**Deel van de User story**: Als gebruiker wil ik in kunnen loggen, zodat ik mijn winkelwagen en verlanglijst kan zien.

User wil:

* Registreren
* Inloggen
* Uitloggen

Voor de code van de winkelwagen en het verlanglijstje, kijk naar de documenten ‘Functionele documentatie verlanglijstje’ en ‘Functionele documentatie winkelwagen.’

Voor de site is een inlog functie nodig. Dit is deel van de conversie verhogende maatregelen. Als een bezoeker iets besluit te kopen, dan kan hij of; inloggen en zijn gegevens opslaan, of zijn NAW-gegevens achterlaten. De bezoeker logt dus in en zijn informatie staat in een database. De aflever-gegevens en dergelijk worden automatisch ingevoerd. Hiervoor moet de klant eerst registreren.

**Registratie (Register.php)**

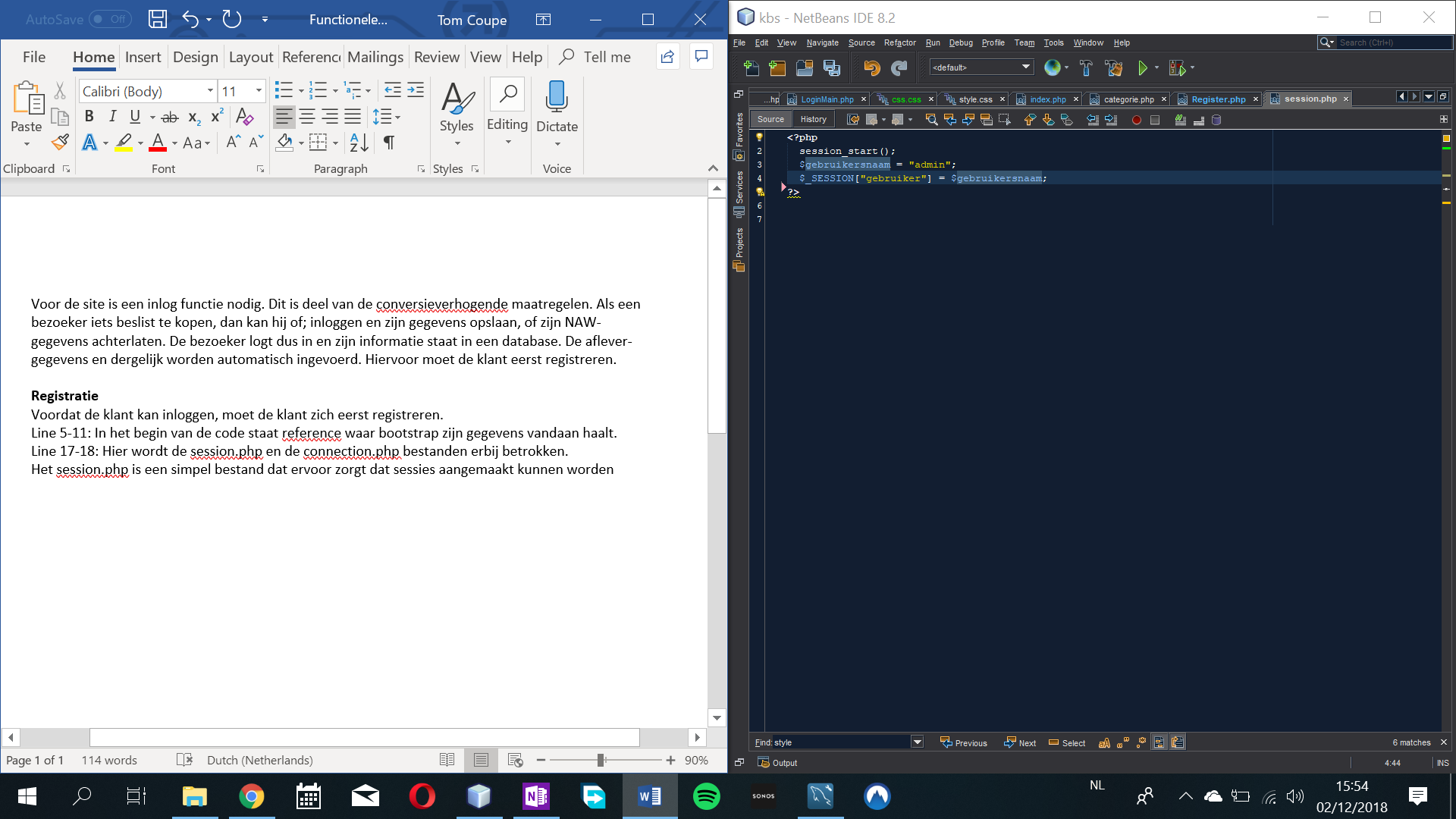
Voordat de klant kan inloggen, moet de klant zich eerst registreren.

