

Onderzoek tools & middelen

Wesley Vermeulen - ICT & Media Design



Inhoudsopgave

Inhoudsopgave	2
Inleiding	3
Onderzoeksmethode	3
Ontwikkelomgevingen	04
Conclusie ontwikkelomgevingen	08
Verschil front- en backend	09
Front-end (JavaScript) frameworks	10
React	12
Vue	13
Conclusie front-end (JavaScript) frameworks	16
Backend frameworks	17
Django	17
Ruby on Rails (Rails)	19
Laravel	20
Conclusie backend frameworks	22
CSS preprocessors	23
Sass / SCSS	23
Less	24
Conclusie CSS preprocessors	24
Tools & hulpmiddelen	25
Can I use	25
CodePen	25
JSFiddle	26
Chrome DevTools	27
Code Validators	27
PageSpeed Insights	28
Versiebeheer	30
Conclusies	31
Literatuurlijst	32

Inleiding

Er zijn veel verschillende tools en middelen om een product te ontwikkelen. Denk hierbij aan frameworks in verschillende talen, zoals Laravel (PHP) als backend framework of React (JavaScript) als front-end framework. Maar zo zijn er ook verschillende opties om CSS mee te schrijven, zoals SCSS of Less.

Om ervoor te zorgen dat de productentool van Buro 3 zo optimaal mogelijk ontwikkeld wordt en om ervoor te zorgen dat de beste tools en middelen worden gebruikt ga ik onderzoek doen naar verschillende tools en middelen. Met de conclusies uit dit onderzoek kan ik vervolgens betere keuzes maken tijdens het maken van een prototype en het ontwikkelen van de tool.

De hoofdvraag bij dit onderzoek luidt als volgt:

“Welke tools en middelen zijn er op de markt, wat zijn de voor- en nadelen ervan en op welke manier kunnen deze tools en middelen een rol spelen in de productentool van Buro 3?”

Om de hoofdvraag te beantwoorden, zullen de tools en middelen onderverdeeld worden in kleinere categorieën zoals ontwikkelomgevingen, front- en backend frameworks, etc. De resultaten van deze categorieën vormen vervolgens een conclusie waarmee antwoord op de hoofdvraag gegeven kan worden.

Onderzoeksmethode

Om antwoord te geven op de onderzoeksvraag wordt er literatuuronderzoek (bibliothecaair onderzoek) verricht. De bronnen die hiervoor gebruikt worden, zijn terug te vinden in de literatuurlijst aan het einde van dit onderzoek. Daarnaast zullen ook eigen ervaring en de huidige tools en middelen van Buro 3 een rol gaan spelen tijdens het kiezen van tools of middelen voor de ontwikkeling van de productentool.

Ontwikkelomgevingen

De eerste stap lijkt heel vanzelfsprekend, maar er zijn heel veel verschillende soorten ontwikkelomgevingen en text editors voor verschillende soorten projecten. Er zit om te beginnen een groot verschil tussen een integrated development environment (IDE) en een tekst editor. Tekst editors zoals Sublime Text of Atom geven je alleen de mogelijkheid om code te schrijven terwijl je met een IDE ook nog verschillende tools tot je beschikking hebt om o.a. te automatiseren of te testen (Digambar, 2021). Hieronder zijn de bekendste IDE's en text editors uitgelicht en worden de voor- en nadelen ervan beschreven.

■ **Visual Studio Code (VS Code)**

VS Code is een gratis, open source text editor ontwikkeld door Microsoft.

Volgens verschillende enquêtes uit 2018, 2019 en 2020 door State of JS en State of CSS komt VS Code als beste text editor naar boven. In 2020 gebruikte 86% van de onderzochte personen VS Code terwijl dit in 2019 nog maar 56,6% was (State Of JS, 2020).

Features

Singh heeft enkele features van VS Code beschreven, namelijk:

- Syntax highlighting
- Functie definities
- Code debuggen
- Custom hotkeys
- Templates
- Git integratie
- Custom thema's

Voordelen

VS Code heeft veel voordelen, hieronder worden er een aantal genoemd waarmee veel mensen het eens zijn volgens Stackshare:

- Ingebouwde terminal
- Snel
- Gratis
- Markdown support
- Veel extensies (die ook bijgewerkt worden)
- Git integratie

Nadelen

Natuurlijk zitten er ook nadelen aan het gebruik van VS Code, Sunny heeft in een blog op Medium er een aantal beschreven:

- Sommige updates kunnen je systeem laten crashen
- Wordt soms erg 'laggy'

Atom

Atom is een open source text editor dat ontwikkeld is door GitHub. Zoals ze zelf aangeven is het een 'hackable text editor for the 21st century'. Met 'hackable' bedoelen ze dat het developers de mogelijkheid geeft om Atom compleet aan te passen naar hun eigen wensen en behoeften. Volgens het onderzoek van State of JS wordt Atom maar door 5,5% van de onderzochte personen gebruikt in zowel 2019 als 2020.

Features

Ook voor Atom heeft Singh enkele features beschreven:

- Zoek en vervang tekst in alle projecten
- Makkelijk bestanden zoeken en openen in één scherm
- Teletype, samenwerken met andere developers binnen de editor
- Compleet aan te passen/hackable
- Package manager

Voordelen

Zowel voordelen als nadelen zijn er moeilijk te vinden zonder reviews te lezen, uit een onderzoek van Software Testing Help waarbij Atom vergeleken is met VS Code komen de volgende voordelen:

- Gratis
- Veel extensies/plugin-ins beschikbaar
- Ontwikkeld door GitHub, dus git integratie is erg goed

Nadelen

Ook de nadelen komen uit het eerdere onderzoek van Software Testing Help:

- Bij veel extensies/plugin-ins wordt atom langzaam
- Atom is heel standaard en alle features moeten vanuit extensies/plugin-ins komen

Sublime Text

Sublime is een cross-platform editor en het uiterlijk is geïnspireerd op de teksteditor Vim. Het programma is uit te breiden door middel van plug-ins.

Features

Features die Singh over Sublime Text heeft geschreven:

- Aan te passen om productiviteit te verhogen
- Ga naar definitie / ga naar symbool
- Meerdere dingen selecteren
- Split editor
- Direct project switchen
- Custom key bindings, menu's, etc.

Voordelen

De voordelen van Sublime Text komen uit enkele reviews op Capterra die in 2020 /2021 zijn geschreven:

- Volledig aan te passen d.m.v. plug-ins
- Erg snel
- Gratis
- Code formatteren

Nadelen

Ook de nadelen komen van de reviews op Capterra:

- Constant de melding om de betaalde versie van Sublime Text te kopen
- Geen geïntegreerde terminal
- Omdat het 3rd party plug-ins zijn mist er wel eens documentatie

PhpStorm

PhpStorm is een PHP IDE die gemaakt is om met frameworks zoals Symfony, Laravel, WordPress, Magento, etc. te werken. Daarnaast kun je direct gebruik maken van versiebeheer, remote deployment, databases/SQL en nog veel meer tools.

Features

Enkele features van PhpStorm volgens Singh:

- Visuele debugger
- Automatische code aanvullen
- Errors zijn goed te zien
- Database / SQL integratie
- Command-line tools
- Code navigator

Voordelen

Op Capterra staan veel reviews over PhpStorm, hier zijn enkele voordelen uitgekomen:

- Standaard vol met features zoals debugger of git integratie
- Short keys
- Professionele interface

Nadelen

Naast voordelen zijn er ook een aantal nadelen:

- Betaald
- Erg zware en grote applicatie voor de pc

Brackets

Met de focus op visuele tools en preprocessor support is Brackets een moderne text editor die het makkelijk maakt om in de browser te designen. Daarnaast is Brackets ook nog eens open source. Vanaf september biedt Adobe geen support meer voor Brackets en zij adviseren gebruikers om over te stappen naar VS Code.

Features

Singh heeft ook verschillende features van Brackets beschreven:

- Inline editor
- Live voorbeeld
- Split editor
- Preprocessor support
- Makkelijke UI
- Veel nuttige extensies beschikbaar

Voordelen

In een onderzoek van Developer Resources uit 2020 komen de volgende voordelen:

- Gratis
- Ingebouwde browser met live preview

Nadelen

Uit hetzelfde onderzoek van Developer Resource komt het onderstaande nadeel:

- Gefocust op front-end development

Coda

Coda is een ontwikkelomgeving uitsluitend voor MacOS dat ontwikkeld is door Panic. Het werd voor het eerst uitgebracht in 2007 en won de Apple Design Award 2007 voor de beste gebruikerservaring. Coda 2 werd uitgebracht in 2012. Volgens State Of JS gebruiken niet veel mensen Coda (maar 1,9%) en daarnaast wordt Coda niet langer meer geüpdate aangezien Coda is veranderd in Nova.

