



3DYNAMISCH

3DYNAMISCH


NORCOM

Auteur:	Kevin Schuit
Cursus:	OOPHP, Bootstrap, MySQL, (JS) plugins
Datum:	12-11-2020
Update:	27-2-2021
Duur:	HBO – 5 dagen → Gevorderd MBO – 10 dagen → Beginner

Meester F. J. Haarmanweg 53A
4538 AN | Terneuzen

Info@3dynamisch.nl
+31 6 36409507

KvK: 69398496
BTW: NL857859729B01



1. Inhoud

1. Inhoud	2
2. Inleiding.....	4
2.1 Projectgrootte	4
2.2 Benodigheden	4
2.3 Klant	4
3. Project	5
3.1 Basisprincipe HTML.....	5
3.2 Basisprincipe CSS.....	5
3.3 Basisprincipe JAVASCRIPT	5
3.4 Basisprincipe PHP	5
3.5 Situatie	5
4. Aan te leveren producten	6
4.1 Wat wordt er verwacht?	6
5. Instructie NORCOM.....	6
5.1 Introductie NORCOM	6
5.2 Waarom MySQLi en niet PDO?	6
5.3 NORCOM - Beginner	7
5.4 NORCOM - Gevorderd.....	9
6. Cursusonderdeel 1 - Aanmaken van de database	10
6.1 Database koppelen	12
7. Definieert de pagina naam.....	14
7.1 Pagina naam aanmaken	14
7.2 Pagina naam meegeven	15
8. Index.php	16
9. classProduct.php – product uitlezen	17
9.1 get & set values.....	19
9.2 getProductList() & getProductById(\$pageld).....	22
10. Prijzen.php	25
11. Product.php	27
12. Offerte.php	31
13. Cursusonderdeel 2 - classUser.php.....	37
14. Register.php	39
15. Login.php.....	43

16.	Account.php	46
17.	Cursusonderdeel 3 – Producten CRUD	48
17.1	classProduct.php	50
	getPostValues()	50
	getGetValues()	51
	Save().....	52
	getNrOfProducten.....	53
	handleGetAction().....	54
	Update()	55
	Delete()	57
17.2	adminProductAanmaken.php	58
	Update form.....	63
	Tabel rijen	68
	Create formulier.....	69
18.	Algemene informatie	72

2. Inleiding

NORCOM is een telecombedrijf in de vaste en mobiele telefonie gevestigd in Friesland. 2 jaar geleden hadden zij bij 3DYNAMISCH een statische website afgenomen, echter deze willen zij graag gedeeltelijk dynamisch maken. Wordpress vinden zij niet zo prettig, maar gezien hun oude template, welk zij er nog mooi uit vinden zien is het voorstel gedaan om dit oude template dynamisch te maken.

2.1 Projectgrootte

Dit project wordt alleen uitgevoerd.

Duur van het project kun je op de voorpagina vinden.

Let op: afhankelijk van de cursus kan de grootte veranderd worden. Er zijn 2 versies van deze cursus. De beginner versie zal focussen op documentatie & structurele opbouw van de php en scripts. De gevorderde versie zal focussen op het innovatieve gebruik van de cursist zijn/haar inzicht met betrekking tot OOPHP, javascript & HTML/(S)CSS.

2.2 Benodigheden

- Laptop/desktop met internettoegang
- WAMP/MAMP/XAMP (Lokale server)
- Online server (indien de mail configuratie niet werkt)

2.3 Klant

De klant voor dit project is de instructeur die deze opdracht overgedragen heeft.

3. Project

Dit project gaat over het maken van een website voor een telecombedrijf. Let erop dat je die taken voor je rekening neemt waar je nog wat van kunt leren. Voor dit project wordt ervan uitgegaan dat je het basisprincipe van HTML/CSS, javascript & PHP kent.

3.1 Basisprincipe HTML

Je snapt de structuur en opbouw van een HTML-pagina, je begrijpt HTML tags en kunt deze toepassen.

3.2 Basisprincipe CSS

Je snapt de structuur van CSS, je snapt hoe je een stylesheet gebruikt en je begrijpt CSS termen.

3.3 Basisprincipe JAVASCRIPT

Je snapt de structuur van Javascript, je begrijpt hoe je deze kunt gebruiken en je snapt het principe van javascript plugins (of wilt dit graag leren).

3.4 Basisprincipe PHP

Je snapt het basisprincipe van PHP, je weet hoe je files include, je weet hoe je variabele dumpst en je snapt hoe je waarden verstuurt en terugkrijgt.

3.5 Situatie

NORCOM heeft zelf geen developers(backend,UI/UX,front-end) in dienst die dit voor hun kunnen uitvoeren. Hiervoor hebben zij weer contact opgenomen met het bedrijf dat hun website oorspronkelijk verzorgd heeft. De lead developer bij 3DYNAMISCH heeft jou aangewezen om deze opdracht te maken.

Alle documentatie betreft de planning is al gedaan, hoe de site eruit komt te zien maakt NORCOM niet uit, zolang het maar voldoet aan de initiële stijl en de website op tijd opgeleverd wordt.

De volgende onderdelen zijn van belang voor NORCOM.

- NORCOM klanten kunnen inloggen en hun accountpagina bezoeken
- NORCOM bezoekers kunnen een product bestellen, NORCOM krijgt hier een mail van.
- NORCOM wil de volgende elementen op hun site zien terugkomen
 - Intro sectie
 - Hoe werkt het?
 - De abonnementen
 - Bestel formulier
 - Testimonials van voorgaande klanten
- NORCOM wil zelf de producten aan kunnen maken via een formulier en indien nodig deze producten ook kunnen bewerken en/of verwijderen.
- NORCOM wil ook voor de telefoons een aparte pagina en deze zelf aan kunnen maken/verwijderen.

4. Aan te leveren producten

NORCOM zoals elke klant geeft niet om de documentatie, waar zij wel om geven is dat de website op tijd opgeleverd wordt. Voor het proces is het wellicht handig om documentatie te maken over de onderdelen die jij gaat verzorgen, dit zal aan het einde wellicht vragen kunnen voorkomen van de klant.

4.1 Wat wordt er verwacht?

- NORCOM website met daarin de Eisen verwerkt bij 2.5.
- Een oplevering van de NORCOM website aan de lead developer van 3DYNAMISCH.

5. Instructie NORCOM

5.1 Introductie NORCOM

Gefeliciteerd je bent nu bezig met de NORCOM-cursus, geleverd en gegeven door 3DYNAMISCH. Je hebt nu de bestanden van de NORCOM-website.

Afhankelijk of je de beginner of gevorderde cursus doet zul je een aantal php bestanden al hebben of niet.

De gevorderde cursus heeft als doel om de gegeven code uit de cursus te halen en zelf juist te implementeren. Het is ook mogelijk om een eigen variant op de gegeven cursus code te maken.

De beginner cursus heeft als doel om met baby stapjes de cursist OOPHP in combinatie met bootstrap en javascript bij te brengen. In deze OOPHP-cursus zit ook MySQLi inbegrepen om de data op te slaan.

5.2 Waarom MySQLi en niet PDO?

Hier zijn vaak vragen over, de één vindt PDO goed voor beginners, waar de ander MySQLi goed vindt. De cursus is gebaseerd op een MariaDB server welk met MySQLi werkt. Mocht het zo zijn dat je liever de PDO-versie doet, dan is het verstandig om dergelijke code te vragen bij de op dat moment beschikbare instructeur. Deze instructeur kan jou op dat moment tijdens de instructie helpen om dergelijke MySQLi code om te schrijven naar PDO.

Voor de gevorderde cursus geldt hetzelfde, echter zal hier de cursist zelf onderzoek moeten doen naar de correcte PDO-versie.

5.3 NORCOM - Beginner

De beginner cursus heeft vanuit deze structuur de volgende bestanden:

- assets
 - bootstrap
 - bootstrap.bundle.js
 - css
 - .sass-cache
 - theme.scss
 - theme.css
 - theme.css.map
 - theme.min.css
 - elements
 - about.php
 - doelgroepen.php
 - footer.php
 - gallery.php
 - head.php
 - hoewerkthet.php
 - nav.php
 - offerte.php
 - prijzen.php
 - scripts.php
 - stappen.php
 - welcome.php
 - font-awesome-4.7.0
 - [bevat mappen voor font-awesome, hier kijken we in deze cursus niet naar]
 - ico
 - [Hier kan je favicon eventueel in – moet deze dan nog wel linken]
 - img
 - template_img
 - Header.jpeg
 - step1.jpg
 - step2.jpg
 - step3.png
 - logo-3d.png
 -

- js
 - src
 - tools
 - sanitizer.js
 - alert.js
 - button.js
 - carousel.js
 - collapse.js
 - dropdown.js
 - index.js
 - modal.js
 - popover.js
 - scrollspy.js
 - tab.js
 - toast.js
 - tooltip.js
 - util.js
 - theme.js
 - theme.min.js
- model
 - classProduct.php
 - classUser.php
 - Database.php [globale verbinding]
- node_modules
 - popper.js [nodig voor bootstrap]
- plugins
- db
 - norcom_db.sql
- Account.php
- adminProductAanmaken.php
- index.php
- login.php
- product.php
- register.php

5.4 NORCOM - Gevorderd

De gevorderde cursus heeft de volgende bestanden **niet**:

- assets
 - elements
 - prijzen.php (prijzen sectie)
 - offerte.php (offerte sectie)
 - model
 - classProduct.php
 - classUser.php
 - Database.php
- Account.php
- adminProductAanmaken.php
- index.php
- login.php
- product.php
- register.php

6. Cursusonderdeel 1 - Aanmaken van de database

Deze cursus heeft de database file beschikbaar voor de cursist. De cursist kan deze database vinden onder het kopje 'assets/db/norcom_db.sql'.

LET OP: Het is raadzaam om dit soort code te kopiëren vanuit het bestand. In wordt kunnen sommige indelingen soms verloren gaan.

```
--
-- Database: `NORCOM`
--
-- -----
--
-- Tabelstructuur voor tabel `Product`
--
CREATE TABLE IF NOT EXISTS `product` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `abonnement` varchar(150) NOT NULL UNIQUE,
  `internetsnelheid` varchar(150) NOT NULL,
  `gbinternet` varchar(150) NOT NULL,
  `belminuten` varchar(150) NOT NULL,
  `sms` varchar(150) NOT NULL,
  `extrakosten` varchar(150) NOT NULL,
  `prijs` varchar(150) NOT NULL,
  `afbeelding` varchar(1024) NOT NULL,
  PRIMARY KEY (`id`))
ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Gegevens worden geëxporteerd voor tabel `product`
--
INSERT INTO `product` (`id`, `abonnement`, `internetsnelheid`,
`gbinternet`, `belminuten`, `sms`, `extrakosten`, `prijs`, `afbeelding`)
VALUES
('1', 'Onbeperkt', '4G', 'onbeperkt', 'onbeperkt', 'onbeperkt', 'nee',
'32.50', 'https://images.pexels.com/photos/47261/pexels-photo-
47261.jpeg?auto=compress&cs=tinysrgb&dpr=2&h=650&w=940'),
('2', 'Start 5GB', '4G', '5 GB', 'onbeperkt', 'onbeperkt', 'nee', '24.00',
'https://images.pexels.com/photos/248528/pexels-photo-
248528.jpeg?Bauto=compress&cs=tinysrgb&dpr=3&h=750&w=1260'),
('3', 'Pro 50 GB', '5G', '50 GB', '200', '200', 'nee', '29.50',
'https://images.pexels.com/photos/1092644/pexels-photo-
1092644.jpeg?auto=compress&cs=tinysrgb&dpr=2&h=650&w=940');

-- -----
--
-- Tabelstructuur voor tabel `mobiele_telefoons`
--
CREATE TABLE IF NOT EXISTS `mobiele_telefoons` (
  `mobiel_id` int(11) NOT NULL AUTO_INCREMENT,
  `mobiel` varchar(150) NOT NULL,
```

```

PRIMARY KEY (mobiel_id))
ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Gegevens worden geëxporteerd voor tabel `mobiele_telefoons`
--

INSERT INTO `mobiele_telefoons` (`mobiel_id`, `mobiel`) VALUES
('1', 'Iphone 11'),
('2', 'Iphone 11 Max Pro');

--
-- Tabelstructuur voor tabel `users`
--
CREATE TABLE IF NOT EXISTS `users` (
  `uid` int(11) NOT NULL AUTO_INCREMENT,
  `uname` varchar(30) DEFAULT NULL,
  `upass` varchar(50) DEFAULT NULL,
  `fullname` varchar(100) DEFAULT NULL,
  `uemail` varchar(70) DEFAULT NULL,
  PRIMARY KEY (uid)
  ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- INSERT INTO gegevens voor de default user (users tabel)
--

INSERT INTO `users` (`uid`, `uname`, `upass`, `fullname`, `uemail`) VALUES
('1',
'admin', 'e259dd0015e616aa9772524773820a9c', 'Admin', 'support@3dynamisch.nl')
;
```

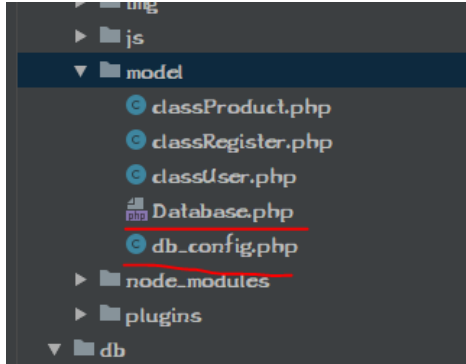
Ga naar: <http://localhost/phpmyadmin> en log in met de correcte gegevens.

