

3D Sound Visualization in Silverlight

Visualizing sound more effeciently



Andreas Renkema

11 juni 2013

3D Sound Visualization in Silverlight

Visualizing sound more effeciently

Gegevens student:

Naam	Renkema, A.C.
Studentnummer	2142037
Afstudeerrichting	ICT & Lifestyle (Voltijd)
Afstudeerperiode	van 4 februari 2013 t/m 30 juni 2013 (105 werkdagen)

Gegevens bedrijf:

Naam	Sorama B.V.
Afdeling	Software development
Plaats	Eindhoven
Bedrijfsbegeleider	Jonge, M. de (<i>Chief Software Development</i>)

Gegevens docentbegeleider:

Naam	Graaf, L. de
------	--------------

Gegevens verslag:

Titel	3D Sound Visualization in Silverlight
Datum van uitgifte	11 juni 2013
Getekend voor gezien door bedrijfs- begeleider:	

Datum:

De bedrijfsbegeleider,

Andreas Renkema

11 juni 2013

Documenthistorie

Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	24/4/2013	Opzet in L ^A T _E X
0.2	Concept	21/5/2013	Meer inhoud en feedback verwerkt
0.3	Concept	28/5/2013	Eerste volledige versie
0.4	Concept	6/6/2013	Feedback verwerkt
1.0	Definitief	10/6/2013	Laatste feedback verwerkt
1.1	Definitief	11/6/2013	Spelling- en grammaticacontrole

Goedkeuring

Versie	Datum goedkeuring	Naam	Functie	Paraaf

Distributie

Versie	Datum verzending	Naam	Functie
0.1	8/5/2013	Lennart de Graaf	Docentbegeleider
0.3	28/5/2013	Lennart de Graaf	Docentbegeleider
0.3	28/5/2013	Merijn de Jonge	Bedrijfsbegeleider
0.4	6/6/2013	Lennart de Graaf	Docentbegeleider

Voorwoord

Dit verslag beschrijft mijn afstudeerstage bij Sorama B.V. te Eindhoven, welke ik heb gelopen van 4/2/2013 tot en met 30/6/2013 . Deze stage heb ik gedaan in het kader van ICT & Lifestyle, een studieroute die zich met name richt op het ontwikkelen van innovatieve systemen.

In september vorig jaar ben ik begonnen naar het zoeken van een geschikte afstudeerstage. Omdat ik er zeker van wilde zijn dat ik een stageplaats zou kunnen krijgen waar ik zoveel mogelijk van kan leren, ben ik vroeg begonnen met mijn zoektocht. Zo selecteerde ik een aantal opdrachten die zich op een lijn bevinden met mijn ambities en mijn interesse opwekken. Na een gesprek bij Sorama was ik er van overtuigd dat dit de geschikte plek voor mij was om af te studeren.

Opvallend aan het bedrijf is dat het gehuisvest is in de kelder van de Technische Universiteit in Eindhoven. Door de innige samenwerking tussen Sorama en de TU zijn er dan ook veel afstudeerders van de universiteit aanwezig. Dit werkt erg prettig omdat je deel uitmaakt van een betrokken groep collega's in plaats van dat je de stagiaire bent die in een afgezonderd hokje zijn eigen werk zit te doen.

Ten slotte zijn er een aantal mensen die ik wil bedanken die mij hebben geholpen. Zo wil ik Merijn de Jonge bedanken dat hij mij in de gelegenheid stelde om de opdracht uit te voeren. Niet alleen dat, ook zijn goede begeleiding heeft bijgedragen tot het succes van deze stage. Ook wil ik Lennart de Graaf bedanken voor zijn feedback op het verslag en de fijne begeleiding. Als laatste wil ik Christine van Woensel bedanken. Zij heeft het verslag gecontroleerd op taal- en spelfouten totdat ze hoofdpijn kreeg zodat andere lezers dat niet hoeven te krijgen.

Inhoudsopgave

Samenvatting	7
Summary	8
Verklarende Woordenlijst	9
1 Inleiding	11
1.1 Opbouw van het verslag	12
2 Bedrijfsbeschrijving	13
2.1 Missie & Visie	13
2.2 Organigram	14
3 Probleemstelling & Doelstelling	15
3.1 Probleemstelling	15
3.2 Deelvragen op de probleemstelling	16
3.3 Opdrachtomschrijving	16
3.4 Doelstelling van het onderzoek	16
4 Onderzoek	17
4.1 Inleiding	17
4.2 Soortgelijke visualisaties	18
4.3 Toegankelijkheid	19
4.4 Besturing van visualisatie	21
4.4.1 Rotatie van het object	21
4.4.2 Afspelen animatie	22
4.4.3 Conclusies	23
4.5 Alternatieve frameworks	23
4.5.1 Alternatieven	23
4.5.2 Keuze alternatief	24
4.6 Conclusies	25
5 Realisatie & implementatie	26
5.1 Inleiding	26
5.2 Eerste prototype	26
5.2.1 Doel	26
5.2.2 Realisatie	27
5.2.3 Resultaat	28
5.2.4 Reactie van gebruikers	28
5.2.5 Evaluatie	29
5.3 Tweede prototype	29
5.3.1 Doel	29
5.3.2 Realisatie	29
5.3.3 Resultaat	31
5.3.4 Reactie van gebruikers	31
5.3.5 Evaluatie	32
5.4 Conclusies & aanbevelingen	32

5.5 Implementatie basis	32
6 Reflectie	33
6.1 Reflectie op onderzoek	33
6.2 Aanbevelingen	33
Literatuurlijst	34
Nawoord	36
Bijlagen	37
I Project Initiatie Document	38
II Geslacht van klanten	58
III Framework keuze	59
IV openFrameworks performance	62

Samenvatting

Deze stage is uitgevoerd bij Sorama B.V. van 4/2/2013 tot en met 30/6/2013 door Andreas Renkema. Sorama is opgericht door Rick Scholte en specialiseert zich in het opsporen en verminderen van geluidsbronnen. Dit wordt gedaan door middel van de door Sorama ontwikkelde geluidscamera en portal. De portal ontvangt data van de geluidscamera en genereert de tweedimensionale beelden, welke vervolgens worden weergegeven. Er is echter data beschikbaar in drie dimensies. Om voor te blijven op de concurrentie is er besloten om te kijken naar wat de verschillende mogelijkheden zijn op het gebied van 3D. Hiervoor wordt er gekeken naar Silverlight en hoe de data hierin kan worden gevisualiseerd.

De standaard oplossing om 3D beelden te genereren in Silverlight is XNA. Tijdens de stage werd het bekend dat dit framework volgend jaar geen support meer gaat krijgen vanuit Microsoft, de ontwikkelaar daarvan. Er moest dus worden gezocht naar een alternatief voor XNA als deze niet meer te gebruiken is. Dit is uiteindelijk WebGL geworden, een techniek die nog in de kinderschoenen staat. Gezien de lengte van het project en de wisselende ondersteuning van WebGL tussen browsers is er besloten om voor de duur van de stage XNA te blijven gebruiken.

Verder is er onderzoek verricht naar hoe het beste de visualisatie kan worden bestuurd zodat nieuwe gebruikers geen probleem hebben met het nieuwe systeem. Ook is er een alternatief kleurenschema onderzocht en toegepast voor mensen met een vorm van kleurenblindheid zodat zij geen last ondervinden van het huidige regenboogachtige schema.

Met deze gegevens zijn er twee visualisatieprototypes gemaakt in XNA. Naar aanleiding van de reactie van gebruikers is de tweede geselecteerd om te implementeren. Het resultaat is een visualisatie die gebruik maakt van 3D in Silverlight met driedimensionale data. Deze geeft beter weer hoe geluid zich voortplant in de ruimte dan de huidige visualisaties.

Summary

This internship has been done by Andreas Renkema at Sorama B.V. from February 4th 2013 until June 30th 2013. Sorama is founded by Rick Scholte and specializes in localizing and reducing sources of sound. This is done with the sound camera and portal, developed by Sorama. The portal uses two dimensional imagery to visualize sound. However, the used data includes three dimensions. In order to stay ahead of the competition, the intern will research what the possibilities are concerning 3D in Silverlight and how the data can be visualized.

The go-to solution to generate 3D images in Silverlight is XNA. During the internship it became known that Microsoft wouldn't support the framework anymore next year. The solution to this problem was to search for an alternative technique which can be used when XNA is deprecated. After heaps of research this was proven to be WebGL, a system that's still in its infancy. With the duration of the project in mind and the fact that WebGL isn't fully supported by all browsers, it has been decided to use XNA until the end of the project.

Furthermore it has been researched how the visualization can be optimally controlled, so new users of the system don't have any trouble. A new color scheme has been developed as alternative to the current rainbow-like scheme, to aid users with a type of colorblindness.

There have been two new prototype developed in XNA. As a result of the interviewing of users, the second prototype has been chosen to be implemented. This results in a visualization that uses 3D in Silverlight with three-dimensional data. It reflects how the measured sounds propagate in space, something that wasn't available until now.

Verklarende woordenlijst

ASP (Active Server Pages)

Engine voor dynamisch gegenereerde webpagina's, ontwikkeld door Microsoft.

.NET

Een framework ontwikkeld door Microsoft dat voornamelijk wordt uitgevoerd op Microsoft Windows. Het bevat onder andere een grote bibliotheek en zorgt voor interoperabiliteit tussen talen (elke taal kan code gebruiken dat geschreven is in een andere taal).

3D (Three dimensional)

Een begrip dat aanduidt dat iets drie dimensies heeft zoals in de fysieke wereld waarin mensen leven.

C++

Een statisch getypeerde, gecompileerde programmeertaal.

C#

Een programmeertaal gebaseerd op C++ en Java, ontwikkeld door Microsoft.

CSV (Comma Seperated Values)

Een bestandsformaat welke tabellarische data opslaat in platte tekst. Voor het onderscheiden van waarden worden, zoals de naam suggereert, komma's gebruikt.

Desktopapplicatie

Een applicatie die rechtstreeks door het besturingssysteem wordt uitgevoerd.

FPS (Frames per second)

Snelheid uitgedrukt in beelden per seconde.

Framework

Een abstractie van software met algemene functionaliteit die kan worden toegepast om een applicatie uit te breiden.

GIMP (GNU Image Manipulation Program)

Een bitmap editor vergelijkbaar met Adobe Photoshop

GPU (Graphics Processing Unit)

Een deel van het systeem dat wordt gebruikt om te renderen wat er op een scherm moet staan.

Microfoonarray

Een raster van microfoons

NAH (Nearfield Acoustic Holography)

Een methode gebruikt om geluidsvelden van een bron te berekenen door het meten van akoestische parameters op een afstand van de bron.

OpenGL (Open Graphics Library)

Een API voor het renderen van twee- en driedimensionale beelden.

Open Source

Het ontwikkelen en uitgeven van eindproducten en de bron daarvan met de intentie dat het door iedereen is aan te passen.

Portal

De webapplicatie van Sorama die toegang verschaft tot het maken en interpreteren van metingen.

Silverlight

Een softwareplatform voor het ontwikkelen van webapplicaties.

WebGL (Web Graphics Library)

Een open standaard voor het specificeren van 3D beelden op het web door middel van OpenGL.

WPF (Windows Presentation Foundation)

Het grafische subsysteem voor het renderen van interfaces voor Windows-applicaties.

XNA (XNA's Not Acronymed)

Een framework gemaakt door Microsoft dat er op gericht is game-ontwikkeling te versnellen.

1 Inleiding

Tegenwoordig zijn er allerlei eisen waar bedrijven aan moeten voldoen om hun producten op de markt te mogen brengen. Dit is niet alleen voor de gebruiker van belang maar ook voor de omgeving waarin het product wordt gebruikt. Zo moeten auto's aan bepaalde normen voldoen wat betreft geluidsemissie om geen overlast te veroorzaken in druk bewoonde gebieden zoals steden of wijken vlakbij snelwegen.[1]

Het is ook mogelijk dat een bedrijf zijn product in akoestisch opzicht wil verbeteren. Een ventilatorfabrikant kan bijvoorbeeld als doel hebben om zijn ventilators zo stil mogelijk te maken.[2] Zij doen dit door bijvoorbeeld kleine vinnen op de ventilatorbladen te plaatsen die de geluidsgolven opbreken waardoor het geluid sneller uitdooft.[3] Om dit soort aanpassingen te valideren is een simpele geluidsniveaumeting niet voldoende, het is dan beter om een precies beeld hebben hoe het geluid ontstaat en de ruimte in beweegt.

Sorama heeft een geluidscamera ontwikkeld die hierbij kan helpen. Deze camera laat zien waar de bronnen van trilling en geluid zich bevinden en hoe deze zich gedragen. Hiermee geeft het bedrijf advies aan klanten hoe ze hun product kunnen verbeteren maar klanten kunnen zelf ook de camera gebruiken om metingen uit te voeren.

In dit verslag wordt er gekeken naar de mogelijkheden met 3D wat betreft geluidvisualisaties. De toegevoegde waarde hiervan is dat een gebruiker niet alleen kan begrijpen hoe het geluid ontstaat maar ook hoe deze zich voortplant in de ruimte.



Figuur 1: Lokalisatie van geluidsbronnen met de geluidscamera

1.1 Opbouw van het verslag

In dit verslag zal het verloop van de stage en het resultaat daarvan worden beschreven. Beginnende in hoofdstuk 2 wordt het stagebedrijf beschreven met daaropvolgend de probleem- en doelstelling in hoofdstuk 3. Vervolgens beschrijft hoofdstuk 4 het uitgevoerde onderzoek met de bijbehorende resultaten. Deze resultaten worden gebruikt om drie prototypes van een driedimensionale geluidsvisualisatie te ontwikkelen wat wordt gedocumenteerd in hoofdstuk 5. In dit hoofdstuk wordt ook beargumenteerd welk prototype is gekozen om te implementeren en hoe dit is gebeurd. Als laatste zal in hoofdstuk 6 worden gereflecteerd op het verloop van het project en zullen er aanbevelingen worden gedaan met betrekking tot de toekomst van het project.

