

SDN

Nummer
152
dec 2024

SDN Magazine is
een uitgave van:

SDN 

MAGAZINE

Van en voor Microsoft & .NET professionals

De interactieve avatar met ChatGPT

Een innovatie in VR onderwijs



Retro computing
met moderne tools



Stop met het bouwen
van CRUD API's



Werken naar een
Security Culture



Save the date:
Future Tech 2025



THE IT CONFERENCE FOR
.NET & MICROSOFT TECHNOLOGIES

Sopra Steria is the main sponsor of Future Tech 2025!

We are a digital solutions provider for tailored, end-to-end corporate technology and software solutions, with over 56,000 employees across 30 countries. With digital tools and market expertise, we help our clients make bold choices and deliver results. We commit to innovation and collaboration and value business development. That's why we're the preferred technology partner of Europe's most successful companies.

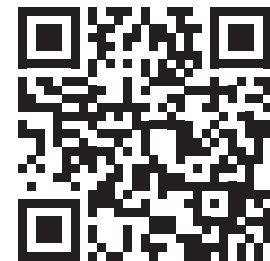
sopra  steria

Call4paper for Future Tech 2025 is now open!

After a completely sold-out edition in 2024, here's your chance to take the stage at Future Tech 2025.

On March 13, 2025, we'll explore the latest innovations in .NET together. At SDN, we're passionate about providing opportunities not just for seasoned speakers but also for those taking their first steps in public speaking. Think you're ready to share your expertise? Submit your session proposal today, and who knows, you could be shining on stage this March! After a completely sold-out edition in 2024, here's your chance to take the stage at Future Tech 2025.

On March 13, 2025, we'll explore the latest innovations in .NET together. At SDN, we're passionate about providing opportunities not just for seasoned speakers but also for those taking their first steps in public speaking. Think you're ready to share your expertise? Submit your session proposal today, and who knows, you could be shining on stage this March!



@futuretech_19



Future Tech



SDN (group)



@FutureTechNL



@futuretechnl.bsky.social



Colofon

Uitgave

Software Development Network
27ste jaargang
Nr. 152 • december 2024

Redactiecommissie:

Annejan Barelds, Jan de Vries,
John Bruin, Menno Jongerius,
Roy Janssen, Roelant Dieben,
Thijs den Hollander.

Auteurs:

Sander ten Brinke, John Bruin,
Sander Hoogendoorn,
Arco Nederveen, Milan Meyer

Listings:

Zie de website www.sdn.nl
voor eventuele source files uit
deze uitgave

Contact:

Software Development Network
info@sdn.nl
redactie@sdn.nl

Adverteren:

Informatie over adverteren en de
tarieven kunt u opvragen bij
Richelle Bussenius
rbussenius@reshift.nl

©2024 Alle rechten voorbehouden.
Niets uit deze uitgave mag worden
overgenomen op welke wijze dan
ook zonder voorafgaande schriftelijke
toestemming van SDN. Tenzij
anders vermeld zijn artikelen op
persoonlijke titel geschreven en
verwoorden zij dus niet noodzakelijkerwijs
de mening van het bestuur en/of de redactie.
Alle in dit magazine genoemde handelsmerken
zijn het eigendom van hun respectievelijke
eigenaren.

Beste SDN-ers,

Met veel plezier presenteren we een nieuwe editie van het SDN Magazine! Ook deze editie staat weer vol met waardevolle inzichten, technische verdieping en inspirerende artikelen.

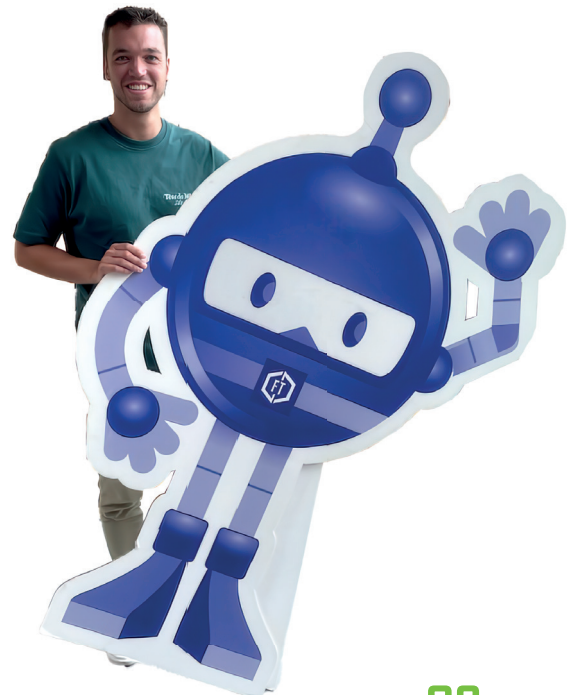
We willen u alvast uitnodigen om 'Future Tech 2025' in uw agenda te noteren! Op 13 maart 2025 duiken we samen in de nieuwste innovaties binnen .NET. Na een volledig uitverkochte editie in 2024, wilt u dit evenement niet missen. De Call for Papers is nu geopend, dus als u een sessie wilt geven – of dit nu uw eerste keer is of u een doorgewinterde spreker bent – grijp uw kans en dien uw voorstel in. Wie weet staat u binnenkort op het podium om uw expertise te delen!

We nemen u in dit magazine mee in de wereld van CRUD API's via Data API Builder, een krachtige manier om gegevens eenvoudig en efficiënt toegankelijk te maken. Daarnaast gaan we in op de toekomst van interactie met technologie in een artikel over de interactieve avatar die gebruikmaakt van ChatGPT. De combinatie van kunstmatige intelligentie en menselijke interactie opent deuren naar toepassingen die tot voor kort ondenkbaar waren.

Verder belichten we een belangrijk aspect van softwareontwikkeling met het artikel "The Sense and Nonsense of Pull Requests". Hoe houdt je dit proces constructief en waardevol? En er is meer: we hebben diverse onderwerpen die niet alleen uw technische kennis zullen verrijken, maar ook inspireren om de nieuwste ontwikkelingen in de praktijk toe te passen.

Tot slot wil ik iedereen bedanken die heeft bijgedragen aan deze uitgave: auteurs, redacteurs en natuurlijk u, onze leden. Veel leesplezier en tot ziens bij Future Tech 2025!

Simon de Groot
Community manager





werkenbijchipsoft.nl

Impact maken in de zorg-ICT?

Kom werken bij ChipSoft!



←
Bekijk hier
onze vacatures!

ChipSoft



In dit nummer



- 6** Retro computing met moderne tools
John Bruin
- 11** Stop met het bouwen van CRUD API's dankzij Microsoft's Data API Builder
Sander ten Brinke

- 16** De Interactieve Avatar met ChatGPT: Een Innovatie in VR Onderwijs
Arco Nederveen
- 20** The sense and nonsense of pull requests
Sander Hoogendoorn



- 24** Security Culture en Security Champions
Milan Meyer

6

Virtual reality (VR) en kunstmatige intelligentie (AI) veranderen de manier waarop we leren en trainen. Sopra Steria heeft samen met het Expertise Centrum Extended Reality van Landstede, een onderwijsinstelling die onder andere opleidingen tot militair verpleegkundige aanbiedt, een VR-applicatie ontwikkeld om studenten beter voor te bereiden op praktijksituaties. In deze VR-omgeving oefenen studenten medische procedures en communicatieve vaardigheden in een realistisch veldhospitaal.

11

Ben jij ondertussen ook wel klaar met het bouwen van CRUD API's? Ik ook! We zijn toch veel liever bezig met het oplossen van creatieve of technische problemen en het leveren van waarde? En tóch besteden we veel tijd aan het bouwen van standaard API's

24

Het is een begrip dat steeds vaker langskomt: Security Culture. Voor de meesten van ons die in IT werken als ontwikkelaar, binnen operations of als tester, is er wel enig idee wat hiermee wordt bedoeld. Maar in veel gevallen is dat idee wel beperkt tot het beeld binnen onze eigen rol. Als ontwikkelaars en binnen ops hanteren we bepaalde standaarden binnen implementaties. Als testers weten we dat we een negatief scenario moeten valideren.



Afbeelding 1

RETRO COMPUTING MET MODERNE TOOLS

Tijdens de coronapandemie kreeg ik door het ontbreken van reistijd en het ontbreken van allerlei avondactiviteiten, opeens heel veel extra vrijetijd. Voor mij was dit een reden om mijn oude hobby, de Commodore 64 weer op te pakken. Op Marktplaats heb ik weer twee mooi opgeknapte Commodore 64's, de oude breadbin (afbeelding 1) en de moderne C64c in lichtbeige uitvoering, en een Commodore 1701 kleurenmonitor op de kop weten te tikken. Met een Ultimate-II+ cartridge heb ik de diskdrive kunnen vervangen door een usb-stick, maar het nostalgische geluid van het draaien van de diskdrive wordt nog wel afgespeeld bij het inladen van een programma.