- Klik op 'nieuw' (dit bevind zich linksboven bij de databases)
- Er verschijnen nu 2 velden en een database overzicht
- Vul bij Databasenaam **norcom** in en laat de engine op latin1_swedish_ci staan.
- Klik op starten
- Selecteer '**norcom**' in de tabel lijst en klik op het tabje **SQL**
- Plak nu de volledige **SQL** in de database file in de SQL query en klik rechts op de knop '**starten**'.
- Je zult nu zien dat je 3 tabellen hebt
 - Mobiele_telefoons
 - Product
 - Users

6.1 Database koppelen

Afhankelijk van de cursus kan het zijn dat dit heel makkelijk of iets lastiger word. Als nog zal hier onder alle code gezet worden met de dergelijke beschrijving.

Ga naar de model folder in **assets/model/** en open **Database.php**



In **Database.php** vind je 2 klassen. De klasse Database is een global variabele en bekend als een **singleton** klasse. Dit houdt in dat deze elke keer in een nieuwe functie aangeroepen moet worden. Deze klasse is gekoppeld aan **classProduct.php**

DB_con is een protected klasse. Dit betekent dat er bij elke call een aantal waarden moeten voldoen. Het gebruikt hiervan zal later behandeld worden. Deze klasse wordt gebruikt bij onze login, register en accountpagina en is gekoppeld aan **classUser.php**

12

Zet als volgt de **Database.php** in elkaar

```
<?php
/**
 * Create database connection
 * Include in klasse files
 */

// Definieer de database waarden
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_DATABASE', 'norcom');
```

Vervolgens zet je de **Database** klasse onder de net toegevoegde regel

```

/**
 * Class Database
 * singleton class - zelfde principe als wordpress (kan de database globaal
 door de gehele website oproepen).
 */

class Database {

    private static $db;
    private $connection;

    private function __construct() {
        $this->connection = new
        MySQLi(DB_SERVER,DB_USERNAME,DB_PASSWORD,DB_DATABASE);
    }

    function __destruct() {
        $this->connection->close();
    }

    public static function getConnection() {
        if (self::$db == null) {
            self::$db = new Database();
        }
        return self::$db->connection;
    }
}

```

Hierna zet je de **DB_con** klasse hieronder.

13

```

/**
 * Class DB_con
 * Beveiligde database connectie voor opslag van users of gevoelige data
 */
class DB_con {
    public $connection;
    function __construct() {
        $this->connection = new mysqli(DB_SERVER, DB_USERNAME,
        DB_PASSWORD,DB_DATABASE);

        if ($this->connection->connect_error) die('Database error -> ' .
        $this->connection->connect_error);
    }

    function ret_obj() {
        return $this->connection;
    }
}
?>

```

Nu koppel je de database aan je hoofd bestanden (= index.php en overige view bestanden in de root map). Deze connectie maak je op regel 2 van je bestand (na je <?php tag)

```

<?php
include 'assets/model/Database.php';

```

7. Definieert de pagina naam

7.1 Pagina naam aanmaken

Om een eenvoudige titel voor elke pagina in te stellen, kun je op basis van de pagina naam een **paginatitel** of **pagina tab titel** invoegen.

Hiervoor moeten wij het een en ander instellen.

- Navigeer naar **assets/elements/head.php**
- Voeg hier de volgende code toe tussen de meta tags en de link tags (boven de title tag)

```
<?php
// indexeert de pagina naam wanneer pagename defined is bij nav.
if (!empty($_GET['pagename'])) {
    // als de pagename gevonden is, koppel deze aan variabele
    $pagename.
    $pagename = $_GET['pagename'];
} else {
    // als er geen pagename gevonden is, geef de pagename aan als
    Home.
    $pagename = 'Home';
}
// Vul de pagename bij de titel in
?>
```

Nu kunnen wij bijna de pagina naam mee geven wanneer deze via de url meegegeven word. Roep nu in de title tag de **\$pagename** aan. Een voorbeeld hiervan is het volgende:

```
<title>NORCOM | <?= $pagename ?></title>
```

7.2 Pagina naam meegeven

Nu wij het moeilijke deel achter de rug hebben is het tijd om de pagina naam zowaar mee te geven. Ga naar **assets/elements/nav.php**

Je menu zal er ongeveer zo uit zien, afhankelijk van de cursus zijn er een aantal al meegegeven of er is er nog geen één meegegeven. Deze versie is eindversie, hierdoor zijn alle waarden al ingevuld.

```
<!-- Navbar navigation: Left -->
<ul class="navbar-nav mr-auto">
  <li class="nav-item">
    <a class="nav-link" href="index.php?pagename=Home">Home</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="index.php?pagename=Home#over-tale-4">Over ons</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="index.php?pagename=Home#prijzen">Abonnementen</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="adminProductAanmaken.php?pagename=Product%20Aanmaken">Product Aanmaken</a>
  </li>

  <li class="nav-item">
    <a class="nav-link" href="login.php?pagename=Inloggen#login">Inloggen</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="index.php?pagename=Home#stappen">Hoe werkt het</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="index.php?pagename=Home#offerte">Bestelling plaatsen</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="account.php?pagename=Mijn%20Account">Account</a>
  </li>
</ul>
```

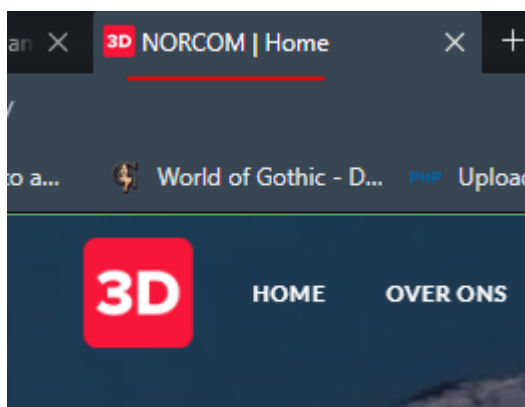
15

Hoe geven wij nu de pagina naam mee? Het volgende moet je aan de link toevoegen

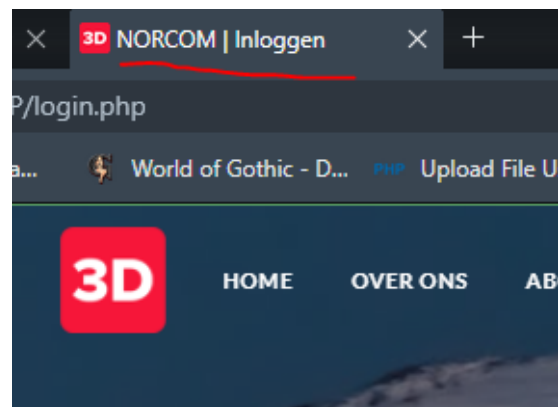
```
index.php?pagename=Home
```

Dit zorgt er nu voor dat wanneer jij op deze link klikt, jij naar de index.php pagina gaat, maar deze pagina vervolgens Home heet in de titel balk of waar je **\$pagename** ook aanroept. Bij de **product aanmaken** sectie zal hierop teruggekomen worden hoe je vanuit dezelfde pagina meerdere pagina's kan maken op basis van **\$pagename**.

Resultaat:



Meester F. J. Haarmannweg 53A
4538 AN | Terneuzen



Info@3dynamisch.nl
+31 6 36409507

KvK: 69398496
BTW: NL857859729B01

8. Index.php

Zoals hieronder getoond ziet index.php eruit als een korte file. Dit komt doordat alle delen die hier weer geinclude zijn, als elementen zijn gemaakt. Hierdoor is ook het bewerken een stuk overzichtelijker. In al onze template pagina's laden wij elementen in, tenzij het onderdeel pagina specifiek is. Dit is bijvoorbeeld het geval bij onze **adminProductAanmaken.php** pagina.

Ook include je zoals hieronder aangegeven de desbetreffende pagina klassen, lijsten en modellen boven in de pagina. Op de volgende pagina, word uitgelegd hoe de product klasse aangemaakt wordt.

```
<?php
include 'assets/model/Database.php';
include 'assets/model/classProduct.php';
$producten = new classProduct();
$product_list = $producten->getProductList();
$mobiell_list = $producten->getMobiellList();
?>

<?php include 'assets/elements/head.php'?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php'?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php'?>

<!-- OVER NOCOM
===== -->
<?php include 'assets/elements/about.php'?>

<!-- Divider met logo -->
<div class="separator">
  
</div>

<!-- DISCOVER DE STAPPEN
===== -->

<?php include 'assets/elements/stappen.php'?>

<br><br><br>
<!-- Divider met logo -->
<div class="separator">
  
</div>

<!-- PRIJZEN
===== -->

<?php include 'assets/elements/prijzen.php'?>

<br><br>
<!-- Divider met logo -->
<div class="separator">
  
</div>

<!-- DOELGROEPEN
===== -->
```


9. classProduct.php – product uitlezen

Een klasse, wat houdt dit in? Een klasse file is eigenlijk een verzameling van functies welk de developer kan gebruiken in zijn/haar website om zo bepaalde systemen (vaak database gerelateerd) te laten werken.

Je noemt de klasse file altijd class[NaamVanHetOnderwerp] in dit geval zijn wij hier bezig met producten, dus werken wij hier met de product klasse. Een klasse definieer je als volgt:

```
<?php
class classProduct
{
}
}
```

Elke klasse heeft ook attributen, deze definieer je aan het begin in je klasse.

```
<?php
class classProduct
{ //Begin van de klasse
/*
 * Definieer alle values als private attributen
 * -----
 * START
 */
    private $id = "";
    private $abonnement = "";
    private $internetSnelheid = "";
    private $gb = "";
    private $belminuten = "";
    private $sms = "";
    private $kosten = "";
    private $prijs = "";
    private $afbeelding = "";
/*
 * -----
 * END
 */
} // Einde van de klasse
```

17

Vervolgens zal het voorkomen dat je in je functies een tabel aan moet roepen. Ook om constant te moeten onthouden hoe jij je tabel moet typen, kunnen wij dit eenvoudiger in een functie verwerken.

Hiervoor doe je het volgende.

```
/**
 * @return string getProductTableName (product)
 */
private function getProductTableName () {
    return $table = 'product';
}
```

Het vorige stukje code voeg je toe aan de al inmiddels gemaakte klasse. Het ziet er dan als volgt uit:

```
<?php
class classProduct
{//Begin van de klasse
/*
 * Definieer alle values als private attributen
 * -----
 * START
 */
    private $id = "";
    private $abonnement = "";
    private $internetSnelheid = "";
    private $gb = "";
    private $belminuten = "";
    private $sms = "";
    private $kosten = "";
    private $prijs = "";
    private $afbeelding = "";
/*
 * -----
 * END
 */

/*-----
GET TABLENAME START
*/

/**
 * @return string getProductTableName (product)
 */
private function getProductTableName () {
    return $table = 'product';
}

/*-----
GET TABLENAME END
*/

} // Einde van de klasse
```

Het is belangrijk dat je alle functies binnen je klasse toevoegt, anders krijg je error meldingen en/of niet werkende functies. Indien je meerdere klasse in 1 klasse file maakt, is het handig om op te letten dat je de functie in de juiste klasse zet of vanuit de front-end files, de juiste klasse linkt aan de juiste functie.

Let op: Wanneer je een functie maakt, zorg ervoor dat je begint met een kleine letter en altijd elk nieuw woord met een hoofdletter aaneenvolgend start. Dus een functie als **get table name** definieer je als **getTable_name()**. Mocht het voorkomen dat je een waarde of waarden mee moet geven, dan plaats je deze tussen de (). Hier komt bij de **getProductById(\$pageId)** functie meer over.

In dit geval zijn wij nu bezig met het ophalen van producten. Hierdoor slaan wij het stuk over de **getPostValues()** voor nu over. Dit komt terug bij het aanmaken van producten.

9.1 get & set values

Om database waarden op te halen heb je gets en sets nodig. Deze gets en sets corresponderen met de **private variabelen** aan het begin van de klasse. Voor elke private variabele is een get en set functie.

