

#### Hilmar van der Veen

Zandvoort 25-05-1987



+31 6 8014 9947



hilmar.van.der.veen@the-future-group.com



Senior Frontend Engineer – React, Redux, TypeScript, Data Enablement



#### **SAMENVATTING**



Senior frontend developer met ruim zeven jaar ervaring in het bouwen van schaalbare, onderhoudsvriendelijke webapplicaties op basis van React, Next.js en TypeScript. In opdrachten bij o.a. de Belastingdienst, Nationale Postcode Loterij, FedEx en Ziggo werkte ik aan portals, low-code editors, design systems en headless CMS-integraties. Ik lever werkende, toegankelijke (WCAG), goed geteste interfaces op die daadwerkelijk gebruikt worden in omgevingen waar kwaliteit, schaalbaarheid en gebruikerswaarde centraal staan.

Ik combineer technische realisatie met overzicht en samenwerking. Naast het opzetten van componentarchitecturen, CI/CD-processen (Azure DevOps) en teststrategieën (Jest, Cypress, Playwright), draag ik actief bij aan teamontwikkeling. Via peer reviews, pair programming en kennisdeling help ik collega's groeien en werk ik aan duurzame verbeteringen in de codebase en het proces.

Tijdens opdrachten ben ik gefocust, volledig beschikbaar en gericht op oplevering. Ik functioneer het best in multidisciplinaire teams die bouwen aan een concreet eindproduct. Daar zorg ik voor tempo, kwaliteit en rust, zodat het team gezamenlijk vooruitkomt en het product daadwerkelijk afkomt.

	Medior	Senior	Lead		Medior	Senior	Lead
Angular 2 - 18	•	•	•	Webpack	•	•	0
React 16 +	•	•	•	FP-patterns (pure functions, immutability, RxJS)	•	•	0
Vue 3+	•	0	0	ECMAScript (latest)	•	•	0
Typescript	•	•	•	GitHub Actions	•	•	0
JavaScript (ES5/ES6/ESNext)	•	•	•	NX (Mono repo)	•	•	0
HTML 5	•	•	0	REST / OpenAPI / SOAP	•	•	0
CSS3, LESS, SASS	•	•	0	OATH 2	•	•	0
Tailwind CSS	•	•	0	OpenID Connect	•	•	0
Styled-components	•	•	0	ASP.NET Web API	•	•	•
Bootstrap / Material UI	•	•	0	GraphQL Server   GraphQL CodeGen	•	•	0
Stencil / Web Components	•	•	0	FIGMA   Sketch   Wireframe	•	•	0
RxJS	•	•	•	WCAG 2.0 (Toegankelijkheid)	•	•	0
Redux / Thunk	•	•	•	Design Systems	•	•	•
Zustand	•	•	•	Component-based design	•	•	•
React Query	•	•	•	Storybook	•	•	•
NextJS   Vite	•	•	•	Cypress	•	•	0
BEM (CSS architectuur)	•	•	0	Playwright	•	•	0
SVG-animaties	•	•	0	Jest	•	•	0
Node.js	•	•	•	Karma	•	•	0
Docusaurus	•	•	0	xUnit   jUnit	•	•	0
C# / .NET Core (3 5 6)	•	•	•	SonarQube	•	•	0
Azure DevOps	•	•	0	ESLint	•	•	0
Azure Pipelines (CI / CD)	•	•	0	Prettier	•	•	0
Jenkins	•	0	0	StyleCop	•	•	0
Tekton	•	0	0	SQL Server (MSSQL)	•	•	0
ArgoCD	•	0	0	LINQ	•	•	0
Git   GitHub   GitLabs   BitBucket	•	•	0	Neo4J (GraphDB)	•	0	0
Docker	•	0	0	Azure Cosmos DB	•	•	0
Kubernetes	•	0	0	JSON	•	•	0
MAVEN	•	0	0	Azure (App Services, Cloud, DevOps, Pipelines)	•	•	0
JIRA	•	•	•	AWS (Lambda   S3 Bucket)	•	0	0
Confluence	•	•	•	Microsoft Identity	•	•	0

LDAP	•	•	0	NPM	•	0	0
JWT	•	•	0	JAVA 8	•	0	0
SPLUNK	•	0	0	Headless CMS   Hippo Bloomreach	•	•	0
Lighthouse / aXe	•	•	0	Entityframework	•	•	0

OPLEIDINGEN			<b>©</b>
Master Physics (AMEP)	UVA		2012 -2014
BSc Physics and Astronomy	UVA	$\boxtimes$	2012 afgerond

TRAININGEN		EQ.
React   Angular   Typescript   python   C#   SQL	Udemy	2020
Scrum master   C#   CSS   HTML	Young Capital	2016
Talen:		
Nederlands (Vloeiend)		
Engels (Vloeiend)		

# **RELEVANTE WERKERVARING**



# **Lead Frontend Developer/ Architect**

Januari 2025 – heden

#### hilmarvanderveen.com - Persoonlijke portfolio in Next.js SSR (App Router)

Mijn persoonlijke website hilmarvanderveen.com is opgezet als schaalbaar portfolio-platform waarin ik mijn werkervaring, projecten en vaardigheden toon binnen een moderne, performante architectuur.

