

Geïntegreerde proef

Struggle

DL3

Groep 4:
Gheeraert Alec, Ostyn Lars, Popal Ares

Voorwoord

Wij besloten om een webshop te maken voor het kledingmerk “Struggle” doordat de eigenaar van het merk een kennis is van één van de groepsleden. Toen we besloten hadden wie voor wat ging zorgen hebben we de rollen verdeeld. Dit door middel van te kijken naar ieder zijn voorkeur. We zijn uitgekomen op: Alec kijkt voor de ‘looks’ van de webshop, Lars zal zich focussen op de database en Ares zal een groot deel van het programmeren doen. Maar natuurlijk zullen we elkaar helpen bij problemen. Met onze website zal je een unieke stijl van kledij kunnen zien die nog nergens anders te zien is doordat de producent van de kledij nog niet zo groot in de markt is.

Inhoud

1	Inleiding	1
2	Functioneel ontwerp	2
2.1	Klantenpagina's	2
2.1.1	Homepage	2
2.1.2	Producten	2
2.1.3	Bestellingen	Fout! Bladwijzer niet gedefinieerd.
2.1.4	Navigatie	2
2.1.5	Registreren en aanmelden	3
2.2	Administratorpagina's	3
2.2.1	Klantenbeheer	3
2.2.2	Productbeheer	3
2.2.3	Bestellingen	Fout! Bladwijzer niet gedefinieerd.
2.3	Datamodel	3
2.3.1	Conceptueel Entity-Relationship Diagram	3
2.3.2	Functioneel Entity-Relationship Diagram	3
2.3.3	tblKlanten	5
2.3.4	tblProducten	5
2.3.5	tblProductCategorie	5
2.3.6	tblBestellingen	6
2.3.7	tblProductBestellingen	6
3	Technisch ontwerp	7
3.1	Hoe ga je alles maken?	7
3.2	Structuur van de website	7
4	Planning	8
4.1	Wie doet wat?	8
4.2	Logboek	8
5	Besluit	9

Onze website zal dienen als online webshop voor een gepersonaliseerd kledingmerk, genaamd Struggle. De website wordt geopend op een inleidende pagina waar er voorbeeld foto's op komen. Vanuit die pagina zal de gebruiker meteen naar de shop pagina kunnen gaan zonder meteen in te moeten loggen, maar de gebruiker kan zich ook meteen inloggen wanneer de gebruiker een admin is. Als de gebruiker nog geen login heeft wordt men gestuurd naar een registratiepagina waar men gratis kan registreren om later het aankopen van producten te versnellen.

1.1 Gebruiker gaat meteen naar de shop vooraleer zich aan te melden.

De klant krijgt de pagina te zien waar de producten allemaal op vertoond worden. Hij/zij kan ook op een kledingstuk naar keuze klikken om de extra informatie over het kledingstuk te kunnen zien, zoals: De beschikbare maten, kleuren en de materie waaruit het product het gemaakt is. De klant kan op ieder moment kiezen om zich aan te melden of zich te registreren als ze nog geen account hebben.

1.2 Gebruiker meldt zich meteen aan.

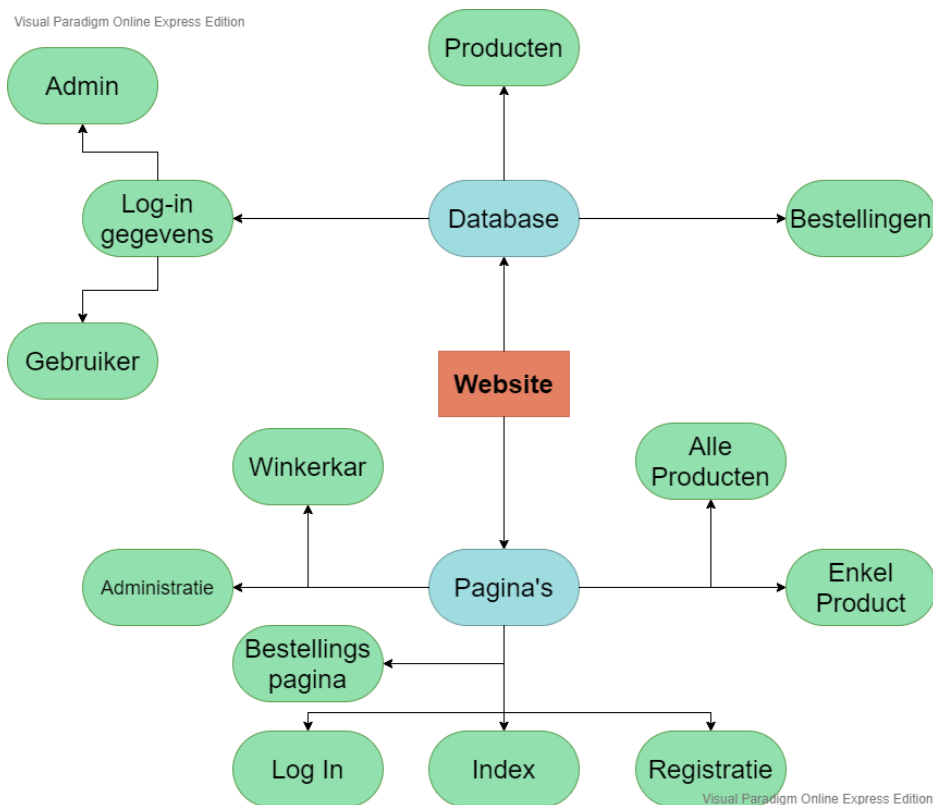
De gebruiker komt op een inlogpagina waar hij/zij kan kiezen om zich te registreren of meteen in te loggen. Als de gebruiker een admin is, komt deze op de adminpagina waar zij recente aankopen kunnen zien. Op deze pagina kunt die ook de shop aanpassen.