2 Bedrijfsbeschrijving

Sorاما BV is een klein bedrijf bestaande uit ongeveer vijftien werknemers. Het is in 2009 opgericht vanuit het promotieonderzoek van Rick Scholte[4] en sindsdien gehuisvest bij de Technische Universiteit Eindhoven (TU/e). Het onderzoek staat in teken van near-field acoustic holography (NAH), het in kaart brengen van geluid binnen het nabijheidsveld. Hiermee helpt Sorاما andere bedrijven om geluidsbronnen te lokaliseren en te elimineren om aan wettelijke voorschriften te voldoen. Een voorbeeld hiervan is een autoruit die resoneert op een bepaalde frequentie en daardoor geluid lekt zoals in figuur 2.



Figuur 2: Een VW Caddy die gemeten wordt

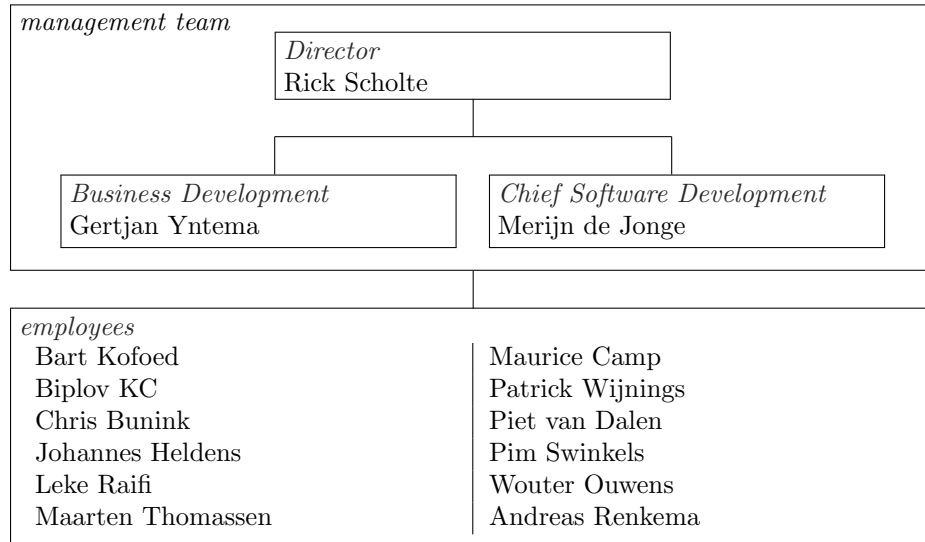
Dit doet het bedrijf door middel van zijn geluidscamera, een array van microfoons waarmee het geluid wordt gemeten. De data die hier uit komt wordt verwerkt en doorgestuurd naar de computer. Op de computer draait de portal, een webapplicatie om metingen met de camera uit te voeren en naderhand te bekijken. Dit systeem is ook te koop zodat klanten zelf metingen uit kunnen voeren. Er worden echter niet alleen camera's verkocht en metingen uitgevoerd, Sorاما geeft ook advies aan klanten en lost zelf geluidsproblemen op.

2.1 Missie & Visie

Missie: “Sorاما neemt voor om het beste bedrijf in de wereld te zijn voor het identificeren en reduceren van ongewenst geluid in producten en omgevingen. Op deze manier zal Sorاما de levenskwaliteit verbeteren van mensen waar ze ook maar zijn: thuis, op het werk of in het verkeer.” ([5], eigen vertaling)

Visie: Sorاما heeft als visie de expert te worden/zijn in het visualiseren en inzicht verkrijgen over geluids- en trillingsbronnen. Tevens willen ze de tools leveren voor oplossingen en de tooling verder uitwerken waarbij samen met kennisinstellingen wordt geïnvesteerd in research.([6], geparafraseerd)

2.2 Organigram



3 Probleemstelling & Doelstelling

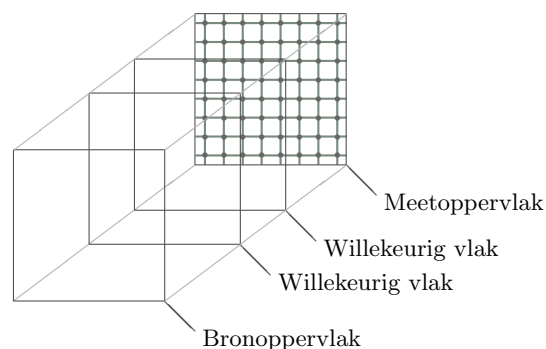
3.1 Probleemstelling

Sorama heeft een geluidscamera (zie figuur 3) ontwikkeld die trillingen en geluid kan visualiseren. Met behulp van deze geluidscamera worden bronnen van geluid, zoals een boor of elektrische tandenborstel, gemeten. Om dit mogelijk te maken bevat de camera een array van microfoons die bij een meting geluid opnemen.



Figuur 3: Close-up van de nieuwste geluidscamera

Het is echter wenselijk om te weten hoe het geluid is ontstaan bij de bron in plaats van hoe het klinkt bij het meetoppervlak. Om dit te kunnen doen wordt er gebruik gemaakt van Near-field Acoustic Holography (NAH), een techniek om van het meetoppervlak terug te rekenen naar het bronoppervlak. Vanuit daar is het mogelijk om het geluidsveld te berekenen tussen het bronoppervlak, de camera en daarachter zoals hieronder te zien is.



Figuur 4: Visualisatie over hoe driedimensionaal wordt gemeten

Aangezien een berekend vlak een tweedimensionaal grid van data is en dit vlak op elke afstand vanaf het bronoppervlak te berekenen is, is er data beschikbaar in alle drie de dimensies. Deze 3D-data wordt echter nog niet in de portal gebruikt. Om de klant mogelijk nog beter van dienst te kunnen zijn wil Sorama tevens kijken naar driedimensionale geluidsvisualisaties binnen de portal. Deze portal is geschreven in Silverlight, een ontwikkelomgeving voor webapplicaties. Deze ontwikkelomgeving gebruikt normaliter het XNA-framework om 3D-functionaliteit toe te voegen. Dit framework doet dat door low-level technologie, die hoort bij het renderen van 3D, te abstraheren zodat de ontwikkelaar zich kan richten op het maken van zijn driedimensionale omgeving.

Er is echter sprake van een bijkomend probleem: Microsoft ziet geen toekomst meer in XNA en beëindigt de support met ingang van april 2014. Er zal dus een alternatief moeten worden gevonden.

Hoe kan een interactieve en toegankelijke driedimensionale geluidsvisualisatie worden ontwikkeld in Silverlight?

3.2 Deelvragen op de probleemstelling

- Wat voor soorten visualisaties van vergelijkbare data bestaan al?
- Wat is een manier om de visualisaties toegankelijker te maken?
- Welke wijzen van besturing kunnen worden toegepast in een interactieve visualisatie zodat het voor een beginnende gebruiker, zonder inhoudelijke kennis van het systeem, gemakkelijk te bedienen is?
- Welke framework of techniek is het beste alternatief op XNA om driedimensionale visualisaties binnen een Silverlight-omgeving te realiseren?

3.3 Opdrachtomschrijving

De opdracht van de stage is om te onderzoeken op welke manier 3D is te programmeren in Silverlight. Ook moet er worden gekeken naar bestaande visualisaties en wat voor soort vergelijkbare producten al bestaan. Met deze kennis zullen er 3 visualisatieprototypes worden ontwikkeld en getest. Uiteindelijk zal het beste prototype worden gekozen welke zal worden gerealiseerd om in de portal te implementeren.

3.4 Doelstelling van het onderzoek

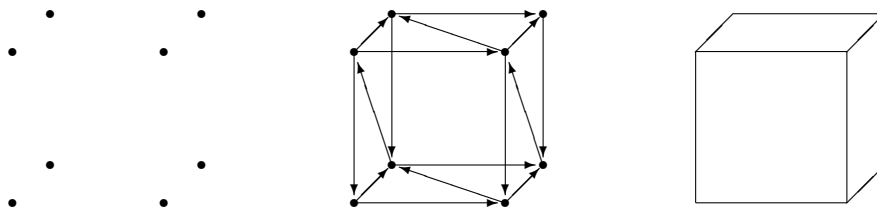
Het doel van het onderzoek is om een driedimensionale geluidsvisualisatie te ontwikkelen die beter inzicht geeft over hoe golven ontstaan en zich voortplanten dan de huidige tweedimensionale beelden die al worden gebruikt. Zo zal een gebruiker sneller en beter begrijpen hoe het gemeten geluid ontstaat en uitdooft.

4 Onderzoek

4.1 Inleiding

Voordat er een zinnig beeld op het scherm ontstaat is er eerst een complex proces nodig. Hierbij biedt XNA ondersteuning. Om dit proces te realiseren zijn er een aantal basiscomponenten nodig. Deze komen in de meeste frameworks voor en zijn: de scène, de camera, objecten bestaande uit vertices en indexen, lichten en een renderer. De scène is een driedimensionale ruimte en bevat de camera, objecten en lichten. De camera is hetzelfde als een videocamera, hij neemt op wat hij ziet in de scène, zonder lichten is het dan ook nogal donker in de ruimte. Als je scène alles bevat behalve de objecten is er nog niet veel interessants te zien. Wat het echt interessant maakt zijn dan ook de objecten. Deze objecten kan je niet zo makkelijk laten zien als een normaal plaatje gezien de manier waarop dit driedimensionaal model is opgebouwd, namelijk uit vertices en indexen.

Het begint met de vertices. Deze geven de punten aan waaruit het 3D-object bestaat, een kubus bijvoorbeeld heeft er acht. De renderer weet echter niet dat deze vertices bij elkaar horen en hoe deze in verbinding met elkaar staan, daar zijn indexen voor. Deze trekken ‘lijntjes’ tussen de vertices en geven zo aan hoe het oppervlak van een object precies is gedefinieerd. Dit is vergelijkbaar met een kleurplaat van kinderen waarbij de punten verbonden moeten worden. De indexen zijn dan de nummers die naast de puntjes staan.



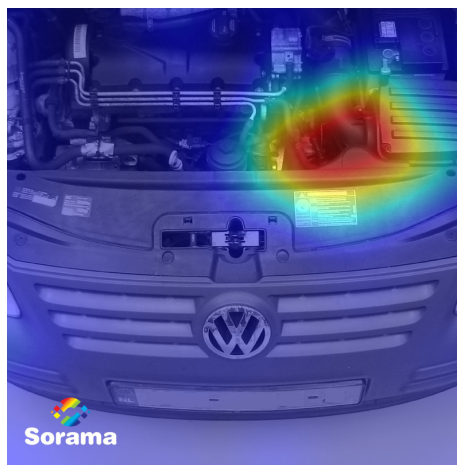
Figuur 5: Opbouw van een driedimensionale kubus

Om dit proces gemakkelijker te maken zijn er frameworks zoals XNA ontwikkeld die dit soort functionaliteit gemakkelijk te gebruiken maken. Zo heeft XNA onder andere een ‘BasicEffect’, een class die standaardinstellingen heeft voor licht, de mogelijkheid geeft om mist te renderen, et cetera. Ook geeft dit framework toegang tot de videokaart zodat rendering sneller is dan op de CPU. Bij aanvang van de stage werd verondersteld dat met XNA zou worden gewerkt om een 3D-omgeving op te zetten binnen de portal. De portal is geschreven in Silverlight, wat sinds versie 5 een subset van XNA gebruikt om 3D-omgevingen mogelijk te maken. Na overleg is er voor gekozen om in eerste instantie drie prototypes te maken. Deze zullen worden getest om dan vervolgens het beste prototype uit te werken tot een implementeerbaar eindproduct. Na twee weken werken is bekend geworden dat XNA niet meer wordt ontwikkeld met als gevolg dat de support eindigt in 2014.[7] Om die reden is er ook gezocht naar alternatieven om driedimensionale visualisaties te implementeren in de portal.

4.2 Soortgelijke visualisaties

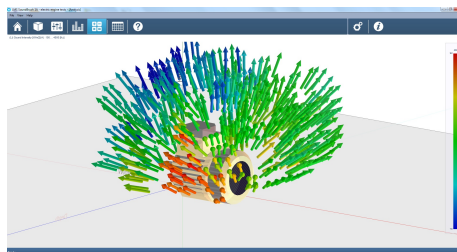
Sorama is niet het enige bedrijf dat metingen doet van geluidsvelden om klanten te helpen. Zo zijn er een aantal andere bedrijven zoals Microflown, LMS en B&K. Deze bedrijven gebruiken niet dezelfde techniek als het stagebedrijf maar hebben wel soortgelijke visualisaties. Deze subsectie beantwoordt dan ook de vraag: “Wat voor soorten visualisaties van vergelijkbare data bestaan al?”

Microflown en B&K hebben allebei een doorzichtige map met kleuren op een foto gelegd. Zo is snel af te lezen aan de visualisatie waar de grootste geluidsbronnen van het object zich bevinden. Een vergelijkbaar resultaat is te zien hieronder.



Figuur 6: Lokalisatie van geluidsbronnen met de geluidscamera

LMS heeft de Soundbrush, een apparaat waarmee in 3D door tracking het geluidsveld wordt gemeten en weergegeven in een driedimensionale omgeving. Alhoewel deze soort visualisatie heel handig is voor hun product is het mogelijk niet goed van toepassing op de portal aangezien de pijlen de richting aangeven, wat data is die Sorama niet beschikbaar heeft. Wel is het interessant om te kijken of een soortgelijke visualisatie mogelijk is door de diepte-data van de Kinect te gebruiken die onderop de geluidscamera is gemonteerd. Zo is het al mogelijk om 3D objecten te reconstrueren met deze speciale camera.[8] De Kinect wordt normaliter gebruikt in combinatie met een Xbox om spelers handelingen uit te laten voeren met hun lichaam.



