26-1-2018

Adviesrapport

GourmetVR

Han Ting Wu & Wouter Bosch

S1103338 & S1100261

# Managementsamenvatting

Virtual reality (VR) is een omgeving die met behulp van een computer wordt gesimuleerd om een gebruiker via diverse zintuigen mee te nemen in een ervaring (Wikipedia, 2017). De moderne VR die we kennen is gebaseerd op een techniek uit 1838, hier was het principe nog zeer eenvoudig en kon het nog lang niet alles wat je nu met VR kan. Sindsdien zijn er meerdere pogingen gedaan om VR goed te realiseren, dit is pas gelukt rond 2013 waar de eerste VR brillen en headsets verschenen (VRwebwinkel, 2018). In de laatste paar jaren ontwikkelt VR zich ontzettend snel, en zijn er ook meerder frameworks voor het bouwen van een VR omgeving. Omdat VR nog vrij nieuw is, zijn de frameworks ook nieuw en zijn nog veel in ontwikkeling. Om goed en efficiënt een VR omgeving te kunnen maken is het noodzakelijk om deze verschillende frameworks te testen.

Het doel van dit onderzoek is om te achterhalen welk framework beter is voor het bouwen van een VR omgeving. Aan de hand hiervan is de volgende hoofdvraag opgesteld: *Welk framework is beter voor het creëren van een VR omgeving, A-Frame of React VR?*

Om een antwoord te kunnen geven op de hoofdvraag is een experiment uitgevoerd waarin een web-based virtuele experience wordt gemaakt, waarin VR gebruikt wordt, voor een mobiel apparaat voorzien van een VR bril of headset. Hiervoor maken we met beide frameworks een VR beleving, die we vervolgens met elkaar vergelijken. Uit het resultaat blijkt dat Aframe een beter framework is voor het creëren van een VR omgeving.

Op basis hiervan wordt aanbevolen dkkdjdkljldjla

Onderzoek: welke framework is beter om VR in te maken // VR omgeving is beter

Inhoudsopgave

[Inleiding 4](#_Toc504760076)

[Het onderzoek 5](#_Toc504760077)

[A-frame 5](#_Toc504760078)

[ReactVR 7](#_Toc504760079)

[Oplossingen 9](#_Toc504760080)

[Advies en conclusie 9](#_Toc504760081)

[Literatuurlijst 10](#_Toc504760082)

# Inleiding

Dit adviesrapport is opgesteld voor betrokkenen en geïnteresseerden in webVR, hier wordt aan de hand van voorbeelden en meningen een idee geschetst van de gebruikte frameworks. Met gebruik van beeldmateriaal en de uitleg van de genomen stappen zullen de meningen dan ook onderbouwd worden. Naderhand van het verslag trekken wij een slotadvies en conclusie over het geheel. Hier zal vervolgens uit komen welke volgens ons het beste uit de test komt. Buiten het advies zal er ook worden uitgezoomd op webVR in het algemeen en zal er wederom aan de hand van voorbeelden een illustratie gegeven worden over wat webVR kan betekenen voor de mensen en waar de grenzen juist liggen, eventuele toekomstige toepassingen en ook waar de knelpunten liggen. Naderhand zal er dus antwoord worden gegeven op de vraag die als rode draad door dit rapport loopt, ‘’Welk framework is beter voor het creëren van een VR omgeving, A-Frame of React VR’’.