Als de gebruiker nog geen login heeft, zal er een registratie pagina komen waar de gebruiker zijn gegevens zal ingeven. Als zij dat gedaan hebben moeten zij zich niet meer inloggen om de website te gebruiken.

Als ze wel een login hebben maar ze geven de verkeerde gegevens in, zal de website dan voorstellen om contact op te nemen met één van de bedrijfsleiders om hem/haar verder te helpen.

Lukt de login wel vlekkeloos, komt de gebruiker dan op de shoppagina en kan hij/zij beginnen met winkelen.

2 Functioneel ontwerp



2.1 Klantenpagina's

2.1.1 Homepage

Wanneer je op de homepage komt, zal je de 4 meest verkochte producten zien die op dat moment het populairst zijn die de klant al een kleine voorproef geven wat er op de webshop zal te vinden zijn. Op de homepage kan de klant op een knop drukken waar hij/zij wordt doorgestuurd naar een pagina waar hij/zij de mogelijkheid heeft om te kunnen inloggen met zijn logingegevens of een hij/zij krijgt mogelijkheid tot registratie voor een nieuwe gebruiker aan te maken. De klant kan ook rechtstreeks naar de productenpagina gaan om al eens te kunnen browsen tussen alle producten die de webshop aan te bieden heeft.

2.1.2 Producten

Wanneer je op de pagina van de producten komt ziet de klant een lijst van alle beschikbare producten. De klant kan door de verschillende categorieën op de pagina gaan om te kijken wat er interessant lijkt. Als de klant op een bepaald product klikt, wordt hij/zij naar een ander pagina gestuurd waar er specifiekere informatie staat over het aangeklikte product. De klant kan ook ieder moment terugkeren van de specificatie pagina terug naar de 'Main page' waar de lijst van alle producten staat.

2.1.3 Navigatie

De navigatie zal op iedere pagina te vinden via een navigatie bar die bovenaan iedere pagina te vinden is. De klant kan zich zo ook makkelijk navigeren door alle pagina's.

2.1.4 Registreren en aanmelden

De klant kan op ieder moment van het browsen door de webshop zich aanmelden of wanneer deze nog geen login heeft, kan men zich makkelijk registreren. Wanneer men als admin wilt aanmelden, geeft men gewoon de voor geprogrammeerde inloggegevens in op de zelfde pagina waar de klanten zich ook inloggen.