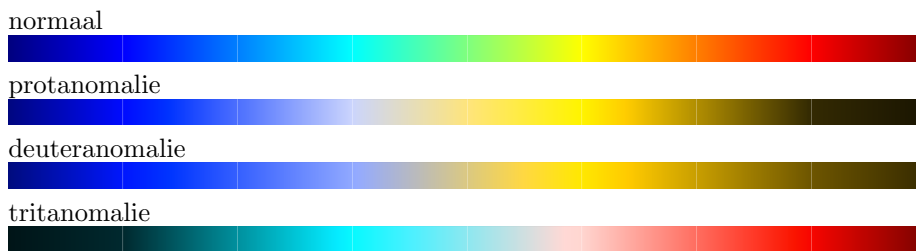
Figuur 7: Visualisatie van LMS[20]

4.3 Toegankelijkheid

Als een product door een grote groep mensen wordt gebruikt, is er altijd een groep waarvoor het product minder toegankelijk is. Zo kunnen dove mensen maar moeilijk met een normale telefoon bellen maar via internet met een camera kan het goed.[21]

Er is een soortgelijk probleem in datavisualisaties. Daar wordt vaak kleur gebruikt om verschillende soorten gegevens en waarden van data te weergeven. De huidige visualisaties in de portal gebruiken bijvoorbeeld de ‘jet’ colormap van Matlab. Om in de stijl van deze visualisaties te blijven wordt dit kleurenschema dan ook gebruikt in de te ontwikkelen prototypes.

Het is echter zo dat tot wel 8% van de mannen en 0.5% van de vrouwen een vorm van kleurenblindheid hebben.[22] Gezien de overgrote meerderheid van gebruikers van de portal man is[23], is het aannemelijk dat daarvan bijna 1 op de 10 moeilijkheden zullen ondervinden bij het onderscheiden van de gebruikte kleuren. Zo is het voor iemand met kleurenblindheid moeilijk om één kleur aan te wijzen als neutraal, ofwel nulpunt, in het geval van de jet colormap zoals te zien in figuur 8. Ook lijken de hogere waardes, ofwel rood, een stuk dichter bij het midden te liggen dan normaal. Hoe kunnen de visualisaties voor deze groep toegankelijker worden gemaakt?



Figuur 8: Matlab's jet colormap

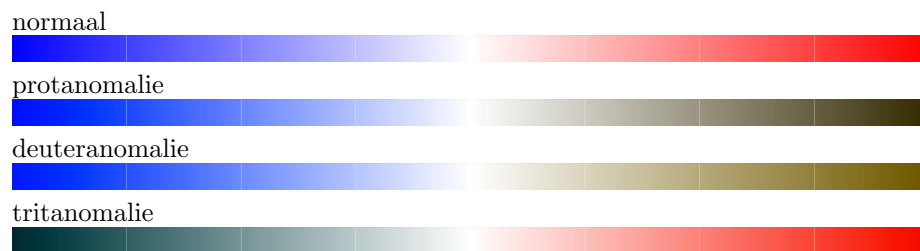
Om dit te realiseren is er besloten om een alternatief kleurenschema te ontwikkelen. Dit schema moet gemakkelijk te interpreteren zijn voor alle mensen met een type kleurenblindheid. Het schema moet waarden kunnen representeren die positief en negatief zijn. Hierdoor is het niet mogelijk om simpelweg van wit naar zwart te gaan omdat er dan informatie ontbreekt over de absolute waarde van een punt. Er zal dus een kleur zijn die, voor allen met een type kleurenblindheid, als het nulpunt kan worden beschouwd. Ook moet er onderscheid kunnen worden gemaakt tussen twee andere kleuren, namelijk welke kleur positief en welke kleur negatief representeert. De intensiteit van deze twee laatste kleuren laat dan de absolute waarde van het punt zien.

Om deze twee kleuren te bepalen is er onderzocht hoe elk type kleurenblindheid precies werkt. Kleurenblindheid kan worden onderverdeeld in 4 groepen:

- **Achromatopsie** Alle soorten kegels functioneren niet en er is dus geen onderscheid in kleur. De wereld wordt minder scherp en in grijsinten waargenomen.

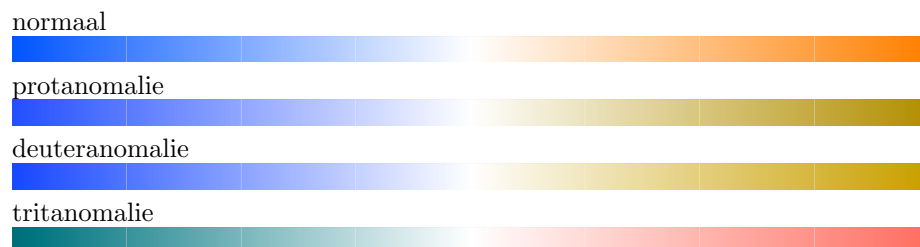
- **Protanomalie, protanopie** Minder gevoelige of ongevoelige L-kegels, respectievelijk.
- **Deutanomalie, deutanopie** Minder gevoelige of ongevoelige M-kegels, respectievelijk.
- **Tritanomalie, tritanopie** Minder gevoelige of ongevoelige S-kegels, respectievelijk.

De L-kegels zijn het meest gevoelig bij licht met een golflengte van 564 tot 580nm, M-kegels van 534 tot 545nm en S-kegels van 420 tot 440nm. Dit komt overeen met rood, groen en blauw licht, respectievelijk. Mensen met protanomalie of deutanomalie worden ook wel rood-groenkleurenblind genoemd. De reden hiervoor is dat groen licht alsnog de L-kegels prikkelt terwijl de M-kegels er niet op reageren in het geval van deutanomalie, omgekeerd hetzelfde bij protanomalie. Hierdoor lijken rood en groen dezelfde kleur te zijn. Dit komt doordat de L en M-kegels qua gevoeligheid dicht bij elkaar liggen in vergelijking met de S-kegels. Om deze reden is er gekozen om een kleurenschema te maken met rood en blauw als maxima.



Figuur 9: Andreas' kleurenschema voor kleurenblinden

Dit kleurenschema blijkt goed te werken, het probleem is alleen dat rood voor mensen met protanomalie zwart wordt. (Figuur 9) Er is daaropvolgend een alternatief schema gevonden die van een lichtblauw naar wit naar oranje loopt.[24] Deze heeft als voordeel dat de twee kleuren altijd te onderscheiden zijn voor alle typen kleurenblindheid zonder dat een van de twee donkerder wordt. Deze is te zien in figuur 10.



Figuur 10: Oranje-blauw kleurenschema voor kleurenblinden

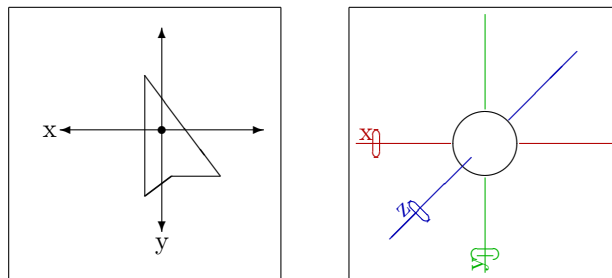
4.4 Besturing van visualisatie

Om er voor te zorgen dat een nieuwe gebruiker met gemak de visualisatie kan besturen wordt er gekeken naar bestaande 3D-programma's zoals Trimble Sketchup, Blender en Autodesk 3ds Max. Daar wordt gekeken naar hoe er interactie plaatsvindt tussen de gebruiker en het programma en hoe dit uitwerking heeft op de 3D-omgeving binnen dit programma. Ook wordt er gekeken naar de besturingselementen van programma's die media kunnen afspelen zoals audio en video voor het afspelen van de animatie. Dit beantwoordt dan ook de deelvraag over de wijzen van besturing.

4.4.1 Rotatie van het object

Om alle zijden van een object te kunnen bekijken zijn er twee opties: het object draaien of het standpunt van de toeschouwer veranderen. Zowel Sketchup als 3ds Max lijken het standpunt te veranderen, terwijl Blender het object roteert. Dit doen alle drie de programma's met het slepen van de muis over het 3D-canvas heen.

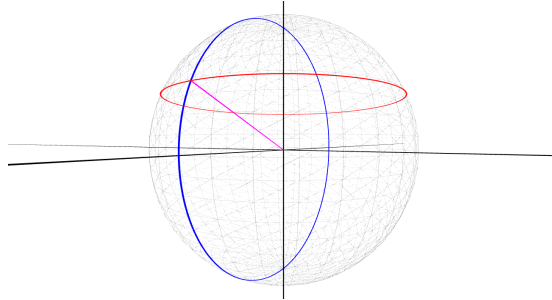
Er is met Blender begonnen omdat de projectleider al bekend is met de twee andere programma's en het een mooie uitdaging lijkt om deze ook te leren kennen. Er is geprobeerd om de besturing hetzelfde te maken als dit programma, deze gebruikt 'trackball rotation'. Hierbij lijkt het object om zijn eigen assen heen te draaien. Het implementeren hiervan blijkt vrij moeilijk te zijn omdat de muis twee dimensies heeft waarover het beweegt terwijl het object drie assen heeft om te draaien. Ook is het koppelen van een bepaalde dimensie van de muis aan een enkele as onbetrouwbaar. De eerste keer dat een object draait gaat het goed, de opeenvolgende keren dat het object draait gaat dit echter niet zoals verwacht. Zo kan het zijn dat een object de verkeerde kant op draait doordat deze bij een andere as volledig is omgekeerd.



Figuur 11: Dimensies van muis & assen van object

Daarna is het veranderen van het standpunt geprobeerd. Het object staat in het midden van de scène en de camera verandert van positie terwijl deze naar het object blijft kijken. Het omhoog slepen van de muis zorgt ervoor dat de bovenkant van het object wordt gezien, het zijwaarts slepen van de muis laat het object rond de Z-as draaien en het scrollen met de muiswiel zorgt voor een zoom-effect. Dit gebeurt door het gebruik van bolcoördinaten. Deze coördinaten vormen een driedimensionaal stelsel met 3 waarden: φ , θ en r . Hierbij staat φ (phi) voor azimut, θ (theta) voor helling en r voor radius. De

azimut loopt horizontaal om het object heen net als lengtegraden, de helling loopt verticaal over het object vergelijkbaar met breedtegraden en de radius definieert de afstand tot het object. (Figuur 12.)



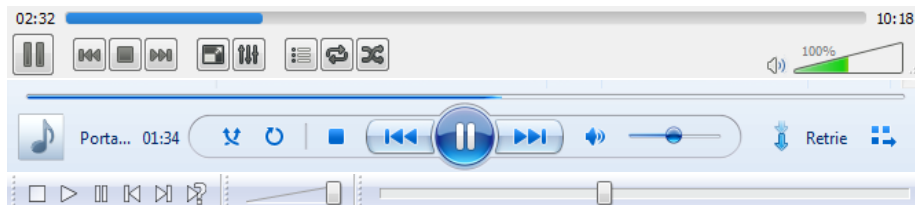
Figuur 12: Rood = azimut, blauw = helling, magenta = radius

Om de camera te verplaatsen met de muis wordt de volgende koppeling gemaakt: de X verandert de azimut, de Y verandert de helling en het scrollen verandert de radius. Nu moet er nog een laatste berekening gebeuren, namelijk het omrekenen van bolcoördinaten naar cartesische coördinaten zodat de camera daadwerkelijk een positie wordt gegeven. Dit gebeurt met de volgende formules:

$$\begin{aligned} X &= r \sin \theta \cos \varphi \\ Y &= r \sin \theta \sin \varphi \\ Z &= r \cos \theta \end{aligned}$$

4.4.2 Afspelen animatie

Aangezien geluid door de tijd heen verandert is er gekeken naar de bediening van andere programma's (Figuur 13) die zowel audio als video kunnen afspelen. Om de visualisatie gemakkelijk bedienbaar te maken voor gebruikers die onbekend zijn met het systeem is er besloten om vast te houden aan de conventies van deze programma's. Zo is er normaal een afspelerbalk welke indiceert hoeveel tijd is verstreken en waarmee het mogelijk is om deze tijd te beïnvloeden. Daarnaast is er een play/pause-knop om het spelen te pauzeren en hervatten, een stopknop om het spelen te stoppen en de tijd terug te zetten naar nul. Ten slotte zijn er over het algemeen knoppen met de functies shuffle, vorige, volgende, repeat en een slider om het volume aan te passen. Aangezien er geen lijst is met af te spelen bestanden en er geen geluid wordt gebruikt worden deze laatste knoppen achterwege gelaten.



Figuur 13: Besturingselementen van VLC Media Player, Windows Media Player en Foobar2000

4.4.3 Conclusies

Het roteren van het object geeft geen bevredigend resultaat, het is niet consistent hoe het object beweegt en het is moeilijk te realiseren. De tweede wijze van besturing is goed te implementeren aangezien er niet veel berekeningen nodig zijn en het consistent blijft hoe de camera om het object heen draait. Daarom is er gekozen om de tweede optie te gebruiken. Voor het afspelen van de animatie door de tijd zullen de 'play/pause'-knop, 'stop'-knop en slider worden gebruikt.

4.5 Alternatieve frameworks

Aangezien XNA volgend jaar geen support meer heeft zal er moeten worden gekeken naar alternatieven.