Auteur: John Bruin

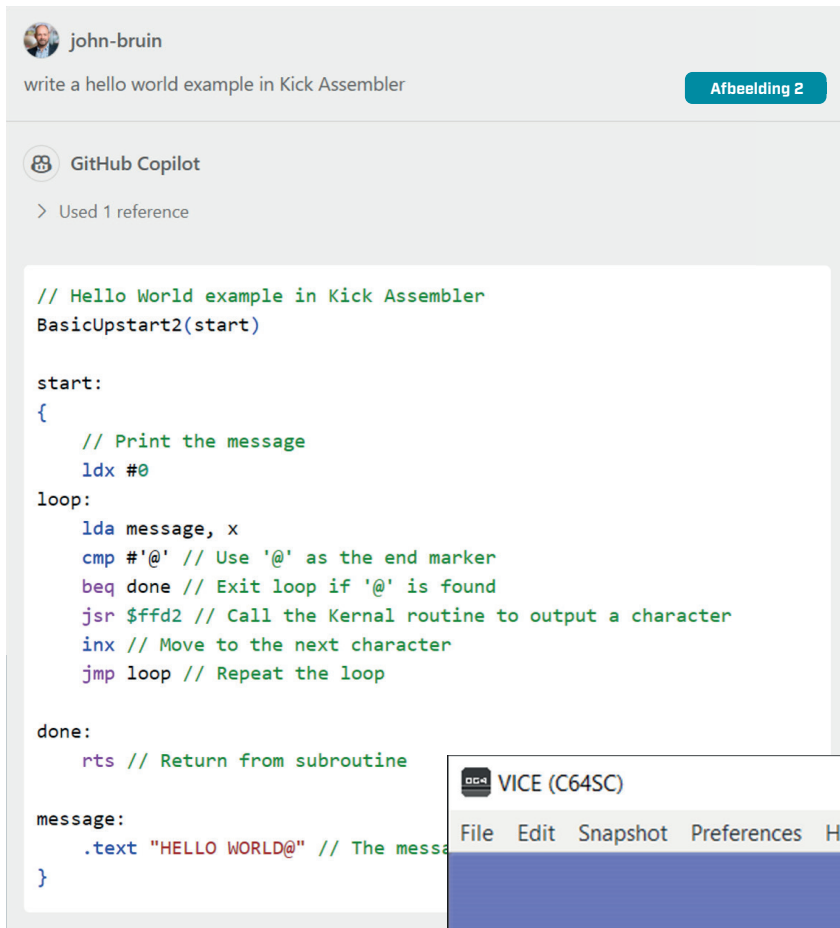
Ik heb vroeger wel in Commodore BASIC geprogrammeerd, maar met de moderne tools en emulators en de grote hoeveelheid informatie op het internet, kwam nu ook de drang om een stapje verder te gaan. Ik ben gaan programmeren in assembly, om zo echt alles uit de 40 jaar oude machine te kunnen halen.

Moderne tools

Vroeger moest je alles op de Commodore 64 (hierna ook afgekort als C64) intikken en compileren, dat was erg omslachtig met het beperkte geheugen en performance. Gelukkig kunnen we tegenwoordig de pc gebruiken. Met een emulator en een moderne debugger kun je het resul-

taat direct op de pc bekijken, maar via een netwerkkabel en FTP kun je het resultaat ook heel snel naar de echte C64 uploaden en testen, want nothing beats the real thing! Na wat online research kwam ik op deze moderne en voor mij vertrouwde ontwikkelomgeving met Visual Studio Code aangevuld met een aantal handig tools:

- > Visual Studio Code, de populaire code-editor van Microsoft
- > Kick Assembler (afgekort "KickAss"), een moderne assembler met scripttaal voor 6510 machine code (geschreven in Java)
- > Kick Assembler 8-Bit Retro Studio VSCode extension, met o.a. intellisense, code outlines, code snippets, code scoping, imports, breakpoints en debug logging
- > Exomizer, compression tool (cruncher) om de executables kleiner te maken
- > VICE, the Versatile Commodore Emulator



van de C64 is nog altijd heel populair en duikt soms zelfs in moderne popmuziek nog op. In een groot aantal nummers van De Jeugd van Tegenwoordig zijn de karakteristieke geluidjes van de SID-chip nog duidelijk te herkennen, want producer en vierde bandlid Bas Bron is ook groot fan van de C64. Ook voor het maken van muziek voor de SID-chip zijn er verschillende moderne tools beschikbaar, zoals GoatTracker of SID Factory II, maar daar ga ik het in dit artikel niet over hebben.

Assembly code

Een goede start om assembly te leren is de website <https://codebase64.org>. Assembly voor

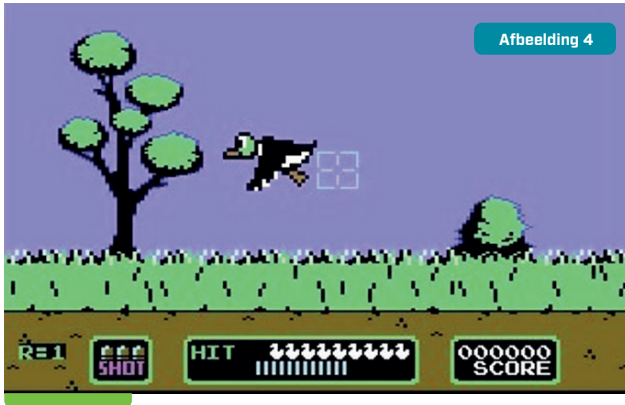
Hello World in assembly door Github Copilot

- > C64 Debugger
- > Multipaint, bitmap tekenprogramma
- > SpritePad, sprite tekenprogramma
- > CharPad, charset tekenprogramma
- > Make, om de verschillende build-stappen te automatiseren
- > GitHub, voor source-control van mijn projecten
- > GitHub Copilot, die zelfs de ruim 40 jaar oude 6510 assembly-code snapt

Met 64kB, een 1MHz cpu, 16 kleuren en een 320x160 resolutie is de C64 natuurlijk erg beperkt, maar die beperking maakt juist creatief! De SID-chip, de 3-stemmige muziekchip



Output van Hello World in VICE voor Windows



Duck Hunt voor de Commodore 64.

de 8 bit 6510 processor bestaat uit een beperkte set 3-letterige instructies waarmee je bijvoorbeeld de waarden van de X en Y registers, de accumulator A en van geheugenlocaties kunt lezen en schrijven. Uiteraard is dit artikel veel te kort voor een basiscursus assembly, maar het eenvoudige **Hello World** voorbeeld in afbeelding 2, geschreven door Github Copilot en de bijbehorende output in afbeelding 3 geeft je misschien beter beeld.

Duck Hunt

Een paar maanden nadat ik mezelf een redelijk niveau van programmeren in assembly had aangeleerd, kwam ik op Marktplaats een vrij zeldzame light gun voor de C64 tegen. Ik kon er echter geen fatsoenlijke software voor vinden. Dus besloot ik om de Nintendo-klassieker Duck Hunt voor de C64 met light gun na te maken. Alle sprites en achtergronden heb ik met tools als Multipaint, SpritePad en CharPad nagemaakt (afbeelding 4). De muziek is door een vriend nagemaakt op een echte C64 en komt heel dicht in de buurt van de originele muziek. Het spelletje is op Transmission64, een online meeting in december 2022 gereleased en in de dagen daarna doken er verschillende YouTube video's op van mijn spelletje met honderden views!



Google Maps op de C64

De code met links naar de gebruikte tools kun je vinden op: <https://github.com/johnbruin/C64-Duck-Hunt>

Back to the future

Ik ben nu 54 jaar en ik had me als puber echt niet kunnen voorstellen dat ik nu, 40 jaar later, nog met de Commodore 64 zou spelen, maar ik ben nog lang niet uitgespeeld! Hieronder nog 3 voorbeelden van de nieuwste ontwikkelingen voor de C64.

WiC64

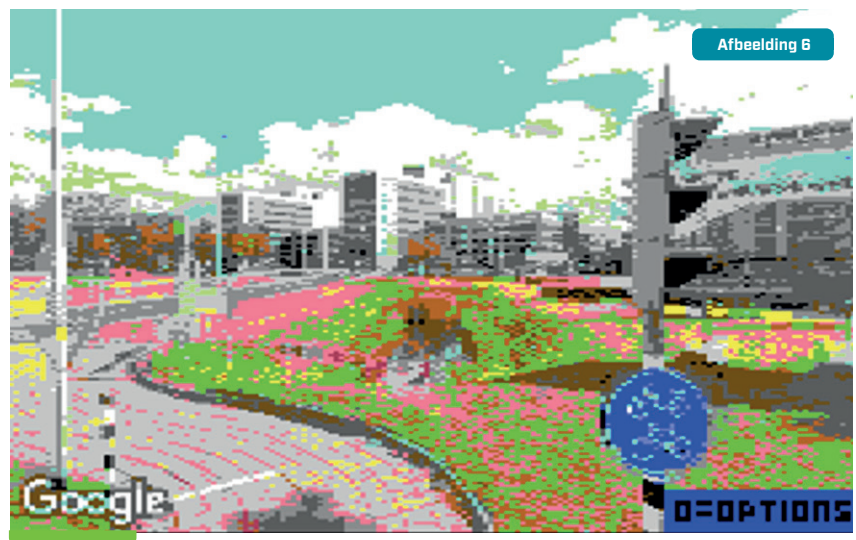
Onlangs heb ik op een bijeenkomst de WiC64 (<https://wic64.net>), The

Wireless Interface for the Commodore 64, gekocht. Dit is een uitbreiding die in de userport past waarmee een internetverbinding via wifi op de C64 mogelijk wordt.

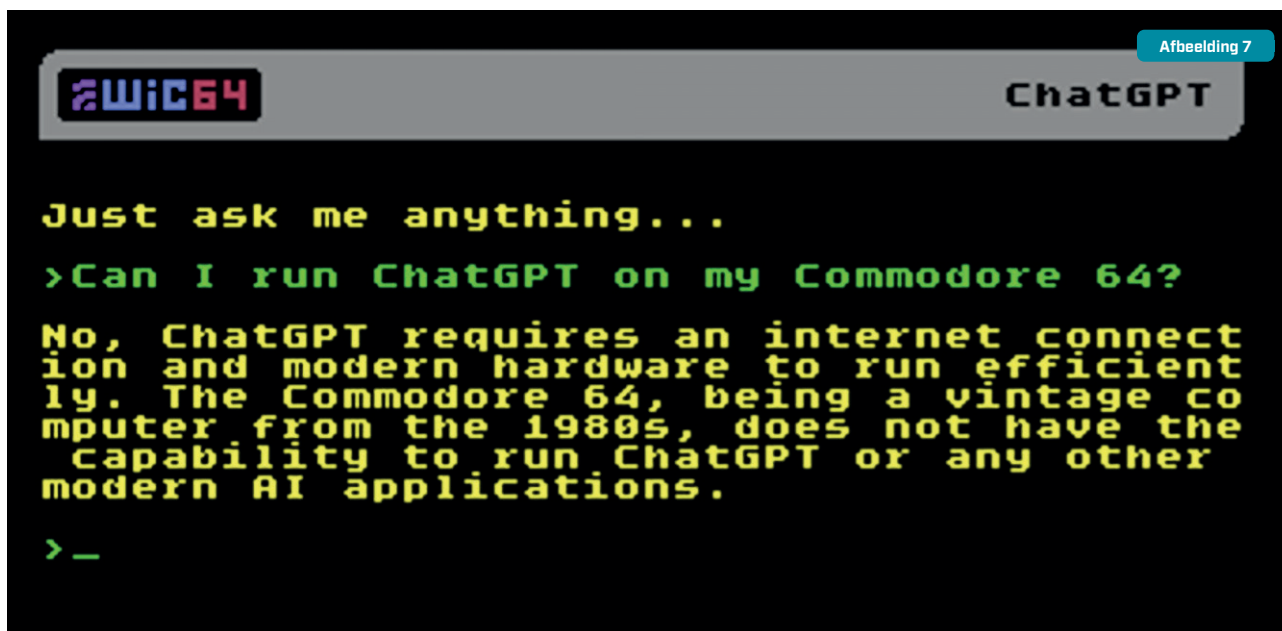
De reacties wanneer ik Google Maps (met Street View) of ChatGPT op deze meer dan 40 jaar oude computer aan mensen laat zien zijn geweldig! (Zie afbeelding 5, 6 en 7). Met de WiC64 is het nu dus ook mogelijk om web APIs aan te roepen en C64 programma's connected te maken.