Ik ontwikkelde de site met Next.js en de nieuwe App Router-structuur, met server-side rendering en dynamische segmenten voor schaalbaarheid en flexibiliteit. De site wordt gehost op Vercel en maakt gebruik van een op maat gemaakte Google Analytics Data Layer, waarmee gebruikersinteracties en route-navigatie inzichtelijk zijn gemaakt.

De toepassing ondersteunt meertaligheid via next-intl, en styling is opgezet met TailwindCSS. Voor state management gebruik ik Redux Toolkit. De codebase is gestructureerd in Git, geformatteerd met Prettier en automatisch getest met Jest en Playwright. De gehele stack is ingericht met oog op schaalbaarheid, onderhoudsgemak en performance, wat deze site tot een robuust en professioneel platform maakt voor zowel bezoekers als voor mezelf als ontwikkelaar.

# Paysafe-Casino.de – SEO-gedreven affiliate platform in Angular Universal

Voor mijn eigen affiliate platform Paysafe-Casino.de ontwikkelde ik een SEO-geoptimaliseerde contentwebsite gericht op de Duitse iGaming-markt. De uitdaging was om een schaalbare, meertalige omgeving op te zetten die zowel snelle laadtijden bood als een hoge indexeerbaarheid voor zoekmachines, met volledige controle over content, tracking en performance.

Ik bouwde het platform met Angular Universal (SSR) en hostte het op Azure App Services, met domeinbeheer via Namecheap. Voor performance en contentstructuur implementeerde ik structured data, server-side rendering en dynamische meta-informatie. Bezoekersgedrag wordt inzichtelijk gemaakt via een maatwerk Google Analytics Data Layer, met meetpunten voor page views, outbound clicks en CTA-interacties.

De site ondersteunt meertaligheid (Duits/Engels) via een custom i18n-structuur. Styling is gerealiseerd met TailwindCSS en aanvullende CSS. RxJS wordt ingezet voor reactieve user experience. Voor codekwaliteit gebruik ik Git, Prettier, ESLint en geautomatiseerde tests met Playwright en Jest. SEO-audits, technische performance en toegankelijkheid worden periodiek gevalideerd met Lighthouse. Het platform voldoet aan moderne webstandaarden, is eenvoudig uitbreidbaar en levert conversiegericht affiliateverkeer op.

Total Control of the	matere repr
Technieken & Tools	Werkwijze
Angular 19 Universal SSR, TypeScript, NextJs SSR, RxJS, TailwindCSS, CSS, JavaScript (ES6+), OpenAPI 3.0, REST, WCAG 2, HTML5, Playwright, Jest, Azure, Meertaligheid, Github, Google	Figma design, Zelf bouw, Jira
Analytics, Google Tag Manager	

Senior Frontend Developer / Technisch Lead		
Januari 2023 - December 2024		Belastingdienst
Klant: Belastingdienst	**************************************	G Colonia Colo
Werklocatie: thuis en 2 dagen op locatie in Apeldoorn		
Voertaal: Nederlands		

#### Project 1: BOLD Web Components - Design System Governance & Frameworkintegratie

De Belastingdienst werkte met het centrale BOLD design system (Stencil Web Components), maar teams ondervonden structurele problemen bij het gebruik ervan in Angular-, React- en Vue-omgevingen. De documentatie was ontoereikend en leidde tot inconsistent gedrag.

Ik nam het voortouw om deze problemen structureel op te lossen. Ik testte alle componenten per framework, schreef heldere documentatie in Confluence met platform-specifieke voorbeelden en edge cases, en ontwikkelde wrappers en polyfills voor compatibiliteit. Voor ontbrekende functionaliteit leverde ik tijdelijke bridge-components, inclusief gecontroleerde uitfasering.

Door mijn inzet konden front-end teams sneller en foutloos integreren met BOLD. De adoptie verbeterde, de toegankelijkheid (WCAG) werd geborgd en componentgedrag werd consistent over teams en frameworks heen.

#### Project 2: CANVAS - Visuele Formuliereneditor (Greenfield Low-code Project)

Redacteuren bij de Belastingdienst moesten zonder programmeerkennis complexe formulierflows kunnen opbouwen. Er was behoefte aan een intuïtieve low-code editor met drag-and-drop-functionaliteit, conditionele logica en versiebeheer.

Ik ontwierp en bouwde de frontend van de editor op basis van een componentgebaseerde architectuur. Gebruikers konden formulieronderdelen samenstellen via een canvas, instellen via configuratiepanelen en testen in een tryout-weergave. Validaties, afhankelijkheden, meertaligheid en toegankelijkheid waren volledig via de UI instelbaar.

