

Geïntegreerde Proef:

Virtual Reality

GIP - EE

Interne Mentoren:

Mevrouw H. De Muelenaere
De heer K. Gobbin



VTI Tielt
Grote Hulststraat 28
8700 Tielt

051 40 05 68
vti@molenland.be
vtitielt.molenland.be

Woord vooraf

Om het diploma van elektriciteit-elektronica te behalen, is het een vereiste om een geïntegreerde proef te maken. De GIP maakt deel uit van het Vlaamse examenprogramma en laat een leerling bewijzen dat hij zelfstandig een opdracht kan maken in zijn vakgebied. De geïntegreerde proef wordt dan ook beoordeeld door een interne jury, waar vooral leerkrachten in zitten en een externe jury die bestaat uit mensen van buiten de school.

Voor mijn GIP heb ik de wereld van VR (Virtual Reality) bestudeerd. Er zijn echter weinig mensen binnen de school die expertise hebben in VR. Mijn opdracht bestond dan vooral uit het bestuderen, onderzoeken en integreren van VR in de verschillende vak-domeinen. Er is ook specifiek gevraagd om uit te zoeken als het mogelijk is om via een virtuele actie, een actie in de reële wereld te krijgen. Met deze nieuwe technologie hoop ik dat we de leerervaring van de leerlingen beter kunnen maken. Ik was zeer blij met deze uitdaging.

Tot slot wil ik alle mensen zoals ouders en leerkrachten die mij geholpen hebben bij het realiseren van mijn GIP bedanken om mij te steunen en te helpen.

Als laatste wil ik u, de lezer van mijn GIP boek bedanken voor het lezen en de interesse.

Inleiding

Nederlands

Voor mijn geïntegreerde proef (GIP) bestudeerde ik de technologie van Virtual Reality (VR). Het doel van mijn opdracht is om te bekijken wat een 3^{de} graad EE kan met een VR headset. Voor mijn GIP heb ik de Oculus Quest 2 VR headset gebruiken. Hierbij onderzoek ik:

- Hoe we de headset kunnen koppelen aan een computer of smartphone.
- De mogelijkheid om zelf een virtuele wereld te maken en deze te gebruiken.
- Of we bij een virtuele actie, een actie in de echte wereld kunnen maken of omgekeerd.
- In welke vak domeinen kunnen we VR integreren.
- Welke andere leuke toepassingen te vinden zijn.

Daarna ga ik evalueren wat er haalbaar is in een derde graad elektriciteit-elektronica en wat niet.

Français

Pour mon test intégré (GIP) j'ai étudié la technologie de la Réalité Virtuelle (VR). Le but de ma mission est de voir ce qu'un EE 3ème degré peut faire avec un casque VR. Pour mon GIP j'ai utilisé le casque Oculus Quest 2 VR. je recherche:

- Comment nous pouvons connecter le casque à un ordinateur ou un smartphone.
- La possibilité de créer et d'utiliser votre propre monde virtuel.
- Si nous pouvons créer une action dans le monde réel avec une action virtuelle ou vice versa.
- Dans quels domaines nous pouvons intégrer la VR.
- Quelles autres applications amusantes peuvent être trouvées.

Ensuite, j'évaluerai ce qui est faisable dans un troisième degré d'électronique électrique et ce qui ne l'est pas.

English

For my integrated test (GIP) I will study the technology of virtual reality (VR). The purpose of my assignment is to see what a 3rd degree EE can do with a VR headset. For my GIP I will be using the Oculus Quest 2 VR headset. I'm investigating:

- How we can connect the headset to a computer or smartphone.
- The ability to create and use your own virtual world.
- Whether we can make a virtual action, an action in the real world or vice versa.
- In which subject domains we can integrate VR.
- What other fun applications can be found.

Then I will evaluate what is possible in a third degree electrical electronics and what is not.