C64 OS

In de jaren '80 was Geos (Graphic



Google Street View op de C64



ChatGPT op de C64

Environment Operating System) een populair grafisch OS voor de Commodore 64. Met hun GeoPaint, GeoWrite en GeoCalc, de voorlopers van Paint, Word en Excel waren ze hun tijd ver vooruit.

In 2022 kwam daar opeens C64 OS (<https://c64os.com>) een heel nieuw operating system voor de C64 met uitgebreide ondersteuning voor moderne hardware zoals SD-kaarten en usb-toetsenborden. Scott Hanselman van Microsoft, zelf ook groot C64 fan, heeft er zelfs een podcast aan besteed: <https://hanselminutes.com/860>.

C compiler

Sinds kort is er ook een nieuwe geoptimaliseerde C compiler beschikbaar voor de C64: Oscar64, Optimizing Small space C Compiler Assembler and Runtime for C64 (<https://github.com/drmortalwombat/oscar64>) Het voordeel van C ten opzichte van assembly is natuurlijk dat het programmeren aanzienlijk eenvoudiger en overzichtelijker is. Het **Hello World** voorbeeld uit afbeelding 2 kan in C gereduceerd worden tot:

`printf("HELLO WORLD");` Assembly geeft altijd nog een veel betere performance en kleinere binary dan een gecompileerd C-programma. Dat komt onder andere door de beperkte adressering van de C64 van maximaal 2 bytes (\$ffff) en door de zeer beperkte ruimte op de stack van maar 256 bytes. Er is veel extra code nodig om bijvoorbeeld een eigen stack bij te houden voor lokale variabelen. Oscar64 is hiervoor geoptimaliseerd en heeft ook de optie om native assembly te gebruiken voor cruciale functies, maar desondanks heeft Oscar64 een performance penalty van 10-20% ten opzichte van assembly.

Oscar64 is opensource en bevat ook veel voorbeeld projectjes, dus zeker interessant om eens wat experimentjes mee te gaan doen!

Conclusie

De Commodore 64-scene is nog volop in leven en dit uit zich op de goedbezochte online en in-person events en op de actieve C-64 Scene Database (<https://csdb.dk>). Ook in Nederland bestaat er nog een grote actieve community en ik ben daarom ook weer lid geworden van de HCC Commodore gebruikersgroep. Leuk om te zien dat dit ook nog jonge mensen aantrekt die zelf helemaal niet met de Commodore 64 zijn opgegroeid. ●

BIO

John Bruin

John Bruin is een Microsoft .NET ontwikkelaar bij Sopra

Steria (voorheen Ordina) met meer dan 20 jaar ervaring. Hij blijft altijd nieuwsgierig naar de laatste ontwikkelingen in het vakgebied. Op dit moment zijn dat: .NET

Aspire, Blazor, AI en Mixed Reality, maar hij blijft ook trouw aan de goede oude Commodore computers waarmee de interesse ooit begon.

sopra  steria

*Happy
Holidays*



Stop met het bouwen van CRUD API's met Microsoft's Data API Builder

Ben jij ondertussen ook wel klaar met het bouwen van CRUD API's? Ik ook! We zijn toch veel liever bezig met het oplossen van creatieve of technische problemen en het leveren van waarde? En tóch besteden we veel tijd aan het bouwen van standaard API's

Auteur: Sander ten Brinke

Gelukkig hoeft dit niet meer dankzij een nieuw opensource-product van Microsoft genaamd *Data API Builder* (<https://github.com/Azure/data-api-builder>). In dit artikel leer je alles over dit toffe product en hoe je er mee van start kan gaan.

Wat is een 'CRUD' API?

Voordat we dieper in Data API Builder duiken, lijkt het mij handig om eerst iets beter uit te leggen wat we precies verstaan onder een CRUD API.

Wat ik bedoel met CRUD API, wordt ook wel een Data API genoemd; een

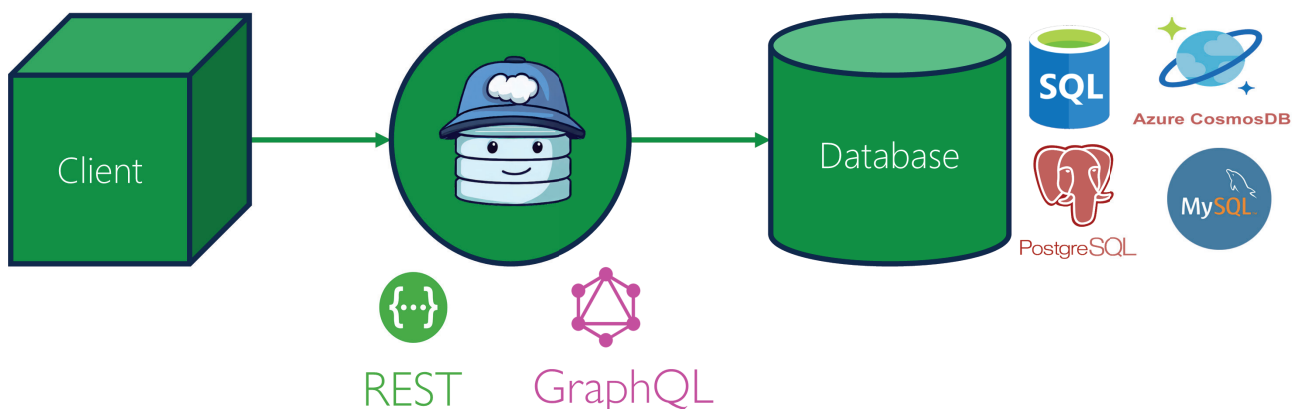
API die eigenlijk niets meer is dan een schil om een database. CRUD, dus, wat staat voor *Create Read Update Delete*. De API heeft een aantal REST of GraphQL endpoints waarmee je entiteiten kan ophalen, maken, bewerken, verwijderen, etc (afbeelding 1).

In principe is hier niets mis mee, maar we steken eigenlijk best veel tijd en moeite in het opzetten van zo'n API. Denk aan het opzetten van de repository, documentatie, het kiezen van je programmeertaal, framework en zo nog veel meer. Daarna zet je een databaseconnectie op, schrijf je endpoints om mee te verbinden en beveilig je deze met authenticatie en autorisatie, maar dan ben je er nog niet. Je zult waarschijnlijk ook iets van ci/cd implementeren zodat je nieuwe code gemakkelijk kunt releasen, waardoor je rekening moet gaan houden met package-, SDK- en beveiligingsupdates.

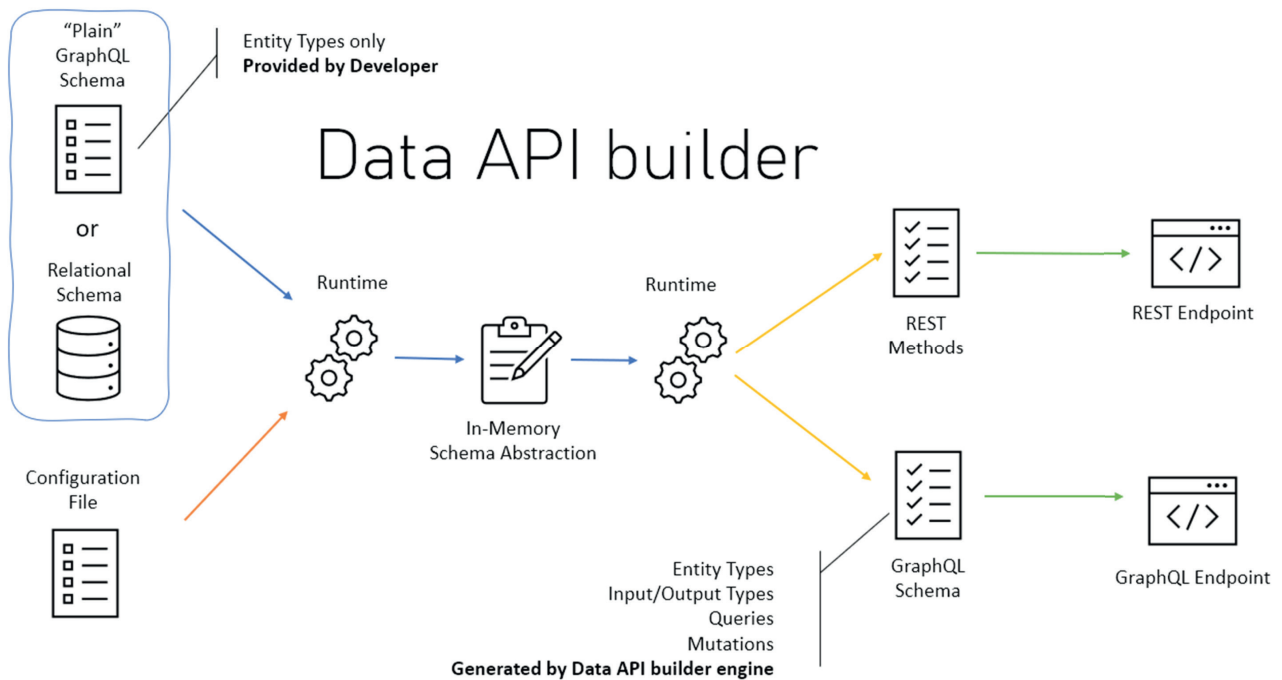
En dit allemaal voor een simpele API, niks meer dan een wrapper om je database heen? Wat een gedoe!

