatie. Verder is het ook nodig om de bootstrap javascript library te includen en de bootstrap css voor de opmaak. De plugin zelf is ook een js file en een css file die je moet includen.

Hieronder vind je de code die in de script tag moet staan van je html code, deze is afkomstig van createOrder.php.

```
1. $(function () {  
2.   $('#laaddatum').datetimepicker({  
3.     format: 'YYYY-MM-DD HH:mm',  
4.     stepping: 5,  
5.     minDate: moment()  
6.   });  
7. });  
8. $('#losdatum').datetimepicker({  
9.   format: 'YYYY-MM-DD HH:mm',  
10.  stepping: 5,  
11.  minDate: moment()  
12. });  
13. } );
```

Dankzij code weet de plugin op welke plaats hij de datetimepicker moet plaatsen, namelijk op de divisies met de id's #laaddatum en #losdatum.

De parameter format bepaald welke gegevens je opvraagt in welke vorm, stepping geeft aan wat het kleinste verschil is tussen waarden. Dit is hier ingesteld op 5 minuten. Het is dus enkel mogelijk om minuten te selecteren die een veelvoud zijn van 5.

De minDate bepaald de kleinste datum die de gebruiker kan ingeven. Hier wordt deze ingesteld op de huidige datum/tijd met de functie moment().

6 Automatische vernieuwing pagina

Op de pagina's opdrachten en order requests staan veel gegevens die kunnen veranderen terwijl je op de pagina zit. Om dit probleem aan te pakken is er een functie die met een ajax call gaat kijken of er iets van de informatie gewijzigd is in de databank. Als dit het geval is dan gaat de pagina vernieuwen.

Aangezien beide pagina's uit meerdere tabbladen bestaan en we een manier moesten vinden om in dezelfde tab te blijven na een refresh, zit hun id in een hidden input type. Deze wordt gewijzigd als je van tabblad veranderd. Het veranderen van een tab gebeurt ook via een javascript functie (zie eerder bij opdrachten) en dus was dit makkelijk te implementeren (zie bijlage 6.1). De ajax call zal een bericht returnen en afhankelijk daarvan gaan we de pagina's vernieuwen. Bij geen veranderingen is dit uiteraard niet nodig. Om te vermijden dat de pagina vernieuwt wanneer je deze net nodig hebt is er een interval ingesteld van X aantal seconden of minuten. We hebben ook een reset functie (zie bijlage 6.2) aangemaakt die dus het interval gaat resetten zodat de refresh functie niet kan doorgaan (deze gaat nazien of er veranderingen zijn en dus dit wil zeggen niet per sé een vernieuwing van de pagina). De reset functie wordt bij elke muisbeweging of elke keyboard toets uitgevoerd. Het dient als een manier om te detecteren of de gebruiker actief bezig is op de pagina. In de test omgeving stond deze interval op een korte tijd om makkelijk te kunnen debuggen.

De opdrachten of orders hebben allemaal een status en het is het aantal van deze status waarmee we veranderingen gaan opzoeken in de databank. De ajax call post dus deze waardes door, afhankelijk van de pagina zijn dit er 3 of 4. Voor elke status wordt er dan vergeleken en als er dan minstens één is waarbij er een verschil op zit wordt een boodschap terug gestuurd die dan er voor zorgt dat de pagina vernieuwt. We hielden ook het id van de huidige tab bij en deze waarde word via een hidden veld met een gekend id door de javascript gebruikt om dan een get te sturen naar de pagina en zo kom je terug in de juiste tab.