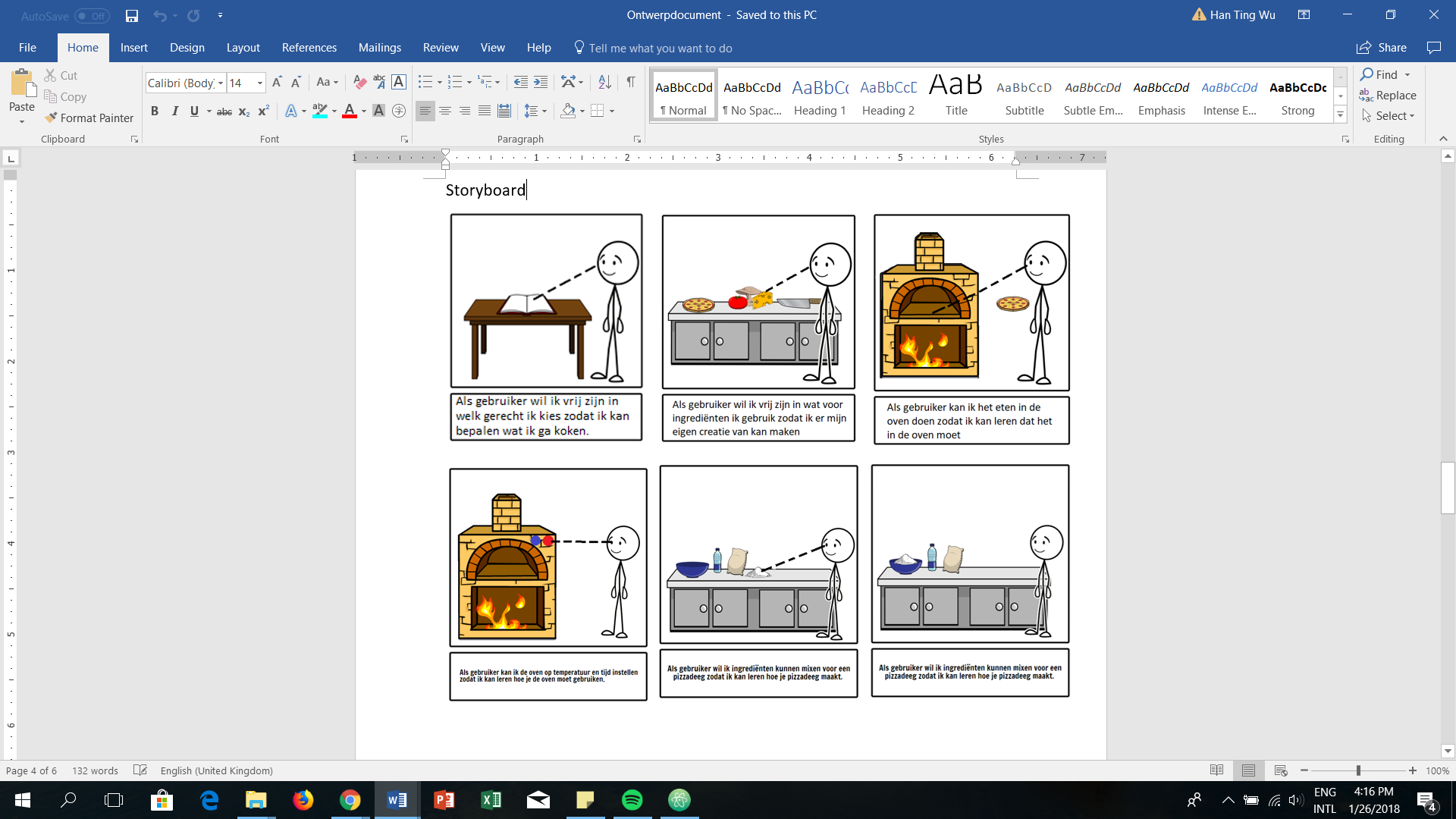
In het rapport kan u tevens de volgende elementen verwachten. Een onderzoek. Het onderzoek zal de inhoudelijke informatie bevatten over het gebruik en de toepassingen van de individuele frameworks. Deze informatie zal, zoals eerder aangegeven, beeldmateriaal en voorbeelden bevatten. Verder zal het verslag een conclusie bevatten die nog terugkijkt op beide frameworks en deze naast elkaar zal leggen. Tot slot zal er een advies worden gegeven en hiermee dus ook het antwoord op de hoofdvraag. Na dit verslag is dan ook het uitgangspunt dat de gebruiker een goed idee heeft gekregen van webVR in het algemeen, de toepassingen hiervan en eventueel een voorkeur voor een framework met bijbehorende redenen.

