# Workshop devdays 2022

TODO: Kvk-Checker plugin

In deze workshop zullen jullie je eigen applicatie koppelen met Focus (SB+). Om de workshop uit te kunnen voeren heb je Visual Studio of VSCode nodig. Omdat we beperkte tijd hebben, hebben we alvast wat bestanden voor je klaargezet.

## Fase 1: De webapplicatie

Je begint met een nieuwe .NET webapplicatie te maken door in een terminal in de huidige map het volgende commando uit te voeren:

**dotnet new web**

Nu worden de bestanden die nodig zijn voor een .NET web applicatie voor je neergezet.

De volgende stap is het toevoegen van de landingpage. Dit is de pagina waarnaartoe Focus navigeert vanuit het “App Center”. Hierbij wordt de url van Focus doorgegeven. Voeg aan program.cs de volgende regel toe voor app.Run():

**LandingPage.Register(app);**

Als je LandingPage.cs bekijkt (deze staat in WorkshopLibrary), dan zie je dat de pagina een url ontvangt om mee te communiceren. Omdat elke focus klant zijn eigen url heeft met daarin de tenant (accountant) en scope (klant) is het niet mogelijk om vooraf in de applicatie configuratie de url van Focus vast te leggen.

Voor nu hebben we even genoeg gedaan aan onze webapplicatie. Voer het commando **dotnet run** uit in een terminal of start de webapplicatie via de IDE. Je webapplicatie opent nu in een browser. Onthoud het poortnummer waarop de applicatie draait.

## Fase 2: De registratie in Focus

Focus kent het “app center” waar alle applicaties geregistreerd staan die gekoppeld kunnen worden. Deze lijst wordt door AFAS beheerd, maar gelukkig kunnen we via een “AFAS import” hier een extra applicatie aan toevoegen.

In de import folder staan de bestanden die we nu gaan importeren. Open AppClient.csv. Maak nu de volgende aanpassingen:

* Zorg voor een unieke Name (zodat je geen conflicten krijgt met andere groepjes)
* Verzin een geheime ClientSecret om te voorkomen dat iemand anders je koppeling misbruikt
* Wijzig het poortnummer in de ClientRedirectUrl en ClientLandingPageUrl naar het poortnummer van jouw applicatie
* Sla deze wijzigingen op

Jullie groep heeft een url gekregen waarop jullie focus applicatie draait. Open deze in een browser, open de spotlight via F3 en ga naar Import. Ga in de lijst-switcher naar ‘Uitgevoerde AFAS imports’ en klik hier op Meer 🡪 Importeer AFAS data.

Importeer nu alle bestanden die in de import map staan en geef een unieke omschrijving

Graphical user interface, application

Description automatically generated

Na een paar seconden zie je als het goed is dat je import gelukt is. Ga nu via F3 naar App Center en klik op jouw applicatie. Aan het einde van de url staat nu een guid. Dit is de clientId die je nodig hebt in de volgende fase, dus zet deze even op je clipboard. Klik nu op “Activeren”. Als het allemaal goed gegaan is zie je nu jouw landing page.

## Fase 3: Autoriseren

Nu gaan we de autorisatie toevoegen aan de webapplicatie. Stop je webserver en voeg in program.cs voor builder.Build() de volgende regel toe (je vult hier je clientId en clientSecret uit stap 2 in):

**FocusAuthorization.SetUp(builder, "***clientId***", "***clientSecret***");**

De FocusAuthorization class staat in de folder WorkshopLibrary. Deze bevat de configuratie die nodig is voor de OAuth autorisatie met Focus.

Nu gaan we een webpagina toevoegen die de functionaliteit gaat uitvoeren. Omdat dit een langlopend proces is gebruiken we een websocket. Voeg de volgende regels toe voor app.Run()

RunPage.Register(app);

RunScriptPage.Register(app, async (Log) => {

  await Log("Executing workshop script version 0.0");

  await Task.Delay(1000);

  await Log("Done");

});

Deze pagina’s zijn geautoriseerd, dus als de gebruiker hier naartoe navigeert, wordt eerst gecontroleerd of er gekoppeld is met Focus en wordt de koppeling tot stand gebracht als die er nog niet is.

Start nu de web applicatie opnieuw en ga verder vanaf de landing page. Je komt nu als het goed is op de run pagina terecht waar je ziet dat je script hierboven uitgevoerd wordt.

## Fase 4: Gegevens ophalen uit Focus

Voor de koppeling gaan we organisaties ophalen uit Focus om deze vervolgens te vergelijken met de gegevens uit de KVK. Hiervoor maken we een nieuwe get-connector aan in Focus. Ga hiervoor in Focus naar “applicatie onderdelen”, doe hier Nieuw 🡪 Get connector

Vul hier deze gegevens in, waardoor je straks de url /api/organisaties-workshop kunt aanroepen:  
Graphical user interface, application

Description automatically generated  
TODO: InstanceId

## Fase 5: Analyse script maken

Nadat de koppeling tot stand is gebracht heeft je webapplicatie een access\_token gekregen waarmee de Focus API’s aangeroepen worden. Normaliter wordt deze opgeslagen in de database van de webapplicatie, maar door de manier waarop je webapplicatie nu geconfigureerd is staan deze gegevens in een cookie.

In de map WorkshopSnippets staan wat stukken code waarmee je met behulp van de access\_token de get-connector van Focus kunt aanroepen. Gebruik deze om de lijst met organisaties te downloaden.

Gebruik vervolgens de code snippets van Kvk om organisaties die een kvk nummer hebben op te zoeken bij de Kvk. De apikey van de kvk die je nodig hebt heb je via mail van ons ontvangen.

## Fase 6: Update connector

Focus kent het concept van signalen, dit zijn berichten van Focus aan de gebruiker die een mogelijk probleem signaleren. Sommige signalen verdwijnen als ze opgelost zijn en andere signalen kun je handmatig afmelden. In deze workshop gaan via een update-connector een signaal aanmaken voor elke organisatie die niet bekend is bij de KvK.

De JSON die we via deze update-connector willen versturen willen we de volgende gegevens laten bevatten:  
  
{   
 "organization": "2ad55b99-d3cf-41b8-8126-e443dc1bb0ef",  
 "message": "De organisatie \*\*\* is niet bekend bij de KVK"  
}

Ga nu in Focus naar “Applicatie onderdelen” en voeg een nieuwe updateConnector toe met de json-schema uit de WorkshopSnippets map die de json hierboven beschrijft.

Graphical user interface, text, application

Description automatically generated

In het veld “Transformatie naar entiteit” beschrijf je in ATF template formaat ([zie hier](https://dev.azure.com/afassoftware/Anta/_git/templating?path=/docs/ReadMe.md&_a=preview) voor de beschrijving van de mogelijkheden van ATF) hoe je de gegevens uit de request gebruikt om een entiteit aan te maken of te updaten. Je hebt hier via autocompletion (ctrl-spatie) de mogelijke entiteiten en velden beschikbaar.

TODO: snippets naar word

TODO: extra opdracht bronomschrijving fixen

TODO: openapi spec en aanroepen

TODO: client onderzoeken

<<template Template(source) format="json">>

{

"Signal": [

{

"Kind": "organization-signal",

"Id": "<<=\message.ToUuid()>>",

"Bron": {

"ComponentId": "Organisatie",

"Id": "<<=\organization>>"

},

"BronOmschrijving": "Een organisatie",

"Message": "<<=\message>>",

"IsLocked": false

}

]

}

<</template>>