In dit geval zien de sets er als volgt uit:

```
/*-----  
Product tabel sets START  
*/  
  
/**  
 * @param Set $id  
 */  
public function setProductId( $id ){  
    if ( is_int(intval($id) ) ){  
        $this->id = $id;  
    }  
}  
  
/**  
 * @param Set $abonnement  
 */  
public function setAbonnement( $abonnement ){  
    if ( is_string( $abonnement ) ){  
        $this->abonnement = trim($abonnement);  
    }  
}  
  
/**  
 * @param Set $internetSnelheid  
 */  
public function setInternetsnelheid( $internetSnelheid ){  
    if ( is_string( $internetSnelheid ) ){  
        $this->internetSnelheid = trim($internetSnelheid);  
    }  
}  
  
/**  
 * @param Set $gb  
 */  
public function setGbInternet( $gb ){  
    if ( is_string( $gb ) ){  
        $this->gb = trim($gb);  
    }  
}
```

```
/**
 * @param Set $belminuten
 */
public function setBelminuten($belminuten ){
    if ( is_string( $belminuten )){
        $this->belminuten = trim($belminuten);
    }
}

/**
 * @param Set $sms
 */
public function setSms($sms ){
    if ( is_string( $sms )){
        $this->sms = trim($sms);
    }
}

/**
 * @param Set $kosten
 */
public function setExtraKosten($kosten ){
    if ( is_string( $kosten )){
        $this->kosten = trim($kosten);
    }
}

/**
 * @param Set $prijs
 */
public function setPrijs($prijs ){
    if ( is_string( $prijs )){
        $this->prijs = trim($prijs);
    }
}

/**
 * @param Set $afbeelding
 */
public function setAfbeelding($afbeelding ){
    if ( is_string( $afbeelding )){
        $this->afbeelding = trim($afbeelding);
    }
}

/*-----
Product tabel sets END
*/
```

De gets zien er als volgt uit:

```
/*-----  
Product tabel gets START  
*/  
  
/**  
 * @return get ProductId  
 */  
public function getProductId() {  
    return $this->id;  
}  
  
/**  
 * @return get Abonnement  
 */  
public function getAbonnement() {  
    return $this->abonnement;  
}  
  
/**  
 * @return get InternetSnelheid  
 */  
public function getInternetSnelheid() {  
    return $this->internetSnelheid;  
}  
  
/**  
 * @return get GbInternet  
 */  
public function getGbInternet() {  
    return $this->gb;  
}  
  
/**  
 * @return get Belminuten  
 */  
public function getBelminuten() {  
    return $this->belminuten;  
}  
  
/**  
 * @return get Sms  
 */  
public function getSms() {  
    return $this->sms;  
}  
  
/**  
 * @return get ExtraKosten  
 */  
public function getExtraKosten() {  
    return $this->kosten;  
}
```

```

/**
 * @return get Prijs
 */
public function getPrijs() {
    return $this->prijs;
}

/**
 * @return get Afbeelding
 */
public function getAfbeelding() {
    return $this->afbeelding;
}

/*-----
Product tabel gets END
*/

```

9.2 getProductList() & getProductById(\$pageld)

Om alle producten op te halen moeten we deze uit de database halen. Dit doen wij met de volgende functie.

Let op dat in deze functie \$conn gebruikt is om een database connectie op te zetten. Mocht je dit niet doen dan haalt deze functie niks op. Mocht je een globale connectie maken, dan dien je dit boven in de pagina te doen (meer hierover in de **classUser.php**)

In deze cursus is hier een query call gedaan met

```

$query = "SELECT * FROM `".$this->getProductTableName()." ORDER BY id";
$result = $conn->query($query);

```

Een alternatief is

```

$result = $conn->query("SELECT * FROM `".$this->getProductTableName()." ORDER BY id");

```

Het alternatieve format word bijvoorbeeld bij de update en save functie toegepast.

```

/**
 * @return array GetProductList (List of product items)
 */
public function getProductList() {

    // Call database = $global wpdb in wordpress
    $conn = Database::getConnection();

    $return_array = array();

    $query = "SELECT * FROM `".$this->getProductTableName()."` ORDER BY
id";
    $result = $conn->query($query);

    // For all database results:
    foreach ($result as $idx => $array) {
        // Nieuw object
        $product = new classProduct();
        // Set info
        $product->setProductId($array['id']);
        $product->setAbonnement($array['abonnement']);
        $product->setInternetSnelheid($array['internetsnelheid']);
        $product->setGbInternet($array['gbinternet']);
        $product->setBelminuten($array['belminuten']);
        $product->setSms($array['sms']);
        $product->setExtraKosten($array['extrakosten']);
        $product->setPrijs($array['prijs']);
        $product->setAfbeelding($array['afbeelding']);

        // Add new object to return array.
        $return_array[] = $product;
    }
    return $return_array;
}

```

In deze functie word de waarde in een set gezet en deze word vervolgens via de sets en gets aan een get gekoppeld. Hoe we deze functie implementeren komen wij straks op.

Hieronder maken wij direct nog een functie. Deze functie toont een product op basis van het gegeven id. Ook hebben wij hierin een fallback gezet mocht er een id niet zijn. Deze functie ziet u op de volgende pagina.

```

/**
 * @param $pageId
 * @return array
 */
public function getProductById($pageId) {

    // Call database = $global wpdb in wordpress
    $conn = Database::getConnection();

    $query = "SELECT * FROM `" . $this->getProductTableName() . "` WHERE id
= " . $pageId . " ";
    $result = $conn->query($query);
    //Setting var as an array
    //Database query
    // Loop through producten
    foreach ($result as $array) {
        // New object
        $product = new classProduct();
        // Set all info
        $product->setProductId($array['id']);
        $product->setAbonnement($array['abonnement']);
        $product->setInternetSnelheid($array['internetsnelheid']);
        $product->setGbInternet($array['gbinternet']);
        $product->setBelminuten($array['belminuten']);
        $product->setSms($array['sms']);
        $product->setExtraKosten($array['extrakosten']);
        $product->setPrijs($array['prijs']);
        $product->setAfbeelding($array['afbeelding']);

        // Add new object to return array.
        $return_array[] = $product;
    }
    // check of return array niet leeg is, zo ja, laad data. Zo niet, toon
    error 1.
    if (!empty($return_array)) {
        // Return array
        return $return_array;
    } else {
        echo '<div class="center col-md-12">
        <p class="alert alert-danger" role="alert">Het gegeven id
        bevat geen product informatie!</p></div>';
    }
}

```

In principe doen wij hier hetzelfde als bij de vorige functie, echter vragen wij hier om het specifieke overeenkomende ID (**\$pageId**). Nu laad alleen de informatie die bij dit ID hoort uit. Mocht de **\$return_array** leeg zijn, dan tonen wij de echo

```

echo '<div class="center col-md-12">
    <p class="alert alert-danger" role="alert">Het gegeven id bevat geen
    product informatie!</p></div>';

```

Hierin zijn ook bootstrap klassen gebruikt, dit zorgt ervoor dat onze error direct de stijl van het thema pakt dat wij gebruiken.

10. Prijzen.php

Assets/elements/prijzen.php is een element van **index.php**. Hierdoor hoeven wij hier niet een klasse aan te maken, maar doen wij dit direct in de **index.php** file.

Zet het volgende direct onder de **Database** inclusie boven in de **index.php**

```
include 'assets/model/classProduct.php';
$producten = new classProduct();
$product_list = $producten->getProductList();
```

Nu kunnen wij alle functies van de product klasse in de index.php file gebruiken.

Terug naar prijzen.php, hier willen wij elk ingevoegd product kunnen tonen. Dus mocht iemand er 5 hebben, dan willen wij die 5 zien. Mocht iemand er meer maken, dan willen wij er meer zien etc...

Nu moeten wij in onze container het volgende onder onze mini intro toevoegen.

```
<div class="row">
    <?php
        // Voor elk $product_list hernoemt als $abonnement, voer de actie
        tussen de { } uit.
        foreach ($product_list as $abonnement){ ?>
            <div class="col-md-4 top">
                
                <h3 class="text-center"><?= $abonnement->
getAbonnement();?></h3>
                <p class="text-center">
                    € <?= $abonnement->getPrijs();?>
                </p>
                <div class="center">
                    <a href="<?= 'product.php?id=' . $abonnement->
getProductId(). '&pagename=Producten';?>" class="button2">Meer
informatie</a>
                </div>
            </div>
        <?php }
    ?>
</div>
```

25

In het kort wat doet dit? Wij hebben in index.php **\$product_list** aangemaakt, deze hernoemen wij nu als **\$abonnement**. Hierin zeggen wij dat voor elk abonnement in de verkregen lijst, wij een nieuwe div met informatie willen tonen.

De **getAfbeelding** laadt een volledig http/https link uit en hierdoor wordt deze weergegeven op de pagina.

getAbonnement, geeft ons de naam van het abonnement.

getPrijs, toont ons de prijs van het abonnement.

Om te zorgen dat wij naar ons single product kunnen wat wij hierna behandelen hebben wij in onze link, welk ook de **pagename** bevat, het **getProductId** nodig.

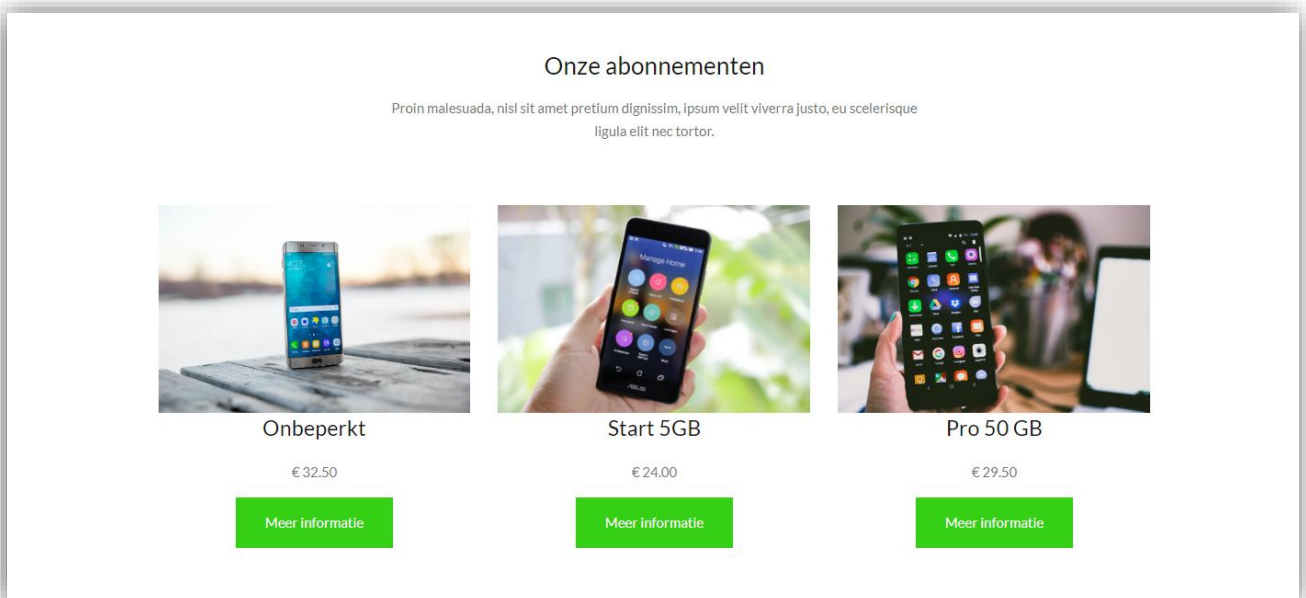
De prijzen sectie zal er als volgt uit zien:

```
<section class="section menu-section" id="prijzen"></section>
<section class="section section_menu section_border_bottom">
  <div class="container">
    <div class="row">
      <div class="col">
        <!-- Heading -->
        <h2 class="section_heading text-center">
          Onze abonnementen
        </h2>
        <p class="section_subheading text-center">
          Proin malesuada, nisl sit amet pretium dignissim, ipsum velit viverra justo, eu scelerisque ligula elit nec tortor.
        </p>
      </div>
    </div> <!-- / .row -->

    <div class="row">
      <?php
        // Voor elk $product_list hernoemt als $abonnement, voer de actie tussen de { } uit.
        foreach ($product_list as $abonnement) { ?>
          <div class="col-md-4 top">
            
            <h3 class="text-center"><?=' $abonnement->getAbonnement();?>'>
            <p class="text-center">
              € <?=' $abonnement->getPrijs();?>'
            </p>
            <div class="center">
              <a href="<?=' product.php?id=' . $abonnement->getProductId() . '&#39; $pagename=Producten';?>" class="button2">Meer informatie</a>
            </div>
          </div>
        <?php }
      <?>
    </div> <!-- / .container -->
  </section>
```

Het resultaat ziet er als volgt uit:

26



11. Product.php

Op deze pagina gaan wij enkele producten uitladen, gebaseerd op het meegegeven ID. Mocht deze pagina er nog niet zijn, kopieer de index.php en maak vanuit hier de product.php pagina.