Het onderzoek **>> onderzoek opzet & resultaten >> waar heb je naar gekeken? (vormgeving, interactie, opbouw van framework) >> hoe is dit aangepakt?**

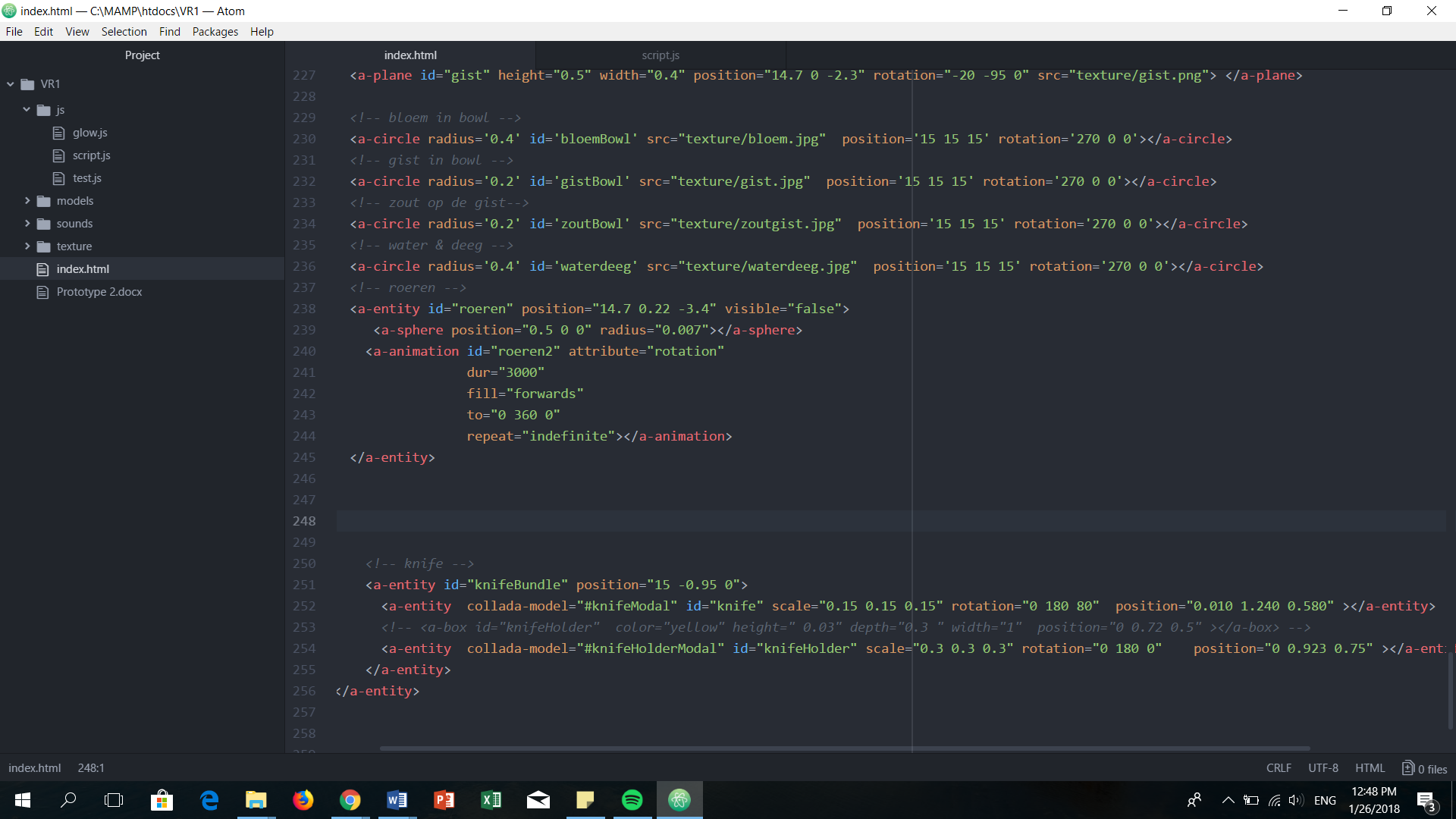
Voor het onderzoek zijn beide frameworks gebruikt om een VR beleving te maken. Voor het resultaat zijn beide ervaringen met elkaar vergeleken. Hierbij is gekeken naar, eigen ervaring met het framework, de toegankelijkheid, het gebruiksgemak, de interactie en het eindresultaat.