Line 5-11: In het begin van de code staat reference waar bootstrap zijn gegevens vandaan haalt.

Line 17-18: Hier wordt de session.php en de connection.php bestanden erbij betrokken.

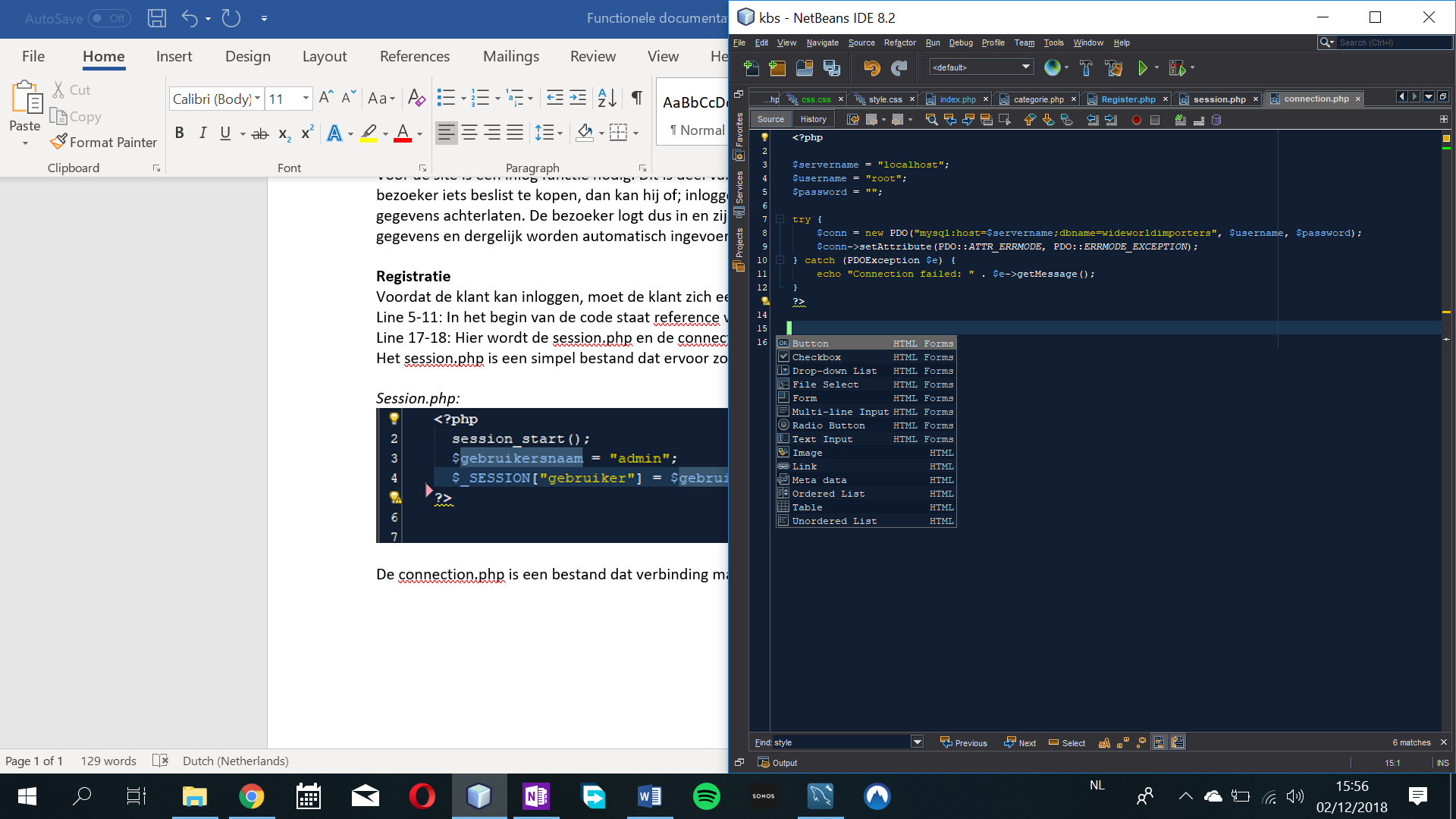
Het session.php is een simpel bestand dat ervoor zorgt dat sessies aangemaakt kunnen worden.

*Session.php:*



De connection.php is een bestand dat verbinding maakt met de MySQL database d.m.v. PDO.