Daarnaast breidde ik backend-API's uit (Java, MySQL) zodat wijzigingen correct werden opgeslagen en reproduceerbaar waren. Dit resulteerde in een krachtig redactietool waarmee redacteuren zelfstandig toegankelijke formulieren konden publiceren, zonder afhankelijkheid van ontwikkelaars.

# Project 3: Ondernemersportaal – Data-driven Frontend Architectuur

Voor het Ondernemersportaal ontwierp ik een datagedreven frontendarchitectuur die het mogelijk maakte om pagina's sneller, consistenter en zonder code duplicatie op te bouwen. De wens was om meerdere aangiftepagina's, zoals btw, loonheffing, dividendbelasting en vliegbelasting, modulair te genereren op basis van backendconfiguratie, met herbruikbare componenten en flexibele gedragsaanpassing.

Ik ontwikkelde een set generieke Angular-componenten waarvan de logica en weergave werden aangestuurd via injectable services en configuratiebestanden. Invoervelden, infoblokken, validaties en hulpteksten pasten hun gedrag dynamisch aan op basis van waarden uit andere componenten of data uit de backend. De architectuur maakte gebruik van conditionele zichtbaarheid, meertalige labels en validatieregels die via JSON-schema's werden ingeladen.

Daarnaast implementeerde ik frontendfilters die bestonden uit dropdowns, zoekvelden en chips, inclusief verwijderbare chips voor actieve filters. Voor tabellen met beperkte datasets schreef ik een wrappercomponent die client-side sortering mogelijk maakte, zonder dat de backend opnieuw aangesproken hoefde te worden. Deze wrapper hield rekening met kolomtypes, toegankelijkheidseisen en sorteerlogica volgens de specificaties van het BOLD design system. De zoekfunctionaliteit en filteropbouw waren vergelijkbaar met patronen uit Azure Al Search en Algolia: gebruikers konden resultaten interactief verfijnen zonder serverrondreis, met behoud van context en bruikbaarheid.

Door deze aanpak konden nieuwe pagina's met minimale effort worden uitgerold, terwijl toegankelijkheid (WCAG), herbruikbaarheid en consistentie geborgd bleven. De architectuur bleek schaalbaar, onderhoudsvriendelijk en zeer geschikt voor hergebruik in soortgelijke portalen binnen de organisatie.

Technieken & Tools	Werkwijze
HTML5, CSS3, JavaScript (ES6+), TypeScript, Angular, React, Web Components,	Scrum, Kanban, Nauwe samenwerking met UX, PO,
CSS, Stencil, WCAG 2.0 (toegankelijkheid), Playwright, Cypress, REST, OpenAPI 3.0,	backend-ontwikkelaars en DevOps. Actieve rol in
JWT, Java 8, Maven, Gitlabs, Git, NPM, Jenkins, Tekton, Argo CD, SPLUNK, Docker,	refinement, teamcoaching, code reviews en CI/CD-
SonarQube, ESLint, Storybook, Confluence, Jira, MySQL, OWASP, OATH 2,	verbeteringen.
headless CMS, Meertaligheid	

# Frontend Developer / Design System Specialist Juli 2022 - Januari 2023 Klant: Nationale Postcode Loterij Werklocatie: thuis en 2 dagen op locatie in Amsterdam Voertaal: Engels

Project: Internationaal Design System voor Marketingcampagnes

In een greenfield-opdracht ontwierp en ontwikkelde ik een schaalbare front-end infrastructuur waarop internationale marketingteams zelfstandig campagnes konden bouwen. In React, TypeScript en Next.js realiseerde ik een configureerbare componentenbibliotheek, waarbij gedrag, layout en styling via JSON-configuratie instelbaar waren. Dankzij de inzet van een NX-monorepo konden meerdere labels — zoals Geniet van het Goede en VIP-kaarten — hetzelfde design system gebruiken, terwijl branding en UX-variaties eenvoudig via thematisering toepasbaar bleven.

Naast de technische implementatie richtte ik een centrale documentatie- en testomgeving in met Storybook, zodat componentgedrag en gebruik voor zowel ontwikkelaars als marketeers inzichtelijk en overdraagbaar werd. De architectuur ondersteunde configuratiegedreven opbouw, hergebruik over applicaties heen, en duidelijke scheiding tussen kernlogica en visuele identiteit. Deze aanpak is inhoudelijk vergelijkbaar met het doel van een Global Data Layer: één gedeeld framework dat schaalbaar, begrijpelijk en breed inzetbaar is binnen uiteenlopende teams en use cases.