Inhoud

Woord vooraf.....	1
Inleiding	2
Nederlands	2
Français.....	3
English	4
Inhoud	5
1 Doelstellingen	7
2 Procesoverzicht.....	8
2.1 Onderzoek VR.....	8
2.2 Instellen Oculus Quest 2.....	9
2.3 Virtuele wereld verkennen.....	9
2.4 Ontwikkelsoftware zoeken.....	9
2.5 Maken van een eigen VR game	9
2.6 Info uit de virtuele wereld naar buiten brengen.....	9
2.7 Besluiten.....	9
3 Planning.....	10
4 Soorten virtuele technologieën onderzoeken	11
4.1 XR (Extended Reality)	11
4.2 VR (Virtual Reality)	11
4.3 AR (Augmented Reality)	12
4.4 MR (Mixed Reality).....	12
5 Oculus Quest 2.....	13
5.1 Specificaties.....	13
5.2 Bediening.....	13
5.3 Belangrijke Info.....	14
5.4 Keuze verantwoording	14
6 Instellingen	15
6.1 Eerste gebruik.....	15
6.2 Connecteren met de smartphone.....	15
6.3 spelersruimte of de guardian	15

7	Virtuele wereld verkennen.....	16
7.1	Toolbar	16
7.2	Environments	17
7.3	Experimental Features	17
7.4	Hand Tracking.....	17
7.5	Passthrough.....	18
8	Oculus wereld als developer.	18
8.1	Oculus Developer Hub downloaden.....	18
8.2	Developer account aanmaken.....	19
8.3	Developer Mode activeren.....	19
8.4	Verbinden met de ODH	20
8.5	ODH toepassingen	20
9	Nieuwe VR wereld in Unity	21
9.1	Installatie Unity	21
9.2	Algemene setup.....	22
9.3	VR-Rig	31
10	Oculus Link.....	32
10.1	Pc vereisten	32
10.2	Setup.....	32
10.3	Toolbar	33
10.4	SteamVR	33
11	Realisaties.....	34
11.1	Handen	34
11.2	Teleportatie.....	35
11.3	Connectie met de buitenwereld.....	36
11.4	Wegwijs in de virtuele wereld	38
11.5	Beeldmateriaal	38
12	Besluiten	39
13	Prijslijst.....	40
14	Begrippenlijst.....	41
15	Figuurlijst	42
16	Programma's.....	44

1 Doelstellingen

Bij een GIP horen natuurlijk ook doelstellingen. Het algemene doel is om te bekijken wat een 3^{de} graad EE kan met een VR headset.

Subdoelen:

- Onderzoek naar VR.
- Alle functies van de Quest 2 onderzoeken.
- Headset koppelen aan een computer of smartphone.
- Zelf een virtuele wereld maken en deze gebruiken.
- Bij een virtuele actie, een actie in de echte wereld realiseren of omgekeerd.
- Echte handen gebruiken in plaats van de controllers.
- Oculus link gebruiken.
- Extra (indien tijd over): de passthrough functie van de oculus gebruiken om AR te verkrijgen.

2 Procesoverzicht



2.1 Onderzoek VR

Onderzoek naar Virtual Reality en andere reality systemen zoals Extended Reality, Augmented Reality, ... en kijken wat deze allemaal inhouden.

2.2 Instellen Oculus Quest 2

Hierbij kan het volgende worden ingesteld.

- wifi netwerk,
- spelers ruimte,
- vloerniveau,
- oculus app,
- oculus software,
- extra hardware.

2.3 Virtuele wereld verkennen

Verkennen van:

- Oculus home omgeving.
- Oculus functies.
- Andere virtuele werelden.

2.4 Ontwikkelsoftware zoeken

Onderzoeken naar:

- Verbinding maken met de computer.
- Welke specificaties je nodig hebt.
- Verschillende software en die gaan beoordelen.

2.5 Maken van een eigen VR game

Hier zou ik willen onderzoeken als het mogelijk is om zelf een virtuele wereld te maken en deze te uploaden naar de Oculus. Bij deze wereld wil ik ook onderzoeken als het mogelijk is om dingen vast te nemen of te verplaatsen. Als ik een actie uitvoer dat er dan ook iets gebeurt bv drukken op een schakelaar en dat het licht dan aan gaat.

2.6 Info uit de virtuele wereld naar buiten brengen

In deze stap wil ik proberen verder te bouwen op de vorige stap. Ik zou bv: in de virtuele wereld op een schakelaar drukken en dat er een lichtje brandt in de echte wereld of bij een actie de ledstrip laten veranderen van kleur en kijken wat nog allemaal mogelijk is.

2.7 Besluiten

Op het einde ga evalueren wat haalbaar is om in een derde graad EE te kunnen realiseren. Dit ga ik vooral doen op basis van mijn eigen ervaring en de tijd die ik er per onderwerp heb ingestoken.