Introducing Data API Builder

En dan komt Data API builder in het spel. Data API Builder (DAB) is een zero-code, opensource-tool die in



Afbeelding 1.



Afbeelding 2.

milliseconden tijdens runtime een API voor je genereert op basis van je database-entiteiten (SQL of NoSQL). Het enige dat DAB nodig heeft, is een databaseverbinding en informatie over endpoints die gegenereerd moeten worden voor specifieke tabellen, views en/of stored-procedures.

Een overzicht van Data API Builder

Stel, je hebt een database met een `dbo.books`-tabel en een `dbo.authors`-tabel. Wanneer je deze entiteiten via DAB beschikbaar stelt, krijg je direct `/graphql`, `api/books` en `api/authors` endpoints waarmee jouw client data kan ophalen, aanmaken en aanpassen.

Maar dat is niet het enige dat DAB biedt! DAB zit namelijk boordevol features die je normaal gesproken zelf zou moeten bouwen:

- > Ondersteuning voor MySQL, PostgreSQL, SQL Server, Azure SQL en Azure CosmosDB NoSQL

- > Het genereert REST en GraphQL endpoints
 - > Het ondersteunt OData zodat veel voorkomende API-features al voor je gebouwd zijn
 - > Caching voor high-traffic-scenario's
 - > Veel opties voor authenticatie en autorisatie
 - > Een OpenAPI-document met SwaggerUI. Lees mijn blogpost over OpenAPI voor meer informatie 😊: <https://stenbrinke.nl/blog/openapi-api-client-generation/>
 - > Een GraphQL playground dankzij Banana Cake Pop (<https://chillicream.com/products/bananacakepop/>)
 - > Docker-support
 - > Integratie met Azure Static Web Apps
 - > en met Azure App Service in de toekomst
- En zo nog veel meer.

Onder het oppervlak gebruikt DAB veel opensource-componenten, zoals ASP.NET Core (<https://github.com/dotnet/aspnetcore>) voor het opzetten van de endpoints, Hot Chocolate (<https://chillicream.com/docs/hotchocolate/v13>) voor het GraphQL gedeelte en FusionCache (<https://github.com/ZiggyCreatures/FusionCache>) voor caching. Dit is handig, want updates van die technologieën komen dan ook direct in DAB terecht.

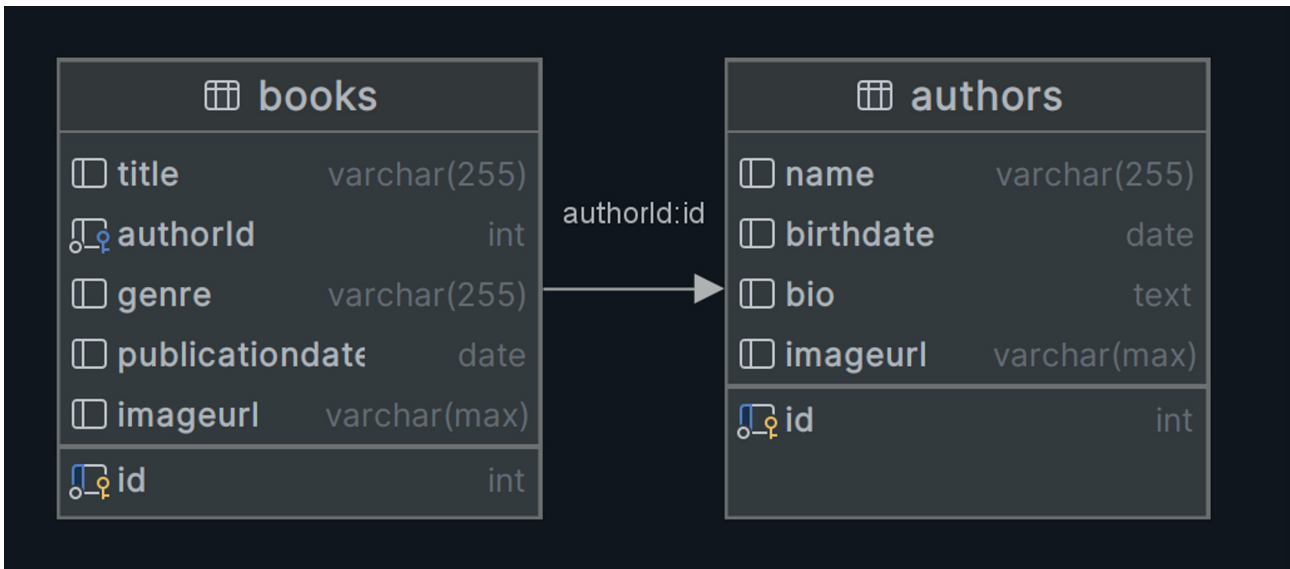
Oh, en misschien wel het meest indrukwekkende? Dit is niet gebonden aan een Microsoft platform. Je kunt het on-premise en in elke cloud draaien! Dus draai jij nog on-premise met een cloud-database of andersom? Dan kun je volledig gebruik maken van DAB.

Oh, en misschien wel het meest indrukwekkende? Dit is niet gebonden aan een Microsoft platform. Je kunt het on-premise en in elke cloud draaien! Dus draai jij nog on-premise met een cloud-database of andersom? Dan kun je volledig gebruik maken van DAB.

Architectuur

Aan de slag!

Je kunt op verschillende manieren gebruik maken van DAB (afbeelding 2). Je kan de .exe downloaden via <https://github.com/Azure/dab>



Afbeelding 3.

ta-api-builder/releases, maar nog makkelijker is om het via dotnet tool `install --global Microsoft.DataApiBuilder --global` te installeren.

Database model

Op de volgende manier stel je DAB in om endpoints te genereren voor de `dbo.books`-tabel die in Microsoft SQL Server draait (afbeelding 3). Als je mee wil doen met een echte database, gebruik dan mijn demo link aan het einde van dit artikel.

```
dotnet tool install Microsoft.DataApiBuilder -g
dab init --database-type mssql --connection-string "@env('sqlldb')"
```

- 1 Dit genereert een `dab-config.json` bestand met alle configuratie voor DAB.
- 2 De connectionstring voor de database wordt uit een environment variable `sqlldb` of een `.env` bestand gehaald. Wel zo veilig!

```
dab add Book --source "dbo.books" --permissions "anonymous:*"
```

DAB zit namelijk boordevol features die je normaal gesproken zelf zou moeten bouwen

- 1 DAB zal nu endpoints genereren op `api/books` voor de inhoud van de `dbo.books` tabel. Andere tabellen worden niet toegankelijk gemaakt.
- 2 Met `--permissions` vertellen we dat iedereen alle acties (Create, Read, Update, Delete) mag uitvoeren. Meer informatie over dit beveiligingssysteem kun je vinden in de documentatie, zie de links onderaan dit artikel.

`dab start`

- 1 DAB start nu op en genereert de REST en GraphQL endpoints in-memory (listing 1).

Je kan ook naar `/swagger` gaan voor SwaggerUI, naar `/api/openapi` voor het OpenAPI document, `/graphql` voor de GraphQL playground en `/` voor het health endpoint. Vergeet niet dat we ook OData kunnen ge-

```
curl https://localhost:5001/api/Book
[
  {
    "id": 1,
    "title": "Pronouncing Azure Correctly: The Definitive Guide",
    "authorId": 4,
    "genre": "Technology",
    "publicationdate": "2024-01-01",
    "imageurl": "https://i.imgur.com/Gx9VJp2.png"
  }
]
```

1

```

{
  // De rest van de configuratie zou hierboven staat hierboven
  "entities": {
    "Book": {
      "source": {
        "object": "dbo.books",
        "type": "table"
      },
      "graphql": { "enabled": true },
      "rest": { "enabled": true },
      "permissions": [
        {
          "role": "anonymous",
          "actions": [ { "action": "*" } ]
        }
      ]
    }
  }
}

```

2

dat is misschien niet gewenst. Daarom kun je ook GraphQL gebruiken. Eerst moeten we DAB vertellen dat er een relatie is tussen een Author en Book:

```
# Laat DAB endpoints genereren voor de authors tabel
```

```
dab add Author --source
"dbo.authors" --permissions
"anonymous:*"
```

```
# Een auteur kan meerdere boeken hebben
```

```
dab update author --relationship books --target.entity
Book --cardinality many
```

Om met GraphQL alle auteurs op te halen met daarbij de titel van alle gerelateerde boeken kun je hetgeen je ziet in listing 3 doen.

```

query {
  authors {
    items {
      name
      books {
        items {
          title
        }
      }
    }
  }
}
Met als output:
{
  "data": {
    "authors": {
      "items": [
        {
          "name": "Mark Russinovich",
          "books": {
            "items": [
              {
                "title": "Pronouncing Azure Correctly: The Definitive Guide"
              }
            ]
          }
        }
      ]
    }
  }
}

```

3

Integratie met Azure Static Web Apps

Zoals ik eerder al zei, is DAB niet gebonden aan een Microsoft-platform. Het is echter wel zo dat DAB goed integreert met Azure Static Web Apps. Dit is een serverless-hosting-platform voor web-apps zoals Hugo en Jekyll maar ook React, Angular, Vue, Blazor, etc.

In de meeste gevallen zal een web applicatie ook met een server communiceren. Hiervoor kun je Azure Static Web Apps gemakkelijk koppelen aan een Azure Compute service zoals Azure App Service, Azure Functions, Azure Container Apps en meer (afbeelding 4). Je krijgt dan een /api endpoint die alle requests doorstuurt naar de ingestelde Azure service. Vervolgens zou DAB kunnen draaien in Azure Container Apps dankzij de Docker support. Dit is echter niet heel praktisch want het opzetten van een Azure Container App om DAB in te

bruiken om data te filteren, sorteren en te pagineren.

Bijvoorbeeld: /api/Book?\$filter=genre eq 'Technology'.

