

Solvas Fleet - Uitbreidingen

Steven Bastiaens
Benjamin Cousaert
Nils Mak
Niko Strijbol
Karel Vandenbussche
David Vandorpe
Domien Van Steendam

Contents

Uitbreiding	1
Beschrijving	1
Nieuwe klassen	2
CAS	2
Uitbreiding	2
Beschrijving	2
Nieuwe klassen en bestanden	3
Evaluatie	3

Uitbreiding

Beschrijving

Een mogelijke uitbreiding die in dit stadium toegevoegd zou kunnen worden, is het creëren van een gebruiker door middel van het inlezen van een identiteitskaart. Deze zou dan later ook kunnen dienen als authenticatie-alternatief.

Het proces om een gebruiker toe te voegen ziet er dan als volgt uit:

1. Een medewerker laat het systeem weten dat hij een gebruiker met eID wil toevoegen.
2. Een medewerker leest een eID in die hij van de gebruiker krijgt, en vult eventuele ontbrekende velden in.
3. De gebruiker kan nu inloggen met zijn eID.

Nieuwe klassen

Deze uitbreiding vereist aanpassingen in zowel de frontend als backend. Zo moet er grafische begeleiding zijn voor deze feature. Ook moet er in de backend ondersteuning komen voor het OpenID protocol, waarmee eID werkt.

Voor de authenticatie is het momenteel moeilijk inschatten hoeveel nieuwe klassen er vereist zijn, aangezien er nog geen authenticatiesysteem is. Voor OpenID zijn er echter plugins te vinden die met Spring Boot werken.

Er zou wel zeker een veld moeten komen bij de tabel **User**, die bijhoudt met welke OpenID account hij geassocieerd dient te worden. Voor de rest hopen we zoveel mogelijk aan de gebruikte plugin toe te vertrouwen.

CAS

Er kan ook gebruik gemaakt worden van een eigen CAS-server, om dan via het principe van “Authentication Delegation” meerdere authenticatie alternatieven aan te bieden naast het klassieke gebruikersnaam en wachtwoord. Denk bijvoorbeeld aan Facebook login, eID, Twitter. Dit vereist iets meer werk, maar we verwachten dat zeker Facebook login een waardevolle feature zou zijn.

Uitbreiding

Beschrijving

Een mogelijke uitbreiding die in dit stadium al toegevoegd kan worden is het controleren en/of opvragen van informatie uit externe databanken. Het voorbeeld dat hier besproken wordt, gebruikt de informatie voortvloeiend uit het BTW-nummer (uit bv. VIES).

In grote lijnen is het de bedoeling om er zeker van te zijn dat het BTW-nummer correct is. Hiervoor wordt het toevoegen van een nieuw bedrijf in 2 stappen verdeeld. Het proces om een bedrijf toe te voegen ziet er dan als volgt uit:

1. De medewerker van Solvas voert het BTW-nummer in van het toe te voegen bedrijf.
2. De server gebruikt dit BTW-nummer en zoekt dit op in een externe databank, zoals VIES.
3. Als het nummer klopt stuurt de server het nummer terug, samen met de andere informatie uit die databank (zoals het adres).
4. De medewerker komt op de huidige pagina om bedrijven toe te voegen, maar de gevonden gegevens zijn al ingevuld, en kunnen eventueel verbeterd worden.

Als bij stap 3 het BTW-nummer niet correct blijkt, wordt een foutboodschap gestuurd en wordt gevraagd om het nummer te verbeteren, of om eventueel toch door te gaan (hierbij is gedacht aan de mogelijkheid dat de externe databank zelf verkeerd of onvolledig is).

Nieuwe klassen en bestanden

Deze uitbreiding kan zowel in de frontend als backend geïmplementeerd worden. Er is gekozen om ze volledig op de frontend te implementeren, omdat de frontend sowieso toch aangepast moeten worden, en op deze manier niets aan de backend moet veranderen. Ook aan de API moet dan niets gewijzigd worden.

Er is geen probleem met het toevoegen aan de frontend: deze controle maakt geen deel uit van het probleemdomein, noch van de API. Het is slechts een uitgebreid hulpmiddel bij het invullen van de gegevens, conceptueel gelijkaardig aan bv. het rood worden van het e-mailinvuerveld als de e-mail niet het juiste formaat heeft, behalve dat deze controle volledig optioneel is: voor ons domein maakt het niet uit of het BTW-nummer al geregistreerd is of niet. Uiteraard is het wel mogelijk om deze vereiste toe te voegen aan het domeinmodel, maar dat doen we hier niet.

Een lijst van nieuwe klassen/bestanden:

- `src/javascripts/components/clients/AddClientVat.tsx`, een formulier waar het BTW-nummer opgegeven wordt door een medewerker. Dit zal uiteraard lijken op reeds bestaande formulieren, maar zal uit 1 in te vullen veld bestaan.
- `src/javascripts/actions/fetch_vat.ts`, het bestand dat de code bevat die de informatie ophaalt uit de externe databank.

Aan te passen klassen/bestanden:

- `src/javascripts/components/clients/Clients.tsx`, waar de knop voor nieuwe bedrijven aangepast moet worden om naar de nieuwe pagina `AddClientVat.tsx` te gaan.
- `src/javascripts/index.tsx`, waar de nieuwe pagina ook als route moet toegevoegd worden.

Evaluatie

In ons oordeel staat de uitbreiding wel tot de verhouding. Uiteindelijk is het niet veel werk, maar kan wel een redelijke meerwaarde bieden aan de gebruikers: ze krijgen minder typwerk. Misschien zelfs belangrijker, het helpt ook om fouten in het BTW-nummer (en de andere opgehaalde gegevens) te vermijden.