7 Hashen wachtwoorden

De wachtwoorden worden opgeslagen in de databank. Het eerste wachtwoord voor elke account is een 6 cijferige code die willekeurig is. We gaan de wachtwoorden eerst hashen alvorens ze toe te voegen in de databank. Hiervoor gebruiken we PBKDF2 (Password-Based Key Derivation Function). Dit is een key stretching algorithm, dit wil zeggen dat het de wachtwoorden intensiever gaat hashen (het hashen gebeurt met SHA256) zodat het langer duurt om deze te hashen (zie bijlage 7.1). Hierdoor worden brute-force en dictionary aanvallen minder effectief. We gaan ook nog een salt toevoegen die willekeurig aangemaakt wordt en dan wordt opgeslagen in de databank, zo is elke salt uniek voor elke gebruiker. (zie bijlage 7.2 en 7.3)

8 Https encryptie op de web server.

Voor de beveiliging van het platform en de privacy van de gebruikers gebruiken we het https protocol. De gebruikers moeten zich aanmelden en ook belangrijke gegevens over de opdrachten uitwisselen en hierdoor is het belangrijk dat al deze data beveiligd beveiligd is. Hieronder is een vergelijking tussen http en https in Wireshark zo kan je zien dat het wachtwoord in http gewoon in tekst te lezen is en in https is alles geëncrypteerd. Doordat we https gebruiken krijg je in de browser een groen slotje te zien en dit schept ook vertrouwen met de gebruiker. We werkten in een Windows omgeving en maakten gebruik van XAMPP. In de Apache folder van XAMPP is er een .bat file om een certificaat te genereren. We hebben de certificaten dus zelf aangemaakt en dan geïmporteerd in onze browser om dit resultaat te bereiken. Dan nog een paar kleine aanpassingen in de config files van XAMPP en alles stond klaar voor deze veilige verbinding.

```
POST /phpsql/test_Project/Project_Web/login_check.php HTTP/1.1
Host: 10.68.241.13
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101 Firefox/53.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: nl,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://10.68.241.13/phpsql/test_Project/Project_Web/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 49
Cookie: PHPSESSID=nvgbv2qeppdvg5d2qp54ptlt0
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
email=superuser%40imp.be&password=mijsgeboortetejaarHTTP/1.1 302 Found
Date: Mon, 15 May 2017 13:30:23 GMT
Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2h PHP/5.6.24
X-Powered-By: PHP/5.6.24
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Location: login.php
Content-Length: 0
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
```

```
GET /phpsql/test_Project/Project_Web/login.php HTTP/1.1
Host: 10.68.241.13
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101 Firefox/53.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: nl,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://10.68.241.13/phpsql/test_Project/Project_Web/login.php
Cookie: PHPSESSID=nvgbv2qeppdvg5d2qp54ptlt0
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Mon, 15 May 2017 13:30:24 GMT
Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2h PHP/5.6.24
X-Powered-By: PHP/5.6.24
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Length: 2711
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
```