Ik integreerde het platform in CI/CD-pijplijnen (GitLab CI), werkte nauw samen met DevOps, UX en marketing, en borgde dat releases snel, betrouwbaar en schaalbaar waren via cloudcomponenten zoals AWS Lambda en S3. Door server-side rendering (SSR) en performanceoptimalisaties werd het platform SEO-proof en toegankelijk conform WCAG 2.0. Het resultaat: een goed gedocumenteerd en breed geadopteerd platform dat de uitrol van campagnes aanzienlijk versnelde, zonder in te boeten op kwaliteit of flexibiliteit.

Technieken & Tools	Werkwijze
React, TypeScript, Next.js, TailwindCSS, SCSS, SVG Animations, Headless CMS, GitLab CI,	Scrum (Agile), Intensieve samenwerking met
AWS Lambda, REST API's (OpenAPI 3.0), SSR, HTML5, JavaScript (ES6+), NPM, Git, Docker,	UX-designers, DevOps en marketingteams.
Cypress, Playwright, ESLint, Prettier, Jira, Confluence, S3-bucket, SPLUNK, SonarQube,	
Zustand, React-query, OWASP, JWT, OATH 2, Storybook, GraphQL Server, GraphQL	
CodeGen, headless CMS, Meertaligheid	

Lead Frontend Developer		
Oktober 2020 - Juli 2022	THLON	
Klant: Athlon	/AI HLON	
Werklocatie: thuis		
Voertaal: Engels		

Project: Migratie & Herbouw van Legacy-applicatie naar Moderne Angular Stack

De bestaande Angular 1.6-applicatie voldeed niet meer aan de hedendaagse eisen op het gebied van performance, onderhoudbaarheid en toegankelijkheid. Mijn opdracht was om zowel de technische architectuur als de gebruikersinterface volledig te vernieuwen, met behoud van bestaande functionaliteit, maar binnen een moderne Angular-stack.

Ik leidde de migratie naar Angular 12 en ontwierp een nieuwe frontendarchitectuur gebaseerd op RxJS en reactive programming. Hierbij stond schaalbaarheid, WCAG-compliance en performance-optimalisatie centraal. Ik herstructureerde de UI op basis van moderne stylingprincipes en begeleidde junior- en medior-developers bij het schrijven van kwalitatieve, goed geteste frontendcode. Daarnaast integreerde ik de frontend met meerdere backend REST API's via OpenAPI 3.0.

Ik ontwikkelde een flexibele componentbibliotheek en bouwde een generiek selfservice-platform met conditionele logica en validatie, gericht op low-code principes. Voor dashboards realiseerde ik dynamische datavisualisaties, met responsive grafieken en sorteerbare tabellen. In samenwerking met UX-designers vertaalde ik wireframes en gebruikersfeedback naar een consistente, toegankelijke frontend.

Het eindresultaat was een toekomstbestendige applicatie, gebouwd op moderne standaarden, volledig WCAG-conform en eenvoudig uitbreidbaar. De herbouwde applicatie maakte snellere interactie mogelijk, verlaagde de foutkans en resulteerde in minder druk op support dankzij duidelijke, goed ontworpen selfservice-functionaliteit.

Technieken & Tools	Werkwijze
Angular 12, TypeScript, RxJS, TailwindCSS, SCSS, JavaScript (ES6+), OpenAPI 3.0, REST, WCAG 2.0, HTML5, Cypress, Jest, Azure, Jenkins, Docker, GitHub, Git, SonarQube, Java 8, ESLint, Prettier, Jira, Confluence, KeyCloak, Bootrstap UI, OWASP, JWT, OATH 2, Storybook, Meertaligheid	Scrum (Agile), Leidend in technische beslissingen, coaching en refinement. Nauwe samenwerking met UX en backend.

Frontend Developer / UI Coach			
Augustus 2020 - Oktober 2020	ר randstad		
Klant: Randstad		ii rarrastaa	
Werklocatie: Amsterdam			
Voertaal: Engels			

Project: Rebranding en Componentontwikkeling voor Randstad- en Yacht-websites

Randstad en Yacht ondergingen een visuele en technische herpositionering waarbij bestaande websites vernieuwd en uitgebreid moesten worden met toegankelijke, consistente UI-componenten conform de nieuwe merkidentiteit.

Ik werkte aan de herontwikkeling van front-endcomponenten in vanilla JavaScript, Angular en React. Hiervoor gebruikte ik SCSS en de BEM-methodiek om structuur en herbruikbaarheid te waarborgen. Ik ontwikkelde modulaire componenten met focus op toegankelijkheid (WCAG 2.0) en begeleidde collega's in het toepassen van Angular best practices en componentgedreven ontwikkelprincipes. Mijn rol als UI Coach hielp teams consistentie en kwaliteit verhogen, met behoud van flexibiliteit tussen platforms.