Bovenin de pagina (mocht dit niet correct in je index.php staan) zet je

```
<?php
include 'assets/model/Database.php';
// Laad classProduct in
include 'assets/model/classProduct.php';
// maak $producten de nieuwe classProduct
$producten = new classProduct();
$product_list = $producten->getProductList();
?>
```

Mocht je nog andere klassen hebben, dan zijn deze niet nodig op deze pagina.

Vervolgens laad je de head, nav, welcome & about secties in. Dit doe je als volgt:

```
<?php include 'assets/elements/head.php'?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php'?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php'?>

<!-- OVER NORCOM
===== -->
<?php include 'assets/elements/about.php'?>

<!-- Divider met logo -->
<div class="separator">
    
</div>
```

Nu kunnen wij ons focussen op onze producten sectie. Hieronder zal de volledige code gezet worden, na deze code zal de uitleg verschaft worden over het **\$pageId** gedeelte.

```
<!-- SINGLE PRODUCT
===== -->
<section class="section menu-section" id="prijzen"></section>
<section class="section section_menu section_border_bottom">
    <div class="container">
        <div class="row">
            <?php

                //Haal het ID op
                $pageId = $_GET['id'];
                //Check of het id leeg is, zo ja toon alternatieve melding, zo
                //niet toon product.
                if(!empty($pageId)){
                    //Haal enkel product op, gebaseerd op ID
                    $single_product = $producten->getProductById($pageId);

                    // check of single product niet leeg is, zo ja, voer functie
                    // uit, zo niet voer error 1 uit.
                    if(!empty($single_product)){
                        foreach ($single_product as $abonnement){ ?>
                            <div class="col-md-12">
                                
                                <h3 class="text-center"><?=$abonnement-
                                >getAbonnement();?></h3>
                                <table class="center table table-striped table-hover">
                                    <tr>
                                        <td>Internet snelheid</td>
                                        <td><?=$abonnement-
                                        >getInternetSnelheid();?></td>
                                    </tr>
                                    <tr>
                                        <td>gb Internet</td>
                                        <td><?=$abonnement-
                                        >getGbInternet();?></td>
                                    </tr>
```

```

        <tr>
            <td>Sms</td>
            <td><?= $abonnement->getSms();?></td>
        </tr>
        <tr>
            <td>Extra Kosten</td>
            <td><?= $abonnement->getExtraKosten();?></td>
        </tr>
        <tr>
            <td>Prijs</td>
            <td><?= $abonnement->getPrijs();?></td>
        </tr>
    </table>
    <div class="center">
        <a href="index.php#offerte"
class="button2">Bestelling plaatsen</a>
    </div>
</div>
<?php }
}else{
    // Toon return error 1 in classProduct.php
}
}else{
    // Toon error wanneer product id 0 of niet gegeven is.
    echo '<div class="center col-md-12">
        <p class="alert alert-danger" role="alert">Er zijn op dit
moment geen producten aanwezig!</p></div>';
    } ?>
</div>
</div> <!-- /.container -->
</section>

```

En hierna laden wij de footer en scripts nog in

```

<!-- FOOTER
===== -->
<?php include 'assets/elements/footer.php'?>
<!-- JAVASCRIPT
===== -->
<?php include 'assets/elements/scripts.php'?>

</body>
</html>

```



Resultaat:



Onbeperkt

Internet snelheid	4G
gb Internet	onbeperkt
Sms	onbeperkt
Extra Kosten	nee
Prijs	32.50

Bestelling plaatsen

30

Meester F. J. Haarmanweg 53A
4538 AN | Terneuzen

Info@3dynamisch.nl
+31 6 36409507

KvK: 69398496
BTW: NL857859729B01

Deze code laadt net als bij de prijzen.php pagina de waarden uit van het product. Het verschil is dat wij hier naar een enkel product kijken. Hoe doen we dit?

Wanneer je de knop op de home goed ingesteld hebt, ga je naar een product met in de URL het id van het product.

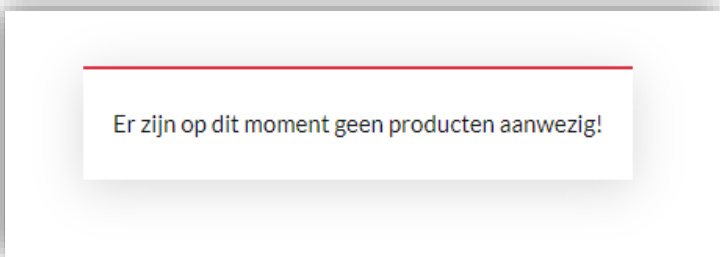
De code hieronder filters dit ID uit de url en vraagt het als item op om weer te geven.

```
//Haal het ID op
$pageId = $_GET['id'];
//Check of het id leeg is, zo ja toon alternatieve melding, zo niet toon product.
if(!empty($pageId)){
//Haal enkel product op, gebaseerd op ID
$single_product = $producten->getProductById($pageId);
```

Wanneer er geen ID meegegeven is, toont de pagina de volgende error.

```
// Toon error wanneer product id 0 of niet gegeven is.
echo '<div class="center col-md-12">
<p class="alert alert-danger" role="alert">Er zijn op dit moment geen
producten aanwezig!</p></div>';
```

Resultaat:



Er zijn op dit moment geen producten aanwezig!

31

12. Offerte.php

De offerte sectie, deze is vaak lastig voor developers om te testen wanneer zij de mail configuratie niet voltooid krijgen bij hun localhost. Je kan hulp aan je instructeur vragen hiervoor. Afhankelijk van de localhost & mail is dit eenvoudig of lastig in te stellen. De meest eenvoudige is xamp. Mocht het niet lukken, geen probleem. Dit formulier is bij meerdere hosters gezet en deze werkt altijd. Mocht je het in actie willen zien, vraag dan aan de instructeur of hij een testdomein op kan zetten. De code hieronder is in stukken gezet, zodat dit eenvoudiger toegelicht kan worden.

Startend met de mail & checks. Zorg dat je bij deze checks alleen de velden in vult die gecheckt moeten worden. De antiflood zorgt ervoor dat iemand niet kan spam klikken. Zorg dat je e-mail overeen komt met het domein waar het op staat.

```
<!-- HTML e-mail formulier -->
<?php
/*
 * *****
 *
 * CONTACT FORMULIER
 *
 *
 * Author: Kevin Schuit
 * Datum: 10 september 2010
 *
 *
 * Pas het e-mail adres aan
 * bij $mail Ontv en upload
 * het naar je webserver..
 * ***** */
$recipients = array("info@jouwdomein.nl"); // <<<----- voer jouw e-
mailadres hier in!

// E-mailadres van de ontvanger
$mail_ontv = implode(',', $recipients);
// Speciale checks voor naam en e-mailadres
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    // voornaam controle
    if (empty($_POST['voornaam']))
        $voornaam_fout = 1;
    // achternaam controle
    if (empty($_POST['achternaam']))
        $achternaam_fout = 1;
    // tel controle
    if (empty($_POST['tel']))
        $tel_fout = 1;
    // abonnement controle
    if (empty($_POST['abonnement']))
        $abonnement_fout = 1;
    // mobiel controle
    if (empty($_POST['mobiel']))
        $mobiel_fout = 1;
    // e-mail controle
    if (function_exists('filter_var') && !filter_var($_POST['mail'],
FILTER_VALIDATE_EMAIL))
        $email_fout = 1;
    // antiflood controle
    if (!empty($_SESSION['antiflood'])) {
        $seconde = 20; // 20 seconden voordat dezelfde persoon nog een keer
een e-mail mag versturen
        $tijd = time() - $_SESSION['antiflood'];
        if ($tijd < $seconde)
            $antiflood = 1;
    }
}
```


Hier checken we eerst of alles daadwerkelijk is ingevuld, zo niet geven wij een error.

```
// Kijk of alle velden zijn ingevuld - naam mag alleen uit letters bestaan
en het e-mailadres moet juist zijn
if (($_SERVER['REQUEST_METHOD'] == 'POST' && (!empty($antiflood) ||
empty($_POST['voornaam']) || !empty($voornaam_fout) ||
empty($_POST['achternaam']) || !empty($achternaam_fout) ||
empty($_POST['tel']) || !empty($tel_fout) || empty($_POST['abonnement']) ||
!empty($abonnement_fout) || empty($_POST['mobiel']) || !empty($mobiel_fout)
|| empty($_POST['mail']) || !empty($email_fout) ||
empty($_POST['onderwerp']))) || $_SERVER['REQUEST_METHOD'] == 'GET') {
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        if (!empty($voornaam_fout))
            echo '<p class="alert alert-danger" role="alert">Uw voornaam is
niet ingevuld.</p>';
        elseif (!empty($achternaam_fout))
            echo '<p class="alert alert-danger" role="alert">Uw achternaam
is niet ingevuld.</p>';
        elseif (!empty($tel_fout))
            echo '<p class="alert alert-danger" role="alert">Uw
telefoonnummer is niet ingevuld.</p>';
        elseif (!empty($abonnement_fout))
            echo '<p class="alert alert-danger" role="alert">Uw abonnement
is niet ingevuld.</p>';
        elseif (!empty($mobiel_fout))
            echo '<p class="alert alert-danger" role="alert">Uw mobiel is
niet ingevuld.</p>';
        elseif (!empty($email_fout))
            echo '<p class="alert alert-danger" role="alert">Uw e-mailadres
is niet juist ingevuld.</p>';
        elseif (!empty($antiflood))
            echo '<p class="alert alert-danger" role="alert">U mag slechts
&acute;&acute;n bericht per ' . $seconde . ' seconde versturen.</p>';
        else
            echo '<p class="alert alert-danger" role="alert">U bent uw
naam, e-mailadres, onderwerp of bericht vergeten in te vullen.</p>';
    }
}
```

Hier maken wij het formulier aan, ook hebben wij hier weer onze foreach van waar wij onze product informatie uitladen.

```
// HTML e-mail formlier
echo '<form id="contact" method="post" action="' .
$_SERVER['REQUEST_URI'] . '" enctype="multipart/form-data"/>';
echo '<div class="form-inline">';
echo '<input type="text" class="form-control half" id="voornaam"
name="voornaam" placeholder="Voornaam" value="" .
(isset($_POST['voornaam'])) ? htmlspecialchars($_POST['voornaam']) : '' .
'" /><br />';
echo '<input type="text" class="form-control half" id="achternaam"
name="achternaam" placeholder="Achternaam" value="" .
(isset($_POST['achternaam'])) ? htmlspecialchars($_POST['achternaam']) : ''
. '" /><br />';
echo '</div>';
echo '<div class="form-inline">';
echo '<input type="text" class="form-control half" id="mail"
name="mail" placeholder="E-mail" value="" . (isset($_POST['mail'])) ?
htmlspecialchars($_POST['mail']) : '' . '" /><br />';
echo '<input type="text" class="form-control half" id="tel" name="tel"
placeholder="Telefoonnummer" value="" . (isset($_POST['tel'])) ?
htmlspecialchars($_POST['tel']) : '' . '" /><br />';
echo '</div>';
echo '<div class="form-inline">';
echo '<select name="abonnement" class="form-control half" required>';
foreach($product_list as $product_obj){
    echo '<option value="'. $product_obj->getAbonnement() . '>'. $product_obj->getAbonnement() . '</option>';
}
echo '</select>';
echo '<select name="mobiel" class="form-control half" required>';
foreach($mobiel_list as $mobiel_obj){
    echo '<option value="'. $mobiel_obj->getMobiel() . '>'. $mobiel_obj->getMobiel() . '</option>';
}
echo '</select>';
echo '<input type="hidden" class="form-control half" id="onderwerp"
name="onderwerp" placeholder="Onderwerp" value="Nieuw abonnement aanvraag"
/><br />';
echo '</div>';
echo '<input class="button button1" type="submit" name="submit" value="
Bestellen " />';
echo '</form>';
} // versturen naar
```

