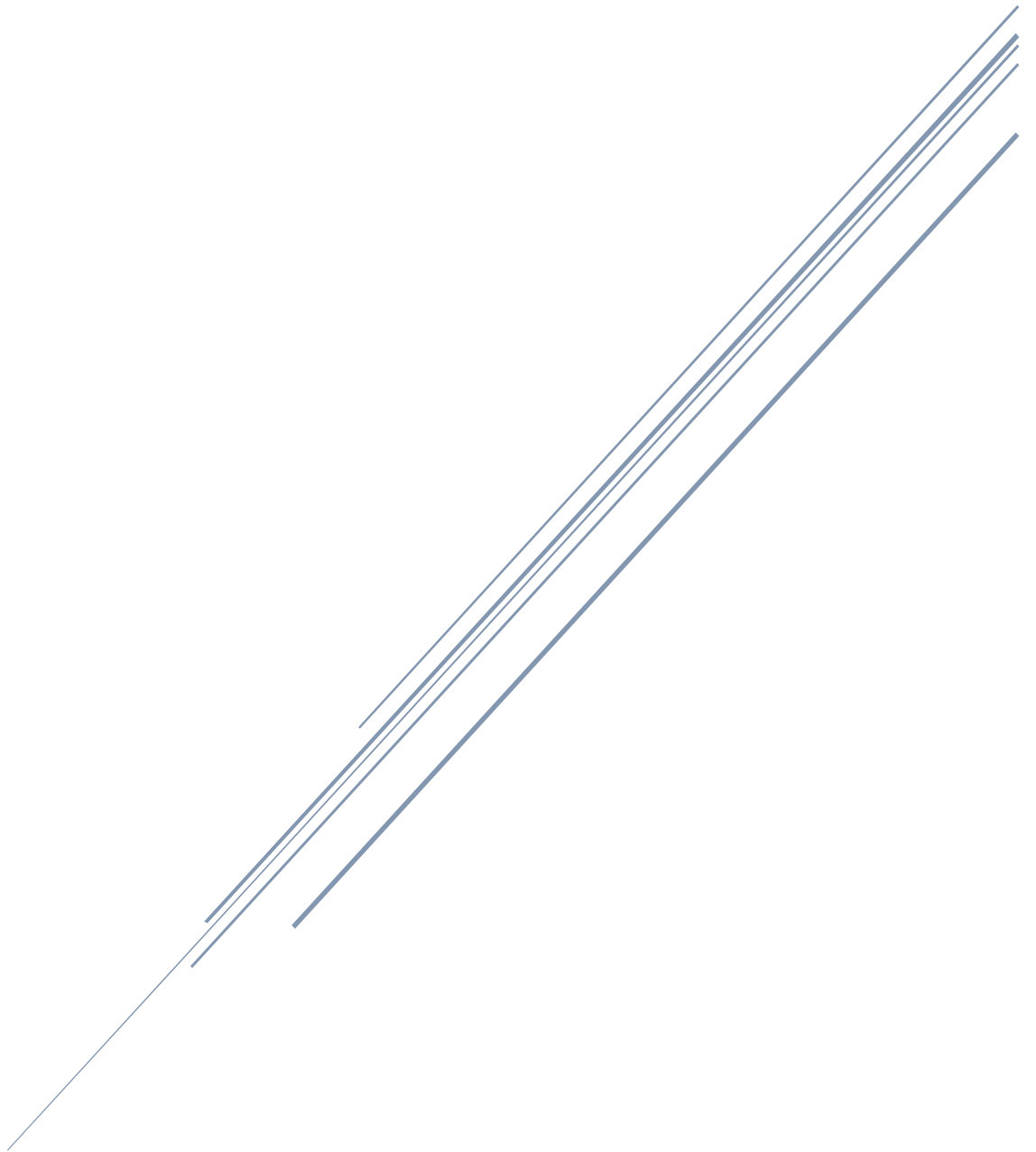


WEB DEVELOPMENT

Opdrachten



NHL

Joris Lops, Dick Bruin (2014)

Inleiding

Het maken van een webapplicatie is best een lastige taak. Dit heeft onder andere te maken met het feit dat er verschillende technieken en concepten gebruikt worden om een webapplicatie te bouwen. Het is dus een ontdekkingsreis langs de wereld van HTML, CSS, Database (SQL) & ASP.NET Webpages (C#).

Deze practicum opdrachten zijn bedoeld om je te laten oefenen met deze verschillende technieken en voor te bereiden op het P3P project. Ter ondersteuning zal het boek 'Beginning ASP.NET Web Pages with WebMatrix' gebruikt worden. Dit boek is speciaal gericht op beginnende softwareengineers en laat stap voor stap zien hoe je een webapplicatie ontwikkeld. De website van Microsoft over [ASP.NET Web Pages](#) bevat tutorials en video's en te leren werken met Web Pages. De eerste opdracht het maken van de introductie tutorial.