Features

Hieronder zijn enkele nieuwe features van Code (2) beschreven (Toolkit, n.d.):

- Code in- en uitvouwen
- Automatisch code aanvullen
- Visuele tabjes
- Focus op code
- Ingebouwde MySQL editor

Voordelen

Er zijn niet veel voor en nadelen van Coda 2 te vinden, de voordelen die ik heb gevonden zijn:

- Ingebouwde FTP
- Geïntegreerde terminal

Nadelen

Zoals hierboven aangegeven is er weinig te vinden over Coda 2, uit een review van Accentsconagua komen de volgende nadelen:

- Alleen voor MacOS
- Geen ondersteuning voor automatische CSS aanvulling (voorvoegsels Opera en Microsoft)
- Geen tot weinig plug-ins

■ Conclusie ontwikkelomgevingen

Er zijn dus veel verschillende text editors / IDE's met verschillende voor- en nadelen. Door de jaren heen heb ik verschillende text editors / IDE's gebruikt, zoals: VS Code, Atom, Sublime Text, PhpStorm en Brackets en hoewel ze bij Buro 3 werken met Coda 2 of Sublime Text, gaat mijn voorkeur uit naar Visual Studio Code. Naast het feit dat deze IDE door heel veel personen gebruikt wordt en gratis is, is er goede documentatie over te vinden en zijn er veel handige plug-ins die de productiviteit kunnen verhogen.

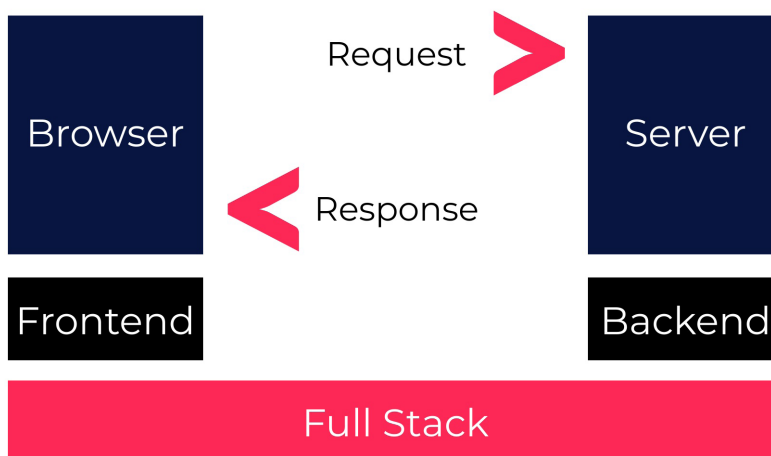
Daarnaast is deze IDE erg snel en makkelijk in gebruik, je kunt direct aan de slag met verschillende programmeer- en codeertalen en door de ingebouwde terminal én git integratie heb je alle benodigdheden om makkelijk aan een project te werken.

Het ligt natuurlijk erg aan het soort project wanneer je een IDE kiest, zo gaat mijn voorkeur voor een Laravel project meteen uit naar PhpStorm. Deze IDE is specifiek gemaakt voor PHP development en dus erg handig i.c.m. Laravel (PHP framework). Helaas is deze IDE niet erg snel en zwaar voor je systeem, waardoor het voor kleinere projecten minder handig is. Daarnaast kun je niet even snel losse documenten openen zonder een compleet project aan te maken.

Verskil front- en backend

Een webapplicatie kan een front-end en een backend bevatten. Een front-end is het visuele gedeelte van een webapplicatie, oftewel wat de gebruiker ziet en waarmee hij/zij interacteert (UI/UX). De backend is alles wat met de server te maken heeft, bijvoorbeeld de verbinding met de database, functies om een gebruiker te controleren, etc. (Intagleo, 2020).

Verskil tussen front-end en backend development



Er zit ook verschil tussen front-end developers en backend developers. Zoals eerder genoemd is de front-end alles wat de gebruiker ziet, een front-end developer houdt zich dus bezig met alle elementen die gebruikers zien in de browser (UI). Front-end developers zijn niet alleen verantwoordelijk voor de UI, maar ook voor de gebruikerservaring van een website of webapplicatie (UX). Front-end talen zijn HTML, CSS en JavaScript en front-end frameworks zijn bijvoorbeeld Angular, React en Vue.

Een backend developer is verantwoordelijk voor de logica die de front-end en de backend van een webapplicatie verbindt, daarnaast zijn ze verantwoordelijk voor het maken, onderhouden, testen en debuggen van de backend van een webapplicatie. Hieronder valt onder andere dus de logica, database, API proces, etc. Backend talen zijn bijvoorbeeld PHP, Ruby en Python en backend frameworks zijn onder andere Laravel, Django en Ruby on Rails.

F Front-end (JavaScript) frameworks

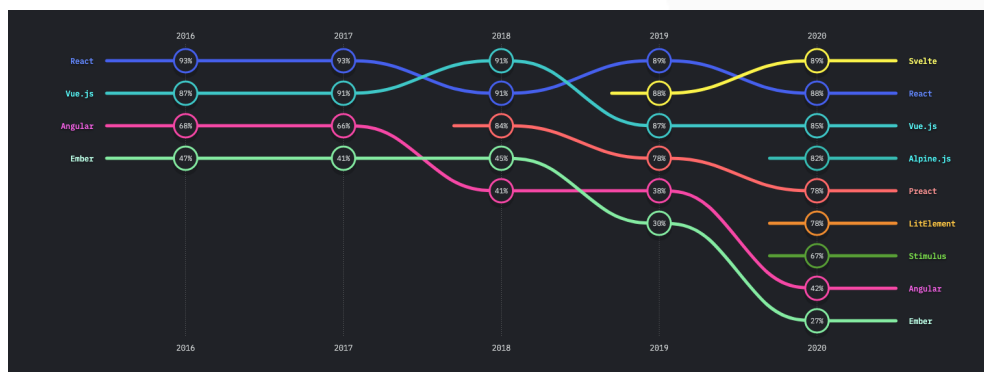
Het verschil tussen een library zoals jQuery en een JavaScript framework is de interactie/manipulatie van de DOM. Frameworks interacteren met de browser en de DOM, terwijl jQuery elementen manipuleert door ze aan te spreken in de DOM zoals Flavio Copes in een blog verteld.

Daarnaast geeft Flavio ook aan dat frameworks zoals Backbone, Ember en AngularJS de eerste golf van waren van moderne JavaScript frameworks. De tweede golf waar we op dit moment in zitten, heeft React, Angular en Vue als belangrijkste frameworks.

Een JavaScript framework is dus erg interessant omdat de productentool van Buro 3 interactief moet worden. Om deze reden heb ik dus ook uitgebreid onderzoek gedaan naar enkele JavaScript frameworks en een conclusie getrokken welk framework het beste bij het eindproduct past.

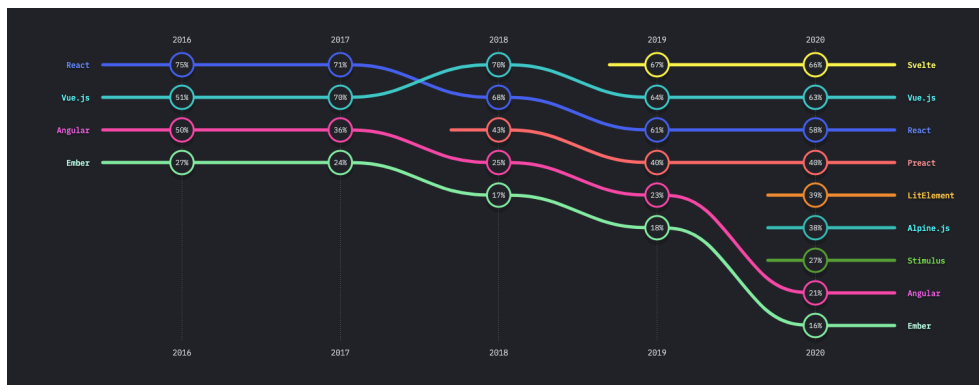
In de onderstaande afbeelding zijn resultaten te zien van een onderzoek van State Of JS. De verschillende grafieken geven verschillende soorten rankings aan. Zo hebben ze onderzoek gedaan naar de tevredenheid, interesse, gebruik en het bewustzijn van front-end frameworks (State Of JS, 2020).

Tevredenheid front-end frameworks, onderzoek State Of JS



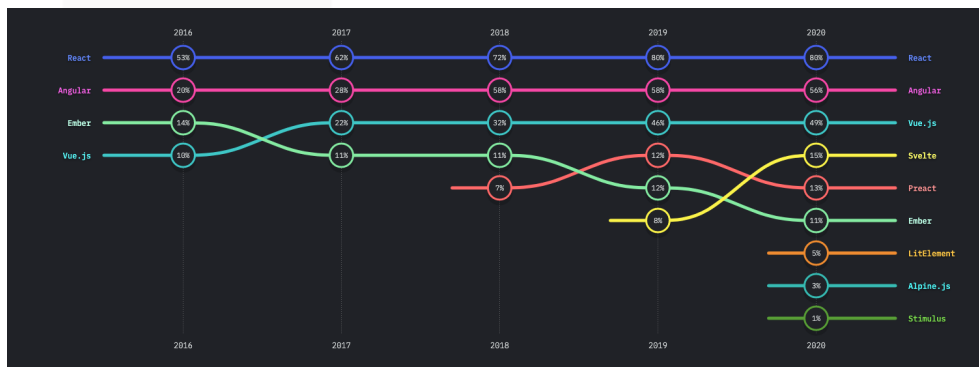
Zoals je ziet is er door de jaren heen veel veranderd qua frameworks en de tevredenheid ervan. De top drie frameworks waar gebruikers op dit moment erg tevreden over zijn liggen erg dicht bij elkaar en zijn Svelte, React en Vue. Ook zie je dat de tevredenheid van Angular erg omlaag is gegaan door de jaren heen.