*Connection.php:*



Line 22-26: Hier wordt een ID aangemaakt voor de nieuwe klant. Om zeker te weten dat er altijd een nieuw en ongebruikt ID wordt gegeven aan de gebruiker zoeken we het hoogste ID-nummer in de database en doen er één bij op. Dit Unieke ID wordt later in de database gezet.

Line 28: Als op de knop registreren gedrukt wordt, dan wordt de onderstaande code uitgevoerd.

Line 29: Hier wordt gekeken of de twee wachtwoorden die ingevoerd zijn hetzelfde zijn, als dit niet het geval is dan krijgt de gebruiker een error message.

Line 31-43: De ingevoerde informatie wordt in een variabele gezet. Ook wordt het wachtwoord gehashed.

Line 45-56: Als de verplichte velden zijn leeggelaten, dan wordt een foutmelding weergeven en de ingevoerde informatie wordt niet in de database gezet. Dit kan pas wanneer alle verplichte velden zijn ingevuld.

Line 58-62: Wanneer de gebruiker aangeeft in welke stad hij/zij woont, dan wordt het in de database gezet. Maar als de stad niet bestaat, dan wordt hier het unieke stadsID aangemaakt om later in de database ‘cities’ te zetten.

Line 64-68:Hier wordt gekeken of de stad die is ingevoerd is in de tabel ‘cities’ staat. 0 betekent dat de stad niet in de database staat. 1 betekent dat er wel een stad in de database staat.

Line 70-75: Sinds het emailadres in de people-tabel de primary-key is, moeten we kijken of het ingevoerde email-adres in de tabel staat. 0 betekent dat het email niet bestaat. 1 betekent dat het email-adres al in de database zit.

Line 77: Hier wordt de datum gecreëerd die later in de database gezet wordt als datum account creatie.

Line 78-79: Eerst wordt gekeken of het ingevulde email-adres bestaat. Als dit het geval is, wordt vervolgens gekeken of de ingevulde stad bestaat. Als dit het geval is dan wordt de stad aangemaakt in de ‘cities’-tabel en de andere ingevulde informatie in de database gezet. Als het email-adres bestaat, wordt dit aangegeven d.m.v. een error message. Als de stad bestaat, dan wordt deze niet aangemaakt sinds hij al in de database staat.

Line 94 & 110: Wanneer de informatie in de database is gezet, dan wordt de header aangepast. Dit is handig omdat de pagina niet vernieuwd wordt.

Line 125- 128: Als alle registratie voldaan is, dan wordt gekeken of de header is aangepast. Als dit het geval is, dan komt er een link te staan die naar de login pagina leidt.

Line 143-174: Dit zijn de vakje waar de informatie in gezet wordt.

Line 178-182: Hier wordt de bootstrap script gezet en het einde van de html.

**Inloggen (LoginMain.php)**

Line 3-12: Dit verwijst naar de style waar bootstrap gebruik van maakt.

Line 19-21: Hier worden de php paginas; session.php en connection.php betrokken.

Line 22-31: Als op de knop inloggen wordt geklikt, dan wordt er gecontroleerd of een veld is leeggelaten.

Line 35-38: Deze query haalt alle informatie op uit de database die overeenkomt met het ingevulde wachtwoord.

Line 39: Hier wordt de informatie uit de query in een array gezet.

Line 41-48: Hier wordt het gehashte wachtwoord die in de database vergeleken met het wachtwoord dat bij het inloggen is ingevuld. Als deze overeenkomen, dan wordt informatie uit de query in sessies gestopt. Deze sessies zijn belangrijk voor functies die alleen toegankelijk zijn voor ingelogde gebruikers. Als de wachtwoorden overeenkomen dan wordt de gebruiker herleidt naar de index.php pagina, oftewel de hoofdpagina.

Line 49-52: Als de wachtwoorden niet overeenkomen, dan wordt aangegeven dat inloggen mislukt is.

Line 64-68: Hier wordt de kleur bepaald voor de error message.

Line 71-75: Dit is het inlogformulier. Deze informatie wordt in de bovenstaande query gebruikt om te controleren of de gebruiker bestaat en het goede wachtwoord in heeft gevuld.

**Uitloggen (logout.php)**

Line 4: Hier wordt de sessie gestart. Dit is nodig omdat we op deze pagina sessies stop gaan zetten.

Line 7-9: Hier worden de sessies beëindigt die relevant zijn aan het inlog-systeem.

Line 11: Zodra de benodigde sessies voor het inloggen zijn beëindigt, wordt de gebruiker herleidt naar de hoofdpagina (index.php).