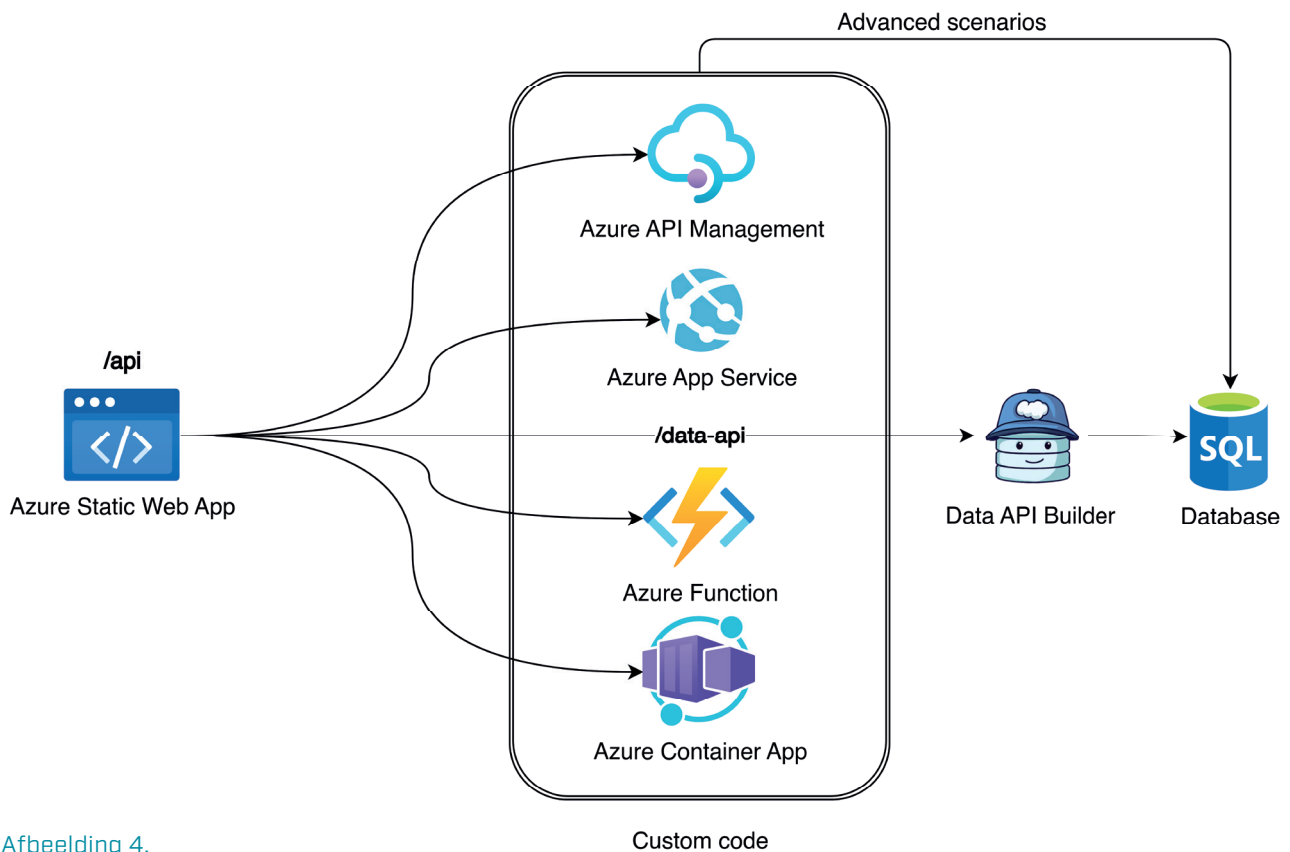
Configuratiebestand

De dab CLI werkt het configuratiebestand bij. Je kan DAB dus ook configureren door zelf het configuratiebestand aan te passen. Dit is een JSON-bestand met een JSON-schema waar de instellingen over de databaseverbinding, runtime, entiteiten en beveiliging instaan. De Book-entiteit die we zojuist hebben gemaakt ziet er als listing 2 uit in *dab-config.json*.

Relaties

Ik vind het indrukwekkend wat DAB allemaal doet in zo weinig tijd. Er is echter iets waar we het nog niet over hebben gehad, namelijk de relaties tussen entiteiten. Een boek is namelijk gerelateerd aan de auteur die het heeft geschreven. Wanneer je een auteur aan een gebruiker toont, wil je ook laten zien welke boeken deze auteur heeft geschreven. En dit kan ook met DAB.

Ten eerste zou je een View kunnen aanmaken in je database en DAB instellen om hier endpoints voor te genereren, maar hiervoor moet je je database aanpassen voor DAB, en



Afbeelding 4.

hosten is extra werk. Daarom heeft Azure Static Web Apps directe integratie met Data API Builder. Je kan een Static Web App linken aan een database in Azure en dan een Data API Builder-configuratiebestand instellen. Wanneer je een deployment doet van een Static Web App zullen requests naar /data-api automatisch naar Data API Builder worden gestuurd. Ontzettend handig!

Static Web App DAB integratie

Met deze setup kun je een web-app bouwen zonder een server te moeten beheren. Mocht je wel wat server-side-code nodig hebben, dan kun je dit gemakkelijk toevoegen door een Azure Function toe te voegen aan je Static Web App.

Meer informatie

DAB bevat veel meer features die ik niet allemaal kan behandelen in dit artikel. Ik raad je aan om de docu-

In de meeste gevallen zal een web applicatie ook met een server communiceren

mentatie te lezen of contact met mij op te nemen als je meer wilt weten. Tot slot wil ik je graag bedanken voor

het lezen van dit artikel en ik hoop dat je net zo enthousiast bent over Data API Builder als ik! ●

Links

Documentatie: <https://learn.microsoft.com/en-us/azure/data-api-builder/>
Source-code: <https://github.com/Azure/data-api-builder/>
Demo: <https://github.com/sander1095/data-api-builder-demo>
Static Web App-integratie: <https://learn.microsoft.com/en-us/azure/data-api-builder/deployment/how-to-host-static-web-apps>
Auteur: <https://stenbrinke.nl>



De interactieve avatar met ChatGPT

Een innovatie in VR onderwijs

Virtual reality (VR) en kunstmatige intelligentie (AI) veranderen de manier waarop we leren en trainen. Sopra Steria heeft samen met het Expertise Centrum Extended Reality van Landstede, een onderwijsinstelling die onder andere opleidingen tot militair verpleegkundige aanbiedt, een VR-applicatie ontwikkeld om studenten beter voor te bereiden op praktijksituaties. In deze VR-omgeving oefenen studenten medische procedures en communicatieve vaardigheden in een realistisch veldhospitaal.

Auteur: Arco Nederveen

De VR-applicatie bevatte oorspronkelijk volledig gescripte dialogen, waarbij studenten vooraf gedefinieerde gesprekken met virtuele patiënten voerden. Momenteel is er echter een innovatieslag gaande: gescripte dialogen worden vervangen door een interactieve avatar die gebruikmaakt van ChatGPT-technologie. Hierdoor ontstaat een veel dynamischere en realistischere interactie, waarin studenten door middel van hun eigen vragen de symptomen van de patiënt moeten achterhalen.

In dit artikel gaan we dieper in op hoe deze interactieve avatar is ontwikkeld en geïntegreerd in de VR-simulatie.

De avatar als virtuele patiënt

Studenten nemen de rol aan van een verpleegkundige die moet achterhalen welke klachten de patiënt heeft. Deze patiënt, in feite een avatar, is niet zomaar een standaard computergestuurd personage, maar wordt aangedreven door een taalmodel dat natuurlijke gesprekken kan voeren. Het is hetzelfde LLM (Large Language Model) dat wordt gebruikt door ChatGPT (gpt-4o). Het unieke van deze aanpak is dat de studenten niet alleen de juiste medische procedures leren, maar ook communicatieve vaardigheden ontwikkelen terwijl ze in gesprek gaan met de avatar.

De kracht van een taalmodel in de simulatie

In essentie genereert een taalmodel woorden en zinnen gebaseerd op de door de gebruiker ingevoerde tekst, ook wel de 'prompt' genoemd. Dit stelt de avatar in staat om realistisch en dynamisch te reageren op de vragen en opmerkingen van de studenten. Door dit taalmodel te gebruiken, ontstaat een interactieve en levensechte ervaring, waarin de patiënt geloofwaardig reageert op uiteenlopende medische vragen en scenario's.

Een belangrijk gegeven is dat het taalmodel zelf geen geheugen heeft. Elke oproep naar het model staat op zichzelf, wat betekent dat de context van het gesprek steeds moet worden bijgehouden en toegevoegd aan nieuwe prompts. Zo kan de avatar reageren alsof hij weet wat er in het voorgaande gesprek is besproken, terwijl dit in feite door de software wordt geënceneerd.

Een realistische patiënt: prompt voorbeeld

De sleutel tot een geloofwaardige ervaring ligt in de prompts die naar het taalmodel worden gestuurd. De system-prompt instrueert de LLM om een bepaalde rol te spelen en deze wordt achter de schermen altijd meegestuurd. Bijvoorbeeld:

- > Je bent een gewonde soldaat die in een bed in een veldhospitaal ligt. Een verpleger komt naar je bed en wil je mentale en fysieke toestand weten. Je hebt de volgende klachten:
- > Je hebt hoofdpijn
- > Je bent misselijk
- > Je taak is om klachten één voor één aan de verpleegster te melden. Meld slechts één van de klachten als de verpleger duidelijk vraagt naar je mentale of fysieke toestand.

De student (verpleegkundige) moet door middel van vragen ontdekken welke klachten de soldaat heeft. De prompt instrueert het taalmodel om slechts één klacht tegelijk te benoemen wanneer er specifiek naar de mentale of fysieke toestand wordt gevraagd. Deze aanpak zorgt ervoor dat studenten niet alleen medische symptomen leren herkennen, maar ook oefenen met geduldig en gericht vragen stellen.

Bijvoorbeeld:

- > **Verpleger:** "Hallo, hoe gaat het met je?"
- > **Patiënt:** "Goedemorgen! Helaas niet zo goed."
- > **Verpleger:** "Wat lijkt het probleem te zijn?"
- > **Patiënt:** "Ik heb hoofdpijn."