Interesse front-end frameworks, onderzoek State Of JS



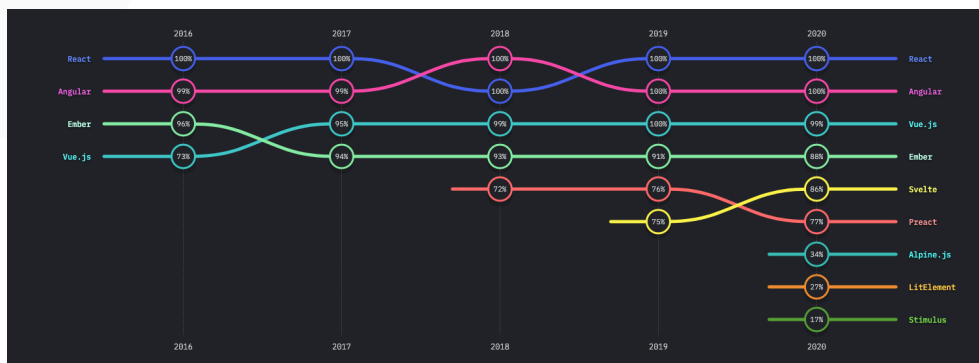
De gebruikers hebben in 2020 de meeste interesse in Svelte en Vue, die erg dicht bij elkaar liggen. Zoals State Of JS zelf aangeeft, houdt de interesse in hoe erg de gebruikers het framework willen leren. Je ziet dat de interesse in React al een aantal jaar op rij omlaag gaat. De interesse in Angular is ook erg laag in 2020.

Gebruik front-end frameworks, onderzoek State Of JS



Ondanks dat het interesse in React omlaag gaat, is het gebruik nog steeds erg hoog. Het percentage dat Angular gebruikt gaat langzaam omlaag in tegenstelling tot Vue dat langzaam omhoog gaat.

Bewustzijn front-end frameworks, onderzoek State Of JS



De top drie frameworks in deze categorie zijn ‘by far’ React, Angular en Vue, die ongeveer allemaal 100% zijn.

De frameworks die ik ga onderzoeken zijn React en Vue. Deze twee worden op dit moment veel gebruikt en hoewel de tevredenheid en interesse in Svelte erg hoog ligt, is dit framework nog niet zo lang op de markt vergeleken met de andere. Daarnaast is de tevredenheid en interesse in Angular erg hard omlaag gegaan, waardoor ik deze niet meeneem in het onderzoek. Voor elk framework ga ik onderzoeken wat de voor- en nadelen ervan zijn en uiteindelijk zal ik een conclusie trekken over welk framework er gebruikt kan worden voor de productentool van Buro 3.

React

Zoals React zelf aangeeft is het een declaratieve, efficiënte en flexibele JavaScript library voor het bouwen van gebruikersinterfaces. Hiermee kun je complexe gebruikersinterfaces samenstellen uit kleine geïsoleerde stukjes code die “componenten” worden genoemd (React, n.d.).

Altexsoft voegt hieraan toe dat ReactJS een JavaScript library is die de snelheid van JavaScript en een nieuwe manier van het renderen van webpagina's combineert, waardoor ze erg dynamisch en reageren op de input van de gebruiker (Altexsoft, 2020).

Daarnaast heeft Altexsoft ook enkele voor- en nadelen van React beschreven die hieronder terug te vinden zijn.

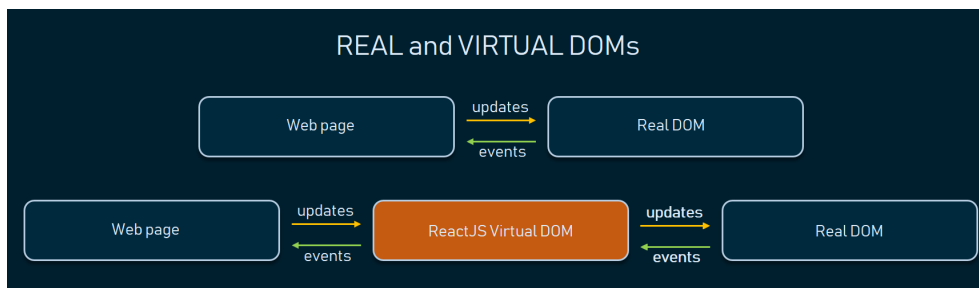
Voordelen

Virtual DOM zorgt voor betere UX en snellere ontwikkeling

Wanneer een gebruiker input geeft op een website, wordt dit constant gecheckt door een server die vervolgens weer de juiste response teruggeeft. Daarvoor moet ook de complete DOM geüpdate worden, wat tegenwoordig niet de juiste oplossing is omdat de DOM enorm veel elementen bevat.

Hiervoor heeft React een oplossing bedacht, zij maken namelijk een kopie van de DOM en laten deze aan de gebruikers zien. Wanneer er dan een input komt vanuit de gebruiker passen ze alleen het juiste element aan en wordt dit direct geüpdatet zonder dat de complete DOM geüpdate hoeft te worden. Dit zorgt voor een snelle updates en het bouwen van erg dynamische UI.

Tevredenheid front-end frameworks, onderzoek State Of JS



Hergebruiken van React componenten

Componenten kunnen ten alle tijden hergebruikt worden, wat veel tijd kan kosten. Je kunt dit ook zien als een designer die niet iedere keer weer een logo hoeft te tekenen, maar ten alle tijden verschillende assets kan gebruiken.

Data flow in één richting

React maakt gebruik van neerwaartse gegevensbinding, waardoor veranderingen in de child componenten geen gevolgen hebben voor de parent componenten. Hierdoor wordt code stabiel.

Open-source Facebook library

Omdat React een van de eerste open-source JavaScript projecten was dat uitgegeven is door Facebook, zijn er veel applicaties en extra tools gemaakt door developers buiten Facebook. Zo zijn een aantal belangrijke features gemaakt door developers die het project op GitHub zagen staan. Op dit moment staat React in de top 5 trending projecten op GitHub met meer dan 152.000 sterren en 1400 gebruikers die werken aan de library.

Nadelen

Ontwikkelingsnelheid

Doordat React constant verandert moeten developers constant nieuwe manieren leren om problemen op te lossen. Hierdoor kunnen sommige developers alle veranderingen niet meer bijhouden.

Slechte documentatie

Dit nadeel heeft vooral te maken met de ontwikkelingsnelheid. Door de constante veranderingen en het uitbrengen van nieuwe versies van verschillende tools is er geen tijd om fatsoenlijke documentatie te schrijven. Hierdoor wordt er veel gedocumenteerd door developers zelf wanneer ze gebruik maken van een bepaalde tool, maar is er moeilijk officiële documentatie te vinden.

HTML in JavaScript

React maakt gebruik van JSX, een syntax extensie wat het mogelijk maakt om HTML en JavaScript door elkaar te gebruiken. Het gebruik van JSX beschermt onder andere tegen code injections, maar veel leden van de development community vinden dat de code erg lijkt op oude spaghetti code, daarnaast klagen ze over de complexiteit van JSX en de moeilijke leercurve.

Vue

Zoals Vue zelf omschrijft is Vue een progressief framework voor het bouwen van user interfaces. In tegenstelling tot andere monolithische frameworks, is Vue vanaf de basis ontworpen om stapsgewijs te kunnen worden toegepast. De core library is alleen gericht op de “view layer” en makkelijk te integreren met andere libraries of bestaande projecten. Daarnaast is Vue ook perfect in staat om geavanceerde Single-page applicaties te ondersteunen wanneer het gebruikt wordt in combinatie met moderne tools en ondersteunende libraries (Vue.js, n.d.).

Ook over Vue heeft altexsoft een blog geschreven en voegt hieraan toe dat Vue niet alleen gebruikt wordt voor web interfaces, maar ook voor zowel desktop als mobiele app development. De HTML extensie en de JavaScript basis hebben Vue in korte tijd een grote favoriet gemaakt, wat resulteerde in het gebruik door bedrijven zoals Adobe, Behance, Alibaba, Gitlab en Xiaomi (Altexsoft, 2019).

In een interview met Evan (Cromwell, 2016), de maker van Vue, wordt verteld dat Vue een poging was om het beste uit Angular te halen en hiermee een custom tool te maken die veel lichter was. Zoals hij aangeeft: "For me, Angular offered something cool which is data binding and a data driven way of dealing with a DOM, so you don't have to touch the DOM yourself."

Altexsoft heeft ook voor Vue enkele voor- en nadelen beschreven die hieronder te lezen zijn.

Voordelen

Klein

De totale grootte van het framework is 18 KB. Dit is erg klein voor een library en hierdoor niet alleen snel te downloaden en installeren, maar dit heeft ook een positieve impact op de SEO en UX.