## A-frame

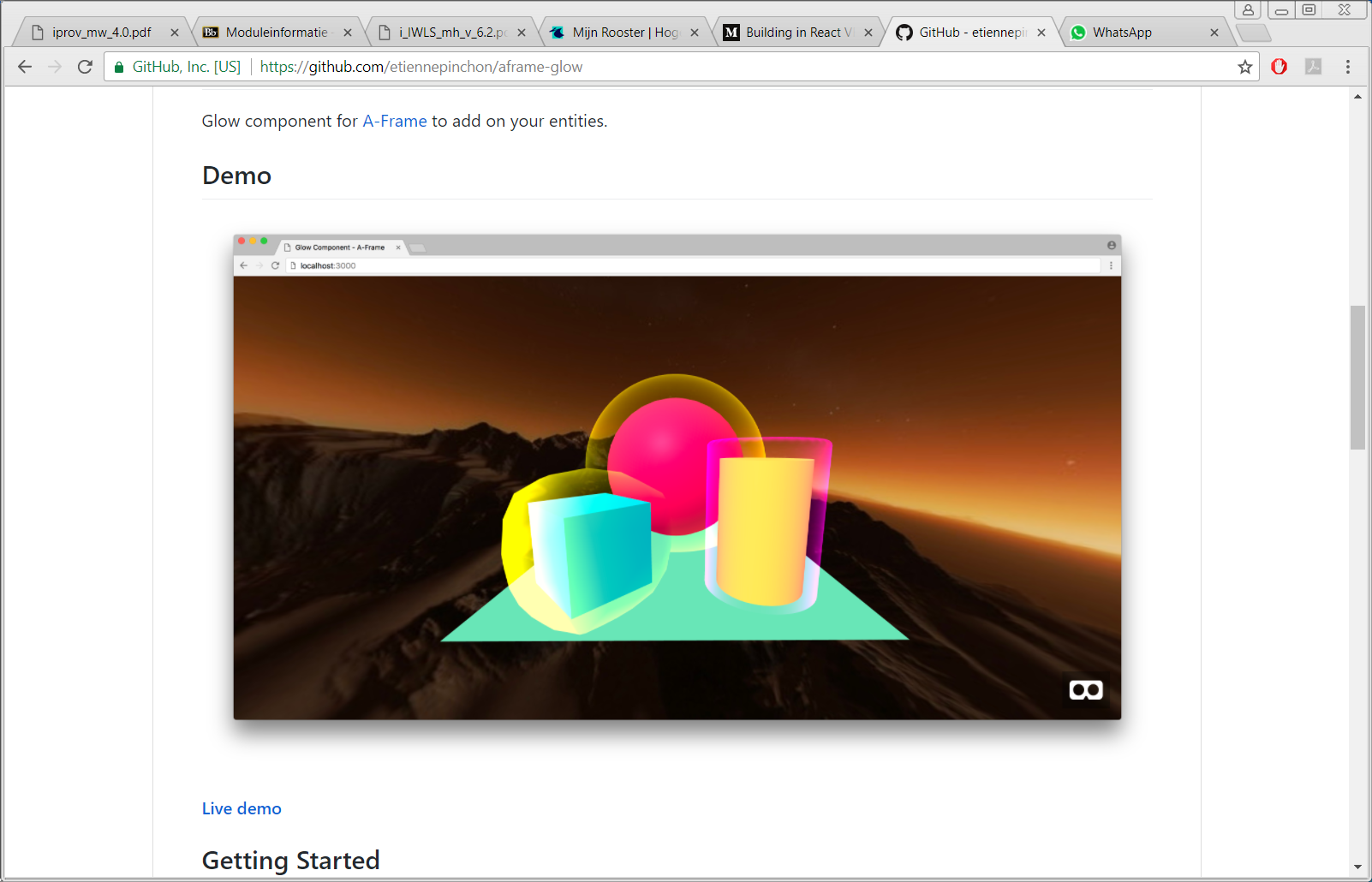
Het eerste framework waarmee we aan de slag zijn gegaan is A-Frame. Voor deze beleving hebben we eerst user stories en storyboards gemaakt.