Om de EC's voor webdevelopment te behalen moeten alle opdrachten gemaakt en afgetekend worden! De eindopdracht opdracht wordt beoordeeld met een cijfer, wat het eindcijfer van dit het vak zal zijn. De EC's worden pas toegekend als de eerste opdracht is afgetkend en de eindopdracht is beoordeeld.

Het is aan te raden om de opdrachten met zijn tweeën te maken, dit betekent wel dat beide studenten het gemaakte werk moeten kunnen uitleggen aan de docent(en). We moedigen het aan om samen te werken, omdat je veel van elkaar kunt leren. Het is raadzaam bij het kiezen van een partner ervoor te zorgen dat het niveau ongeveer gelijk is. Anders bestaat het gevaar dat de een al het werk doet en de andere er niets van leert.

Het is zinvol om informatie op Internet te zoeken. De volgende verwijzingen bevatten nuttige documentatie, voor de gebruikte technieken:

- <http://www.asp.net/web-pages>
- <http://www.w3schools.com/>
 - <http://www.w3schools.com/html/default.asp>
 - <http://www.w3schools.com/sql/default.asp>
 - http://www.w3schools.com/aspnet/webpages_intro.asp
- <http://www.htmldog.com>

Vergeet niet: programmeren moet leuk zijn, happy coding ☺.

Week 1 - Opdracht 0: HTML & CSS

Om webapplicaties te kunnen maken is basiskennis van HTML/CSS nodig. Dit kun je leren:

- <https://www.codecademy.com/en/tracks/web>

Week 2 - Opdracht 1: Maken van statische website voor de eindopdracht

Voor de eindopdracht (zie opdracht 4) moet een hoop gedaan worden. Stap 1 is het maken van een statische website voor de User Interface. Maak hiervoor deze opdracht.

Lees opdracht 4 goed door, maak voor deze opdracht een statische website m.b.v. van HTML/CSS. Gebruik voor de layout een van de volgende technieken:

- Materialized Design (<http://materializecss.com/>)
- Bootstrap (<https://getbootstrap.com/>)

Met statische website wordt het volgende bedoeld: de website bevat geen code op de server en geeft alleen de User Interface weer in HTML/CSS m.b.v. Bootstrap of Materialized Design.

Later wordt de server code toegevoegd om de applicatie functionaliteit te geven.

Je kan voor nu gewoon zelf dummy gegevens bedenken en deze in de html website laten zien.

Door eerst de UI te ontwikkelen is het makkelijk om te bekijken of alle functionaliteit in je website is ondergebracht (met andere woorden ben je geen functionaliteit vergeten) en of de UI bruikbaar is voor (eind)gebruikers.

Week 3 - Opdracht 2: Inleiding ASP.NET Web Pages

De ontwikkelaar van ASP.NET Web Pages is Microsoft. Op de product webpagina van ASP.NET Web Pages staat een introductie tutorial.

[Introducing ASP.NET Web Pages \(9 Tutorials\)](http://www.asp.net/web-pages/overview/getting-started/introducing-aspnet-web-pages-2/getting-started)

(<http://www.asp.net/web-pages/overview/getting-started/introducing-aspnet-web-pages-2/getting-started>)

De opdracht is om deze tutorials te maken. Dit is een goed start voor de eindopdracht.

Deze opdracht bestaat uit 9 onderdelen. Mijn verwachting is dit aardig wat tijd kan kosten, maar dat het zeker de moeite waard is! Omdat de belangrijkste technieken en concepten de revue passeren om een web applicatie te kunnen ontwikkelen m.b.v. ASP.NET Web Pages.

Onderdeel 9 van deze tutorial reeks is optioneel.

Week 3/4 - Opdracht 3: State management

Er zijn verschillende state management technieken. State management is een duur woord voor het doorgeven van gegevens tussen verschillende HTTP-Request (web pagina's). De onderstaande URL beschrijft de verschillende technieken die ASP.NET Web Pages heeft voor state management.

<https://www.mikesdotnetting.com/article/192/transferring-data-between-asp-net-web-pages>

Het idee is dat jullie de verschillende technieken gaan gebruiken in een website die uit 5 pagina's bestaat en die een eenvoudig bestel proces voorstelt. Dit proces wordt gevisualiseerd in de afbeelding op de volgende pagina. In deze afbeelding is te zien welke state management techniek waar gebruikt moet worden.

Het idee is dat de verschillende state management technieken worden gebruikt zoals in het artikel van mikedotnetting staan. In de afbeelding is gevisualiseerd waar je welke techniek moet gebruiken.

Na het maken van de opdracht moet je de volgende vragen kunnen beantwoorden bij het aftekenen.

-Waarvoor is state-management noodzakelijk?

-Wat is het belangrijkste verschil tussen de verschillende technieken?

-Geef een voorbeeld van de gebruikte state management technieken en waar en in welke situatie deze toepasbaar zijn. Kan je de gebruikte technieken aanwijzen op de website van bol.com?

-Kan je er achter komen hoe sessies onderwater werken: welke state management techniek gebruiken sessies onderwater, om aan de juiste gebruiker de juiste sessions state aan te bieden? Tip: bekijk de request in Chrome Developer tools?