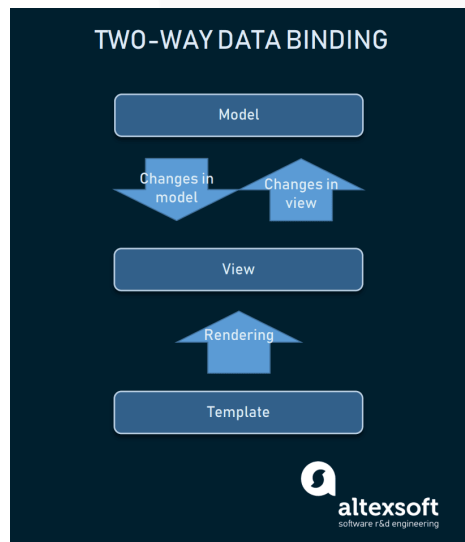
Virtuele DOM

Net zoals React maakt Vue gebruik van een virtuele DOM, waardoor het updaten en renderen van een pagina erg snel wordt en de prestatie van een applicatie verbeterd wordt.

Two-way data binding

Vue maakt gebruik van "two-way data binding" dat is overgenomen van Angular. Dit houdt in dat er een verbinding is tussen model data en de UI. Componenten bevatten data die ten alle tijden geüpdatet kunnen worden, waardoor het makkelijker wordt om gerelateerde componenten te updaten en data te volgen. Daarnaast kunnen er hierdoor real-time updates uitgevoerd worden.

Datamodel van Vue



Componenten in één bestand

Alle stukjes van je Vue applicatie zijn componenten die verschillende elementen vormen in je interface. Al deze componenten kunnen geschreven worden in één document zonder ze te verdelen in losse documenten. Daarnaast is het nog steeds mogelijk om componenten in losse documenten te schrijven om leesbaarheid van code te verhogen.

Sterk ecosysteem van tools

Tijdens het bestaan van Vue heeft het veel krachtige tools gekregen waarmee je kunt werken. Zo is er support voor Babel en TypeScript, unit testing, end-to-end test tools en een plugin installatie systeem. Daarnaast heeft Vue ook haar eigen debug tool voor de browser, etc.

Makkelijk te leren

Om aan de slag te gaan met Vue heb je geen diepgaande kennis van libraries, JSX en TypeScript nodig, zoals wel het geval is bij veel andere front-end frameworks. Je hebt slechts basiskennis van HTML, CSS en JavaScript nodig.

Ook wordt Vue ondersteund door populaire IDE's en text editors zoals Sublime Text, Visual Studio Code en Atom wat het leren van Vue makkelijker maakt.

Beknopte documentatie

De documentatie van Vue is voor elk type persoon erg geschikt. Het heeft een goede structuur en bevat alle mogelijke onderwerpen. Het beschrijft alles van installatie tot moeilijkere onderwerpen zoals interactiviteit en schaalbaarheid van de app. Daarnaast wordt Vue ook vergeleken met andere JavaScript frameworks zoals React en Angular.

Community support

Vue bestaat dankzij haar community. Hierdoor zijn leden van deze community erg actief op zowel Discord en andere fora.

Nadelen

Taalbarrière

Omdat Vue erg populair is in China, is een groot deel van de content en discussies op het internet in het Chinees. Wanneer er wordt gezocht naar oplossingen, plug-in beschrijvingen of andere instructies is er de kans dat deze alleen in het Chinees te vinden zijn.

Gebrek aan ondersteuning voor grote projecten

Vue is een vrij nieuw framework en de community en het development team is in vergelijking tot Angular nog erg klein. Daarnaast wordt het niet gefinancierd door grote bedrijven.

Voordat Vue gebruikt gaat worden in grote projecten moet de technologie erg stabiel zijn en goed ondersteund worden, zodat problemen snel verholpen kunnen worden. Ondanks dat Vue niet zo veel problemen heeft, wordt het op dit moment alleen nog maar gebruikt in kleinere projecten.

Gelimiteerde middelen

Zoals eerder genoemd heeft Vue veel tools waarmee je kunt werken, maar dit ecosysteem is nog lang niet zo groot als React of Angular. Bestaande plug-ins die i.c.m. andere frameworks gebruikt kunnen worden, worden vaak ook nog niet ondersteund door Vue.

Gebrek aan ervaren developers

Ondanks de grote community support zijn er nog maar weinig ervaren Vue developers.

■ Conclusie front-end (JavaScript) frameworks

Zowel React als Vue maken beide gebruik van een virtuele DOM en interactieve componenten. Dit is erg belangrijk, omdat de productentool van Buro 3 interactief moet worden. React blinkt daarentegen wel uit in de hoeveelheid tools en plug-ins die er beschikbaar zijn, dit komt mede omdat Vue minder lang op de markt is.

Daarnaast is er een groot verschil in de opbouw van React en Vue met betrekking tot HTML en CSS. React maakt gebruik van JSX, een syntax dat het mogelijk maakt om HTML in JavaScript te gebruiken. Vue daarentegen maakt gebruik van templates, waarin alle gevalideerde HTML syntax te gebruiken is. Voor veel developers is dit makkelijker te gebruiken, omdat er geen extra kennis van JSX voor nodig is. Vue biedt naast templates nog wel de mogelijkheid om JSX te gebruiken.

Betreft documentatie wint Vue het, deze is niet alleen erg uitgebreid en duidelijk, maar packages zoals Vuex en Vue Router worden onderhouden door het Vue development team zelf. In tegenstelling tot React waar vergelijkbare packages voor zijn, namelijk Redux en React Router, maar deze hebben hun eigen documentatie en development team.

Voor de ontwikkeling van de productentool van Buro 3 gaat de voorkeur uit naar Vue. De documentatie spreekt me erg aan en is heel duidelijk. Daarnaast hoef ik geen extra kennis op te doen over JSX, maar kan ik met mijn HTML en CSS kennis gewoon gebruik maken van Vue.

Ook is Vue perfect voor kleinere applicaties, omdat er geen uitgebreide code voor nodig is om een Vue project op te zetten, wat wel het geval is bij React. Vue is ook makkelijk te integreren in bestaande projecten, wat het mogelijk maakt om eerst een idee uit te werken in standaard HTML en CSS en dit later om te zetten naar Vue.

B Backend frameworks

Net als front-end frameworks, zijn er ook een heleboel verschillende backend frameworks. Deze frameworks zijn in verschillende programmeertalen geschreven en hebben verschillende features. De keuze voor een bepaalde backend framework ligt heel erg aan de programmeertaal en of je met een bepaald framework de features van je webapplicatie kunt bouwen.

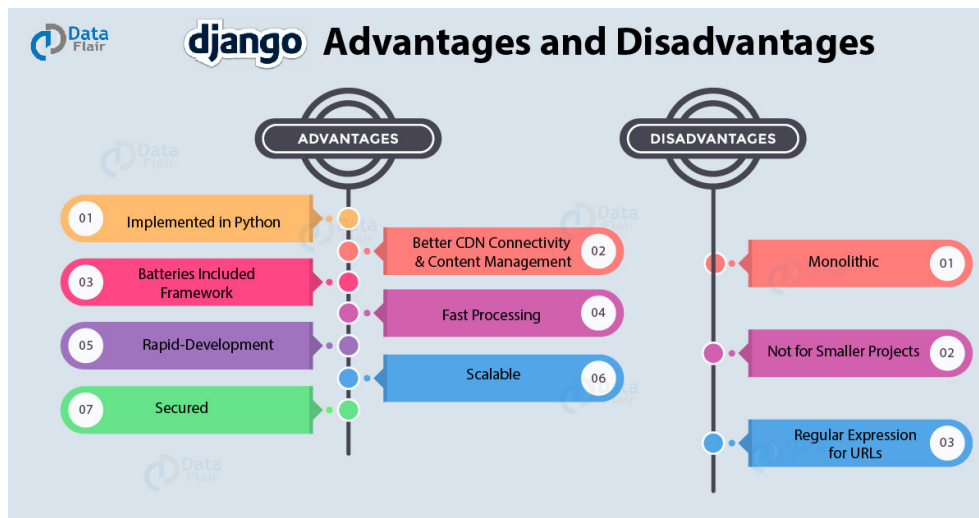
De voordelen van een backend framework zijn onder andere tijdsbesparing en schaalbaarheid, omdat er al veel functies en modules in een framework zitten waardoor het makkelijker wordt om een webapplicatie te bouwen en uiteindelijk dus ook sneller.

Er worden op het internet veel verschillende backend frameworks genoemd wanneer het aankomt op “de beste backend frameworks”. Hierdoor is het lastig ook echt de beste te kiezen en daarom ben ik, voor dit onderzoek, voor een drietal frameworks gegaan die gebruik maken van verschillende programmeertalen. De frameworks die ik ga onderzoeken, zijn: Laravel, Django en Ruby on Rails.

Deze drie frameworks komen vaak voor op verschillende websites, zijn redelijk bekend en worden gebruikt door grote organisaties. Zo geeft Jessica Clark op Back4App aan dat Django bijvoorbeeld gebruikt wordt voor Instagram en Pinterest en Ruby on Rails voor Shopify en GitHub. Laravel is een van de bekendste PHP frameworks en hier heb ik al enige ervaring mee.