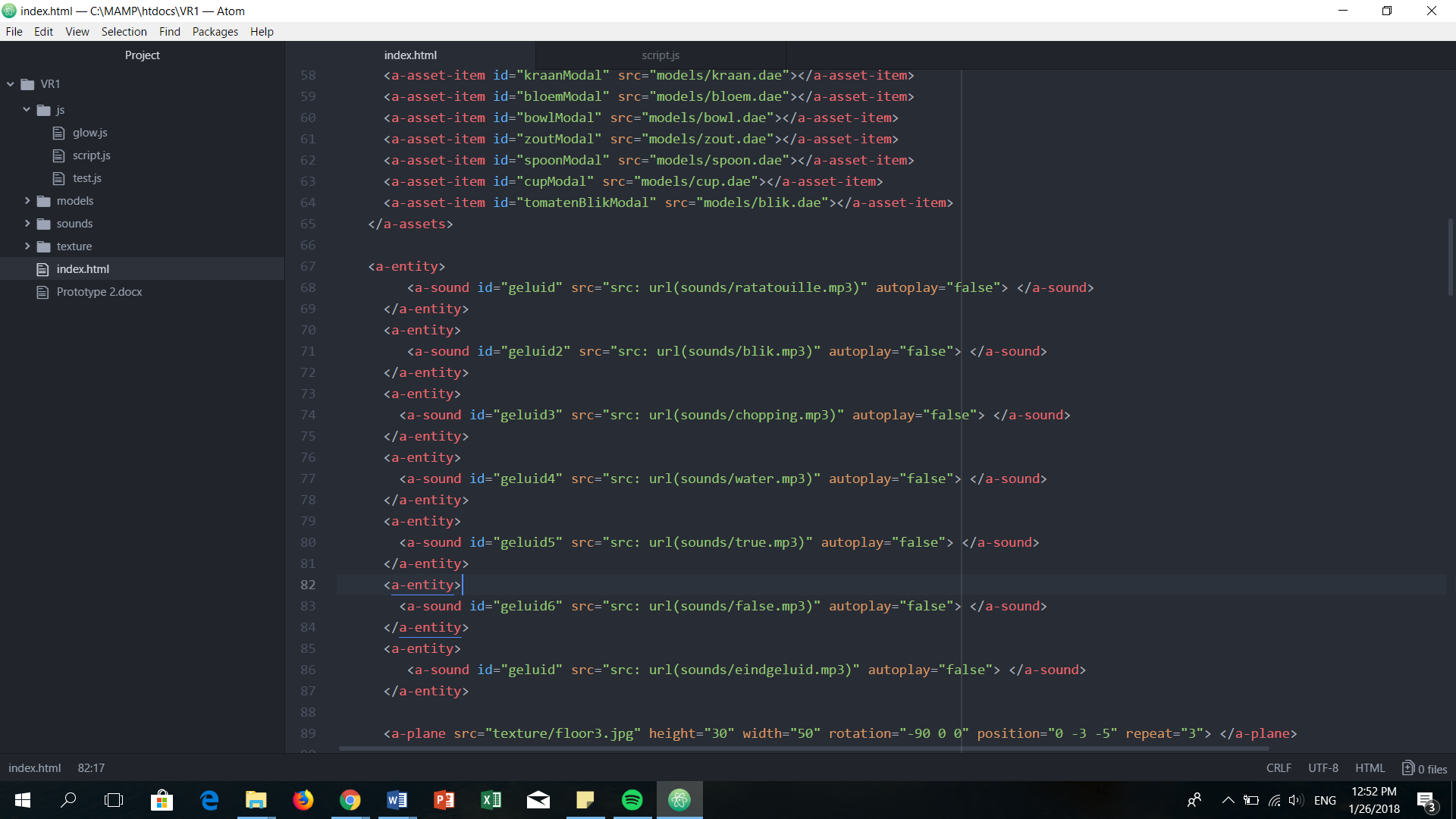


Vervolgens zijn we gaan experimenteren met het framework. Zo hebben we geëxperimenteerd met animaties. Over het algemeen zijn animaties binnen A-Frame gemakkelijk te gebruiken. Echter, het heeft zo zijn beperkingen. Zo is het goed positioneren van de begin en eindpositie soms best nog wel een klus en erg nauwkeurig werk. Verder hadden we ook last van een bug, waardoor animaties onbedoeld opnieuw werden afgespeeld. Dit kwam doordat er een nieuw object in de JavaScript werd gecreëerd die nog niet in de HTML stond. En dit triggerde op de een of andere manier de animatie waardoor die zich herhaald. **PLAATJE??**



Ook hebben we geëxperimenteerd met 3D modellen inladen. We hebben zowel zelfgemaakte 3D modellen, gemaakt met Blender, als modellen van internet ingeladen. Alle Blender modellen waren makkelijk in te laden, wel hadden we problemen met texturen die niet meegingen. Modellen van internet waren ook gemakkelijk in te laden, alleen waren ze niet altijd zichtbaar. Verder waren de texturen hiervan ook niet zichtbaar.

Met A-Frame hebben we ook geluid en belichting uitgetest. Voor geluid kan je zeer gemakkelijk mp3 bestanden gebruiken. Deze kan je gelijk afspelen of pas aanzetten bij bepaalde interacties, ook kan je ze op loop zetten en het volume aanpassen. Voor belichting hebben we een component gebruikt dat glow heet, hiermee kan je objecten laten oplichten. Deze kan je ook gemakkelijk aan en uit zetten.