3 Planning

Een goede planning is zeer belangrijk in zo'n groot proces. Er zijn bepaalde deadlines die moeten gehaald worden. Ik heb hieronder even de belangrijkste deadlines opgesomd.

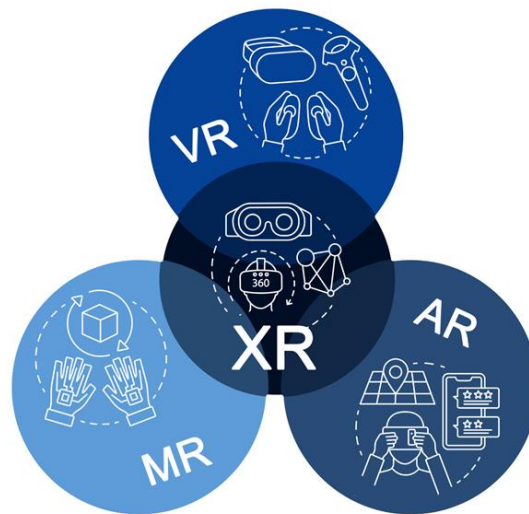
Datum:	Activiteit
7 okt.	Oudercontact
14 okt.	Eerste contact (enkel interne jury)
23 dec.	GIP-rapport
15 mrt.	Tweede voorstelling (met externe mentoren)
1 apr.	GIP-rapport (eventueel laatste - schriftelijke - waarschuwing)
22 apr.	Het volledige werk indienen ter controle
5 mei	Alle leerlingen krijgen hun werk terug
9 mei	Laatste controle
23 mei	Inleveren werk (3 ingebonden exemplaren + digitale versie)
1 juni	Eindvoorstelling

Naast deze deadlines heb ik ook zelf doelen opgezet zodat ik naar het einde toe niet met grote tijdsproblemen zit. Dit zijn mijn doelen:

- sep/okt → instellen en verkennen van de Oculus Quest 2
- nov/dec → maken van een eigen virtuele wereld
- jan/feb → het naar buiten brengen van informatie
- mrt/apr → nog aanpassingen brengen of andere dingen uittesten en besluiten formuleren

Er zal ook een logboek beschikbaar zijn op de Office 365 app planner, met daarin een gedetailleerde beschrijving.

4 Soorten virtuele technologieën onderzoeken



Figuur 1 Soorten virtuele technologieën

4.1 XR (Extended Reality)

Extended Reality (XR) is een term die verwijst naar alle gecombineerde echte en virtuele omgevingen die door de mens-computer-interacties geproduceerd wordt. Dit doormiddel van computertechnologie en draagbare apparaten. Het omvat bijvoorbeeld representaties zoals Augmented Reality (AR), Mixed Reality (MR) en Virtual Reality (VR) en regio's die daartussen schommelen.

XR is een overkoepelende term die het bereik van "volledig echt" tot "volledig virtueel" omvat. Maar de technologie ervan ligt in de uitbreiding van de menselijke ervaring, vooral in de zin van aanwezigheid (vertegenwoordigd door VR) en het verwerven van technologie (vertegenwoordigd door AR). Met de voortdurende ontwikkeling van mens-computerinteractie, evolueert deze technologie ook voortdurend.

4.2 VR (Virtual Reality)

Virtual Reality (VR) is een gesimuleerde ervaring die vergelijkbaar of compleet anders kan zijn dan de echte wereld. Toepassingen van Virtual Reality zijn onder meer entertainment (vooral videogames), onderwijs (zoals medische of militaire training) en zaken (zoals virtuele vergaderingen).

Momenteel gebruiken standaard Virtual Reality-systemen Virtual Reality-headsets of multi-projectie-omgevingen om realistische beelden, geluiden en andere sensaties te genereren om de daadwerkelijke aanwezigheid van de gebruiker in de virtuele omgeving te simuleren. Een persoon die een virtual reality-apparaat gebruikt, kan rondkijken in de kunstmatige wereld, eromheen bewegen en communiceren met virtuele functies of objecten. Dit effect wordt meestal geproduceerd door een VR headset, die bestaat uit een op het hoofd gemonteerd display en een klein scherm.