Voor het zoeken is er op internet gezocht naar open source implementaties van XNA om de overgang tussen frameworks zo simpel mogelijk te maken. Ook is er gezocht naar wat anderen op internetfora deden om 3D te programmeren binnen Silverlight 5. Alle gevonden alternatieven zijn bekeken en er is een lijst gemaakt met voor- en nadelen. Ook zijn de alternatieven uitgetest en is er gekeken of ze aan de voorwaarden voldoen. (Bijlage III.)

Een van de voorwaarden is dat de framework minimaal Windows 7 en nieuwer ondersteund omdat dit ook het geval is met de portal. Daarnaast is het belangrijk dat de prestaties goed zijn zodat er geen haperingen in het beeld zijn voor gebruikers met oudere systemen en dat het alternatief uitbreidbaar is qua functionaliteit.

4.5.1 Alternatieven

Met voorgaande voorwaarden is er gezocht naar alternatieven voor XNA, daarbij kwamen de volgende frameworks aan bod:

ANX

Een veelbelovend alternatief aangezien het een 'drop-in replacement' van het XNA framework is.[9] Later bleek dat Windows 8 hierbij benodigd is om te ontwikkelen.

Grommet

Afgewezen omdat alleen 2D-functionaliteit en input beschikbaar is.[10] Deze optie is verder niet geëvalueerd.

MonoXNA

Een sterk verouderd framework met de laatste update rond 2009 of 2010.[11] Het is hiermee niet gelukt om een project te compileren doordat oudere versies nodig zijn van software zoals DirectX.

MonoGame

Dit framework is origineel gebaseerd op MonoXNA in combinatie met SilverSprite.[12] Er is enkel een lege sample mee gecompileerd, waarna het duidelijk werd dat het niet mogelijk is om dit binnen een Silverlight-omgeving te gebruiken.

Uit deze resultaten blijkt dat er geen directe vervanging is voor XNA binnen Silverlight. Ontevreden met deze uitkomst is er verder gezocht buiten de strekking van Silverlight naar frameworks om 3D te programmeren. Zo is er gedacht aan het vooraf renderen van beelden en frameworks buiten Silverlight en C# die ook de mogelijkheid hebben om op het web beelden te genereren. Dit heeft het volgende opgeleverd:

openFrameworks

Dit is een zeer uitgebreid framework in C++ met tal van mogelijkheden wat betreft 3D, connectie met hardware en interactie.[13] Hiermee is het alleen mogelijk om een desktopapplicatie te maken en geen webapplicatie.[14] Ook is er naar gekeken om dit framework als renderserver te gebruiken om driedimensionale beelden te renderen. Dit was echter zodanig langzaam dat dit niet rendabel is. Dit is te lezen in bijlage IV.

Silverlight-WebGL

Dit is een implementatie om een WebGL-canvas vanuit Silverlight te besturen.[15] De prestaties zijn erg goed omdat de rendering direct op de GPU plaats vindt. De problemen van dit alternatief zijn dat er geen mogelijkheid tot gebruikersinteractie is, sommige features hadden geen support bij een aantal browsers. Ook is er geen error reporting waardoor één fout in de code resulteert in een leeg scherm zonder duidelijke oorzaak.

Processing

Open source framework gebaseerd op Java voor mensen die beelden, animaties en interacties willen maken.[16] Deze werkt ook op het web door middel van Processing.js[17], dit is echter zeer langzaam.[18]

Perspective 3D

Ondanks dat er specifiek werd gezocht naar frameworks buiten Silverlight en C# is dit framework alsnog gevonden. Deze implementatie is gebaseerd op WPF 3D binnen Silverlight in plaats van XNA.[19] Deze wordt wel op dezelfde manier toegepast en is direct binnen Silverlight te gebruiken. Hierbij is het echter niet mogelijk om zelf modellen te maken wat de mogelijkheden in hoge mate beperkt.

4.5.2 Keuze alternatief

Op basis van deze kennis is besloten om voor de rest van het project XNA te gebruiken en WebGL voor het vervolg. Er is voor het verloop van dit project niet voor een ander framework gekozen om onder andere de volgende redenen: WebGL is nog niet overal even goed ondersteund en het ontwikkelen van een visualisatie hierin wordt daardoor gecompliceerder. Perspective 3D is een erg goede optie gezien de Silverlight-integratie maar deze heeft geen optie om zelf modellen te maken wat de mogelijkheden te veel beperkt. Er is dus gekozen om WebGL te gebruiken wanneer XNA niet meer kan worden gebruikt. Dit omdat WebGL de meeste mogelijkheden biedt, goede prestaties heeft en als het framework volwassener is de ondersteuning waarschijnlijk ook beter zal zijn.

4.6 Conclusies

De soorten visualisaties zijn niet erg uitgebreid, wat vooral veel voor komt is een afbeelding van het object met daar over heen een map van kleuren.

Het alternatieve kleurenschema is ontwikkeld om mensen met een type kleurenblindheid beter van dienst te zijn. Voor deze groep gebruikers werkt het oranje-blauwschema beter dan het huidige regenboogachtige schema van Matlab. Het is voor hen makkelijker te bepalen welke punten in de visualisatie nul zijn, welke kleur maximum en welke minimum is.

Voor de besturing is er voornamelijk gebruik gemaakt van al bestaande conventies zodat nieuwe gebruikers snel overweg kunnen met het systeem. Zo is het gemaakt dat het verslepen van de muis de camera beweegt volgens bolcoördinaten en niet het object zelf. Ook zijn de mediaknoppen van diverse mediaspelers samengevoegd en toegepast om de animatie te regelen in de visualisatie.

Tenslotte is er onderzocht welk framework als alternatief kan worden gebruikt om de visualisaties in te maken wanneer XNA geen ondersteuning meer ontvangt. Er is gekozen voor WebGL omdat deze het beste uit te breiden is, goede prestaties heeft en te interfacen is met Silverlight. Het is echter zo dat ten tijde van het project er met XNA zal worden gewerkt. Gezien de duur van het project zou het te veel tijd kosten om alles om te zetten naar WebGL en daarin verder te ontwikkelen.

5 Realisatie & implementatie

5.1 Inleiding

Als eerste werd er in XNA gewerkt om erachter te komen hoe het framework werkt en op welke manieren animaties konden worden gemaakt. Dit werd voornamelijk gedaan vanuit het boek *Silverlight in Action*[25], waarvan hoofdstuk 26 & 27 specifiek over 3D gaan. Tijdens het leren over XNA werd het bekend dat Microsoft geen toekomst meer ziet in het framework en dus de support beëindigt met ingang van april 2014.[7] Dit houdt in dat er geen updates meer komen voor het framework, zoals updates voor functionaliteit, beveiliging en compatibiliteit.[26] Het gevolg hiervan is dat de beveiligingsgaten niet worden gedicht en het product in de toekomst minder goed of zelfs niet meer zal werken. Tijdens het ontwikkelen van het eerste prototype zijn er dus verschillende alternatieven onderzocht en is er besloten om het eerste prototype in openFrameworks te maken. De reden hiervan is dat het framework zeer uitgebreid is wat betreft mogelijkheden zoals bestanden lezen, 2D/3D beelden en input. Zo is het gemakkelijk om snel een prototype te maken en te testen. Bij het realiseren van het tweede prototype is er ook begonnen met openFrameworks. Daarna is het prototype geport naar XNA omdat dit framework is gekozen voor de rest van het project, zoals beschreven in paragraaf 4.5.1.

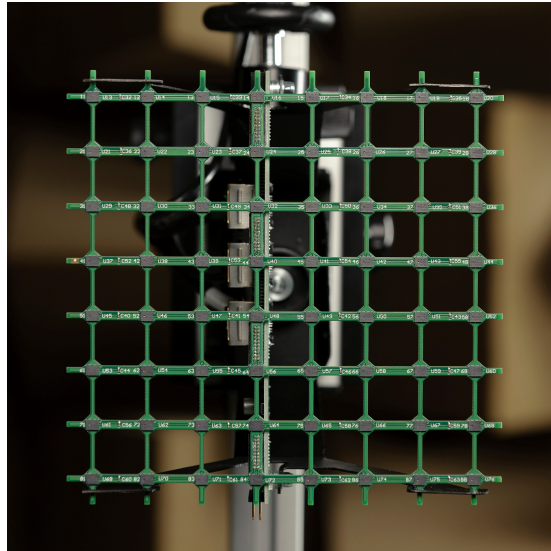
5.2 Eerste prototype

5.2.1 Doel

Het doel van dit prototype is om te kijken naar wat voor waarde 3D kan toevoegen aan de bestaande visualisaties. Zo zal de driedimensionale data niet direct worden gebruikt maar wordt er een andere manier gezocht om de data van de bestaande beelden te visualiseren. De data die op het moment het gemakkelijkst is om te gebruiken is de data direct van het microfoonarray. Deze microfoondata staat in een csv-bestand, een manier om tabelvormige gegevens als platte tekst op te slaan. In dit geval is elke regel een momentopname in de tijd en bevat 64 waarden, elke daarvan representeert de uitslag van één microfoon.

```
2.6808, 2.9762, 1.3734, 3.4713, 3.7007, 1.8305, 3.4259, 1.6275, ...
2.7392, 3.1374, 1.4364, 3.1962, 3.3071, 1.4936, 3.1746, 1.6105, ...
2.8054, 3.2821, 1.3341, 2.9009, 2.923, 1.1562, 2.8326, 1.408, ...
2.9255, 3.2438, 1.1346, 2.603, 2.7294, 0.96048, 2.5765, 1.1053, ...
2.9583, 3.1111, 0.89694, 2.4892, 2.7804, 1.0084, 2.464, 0.81233, ...
2.9993, 2.9131, 0.76179, 2.6255, 3.1019, 1.2748, 2.5823, 0.66528, ...
2.8922, 2.7453, 0.75046, 2.8877, 3.488, 1.632, 2.8733, 0.75131, ...
2.8284, 2.6712, 0.89476, 3.2423, 3.8558, 1.9401, 3.188, 0.98547, ...
2.7316, 2.7241, 1.1085, 3.4679, 3.9678, 2.0354, 3.4291, 1.3078, ...
2.6878, 2.8845, 1.334, 3.5165, 3.8011, 1.9414, 3.4614, 1.5678, ...
```

Het array heeft geen 64 microfoons achter elkaar zoals in het bestand maar ze zijn gepositioneerd in een 8x8 grid. Dit heeft overeenkomsten met een LED-kubus waarbij een laag ook bestaat uit componenten in een grid. Zo kwam het idee om de data te visualiseren als zo een kubus.

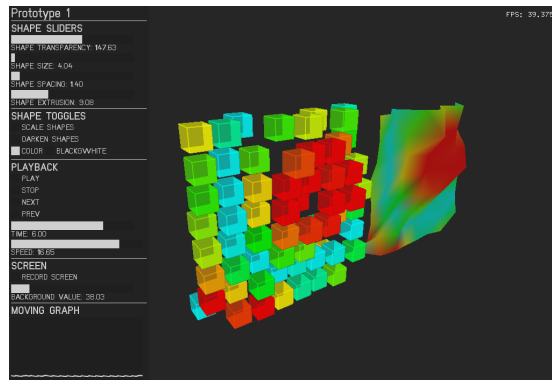


Figuur 14: Het kleine microfoonarray

5.2.2 Realisatie

Het lezen van data uit een bestand gaat vrij langzaam, vandaar dat deze als eerste moet worden ingeladen. Bij het inladen wordt elke regel gelezen en worden de waardes gescheiden, waarna deze in een aparte array worden gezet. Er is een 8x8 grid van punten geplaatst in een 3D-ruimte om de microfoons te representeren. De bijbehorende waarden worden daarna elke keer in het grid geladen op de geschikte plek met X- en Y-coördinaten.

Om het effect van een LED-kubus na te maken is er een kubus geplaatst op elk punt van het grid om een microfoon te representeren. Deze kubus heeft allerlei eigenschappen die aan te passen zijn zoals kleur, positie en formaat. Als eerste is de kleur aangepast op basis van de waarden zoals dat in een LED-kubus gebeurt. Bij een hoge waarde wordt de kubus rood en bij een lage waarde wordt deze blauw overeenkomstig met het kleurenschema van de bestaande beelden in de portal. Om aan te geven dat het grid niet alleen is geïnspireerd door LED's maar ook door microfoons, is er gekeken naar de werking van een microfoon. Een microfoon kan geluid opnemen doordat een membraan meetrilt met het binnenkomende geluid waarmee een elektronisch signaal wordt afgegeven. Op basis hiervan wordt de Z-coördinaat van een kubus aangepast en gaat deze oververloopt van tijd bewegen. Zo is het eerste prototype ontstaan.



Figuur 15: Prototype 1, gemaakt in openFrameworks

Naast alleen de kleur en de positie te veranderen is er gekeken naar andere aanpassingen gebaseerd op andere gegevens afleidbaar van de gebruikte data. Het is dan mogelijk om meerdere soorten data tegelijk te weergeven, zoals de verhouding tussen verschillende waarden in het grid en het verschil in waarde relatief aan een punt zelf. Zo is er geëxperimenteerd met het formaat van de kubus, als deze over verloop van tijd weinig verandert zal de kubus ook klein zijn. Daarnaast is gekeken naar de ruimte tussen kubussen, de hoeveelheid uitslag van een kubus, een zwart-witkleurenschema en de kleurverzadiging afhankelijk van de gevoeligheid van een microfoon. Tevens kunnen andere variabelen worden aangepast zoals achtergrondkleur, de snelheid van de animatie enzovoort.

5.2.3 Resultaat

Het resultaat is een prototype met goede prestaties. Dit betekent dat hij tijdens het afspelen van de animatie geen, door de gebruiker observeerbare, haperingen vertoont. Het prototype is interactief zodat de gebruiker het object van alle kanten kan bekijken, de animatie kan afspelen en stoppen, stapsgewijs door de tijd heen gaan, de data op verschillende manieren kan weergeven en de positie tussen vormen onderling kan aanpassen naar eigen inzicht.