Django

Volgens de website van Django, is Django een Python framework van hoog niveau dat snelle ontwikkeling en een schoon, pragmatisch ontwerp aanmoedigt. Het is gebouwd door ervaren developers en neemt een groot deel van het gedoe van webontwikkeling uit handen, zodat je de focus kunt leggen op het schrijven van je app zonder dat je het wiel opnieuw hoeft uit te vinden. Daarnaast is Django gratis en open-source. Ze geven hierbij ook 3 USP's, namelijk: snel, veilig en schaalbaar (Django, n.d.)



Voordelen

Enkele voordelen die veel voorbij komen op internet, zijn:

“Batteries included” filosofie

Deze filosofie betekent dat Django veel verschillende features en functionaliteiten bevat wanneer het wordt “geïnstalleerd”. In plaats van je eigen code schrijven, is het alleen nodig om verschillende packages te importeren voor deze functionaliteiten. Hierdoor bespaart het developers vaak veel tijd, maar kunnen zij ook de focus leggen op geavanceerde functionaliteiten van de webapplicatie.

Een aantal van deze packages zijn bijvoorbeeld:

- Database migraties
- MVC layout
- Multi-site en multi-language support
- Gratis API
- Url routing
- Ajax support
- Sitemap package om Google sitemap XML te genereren
- Etc.

Python

Django is ontwikkeld met behulp van de programmeertaal Python. Voordelen van Python zijn onder andere eenvoud en de ondersteuning voor externe libraries. Door deze voordelen wordt kan de ontwikkelsnelheid verhoogt worden. Daarnaast maakt Django het makkelijke voor developers om applicaties te ontwikkelen met nette, leesbare en makkelijk onderhoudbare code omdat het dus gebruik maakt de regels van Python.

Nadelen

Een van de grootste nadelen van Django is het gebruik voor kleinere projecten. Doordat Django met een heleboel standaard code komt waarmee functionaliteiten makkelijker te implementeren zijn, wordt het ook een zwaarder framework voor kleinere websites.

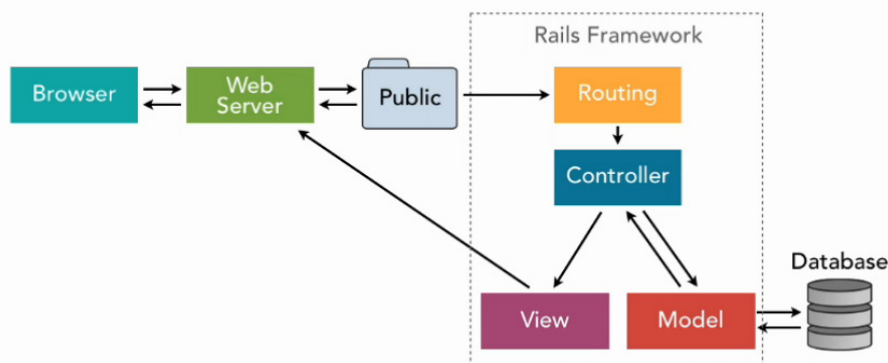
Hierdoor wordt het minder geschikt voor kleinere projecten en producten met maar een weinig features en voorwaarden.

Ruby on Rails (Rails)

Rails is een framework dat alles bevat om een webapplicatie te maken die wordt ondersteund door een database volgens het Model-View-Controller (MVC) -patroon (Ruby on Rails, 2021). Om Rails te begrijpen is het belangrijk dat het MVC-patroon begrepen wordt. Dit patroon verdeelt een applicatie in drie lagen: Model, View en Controller, met elk een specifieke verantwoordelijkheid.

Ruby on Rails MVC-patroon architectuur

Rails architecture



Voordelen

Victor (Rak, 2021) heeft enkele voordelen van Ruby on Rails benoemd:

Tijdsbesparing

Rails is relatief makkelijk te leren en wanneer je al enige ervaring hebt met Rails, kun je snel webapplicaties ontwikkelen. Het kost daarentegen wel wat tijd om alle functies van Rails te leren.

Veel handige tools en libraries

Er zijn heel veel handige tools en libraries (gems) door de community gemaakt die gebruikt kunnen worden in je eigen applicatie. Deze libraries beïnvloeden niet alleen de content van je code, maar de complete ontwikkel ervaring.

Hoge standaarden

Van de applicaties die gebruik maken van Rails wordt verwacht dat ze standaard ontwikkel patronen volgen, zoals:

- “Coding by convention”: developers moeten functies specificeren die onconventioneel zijn;
- “Don’t repeat yourself”: geen dubbele code schrijven;
- “Active record pattern”: een specifieke manier om data te verkrijgen uit de database.

Nadelen

Volgens Michal (Poczwadowski, 2017) zitten er ook een aantal nadelen aan het gebruik van Rails, namelijk:

Hoge prijs voor het maken van een fout tijdens de ontwikkeling

Het maken van fouten in de architectuur van je webapplicatie tijdens de beginfase van je project kan meer tijd kosten dan in andere frameworks. Omdat je met Rails enorm snel kunt prototypen, kan het zo zijn dat een onervaren ontwikkelteam onlogische fouten maakt waardoor de prestatie van je webapplicatie slechter wordt in de toekomst. Deze fouten zijn lastig op te lossen omdat Rails een open framework is waar alle componenten nauw met elkaar samenwerken en afhankelijk zijn van elkaar.

Gebrek aan flexibiliteit

Rails is een framework met veel standaard dependencies en modules. Om een project op te starten, moeten ontwikkelaars routing, database migraties en andere modules configureren die bij het framework worden geleverd. Deze standaard modules zijn goed als je een webapplicatie wilt maken met een aantal standaard functionaliteiten, maar ze kunnen een averechts effect op je hebben als je iets unieks in gedachten hebt.

Runtime-snelheid en prestaties

Een van de meest voorkomende argumenten tegen Rails is de trage runtime-snelheid, waardoor het lastiger wordt om je applicatie te schalen. Hoewel het waar is dat andere frameworks iets sneller zijn dan Rails, is het onwaarschijnlijk dat je iets merkt van de mindere prestatie, tenzij de applicatie vergelijkbaar is met een grote website zoals Twitter.

Daarentegen moet je er wel rekening mee houden, dat wanneer je wel iets merkt van deze mindere prestatie, het niet zo makkelijk is om de prestaties van Rails te verbeteren. Je moet dan interne componenten vervangen óf overstappen naar een ander framework.

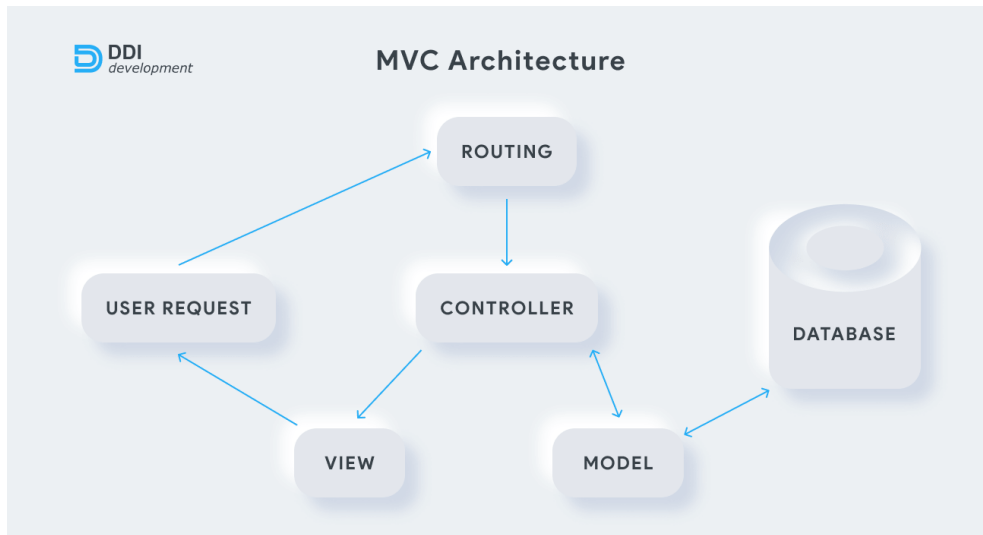
Laravel

Zoals het team zelf aangeeft, is Laravel een webapplicatie framework met een expressieve en elegante syntax. Een framework moet structuur en een startpunt bieden voor het maken van een applicatie, waardoor jij je alleen nog maar hoeft te focussen op het maken van iets geweldigs terwijl zij al het werk doen (Laravel, n.d.).

Laravel is een PHP framework met het MVC-patroon (Model, View en Controller). Hierbij regelt de model alle fundamentele dingen van een applicatie, de view zorgt voor de UI van

een applicatie en de controller krijgt de input van de gebruiker en maakt verbinding met de model en de view om alle informatie door te spelen.

Het MVC-patroon



Voordelen

DDI Development (DDI Development, 2020) heeft enkele voor- en nadelen onder elkaar gezet, in de onderstaande foto zijn kort de voordelen te zien en ik daarnaast heb ik enkele voordelen ook nog wat verder toegelicht.