Let op: AddAdress.cshtml doet eerst een keer een post naar zichzelf.

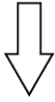
Er staat op de volgende pagina een schematische weergave!

1 Login.cshtml

Gebruikersnaam

Wachtwoord

State Management Techniek

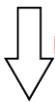


2 AddProduct.cshtml?username=Joris&password=geheim

Selecteer een product
Bal

Gebruikersnaam
Joris
Password
geheim

Query Strings - GET



3 AddAddress.cshtml

Straat

Nummer

Plaats

Gebruikersnaam
Joris
Password
geheim
Product
bal
true

Hidden Form Fields - POST

Hidden Form Fields
POST

Session Variabelen - GET

3a Gegevens Opslaan in Session

Gegevens uit Post halen en opslaan in Session Variabelen

3b Redirecten naar volgende pagina

```
if (IsPost &&  
    Request.UrlReferrer.AbsolutePath == Request.Url.AbsolutePath)  
{  
    //opslaan van gegevens in session  
    Response.Redirect("Overview.cshtml");  
}
```

4 Overview.cshtml

Bestel Overzicht
Gebruikersnaam
Joris
Password
geheim
Product
bal
Straat
Stationsweg
Nummer
10
Plaats
Leeuwarden

Session Variabelen &
Query String - GET

Place Order



5 OrderComplete.cshtml?status=cancel

Session leeg maken

5 OrderComplete.cshtml?status=ok

Gegevens opslaan in database
en weergeven

Bedankt voor de bestelling!
Gebruikersnaam: Joris
Wachtwoord: Geheim
Product: bal
Straat: Stationsweg
Nummer: 10
Plaats: Leeuwarden



Session leeg maken

Session Variabelen & Query String - GET

Week 4/5 - Opdracht 4 – Eindopdracht - Hebberige bruiden.

In week 6 volgt een kort assessment waarin je de applicatie demonstreert en de code uitlegt aan een docent.

Deze eindopdracht demonstreer je in een kort assessment bij een docent en leg je de technische werking uit. T.z.t. zal er een intekenlijst komen om je in te schrijven voor het assessment (begin week 6 zal dit waarschijnlijk plaats vinden).

Bij een bruiloft is het de gewoonte een cadeau(tje) te geven. Maar hoe zorg je ervoor dat je geen dubbele geschenken krijgt. Dit is natuurlijk niet wat een hebberig bruidspaar wil. Daarom wil weddingplanner “a Special Day” een web applicatie waarin een bruidspaar een wensenlijst kan samenstellen.

Het idee is als volgt. Het bruidspaar maakt een lijst aan van de geschenken die ze hopen te krijgen van de gasten. Ze zijn natuurlijk wispelturig (zo gaat dat met nog niet getrouwde stellen), dat ze met enige regelmaat nieuwe geschenken willen toevoegen of verwijderen. Ze moeten ook de volgorde kunnen veranderen. Een cadeau dat boven aan in de lijst staat willen ze super graag hebben. Een cadeau wat onderaan staat minder graag.

Een bruidspaar moet een account kunnen aanmaken en kunnen inloggen. Er zijn natuurlijke meerdere bruidsparen die gebruik gaan maken van deze webapplicatie. Als een bruidspaar zich registreert krijgen ze een unieke code. Deze unieke code geeft het bruidspaar aan de genodigden.

Deze gebruikt deze code om de juiste wensenlijst op te vragen. Vervolgens kan iedere gast aangeven welke cadeaus hij wil kopen. Als een genodigde een cadeautje koopt is deze niet meer zichtbaar voor de andere gasten. De geschenken blijven uiteraard wel zichtbaar voor het bruidspaar.

Het is de bedoeling om met behulp van ASP.NET Webpages een simpele webapplicatie te maken.

LET OP! Alvorens je begint is het aan te raden ~~verplicht~~ om het database ontwerp te laten goedkeuren bij het practicum van databases!

Belangrijke opmerking:

- Gebruik validatie om te controleren of gebruikers correcte invoer geven
- **Gebruik** Membership and Roles Management om ervoor te zorgen dat bruidsparen veilig kunnen inloggen. Zie hoofdstuk 12 voor details. Je mag uiteraard ook je eigen systeem maken. Of een inlog systeem faken, dit zou je kunnen doen door gebruik te maken van unieke code voor ieder bruidspaar (de unieke code).
- Mijn tip: gebruik Membership and Roles Management, of indien je Web Pages programmeren lastig vindt fake het inlogsysteem dan!
 - Zie voor uitleg
 - <https://www.asp.net/web-pages/overview/security>
 - http://www.w3schools.com/asp/webpages_security.asp
 - Hoofdstuk 12 van het boek
 - Als je zelf een login systeem maakt, zijn we genereus en kennen een extra hoog eindcijfer toe, zie opdracht 3 voor uitleg.

Boek: Hoofdstukken 3, 5, 10, 11 en 12.