Verder hebben we ook nog geëxperimenteerd met geometrie, hiermee kan je de vorm van entities aanpassen. Geometrie is vrij gemakkelijk toe te passen. In plaats van dat je de basisvormen sphere, cylinder, box etc., heb je veel meer mogelijkheden. Het fijne van geometrie is dat je geheel vrij bent in het creëren van een vorm naar wens, de vrijheid is tevens ook het nadeel. Doordat je zo vrij bent zijn er zoveel mogelijkheden waardoor het snel overweldigend wordt.

Ook hebben we nog geëxperimenteerd met API’s. Naar onze mening zijn API’s wel te gebruiken met A-Frame, zolang je een geschikte API hebt die werkt. Tijdens het experimenteren hiervan zijn er veel API’s geweest waarbij we een Cross-Origin Header Error (CORS) kregen. Deze vonden wij moeilijk om op te lossen, omdat er weinig documentatie voor is hoe dat precies moet. Ook heeft elke API en andere onderliggende oorzaak, waardoor elke oplossing anders is. Er is niet één oplossing voor deze error, dat maakt het erg lastig.

Tenslotte hebben we nog geëxperimenteerd met event emitters. Event emitters zijn functies die op een bepaald object een event afgeven waarnaar geluisterd kan worden, waar volgens weer als reactie een stuk code op uitgevoerd kan worden. Dit wordt vooral gebruikt bij interacties en animaties. Event emitters zijn vrij gemakkelijk te gebruiken en we zijn hiermee niet tegen problemen aangelopen.