Voordelen Laravel



Why to choose Laravel

- Built-in authorization system
- MVC architecture
- Cache memory integration
- Seamless automation testing
- Simple URL routing setup
- Smooth database migration
- Detail-oriented documentation
- Built-in libraries
- PHP community support

Ingebouwd authenticatiesysteem

Alle benodigde opties voor een authenticatiesysteem zijn standaard ingebouwd. Hierdoor is deze logica erg geavanceerd en daardoor ook veiliger. Dit systeem bevat onder andere de login, registratie en wachtwoord vergeten stappen die met één command toegevoegd kunnen worden.

Cache integratie

Met de cache integratie kunnen objecten opgeslagen worden in het interne systeem. Hierdoor kunnen de prestaties van de applicatie verbeterd worden, waardoor er een krachtige backend opgezet kan worden met een geavanceerd geheugensysteem.

Simpele url routing configuratie

Binnen Laravel is het mogelijk om urls aan te maken naar verschillende routes en pagina's binnen je applicatie. Url routing is van vitaal belang voor je applicatie, omdat gebruikers hierdoor de juiste inhoud of pagina kunnen vinden. Daardoor krijg je een betere UX en interactiviteit, wordt het risico op een lege url verkleind en krijg je een snellere ontwikkeling door de automatische url configuratie te laden.

Nadelen

Natuurlijk zitten er ook nadelen aan het gebruik van Laravel, volgens Cyblance (Cyblance, n.d.) zijn dat:

Gebrek aan ondersteuning

In vergelijking met Django en Rails heeft Laravel een beperkte ingebouwde ondersteuning omdat het een lichtgewicht framework is. Dit probleem kan opgelost worden met tools van andere partijen, maar voor grotere applicaties kan dit een lastige taak worden.

Complex om mee te starten

Als je nieuw bent in het gebruik van Laravel is het misschien mogelijk dat je een heleboel documentatie moet doorlezen. Het is daardoor handig als je al enige ervaring hebt in Laravel wanneer je start met de ontwikkeling van een nieuwe applicatie.

Conclusie backend frameworks

Allereerst zijn het drie totaal verschillende frameworks die gebruik maken van verschillende programmeertalen. Ik denk dat Django niet het juiste framework is voor dit project, omdat het nadeel hiervan is dat het niet geschikt is voor kleinere applicaties. Daarnaast denk ik dat zowel Ruby on Rails en Laravel ook niet geschikt zijn voor dit project, omdat het handig is als je al enige kennis van deze frameworks is en omdat er binnen Rails weinig ruimte voor fouten is.

Mijn conclusie is dan ook om voor dit project geen backend framework te gebruiken. Ik heb maar beperkte tijd om dit project te ontwikkelen en aangezien ik al enige tijd kwijt ben met kennis opdoen over Vue als front-end framework denk ik dat het verstandig is als ik geen extra tijd steek in het uitvogelen van een backend framework. Om de ontwikkeling van dit project makkelijker te maken kan ik bijvoorbeeld gebruik maken van WordPress om tijd te besparen.

Dit betekent niet dat er in de toekomst geen gebruik gemaakt kan worden van een backend framework. In een later stadium zou er bijvoorbeeld gekeken kunnen worden naar Laravel, omdat hier al een authenticatie systeem zit ingebouwd en daarnaast ook e-mail integratie om bijvoorbeeld marketing mails te versturen. Ook Rails zou gebruikt kunnen worden omdat hiermee snel applicaties ontwikkeld kunnen worden.

CSS preprocessors

Een CSS preprocessor is een programma dat CSS genereert vanuit de unieke syntax van depreprocessor. Er zijn veel soorten CSS preprocessors en ze bieden allemaal extra features die niet bestaan in vanilla CSS. Hieronder zijn enkele CSS preprocessors uitgelicht.

Sass / SCSS

Met Sass kun je gebruik maken van features zoals variabelen, nesting, mixins, imports, en nog veel meer. Een Sass bestand kan uiteindelijk omgezet worden naar een standaard CSS, waarbij alle extra features in Sass zoals het gebruik van variabelen, automatisch omgezet worden naar CSS.

Het verschil tussen Sass en SCSS is het gebruik van puntkomma's en zogenoemde brackets.

Voorbeeld hoe SCSS omgezet wordt naar CSS

SCSS	Sass	CSS
<pre>nav { ul { margin: 0; padding: 0; list-style: none; } li { display: inline-block; } a { display: block; padding: 6px 12px; text-decoration: none; } }</pre>		<pre>nav ul { margin: 0; padding: 0; list-style: none; } nav li { display: inline-block; } nav a { display: block; padding: 6px 12px; text-decoration: none; }</pre>

Voorbeeld hoe Sass omgezet wordt naar CSS

SCSS	Sass	CSS
<pre>nav ul margin: 0 padding: 0 list-style: none li display: inline-block a display: block padding: 6px 12px text-decoration: none</pre>		<pre>nav ul { margin: 0; padding: 0; list-style: none; } nav li { display: inline-block; } nav a { display: block; padding: 6px 12px; text-decoration: none; }</pre>

Less

Net zoals Sass en SCSS kun je met Less gebruik maken van verschillende features zoals variabelen, nesting, etc. Daarnaast is het met Less mogelijk om gebruik te maken van 'guarded mixins'. Dit zijn mixins die alleen worden toegepast wanneer er een bepaalde voorwaarde waar is.

Guarded mixins in Less ([links](#)) en hoe ze gebruikt worden ([rechts](#))

```
.set-bg-color (@text-color) when (lightness(@text-color) >= 50%) {  
  background: black;  
}  
.set-bg-color (@text-color) when (lightness(@text-color) < 50%) {  
  background: #ccc;  
}
```

```
.box-1 {  
  color: #BADA55;  
  .set-bg-color(#BADA55);  
}
```

Conclusie CSS preprocessors

Zowel Sass als SCSS en Less zijn erg handig tijdens de ontwikkeling van een project. Bij Buro 3 maken ze gebruik van Less en zelf maak ik al jaren gebruik van SCSS. De keuze om één van deze preprocessors te gebruiken is niet zo moeilijk, aangezien het veel voordelen en extra functies toevoegt aan het schrijven van CSS.

Voor dit project ga ik gebruik maken van Less, omdat ze bij Buro 3 hier ervaring mee hebben en ik dus altijd iemand kan aanspreken voor hulp. Omdat Less erg lijkt op SCSS kan ik met de kennis die ik hierover heb makkelijk aan de slag.

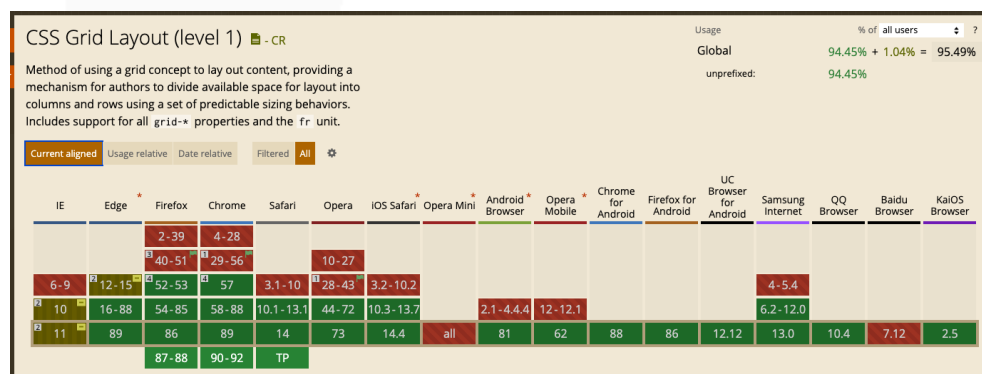
T Tools & hulpmiddelen

Naast verschillende ontwikkelomgevingen, front- en backend frameworks en CSS preprocessors zijn er nog een heleboel andere tools en hulpprogramma's waar ik gebruik van kan maken tijdens de ontwikkeling van dit project. Hieronder licht ik enkele tools en hulpprogramma's toe die niet cruciaal zijn tijdens de ontwikkeling, maar wel erg handig kunnen zijn.

Can I use

Can I use biedt up-to-date browserondersteuning tabellen aan om te checken of front-end web technologieën ondersteund worden op desktop en mobiele browsers. Op deze website zijn onder andere de categorieën CSS, HTML5 en JS te vinden en worden browsers zoals IE, Firefox en Chrome vergeleken. (Can I use, 2021).