Hier op de volgende pagina bouwen wij eerst ons formulier op en laden wij alle waarden in, hierna checken wij de content met een spambot protectie script, waarna wij het formulier verzenden. Afhankelijk van de server & correcte mail instelling, geeft het formulier een error of succes melding.

```

else {

    // set datum
    $datum = date('d/m/Y H:i:s');

    $inhoud_mail .= "Naam: " . htmlspecialchars($_POST['voornaam']) . ' '.
    htmlspecialchars($_POST['achternaam']) . "\n";
    $inhoud_mail .= "Telefoonnummer: " . htmlspecialchars($_POST['tel']) .
    "\n";
    $inhoud_mail .= "E-mail adres: " . htmlspecialchars($_POST['mail']) .
    "\n";
    $inhoud_mail .= "Gekozen abonnement: " .
    htmlspecialchars($_POST['abonnement']) . "\n";
    $inhoud_mail .= "Gekozen telefoon: " .
    htmlspecialchars($_POST['mobiel']) . "\n";

    $headers = 'From: ' . htmlspecialchars($_POST['voornaam']) . ' '.
    htmlspecialchars($_POST['achternaam']) . ' <' . $_POST['mail'] . '>';

    $headers = stripslashes($headers);
    $headers = str_replace('\n', '', $headers); // Verwijder \n
    $headers = str_replace('\r', '', $headers); // Verwijder \r
    $headers = str_replace("\"", "\\\"", str_replace("\\\"", "\\\"",
    $headers)); // Slashes van quotes

    $_POST['onderwerp'] = str_replace('\n', '', $_POST['onderwerp']); //
    Verwijder \n
    $_POST['onderwerp'] = str_replace('\r', '', $_POST['onderwerp']); //
    Verwijder \r
    $_POST['onderwerp'] = str_replace("\"", "\\\"", str_replace("\\\"",
    "\\\"", $_POST['onderwerp'])); // Slashes van quotes

    if (mail($mail_ontv, $_POST['onderwerp'], $inhoud_mail, $headers)) {
        // zorg ervoor dat dezelfde persoon niet kan spammen
        $_SESSION['antiflood'] = time();



        echo '<script> alert("Uw offerte aanvraag is succesvol verzonden,
er zal zo spoedig mogelijk contact met u worden opgenomen.")</script>
        <script>window.location.href = "index.php#offerte";</script>';
    } else {
        echo '<h1>Verzending mislukt</h1>

        <p><b>Onze excuses.</b> Het contactformulier kon niet verzonden
worden.</p>';
    }
}

```

Bestelling plaatsen

In aliquam sem ut lorem accumsan accumsan quis sed mi.

Voornaam	Achternaam
E-mail	Telefoonnummer
Abonnement 	Telefoon 

Bestellen

13. Cursusonderdeel 2 - classUser.php

Deze klasse wordt puur gebruikt voor het login script. Net als bij onze product klasse starten wij hier met onze klasse, echter maken wij nu eerst contact met de database, hierna kunnen wij zonder de database te herroepen in elke functie onze connectie gebruiken. Hieronder is het gehele script. Tijdens de cursus les zal deze uitgelegd worden.

```
<?php

class User{
    protected $db;
    public function __construct(){
        $this->db = new DB_con();
        $this->db = $this->db->ret_obj();
    }

    /*** for registration process ***/

    public function reg_user($name,$username,$password,$email){
        //echo "k";

        $password = md5($password);

        //checkt of gebruikersnaam en email beschikbaar zijn
        $query = "SELECT * FROM users WHERE uname='$username' OR
uemail='$email'";

        $result = $this->db->query($query) or die($this->db->error);

        $count_row = $result->num_rows;

        //if gebruikersnaam beschikbaar is, insert into database

        if($count_row == 0){
            $query = "INSERT INTO users SET uname='$username',
upass='$password', fullname='$name', uemail='$email'";

            $result = $this->db->query($query) or die($this->db->error);

            return true;
        }
        else{return false;}
    }

    /*** for login process ***/
    public function check_login($emailusername, $password){
        $password = md5($password);

        $query = "SELECT uid from users WHERE uemail='$emailusername' or
uname='$emailusername' and upass='$password'";

        $result = $this->db->query($query) or die($this->db->error);

        $user_data = $result->fetch_array(MYSQLI_ASSOC);
```

```

$count_row = $result->num_rows;

if ($count_row == 1) {
    $_SESSION['login'] = true; // deze login gebruikt het sessie
ding
    $_SESSION['uid'] = $user_data['uid'];
    return true;
}

else{return false;}

}

public function get_fullname($uid){
    $query = "SELECT fullname FROM users WHERE uid = $uid";

    $result = $this->db->query($query) or die($this->db->error);

    $user_data = $result->fetch_array(MYSQLI_ASSOC);
    echo $user_data['fullname'];
}

public function get_uname($uid){
    $query = "SELECT uname FROM users WHERE uid = $uid";

    $result = $this->db->query($query) or die($this->db->error);

    $user_data = $result->fetch_array(MYSQLI_ASSOC);
    echo $user_data['uname'];
}

public function get_uemail($uid){
    $query = "SELECT uemail FROM users WHERE uid = $uid";

    $result = $this->db->query($query) or die($this->db->error);

    $user_data = $result->fetch_array(MYSQLI_ASSOC);
    echo $user_data['uemail'];
}

/**/ starting the session ***/
public function get_session(){
    return $_SESSION['login'];
}

public function user_logout() {
    $_SESSION['login'] = FALSE;
    unset($_SESSION);
    session_destroy();
}
}

```

14. Register.php

Op deze pagina maken wij ons registratieformulier. Mocht je deze pagina nog niet hebben, dan kun je de product.php pagina kopiëren.

Boven in je pagina komt het volgende:

```
<?php
include 'assets/model/Database.php';
include_once 'assets/model/classUser.php';
$user = new User();
// Checking for user logged in or not
//if (!$user->get_session())
//{
//    header("location:account.php?pagename=Mijn%20Account");
//}
?>
```

Hieronder onze secties

```
<!-- HEAD
===== -->
<?php include 'assets/elements/head.php' ?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php' ?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php' ?>
```

En nu starten wij ons registratieformulier

Dit formulier is vrij abstract opgebouwd. Ook zijn hier 2 soorten error checks meegegeven, 1 via bootstrap en een ander via javascript. Je kunt zelf kiezen welke je wilt houden al is mijn voorkeur bootstrap.

```

<!-- REGISTREREN
===== -->

<section class="section menu-section" id="register"></section>
<section class="section section_menu section_border_bottom">
    <div id="container" class="container">

        <?php
            if (isset($_POST['submit'])) {
                extract($_POST);
                $register = $user->reg_user($fullname, $uname, $upass,
                $uemail);
                if ($register) {
                    // Registration Success
                    echo "<div class='alert alert-success text-center' >
                        <strong>Registratie gelukt</strong>, klik <a
href='login.php?pagename=Inloggen'>hier</a> om in te loggen.
                        </div>";
                } else {
                    // Registration Failed
                    echo "<div class='alert alert-danger text-center' >
                        <strong>Registratie mislukt</strong>, deze e-mail
of gebruikersnaam bestaat al.
                        </div>";
                }
            }
        ?>

        <h1>Registreren</h1>
        <div class="row">
            <div class="col-md-12">
                <form action="" method="post" name="reg" class="needs-
validation" novalidate>
                    <small><span class='red'>*</span> = verplicht</small>
                    <div class="form-row">
                        <div class="form-group col-md-6">
                            <label for="fullname">Naam <span
class='red'>*</span></label>
                            <input type="text" class="form-control"
id="fullname" name="fullname" placeholder="" required>
                            <div class="valid-feedback">
                                Naam is ingevuld!
                            </div>
                            <div class="invalid-feedback">
                                Naam is niet ingevuld!
                            </div>
                        </div>
                        <div class="form-group col-md-6">
                            <label for="uname">Gebruikersnaam <span
class='red'>*</span></label>
                            <input type="text" class="form-control" id="uname"
name="uname" placeholder="" required>
                            <div class="valid-feedback">
                                Gebruikersnaam is ingevuld!
                            </div>
                            <div class="invalid-feedback">
                                Gebruikersnaam is niet ingevuld!
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```



```

        <div class="form-row">
            <div class="form-group col-md-6">
                <label for="uemail">E-mail <span
class='red'>*</span></label>
                <input type="email" class="form-control"
id="uemail" name="uemail" placeholder="" required>
                <div class="valid-feedback">
                    E-mail is ingevuld!
                </div>
                <div class="invalid-feedback">
                    E-mail is niet ingevuld!
                </div>
            </div>
            <div class="form-group col-md-6">
                <label for="upass">Wachtwoord <span
class='red'>*</span></label>
                <input type="password" class="form-control"
id="upass" name="upass" placeholder="" required>
                <div class="valid-feedback">
                    Wachtwoord is ingevuld!
                </div>
                <div class="invalid-feedback">
                    Wachtwoord is niet ingevuld!
                </div>
            </div>
            <div>
                <input class="btn btn-primary" type="submit" name="submit"
value="Registreren" onclick="return(submitreg());">
            </div>
            <br><br>
            <a href="login.php?pagename=Inloggen#login"> Al een account?
Klik hier!</a>
        </div>
    </div>
</section>

```

Vervolgens de footer met de registratie error javascript.

```
<!-- FOOTER
===== -->
<?php include 'assets/elements/footer.php' ?>
<!-- JAVASCRIPT
===== -->
<?php include 'assets/elements/scripts.php' ?>

<script>
    function submitreg() {
        var form = document.reg;
        if (form.name.value == "") {
            alert("Enter name.");
            return false;
        } else if (form.uname.value == "") {
            alert("Enter username.");
            return false;
        } else if (form.upass.value == "") {
            alert("Enter password.");
            return false;
        } else if (form.uemail.value == "") {
            alert("Enter email.");
            return false;
        }
    }
}
</script>
</body>
</html>
```

Resultaat:

Registreren

*=verplicht

Naam *	Gebruikersnaam *
<input type="text"/>	<input type="text"/>
E-mail *	Wachtwoord *
<input type="text"/>	<input type="password"/>

REGISTREREN

[Al een account? Klik hier!](#)

15. Login.php

Net als bij het registratieformulier is hier eenzelfde formulier. Het begin is iets anders, maar de code zal voor zich spreken. Mocht dit onduidelijk zijn, dan kan er tijdens de instructie cursus uitleg worden gegeven.

```
<?php
session_start();
include 'assets/model/Database.php';
include_once 'assets/model/classUser.php';
$user = new User();

$error = '';
if (isset($_POST['submit'])) {
    extract($_POST);
    $login = $user->check_login($emailusername, $password);
    if ($login) {
        // Registration Success

header("location:adminProductAanmaken.php?pagename=Product%20Aanmaken");
    } else {
        // Registration Failed
        $error = "<div class='alert alert-danger text-center' >
                <strong>Inloggen mislukt</strong>, deze e-mail en
gebruikersnaam combinatie bestaat niet.
                </div>";
    }
}
?>
```

43

```
<!-- HEAD
===== -->
<?php include 'assets/elements/head.php' ?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php' ?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php' ?>
```

```

<!-- INLOGGEN
===== -->

<section class="section menu-section" id="login"></section>
<section class="section section_menu section_border_bottom">
  <div id="container" class="container">
    <?= $error;
    ?>
    <h1>Inloggen</h1>
    <div class="row">
      <div class="col-md-12">
        <form action="" method="post" name="login" class="needs-
validation" novalidate>
          <small><span class='red'>*</span> = verplicht</small>
          <div class="form-row">
            <div class="form-group col-md-6">
              <label for="emailusername">Gebruikersnaam of e-
mail <span class='red'>*</span></label>
              <input type="text" class="form-control"
id="emailusername" name="emailusername" placeholder="" required>
              <div class="valid-feedback">
                Gebruikersnaam of e-mail is ingevuld!
              </div>
              <div class="invalid-feedback">
                Gebruikersnaam of e-mail is niet ingevuld!
              </div>
            </div>
          </div>
          <div class="form-row">
            <div class="form-group col-md-6">
              <label for="password">Wachtwoord <span
class='red'>*</span></label>
              <input type="password" class="form-control"
id="password" name="password" placeholder="" required>
              <div class="valid-feedback">
                Wachtwoord is ingevuld!
              </div>
              <div class="invalid-feedback">
                Wachtwoord is niet ingevuld!
              </div>
            </div>
          </div>
          <input class="btn btn-primary" type="submit"
name="submit" value="Inoggen" onclick="return(submitlogin());">
        </form>
        <br><br>
        <a href="register.php?pagename=Registreren#register"> Nog
geen account? Klik hier!</a>
      </div>
    </div>
  </div>
</section>

```

```

<!-- FOOTER
===== -->
<?php include 'assets/elements/footer.php' ?>
<!-- JAVASCRIPT
===== -->
<?php include 'assets/elements/scripts.php' ?>

<script>
    function submitlogin() {
        var form = document.login;
        if (form.emailusername.value == "") {
            alert("Voer e-mail of gebruikersnaam in.");
            return false;
        } else if (form.password.value == "") {
            alert("Voer wachtwoord in.");
            return false;
        }
    }
}
</script>
</body>
</html>

```

Resultaat:

Inloggen

* = verplicht

Gebruikersnaam of e-mail *

Wachtwoord *

....

INOGGEN

[Nog geen account? Klik hier!](#)

16. Account.php

Dit is de accountpagina, net als de 2 voorgaande is de opbouw hier hetzelfde, echter wordt er nu informatie van de ingelogde gebruiker weergegeven en kunnen wij alleen deze pagina op als onze sessie actief is.

Start sessie, laad model, checkt of je sessie actief is, zo niet stuurt je naar login, bij uitloggen, stuurt je naar login.