4.3 AR (Augmented Reality)

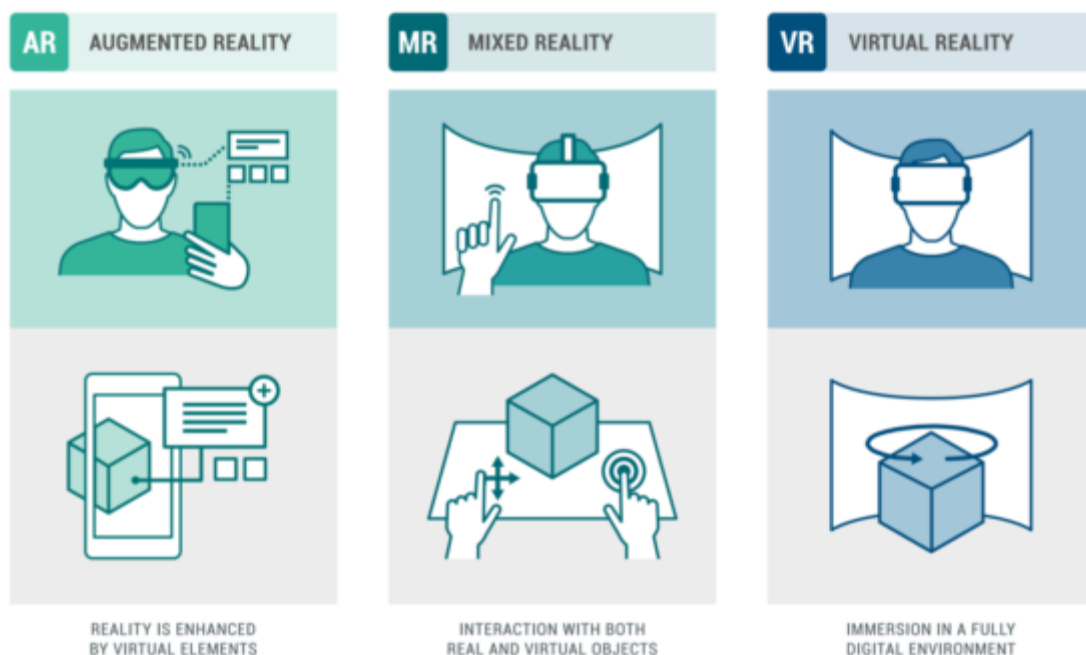
Augmented Reality (AR) is een interactieve ervaring van de echte wereld waarin objecten in de echte wereld worden aangevuld met door de computer gegenereerde perceptuele informatie. AR kan worden gedefinieerd als een systeem dat drie basisfuncties omvat: de integratie van echte en virtuele werelden, realtime interactie en nauwkeurige 3D-registratie van virtuele en echte objecten.

De primaire waarde van Augmented Reality is de manier waarop componenten van de digitale wereld worden geïntegreerd in de perceptie van mensen van de echte wereld, niet als een eenvoudige weergave van gegevens, maar door de integratie van meeslepende sensaties die worden gezien als een natuurlijk onderdeel van de omgeving.

4.4 MR (Mixed Reality)

Mixed Reality (MR) is een technologie die de scheidslijn tussen de fysieke en virtuele wereld doet vervagen. De omgeving en alles daarin wordt in kaart gebracht. De digitale objecten worden vervolgens in de omgeving geprojecteerd. Deze digitale objecten hebben niet alleen interactie met de omgeving, maar ook met elkaar. Met MR-brillen zoals Hololens of Magic Leap kunnen gebruikers ook hun handen vrij houden. Mixed Reality toont 100% digitale objecten in de echte wereld waarmee gebruikers kunnen communiceren. Dus Mixed Reality is een hybride van VR en AR.