Voorbeeld ondersteuning CSS Grid op verschillende browsers

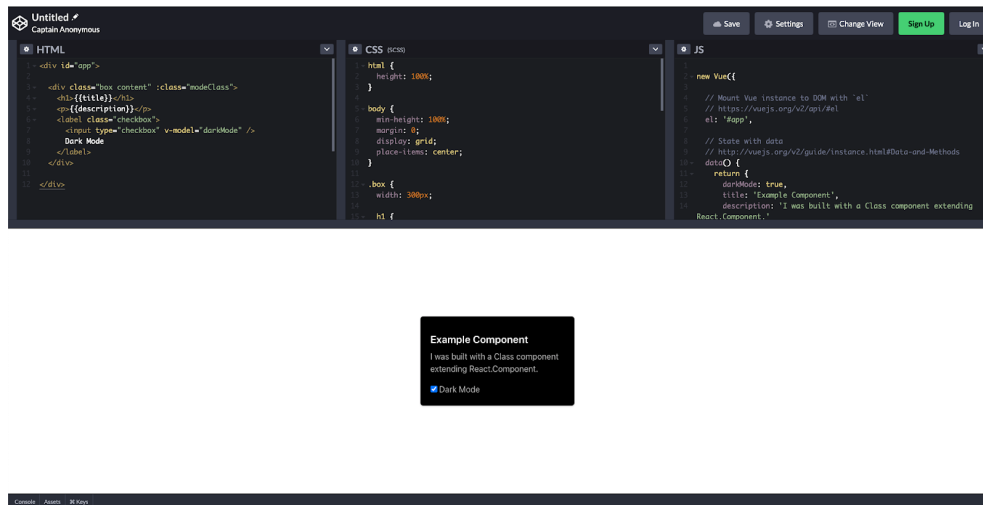


CodePen

CodePen is een sociale ontwikkelomgeving voor front-end designers en developers. Op deze website kun je werk laten zien, testcases bouwen om te leren en debuggen. Daarnaast is er ook veel inspiratie te vinden. (CodePen, n.d.).

Als je een oplossing voor een probleem aan het zoeken bent kom je vaak uit op verschillende pens op CodePen. Andere gebruikers kunnen laten zien hoe ze problemen oplossen of maken voorbeelden.

De UI van CodePen

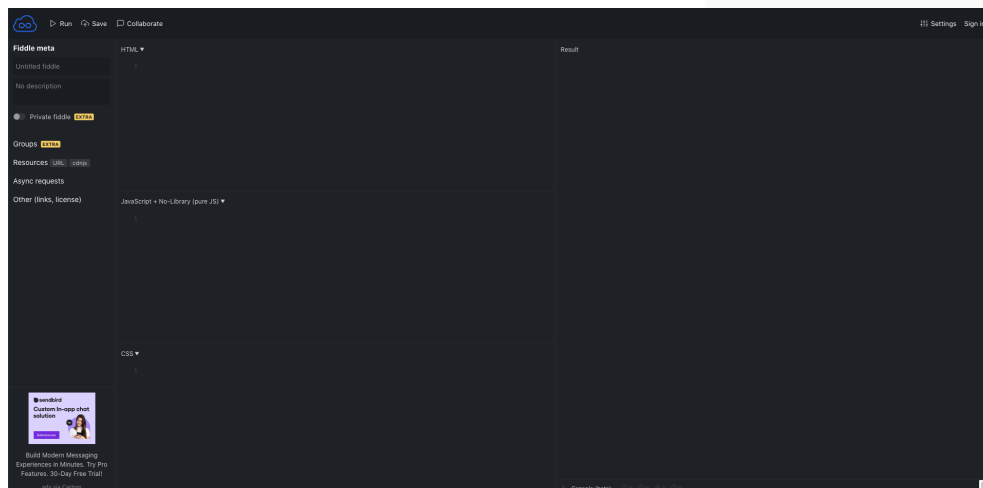


JSFiddle

JSFiddle is een online ontwikkelomgeving en online community voor het testen en presenteren van, door gebruikers gemaakte en collaboratieve, HTML-, CSS- en JavaScript-codefragmenten, ook wel ‘fiddles’ genoemd. Daarnaast is het met JSFiddle mogelijk om Ajax calls te doen, wat niet mogelijk is met bijvoorbeeld CodePen. (Wikipedia, 2020).

Net zoals CodePen kom je vaak bij een fiddle op JSFiddle uit als je een oplossing voor een bepaald probleem aan het zoeken bent. Gebruikers laten deze fiddles bijvoorbeeld achter in de comments op bijvoorbeeld StackOverflow.

De UI van JSFiddle



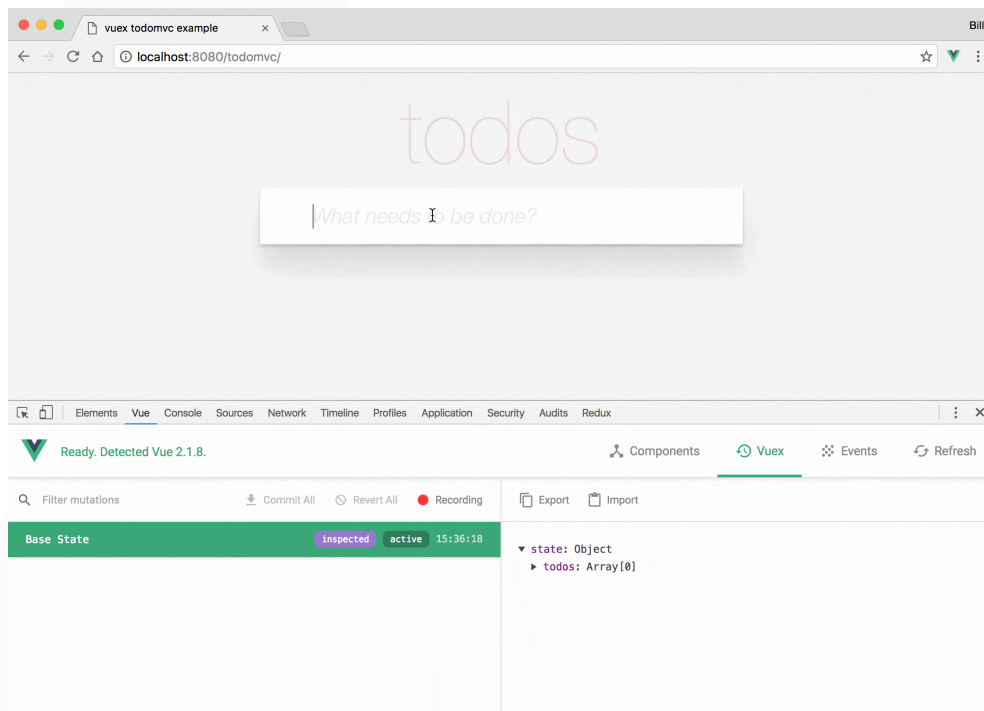
Chrome DevTools

De Chrome DevTools is een set van web developer tools die standaard in Chrome zitten. Deze tools kunnen je helpen om pagina's on-the-fly te wijzigen en problemen te onderzoeken, waardoor het je kan helpen sneller betere websites te bouwen. (Chrome DevTools, 2020).

Naast een overzicht van de elementen van een website, een ingebouwde console, een netwerk tabblad en meer, krijg je ook een overzicht van je Vue elementen met bijbehorende properties wanneer een website gebruik maakt van Vue. Hiervoor heb je alleen een Chrome extensie nodig.

Dit kan dus erg handig zijn wanneer ik aan de slag ga met de ontwikkeling van de tool, omdat ik hoogstwaarschijnlijk gebruik ga maken van Vue als front-end framework.

Voorbeeld van de Vue extensie voor Chrome



Code Validators

W3C is één van de online markup validation services die gebruikt kunnen worden om je code te valideren aan de hand van de regels die gesteld zijn voor het schrijven van bijvoorbeeld HTML.

Wanneer er bestanden of losse code door de validator gehaald worden, krijg je verschillende waarschuwingen of errors terug die vervolgens opgelost kunnen worden om de prestaties van je webapplicatie te verbeteren.

De resultaten van de check op mijn afstudeerportfolio

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change.

Showing results for <https://afstudeerstage.wesleyvermeulen.com/>

Checker input

Show

☐ source

☐ outline

☐ image report

Options...

Check by address

<https://afstudeerstage.wesleyvermeulen.com/>

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

1. **Warning** Consider avoiding viewport values that prevent users from resizing documents.

From line 14, column 1; to line 14, column 128

`<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">`

2. **Error** Start tag <body> seen but an element of the same type was already open.

From line 78, column 1; to line 78, column 6

`</script><body></body>`

Document checking completed.

Used the HTML parser. Externally specified character encoding was UTF-8.

Total execution time 364 milliseconds.

PageSpeed Insights

Deze tool analyseert de content van een webpagina en doet vervolgens suggesties om die pagina sneller te maken. Wanneer een pagina wordt geanalyseerd komen er scores uit voor verschillende categorieën. Per categorie kan er vervolgens bekeken worden waarop de pagina beter kan scoren en hoe dit opgelost kan worden. (PageSpeed Insights, n.d.).

De analyse van mijn afstudeerportfolio

PageSpeed Insights

HOME

DOCS

MOBIEL

DESKTOP

https://afstudeerstage.wesleyvermeulen.com/

ANALYSEREN

92

https://afstudeerstage.wesleyvermeulen.com/

0-49

50-89

90-100

Veldgegevens

Het rapport Chrome-gebruikerservaring heeft niet voldoende echte sn@nd@tgegevens voor deze pagina.