```
<?php
session_start();
include 'assets/model/Database.php';
include_once 'assets/model/classUser.php';
$user = new User();

$uid = $_SESSION['uid'];

if (!$user->get_session()){
    header("location:login.php?pagename=Inloggen");
}

if (isset($_GET['q'])){
    $user->user_logout();
    header("location:login.php?pagename=Inloggen");
}

$page = basename(__FILE__);
?>
```

Laadt andere secties in

```
<!-- HEAD
===== -->
<?php include 'assets/elements/head.php' ?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php' ?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php' ?>
```

Beëindigt je sessie

```
<!-- UITLOGGEN & NAAM MELDING
===== -->
<section class="section section_menu section_border_bottom">
    <div id="container" class="container">
        <div id="header">
            <a href="<?=$page;?>q=logout">Uitloggen </a> | <?php $user-
>get_fullname($uid); ?>
        </div>
    </div>
</section>
```

Toont de informatie

```
<section class="section section_menu section_border_bottom">
  <div class="container">
    <h2>Mijn account</h2>
    <table>
      <tr>
        <td>Volledige naam:</td>
        <td><?php $user->get_fullname($uid); ?></td>
      </tr>
      <tr>
        <td>Gebruikersnaam:</td>
        <td><?php $user->get_uname($uid); ?></td>
      </tr>
      <tr>
        <td>E-mail:</td>
        <td><?php $user->get_uemail($uid); ?></td>
      </tr>
    </table>
  </div>
</section>
```

Laad footer en scripts in

```
<!-- FOOTER
===== -->
<?php include 'assets/elements/footer.php' ?>
<!-- JAVASCRIPT
===== -->
<?php include 'assets/elements/scripts.php' ?>

</body>
</html>
```

47

Resultaat:

[Uitloggen](#) | Admin

Mijn account

Volledige naam: Admin

Gebruikersnaam: admin

E-mail: support@3dynamisch.nl

17. Cursusonderdeel 3 – Producten CRUD

Bij dit onderdeel gaan we in op de producten CRUD. Wat houdt een CRUD in? Create, Read, Update & Delete. Kort gezegd, we willen iets aanmaken, dit in een overzicht zetten, dit kunnen updaten en weer verwijderen.

Nu willen wij natuurlijk niet dat onze pagina ook bezocht kan worden door non-admins, daarom hebben wij hiervoor een register en login gemaakt en beveiligen wij onze pagina, zodat alleen onze 3DYNAMISCH admin erin kan.

Het basisprincipe, begint weer bij onze klasse **classProduct.php**.

Voorheen hebben wij enkele functies overgeslagen, zoals **getPostValues()** deze functie haalt voor ons al onze waarden op, nadat deze verstuurd zijn in onze producten pagina. Een uitgebreider verslag dan hier beschreven is, zal tijdens een fysieke cursus behandeld worden.

Resultaat Create:

Uitloggen | Admin

Abonnementen

OVERZICHT | PRODUCT AANMAKEN

* = verplicht

Abonnement *	Internetsnelheid *
<input type="text"/>	2G
Gb Internet *	Belminuten *
5 GB	250
Sms *	Extra kosten *
100	Nee
Prijs *	Afbeelding link *
€ 5.00	https://google.com/hallo.jpg

AANMAKEN



Resultaat Read:

Uitloggen | Admin

Abonnementen

OVERZICHT PRODUCT AANMAKEN

PID	Abonnement	Internetsnelheid	Gb Internet	Belminuten	Sms	Extra Kosten	Prijs	Afbeelding URL	Actions
1	Onbeperkt	4G	onbeperkt	onbeperkt	onbeperkt	nee	32.50	https://images.pexels.com/photos/47261/pexels-photo-47261.jpeg?auto=compress&cs=tinysrgb&dpr=2&h=650&w=940	<button>UPDATE</button> <button>DELETE</button>
2	Start 5GB	4G	5 GB	onbeperkt	onbeperkt	nee	24.00	https://images.pexels.com/photos/248528/pexels-photo-248528.jpeg?Bauto=compress&cs=tinysrgb&dpr=3&h=750&w=1260	<button>UPDATE</button> <button>DELETE</button>
3	Pro 50 GB	5G	50 GB	200	200	nee	29.50	https://images.pexels.com/photos/1092644/pexels-photo-1092644.jpeg?auto=compress&cs=tinysrgb&dpr=2&h=650&w=940	<button>UPDATE</button> <button>DELETE</button>

Resultaat Update:

Uitloggen | Admin

Abonnementen

OVERZICHT PRODUCT AANMAKEN

* = verplicht

Abonnement * Internetsnelheid *

Onbeperkt 4G

Gb Internet * Belminuten *

5 GB 250

Sms * Extra kosten *

100 Nee

Prijs * Afbeelding link *

32,50 https://images.pexels.com/photos/47261/pexels-photo-47261.jpeg?auto

UPDATE

17.1 classProduct.php

Ga naar **classProduct.php** en onder de **getTableName()** functie zet je de volgende functie.

getPostValues()

```

/*-----
GET POST VALUES START
*/
/**
 * @return string getPostValues($post_check_array)
 */
public function getPostValues() {

    // Definieer paramameter values check
    $post_check_array = array (
        // on submit actie
        'add' => array('filter' => FILTER_SANITIZE_STRING ),
        'update'  => array('filter' => FILTER_SANITIZE_STRING ),
        'delete'  => array('filter' => FILTER_SANITIZE_STRING ),
        // form velden + update form velden
        // abonement
        'abonement'  => array('filter' => FILTER_SANITIZE_STRING ),
        // internetsnelheid
        'internetsnelheid'  => array('filter' => FILTER_SANITIZE_STRING ),
        // gb
        'gb'  => array('filter' => FILTER_SANITIZE_STRING ),
        // belminuten
        'belminuten'  => array('filter' => FILTER_SANITIZE_STRING ),
        // belminuten
        'sms'  => array('filter' => FILTER_SANITIZE_STRING ),
        // kosten
        'kosten'  => array('filter' => FILTER_SANITIZE_STRING ),
        // prijs
        'prijs'  => array('filter' => FILTER_SANITIZE_STRING ),
        // afbeelding
        'afbeelding'  => array('filter' => FILTER_SANITIZE_STRING ),
        // Id of current row
        'id'  => array( 'filter'  => FILTER_VALIDATE_INT ),
    );
    // filter de input:
    $inputs = filter_input_array( INPUT_POST, $post_check_array );
    // RTS (Return to sender) - stuurt de values terug
    return $inputs;
}
/*-----
GET POSTVALUES END
*/

```

50

Deze functie zorgt er voor dat alle verstuurde data gefilterd word, voor het gebruikt word.

getGetValues()

Hierna voegen we de volgende functie onder de getPostValues() toe.

Deze functie checkt en filtert de action & id voor het deze terug stuurt.

```
/**
 * getGetValues :
 * Filter input en haal GET parameters op
 *
 * @return array bevat GET input veld
 */
public function getGetValues() {
    //Define parameter check
    $get_check_array = array (
        //Action
        'action' => array('filter' => FILTER_SANITIZE_STRING ),

        //Id van huidige rij
        'id' => array('filter' => FILTER_VALIDATE_INT );

    //Get filtered input:
    $inputs = filter_input_array( INPUT_GET, $get_check_array );

    // RTS (Return to sender) - stuurt de values terug
    return $inputs;
}
```

Nu hebben wij 2 functies toegevoegd, het gebruik hiervan komt later.

51

Nu maken wij de save functie aan. Deze hebben wij nodig om informatie op te slaan. Zie comment voor meer uitleg over de functie.

Save()

```

/* SAVE START
 * 1. Try(Insert into) throw(Exception error) catch(Exception error ->
Empty (zie 2))
 * 2. Wanneer empty, toon niks -> weergave = FALSE -> False toont false
bericht op adminProductAanmaken.php
 * 3. Wanneer succesvol -> Toon True bericht op register.php
 * 4. Zie uitleg hieronder voor meer specifieke informatie.
 */
public function save($input_array){
    try {
        //Check of gegeven values niet verstuurd zijn -> Throw (mist
verplichte velden.)
        if (!isset($input_array['abonnement']) OR
            !isset($input_array['internetsnelheid'])OR
            !isset($input_array['gb'])OR
            !isset($input_array['belminuten'])OR
            !isset($input_array['sms'])OR
            !isset($input_array['kosten'])OR
            !isset($input_array['prijs'])OR
            !isset($input_array['afbeelding'])) {
            // U mist verplichte velden
            throw new Exception("U mist verplichte velden");
        }
        // Check of de gegeven velden een waarde kleiner dan 1 hebben (0) -
> Throw (Verplichte velden zijn leeg.)
        if ( (strlen($input_array['abonnement']) < 1) OR
            (strlen($input_array['internetsnelheid']) < 1) OR
            (strlen($input_array['gb']) < 1) OR
            (strlen($input_array['belminuten']) < 1) OR
            (strlen($input_array['sms']) < 1) OR
            (strlen($input_array['kosten']) < 1) OR
            (strlen($input_array['prijs']) < 1) OR
            (strlen($input_array['afbeelding']) < 1)) {
            // Verplichte velden zijn leeg
            throw new Exception("Verplichte velden zijn leeg") ;
        }

        // Voeg de database connectie toe.
        $conn = Database::getConnection();

        // Insert query
        $conn->query("INSERT INTO `". $this->getProductTableName(). "` (
            `abonnement`, `internetsnelheid`, `gbinternet`,
            `belminuten`, `sms`, `extrakosten`, `prijs`, `afbeelding`)".
            " VALUES ( '". $input_array['abonnement']. "',
            '". $input_array['internetsnelheid']. "', '". $input_array['gb']. "',
            '". $input_array['belminuten']. "',
            '". $input_array['sms']. "', '". $input_array['kosten']. "',
            '". $input_array['prijs']. "', '". $input_array['afbeelding']. "')");

        // Indien een error, roept error gegevens op:
        if ( !empty($conn->last_error) ){
            $this->last_error = $conn->last_error;
            // Helaas, toon error bericht in register.php
            return FALSE;
        }
    }
}

```

```

    } catch (Exception $exc) {
        // Roep exception melding aan (word hier niet gebruikt)
        // echo $this->$exc;
        // return FALSE
    }
    // Mooi, toon succes bericht in register.php
    return TRUE;
}
/*-----
* SAVE END
*/

```

getNrOfProducten

Nu willen wij het aantal rijen checken die wij ophalen.

```

/**
 *
 * @return int aantal producten in database
 */
public function getNrOfProducten() {
    // Call database = $global wpdb in wordpress
    $conn = Database::getConnection();

    $query = "SELECT COUNT(*) AS nr FROM `". $this-
>getProductTableName(). "`";
    $result = $conn->query( $query );
    $followingdata = $result->fetch_assoc();

    return $followingdata['nr'];
}

```

handleGetAction()

Om onze acties te behandelen gebruiken wij

```
/**
 * geeft aan wat update en delete moeten doen.
 */
public function handleGetAction( $get_array ){
    $action = '';

    switch($get_array['action']){
        case 'update':
            // Indicate current action is update if id provided
            if ( !is_null($get_array['id']) ){
                $action = $get_array['action'];
            }
            break;

        case 'delete':
            // Delete current id if provided
            if ( !is_null($get_array['id']) ){
                $this->delete($get_array);
            }
            $action = 'delete';
            break;

        default:
            // Oops
            break;
    }
    return $action;
}
```

Update()

Wanneer wij ons product willen updaten, dienen wij de volgende informatie mee te sturen. Eerst checken wij de velden, waarna wij vervolgens een database connectie maken en de data verzenden. Gebaseerd op de uitkomst krijgen wij een TRUE of FALSE mee terug.