Het bijzondere aan deze technologie is dat virtuele objecten en data vastliggen in de echte wereld. Mixed Reality combineert technologieën uit VR en AR. Mixed Reality wordt niet alleen ingezet om efficiënter en slimmer te werken, maar ook om marketing en sales te stimuleren



Figuur 2 AR/VR/MR

5 Oculus Quest 2

5.1 Specificaties

De Oculus Quest 2 is een VR headset die je ook standalone kan gebruiken. Dit wil zeggen dat je er geen computer voor nodig hebt om spelletjes te spelen, hij moet ook niet verbonden worden met een kabel. Dit zorgt ervoor dat het gebruik zeer gemakkelijk en comfortabel is. Je kan deze headset ook gebruiken als een developer. In dit geval is het hebben van een goede computer met een aantal eisen nodig. Dit zijn de specificaties van de Quest 2:

- 6 GB RAM
- 128 GB geheugenopslag
- Met of zonder kabel
- 4 camera's
- 2 controllers

5.2 Bediening

Headset:

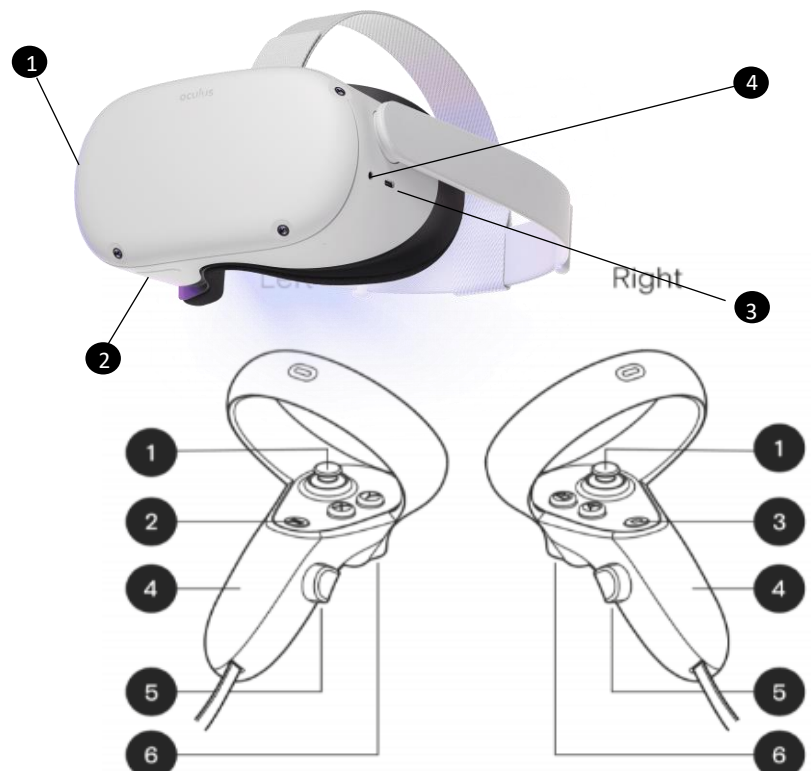
1. Aan/uit knop
2. Volume knop +/-
3. USB C Input opladen/data
4. Audio uitgang

Controllers:

1. Joystick
2. Menu knop
3. Oculus knop
4. Batterij klepje
5. Grip knop
6. Trigger



Figuur 3 Oculus Quest 2 VR Bril



5.3 Belangrijke Info

Bij het gebruik van de Quest 2 hou je het best rekening met:

- Zonlicht kan permanente schade aan de lenzen veroorzaken, hou de headset daarom uit de buurt van ramen en gebruik hem niet buiten.
- Gebruik geen alcoholische doekjes om de headset schoon te maken, omdat dit de headset kan beschadigen. In plaats daarvan kunt u niet-alcoholische, antibacteriële doekjes gebruiken.
- Bij een fabriek reset worden alle games en instellingen verwijderd.
- Laat de controllers niet vallen en gooi ze niet tegen de grond of tegen de muur.

5.4 Keuze verantwoording

Ik koos voor de Oculus Quest 2 omdat hij de volgende kenmerken heeft:

- werkt standalone;
- geen extra sensoren;
- hand tracking;
- goedkoop;
- goede kwaliteit;
- veel info over terug te vinden;
- developer mode.



Figuur 4 Oculus Quest 2

6 Instellingen

6.1 Eerste gebruik.

Voor de installatie van de Oculus Quest 2 moeten we eerst gedurende een tweetal seconden op de aan-knop drukken tot je het logo van Oculus ziet verschijnen. Daarna volgen we de stappen die op het scherm komen.



6.2 Connecteren met de smartphone.

Maak de verbinding met je telefoon door de Oculus app te installeren in de app store. Open de app en log je in met je Facebook account. Ga naar apparaten en druk op “nieuw apparaat koppelen” kies dan voor de Oculus Quest 2 en druk op doorgaan. Wacht tot de update gedaan is en druk vervolgens op ok. Zet je Headset terug op en druk op continue.