5.2.4 Reactie van gebruikers

Het prototype vereist echter vrij veel uitleg aan de gebruikers. Zo is het niet direct duidelijk wat ze precies zien en hoe de extra gegevens van de data er in is verwerkt. Ook is er de opmerking dat de kubussen alleen maar kleine stukjes van de data representeren aangezien er niks tussen de vormen in zit. Wel zijn ze positief over hoe de interactie is met de visualisatie op zich en hoe deze wordt bestuurd. Het is duidelijk en eenduidig met hoe normaal een 3D model wordt bekeken. Als laatste is er de opmerking dat het een leuke manier is om de data te visualiseren, alleen niet goed van toepassing als implementatie in de portal.¹

¹Dit alles is mondeling besproken. Hiervan is geen logboek of opname beschikbaar.

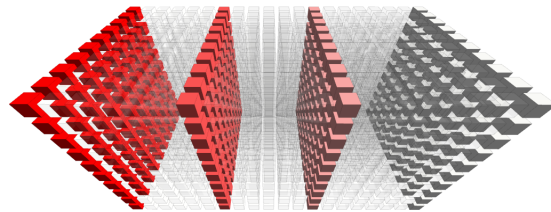
5.2.5 Evaluatie

Het prototype heeft niet het beoogde resultaat behaald. De verwachting was dat mensen positiever zouden reageren op het feit dat er meerdere soorten data in een keer beschikbaar zijn. Ook is het jammer dat er veel uitleg nodig is om duidelijk te maken wat de visualisatie precies laat zien, iets wat niet mogelijk is als de gebruiker het systeem zelf gebruikt.

5.3 Tweede prototype

5.3.1 Doel

Het doel van dit prototype is om te zorgen dat het inzichtelijk wordt hoe geluidsgolven in de ruimte voortplanten. Hiervoor wordt gebruik gemaakt van het feit dat er op elke afstand van het meetoppervlak data te berekenen is. Aangezien de gebruikers bij het eerste prototype niet erg te spreken waren over de kubussen is er teruggespreng naar het vlak. De portal laat standaard het bronoppervlak zien zoals aangegeven is in figuur 16, welke is berekend uit het meetoppervlak. Vanaf het bronoppervlak is het ook mogelijk om op willekeurige afstand een vlak te berekenen.



Figuur 16: Rood is het bronoppervlak, lichtere kleuren zijn vlakken verder daar van af

Door op regelmatige afstanden een vlak uit te rekenen, ontstaat er een stapel van vlakken.

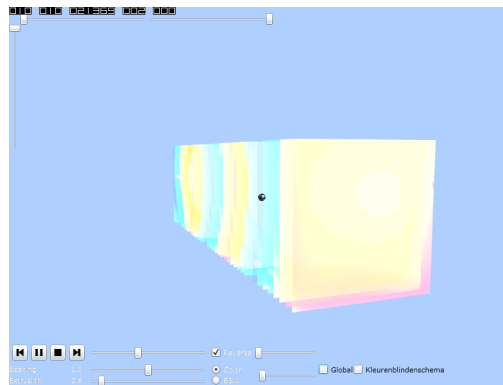
5.3.2 Realisatie

Om te beginnen is er een dataset aangeleverd met de benodigde vlakken, 41 in totaal. Deze set data komt in de vorm van 41 csv-bestanden met een grid van complexe getallen. Complexe getallen zijn een combinatie van een reëel getal en een imaginair getal. Deze complexe getallen worden gebruikt om golven te beschrijven waarmee het mogelijk is ze uit te rekenen met een gespecificeerde tijd. Voor het gemak is er begonnen in Processing een script te maken om deze complexe getallen om te zetten naar bruikbare getallen.

Daarna is in XNA gestart met vlakken die evenveel vertices als datapunten hebben. Elke vertex heeft dan drie gegevens, namelijk positie, normaal en kleur. De positie wordt uitgedrukt in X, Y en Z waarbij X en Y de plaats op het vlak bepalen en Z de uitslag. De normaal is de richting waarin de vertex wijst, dit wordt gebruikt bij het genereren van schaduwen. Als laatste is de er kleur, deze is aan te passen naar eigen inzicht. In het geval van de geluidsvisualisatie wordt de kleur aangepast op basis van de Z-waarde met het jet kleurenschema om

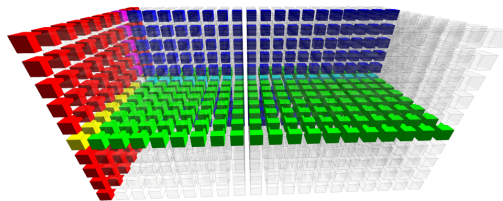
deze waarde aan te geven. Naast het jet kleurenschema is er een kleurenschema ontwikkeld voor kleurenblinden wat wordt beschreven in 4.3. Deze is in het prototype beschikbaar gemaakt door middel van een knop mocht een gebruiker dit schema willen gebruiken.

De vlakken zijn daarna achter elkaar gezet om een balk te vormen waar het mogelijk is om te zien hoe de golven er door propageren. Dit is echter niet zo'n succes door het grote aantal vlakken die elkaar overlappen. Als de vlakken half doorzichtig zijn wordt het beeld nog slechter zoals te zien op de volgende pagina. Ook gaat de prestatie dan zodanig omlaag dat er haperingen in het beeld komen.



Figuur 17: 41 objecten met een snelheid van maar 10 FPS in plaats van 60

Om dit op te lossen is er voor gekozen om niet alle data weer te geven maar alleen uitsnedes daarvan. Om dit mogelijk te maken is het Processing-script aangepast om van de gebruikte dataset alleen maar een bepaalde kolom van elk vlak te gebruiken en deze achter elkaar te zetten. Een illustratie hiervan is te zien in figuur 18. Deze uitsnede wordt op dezelfde manier gebruikt als het normale meetoppervlak in de visualisatie, echter is deze 90 graden gedraaid zodat hij loodrecht op het meetoppervlak staat en de ruimte in gaat. Het resultaat hiervan is te zien in de volgende paragraaf.

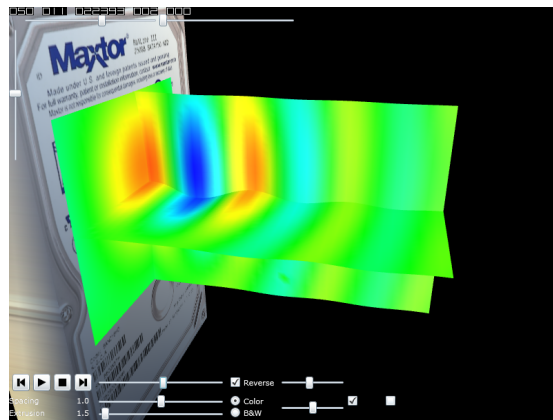


Figuur 18: Data met diepte-uitsnede toegepast. Rood is bronoppervlak, groen en blauw zijn uitsnedes

Het voordeel hiervan is dat er nu alleen maar enkele vlakken in de visualisatie staan wat er voor zorgt dat de prestatie weer goed is. Ook is het dan mogelijk om een bepaald gebied te selecteren waar de gebruiker in geïnteresseerd is zonder obstructies van andere vlakken.

Een nieuwe probleem dat is opgetreden door het gebruik van extra data heeft te maken met de kleuren. De gebruikte vlakken kunnen met elkaar kruisen en tegen elkaar aan staan. Het probleem is echter dat de kleuren van elke vertex per vlak wordt gegenereerd, gebaseerd op de minima en maxima van elk vlak. Waar de vlakken in elkaar overlopen is het mogelijk dat de ene vertex geel is en de andere rood. Om dit op te lossen is er voor gezorgd dat de kleuren worden gegenereerd met globale minima en maxima waardoor de kleuren op aanliggende en kruisende vertices altijd hetzelfde zijn.

5.3.3 Resultaat



Figuur 19: Prototype 2, met uitsnedes

Het resultaat is een prototype met goede prestaties waarbij de golven duidelijk in de ruimte bewegen in tegenstelling tot de al bestaande visualisaties in de portal. De besturing is hetzelfde als bij de vorige prototype. De vlakken zijn verschuifbaar ten opzichte van elkaar zodat de gebruiker het gebied van interesse kan selecteren. Hierbij wordt tegelijk de correcte data ingeladen. Met dit prototype is het ook mogelijk om het alternatieve kleurenschema te selecteren mocht de gebruiker daarmee beter de visualisatie kunnen interpreteren.

5.3.4 Reactie van gebruikers

De reacties van de gebruikers is ronduit positief. Iedereen ziet direct wat de visualisatie weergeeft en wat voor voordeel dit heeft, namelijk het beter begrijpen hoe het geluid van het object afstraalt. Ook merkte een op dat het mooi zou zijn om doorzichtigheid toe te passen op de vlakken zodat het duidelijk is waar het geluid precies is ontstaan.² Een nadeel van dit prototype is dat het selecteren van een andere uitsnede er voor zorgt dat het programma een tijdje niet reageert en vast gelopen lijkt te zijn, wat verwarrend werkt. Dit komt doordat er elke keer een heel bestand moet worden ingeladen.

²Dit alles is mondeling besproken. Hiervan is geen logboek of opname beschikbaar.

5.3.5 Evaluatie

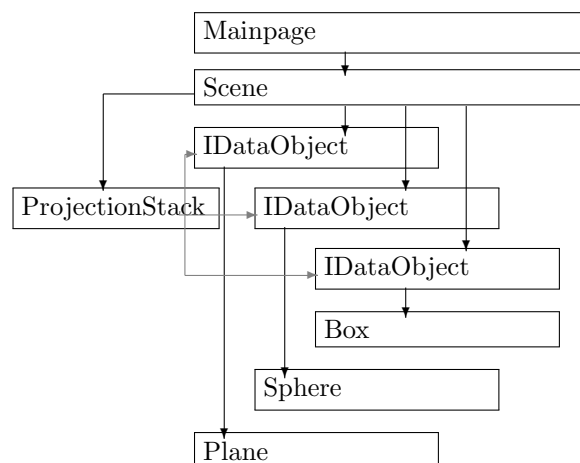
Het prototype heeft aan de verwachtingen voldaan. Zo is het duidelijk hoe de golven door de ruimte lopen en is de prestatie bij het bekijken en besturen goed. Wel moet bij de implementatie er voor worden gezorgd dat bij het selecteren van een andere uitsnede de animatie niet vastloopt en niet reageert. Er is echter een grote kans dat dit goed te doen is aangezien er straks een dataset wordt ingeladen direct vanaf de server in plaats van lokale bestanden.

5.4 Conclusies & aanbevelingen

Gebruikers zijn niet erg positief over het eerste prototype waar er wordt getracht verschillende soorten data te visualiseren door middel van objecten en de eigenschappen daarvan zoals formaat en kleur. Wel zijn ze erg positief over het tweede prototype waar uitsnedes worden gevisualiseerd door middel van vlakken, waarbij kleuren de verschillende waarden over tijd laten zien. Gezien dit prototype aan de verwachtingen voldoet en de reacties van de gebruikers positief zijn is er besloten om deze te implementeren. De manier waarop deze wordt geïmplementeerd is zo dat de visualisatie, oftewel de objecten zoals vlakken zelf, gemakkelijk uitwisselbaar zullen zijn. Zo is het later mogelijk om andere soorten visualisaties te creëren zonder dat het hele product moet veranderen.

5.5 Implementatie basis

Tijdens het schrijven van dit verslag wordt er nog gewerkt aan de implementatie van het eindproduct en is er nog geen vaste vorm. Het plan is echter om het product als volgt te implementeren. De mainpage heeft een scène, deze zorgt ervoor dat alle objecten worden getekend, de camera op zijn plaats staat en de lichten goed werken. Hierin staan ook een aantal IDataObjects. De IDataObject is een object die op verschillende manieren kan worden weergegeven, zoals door een doos of een vlak. In dit object kan een ProjectionStack worden geladen, de set van data die wordt gevisualiseerd. Zo is de scène niet afhankelijk van wat voor objecten er in staan, hoe groot de dataset is en is het gemakkelijk verschillende soorten visualisaties te maken. Zo is het product gemakkelijk uit te breiden en aan te passen in de toekomst.



6 Reflectie

6.1 Reflectie op onderzoek

Het onderzoek is over het algemeen goed verlopen. Doordat pas tijdens de stage bekend werd dat de ondersteuning voor XNA op zou houden is er vooraf geen rekening gehouden met het onderzoeken van alternatieven. Dit gedeelte van het onderzoek heeft er voor gezorgd dat er minder tijd over was voor het maken van prototypes en deze te testen. Desalniettemin is de gekozen prototype goed te gebruiken in de portal. Daarbij zijn de resultaten wat betreft alternatieven van toegevoegde waarde wat betreft de toekomst van het project. Zo kan geld en tijd worden bespaard omdat men vooraf weet dat ze over moeten stappen op een nieuw framework en het volledig opnieuw moet worden gedaan. Wel kan er nog meer worden geëxperimenteerd met visualisatieconcepten.

6.2 Aanbevelingen

Een aanbeveling is om te kijken naar de mogelijkheid om shaders te maken en te gebruiken in de driedimensionale visualisaties. Dit zou goed kunnen werken om speciale effecten te bereiken zoals betere schaduwen, het bekende anaglyph effect met rood-cyaan bril voor een ‘echte’ 3D ervaring enzovoort. Daarnaast zou het mogelijk moeten zijn om met geometry shaders de objecten nog vloeiender te maken door middel van tessellation. Dit zou ten goede komen aan het verloop van kleuren in de visualisaties.