Het taalmodel geeft op een passende en natuurlijke manier antwoord.

Zo worden studenten uitgedaagd om effectief en empathisch te communiceren.

Technische implementatie met Unity3D en Azure Speech SDK

De applicatie is ontwikkeld in Unity3D, een game-engine die vaak wordt gebruikt voor het maken van serious games. Door middel van de Azure Speech SDK worden spraakherkenning en spraaksynthese geïntegreerd in de applicatie. Dit betekent dat de studenten met hun stem vragen kunnen stellen aan de avatar (Speech-to-Text) en dat de avatar vervolgens een gesproken antwoord kan geven (Text-to-Speech). Met de Azure Speech SDK is een vloeiend gesprek mogelijk zonder haperingen. Om dit te bereiken, moet er wel speciale aandacht worden besteed aan de afhandeling van callbacks uit verschillende threads en het stottervrij afspelen van de audio. Om het geheel nog wat realistischer te maken bewegen ook de lippen van de avatar mee met de gesproken tekst.

Integratie in Unity3D

De integratie van het taalmodel in Unity is gedaan met behulp van een Unity-package gemaakt door RageAgainstThePixel2. Deze package maakt het eenvoudig om binnen Unity gebruik te maken van het gpt-4o taalmodel dat wordt gehost in een Azure OpenAI omgeving. Prompt engineering speelt een cruciale rol in het laten begrijpen van de output van het taalmodel. Door instructies mee te geven in de prompt, wordt de output in een gestructureerd formaat gegeven, bijvoorbeeld in JSON. Dit maakt het mogelijk voor de software om de antwoorden van het taalmodel te verwerken.



Desktopversie van de Avatar.

Een voorbeeld van een dergelijke prompt kan er als volgt uitzien:

> Je hebt de volgende klachten: hoofdpijn, misselijkheid. Vertel slechts één klacht per keer als de verpleger er specifiek naar vraagt. Antwoord altijd in het volgende JSON-formaat:

> { 'response': '<antwoord>', 'klacht-Nummer': '<klacht-nummer>' }.

Hierdoor kan de simulatie op intelligente wijze omgaan met de antwoorden van de avatar en zo een doorlopende interactie bieden.

Uitdagingen bij het gebruik van LLM's

Bij het implementeren van Large Language Models (LLM's) in applicaties komen verschillende uitdagingen kijken. LLM's ontwikkelen zich snel, waardoor updates, regelmatige controle en goed beheer nodig zijn om nauwkeurigheid en consistentie te behouden. Ook kunnen LLM's soms onvoorspelbare antwoorden geven, wat het lastig maakt om consistentie en kwaliteit te waarborgen, vooral bij complexe taken. Tot slot zijn ethische overwegingen, zoals het beperken

van vooroordelen en transparantie naar gebruikers toe, belangrijke aspecten bij de ontwikkeling van LLM-gebaseerde applicaties. Het is belangrijk om deze uitdagingen proactief aan te pakken voor sterke en betrouwbare toepassingen.

Conclusie: een toekomstgericht onderwijs met VR en AI

De combinatie van VR en AI biedt ongekende mogelijkheden voor onderwijs en training. Door het gebruik van een interactieve avatar op basis van een taalmodel in de VR-simulatie kunnen studenten medische

en communicatieve vaardigheden onder realistische omstandigheden oefenen. Dit project toont aan hoe technologieën zoals LLM's, spraakherkenning (Speech-to-Text), spraaksynthese (Text-to-Speech) en VR elkaar kunnen versterken om een leerervaring te bieden boeiend, effectief en toekomstgericht is. Met plannen voor verdere uitbreidingen, zoals het toevoegen van meer patiënten en scenario's, zal deze innovatieve benadering van onderwijs in de toekomst nog verder worden verfijnd en toegepast in andere leer-situaties. ●

Referenties:

<https://github.com/Azure-Samples/cognitive-services-speech-sdk>
<https://github.com/RageAgainstThePixel/com.openai.unity>

BIO

Arco Nederveen

Arco Nederveen is XR en AI-ontwikkelaar bij Sopra Steria en betrokken bij de ontwikkeling van innovatieve toepassingen met VR en AI.

Netcompany

Problem solvers

We use the full technology stack to develop smarter, more connected, and sustainable societies.

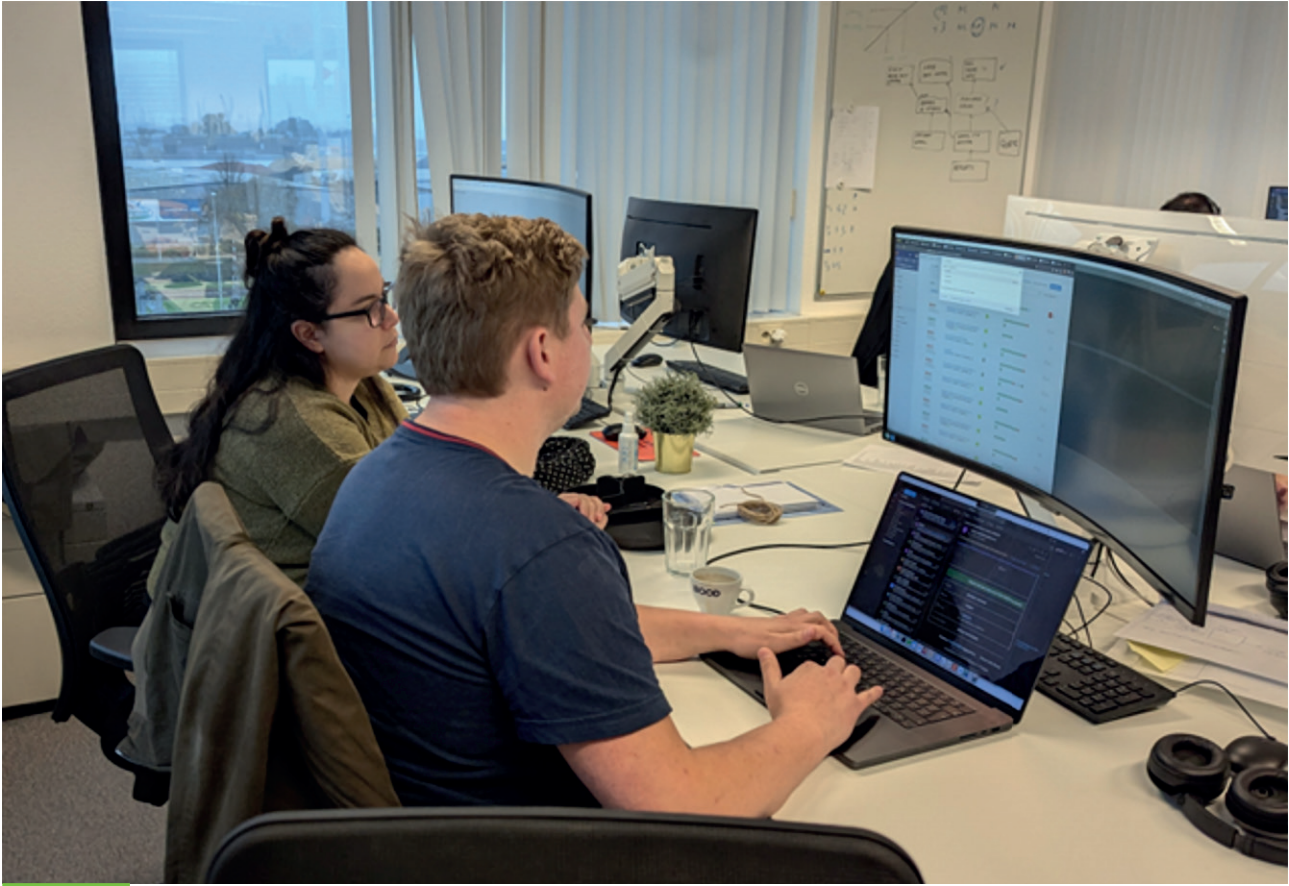
Join our talented people and create IT solutions that are fit for the future.

Apply today.

www.netcompany.com

Explore here.





We often work together closely.

The sense and nonsense of **pull requests**

Recently, I posted my doubts about the effectiveness of pull requests on LinkedIn. Surprisingly, my simple post led to a bonfire of discussions and eventually had over 350,000 views. I struck a nerve.

Auteur: Sander Hoogendoorn

So, what is a pull request? Essentially, a pull request is a proposal to merge changes from one code branch to another. Other people working on the codebase review and

discuss this proposal before integrating the changes into the other branch, usually the main one. This procedure allows collaborators to reason about the quality of the propo-

sed changes and, in doing so, uphold the quality of the entire codebase. Sounds straightforward, right?

Pull requests make perfect sense when you don't trust the changes proposed to your codebase. The most common scenario is open-source projects. In open-source projects, anyone can suggest improvements to the codebase. This is precisely how and why open-source works.

Open-source projects reap the wisdom of the crowd. As one of the

authors of the open-source framework Easy.ts, I see this in practice every day. We don't always personally know the people who propose changes, so we can't always value their experience or personal preferences, which often are even a matter of personal taste.

This is where pull requests come in handy. They allow us to discuss people's coding decisions before merging their code into our projects. It's an excellent opportunity to share skills and create an understanding of different approaches.

Trust

The keyword here is trust. When we don't trust the proposed code changes, we review and discuss them with the proposers. Here, pull requests make sense.

Let's step back from open-source and examine everyday projects and code written in organizations and teams. For starters, my own team at ibood.com. It comprises fifteen people with varying experience, backgrounds, and knowledge of different programming languages. Some have decades of experience, and some have two years.

Ours is a relatively small team. To maintain the quality of our code, we apply a variety of simple techniques:

- > We standardized our software architecture, and we all comprehend its layers and patterns. Consequently, we can open any of our 150+ repositories and understand what we find there, even if we haven't seen a repo before.
- > We often work together closely, for instance, using pair and mob programming, image 1. Even remotely. So, most changes are crafted by multiple people. Hence, there is much less of a need for additional checks afterward.
- > We have weekly knowledge-shar-



Pull requests are valid in environments where trust is low. This is the case with open-source, where we know little about the people who propose changes

ing using the lean coffee approach. During these sessions, anyone can discuss any topic, offering rapid and immediate learning.