2.2 Administratorpagina's

2.2.1 Klantenbeheer

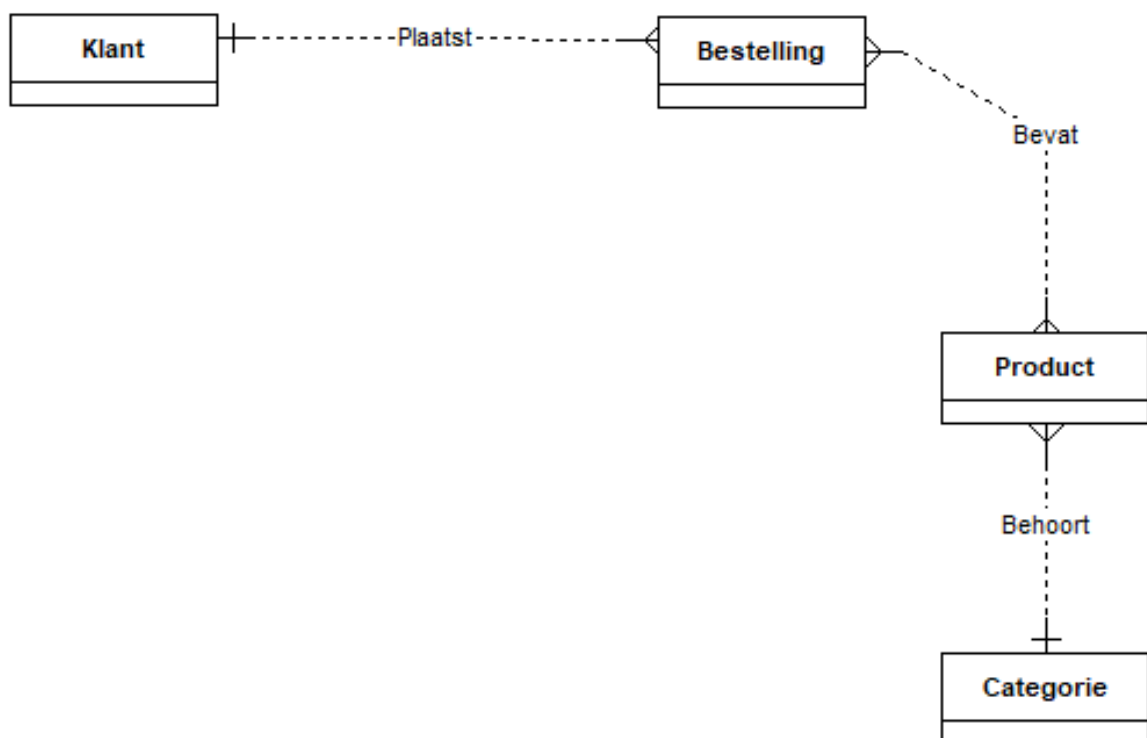
De admin heeft de mogelijkheid om alle klanten te zien die al hebben ingelogd op de webshop. Hij/zij kan dan ook de gegevens van de ingelogde personen zien die de klant heeft moeten ingeven tijdens het registreren. Hierin kan de admin ook zien wat klanten al hebben besteld binnen de webshop.

2.2.2 Productbeheer

De admin kan via de website het productbeheer bekijken en aanpassen waar nodig.