Afbeelding van Wireshark bij het inloggen over http.

```
.....Q%<$.B...).k.P...p{'.$.O-Z{.m...%....}.0')0...+./.....0.
.....3.9././5.
.....Y.....
.....#...i...WT...).3.....a.6.l.%Z...C.P.....X
.....2X...@.HLA...T...ks...S...UP.B...~YDc...n[B#m./.(.W.eY.....*V.CC*"-g.m.TQ#...u,k&|...../.....%.*.....h2.http/1.1.....
.....<...U.FQV...dg[0...f]^D...57.B2Qp...V...[ko.&...{>$.O-Z{.m...%....}.0')0...0.....http/1.1.....
.....J..lW.0.m1.6.....A...5.w.%of.]Gill.l&y7.#...l...gP+.....V'i.f.....mq...S.....8f...../.....KV_EQ...}.03+...;Tf...
.....$y'<'Q(F{S.R...}.O...}yS...%.V.....oW.
$Y...X...k;...e+e.x...+...1#J...Z...w.o...e.q>...V...Y...B/>FowN^k/.....ot...C...[.yIm.v5.c.Buv.....
.....e..L*bv5(.....<G...s.....Av)...?2"i...s.....h...u...sF...Z...Q..IM0...F.../.....?fJ...t.j7[%].....|W.r...lX...tw.....yw.e.....Ec...e..K.S0.
$].....dCj.y8...+a.JT].1Qz...P...4...X.....V...J...G...e.....8...m...m...|...V...A.u.T.S.....g...D.O.....z...$.C...b...L...W.b.oq...d.h.&...
+T.t...s...w.f.k.B...+...f...l...l...K...T0T.TS...f.Q.S.^[+...2...#...N...$.KFN.Tiu.....=...%.B.....#...j...w...8.B.W.../.....
po.....ssG<...<...
I...T...T...C...8.W...b...).*.s...i...*.7L...;o.Uv)*.I...}.M.H...{.o.n.
$.ph...X.X.(.'O^6N\...T.M...^>...Kz.U...FqA.....-P.P+&...C...Jf...h
.....?K.3q...e
.....9i...Xc~g~H...G...t.....m...&.trg>...?n...g...@.8c5;...+.#6D.o]...2.Y...^...
.....\...3.3p...1.9me(...../.....n'.K'.Hj.F...e;/...)+...i...$.B.p.8.V&...].Y+...E...J...hv...CI)...F.P...lNB...g.2]ia@tT(Yg.@...EsN...S...
D'(j...?k...y...Vm.b.c.4.4u.9aV...P...w...x...j...R...H...J...E...>...i...k.9.)...K...ts[.g...U...hM3...R...X...W...-95...k...M3...NE]>...1...IBd...
%...hu...T*...K.Kj...v...aV^...^...y...S%{.l.S]...X.Q.O.S...
%...+...l...o...A...N...4...g...aZ.u1.?].Vj.te1.e.1.13g.3.*"...)*...j...h.S...7.bW...p...+v^...b...}.W_#.....%r.
6.....e...Qv...pD.I...T.R]...B...af...cn.D...>2...q
4:aw...x...>.(k...E?...F...%.RH.S...4A...5Y...Yt20a...>...H...}.p...G$?J4...T...c...l.7.....P5...A...}.0E.a7...;7.Bd[9...{8.Tf...2P...
2...@...b...[E...I=7...$.I...HK...7...t...t...g...bG.5...I.e1.....(f.s...i...)^.k9<...p...<...PWE@...o...tOC...}.X...J.&...5.<i...
5.v.[$.Ha.8]...s...<@.6...c6[...s...].+...b...#...A/g$y0.7#?
f...o...XrhQ...2.2.3]...l.I]b...C\X...v...X.O.I.GZ...p...i.B+...GO.
zhr...UNl.D...g...PXcv.@Lo...o...a...l...p...=R...c.l.h%?u...uf.K.qNj.D]...<.....d...x3...\\./...II.h.eD.a2m.3...X.O.q...{.....d.....d...1...y[...F...
6.P.g?...k...o...L.8.H...l.q.m...k...t;
73...4v6...Q...h...y...p#4.2.3...61...].L.&.h.NH...q]q.s...K...Lof...".c...l...r...P.J...3...o.%mZ.SR...}{1...}C...
5.c]...e...C4...e...I...h...l...SuD5.E...}."P...
7.v1...I4&t)...n...3...%...Y.U...y...l.X.y...-g...+].7...h...TD^B...h...N.V...<w...I.I...
lE@...gc;0...W%...8.I.u.'D...LB...C...U...+8...y...w.l)y.^j.D=.....N.9)p:DN70...x.b.Ek...t+...4J...|.../...x...
7.2...@7op...>...r...j8g.U"...3...>3...U...+01...{T.U...R...p
.....N...l...d.../.....(t.OV...t{X...O...+^...+$.V...S.Fg...$p.b...4...F.IN...4...nJ...r...#U.k3...A...~d.l...n.@)...pr...k...c...J.zB'g.
...).%.%ku#8.S...
...N&.../...LNE...C...0
a.\.Z...s...@
...U.../...D...f.../6R3...1...s...0...7hx{K.41.I.j.v.N...#...^...Vy.P...~RM.2;0... (1...%...$S.R.Y...$.1MI...&J.%...K.O"...a...?
w.V...n.O...D"...f.../Q#51...JL...^...L...Z...Q.m...l...r...g=q.o4.Y.1.6...$.u...a6...l...e...}....c
...pxJ...ulrx...&.tu...3.Ej...z0...:...=...].VW...#V...0...54...M"?...n...8...6.Zf...#V...q...l...o...t...K4...DR...!
@.8...H-@]K[...U...f...r...E.E=Z.?#:&0...ms...U...05...NC...9U...A...ekq0...w...s
.sV.../2.B...U.P...V...A...N.H...N.T...a.k.T...k&...C...C...9us...
...ZL2VT...T
{.m...").*...QT...*oWj...l...Q...O...De^a.B.*nc...C...B...p>...TZ...p...H...H#D2...Sogk*.@E#A.cif.kS...3...[J...e.nl
...i...E...V...<7...T...t...l.Y...+...
'g...0.YZf...o.1.09.@...w.6.m...
CU.x?...}F...".(=Z...^...=...W.y...H...XV...=...F...g...n
I...=...s.m.
.@.8.Ao)f?...Ef]...2...'.F...v]5...#...6j.U...%)^~$+...o.4.#0...%.k...L...$.2']...S...+`N...l...E...(.4...[...T.W...>z...[y...+...4'...
...H...w...*m
sR.O(.5...}k3=c.a^...IL...P]U...b...b...((jA...
...D.J.I.F...Idy.j...8..0...+vJ.z...P...M/O...O...e...}x...?^.*%5...P...t...mG^...g.8k...|...7.IU...z.F^A.W.
...V.WV...5...+9~D...R...3P...7...tHn...T...S...U...T.1...$.../...[e1.Z.L...IM.Ef...2...3...%.DY;#Y...
8:V.WV...T...x...}.y.r.17...f'g...[...A...
X0sc...I...R...{...r.0.[.'V.e.8...}.dpdAd.A.z&...&*/...L...a...O...I+
.(T...K...m3>.04Z...gE...d...8...?j...IB]qQ
```