Origin Summary

Het rapport Chrome-gebruikerservaring heeft niet voldoende echte sn@nd@tgegevens voor deze herkomst.

Labgegevens

First Contentful Paint

1,3 s

Time to Interactive

1,3 s

Speed Index

1,3 s

Total Blocking Time

0 ms

Largest Contentful Paint

1,3 s

Cumulative Layout Shift

0,001

Waarden worden geschat en kunnen variëren. De prestatiescore wordt rechtstreeks berekend op basis van deze statistieken. See calculator.

Een andere tool waarmee gelijke analyses gedaan kunnen worden is Lighthouse. Deze tool is te vinden in de Chrome DevTools en geeft een soortgelijke score aan, met verschillende oplossingen en uitleg.

Versiebeheer

Versiebeheer is een systeem dat wijzigingen in een bestand of set bestanden in de loop van de tijd vastlegt, zodat je specifieke versies later kunt terugroepen. Programma's zoals Google Docs hebben ingebouwde versiebeheer, maar ook een kopie opslaan van bijvoorbeeld een Photoshop bestand is een voorbeeld van versiebeheer.

Voor de ontwikkeling van de productentool ga ik gebruik maken van Bitbucket. Naast bijvoorbeeld Gitlab en Github, is Bitbucket één van de vele versiebeheer programma's. Ik heb voor dit programma gekozen, omdat ze bij Buro 3 gebruik maken van Bitbucket voor alle projecten.



Conclusies

Er zijn voor de verschillende onderdelen van een project dus heel veel verschillende tools & middelen die de ontwikkelsnelheid kunnen verhogen of het oplossen van problemen makkelijker kunnen maken. Daarnaast biedt een front-end framework zoals Vue weer net andere features aan als bijvoorbeeld React.

Ik zal de conclusies uit de verschillende onderzoeken nog eens samenvatten. Qua IDE ga ik gebruik maken van Visual Studio Code. Niet alleen heb ik al ervaring met deze IDE, er zit ook git ingebouwd en heeft een ingebouwde terminal.

Mijn keuze voor front-end framework gaat uit naar Vue, aangezien ik met de kennis die ik op dit moment bezit al aan de slag kan met dit framework. Daarnaast heeft het een erg uitgebreide documentatie waarin veel voorbeelden staan en is dit framework ideaal voor kleinere applicaties omdat er geen uitgebreide code voor nodig is om een Vue project op te zetten. Ook kan Vue ook op een later moment toegevoegd worden aan een project.

Voor dit project ga ik geen gebruik maken van een backend framework. Niet alleen heb ik maar beperkte tijd om te ontwikkelen, waardoor ik dus minder tijd heb om uit te vogelen hoe de verschillende frameworks werken, maar de tool heeft in dit stadium nog geen geavanceerde functies. Dit betekent niet dat er in de toekomst geen gebruik gemaakt kan worden van een backend framework, maar voor deze versie is het nog niet van belang dat er gebruik wordt gemaakt van een backend framework.

Qua CSS preprocessor zou ik zowel Sass als SCSS of Less kunnen gebruiken. Ik ga voor dit project sowieso gebruik maken van een CSS preprocessor, omdat je dan gebruik kan maken van features zoals variabelen, mixins of functies binnen je CSS. Daarnaast heb ik al ervaring met Sass/SCSS en gebruiken ze binnen Buro 3 voor alle projecten Less.

De tools en hulpprogramma's die ik heb benoemd zullen misschien niet allemaal evenveel of überhaupt gebruikt worden, maar kunnen houvast bieden wanneer ik tegen problemen aanloop of wanneer ik inspiratie wil opdoen. Daarnaast kunnen ze helpen bij het optimaliseren van de tool.

Als laatste ga ik Bitbucket gebruiken als versiebeheer voor dit project. Deze tool wordt op dit moment al door Buro 3 gebruikt voor alle lopende en afgeronde projecten en dit was daarom een logische keuze om ook te gebruiken tijdens de ontwikkeling van de productentool.

Literatuurlijst

Accentsconagua. (n.d.). Coda 2 Beoordeeld. Accentsconagua.

<https://nl.accentsconagua.com/articles/code/coda-2-reviewed.html>

Altexsoft. (2019, september 11). The Good and the Bad of Vue.js Framework Programming. Altexsoft.

<https://www.altexsoft.com/blog/engineering/pros-and-cons-of-vue-js/>

Altexsoft. (2020, juli 7). The Good and the Bad of ReactJS and React Native. Altexsoft.

<https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-reactjs-and-react-native/>

Can I use. (2021, februari 27). About. Can I use.

<https://caniuse.com/ciu/about>

Chrome DevTools. (2020, juli 14). Chrome DevTool. Chrome DevTool.

<https://developers.google.com/web/tools/chrome-devtools>

Clark, J. (n.d.). Top 10 backend frameworks. Back4App.

<https://blog.back4app.com/backend-frameworks/>

CodePen. (n.d.). The best place to build, test, and discover front-end code. CodePen.

<https://codepen.io/>

Copes, F. (2020, maart 22). What is a JavaScript Frontend Framework? Flavio Copes.

<https://flaviocopes.com/what-is-a-frontend-framework/>

Cromwell, V. (2016, november 3). Evan You. Between the Wires.

<https://web.archive.org/web/20170603052649/https://betweenthewires.org/2016/11/03/evan-you/>

Cyblance. (n.d.). Pros And Cons Of Using The PHP Laravel Framework For Web App Development. Cyblance.

<https://www.cyblance.com/laravel-framework/pros-and-cons-of-using-the-php-laravel-framework-for-web-app-development/>

DDI Development. (2020, juli). Advantages and disadvantages of Laravel Framework for web Development. DDI development.

<https://ddi-dev.com/blog/programming/pros-and-cons-of-laravel-framework-for-web-app-development/>

Digamber. (2021, februari 16). Top 10 Best IDE for Software Development in 2021. Positronx.io.

<https://www.positronx.io/top-10-best-ide-for-software-development/>

Django. (n.d.). Django. Django.

<https://www.djangoproject.com/>

Fatima, W. (2020, september 9). Sublime Vs Atom Vs Brackets. Developer Resources.

https://ssiddique.info/sublime-vs-atom-vs-bracket.html#Which_text_editor_is_better_Atom_or_Sublime_text_or_Brackets

Intagleo. (2020, oktober 15). The Most Popular Backend Frameworks for Web Development in 2020. Intagleo Systems.

<https://www.intagleo.com/blog/most-popular-backend-frameworks-for-web-development-in-2019/>

Laravel. (n.d.). Meet Laravel. Laravel.

<https://laravel.com/docs/8.x>

PageSpeed Insights. (n.d.). Over PageSpeed inzichten. PageSpeed Insights.

<https://developers.google.com/speed/pagespeed/insights/?hl=nl>

PhpStorm Reviews. (2020/2021). Capterra.

<https://www.capterra.com/p/186624/PhpStorm/reviews/>

Poczwadowski, M. (2017, augustus 2). Ruby on Rails: Pros & Cons. What You Should Know Before Choosing The Technology. Netguru.

<https://www.netguru.com/blog/pros-cons-ruby-on-rails>

Rak, V. (2021, februari 5). Pros and Cons of Ruby on Rails. Sloboda Studio.

<https://sloboda-studio.com/blog/pros-and-cons-of-ruby-on-rails/>

React. (n.d.). Tutorial: Intro to React. React.

<https://reactjs.org/tutorial/tutorial.html#what-is-react>

Ruby on Rails. (2021, februari 17). Welcome to Rails. Ruby on Rails.

<https://api.rubyonrails.org/>

Singh, V. (2021, februari 23). Best Web Development IDE. Hackr.io.

<https://hackr.io/blog/web-development-ide>

Software Testing Help. (2021, februari 18). Visual Studio Code Vs Atom: Which Code Editor Is Better. Software Testing Help.

<https://www.softwaretestinghelp.com/visual-studio-code-vs-atom/#:~:text=Visual%20Studio%20Code%20has%20a%20greater%20number%20of%20built%20in,easy%20to%20add%20and%20configure.>

Stackshare. (n.d.). Visual Studio Code. Stackshare.

<https://stackshare.io/visual-studio-code>

State Of JS. (2020). Front-end Frameworks. State Of JS 2020.

<https://2020.stateofjs.com/en-US/technologies/front-end-frameworks/>

State Of JS. (2020). Text Editors. State Of JS 2020.

https://2020.stateofjs.com/en-US/other-tools/text_editors

Sublime Text Reviews. (2020/2021). Capterra.

<https://www.capterra.com/p/179535/Sublime-Text/reviews/#:~:text=Pros%3A%20The%20Sublime%20Text%20is,of%20the%20popular%20coding%20languages.>

Toolkit. (n.d.). Coda 2. Toolkit.

<https://hackdesign.org/toolkit/coda-2>

Vue.js. (n.d.). Introduction. Vue.js.

<https://v3.vuejs.org/guide/introduction.html#what-is-vue-js>

Wikipedia. (2020, december 7). JSFiddle. Wikipedia.

<https://en.wikipedia.org/wiki/JSFiddle>



Onderzoek tools & middelen

Wesley Vermeulen - ICT & Media Design