Daarnaast ontwikkelde ik stylingcomponenten voor het Hippo Bloomreach headless CMS en testte ik alle visuele en functionele elementen op toegankelijkheid en responsive gedrag. Ik werkte nauw samen met UX-designers en business stakeholders om pagina's en flows op elkaar af te stemmen en ervoor te zorgen dat de nieuwe componenten naadloos aansloten op de vernieuwde merkidentiteit.

Het resultaat was een uniforme, schaalbare UI-bibliotheek die de visuele kwaliteit, toegankelijkheid en performance van de Randstad- en Yacht-platformen merkbaar verbeterde. Door mijn coaching en technische begeleiding nam de ontwikkelsnelheid toe en werd de kwaliteit van releases stabieler.

Technieken & Tools	Werkwijze
React, Angular, SCSS, JavaScript (vanilla), BEM, WCAG 2.0, component-based	Scrum (Agile), Coachen van teams, documenteren
design, Hippo Bloomreach (headless CMS), FTL, Maven, GitLab, Jenkins, Git,	van componentrichtlijnen en kwaliteitsborging in
SonarQube, ESLint, Prettier, Bitbucket, Confluence, Github, JAVA 12, Spring Boot	CI/CD-omgeving.

Fullstack Developer / Frontend-Lead (Angular & .NET Core)		
Augustus 2019 - Maart 2020	Oneninlan	
Klant: Omniplan	Omniplan Delivering Personal Finance	
Werklocatie: Amsterdam		
Voertaal: Nederlands		

# Project: Modulaire Authenticatieomgeving & Low-code Frontend

Omniplan vernieuwde haar platformarchitectuur om veilige integratie met externe partijen mogelijk te maken. Zowel de frontend als backend werd volledig herbouwd. Ik was verantwoordelijk voor de implementatie van een nieuwe Angular 7 frontend in combinatie met een herontworpen .NET Core backend, inclusief architectuur en een flexibel autorisatie- en configuratiesysteem.

Ik ontwikkelde een modulaire frontend met HTML, CSS, TypeScript en RxJS, gericht op herbruikbaarheid, toegankelijkheid en performance. Voor de backend ontwierp ik een op Microsoft Identity en JWT gebaseerde autorisatiestructuur, waarmee externe organisaties veilig toegang konden krijgen tot REST API's van Omniplan. Deze hybride architectuur maakte veilige authenticatie mogelijk tussen partijen zonder directe systeemkoppeling.

Daarnaast ontwierp ik een nieuwe microservicesarchitectuur op basis van Azure App Services en Docker/Kubernetes, en herstructureerde ik het datamodel in Entity Framework op MSSQL voor betere schaalbaarheid. Aan de frontendzijde realiseerde ik een dynamisch configureerbare interface, waarin validaties, permissies en invoervelden direct gestuurd werden via backendconfiguraties – vergelijkbaar met een low-code omgeving.

Het resultaat was een schaalbaar, veilig en onderhoudsvriendelijk platform dat eenvoudig kon worden aangepast aan nieuwe functionele wensen, met minimale ontwikkelinspanning en maximale flexibiliteit in externe integraties.

Technieken & Tools	Werkwijze	
Angular 7, TypeScript, RxJS, HTML5, CSS3, C#, ASP.NET Core, Microsoft Identity,	Scrum (Agile), Verantwoordelijk voor	
JWT, REST API, Entity Framework, MSSQL, configuration-based UI, Docker,	technische architectuur,	
Kubernetes, Azure App Services, Azure Cloud Services, DevOps, TFS, Git,	integratieaanpak en fullstack	
Confluence, xUnit, SonarQube, ESLint, Prettier, OWASP, OATH 2	implementatie.	

Full stack software developer (Angular & .NET Core )		
April 2019 – Juli 2019	ORTEC	
Klant: Ortec	OICILO	
Werklocatie: Zoetermeer		
Voertaal: Engels		

# Project: Logistiek Portaal voor FedEx – Route-inzicht en Optimalisatie

Voor FedEx ontwikkelde ik een portaal op basis van Angular microfrontends, waarbij afzonderlijke tools als zelfstandige webpagina's benaderd werden vanuit een centrale hoofdinterface. Ik was verantwoordelijk voor het bouwen van frontendpagina's waarop chauffeurs onder andere konden controleren of pakketten correct waren geregistreerd, de berekende laadvolgorde konden inzien, en realtime hun route konden volgen via een interactieve kaart. Deze pagina's waren opgebouwd uit herbruikbare en schaalbare componenten, ontworpen binnen een datagedreven architectuur, waarbij gedrag en presentatie automatisch reageerden op pakketdata zoals barcodes, gewichten en adressen.

De componenten maakten gebruik van RxJS-observables voor realtime updates, foutmeldingen en validatiefeedback. Ik implementeerde interactieve UI-elementen zoals modals en kaartvisualisaties, die direct inspeelden op events vanuit de backend. De frontend communiceerde via OpenAPI-specifieke REST-endpoints met .NET Core-microservices die verantwoordelijk waren voor berekeningen en datalevering. Door deze modulaire opzet konden tools los van elkaar worden ontwikkeld en ingezet, met behoud van een consistente gebruikerservaring binnen het hoofdportaal.