Verder zou het nuttig zijn om te kijken naar wat er mogelijk is met de Kinect die onder op de geluidscamera is gemonteerd. Deze wordt tot nu toe gebruikt om een afbeelding te maken van het gemeten object met de normale camera. De Kinect bevat echter ook een diepte-camera, het zou interessant kunnen zijn om te kijken naar de mogelijkheden van deze diepte-data om bijvoorbeeld een object te creëren waarvan het geluid afstraalt. Ook raad ik aan om het mogelijk te maken verschillende metingen aan elkaar te koppelen van bijvoorbeeld verschillende kanten van een object. Zo kan een meting van een kamer beter in beeld worden gebracht voor een gebruiker.

Daarnaast is het zo dat Silverlight op dit moment tot 2021 support zal krijgen, wat daarna gebeurt is niet zeker te zeggen. Als WebGL uiteindelijk inderdaad wordt gebruikt om de visualisaties te implementeren is het slim om ook te kijken naar wat voor rol HTML5 en Javascript kunnen spelen, om zo een webapplicatie te maken zonder Silverlight.

Literatuurlijst

- [1] (30 mei 2007). Regulation No 51 of the Economic Commission for Europe of the United Nations (UN/ECE) — Uniform provisions concerning the approval of motor vehicles having at least four wheels with regard to their noise emissions Addendum 50: Regulation No 51 van *Official Journal of the European Union*. Geraadpleegd op 22 mei 2013 van <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2007:137:0068:0115:EN:PDF>
- [2] (2013). Products van *Blacknoise technologies*. Geraadpleegd op 22 mei 2013 van <http://www.blacknoise.com/en/site/page/products.it>
- [3] (2013). NB-eLoop®Serie 120mm van *Blacknoise technologies*. Geraadpleegd op 4 juni 2013 van http://www.blacknoise.com/en/products/it/8/Noiseblocker-NB_eLoop_Series_120mm
- [4] Scholte, R. (2008). *Fourier Based High-resolution Near-field Sound Imaging*. Gedownload van <http://alexandria.tue.nl/extra2/200910261.pdf>.
- [5] (z.d.). Mission Statement. Van *Sorama, visualizing sounds and vibrations*. Geraadpleegd op 1 mei 2013, van <http://www.sorama.eu/About/Mission>.
- [6] (2012?). Wat verwacht je van de toekomst? Van *Maak dat ding stiller*. Gedownload op 7 mei 2013, van <http://www.stw.nl/sites/stw.demo.infi.nl/files/32-Sorama-6618-fotos-def.pdf>, nu <http://www.stw.nl/sites/stw.nl/files/32-Sorama-6618-fotos-2013-def.pdf>.
- [7] Hruska, J. (1 februari 2013). Microsoft kills Xbox 360/PC cross-platform development, declares DirectX “no longer evolving” *ExtremeTech*. Geraadpleegd op 18 februari 2013, van <http://www.extremetech.com/gaming/147289-microsoft-kills-xbox-360pc-cross-platform-development-declares-directx-no-longer-evolving>
- [8] z.d. ReconstructMe van *Profactor* Geraadpleegd op 8 juni 2013, van <http://reconstructme.net/>
- [9] Rosenkranz, R. (4 februari 2013). What is the purpose of the ANX.Framework? van *ANX.Framework - Home*. Geraadpleegd op 21 februari 2013, van <http://anxframework.codeplex.com/>
- [10] Leduc, B. (30 oktober 2010). .NET Features van *Grommet - A .NET Micro Framework Library - Home*. Geraadpleegd op 21 februari 2013, van <http://grommet.codeplex.com/>
- [11] (24 februari 2013). *MonoXNA — managed game development* Geraadpleegd februari 2013, van <http://www.monoxna.org/>, nu <http://web.archive.org/web/20130224140905/http://www.monoxna.org/>.
- [12] (z.d.). Acknowledgements van *MonoGame*. Geraadpleegd op 21 februari 2013, van <http://monogame.codeplex.com/>.

-
- [13] (z.d.). about, *openFrameworks*. Geraadpleegd op 8 mei 2013 van <http://www.openframeworks.cc/about/>
- [14] (23 november 2008). Online application with openFrameworks *openFrameworks forum*. Geraadpleegd op 8 mei 2013 van <http://forum.openframeworks.cc/index.php?topic=1342.msg7234#msg7234>
- [15] (z.d.). *Silverlight-WebGL* Geraadpleegd op 26 februari 2013, van <https://github.com/Arloreat/Silverlight-WebGL>.
- [16] (z.d.). *Processing* Geraadpleegd rond 3 mei 2013, van <http://processing.org/>.
- [17] (z.d.). *Processing.js* Geraadpleegd rond 7 mei 2013, van <http://processingjs.org/>.
- [18] Kirn, P. (15 mei 2008). Processing.js: Very Cool, But JavaScript Nuts Go Overboard van *Create Digital Motion*. Geraadpleegd op 13 mei 2013 van <http://createdigitalmotion.com/2008/05/processingjs-very-cool-but-javascript-nuts-go-overboard/>
- [19] De Wit, O. (1 juni 2010). Perspective : Easy 3D programming with WPF, *odewit*. Geraadpleegd op 5 maart 2013, van <http://www.odewit.net/ArticleContent.aspx?id=Wpf3DIntro&lang=en>
- [20] (20 februari 2013). Intuitive user centric software van *LMS*. Geraadpleegd op 28 mei 2013 van <http://www.lmssoundbrush.com/product/software>
- [21] Curré, C. (8 september 2008). Telefoon voor doven van *SYNC*. Geraadpleegd op 5 juni 2013 van <http://sync.nl/telefoon-voor-doven/>
- [22] Wong, B. (juni 2011). Color blindness van *Nature Methods*. Geraadpleegd op 27 mei 2013 van <https://my.vanderbilt.edu/neurosciencenews/files/2012/10/CB.pdf>.
- [23] (29 maart 2013). Foto van klanten in bijlage II. Geraadpleegd op 6 juni 2013.
- [24] Aisch, G. (24 november 2011). Say Goodbye to Red-Green Color Scales van *vis4.net*. Geraadpleegd op 2052013 van <http://vis4.net/blog/posts/goodbye-redgreen-scales/>
- [25] Brown, P. (2012). *Silverlight in Action* (1^e druk). Shelter Island, NY: Manning.
- [26] (1 april 2013). What is the difference between Mainstream Support, Extended Support, and online self-help support for software products? van *Microsoft Support Lifecycle Policy FAQ*. Geraadpleegd op 3 mei 2013 van <http://support.microsoft.com/gp/lifepolicy>
- [27] Few, S. (februari 2008). Practical Rules for Using Color in Charts (p. 11) van *Visual Business Intelligence Newsletter*. Geraadpleegd op 15 april 2013 van http://www.perceptualedge.com/articles/visual_business_intelligence/rules_for_using_color.pdf

Nawoord

De afgelopen vijf maanden waren, op zijn zachtst gezegd, plezierig. Het project was erg leuk om aan te werken. Ik heb er ook veel van geleerd: niet alleen op 3D-gebied maar ook op algemene softwareontwikkeling. Zo heb ik C++ geleerd, een taal waar ik nog geen ervaring mee had. Ook ben ik meer te weten gekomen over het converteren van kleuren, color spaces, enzovoort. Ik hoop dit goed toe te kunnen passen binnen mijn hobbyprojecten. De stage heeft mij ook als persoon ontwikkeld: ik kan nu beter plannen, beter met meerdere activiteiten tegelijk bezig zijn, het gebruik van een agenda waarderen, et cetera. Ik kijk daarom ook uit naar de toekomst. Als ik afstudeer kan ik beginnen aan mijn echte carrière bij Sorama en zal ik, als alles goed gaat, trouwen met mijn steun en toeverlaat: Christine (Renkema-)van Woensel.

Ik hoop ook dat u, als lezer, wijzer bent geworden over de werking van 3D en dat u bent overtuigd dat de gemaakte keuzes juist zijn. Daarnaast wil ik u bedanken voor het lezen van dit verslag want dat betekent dat dit werk niet voor niks is geweest.



Bijlagen

I Project Initiative Document

Project Initiatie Document

Sorama BV

Project: 3D Sound Visualization

Andreas Renkema

16 april 2013

Documenthistorie

Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	12/2/2013	Opzet eerste versie in L ^A T _E X.
0.2	Concept	22/2/2013	Verwerking feedback Merijn & Lennart.
0.3	Predefinitief	14/3/2013	Aanpassing communicatieplan en planning.
0.3	Definitief	9/4/2013	Spellingfouten verbeterd en vergeten deelvragen toegevoegd.

Goedkeuring

Versie	Datum goedkeuring	Naam	Functie	Paraaf
0.3	14/3/2013	Merijn de Jonge	Chief software development	

Distributie

Versie	Datum verzending	Naam	Functie
0.1	12/2/2013	Merijn de Jonge	Bedrijfsbegeleider
0.1	12/2/2013	Lennart de Graaf	Docentbegeleider
0.2	22/2/2013	Merijn de Jonge	Bedrijfsbegeleider
0.2	22/2/2013	Lennart de Graaf	Docentbegeleider
0.3	14/3/2013	Merijn de Jonge	Bedrijfsbegeleider
0.3	14/3/2013	Lennart de Graaf	Docentbegeleider

Managementsamenvatting

Doel van dit document

Het doel van dit document is om het project te definiëren waarna het als maatstaaf kan worden gebruikt bij het beoordelen van het project. Zo kan tijdens het verloop van het project worden beoordeeld of de doelstellingen, zoals onderzoek en prototypes, volgens de planning op tijd zijn opgeleverd en of er voldoende tijd over is voor de rest van het project. Ook kan dit document voorkomen dat de opdrachtnemer extra werk krijgt wat niet bij de opdracht hoort.

Aanleiding

Sorama BV is een start-upbedrijf met een zelf ontwikkeld product, namelijk een geluidscamera. Deze camera kan worden gebruikt om akoestische eigenschappen van objecten te meten en geluidsbronnen te lokaliseren. Een voorbeeld hiervan is een harde schijf uit een computer die bij het ronddraaien van de schijven te veel geluid maakte. Het bleek dat de dekplaat resoneerde en door een simpele aanpassing daarvan werd het geluid geëlimineerd. Hierbij wordt een algoritme gebruikt om het signaal terug te calculeren van het meetoppervlak oftewel de camera naar de bron en op elke afstand hiertussen. De beelden die met dit systeem worden gegenereerd zijn nog tweedimensionaal, terwijl de data al drie dimensies bevat aangezien er op elke afstand van de camera data beschikbaar is. Om marktleider te blijven en om te blijven innoveren wil Sorama ook kijken naar de mogelijkheden om driedimensionale beelden te gebruiken om het geluid te visualiseren.

Globale aanpak

De opdracht is om onderzoek uit te voeren naar de mogelijkheden met 3D in de bestaande productomgeving, ook wel de portal genoemd. Als eerste zal de projectleider zichzelf vertrouwd maken met de ontwikkelomgeving en zal een boek gebruiken om te leren programmeren in de omgeving. Daarna worden er verschillende frameworks bekeken om te kijken welke het meeste geschikt is voor het uiteindelijke doel, een driedimensionale geluidsvisualisatie. Hierbij wordt rekening gehouden met criteria zoals snelheid, datainvoer en interactiemogelijkheden. Er wordt naar snelheid gekeken om dat 3D rendering een intensief proces is waarbij veel rekenkracht is vereist. Als de gebruiker een langzamere computer gebruikt zal deze de visualisaties niet goed kunnen weergeven en de hele computer vertragen. Ook is datainvoer belangrijk aangezien de data van de camera op een bepaalde manier beschikbaar wordt gemaakt en het framework compatibel hier mee moet zijn. Als laatste zijn de mogelijkheden van interactie belangrijk, verschillende soorten randapparatuur zoals touchscreen en muis vereisen ook andere manieren van besturing. Met deze kennis worden een drietal visualisaties ontworpen en ontwikkeld waarna deze drie worden getest met gebruikers. Van deze drie wordt de beste gekozen en verder uitgewerkt zodat deze kan worden toegepast in de portal.

Globale kosten en doorlooptijd

De kosten voor dit project zijn alleen maar uren gemaakt door de projectleider en de uren aan hulp verleend door de projectondersteuning. Kosten zoals softwarelicenties voor de ontwikkelomgeving en het gebruik van een computer zijn er niet aangezien deze door de stagiaire zelf zijn aangeleverd. Er zijn nauwelijks kosten en dus ook vrij weinig risico voor Sorama. De doorlooptijd bedraagt 5 maanden, van 4 februari 2013 t/m 30 juni 2013.

Inhoudsopgave

1	Inleiding	6
1.1	Doel van dit document	6
1.2	Opbouw van dit document	6
2	Achtergrond	7
2.1	Organigram	7
3	Projectdefinitie	8
3.1	Projectdoelstellingen	8
3.2	Gekozen oplossing of aanpak	8
3.3	Scope van het project	9
3.4	Producten c.q. eindresultaat	9
3.5	Uitsluitingen	10
3.6	Beperkingen	10
3.7	Afhankelijkheden	10
3.8	Randvoorwaarden	10
3.9	Aannames	11
4	Projectorganisatiestructuur	12
4.1	Opdrachtgever	12
4.2	projectleider	12
4.3	Projectborging	12
4.4	Projectsupport	12
5	Bijlagen	13
A	Communicatieplan	14
B	Initiële Business Case	15
C	Initieel Projectplan	17

1 Inleiding

1.1 Doel van dit document

Het doel van dit document is om het project en de onderdelen daarvan te definiëren. Aan het einde kan dit PID als maatstaaf worden gebruikt bij het verloop van het project. Zo kan er gekeken worden of de kosten niet te hoog opgelopen zijn, of de planning nog actueel en haalbaar is en of de doelstellingen van het project haalbaar zijn. Zo zijn er een paar belangrijke aspecten:

- Wie is verantwoordelijk voor het goede verloop van welk projectdeel?
- Hoe en wanneer zullen de acties die in dit PID besproken zijn worden gerealiseerd?
- Wat als het project te veel werk blijkt te zijn voor de beperkte duur van de stage? Wie neemt het werk dan over?