```
/**
 * Functie die producten verwijderd
 */
public function update($input_array){
    try {
        //Check of gegeven values niet verstuurd zijn -> Throw
(mist verplichte velden.)
        if (!isset($input_array['abonnement']) OR
            !isset($input_array['internetsnelheid'])OR
            !isset($input_array['gb'])OR
            !isset($input_array['belminuten'])OR
            !isset($input_array['sms'])OR
            !isset($input_array['kosten'])OR
            !isset($input_array['prijs'])OR
            !isset($input_array['afbeelding'])){
            // U mist verplichte velden
            throw new Exception("U mist verplichte velden");
        }
        // Check of de gegeven velden een waarde kleiner dan 1 hebben
(0) -> Throw (Verplichte velden zijn leeg.)
        if ( (strlen($input_array['abonnement']) < 1) OR
            (strlen($input_array['internetsnelheid']) < 1) OR
            (strlen($input_array['gb']) < 1) OR
            (strlen($input_array['belminuten']) < 1) OR
            (strlen($input_array['sms']) < 1) OR
            (strlen($input_array['kosten']) < 1) OR
            (strlen($input_array['prijs']) < 1) OR
            (strlen($input_array['afbeelding']) < 1)){
            // Verplichte velden zijn leeg
            throw new Exception("Verplichte velden zijn leeg") ;
        }

        // Call database = $global wpdb in wordpress
        $conn = Database::getConnection();
        // Update query
        /**
        * Voorbeeld waarden, zorg dat de input_array['value'] tussen ' '
        staat, anders leest de MYSQL query de waarden niet correct.
        * Pas deze query zo nodig aan om indien nodig jouw mysql te
        testen
        */
        // $conn->query("UPDATE `product` SET `abonnement` = 'Testje',
        `internetsnelheid` = '3G', `gbinternet` = '50 GB', `belminuten` = '100',
        // `sms` = '250', `extrakosten` = 'Nee', `prijs` = '500',
        // `afbeelding` =
        'https://images.pexels.com/photos/1092644/pexels-photo-
        1092644.jpeg?auto=compress&cs=tinysrgb&dpr=2&h=650&w=940'
        // WHERE `product`.`id` = 29;");

        $conn->query("UPDATE `".$this->getProductTableName()."` SET
        `abonnement` = '".$input_array['abonnement']."'");
```

```

        `internetsnelheid` = '". $input_array['internetsnelheid']."',
`gbinternet` = '". $input_array['gb']."', `belminuten` =
' ". $input_array['belminuten']."',
        `sms` = '". $input_array['sms']."', `extrakosten` =
' ". $input_array['kosten']."', `prijs` = '". $input_array['prijs']."',
        `afbeelding` = '". $input_array['afbeelding']."'
        WHERE `". $this->getProductTableName(). "`.`id` =
' ". $input_array['id']. "';");

        /**
         * Update Dump bestand, uncomment om te zien hoe de SQL eruit
        ziet.
        */

//        var_dump("UPDATE `". $this->getProductTableName(). "` SET
`abonnement` = '". $input_array['abonnement']."',
//        `internetsnelheid` = '". $input_array['internetsnelheid']."',
`gbinternet` = '". $input_array['gb']."', `belminuten` =
' ". $input_array['belminuten']."',
//        `sms` = '". $input_array['sms']."', `extrakosten` =
' ". $input_array['kosten']."', `prijs` = '". $input_array['prijs']."',
//        `afbeelding` = '". $input_array['afbeelding']."'
//        WHERE `". $this->getProductTableName(). "`.`id` =
' ". $input_array['id']. "';");

        // Indien een error, roept error gegevens op:
        if ( !empty($conn->last_error) ){
            $this->last_error = $conn->last_error;
            // Helaas, toon error bericht in register.php
            return FALSE;
        }
    } catch (Exception $exc) {

    }

    return TRUE;
}

```


Delete()

Functie om een rij te verwijderen op basis van meegeleverd ID.

```
/**
 * Functie die de producten verwijderd
 */
public function delete($input_array){
    try {
        // Check input id
        if (!isset($input_array['id']) ) throw new Exception(__("Missing mandatory fields") );

        // Call database = $global wpdb in wordpress
        $conn = Database::getConnection();

        $conn->query("DELETE FROM `" . $this->getProductTableName() . "`
WHERE id = ".$input_array['id']."");

    } catch (Exception $exc) {
        return FALSE;
    }
    return TRUE;
}
/*-----
FUNCTIONS PRODUCT END
*/
```

17.2 adminProductAanmaken.php

Op deze pagina maken wij de gehele CRUD. De opbouw van de code is precies zoals deze in de pagina gezet is. Deze is in stukken gesneden voor extra uitleg. Meer zal tijdens de les verschaft worden, met eventueel de reden achter gekozen elementen en functies.

Hier starten wij onze sessie, declareren wij onze klasse en zorgen wij dat onbevoegden van de pagina afgegooid worden.

```
<?php
session_start();
include 'assets/model/Database.php';
// Include model:
include_once 'assets/model/classUser.php';
include_once "assets/model/classProduct.php";

// Declare class variable:
$user = new User();
$product = new classProduct();

$uid = $_SESSION['uid'];

if (!$user->get_session()){
    header("location:login.php?pagename=Inloggen");
}

if (isset($_GET['q'])){
    $user->user_logout();
    header("location:login.php?pagename=Inloggen");
}
```

58

Declareer de pagina **basename** en Koppel deze aan **\$page**, indien nieuwsgierig, zet **var_dump(\$page)**; neer. Kopet getGetValues aan de variabele \$get_array. Hierna, volg elke actie via handleGetAction.

```
$page = basename(__FILE__);

// Add params to base url
$base_url = $page;

// Get the GET data in filtered array
$get_array = $product->getGetValues();

// Keep track of current action.
$action = FALSE;
if (!empty($get_array)) {
    // Check actions
    if (isset($get_array['action'])) {
        $action = $product->handleGetAction($get_array);
    }
}
```

Maak de **\$post_array** en voer wanneer form submit (add) verzonden is, de save functie uit.

```
/* Na checken */
// Get the POST data in filtered array
$post_array = $product->getPostValues();

// Collect Errors
$error = FALSE;
// Check the POST data
if (!empty($post_array['add'])) {

    // Check the add form:
    $add = FALSE;
    // Save event types
    $result = $product->save($post_array);
    if ($result) {
        // Save was succesfull
        $add = TRUE;
    } else {
        // Indicate error
        $error = TRUE;
    }
}
```

Zelfde als hierboven, maar voor delete

```
$Edelete = FALSE;
// Check the POST data
if (!empty($post_array['delete'])) {

    // Check the add form:
    $delete = FALSE;
    // Save event types
    $result = $product->delete($_POST['id']);
    if ($result) {
        // Save was succesfull
        $delete = TRUE;
    } else {
        // Indicate error
        $Edelete = TRUE;
    }
}
```

59

Wanneer update verzonden is, voer update functie uit

```
// Check the update form:
if (isset($post_array['update'])) {
    // Save event types
    $product->update($post_array);
}
?>
```

Hierna komen weer onze pagina's

```
<!-- HEAD
===== -->
<?php include 'assets/elements/head.php' ?>

<!-- NAVBAR
===== -->
<?php include 'assets/elements/nav.php' ?>

<!-- WELCOME
===== -->
<?php include 'assets/elements/welcome.php' ?>
```

Check of UID gelijk is aan 1 (=3dynamisch admin), zo ja geef toegang, zo niet, toon error (= vermeld onderin de pagina) Voeg ook de logout optie toe.

```
<?php
// checkt of het gegeven uid 1 is, coorespondeert met de standaard data -
// DEFAULT id 1 is admin met password 3dynamisch
if ($uid == '1'){?>
<!-- UITLOGGEN & NAAM MELDING
===== -->
<section class="section section_menu section_border_bottom">
  <div id="container" class="container">
    <div id="header">
      <a href="<?=$page;?>?q=logout">Uitloggen </a> | <?php $user-
>get_fullname($uid);?>
    </div>
```

Knoppen om overzicht te bekijken waar update en delete staan en optie op de product toevoegen pagina in te zien.

Hierna 2 checks voor add en delete if success.

```
<h2>Abonnementen</h2>
<a class="btn btn-primary"
href="adminProductAanmaken.php?pagename=Overzicht#overzicht">Overzicht</a>
<a class="btn btn-primary"
href="adminProductAanmaken.php?pagename=Product%20Aanmaken#aanmaken">Product
Aanmaken</a>
<?php
if (isset($add)) {
    echo($add ? '<div class="alert alert-success text-center">
                <strong>Gelukt!</strong> Product is succesvol
aangemaakt.</div>'
        :
        '<div class="alert text-center alert-danger">
                <strong>Error!</strong> Product kon niet worden
aangemaakt.
                </div>');
}

if (isset($delete)) {
    echo($delete ? '<div class="alert alert-success text-center">
                <strong>Gelukt!</strong> Product is succesvol
verwijderd.</div>'
        :
        '<div class="alert text-center alert-danger">
                <strong>Error!</strong> Product kon niet worden
verwijderd.
                </div>');
}
```

File pad om de pagina terug te vinden na het updaten.

```
// Define file path
$filepath =
'http://localhost/OpdrachtOOPHP/adminProductAanmaken.php?pagename=Overzicht
';
```

Start Overzicht (als pagename overzicht is, toon table (Read, update, delete functies))

```
if($pagename == 'Overzicht'){
?>

<table class="table table-striped table-hover" id="overzicht">
<?php if($action!='update'){?>
<thead class="table-primary">
<tr>
    <th width="10">PID</th>
    <th width="500">Abonnement</th>
    <th width="2000">Internetsnelheid</th>
    <th width="2000">Gb Internet</th>
    <th width="2000">Belminuten</th>
    <th width="2000">Sms</th>
    <th width="2000">Extra Kosten</th>
    <th width="2000">Prijs</th>
    <th width="2000">Afbeelding URL</th>
    <th colspan="2" width="200">Actions</th>
</tr>
</thead>
```

Als er geen producten zijn toon dit bericht.

```
<?php } ?>
<br><br>
<?php
/**
if ($product->getNrOfProducten() < 1) {
?>
<tr>
    <td colspan="3">
        <div class="alert text-center alert-danger">
            Er zijn nog geen producten, start met deze aan te
            maken.
        </div>
    </td>
</tr>
<?php } else {
$product_list = $product->getProductList();
```

Update form

Volledige update formulier – Uitleg tijdens fysieke les met redenering achter functies.

```

/** Show all event types in the tabel
foreach ($product_list as $product_obj) {

    // Add params to base url update link
    $upd_link = $filepath . '&action=update&id=' . $product_obj->getProductId();

    // Add params to base url delete link
    $del_link = $filepath . '&action=delete&id=' . $product_obj->getProductId();
    ?>

    <?php
    // If update and id match show update form
    // Add hidden field id for id transfer
    if (($action == 'update') && ($product_obj->getProductId() == $get_array['id'])) {
        ?>
        <div class="row">
            <div class="col-md-12">
                <form action="<?=$base_url;?>" method="post"
class="needs-validation" novalidate">
                    <small><span class='red'>*</span> = verplicht</small>
                    <div class="form-row">
                        <div class="form-group col-md-6">
                            <input type="hidden" id="id" name="id"
value="<?=$product_obj->getProductId(); ?>">
                            <label for="abonnement">Abonnement <span
class='red'>*</span></label>
                            <input type="text" class="form-control"
id="abonnement" name="abonnement" value="<?=$product_obj->getAbonnement(); ?>"required>
                            <div class="valid-feedback">
                                Abonnement is ingevuld!
                            </div>
                            <div class="invalid-feedback">
                                Abonnement is niet ingevuld!
                            </div>
                        </div>
                        <div class="form-group col-md-6">
                            <label for="internetsnelheid">Internetsnelheid
<span class='red'>*</span></label>
                            <select class="custom-select"
name="internetsnelheid" required>
                                <?php if($product_obj->getInternetsnelheid() == '2G'){
                                    $internetsnelheid1 = 'selected ';
                                }elseif($product_obj->getInternetsnelheid()
== '3G'){
                                    $internetsnelheid2 = 'selected ';
                                }elseif($product_obj->getInternetsnelheid()
== '4G'){
                                    $internetsnelheid3 = 'selected ';
                                }elseif($product_obj->getInternetsnelheid()
== '5G'){

```

```

        $internetsnelheid4 ='selected ' ;
    }else{}
    ?>
    <option <?php
if(!empty($internetsnelheid1)){echo $internetsnelheid1;} ?>
        value="2G">2G</option>
    <option <?php
if(!empty($internetsnelheid2)){echo $internetsnelheid2;} ?>
        value="3G">3G</option>
    <option <?php
if(!empty($internetsnelheid3)){echo $internetsnelheid3;} ?>
        value="4G">4G</option>
    <option <?php
if(!empty($internetsnelheid4)){echo $internetsnelheid4;} ?>
        value="5G">5G</option>
    </select>
    <div class="valid-feedback">
        Internetsnelheid is ingevuld!
    </div>
    <div class="invalid-feedback">
        Internetsnelheid is niet ingevuld!
    </div>
</div>
<div class="form-row">
    <div class="form-group col-md-6">
        <label for="gb">Gb Internet <span
class='red'>*</span></label>
        <select class="custom-select" name="gb"
required>
            <?php if($product_obj->getGbInternet() ==
'5 GB'){
                $internet1 ='selected ' ;
            }elseif($product_obj->getGbInternet() ==
'50 GB'){
                $internet2 ='selected ' ;
            }elseif($product_obj->getGbInternet() ==
'100 GB'){
                $internet3 ='selected ' ;
            }elseif($product_obj->getGbInternet() ==
'250 GB'){
                $internet4 ='selected ' ;}
            elseif($product_obj->getGbInternet() ==
'Onbeperkt'){
                $internet5 ='selected ' ;
            }else{}
            ?>
            <option <?php if(!empty($internet1)){echo
$internet1;} ?>
                value="5 GB">5 GB</option>
            <option <?php if(!empty($internet2)){echo
$internet2;} ?>
                value="50 GB">50 GB</option>
            <option <?php if(!empty($internet3)){echo
$internet3;} ?>
                value="100 GB">100 GB</option>
            <option <?php if(!empty($internet4)){echo
$internet4;} ?>
                value="250 GB">250 GB</option>

```