6.3 Spelersruimte of de guardian

Eerst gaan we het vloerniveau gaan instellen om daarna te kiezen tussen een stationary of een roomscale boundary. Bij de roomscale boundary kan je grote ruimtes gaan gebruiken. Bij stationary boundary is het de bedoeling dat je neerzit.

Vloerniveau:

Zet het vloerniveau goed door recht te staan, naar beneden te kijken en te zorgen dat de kruisjes op de grond liggen. Is dit niet het geval ga met je controller naar de grond tot de kruisjes op de grond zijn en druk dan op een willekeurige knop, druk daarna op continue.

Roomscale boundary:

Bij de roomscale boundary kies je hoe groot je spelers oppervlakte is door dit te tekenen. Daarvoor richt je de controller naar de grond, druk je op de trigger button en maak je een gesloten oppervlakte, vervolgens druk op continue.

Stationary boundary:

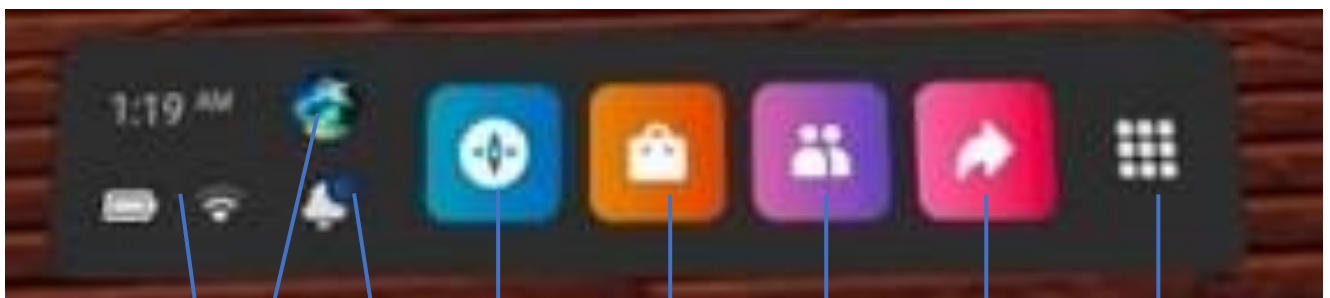
Je kan ook kiezen voor een zittend spelers gebied. Dit doe je door op “switch to stationary boundary” te drukken en zorg dat niets in de weg staat. Druk dan op confirm.

Extra:

Je kan je zicht resetten door recht voor je te kijken en dan lang op de Oculus knop te drukken. Druk daarna op de Oculus button om de setup te bevestigen.

7 Virtuele wereld verkennen

7.1 Toolbar

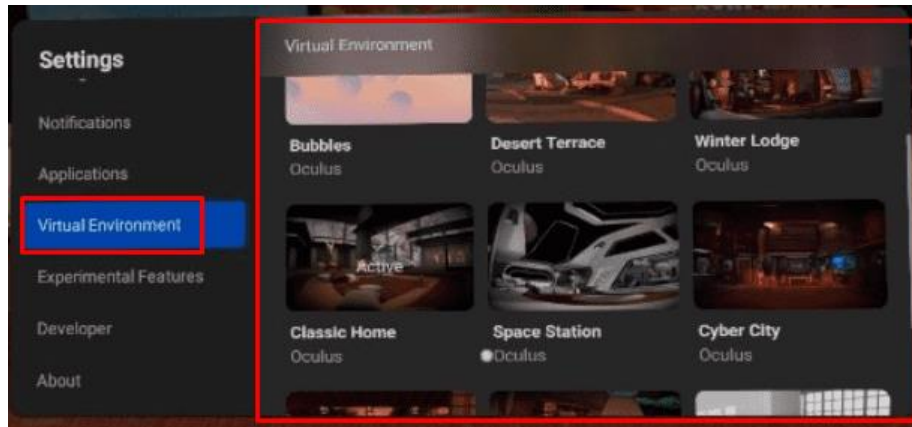


Figuur 5 Toolbar

1. Snelle instellingen: bevat al je instellingen op systeemniveau op één plek.
2. Account: hier kan je naar je profiel gaan en aanpassen ook kan je hier een Avatar maken.
3. Meldingen: toont je huidige meldingen, waaronder ook de systeemdownloads en app updates.
4. Explore: hier komen de laatste nieuwtjes en evenementen.
5. Winkel: hier kan je games kopen en downloaden.
6. Sociaal: hier kan je chatten met je Oculus vrienden of via Messenger.
7. Delen: bevat opties om te casten en foto's en video's maken in VR.
8. Apps: bevat alles gedownloade apps en meer.