Naast deze specifieke punten binnen A-Frame zijn er nog algemene punten die ons zijn opgevallen. Zo merkten wij dat het lokaal via MAMP/XAMP en hosten via firebase nog kan verschillen van elkaar. Zo werkte de omgeving bijvoorbeeld prima lokaal, maar op firebase waren sommige objecten niet zichtbaar. Dit kwam uiteindelijk doordat firebase hoofdletter gevoelig is en lokaal niet.

## ReactVR

ReactVR is wat wij als tweede framework hebben gebruikt. We zijn begonnen met kijken naar de documentatie van ReactVR in de hoop dat we het konden begrijpen. Helaas hadden we daar moeite mee dus toen zijn we een tutorial gaan opzoeken en hebben we daar onze informatie vandaan gehaald. Naast de tutorial zijn we zelf nog gaan experimenteren om te kijken wat we konden maken. Waar we uiteindelijk geëxperimenteerd mee hebben waren: animaties, panos, views, interacties, componenten en stylesheet.

Animaties:

Helaas zijn de animatie mogelijkheden in ReactVR niet heel uitgebreid. Op dit moment is het alleen mogelijk om View, Text en Image te animeren met Animated.View, Animated.Text, Animated.Image. Dit betekend dat je dus eigenlijk alleen containers, tekst en plaatjes kan animeren. Volgens de documentatie kan je zelf een animation maken met createAnimatedComponent, maar daar is verder geen tot weinig onduidelijke documentatie over. Animaties zijn op zichzelf niet heel moeilijk om te begrijpen als je eenmaal de syntax van ReactVR begrijpt maar als je dat nog niet snapt dan zijn het wel lastige obstakels om te overkomen.

Pano en views:

Pano image is een panorama foto die in een bol van 1000 meter om de gebruiker heen wordt gezet. Er zijn twee soorten: equirectangular panos en cubic panos. Equirectangular panos bestaat uit een foto met een aspect ratio van 2:1. De foto’s hiervoor zijn ook gemaakt met een speciale 360 camera. Deze foto’s worden dan in een bol vorm om de gebruiker heen gezet. Cubic panos bestaan uit 6 foto’s die in een kubus vorm om je heen gezet worden. Pano image is een van de makkelijker elementen van ReactVR. In ReactVR mag je maar 1 component in je render hebben maar vaak wil je meer dan 1 component om dit op te lossen heb je views. Views zijn eigenlijk gewoon containers waarin je dingen samen groepeert. Views zijn heel eenvoudig en de syntax is eigenlijk niet echt anders dan een simpele div tag van html.

Interacties

Bij ReactVR heb kan je in een component een onClick gebruiken en dan daarna een functie aan verbinden. In de kleine applicatie die ik had gemaakt werd een functie aangeroepen die plus 10 of min 10 bij een applicatie deed. De syntax is moeilijk om te bedenken en te schrijven als je geen ervaring er mee hebt. Maar als je het 1 keer gezien heb kan je het wel begrijpen om dat het best logisch gestructureerd is.

Components

Components zijn herbruikbare gebruikersinterface elementen die gebruikt kunnen worden als tags. Er zijn een aantal standaard ingebouwde componenten zoals Text en Image. Maar het is ook mogelijk om zelf een component te maken met React.Component. Elk gebruikers component heeft een render die child-components terug geeft. De standaard componenten zijn niet heel erg ingewikkeld omdat er duidelijke documentatie over is en het logisch gestructureerd is.

Stylesheet

Stylesheets wordt gebruikt om bijvoorbeeld rotatie positie en schaal mee te geven. Je kan ook al van te voren een stylesheet aanmaken en daar een naam aan geven. Als je dan die naam aan elementen geeft, dan hebben die elementen allemaal de stijl die bij de naam hoort. Het doet wel een beetje denken aan de manier hoe CSS werkt. Doordat het lijkt op de CSS structuur is het redelijk makkelijk te begrijpen en te maken.

