Documentatie inschrijfformulier met SQL / Database



Bas Geijtenbeek

Angelo van ’t Riet

# Inhoudsopgave

3. Inleiding

4. Index.php

5. Account\_data.php

6. Error.php

7. Forgot.php

8. Login.php

9. Logout.php

10. Profile.php

11. Reset.php

12. Reset\_password.php

13. Succes.php

14. Verify.php

# Inleiding

In dit document kun je al de documentatie vinden die hoort bij de casus van het examen. Dit gaat om een inschrijfformulier met achterliggende database. Doormiddel van SQL en PHP is er een dynamische verbinding tussen dit formulier en database. We zullen alle templates langs gaan in dit document en de code verantwoorden die in de templates te vinden zijn.

# Index.php

De index.php is onze homepagina. Hier vind je alle informatie en documenten terug m.b.t. het examen.

Het hoofdmenu bestaat uit HTML, CSS en Javascript.



De 4 pagina’s die te vinden zijn, zijn de homepagina, de documentatie pagina, bronnen pagina en een link naar ons login systeem wat doormiddel van een pop-up verschijnt.

De effecten worden uit verschillende javascript bestanden gehaald die weer in de footer worden ingeladen.

Dit gaat om:

Index.js

Overlay.js

Verder vind je op deze pagina een intro tekst en verder niks spannends.

# Account\_data.php

Dit bestand haalt de gegevens op als je ingelogd ben in het registratie formulier. Dit kun je zien door middel van de functie $\_SESSION.

Door het gebruiken van session maak je bepaalde informatie alleen zichtbaar als de gebruiker ingelogd is op de website en geef je eventueel toestemming om bepaalde handelingen te mogen doen.

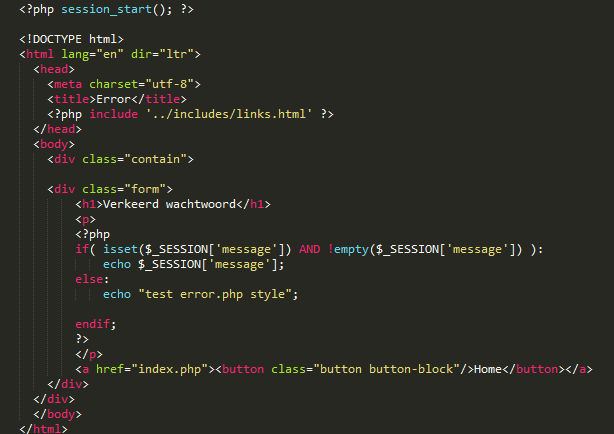
In ons geval kun je dan je eigen account in zien. Je kunt deze informatie niet inzien als je geen account hebt gemaakt. Met $\_SESSION geef je dus aan dat deze informatie zichtbaar word / mag worden als je bent ingelogd.



# Error.php

Dit bestand geeft eigenlijk al in de titel aan wat het doet. Dit is wat naar voren komt als je inlogt met bijvoorbeeld verkeerde inloggegevens. Je krijgt een foutmelding als ongeldige inloggegevens gebruikt worden.

Doormiddel van isset kijkt het bestand of deze velden niet leeggelaten zijn. Ook dan krijg je een foutmelding namelijk. Deze velden moeten altijd ingevuld zijn.



# Forgot.php

Dit bestand bestaat eigenlijk uit 2 delen. Ik begin met het gedeelte van de code om je wachtwoord te resetten. Bovenaan het bestand gebruiker we een require. Deze hebben we nodig omdat we bij de database moeten zijn. Door middel van een SQL statement:



Hier word gekeken of je e-mail adres wel bestaat in de database.

Anders krijg je de melding:

**Gebruiker met deze mail bestaat niet!**

En ga je terug naar de error pagina.

Als deze wel bestaat word er een array gemaakt met gebruikersdata: email, hash en eerste naam. Het e-mailadres wat hier staat daar word de link naar verstuurd.