- > We write unit tests for all our code, often applying test-driven development.
- > Our CI/CD pipelines are fully automated and provide building, linting, running tests at various stages of the pipelines, static code analysis, and deployments, even to production. Our pipelines provide many quality checkpoints.

Consequently, we don't do pull requests. Each team member pushes their changes directly onto the main branch. We practice trunk-based development; we don't branch. We have built an environment where trust rules, independent of the individual's level of seniority.

The Law of Raspberry Jam

Pull requests are valid in environments where trust is low. This is the case with open-source, where we know little about the people who propose changes. But in teams like my current and many others I've worked with, trust is not an issue. Trust, however, has become an issue in many organizations. Quite some comments on my recent post demonstrate this, often literally: "We

can't trust individual developers to commit to the main branch directly, especially juniors. There is too much at stake."

Lack of trust usually comes with scaling. The more people work on your codebases, the harder it becomes to maintain quality. With a strong ambition to grow teams rapidly and a scarcity of good developers, organizations hire less experienced people and, consequently, struggle with quality.

It is Jerry Weinberg's Law of Raspberry in action: **the further you spread it, the thinner it gets**. And so, pull requests to the rescue.

Timeliness

However, using pull requests and reviewing changes has a severe drawback. The keyword here is **timeliness**.

Most organizations that apply pull requests formally are large or scaling up to become large. The people doing reviews are often the more senior in the development organization. They are also the busiest people — not least because they need to review all pull requests.

Reviewers frequently become bottlenecks.

And so, reviews often don't happen immediately. When you do a

pull request, time goes by before a reviewer picks it up. Sometimes, this is a matter of hours, but if reviewers are in meetings, have a day off, the weekend starts, or one of their kids is sick, the wait becomes longer. Days go by.

Delays have several consequences. When they must wait, developers will pick up another feature. When a review returns several days later with feedback that requires action, the developer must return to the previous feature. If developers have multiple pull requests open—not uncommon—they constantly need to refocus. This is called context switching. It is costly. Multitasking is a myth. Second, as a developer, you want to avoid unnecessary long feedback loops. So, you start to prevent pushing small changes by collecting them until you have finished all the work on a feature. Worse, you only push the changes when you are confident that everything people would have wanted and more is there. You build too much. You over-engineer. The longer you wait, the bigger your change becomes. The bigger the change, the harder it is to merge it back to the main branch, as others will have made changes in the meantime. Despite your good intentions, pull requests continuously increase delays in situations like these.

Missing the bus

So, what's the optimum? Trust is not much of an issue in smaller organizations and teams than in larger ones. Someone replied to my post saying they work with thirty teams on their mobile app and release it once every two weeks. They are strongly biased towards pull requests and review every change someone makes. Their strategy is risk-averse.

At ibood.com, we are not. We immediately push every change to our



As a developer, you want to avoid unnecessary long feedback loops. So, you start to prevent pushing small changes by collecting them until you have finished all the work on a feature

e-commerce platform to production. Yes, severely checked and tested in our pipelines, but still immediately. To us, making a mistake in production is not the end of the world. As Jason Gorman says, "Missing the bus is not a big issue if you know that the next one will be there in five minutes." Ultimately, it all comes down to this: How bad is it if you miss the bus? Large organizations that often work remotely need more trust. The more

risk-averse they become, the more they apply pull requests and reviews. Conversely, when your teams are small and coherent, many techniques help build trust and speed. Pair programming, mob programming, test-driven development, and fully automated delivery pipelines all help avoid dreadful waits and context-switching.

At ibood.com, the next bus comes in five minutes. When does yours?●

THE WORLD NEEDS MORE INNOVATORS

A team of six people created Tikkie. After successfully launching an app like Tikkie, it would be easy to say “Okay, we made it”. But that’s not how innovation works. Innovation is about constantly challenging yourself and spotting opportunities. Daring to acknowledge that something can always be better. Because it’s this mindset that makes the difference. And to make a difference we need more innovators – people like those who developed Tikkie and Groepie. We need you. Employees that will continue to motivate others, and us.

www.workingatabnamro.com



ABN·AMRO

Werken naar een **Security Culture**

Het is een begrip dat steeds vaker langskomt: Security Culture. Voor de meesten van ons die in IT werken als ontwikkelaar, binnen operations of als tester, is er wel enig idee wat hiermee wordt bedoeld. Maar in veel gevallen is dat idee wel beperkt tot het beeld binnen onze eigen rol. Als ontwikkelaars en binnen ops hanteren we bepaalde standaarden binnen implementaties. Als testers weten we dat we een negatief scenario moeten valideren.

Auteur: Milan Meyer

Dit is echter meer een vorm van security awareness dan van een security culture, wat veel breder is en zich ook niet beperkt tot de technische rollen binnen een organisatie. Maar hoe kun je groeien van security awareness naar een organisatie met een echte security culture? Om te begrijpen hoe we deze transitie kunnen maken, is het belangrijk om te definiëren wat we verstaan onder cultuur. Cultuur vertaalt zich naar een groep mensen die dezelfde overtuigingen, waarden en gedragingen delen. Elke organisatie heeft een bepaalde cultuur. Bekende voorbeelden zijn de manier van werken of bepaalde waarden over hoe we met elkaar samenwerken. Een security culture is onderdeel van de cultuur van een organisatie en net zoals een manier van werken, definieert het hoe we omgaan met het aspect security.

Waarom is het nodig?

Cultuur werkt op twee manieren. Als individu worden we beïnvloed door de waarden en het gedrag van de groep, of specifieker, de organisatie. Maar individuen beïnvloeden ook de groep, omdat hun gedrag wordt gezien en gevolgd door anderen. Als mens hebben we de neiging om ons te gedragen zoals er van ons wordt verwacht. Dit is waar een security culture een grote impact kan hebben op het gedrag van medewerkers. Door het onderdeel te maken van de cultuur van een organisatie omvat het iedereen en maakt het security tot ieders verantwoordelijkheid. Aangezien de meeste security-incidenten gerelateerd zijn aan menselijke fouten, is er op dit front veel te winnen.

Hoe te beginnen

Net als bij de meeste organisatorische veranderingen is het altijd

makkelijker als de verandering van bovenaf wordt geïnitieerd en gedragen. Dit betekent dat de verandering wordt aangestuurd door het hogere management en men de nodige tijd, ondersteuning, tools en andere noodzakelijke middelen krijgt om de verandering tot een succes te maken. Dit laat nog steeds genoeg uitdagingen over, maar wat als de noodzaak voornamelijk wordt gevoeld vanuit de ontwikkelaars? Wat als stakeholders begrijpen dat security belangrijk is, maar het belangrijker vinden om nieuwe functionaliteit te implementeren en de time-to-market te verkorten door hier en daar een paar stappen over te slaan?

Uit mijn ervaring als ontwikkelaar is dit helaas niets nieuws en hoewel het veel meer moeite kost om verandering van onderaf te initiëren, benadrukken beide scenario's een fundamenteel aspect van het werken aan een cultuur: gedragsverandering.

Gedragsverandering

Misschien is één van de grootste uitdagingen het veranderen van het gedrag van mensen. Binnen een organisatie heeft iedereen een rol die bepaalde resultaten vereist. Het invullen van een security-aspect binnen die rol (of resultaten) kan als verstorend en uitdagend worden ervaren.

Om iedereen mee te krijgen, kun je overwegen om klein te beginnen. Identificeer met elkaar alle zaken die moeten veranderen en kies één item



dat een kleine verandering vereist, maar een snelle, meetbare impact heeft. Een snelle overwinning stimuleert vaak de betrokkenheid, wat hard nodig is in de volgende stappen. Een andere aanpak kan zijn om eerst de veranderingen aan te pakken die de grootste risico's voor de organisatie vormen. Dit is vaak uitdagender, omdat dit in de meeste situaties buiten de directe invloedssfeer ligt en meer obstakels toevoegt voor gedragsverandering. Welke benadering je ook kiest, iedere stap in de goede richting is waardevol—er is geen goed of fout.

Hoe verder?

Moeten we ons dus alleen richten op ons gedrag als groep en individu? Helaas is het iets complexer. Gedragsverandering zal waarschijnlijk impact hebben op bepaalde

Misschien is één van de grootste uitdagingen het veranderen van het gedrag van mensen

manieren van werken, maar in veel gevallen vereist dit ook investeringen, zoals trainingen en dashboards om de impact van veranderingen te meten en waar nodig bij te sturen. Dit kan zowel op technisch gebied als op procesniveau of binnen een leiderschapsroadmap. Het hangt allemaal af van het perspectief en wat nodig is om de resultaten te meten die relevant zijn voor de veranderingen vanuit die rol. Voor een CISO kan dit betekenen dat

een medewerker enquête wordt uitgevoerd om inzicht te krijgen in hoe veranderingen verschillende rollen beïnvloeden en wat hun uitdagingen zijn op het gebied van security. Voor een ontwikkelaar kan dit betekenen dat statistieken van SCA-tooling worden gebruikt om de resultaten bij de start te vergelijken met de resultaten na twee maanden. Vanuit operations kan het betekenen dat er een ticketsysteem wordt geïmplementeerd voor een ondersteunend

audittraject. Zoals altijd: **it depends**. Het belangrijkste is om resultaten te meten, waar nodig aan te passen en successen te benadrukken. Maar hoe dragen we de gewenste cultuur uit binnen een organisatie?

Opzetten van een Security Champions-programma

In kleinere organisaties wordt men makkelijker door elkaar beïnvloed vanwege kortere communicatielijnen. Vaak draagt één persoon verantwoordelijkheden die in grotere organisaties zijn gesplitst in meerdere lagen en bedrijfsrollen die elk op een andere manier door security worden beïnvloed in het dagelijkse werk. Van het securityteam eisen dat ze al deze verschillende aspecten en uitdagingen volledig begrijpen, is waarschijnlijk onhaalbaar, vooral op de lange termijn. De mensen die hun werk het beste begrijpen zijn natuurlijk degenen die het daadwerkelijk doen, dus hoe krijgen we iedereen in het bedrijf mee? Hoe krijgen we inzicht in de veiligheidsuitdagingen waarmee zij worden geconfronteerd en hoe kunnen we het gedrag van medewerkers in verschillende lagen van een organisatie verbeteren? Dit is waar een Security Champions-programma bij kan helpen.