1.2 Opbouw van dit document

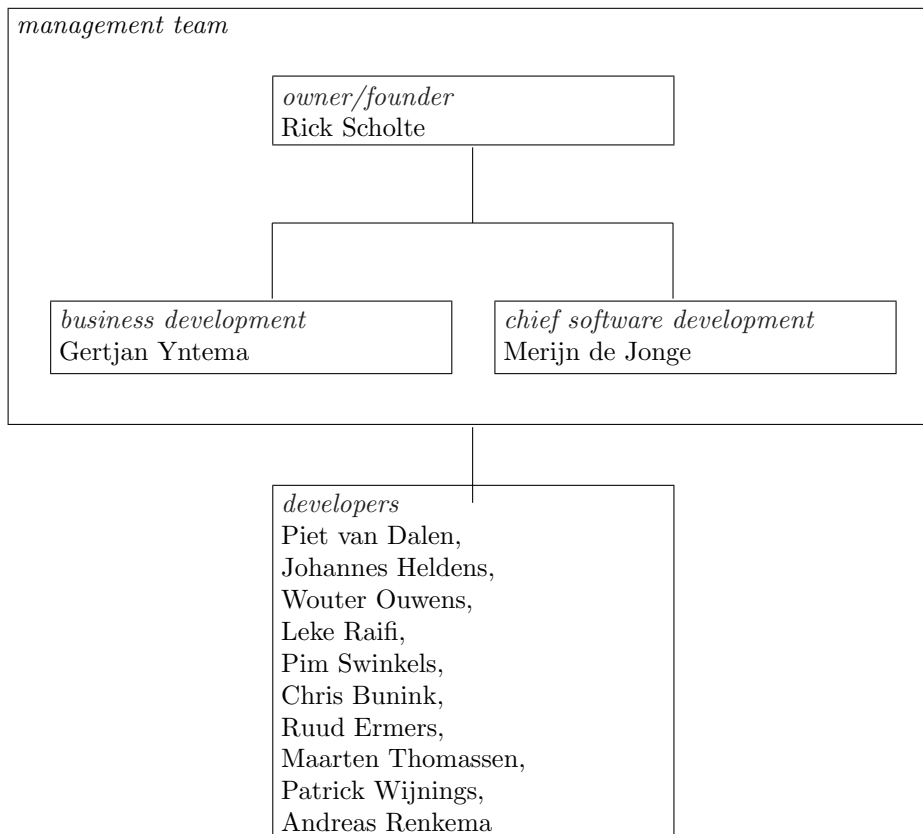
Het opbouw van dit document begint met de achtergrond van het bedrijf en de structuur van de organisatie. Hierop volgend wordt gekeken naar de projectdefinitie waar onder andere de probleemstelling, eindresultaat en uitsluitingen worden besproken. Daarna komt de projectorganisatiestructuur aan bod waar wordt aangegeven wat de relaties zijn tussen mensen in het project. Ten slotte zijn er de bijlagen bestaande uit het communicatieplan, de business case en het projectplan.

2 Achtergrond

Sorama BV is een start-upbedrijf ontstaan vanuit het promotieonderzoek van Rick Scholte, tevens oprichter van het bedrijf. Zijn afstuderen stond in teken van *near-field acoustic holography*, het in kaart brengen van geluiden, waarbij het hoofddoel was om een robot van Philips stiller te maken. Hierbij werd gebruikt gemaakt van een bewegende microfoon om het geluid in kaart te brengen. Na de opdracht kreeg hij toestemming voor een promotieonderzoek waar een prototype van het huidige product werd ontwikkeld en in plaats van één bewegende microfoon kwam er een array van microfoons. Het uiteindelijke resultaat is het product van Sorama, een camera waarmee geluiden worden gevisualiseerd.

Bij Sorama werken 11 mensen waarvan het overgrote deel parttime werkt en de meesten werken in verschillende vakgebieden. Er is geen duidelijke organisatiestructuur aanwezig, afgezien van het management team welke de rest van de werknemers leidt.

2.1 Organigram



3 Projectdefinitie

3.1 Projectdoelstellingen

Sorama heeft een geluidscamera waarmee metingen worden uitgevoerd waarvan de resultaat in een Silverlight-omgeving (ook wel de portal genoemd) worden weergegeven. Hierbij worden verschillende geanimeerde beelden gegenereerd voor de gebruiker met gegevens op een bepaalde afstand van het meetobject. De beelden die worden gegenereerd laten geluidsdruk, -snelheid en -intensiteit zien. Nu willen ze weten wat de mogelijkheden zijn wat betreft 3D en hoe dit kan helpen bij het visualiseren van geluid. Zo kan er worden gedacht aan visualisaties die hetzelfde laten zien als de huidige beelden maar dan met meer interactie zoals de mogelijkheid om in drie dimensies om het gemeten object heen te draaien.

3.2 Gekozen oplossing of aanpak

In dit project wordt er onderzoek gedaan naar 3D en hoe dit kan helpen om geluid te visualiseren. De onderzoeksvraag is “Welke mogelijkheden zijn er om driedimensionale geluidvisualisaties te ontwikkelen in Silverlight?” met de volgende deelvragen:

- Wat voor soorten visualisaties van vergelijkbare data bestaan al?
- Wat is de meest efficiënte techniek om driedimensionale beelden te genereren als het aankomt op rekencapaciteit?
- Welke conventies bestaan er al om de visualisaties het meest toegankelijk te maken?
- Welke wijzen van besturing kunnen worden toegepast in een interactieve visualisatie zodat het voor een beginnende gebruiker begrijpelijk is?

Het onderzoek is verdeeld in verschillende fases die hierna volgend verder worden gespecificeerd.

In de eerste fase zal de projectleider zichzelf bekend maken met de ontwikkelomgeving van Silverlight en C# en de werking van de portal. Silverlight en C# zijn gekozen omdat dit in de portal wordt gebruikt en de ontwikkelde visualisatie moet worden toegepast in diezelfde portal. Hij gebruikt hierbij het boek *Silverlight 5 in Action*¹ om Silverlight te leren kennen zodat hij zelf een project kan opzetten, ontwikkelen en compileren. Ook zal hij kennis maken met de portal om te weten hoe de metingen worden uitgevoerd en gevisualiseerd zodat er een beter idee is hoe de visualisaties moeten werken.

Zodra deze oriëntatie is voltooid zal in de tweede fase onderzoek worden gedaan naar verschillende frameworks waarmee 3D kan worden geprogrammeerd omdat niet alle functionaliteit hetzelfde is en de support van de vooraf gekozen afloopt. XNA was bijvoorbeeld degene waarmee de visualisaties zou worden

¹Pete Brown, *Silverlight 5 in Action* (Shelter Island NY: Manning Publications Co.)

geprogrammeerd, echter is het zo dat Microsoft de ontwikkeling heeft beëindigd en support eindigt april 2014.²

Gezien Sorama mogelijk lang gebruik wil maken van 3D zal er dus een alternatief worden onderzocht en gekozen. Hierbij wordt gekeken of het actief wordt ontwikkeld, hoeveel support er voor is, het gebruik door anderen ontwikkelaars in producten, mogelijkheden wat betreft 3D, de licentie om het framework te gebruiken in een commercieel product en prestaties. Om het beste framework te kiezen zal er een serie van tests worden uitgevoerd, zoals welke het snelste is met dezelfde animatie geprogrammeerd. Ook wordt er gekeken naar de stabiliteit, bijvoorbeeld door de animatie te laten draaien om te kijken welke na een bepaalde tijd nog steeds naar behoren presteert. Op deze manier wordt het beste framework gekozen.

In de derde en laatste fase zal met dit framework een aantal prototypes van visualisaties worden gerealiseerd. In het begin zal er een brainstorm plaats vinden met een aantal mensen om concepten te bedenken. Deze concepten worden gebruikt om een aantal prototypes met het gekozen framework te realiseren. Deze prototypes zullen qua uiterlijk nog niet perfect zijn maar wel functioneel volledig. De prototypes kunnen dan worden getest door gebruikers zodat ze nog verder kunnen worden verbeterd. Uiteindelijk wordt er een van deze prototypes uitgekozen en uitgewerkt tot eindproduct. Deze visualisatie zal dan in de portal worden toegepast in dezelfde stijl als de bestaande visualisaties.

3.3 Scope van het project

De nieuwe visualisatie zal er zijn om klanten beter het product te laten gebruiken en de resultaten van metingen te interpreteren. Het project gaat dus ook puur over het ontwikkelen en realiseren van de visualisatie. De andere werknemers van Sorama zullen vrij weinig hinder ondervinden van het project aangezien zij alleen maar betrokken zijn bij het project voor hulp wanneer dit nodig mocht zijn. Ook zijn zij niet van dit project afhankelijk om hun werk verder uit te voeren. Zij hoeven dan ook geen voorrang te geven boven andere projecten. Mocht de managementteam het noodzakelijk vinden om dit project een hogere prioriteit geven is dit mogelijk.

3.4 Producten c.q. eindresultaat

Als eerste product is er de scriptie welke het onderzoek en de onderzoeksresultaten omschrijft. Deze onderzoeksresultaten zijn onder andere aanbevelingen voor 3D-framework en wat voor soort visualisatie het beste werkt. Het tweede product zijn de drie gemaakte prototypes inclusief het uitgewerkte eindproduct, de visualisaties die in de portal wordt toegepast. Deze animatie zal door de gebruiker ruimtelijk te bekijken zijn door er om heen te draaien en in of uit te zoomen. Daarnaast zal de animatie in verschillende snelheden af te spelen zijn en te pauzeren wanneer gewenst. Ook zal het mogelijk zijn de verschillende sets van data te kiezen waaronder geluidsdruk, -intensiteit en -snelheid. Deze

² Hruska, J. (1 februari 2013). Microsoft kills Xbox 360/PC cross-platform development, declares DirectX “no longer evolving”. *ExtremeTech*. Verkregen op 18 februari 2013 van <http://www.extremetech.com/gaming/147289-microsoft-kills-xbox-360pc-cross-platform-development-declares-directx-no-longer-evolving>

visualisatie zal ook een uitgebreidere gebruiksinteractie hebben dan de huidige visualisaties, namelijk op het gebied van zoomen en ronddraaien.

3.5 Uitsluitingen

- Onderhoud en updates van product.
- Alles wat buiten de portal en de animaties ligt zoals hardware.
- Aanpassingen aan de al bestaande tweedimensionale animaties.
- Klantencontact.

3.6 Beperkingen

Beschikbaar Budget	Beschikbare resources	Gewenste opleverdata	Beschikbare Doorlooptijd
685 uur	Developmentteam van Sorama	30 juni 2013	18 weken

3.7 Afhankelijkheden

Er zijn een aantal afhankelijkheden in het project, waaronder software en hardware.

- Specifieke benodigde software:
 - Windows 7
 - Visual Studio 2012
 - Silverlight 5 Toolkit
 - XNA Gamestudio 4.0
 - MiKTeX
 - Git
- Hardware:
 - Computer met bovenstaande software.
 - Extern scherm.
 - Backupcomputer mocht eerste computer qua prestaties niet voldoen.

3.8 Randvoorwaarden

Er zijn geen randvoorwaarden naast een beschikbare werkplek met voorgaand besproken computer.

3.9 Aannames

Er wordt aangenomen dat er verschillende sets van meetgegevens beschikbaar zijn of dat deze gemaakt kunnen worden met metingen om de visualisaties op te ontwikkelen en te testen. Ook wordt er aangenomen dat er een werkplek aanwezig is om te werken en dat er literatuur gerelateerd aan de opdracht voor handen is. Ook wordt aangenomen dat er mensen bereid zijn om de prototypes te testen en hun bevindingen kenbaar te maken.

4 Projectorganisatiestructuur

4.1 Opdrachtgever

Sorاما is de opdrachtgever met Merijn de Jonge als vertegenwoordiger. Hij levert informatie over wat er moet worden opgeleverd, eisen en feedback op resultaten. Ook is hij degene aan wie het eindproduct wordt opgeleverd.

4.2 projectleider

Andreas Renkema is als projectleider, opdrachtnemer en enige projectlid bezig met de uitvoering van het project. Hij stelt het PID op, is de voornaamste persoon als het aankomt op contact met de opdrachtgever en beheerst de voortgang van het project. Hij houdt bij of hij het werk in de gegeven tijd kan maken en zo niet zal hij de hulp inroepen van anderen. Mocht er iemand zijn gevraagd om feedback te geven let er hij op dat dit snel wordt ontvangen zodat de oplevering geen vertraging oploopt.

4.3 Projectborging

Merijn de Jonge is verantwoordelijk voor projectborging. Hij controleert onder andere of elke sprint alle taken zijn afgerond, of de projectleider voldoende voortgang maakt door middel van een mondelinge controle bij de dagelijkse meeting en door aanwijzingen te geven zoals boeken om Silverlight te leren. Ook is hij beschikbaar om vragen te stellen over het project zoals de manier waarop data van de camera wordt verwerkt.

Lennart de Graaf is verantwoordelijk voor de borging wat betreft de scriptie. Hij meldt het er als er onderdelen blijken te missen en geeft feedback op de scriptie.