Vervolgens word er een sessie gestart en krijg je de melding: Bekijk je e-mail voor de link om je wachtwoord re resetten.

Vervolgend word deze mail verstuurt naar je e-mailadres met de link daarin om je wachtwoord te resetten.

Verder is dit bestand gekoppeld aan de reset.php, hier kom ik nog op terug.



# Login.php

Dit is de code om in te loggen in ons systeem. Hier word gekeken of je gebruikers naam en wachtwoord bestaan en of deze wel matchen aan elkaar.

De code begint met een escape, dit hebben we zo gedaan om SQL injecties te voorkomen.

Verder worden de resultaten opgehaald vanuit het statement:



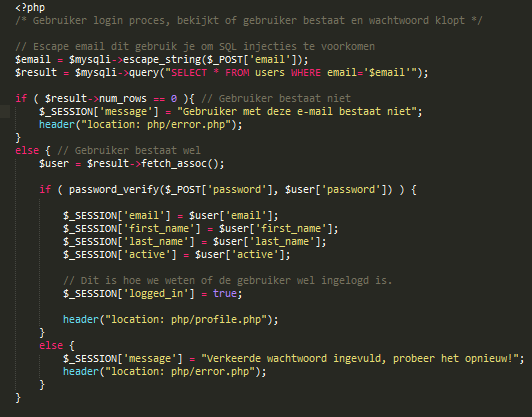
Als hier geen resultaten uitkomen betekent dat de gebruiker niet in de database zit en dus ook niet bestaat.

Als de gebruiker wel bestaat verifieerd hij de post en zet hij de gebruiker op active.

Dit gebeurt door middel van: 

Als het wachtwoord niet klopt krijg je de error.php voor je met een melding.

Dit gebeurt allemaal doormiddel 2 if else statement.



# Logout.php

Hier gebeurt niet heel veel bijzonders gedaan. De sessie word aangemaakt vervolgens onklaar gemaakt en vernietigd. Je bent daarna uitgelogd. Session unset vernietigd een bepaalde sessie en destroy vernietigd ze allemaal. Dit is beetje ter extra veiligheid.



# Profile.php

Deze pagina is alleen zichtbaar als je ingelogd bent en dus een account hebt. Hier kun je de informatie inzien die je hebt gemaakt bij het inloggen.

Als je niet ingelogd bent en toch op de pagina te komen krijg je de melding dat je eerst moet inloggen op de pagina te komen.

Als je wel ingelogd bent haalt hij de gegevens op die bij je account horen en word de sessie op actief gezet zodat je de pagina en je gegevens kan inzien.

Ook word er gecheckt of je account wel geactiveerd is, als dit nog niet gebeurd is zal je eerst nog je account moeten verifiëren om de pagina te bereiken.



# Reset.php

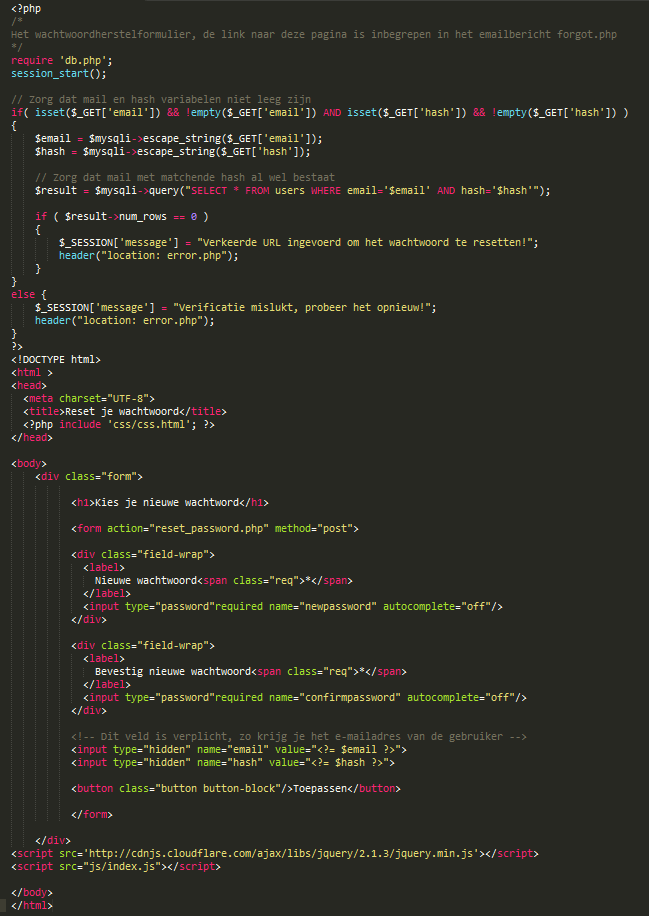
Dit bestand is gekoppeld aan de forgot.php