Na deze specifieke punten binnen ReactVR zijn er nog een aantal algemene punten die opvallen. Zo is de documentatie niet heel erg uitgebreid en is het moeilijk om op het internet oplossingen voor je bugs te vinden omdat het een erg nieuwe techniek is. Daarnaast gebruikt ReactVR JSX syntax en dat maakt erg moeilijk om te begrijpen als je geen ervaring al hebt met JSX. Verder omdat het een 3d omgeving is het moeilijk om te begrijpen wat er ruimtelijk aan de hand is met je objecten voor dat je het ervaren hebt in de VR omgeving. Als laatste hoef je bij ReactVR niet nog aparte hosting te vinden voor het ontwikkelen van de omgeving omdat het deel is van de setup van ReactVR.

# Oplossingen

Zowel het gebruik van A-Frame als van ReactVR heeft zo zijn voor- en nadelen. Hieronder alle voor- en nadelen van beide frameworks op een rijtje.

A-Frame is over het algemeen vrij gemakkelijk te gebruiken. Er is genoeg documentatie om een basis VR beleving mee te maken. Voor meer geavanceerdere belevingen is de documentatie erg schaars. Ook is A-Frame vrij gemakkelijk om in te komen en te leren. Zo is de syntax erg begrijpelijk en kan je snel op weg. Je kan met de basiscode al veel objecten in je beleving plaatsen en ermee interacteren. NOG MEER VOORDELEN

ReactVR is een vrij nieuw framework, nieuwer dan A-Frame, waardoor de documentatie ook zeer schaars is. Daarnaast is het ook moeilijk om naast de documentatie iets erover te vinden omdat het zo nieuw is. Verder is ReactVR erg lastig om te leren als je geen ervaring hebt met JSX. En als laatste moet je componenten maken om objecten in je beleving te laten zien. Een van de voordelen van ReactVR is dat ze een functie *‘cubic panos’ hebben*, deze functie zorgt ervoor dat het een hoop minder scheelt veel werk. Een ander voordeel is dat ReactVR een erg hoge performance heeft.

VOORDEEL REACTVR!!??

**AFRAM: Makkerlijke te leren, syntax begrijpelijker, geen componentne importeren (om bv iets te laten zien)**

**REactVR voordelen: sky-box 2 manier >> cubic panos >> 6 plaatjes worden gelijk goed neergezet**

# Advies

In dit onderzoek is gezocht naar de vraag: “Welk framework is beter voor het creëren van een VR omgeving, A-Frame of React VR?” Hiervoor zijn de ervaringen en resultaten van twee frameworks met elkaar vergeleken.

Uit deze ervaringen en resultaten raden wij A-Frame aan. De belangrijkste reden waarom wij A-Frame aanraden is omdat A-Frame makkelijker te leren is. Dit is mede doordat het geen gebruik maakt van JSX, maar ook omdat de syntax veel lijkt op HTML. Waardoor mensen met basis kennis van HTML en Javascript minder moeite zullen hebben om het te begrijpen. Daarnaast is A-Frame uitgebreider dan ReactVR en is er meer mogelijk. Zo kan je in A-Frame aan alles een animatie geven. Bij ReactVR zijn daar minder mogelijkheden voor. A-Frame heeft ook wat nadelen zoals: prestatieproblemen en het mist functionaliteit op sommige plekken. Ondanks deze nadelen raden wij toch A-Frame aan omdat als je kijkt naar de voordelen en nadelen, het meer voordelen heeft dan ReactVR.

Verder is het erg voordelig dat je geen componenten hoeft te importeren, je kan zo

**uitgebreider dan reactVR. ARAME kan minder handelen/aan >> meer last van performance issues**

**Aframe geen cubic panos >> zelf positioneren >> 6 plaatjes**

# Literatuurlijst

Wikipedia. (2017). *Virtuele werkelijkheid.*  
Geraadpleegd op 25 januari 2018 van,   
<https://nl.wikipedia.org/wiki/Virtuele_werkelijkheid>

VRwebwinkel. (2018). *Geschiedenis van Virtual Reality.*   
Geraadpleegd op 25 januari 2018 van,  
<https://vrwebwinkel.nl/geschiedenis-virtual-reality/>