Het platform werd CI/CD-gedreven uitgerold via een DevOps-pijplijn op Azure Cloud, waarbij ik gebruik maakte van Docker voor de containerisatie. Binnen een multidisciplinair Scrum-team werkte ik samen met DevOps- en data-engineers en borgde ik dat de frontends performant, toegankelijk (WCAG) en goed testbaar waren. Tools zoals xUnit, StyleCop en SonarQube ondersteunden de kwaliteitscontrole. Het eindresultaat was een robuust en fouttolerant portaal dat chauffeurs realtime inzicht en ondersteuning bood, en de efficiëntie van het laad- en bezorgproces aantoonbaar verbeterde.

Technieken & Tools	Werkwijze
Angular 7, Angular Material, TypeScript, RxJS, SCSS, HTML5, REST API, OpenAPI, .NET Core, C#, Azure Cloud, Azure Cosmos DB, Docker, Microservices, data-driven UI, xUnit, StyleCop, SonarQube, ESLint, Prettier, Git, Gitlab, Jira, Confluence, OWASP, JWT, OATH 2	Werken in DevOps-team met CI/CD, focus op foutbestendigheid, schaalbaarheid en UX-consistentie.

Fullstack Developer (ReactJS, .NET Core, Neo4J)	O Divo Cold
Maart 2018 – April 2019	Bluefield     smart access
Bluefield smart access	Siliul I decess
Werklocatie: Maarsen Utrecht	
Voertaal: Nederlands	

#### Project 1: Data-driven Webapplicatie met Cyclusvisualisatie

Binnen Bluefield werkte ik aan een webapplicatie die complexe cyclische processen inzichtelijk moest maken voor eindgebruikers. Deze processen werden aangeleverd via CSV-bestanden en opgeslagen in een Neo4J graph-database. De klant wilde de status van deze cycli visueel kunnen volgen, inclusief foutdetectie en herstelopties.

Mijn taak was het ontwerpen en ontwikkelen van een visueel dashboard waarin cycli overzichtelijk weergegeven werden, met interactieve elementen en een heldere statusweergave per cyclus.

Ik bouwde de frontend in ReactJS, met gebruik van modern JavaScript (ES6+), HTML en CSS. De applicatie was gekoppeld aan een .NET Web API die de data uit Neo4J ontsloot. Ik ontwierp een tegelgebaseerde gebruikersinterface waarin elke cyclus een kleurgecodeerde status kreeg, zoals "succesvol", "mislukt" of "onbekend". Via een interactieve detailweergave konden gebruikers doorklikken naar uitgebreide informatie per cyclus, waaronder data en aantallen. Bij foutieve verwerking bood de interface de mogelijkheid om de betreffende cyclus opnieuw op te starten. Alle logica werd ondersteund door een schaalbare Redux/Redux-Thunk architectuur, gericht op herbruikbaarheid en onderhoudsgemak.

De applicatie gaf gebruikers real-time inzicht in duizenden processen en maakte snelle bijsturing mogelijk bij fouten. Dit zorgde voor meer grip op dataverwerking, minder handmatige tussenkomst en een betrouwbaardere rapportagestroom.

#### Project 2: Tommy Hilfiger - Toegangscontrole & Parkeerbeheer

Tommy Hilfiger had behoefte aan een intern systeem om parkeerplaatsen te beheren en bezoekers gecontroleerd toegang te verlenen tot het terrein. Het systeem moest gebruikers in staat stellen reserveringen te maken en toegangsbewijzen te printen via een eenvoudige interface.

Mijn opdracht was om een gebruiksvriendelijke webapplicatie te ontwikkelen waarin medewerkers parkeerplaatsen konden reserveren en bijbehorende tickets konden genereren en afdrukken.

Ik ontwikkelde een dynamische kalenderinterface in React, waarmee gebruikers per dag parkeerplaatsen konden inplannen. De applicatie was ontworpen met het oog op gebruiksgemak en minimale leercurve. Ik zorgde voor een consistente visuele stijl en bouwde herbruikbare UI-componenten die eenvoudig geïntegreerd konden worden in andere delen van het platform.

Het systeem werd met succes in gebruik genomen binnen de organisatie. Medewerkers konden snel en zelfstandig parkeerplaatsen reserveren en tickets printen, wat de administratieve last verminderde en de toegangsstroom aanzienlijk verbeterde.

# Project 3: Ziggo - Netwerkbeheer & Storingsvisualisatie

Voor het technisch team van Ziggo ontwikkelde ik ondersteunende tools waarmee netwerkstoringen sneller gelokaliseerd en opgelost konden worden. Het doel was om monteurs en beheerders real-time inzicht te geven in de status van het netwerk.