4.4 Projectsupport

Merijn, Maarten, Wouter en Patrick zijn developers bij Sorاما, deze kunnen Andreas ondersteunen waar nodig bij het ontwikkelen van de 3D-visualisaties. Zo kunnen ze bijvoorbeeld helpen met vinden van oplossingen voor problemen en aanwijzingen geven.

5 Bijlagen

A Communicatieplan

Inleiding

Dit communicatieplan benoemt alle partijen die een (positief of negatief) belang hebben bij het project en de wijze waarop zij bij het project zullen worden betrokken en welke communicatievormen daarbij gebruikt worden. Het gaat hierbij om partijen en communicatie buiten de formele projectmanagementstructuur zoals beschreven in het PID.

Belanghebbenden bij het project

Wie	Namens	Belang	Communicatievorm
Opdrachtgever	Sorama	Product	Email, mondeling
Projectsupport	Projectleider	Advies	Email, mondeling
Docentbegeleider	Projectleider	Feedback	Email, mondeling

Communicatiekanalen

Van	Naar	Informatie	Medium	Frequentie van data
Projectleider	Projectsupport	Status, vragen	Email, mondeling	Dagelijks
Projectsupport	Projectleider	Advies	Email, mondeling	Als nodig
Opdrachtgever	Projectleider	Feedback, opdracht	Email, mondeling	Wekelijks
Projectleider	Opdrachtgever	Status, vragen	Email, mondeling	Wekelijks
Projectborging	Projectleider	Status	Email, mondeling	Dagelijks
Projectleider	Projectborging	Status	Email, mondeling	Dagelijks
Projectleider	Docentbegeleider	Conceptversies van eindproducten	Voornamelijk email	Als nodig
Docentbegeleider	Projectleider	Feedback	Voornamelijk email	Als nodig

B Initiële Business Case

Inleiding

Dit document bevat de overwegingen om het huidige project te starten. Het vormt de rechtvaardiging van het project en zal om deze reden worden beoordeeld door de stuurgroep. De onderbouwing van het project zal regelmatig worden geëvalueerd op basis van deze business case. Het beschrijft de verwachte te investeren kosten ten opzichte van de verwachte voordelen en besparingen en de projectrisico's. De kosten -baten afweging wordt onderbouwd middels een investeringsvoorstel. De Business Case begint met een beschrijving van de gemaakte afwegingen voor het project.

Redenen

Sorama is ontstaan vanuit het product van Rick Scholte namelijk de geluidscamera. Dit is ook het enige product van Sorama dus de inzet is zeer hoog. Om voor te blijven op de concurrentie met het bedrijf zal Sorama dus ook constant moeten innoveren met het product. Als ze dit niet doen bestaat de mogelijkheid dat er een concurrent met een soortgelijk product die in sommige opzichten beter zou kunnen zijn dan het product van Sorama waardoor klanten naar de concurrent stappen met hun problemen. Ook door de keuzen om te blijven innoveren zullen concurrerende bedrijven een steeds grotere inhaalslag moeten maken waardoor ze minder snel zullen proberen te concurreren met Sorama.

Voordelen

Voordelen voor Sorama als het project ten einde is, is dat Sorama een beter en gemakkelijker te gebruiken product heeft. Hierdoor zullen klanten met positieve ervaringen met de camera sneller terugkeren naar Sorama. Ook kunnen de nieuwe visualisaties worden gebruikt in reclame zodat het nog duidelijker wordt voor de buitenwereld hoe geavanceerd het product eigenlijk is.

Tijd en kosten

Het project zal ongeveer 18 weken duren. De kosten hiervan zijn het loon voor de projectleider en de uren die de projectsupport kwijt zal zijn aan ondersteuning alhoewel dit zeer gering is. Het loon bedraagt 300 euro per maand voor 5 maanden, 1500 euro totaal aan het eind van het project. Omdat de gebruikte licenties voor software zoals Visual Studio door de stagiaire zijn aangeleverd kost Sorama dit niets.

Risico's

Er is weinig risico voor Sorama gezien de lage kosten en mocht het project mislukken is dat vervelend maar brengt het het product op zich niet in gevaar.

Investeringsvoorstel

De investering die wordt gemaakt is grotendeels buiten het normale bedrijfsproces om afgezien van de kosten en een enkele bezoek van de bedrijfsbegeleider. Als het project is afgelopen zal Sorama de investering makkelijk kunnen terug verdienen door een waardevoller product en een grotere voorsprong op de concurrentie te hebben.

C Initieel Projectplan

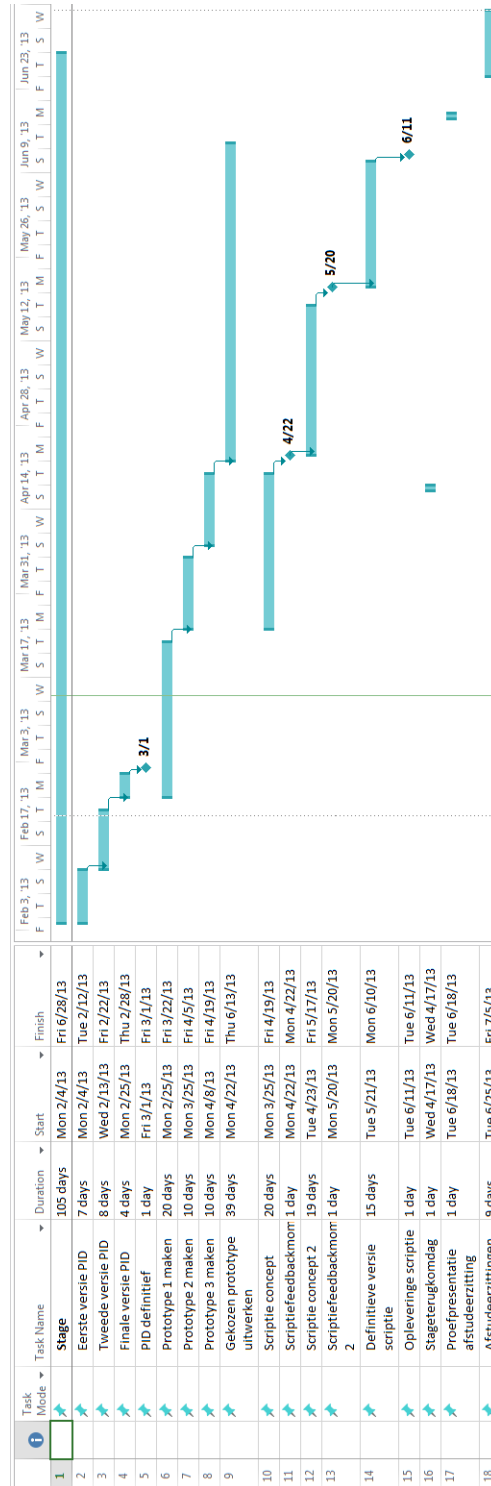
Productdecompositiestructuur

- Algemeen
 - PID
 - Projectplan
 - Businesscase
- Visualisatie
 - Brainstorm
 - Schetsen en concepten
 - Visualisatieprototype 1
 - Visualisatieprototype 2
 - Visualisatieprototype 3
 - Uitgewerkte visualisatie
- Documentatie
 - Verantwoording keuze framework
 - Documentatie van code

Urenverdeling

Als er meer dan 1 taak tegelijk wordt uitgevoerd zal dit resulteren dat de uren als volgt worden verdeeld: 1 dag (8 uur) in de week voor de scriptie en 4 dagen (32 uur) per week voor de opdracht zelf.

Planning



II Geslacht van klanten



Figuur 20: Bijna alle klanten van Sorama, maart 2013

Zoals in bovenstaande foto te zien is, is er maar één vrouw tussen de negentien mannen (uitgezonderd twee Sorama medewerkers). Hieruit wordt geconcludeerd dat een overgrote meerderheid van de klantenbasis ook man is.

III Framework keuze

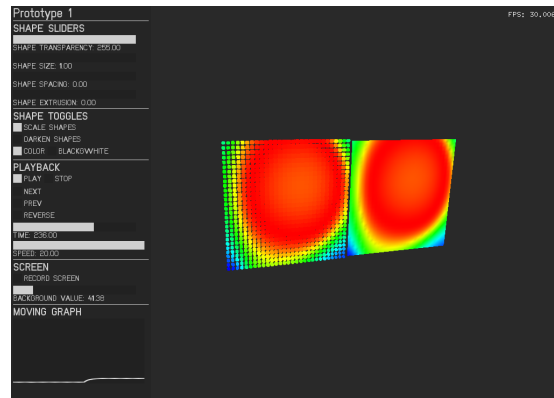
Hier staan de punten waar rekening mee wordt gehouden bij het kiezen van een alternatief voor XNA. Zo is het belangrijk dat het ontwikkelen van de visualisatie in dezelfde omgeving kan als de portal, namelijk Visual Studio 2012. Ook is het natuurlijk belangrijk dat de visualisatie online moet kunnen staan aangezien de portal op het web draait. Daarnaast zijn er een aantal functies belangrijk om er zeker van te zijn dat het framework breed inzetbaar is, zoals index en vertex buffers.

Feature	XNA	Perspective 3D	Silverlight WebGL	openFrame- works	Processing
Silverlight	Ja	Ja	Nee	Nee	Nee
VS2012	Ja	Ja	Ja	Nee	Nee
Online	Ja	Ja	Ja	Nee	Ja
Controls	Ja	Ja	Ja	Ja	Ja
Buffers	Ja	Nee	Ja	Ja	Ja
Snelheid	Hoog	Redelijk	Hoog	Hoog	Laag
Gemak	Hoog	Hoog	Laag	Hoog	Hoog
Uitbreidbaar	Ja	Nee	Ja	Ja	Ja
Stabiliteit	Hoog	Hoog	Laag	Hoog	Hoog
Support	Fora + Microsoft	Nee	Nee	Fora	Fora
Debugging	Ja	Ja	Nee	Ja	Nee

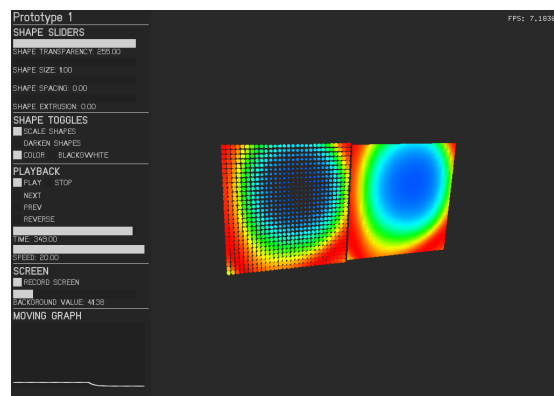
Framework	Voordelen	Nadelen
ANX Framework	Heeft een projectconverter om bestaande projecten om te bouwen.	Werkte niet, kon ANX.Framework namespace niet vinden en veel errors kwamen daarvoor al.
	Standaard template van ANX gaf de melding dat Windows 8 nodig was.	Source-compatible replacement van XNA Framework, omzetten van huidig werk is dus zeer makkelijk.
Monogame	Werkte in 1 keer met installer zonder problemen, kon alleen niet direct de templates vinden.	
	Sample werkte gewoon, gaf een blauwe achtergrond zoals in XNA maar dan zonder XNA.	
		Niet zeker of het binnen Silverlight zal werken.
		Werkt toch met XNA.
		Kan niks werkends krijgen anders dan de eerste sample.
Grommet		Alleen maar 2d en input.
		Is eigenlijk alleen voor .NET Micro Framework.
MonoXNA	Wordt gebruikt door Monogame,	laatste update was eind 2009 ongeveer.
openFrameworks	C++ is zeer snel.	
	Van alles beschikbaar, 3D, Audio, Video, etc.	
		Lowlevel, nog niet uitgetest.
	Heeft simpele vertices and indices	
	Samples werken tot nu toe in Code::Blocks met of 0.7.4	
	Werkt heel goed, echt heel goed.	
	2x zo snel als Silverlight + XNA	
		geen integratie met .NET
	misschien een mogelijkheid om een stream te maken in Silverlight die de serverapplicatie in openframeworks aanroept ofzo?	
	Doet precies alles wat ik wil, zelfs transparency.	
	Uitbreidbaar.	60

Framework	Voordelen	Nadelen
Silverlight-WebGL	Create 3D web browser applications using Silverlight plugin on Windows and Mac OSX.	Cannot use as a Silverlight Control and mix with other Silverlight controls.
	Easily port existing WebGL JavaScript code to Silverlight.	Cannot hide Silverlight control, it must be visible somewhere even if only 1x1 pixels.
	Enable C# developers to put cross platform 3D applications in a web browser.	Cannot be used for out-of-browser 3D applications.
		Cannot show file:// URL based textures in IE9 or Chrome, but Firefox works OK.
	Werkt tot nu toe in Firefox inclusief animatie.	Debugging & breakpoints alleen in Internet Explorer
		Internet Explorer heeft IEWebGL plugin nodig wat weer iets met Chrome heeft.
		Chrome kon de animatie niet weergeven dus werkte het ook niet in Internet Explorer, alleen Firefox.
		Heel lowlevel, duurt eeuwen voordat je echt een animatie hebt gemaakt die nuttig is
		Geen mogelijkheid gevonden tot input
Processing	Heel erg simpel	Java
		Kan geen indexed vertex buffer vinden.
		Langzaam
		geen integratie met .NET
Perspective 3D	Werkt direct in Silverlight	
	Gebruikt WPF 3D	
		Geen eigen modellen aan te maken
		Ietwat langzaam

IV openFrameworks performance



Figuur 21: Prototype 1, geen frames aan het opslaan.



Figuur 22: Prototype 1, frames aan het opslaan.

Bij figuur 21 wordt er een snelheid van ongeveer 30FPS gemeten, terwijl bij 22 de snelheid rond de 7FPS is. De snelheid is dus uiteindelijk ruim 4 keer zo laag.