Afbeelding van inloggen over het https protocol.

9 Verificatie via mail

Als een gebruiker zijn account is aangemaakt door iemand van de beheerders dan krijgt deze een mail met daarin een link naar de activatiepagina en ook een willekeurig wachtwoord bestaande uit 6 cijfers, hiermee kan de gebruiker dan inloggen en daarna zijn wachtwoord wijzigen naar zijn voorkeur. De beheerder stuurt een form door naar een php pagina en daar wordt de informatie verwerkt en worden er query's gestuurd om de gebruiker dan ook weldegelijk aan te maken in de databank. Hierna wordt de gebruiker op de hoogte gebracht door een mail die hem naar de verificatie pagina zal brengen waarop hij als alles goed verloopt door verwezen wordt naar de log in pagina waar de gebruiker dan kan aanmelden. In de link zit een token waarmee de gebruiker zich authenticiseert. Als dit token fout is of het token is vervallen dan komt er een boodschap op om de gebruiker te informeren(zie bijlage 9.1).



Logistic Management Platform
23:29

Activeerings e-mail voor account bij LMP

Aan: Jonas Van Noten

Beste Jonas Van Noten, klik op deze link om je account te verifiëren en gebruik dan om in te loggen deze pincode as wachtwoord : 497155.
http://localhost/project_web/verification.php?id=57&token=8fbcfcbcb8af4ca637829ce1908d446f3d73888ac4ea4b77c97370d896cf66ef

Conclusie

Over het algemeen durven we zeggen dat we zijn geslaagd in het maken van een helder communicatiekanaal tussen klant en planningsafdeling. Hier en daar zijn we wel afgeweken van het vooraf opgestelde plan. Soms uit noodzaak, soms om een betere designkeuze te maken.

Een van de punten waar we sterk van zijn afgeweken was de mogelijkheid tot losse invoer in de pagina om een transportopdracht te maken. Het originele plan was om de klant de mogelijkheid te geven van locaties in te laden, maar hem ook de vrijheid te geven van de velden manueel in te voegen. Dit had als voordeel voor de klant dat wanneer hij slecht een keer ergens moest leveren, zoals bij een thuislevering, hij hier geen locatie template voor moest opslaan. Hierdoor hadden we in onze databank geen relatie tussen opdracht en locaties, en toen ook nog geen relatie tussen opdracht en goederen. Dit gaf als probleem dat we een grote hoeveelheid redundante informatie gingen opslaan in de tabel opdracht.