Ik bouwde een WPF Windows-applicatie waarmee beheerders netwerkcomponenten zoals fibernodes, CMTS'en en taps konden beheren. Daarnaast realiseerde ik een ReactJS-webapplicatie met een interactieve kaartweergave, waarin storingen direct zichtbaar werden gemaakt. Foutieve nodes werden visueel gemarkeerd op de kaart, zodat gebruikers snel konden ingrijpen. Deze frontendcommuniceerde met een Angular 6-laag en werd aangestuurd via microservices in C#, waarbij data werd verwerkt via RESTful APIs en berichten werden afgehandeld via RabbitMQ en MassTransit. De combinatie van visuele feedback, messaging en integratie zorgde voor een robuust platform met snelle doorlooptijden.

De ontwikkelde tools verbeterden de efficiëntie van het storingsbeheer aanzienlijk. Monteurs konden sneller reageren op problemen, wat leidde tot kortere hersteltijden en hogere klanttevredenheid. De technische opzet was schaalbaar en goed onderhoudbaar dankzij de toepassing van een component-based architectuur en DevOps deployment via Docker.

Technieken & Tools	Werkwijze
ReactJS, Redux, Redux-Thunk, Angular 6, TypeScript, HTML5, CSS3, JavaScript (ES6+), RxJS, .NET Core, C#, Neo4J, SQL Server, RESTful Web API, RabbitMQ, MassTransit,	Scrum (Agile), Fullstack ontwikkeling met CI/CD, component-based architectuur en DevOps
Docker, Azure Cloud, Azure Cosmos DB, xUnit, StyleCop, SonarQube, Prettier, ESLint,	deployment pipelines.
GitHub, Git, Jira, Confluence, Bootstrap CSS	

Frontend Developer (ReactJS & WPF)	
December 2017 – Maart 2018	_ PERSOONLIJKE
Niped	<b>⊘</b> Gezondheidscheck
Werklocatie: Hoorn	
Voertaal: Engels	

# Project: Website en Desktopinterface voor 'De Persoonlijke Gezondheidscheck'

Voor Niped ontwikkelde ik zowel een webapplicatie als een desktoptool ter ondersteuning van hun gezondheidsplatform, waarmee gebruikers inzicht kregen in vitaliteit en gezondheidsrisico's. De bestaande frontend vroeg om uitbreiding, optimalisatie en meer stabiliteit. Tegelijkertijd was er behoefte aan een beheerinterface voor interne contentredactie.

Ik werkte aan twee sporen: het verbeteren van de bestaande ReactJS-applicatie en het ontwikkelen van een nieuwe WPF-desktopinterface in C#. In de React-applicatie breidde ik functionele componenten uit en verbeterde ik de performance en foutafhandeling. Ik implementeerde toegankelijkheidsverbeteringen (HTML5/CSS) en zorgde voor consistentie in styling, interactie en herbruikbaarheid op basis van ESG+ design patterns. De frontend-architectuur werd opgebouwd rond Redux en React Router.

Parallel ontwierp en ontwikkelde ik een desktopinterface in WPF (MVVM-patroon), waarin medewerkers eenvoudig content zoals teksten en afbeeldingen konden beheren. Deze interface was gekoppeld aan RESTful API's en ondersteunde beeldredactie, planning en contentbeheer. Ik bewaakte de codekwaliteit met SonarQube en werkte nauw samen met backend developers. Binnen het Scrumteam droeg ik actief bij aan refinement, demo's en planningen.

De combinatie van web- en desktopverbeteringen resulteerde in een snellere, stabielere en visueel consistentere gebruikerservaring. De nieuwe desktoptool maakte interne workflows efficiënter en verbeterde de kwaliteit en actualiteit van gepubliceerde content.

Technieken & Tools	Werkwijze
ReactJS, Redux, React Router, HTML5, CSS3, JavaScript (ES6+), RESTful WebAPI, WPF (C#), MVVM, SonarQube, ESLint, Prettier, Scrum, Jira, Bitbucket, SourceTree, Bootstrap CSS, Confluence, FHIR – Bekend met medische datamodellen (Niped), snel	Scrum (Agile), Hybride front- en desktopontwikkeling met focus op UX, herbruikbaarheid en technische kwaliteit.
inzetbaar in zorgcontext	

# Fullstack software developer / Migratie-specialist (VueJS & .NET) Aug 2017 – Dec 2017 Conclusion Amsterdam

# Project: Modernisering van CLASS Leeromgeving

De bestaande CLASS-leeromgeving voldeed als ASP.NET Web Forms-applicatie niet langer aan moderne eisen voor toegankelijkheid, UX en onderhoudbaarheid. Mijn opdracht was tweeledig: het beheren van de legacy-applicatie en tegelijkertijd het moderniseren van de frontend met VueJS, gericht op performance, herbruikbaarheid en gebruikersvriendelijkheid.