2.3 Datamodel

2.3.1 Conceptueel Entity-Relationship Diagram

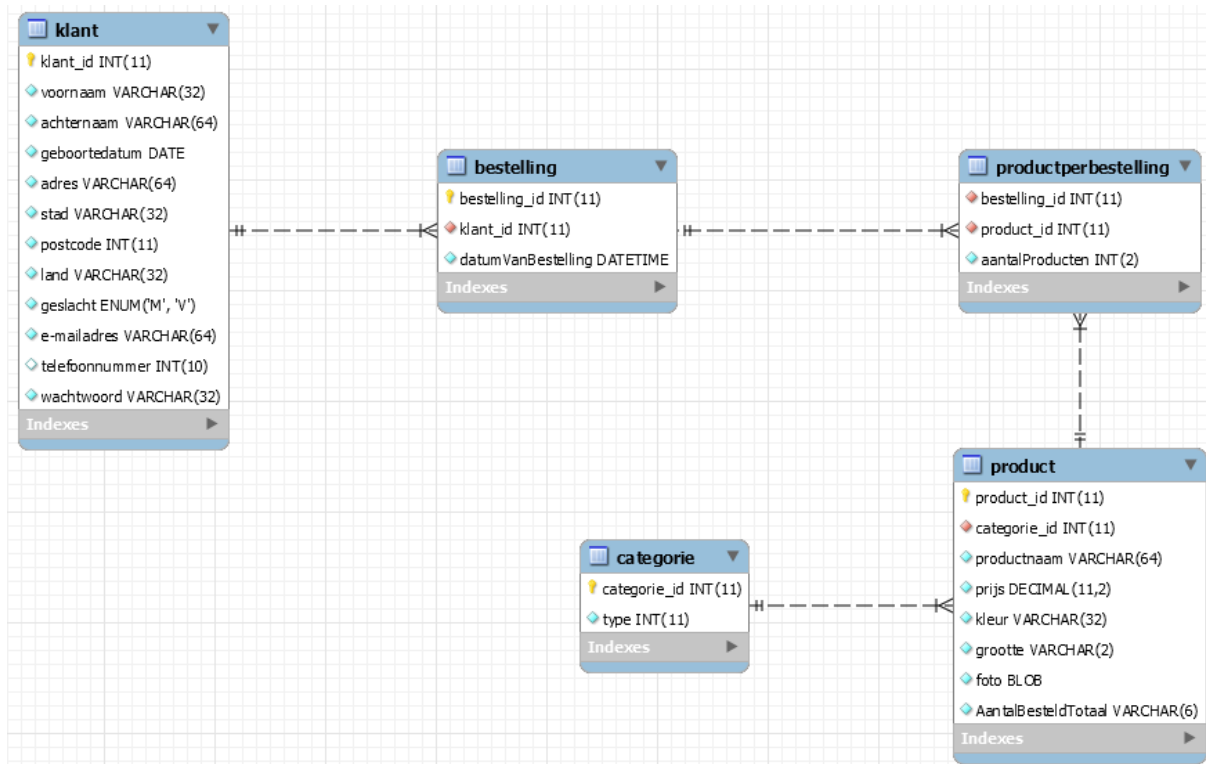


Een klant kan meerdere bestellingen plaatsen, maar een bestelling kan maar tot 1 klant behoren.

Een bestelling kan meerdere producten bevatten en een producten kan in meerdere bestellingen voorkomen.

Een categorie kan meerdere producten bevatten, maar een product kan maar tot 1 categorie.

Functioneel Entity-Relationship Diagram




Dit zijn de tabellen waar alle gegevens van de webshop zullen worden in opgeslagen worden.

2.3.2 tblKlanten

	#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1	klant_id 	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2	voornaam	varchar(32)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	3	achternaam	varchar(64)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	4	geboortedatum	date			No	None	
<input type="checkbox"/>	5	adres	varchar(64)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	6	stad	varchar(32)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	7	postcode	int(11)			No	None	
<input type="checkbox"/>	8	land	varchar(32)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	9	geslacht	enum('M', 'V')	latin1_bin		No	None	
<input type="checkbox"/>	10	e-mailadres	varchar(64)	latin1_swedish_ci		No	None	
<input type="checkbox"/>	11	telefoonnummer	int(10)			Yes	0	
<input type="checkbox"/>	12	wachtwoord	varchar(32)	latin1_swedish_ci		No	None	


De admin wordt gescheiden van gewone gebruikers via een hard gecodeerd emailadres en een hard gecodeerd wachtwoord.

2.3.3 tblProducten

	#	Naam	Type	Collatie	Attributen	Leeg	Standaardwaarde	Extra
<input type="checkbox"/>	1	product_id 	int(11)			Nee	Geen	AUTO_INCREMENT
<input type="checkbox"/>	2	categorie_id 	int(11)			Nee	Geen	
<input type="checkbox"/>	3	productnaam	varchar(64)	latin1_swedish_ci		Nee	Geen	
<input type="checkbox"/>	4	prijs	decimal(11,2)			Nee	Geen	
<input type="checkbox"/>	5	kleur	varchar(32)	latin1_swedish_ci		Nee	Geen	
<input type="checkbox"/>	6	grootte	varchar(2)	latin1_swedish_ci		Nee	M	
<input type="checkbox"/>	7	foto	blob			Nee	Geen	
<input type="checkbox"/>	8	AantalBesteldTotaal	varchar(6)	latin1_swedish_ci		Nee	0	



AantalBesteldTotaal: Hierin zal getoond worden hoeveel keer het artikel al besteld is geweest om zo te kunnen zien wat er gerestocked moet worden en wat er niet erg geïnteresseerd is voor de klant.

2.3.4 tblProductCategorie

	#	Naam	Type	Collatie	Attributen	Leeg	Standaardwaarde	Extra
<input type="checkbox"/>	1	categorie_id 	int(11)			Nee	Geen	AUTO_INCREMENT
<input type="checkbox"/>	2	type	int(11)			Nee	Geen	



Dit is een tabel dat ervoor zorgt dat het zoeken voor en bepaald product makkelijker maakt.

2.3.5 tblBestellingen

	#	Naam	Type	Collatie	Attributen	Leeg	Standaardwaarde	Extra
<input type="checkbox"/>	1	bestelling_id 	int(11)			Nee	Geen	
<input type="checkbox"/>	2	klant_id 	int(11)			Nee	Geen	
<input type="checkbox"/>	3	datumVanBestelling	datetime			Nee	CURRENT_TIMESTAMP	

In deze tabel zal de info terecht komen van de bestelling, de klant van wie deze bestelling is en de datum wanneer de bestelling is geplaatst.