Toen we dit aanpasten zorgde dit wel voor een aanzienlijke vertraging, aangezien we onze databanken moesten herschrijven.

Dit is later nog een keer gebeurt, toen we ons realiseerden dat om meerdere producten te kunnen opslaan in de opdrachttabel, we een veel op veel relatie nodig hadden tussen opdrachten en goederen. Toen hebben we weer onze sql code grondig aangepast.

Waar we achteraf ook bedenkingen bij hebben is de admin user, nu kan die enkel gebruikers aanpassen, creëren en uitzetten. We denken dat het beter kan zijn om de admin ook dezelfde permissies te geven als de personeel accounts, aangezien de admin waarschijnlijk een interne werknemer is in het transportbedrijf en dus ook transportopdrachten zou moeten kunnen aanmaken.

Wat ik (wat hier volgt is de mening van Arno Willaert) ook heb ontdekt halverwege het project is dat wanneer ik hieraan opnieuw zou beginnen, zou ik zeker en vast gebruik maken van Angular.js om formulieren te verwerken, en responsieve elementen in de website te bouwen.

De jQuery plugins die we hebben gebruikt (datatables en datetimepicker) kunnen we alleen maar aanraden, ze waren relatief eenvoudig bij ingebruikname en configuratie.

Bijlage

Transportopdracht aanmaken

Laad Informatie

Laad locatie

Naam

Naam...

Straat

Straat...

Nummer

Nummer...

Stad

Stad...

Postcode

Postcode...

Land

Landcode...

Telefoon

Telefoon...

Los Informatie

Los locatie

Naam

Naam...

Straat

Straat...

Nummer

Nummer...

Stad

Stad...

Postcode

Postcode...

Land

Landcode...

Telefoon

Telefoon...

4.3.1 Afbeelding van de pagina transportopdracht aanmaken.

```
function klik(id)
{
    document.getElementById("li1").className = '';
    document.getElementById("li2").className = '';
    document.getElementById("li3").className = '';
    document.getElementById("li4").className = '';
    document.getElementById("li"+id).className = 'active';
    document.getElementById("div1").style.display = "none";
    document.getElementById("div2").style.display = "none";
    document.getElementById("div3").style.display = "none";
    document.getElementById("div4").style.display = "none";
    document.getElementById("div"+id).style.display = "block";
    document.getElementById("currid").value = id;
}
```

4.4.1 functie om van tab te wisselen en de juiste tabel te tonen.

```
function check(){

    //prepare request
    $request = $.ajax({
        method:"POST",
        url:"count_check_ajax.php?r=" + new Date().getTime(),
        data: {badge1: $("#badge1").html() , badge2: $("#badge2").html()
, badge3: $("#badge3").html() , badge4: $("#badge4").html() }
    });
```

```

        $request.done(function(message)
        {
            if(message==0)
            {
                id = document.getElementById("currid").value
                window.location.href = "opdrachten.php?keuze="+id;
            }
        });

        $request.fail(function(message)
        {
            test.innerHTML = 'error';
        });
    }
}

```

6.1 functie die ajax call aanmaakt en zal vernieuwen als er een verschil is.

```

function reset(){
clearInterval(timer);
timer = setInterval(check, 3000);
}
<body onmousemove="reset();" onkeydown="reset();">

```

6.2 functie reset die de interval reset en eventhandlers op de body.

The following benchmarks demonstrate the running time for various iteration counts, using the SHA256 hash function. The benchmarks were run on an AMD Phenom 9600 2.3GHz CPU.

```

1 iteration : 0.000149 seconds
2 iterations: 0.000036 seconds
4 iterations: 0.000038 seconds
8 iterations: 0.000052 seconds
16 iterations: 0.000090 seconds
32 iterations: 0.000157 seconds
64 iterations: 0.000297 seconds
128 iterations: 0.000623 seconds
256 iterations: 0.001141 seconds
512 iterations: 0.002258 seconds
1024 iterations: 0.004594 seconds
2048 iterations: 0.009575 seconds
4096 iterations: 0.018386 seconds
8192 iterations: 0.036070 seconds
16384 iterations: 0.073297 seconds
32768 iterations: 0.145324 seconds
65536 iterations: 0.294785 seconds
131072 iterations: 0.577492 seconds
262144 iterations: 1.173854 seconds
524288 iterations: 2.334627 seconds
1048576 iterations: 4.688382 seconds
2097152 iterations: 9.249891 seconds
4194304 iterations: 18.69492 seconds
8388608 iterations: 36.90171 seconds
16777216 iterations: 75.31797 seconds