Ik ontwikkelde een modulaire frontend met VueJS, gebaseerd op HTML, CSS en JavaScript (ECMA 5/6). Componenten werden opgebouwd volgens component-based designprincipes, met focus op toegankelijkheid, semantiek en schaalbaarheid. Ik implementeerde herbruikbare UI-elementen en voerde optimalisaties uit voor foutafhandeling, validatie en performance. De frontend communiceerde via RESTful WebAPI's met een backend in ASP.NET Web Forms, en maakte gebruik van CQRS en NHibernate.

Daarnaast fungeerde ik als aanspreekpunt richting klant bij vragen over SLA's, performance en technische issues. Ik voerde root causeanalyses uit, documenteerde bevindingen en initieerde verbeteracties in samenwerking met het team. Testkwaliteit werd geborgd via xUnit-tests en SonarQube. In het Scrumteam leverde ik actief input tijdens planningen, demo's en sprintreviews.

De succesvolle migratie naar VueJS resulteerde in een snellere en gebruiksvriendelijkere leeromgeving. De stabiliteit en performance namen merkbaar toe, terwijl de onderhoudslast afnam. Tegelijk bleef de beschikbaarheid voor klanten gegarandeerd. Mijn werk leverde een hogere klanttevredenheid en betere UX op.

Technieken & Tools	Methodieken
VueJS, HTML5, CSS3, JavaScript (ECMA 5/6), ASP.NET Web Forms, C#, RESTful	Scrum (Agile) – Migratie- en transformatieproject
WebAPI, CQRS, NHibernate, SQL, Sitecore ASP.NET, xUnit, TFS, Jenkins, Octopus	binnen multidisciplinair team met focus op
Deploy, TSUnit, Prettier, ESLint, SourceTree, Bitbucket, Jira, Confluence, UI-Kit CSS	testbaarheid, performance en SLA's.

Fullstack software developer (Angular & .NET Core)		
Augustus 2016 – Augustus 2017	(÷1-	
Transedev		ransdev
Werklocatie: Hilversum	mosii	ETT NOPIRED BY TOO
Voertaal: Engels		
Project: Generieke API-laag en Reizigersinterface voor Mobili	eitssystemen	

#### Project 1: API Garden - Gestandaardiseerde API-laag voor legacy-integratie

Transdev beschikte over een verouderde Delphi-backend waarvan de functionaliteit toegankelijk moest worden gemaakt voor moderne web- en mobiele applicaties. Ik ontwierp en bouwde een generieke API-laag ('API Garden') met RESTful WebAPI's in .NET Core, die fungeerde als veilige en herbruikbare koppellaag tussen nieuwe frontends en legacy-logica. Deze API werd voorzien van moderne securityprincipes, authenticatie en autorisatie op applicatieniveau.

Dankzij de nieuwe architectuur konden externe systemen veilig en uniform communiceren met het bestaande backendsysteem. De API Garden was een belangrijke stap richting een modulair en toekomstbestendig applicatielandschap.

#### Project 2: Telefonie-integratie - Wachtrijmodule met real-time statusinformatie

Reizigers die Transdev telefonisch benaderden kregen geen inzicht in wachttijden, wat tot onvrede en herhaalverkeer leidde. Ik ontwikkelde een wachtrij-integratiemodule waarmee realtime statusinformatie van bellers inzichtelijk werd via het ICE-lib platform. Wanneer een beller in de wachtrij werd geplaatst, kregen gebruikers direct een visuele weergave van wachttijd en positie.

Deze uitbreiding verhoogde de transparantie richting klanten en verminderde de druk op de telefonische service. Het leverde aantoonbaar meer klanttevredenheid op en werd structureel opgenomen in de bredere serviceketen.

# Project 3: Reizigersportaal – Angular-app voor real-time ritinformatie

Voor het reizigersportaal bouwde ik een Angular 2-applicatie waarmee reizigers real-time informatie over bus- en taxiritten konden opvragen. Ik ontwikkelde componenten met TypeScript en RxJS en bouwde een dynamische frontend waarmee gebruikersinformatie uit de API op een toegankelijke en gebruiksvriendelijke manier werd gepresenteerd.

De UI was opgebouwd met HTML en CSS en geoptimaliseerd voor mobiel gebruik. De applicatie verbeterde de selfserviceopties voor reizigers en verminderde de druk op klantcontactkanalen.

Technieken & Tools	Methodieken
Angular 2, TypeScript, RxJS, HTML5, CSS3, RESTful WebAPI, .NET Core, ASP.NET	Scrum (Agile), Multidisciplinair team met focus op
MVC, C#, SQL Server, ICE-lib, xUnit, API Gateway, Jenkins, SonarQube, TSLint,	schaalbare integraties, realtime data en selfservice-
Bitbucket, Jira, Confluence, SourceTree, OWASP, JWT, OATH 2	oplossingen.