2.3.6 tblProductPerBestellingen

	#	Naam	Type	Collatie	Attributen	Leeg	Standaardwaarde	Extra
<input type="checkbox"/>	1	bestelling_id 	int(11)			Nee	Geen	
<input type="checkbox"/>	2	product_id 	int(11)			Nee	Geen	
<input type="checkbox"/>	3	aantalProducten	int(2)			Nee	1	

Deze tussentabel toont de hoeveelheid artikelen van een product dat er besteld worden door de klant.

3 Technisch ontwerp

3.1 Hoe ga je alles maken?

In het functioneel ontwerp heb je beschreven hoe de website eruitziet en wat de mogelijkheden zijn. Uit die analyse moet je nu kunnen halen wat je allemaal nodig hebt om de website te realiseren. In dit onderdeel bespreek je m.a.w. **HOE** je het allemaal gaat maken.

Welke technologieën heb je nodig om ieder onderdeel van het project te realiseren?

- PHP? Wat? Waarom?
- MySQL? Wat? Waarom?
- HTML5? CSS? Bootstrap? Versie? Wat? Waarom?
- Extra libraries nodig? Bv Datatables, Datepicker, ...?

Motiveer jouw keuze. Zoek bv op het internet wat voor- en nadelen kunnen zijn van de gekozen technologie.

Als je nog niet weet hoe je iets zal implementeren (wat nog perfect normaal kan zien in deze fase van het project), vermeld dan welke componenten je nodig denkt te hebben. Ga je die zoeken op het internet? Ga je die zelf bouwen? Wat is de deadline voor de keuze?

Als je beslist hebt om iets te gebruiken dat je op het internet gevonden hebt, motiveer dan jouw keuze. Welke componenten heb je allemaal onderzocht? Wat waren de voordelen en nadelen? Waarom kies je voor een bepaalde library of hulpmiddel?

3.2 Structuur van de website

In het technisch ontwerp ga je al verder in detail uitwerken welke concrete pagina's je zal moeten maken en waar die zullen staan op de webserver. M.a.w. de folderstructuur wordt uitgetekend. Zijn er onderdelen die je zal hergebruiken, zoals bv de navigatiebalk. Hoe pakt je dit aan?

Aan de hand van flowcharts kan je verschillende processen beschrijven, bv iemand logt aan, maar is nog geen klant. Wat kan die doen? Wat moet die doen? Welke pagina wordt vervolgens getoond? Kan de gebruiker al een bestelling ingeven voordat hij/zij klant is?

Je zou een antwoord moeten hebben op deze en vele andere vragen VOORDAT je begint te implementeren!

Uit het functioneel ontwerp en het technisch ontwerp haal je **wat** je gaat maken en **hoe** je het gaat maken. Je hebt dus de scope van het project afgebakend.

4.1 Wie doet wat?

De volgende stap is een planning maken, rekening houdend met de deadlines. Je moet inderdaad meer doen dan alleen een website maken! Je moet aan dit document werken, je moet implementeren, je moet testen, je moet een presentatie maken, ...

Je maakt een overzicht van alle taken en deeltaken. Zorg je dat je de basis kan onderscheiden van de uitbreidingen. Wat moet minimum afgewerkt zijn? Wat zou een leuke gadget kunnen zijn? Als je tijd te kort zou komen, moet je de gadgets kunnen laten vallen maar moet de basis wel blijven werken!

Beslis wie wat zal doen!

Maak duidelijke afspraken en zorg dat het werk eerlijk verdeeld wordt!

4.2 Logboek

Julie houden het logboek up-to-date. In het logboek verwijs je naar alle (deel-)taken die je hierboven opgesomd hebt. Je bent eerlijk in het logboek. Hieruit zal geleidelijk aan blijken of de planning haalbaar is, realistisch was en/of bijgestuurd moet worden. Na verloop van tijd zal bekeken kunnen worden of de scope eventueel verminderd moet worden of uitgebreid mag worden.

5 Besluit

In een volgende fase voeg je een besluit toe aan dit document. Hierin vermeld je wat je geleerd hebt. Wat vond je moeilijk? Wat ging vlot? Wat is niet gelukt? Waar ben je tevreden over? Wat zou je de volgende keer anders aanpakken? ...