Wat zijn Security Champions?

Security Champions zijn de bedrijfsbrede security ambassadeurs binnen een organisatie. Vanuit diverse rollen binnen een bedrijf helpen ze om security te integreren in verschillende lagen en functies. Hoewel het idee van Security Champions vaak wordt geassocieerd met softwareontwikkeling (waarbij veel Security Champions meestal een technische achtergrond hebben), is het concept veel breder en kan het ook niet-technische rollen omvatten. Bijvoorbeeld:



- > Projectmanagers of productmanagers: hoewel deze rollen niet direct technisch zijn, kunnen ze een aanzienlijke impact hebben door ervoor te zorgen dat security in elke projectfase of productroadmap wordt opgenomen.
- > HR: een Security Champion in HR kan ervoor zorgen dat onboarding beveiligingsbewustzijnstrainingen omvat en pleiten voor het veilig omgaan met werknemersinformatie.
- > Finance en Legal: Champions in deze gebieden kunnen zich richten op naleving van regelgeving omtrent financiële informatie en gegevensbescherming, terwijl ze collega's ook bewust maken van de risico's van financiële phishingaanvallen.

Een belangrijk voordeel van Security Champions in niet-technische rollen is dat zij een beveiligingsperspectief brengen naar delen van de organisatie die de technische teams misschien niet bereiken.

Vergroten van het Securityteam

Security Champions kunnen helpen om een deel van de druk van het securityteam weg te nemen door te

fungeren als het primaire aanspreekpunt voor hun specifieke team of afdeling. Ze kunnen bepalen wanneer het nodig is om het securityteam in te schakelen. Security Champions dragen aanzienlijk bij aan het beveiligingsbeleid van een organisatie waarbij ze als groep een community vormen over alle functies of afdelingen heen, die helpen bij het identificeren van overlappende problemen en uitdagingen. Deze verantwoordelijkheden brengen echter bepaalde risico's met zich mee. Er is goede training en begeleiding nodig om ervoor te zorgen dat Security Champions uitgerust zijn om hun rol goed te vervullen. Maar hoe definiëren we deze rol en waar begin je?

De reis van duizend mijl

Er zijn veel bronnen over hoe je een Security Champions-programma kunt starten, dus het is niet nodig om het wiel helemaal opnieuw uit te vinden. Bronnen zoals de OWASP Security Champions playbook kunnen helpen bij het definiëren van de eerste stappen. Zoals bij de meeste organisatorische veranderingen begint het met het identificeren van

de verschillende rollen of teams die betrokken zijn. Deze stap is nauw verbonden met het definiëren van de rol en doelen van je securityprogramma.

Afhankelijk van het bedrijf, hoef je misschien in eerste instantie geen Champions buiten technische functies aan te stellen. Je zou kunnen beginnen met een pilotprogramma binnen een specifiek team en zo uitbreiden op basis van de geleerde lessen. Wat de beste benadering voor de situatie ook mag zijn, het identificeren van de teams en het definiëren van de rol is fundamenteel voor de volgende stap: het nomineren van Security Champions.

Security Champions Nomineren

Security Champions worden niet aangesteld – ze worden genomineerd, wat een belangrijk verschil is. Een belangrijk deel van de rol omvat een passie voor het verbeteren van security, iets wat je niet kunt aanleren. Maar hoe selecteer je Champions? Moet je een inschrijfformulier ophangen bij de koffiemachine? Interviews houden met kandidaten die door managers zijn geselecteerd? Of laat je het geheel vrijwillig? Wederom, it depends. Misschien werkt een hybride aanpak het beste voor jouw situatie.

Belangrijk is dat de voordelen van een rol als Security Champion duidelijk zijn. Deze omvatten persoonlijke ontwikkeling, bijdragen aan het beveiligingsprofiel van het bedrijf, het verhogen van je eigen marktpositie en kansen om beveiligingsconferenties bij te wonen. Het voert allemaal terug naar het basisprincipe voor het nomineren van Champions: een goed gedefinieerde rol die ervoor zorgt dat men weet wat er van hen wordt verwacht en hoe ze ondersteund zullen worden. Dit bevat ook de betrokkenheid van leiderschap om



Security Champions zijn de bedrijfsbrede security ambassadeurs binnen een organisatie

ervoor te zorgen dat Champions de benodigde tijd en middelen hebben, het opzetten van communicatiemiddelen en het houden van regelmatige vergaderingen ter ondersteuning van het wij-gevoel.

Het opbouwen van een kennisbank is ook essentieel. Hoewel een kennisbank uitstekend is voor referentiemateriaal zoals checklists, best practices en organisatorische procedures, is het net zo belangrijk om ervoor te zorgen dat je Champions praktijkervaring opdoen. Dit versterkt hun zelfvertrouwen, waardoor ze geleidelijk meer complexe scenario's kunnen overnemen en de rol interessant blijft – een cruciaal aspect van elk Security Champions-programma.

One size does not fit all

Hoewel er een algemeen kader is om te volgen, hangt de specifieke implementatie van een Security Champions-programma af van de situatie van je bedrijf. Hierbij is het vooral belangrijk om mee te nemen welke uitdagingen je wilt aanpakken bij het definiëren van de rollen van Champions in verschillende delen van het bedrijf.

In mijn ervaring als Security Champion heb ik veel meer inzicht gekregen in de bredere veiligheidsaspecten van een organisatie. Dit heeft me geholpen collega's te ondersteunen en mijn rol als sparringpartner binnen organisaties te versterken.

Veel bedrijven worstelen op de een of andere manier met beveiliging. Zal een Security Champions-programma al deze problemen oplossen? Natuurlijk niet, maar het draagt wel bij aan de security culture. Na verloop van tijd kunnen mensen hun eigen initiatieven nemen, waardoor security echt ieders verantwoordelijkheid wordt, zoals het zou moeten zijn. ●



BIO

Milan Meyer

Milan Meyer is lead engineer en architect en werkt als Principal Expert bij Sopra Steria. Vanuit zijn interesse in AppSec en Secure Software Development vervult hij bij Sopra Steria de rol van Security Champion binnen de Microsoft unit.

Bel Pieter.

+31 (0)6 8209 0562
werkenbij.arcady.nl



Let's meet at

Bij Arcady werk je voor klanten als Uitgekookt, Univé en Wehkamp.

Jij brengt hen verder met slimme technologie en innovatieve oplossingen. Stilstand is geen optie, vooruitgang is jouw missie.

Futuretech '25

Zien we jou 13 maart 2025 op onze stand in de Jaarbeurs Utrecht?

 **UITGEKOOKT.NL**

unive

 **wehkamp**

Ontwikkelprogramma's & kennissessies in onze eigen Arcademy.

Veel ruimte voor ondernemerschap en eigen initiatieven.

Relaxte informele werksfeer met knetterfijne collega's.

Jobs die je niet wil missen.

.NET Developer

Development 32-40 uur

Met een herkenbare drive en overtuiging.



Front-end Developer

Development 32-40 uur

Jij weet wat je doet. Van front to end.



Azure Developer

Development 32-40 uur

Jij brengt de cloud tot leven.



Full Stack Developer

Development 32-40 uur

Met een herkenbare drive en overtuiging.



Burgemeester Roelenweg 13K Zwolle



betabit

managed
software
development



De beste
bedrijfs-
kritische
software
in Azure
—

Rotterdam
Amsterdam
Apeldoorn
Eindhoven
Belgrado

Kennis
maken? Wij
drinken graag
een koffietje
met jou!
—

Wij zijn altijd op zoek naar:
.NET Azure ontwikkelaars
Test Automation Engineers
Azure Talent Trainees



Save the date: Future Tech 2025



THE IT CONFERENCE FOR
.NET & MICROSOFT TECHNOLOGIES

Sopra Steria is the main sponsor of Future Tech 2025!

We are a digital solutions provider for tailored, end-to-end corporate technology and software solutions, with over 56,000 employees across 30 countries. With digital tools and market expertise, we help our clients make bold choices and deliver results. We commit to innovation and collaboration and value business development. That's why we're the preferred technology partner of Europe's most successful companies.



Call4paper for Future Tech 2025 is now open!

After a completely sold-out edition in 2024, here's your chance to take the stage at Future Tech 2025.

On March 13, 2025, we'll explore the latest innovations in .NET together. At SDN, we're passionate about providing opportunities not just for seasoned speakers but also for those taking their first steps in public speaking. Think you're ready to share your expertise? Submit your session proposal today, and who knows, you could be shining on stage this March!

After a completely sold-out edition in 2024, here's your chance to take the stage at Future Tech 2025.

On March 13, 2025, we'll explore the latest innovations in .NET together. At SDN, we're passionate about providing opportunities not just for seasoned speakers but also for those taking their first steps in public speaking. Think you're ready to share your expertise? Submit your session proposal today, and who knows, you could be shining on stage this March!




@futuretech_19



Future Tech



SDN (group)



@FutureTechNL



@futuretechnl.bsky.social

.NET Developer Zorg-ICT

Salarisindicatie:
45.000 - 70.000 EUR

Locatie:
Amsterdam

Technologieën:
.NET Framework, C#, Cloud, .NET



Azure DevOps Engineer

Salarisindicatie:
60.000 - 85.000 EUR

Locatie:
Nieuwegein

Technologieën:
Azure, Power BI, PowerShell, C#,
DevOps, CI/CD, .NET



Azure Developer (Cloud Engineer)

Salarisindicatie:
40.000 - 65.000 EUR

Locatie:
Zwolle

Technologieën:
Cloud, .NET Framework, Azure, IT



Cloud Developer

Salarisindicatie:
55.000 - 65.000 EUR

Locatie:
Apeldoorn

Technologieën:
AWS, Azure, C#, Cloud, DevOps, Docker, ICT, Java,
JavaScript, Kubernetes, Python, microservices



Find all these vacancies and more

at [Devitjobs.nl](https://devitjobs.nl)

sopra  steria

*Happy
Holidays*