```

7.1 Benchmark url: <https://defuse.ca/php-pbkdf2.htm>

```

function pbkdf2($algorithm, $password, $salt, $count, $key_length, $raw_output = false)
{
    $algorithm = strtolower($algorithm);
    if(!in_array($algorithm, hash_algos(), true))
        trigger_error('PBKDF2 ERROR: Invalid hash algorithm.', E_USER_ERROR);
    if($count <= 0 || $key_length <= 0)
        trigger_error('PBKDF2 ERROR: Invalid parameters.', E_USER_ERROR);

    if (function_exists("hash_pbkdf2")) {
        // The output length is in NIBBLES (4-bits) if $raw output is false!

```

```

        if (!$raw_output) {
            $key_length = $key_length * 2;
        }
        return hash_pbkdf2($algorithm, $password, $salt, $count, $key_length, $raw_output);
    }

    $hash_length = strlen(hash($algorithm, "", true));
    $block_count = ceil($key_length / $hash_length);

    $output = "";
    for($i = 1; $i <= $block_count; $i++) {
        // $i encoded as 4 bytes, big endian.
        $last = $salt . pack("N", $i);
        // first iteration
        $last = $xorsum = hash_hmac($algorithm, $last, $password, true);
        // perform the other $count - 1 iterations
        for ($j = 1; $j < $count; $j++) {
            $xorsum ^= ($last = hash_hmac($algorithm, $last, $password, true));
        }
        $output .= $xorsum;
    }

    if($raw_output)
        return substr($output, 0, $key_length);
    else
        return bin2hex(substr($output, 0, $key_length));
}

```

7.2 gebruikte functie voor hashen wachtwoorden source: <https://defuse.ca/php-pbkdf2.htm>

```

$bytes = openssl_random_pseudo_bytes($length, $true);
$salt = bin2hex($bytes);

```

7.3 gebruikte code om de salt aan te maken

```

if(isset($_GET['id']) & isset($_GET['token']))
{
    $klant_id = prep($_GET['id']);
    $token = prep($_GET['token']);

    $link = mysqli_connect(hostname, user, password) or die("Could not connect to
hostname");

    mysqli_select_db($link,database) or die("databank $database niet beschikbaar");
    //kijken of token juist is
    $query = "Select * from verificatie where klant_id=$klant_id and token='$token'";
    $result = mysqli_query($link, $query) or die ("Foute token");
    if (mysqli_num_rows($result)==1)
    {
        while($rij = mysqli_fetch_array($result))
        {
            $priv=$rij['priv'];
            $aanvraagtijd=$rij['aanvraagtijd'];
        }
        //nakijken of token al vervallen is

        $datetime_aanvraag = new DateTime($aanvraagtijd);
        $datetime_now = new DateTime();
        // $add is de var voor de tijd dat de klant heeft na de mail om deze te bevestigen.
        $add="1 day";
        date_add($datetime_aanvraag, date_interval_create_from_date_string($add));
        if ($datetime_now>$datetime_aanvraag)
        {
            echo ("deze link is vervalen");
            exit();
        }
        //priv van klant instellen zodat deze kan inloggen
        $query = "UPDATE klant SET priv=$priv WHERE priv=0 and id=$klant_id";
        $result = mysqli_query($link, $query) or die ("Error");
        //token deleten
        $query = "DELETE FROM verificatie WHERE klant_id=$klant_id";
        $result = mysqli_query($link, $query) or die ("Error");
    }
}

```

9.1 validatie van het token