Eerst word er gekeken of de hash en mail variabelen niet leeg zijn, vervolgens kijkt hij of deze wel bestaan en matched deze als dit lukt kun je gewoon het wachtwoord resetten.

Als er bijvoorbeeld een verkeerde URL word ingevuld om het wachtwoord te resetten krijg je hier een foutmelding van.

Als de verificatie niet gelukt is krijg daar ook een melding van.

In beide gevallen kun je dan in ieder geval nog niet je wachtwoord resetten.



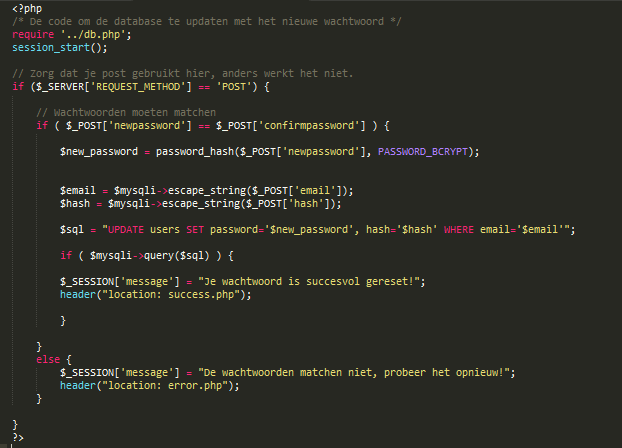
# Reset\_password.php

Dit is de code om de database te updaten met het nieuwe aangemaakte wachtwoord. Let op dat hier post gebruikt word anders gaat dit niet werken.

Als je het nieuwe wachtwoord aanmaakt moet je die 2 keer invullen. Zorg dat deze matchen anders werkt dit niet. Dit is ter extra veiligheid. Als deze niet matchen krijg je daar een aparte foutmelding van.

Als de wachtwoorden wel matchen word er een statement uitgevoerd waardoor de velden in de database worden vernieuwd en het nieuwe wachtwoord word opgeslagen.

Als dit statement is uitgevoerd krijg je een melding en kun je weer inloggen met de nieuwe gegevens.



\\\\

# Succes.php

Dit bestand doet eigenlijk ook niet zoveel. Als een handeling gelukt is krijg je de succes message. Echter worden deze opgehaald uit andere templates door middel van sessions. Vanuit die sessions haalt hij dit bestand dan weer op.

# Verify.php

Deze code gebruiken we om de accounts te activeren. Als je dus een account aangemaakt hebt krijg je een e-mail met een activatie link.

Als dit nog niet gebeurt is staat je account in de database met status 0 (niet actief)

Als je het account al geactiveerd hebt of een foute URL invult krijg je daar een aparte melding van. Anders krijg je de melding dat je account geactiveerd is.

Als je account geactiveerd is word doormiddel van een querie de status op 1 gezet. Je kunt dan gebruik maken van je account.