```

<option <?php if(!empty($internet5)){echo
$internet5;} ?>
value="Onbeperkt">Onbeperkt</option>
</select>
<div class="valid-feedback">
    Gb Internet is ingevuld!
</div>
<div class="invalid-feedback">
    Gb Internet is niet ingevuld!
</div>
</div>
<div class="form-group col-md-6">
    <label for="belminuten">Belminuten <span
class='red'>*</span></label>
    <select class="custom-select" name="belminuten"
required>
        <?php if($product_obj->getBelminuten() ==
'250'){
            $bel1 ='selected ';
        }elseif($product_obj->getBelminuten() ==
'500'){
            $bel2 ='selected ';
        }elseif($product_obj->getBelminuten() ==
'Onbeperkt'){
            $bel3 ='selected ';
        }else{}
        ?>
        <option <?php if(!empty($bel1)){echo
$bel1;} ?> value="250">250</option>
        <option <?php if(!empty($bel2)){echo
$bel2;} ?> value="500">500</option>
        <option <?php if(!empty($bel3)){echo
$bel3;} ?> value="Onbeperkt">Onbeperkt</option>
    </select>
    <div class="valid-feedback">
        Belminuten is ingevuld!
    </div>
    <div class="invalid-feedback">
        Belminuten is niet ingevuld!
    </div>
</div>
</div>
<div class="form-row">
    <div class="form-group col-md-6">
        <label for="sms">Sms <span
class='red'>*</span></label>
        <select class="custom-select" name="sms"
required>
            <?php if($product_obj->getSms() == '100'){
                $sms1 ='selected ';
            }elseif($product_obj->getSms() == '200'){
                $sms2 ='selected ';
            }elseif($product_obj->getSms() == '250'){
                $sms3 ='selected ';
            }elseif($product_obj->getSms() ==
'Onbeperkt'){
                $sms4 ='selected ';
            }else{}

```

```

        ?>
        <option <?php if(!empty($sms1)){echo
$sms1;} ?>
            value="100">100</option>
        <option <?php if(!empty($sms2)){echo
$sms2;} ?>
            value="200">200</option>
        <option <?php if(!empty($sms3)){echo
$sms3;} ?>
            value="250">250</option>
        <option <?php if(!empty($sms4)){echo
$sms4;} ?>
            value="Onbeperkt">Onbeperkt</option>
        </select>
        <div class="valid-feedback">
            Sms is ingevuld!
        </div>
        <div class="invalid-feedback">
            Sms is niet ingevuld!
        </div>
    </div>
    <div class="form-group col-md-6">
        <label for="kosten">Extra kosten <span
class='red'>*</span></label>
        <select class="custom-select" name="kosten"
required>
            <?php if($product_obj->getExtraKosten() ==
'Ja'){
                $kosten ='selected';
            }else{
                $kosten =' ';
            }
            ?>
            <option value="Nee">Nee</option>
            <option <?=$kosten;?>
value="Ja">Ja</option>
        </select>
        <div class="valid-feedback">
            Extra kosten is ingevuld!
        </div>
        <div class="invalid-feedback">
            Extra kosten is niet ingevuld!
        </div>
    </div>
    <div class="form-row">
        <div class="form-group col-md-6">
            <label for="prijs">Prijs <span
class='red'>*</span></label>
            <input type="number" class="form-control"
step="0.01" lang="en-US" id="prijs" name="prijs" value="<?= $product_obj-
>getPrijs(); ?>" placeholder="€ 5.00" required>
            <div class="valid-feedback">
                Prijs is ingevuld!
            </div>
            <div class="invalid-feedback">
                Prijs is niet ingevuld!
            </div>
        </div>
    </div>

```

```

        </div>
        <div class="form-group col-md-6">
            <label for="afbeelding">Afbeelding link <span
class='red'>*</span></label>
            <input type="text" class="form-control"
id="afbeelding" name="afbeelding" value="<?= $product_obj->getAfbeelding();
?>" placeholder="https://google.com/hallo.jpg" required>
            <div class="valid-feedback">
                Afbeelding is ingevuld!
            </div>
            <div class="invalid-feedback">
                Afbeelding is niet ingevuld!
            </div>
        </div>
    </div>
    <input class="btn btn-primary" type="submit"
name="update" value="Update">
</form>
<br><br>
</div>
</div>
</div>
</section>

```

Tabel rijen

Toon tabel rijen

```
<tr>
    <?php } else { ?>
        <?php if ($action !== 'update') {
            // If action is update don't show the action button
            ?>
            <td width="10"><?=$product_obj->getProductId();
?></td>
            <td width="180"><?=$product_obj->getAbonnement();
?></td>
            <td width="200"><?=$product_obj->getInternetsnelheid();
?></td>
            <td width="200"><?=$product_obj->getGbInternet();
?></td>
            <td width="200"><?=$product_obj->getBelminuten();
?></td>
            <td width="200"><?=$product_obj->getSms(); ?></td>
            <td width="200"><?=$product_obj->getExtraKosten();
?></td>
            <td width="200"><?=$product_obj->getPrijs(); ?></td>
            <td width="200"><?=$product_obj->getAfbeelding();
?></td>
            <td><a class="btn btn-success" href="<?=$upd_link;
?>">Update</a></td>
            <td><a class="btn btn-danger" name="delete" href="<?=$del_link; ?>">Delete</a></td>
        <?php
        } // if action !== update
        ?>
        <?php } // if action !== update ?>
    </tr>
    <?php
}
?>

<?php } }else {
?>
</table>
```

```

<?php
// Check if action = update : then end update form
echo(($action == 'update') ? '</form>' : '');

if ($action !== 'update') {
    ?>
    <div class="row" id="aanmaken">
        <div class="col-md-12">
            <form action="<?= $base_url; ?>" method="post"
class="needs-validation" novalidate
            ">
                <small><span class='red'>*</span> = verplicht</small>
                <div class="form-row">
                    <div class="form-group col-md-6">
                        <label for="abonnement">Abonnement <span
class='red'>*</span></label>
                        <input type="text" class="form-control"
id="abonnement" name="abonnement" placeholder=""
                        required>
                        <div class="valid-feedback">
                            Abonnement is ingevuld!
                        </div>
                        <div class="invalid-feedback">
                            Abonnement is niet ingevuld!
                        </div>
                    </div>
                    <div class="form-group col-md-6">
                        <label for="internetsnelheid">Internetsnelheid
<span class='red'>*</span></label>
                        <select class="custom-select"
name="internetsnelheid" required>
                            <option value="2G">2G</option>
                            <option value="3G">3G</option>
                            <option value="4G">4G</option>
                            <option value="5G">5G</option>
                        </select>
                        <div class="valid-feedback">
                            Internetsnelheid is ingevuld!
                        </div>
                        <div class="invalid-feedback">
                            Internetsnelheid is niet ingevuld!
                        </div>
                    </div>
                </div>
                <div class="form-row">
                    <div class="form-group col-md-6">
                        <label for="gb">Gb Internet <span
class='red'>*</span></label>
                        <select class="custom-select" name="gb" required>
                            <option value="5 GB">5 GB</option>
                            <option value="50 GB">50 GB</option>
                            <option value="100 GB">100 GB</option>
                            <option value="250 GB">250 GB</option>
                            <option value="Onbeperkt">Onbeperkt</option>
                        </select>

```

```

        <div class="valid-feedback">
            Gb Internet is ingevuld!
        </div>
        <div class="invalid-feedback">
            Gb Internet is niet ingevuld!
        </div>
    </div>
    <div class="form-group col-md-6">
        <label for="belminuten">Belminuten <span
class='red'>*</span></label>
        <select class="custom-select" name="belminuten"
required>
            <option value="250">250</option>
            <option value="500">500</option>
            <option value="Onbeperkt">Onbeperkt</option>
        </select>
        <div class="valid-feedback">
            Belminuten is ingevuld!
        </div>
        <div class="invalid-feedback">
            Belminuten is niet ingevuld!
        </div>
    </div>
</div>
<div class="form-row">
    <div class="form-group col-md-6">
        <label for="sms">Sms <span
class='red'>*</span></label>
        <select class="custom-select" name="sms" required>
            <option value="100">100</option>
            <option value="200">200</option>
            <option value="250">250</option>
            <option value="Onbeperkt">Onbeperkt</option>
        </select>
        <div class="valid-feedback">
            Sms is ingevuld!
        </div>
        <div class="invalid-feedback">
            Sms is niet ingevuld!
        </div>
    </div>
    <div class="form-group col-md-6">
        <label for="kosten">Extra kosten <span
class='red'>*</span></label>
        <select class="custom-select" name="kosten"
required>
            <option value="Nee">Nee</option>
            <option value="Ja">Ja</option>
        </select>
        <div class="valid-feedback">
            Extra kosten is ingevuld!
        </div>
        <div class="invalid-feedback">
            Extra kosten is niet ingevuld!
        </div>
    </div>
</div>
<div class="form-row">
    <div class="form-group col-md-6">

```

```

        <label for="prijs">Prijs <span
class='red'>*</span></label>
        <input type="number" class="form-control"
step="0.01" lang="en-US" id="prijs" name="prijs"
        placeholder="€ 5.00" required>
        <div class="valid-feedback">
            Prijs is ingevuld!
        </div>
        <div class="invalid-feedback">
            Prijs is niet ingevuld!
        </div>
    </div>
    <div class="form-group col-md-6">
        <label for="afbeelding">Afbeelding link <span
class='red'>*</span></label>
        <input type="text" class="form-control"
id="afbeelding" name="afbeelding"
        placeholder="https://google.com/hallo.jpg"
required>
        <div class="valid-feedback">
            Afbeelding is ingevuld!
        </div>
        <div class="invalid-feedback">
            Afbeelding is niet ingevuld!
        </div>
    </div>
    <input class="btn btn-primary" type="submit" name="add"
value="Aanmaken">
    </form>
    <br><br>
</div>
</div>
</div>
</section>
<?php
    } // if action !== update
}

```

Toon melding, wanneer de gebruiker geen admin is.

```

}else{?>
    <section class="section section_menu section_border_bottom">
        <div id="container" class="container">
            <div id="header">
                <p class="alert alert-danger">U bent niet gemachtigd om deze
pagina in te zien!</p>
            </div>
        </div>
    </section>
<?php }

```

Als de paginanaam overzicht is, verberg de footer, anders tonen en heel overzicht verbergen. Laad footer en javascript en sluit file.

```
if ($pagename=='Overzicht') {} else{?>

<!-- FOOTER
===== -->
<?php include 'assets/elements/footer.php' ?>

<?php } ?>
<!-- JAVASCRIPT
===== -->
<?php include 'assets/elements/scripts.php' ?>

</body>
</html>
```

18. Algemene informatie

Het doel van deze cursus is om studenten die lessen gemist hebben of waar de stof niet voldoende gegeven is, toch verder te helpen. Hieronder vind je kort informatie over enkele functies in het **classProduct.php** script.

Functie	Beschrijving
getProductTableName()	Haalt de tabel naam op
getPostValues()	Haalt alle geposte waarden op en filtert deze
getGetValues()	Haalt alle waarden op die iets moeten geven (action en id)
Save(\$input_array)	Slaat de gegeven waarden op
getProductList()	Haalt alle waarden in gegeven database tabel op en set deze in een get
getProductById(\$pageld)	Haalt alle waarden in gegeven database tabel voor gegeven ID op en set deze in een get
getNrOfProducten()	Haalt aantal producten op
handleGetAction(\$get_array)	Voert de actie door voor update/delete
Update(\$input_array)	Update de velden met gegeven data voor gegeven ID
Delete(\$input_array)	Delete het veld voor gegeven ID
Set[Value]	Zet de opgehaalde data klaar in functie
Get[Value]	Geeft de opgehaalde value uit functie set

Rol	Naam	E-mail
Auteur/instructeur	Kevin Schuit	Kevin.schuit@3dynamisch.nl