7.2 Environments

In onze virtuele wereld zitten ook verschillende omgevingen. Dit kan aangepast worden door op de knop snelle instellingen te drukken en op het tandwielje boven in de rechterhoek. Dan klikken we op “virtual environments”. Daar zien we alle standaard omgevingen staan. Om die te gebruiken, moet je ze eerst downloaden door er gewoon op te drukken en daarna kan je ze selecteren.



Figuur 6 Virtual Environment

7.3 Experimental Features

Bij de experimentele functies zijn er ook nog een paar leuk dingen die we kunnen gebruiken. Het is bijvoorbeeld mogelijk voor mensen met draadloze oortjes om die te verbinden met Bluetooth, maar ook je bureau of je zetel kan je in je virtuele omgeving plaatsen. Dit vinden we terug onder de environments.

7.4 Hand Tracking

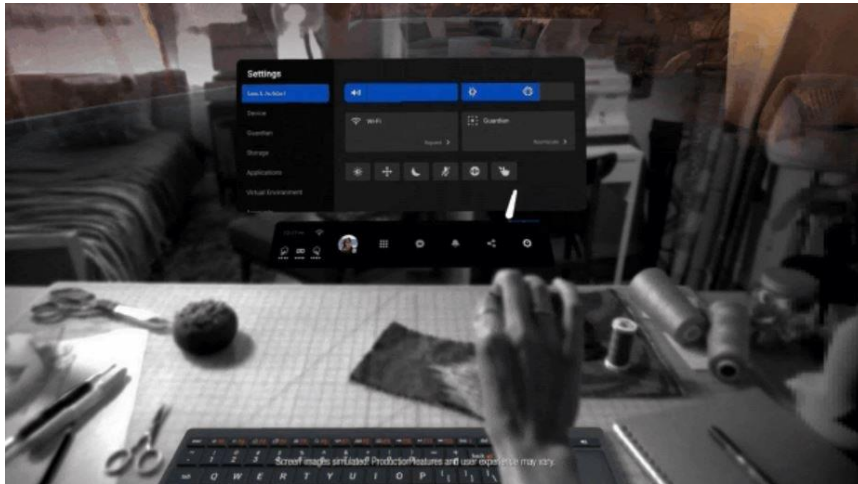
Bij de Oculus Quest 2 is er ook de mogelijkheid om i.p.v. de controllers, ook gewoon je handen te gebruiken om alles te bedienen. Hiervoor gaan we terug naar de instelling en drukken we op device. Dan drukken we op “Hands and controllers” daarna zet je hand tracking aan. Zo zal er automatisch gewisseld worden tussen jouw handen en de controllers.



Figuur 7 Hands tracking

7.5 Passthrough

Door de 4 ingebouwde camera's is het ook mogelijk om de echte wereld door de virtuele bril te zien. Dit kan namelijk handig zijn wanneer je even gedesoriënteerd bent. Om snel te kunnen switchen van de virtuele wereld naar de echte wereld heeft Oculus de "Double tab for Passthrough" toegevoegd. Dit houdt in dat je twee keer tegen de zijkant van je Quest 2 moet tikken om te schakelen. Dit moet je wel aanzetten in de settings. Daarvoor gaan we naar instellingen. Vervolgens klikken we op Guardian. Daar gaan we dan naar "Double tab for Passthrough" en zetten dat aan.



Figuur 8 Passthrough

8 Oculus wereld als developer.

De Oculus Quest 2 is niet alleen gemaakt voor het spelen van games. De makers hebben ervoor gezorgd dat ook ontwikkelaars kunnen werken met de Quest 2. Ze hebben zelfs een volledige Oculus developer platform gemaakt waar je dus makkelijk aan alle info kan geraken, maar ook het beheren van een organisatie waarbij verschillende mensen aan verschillende apps kunnen werken.

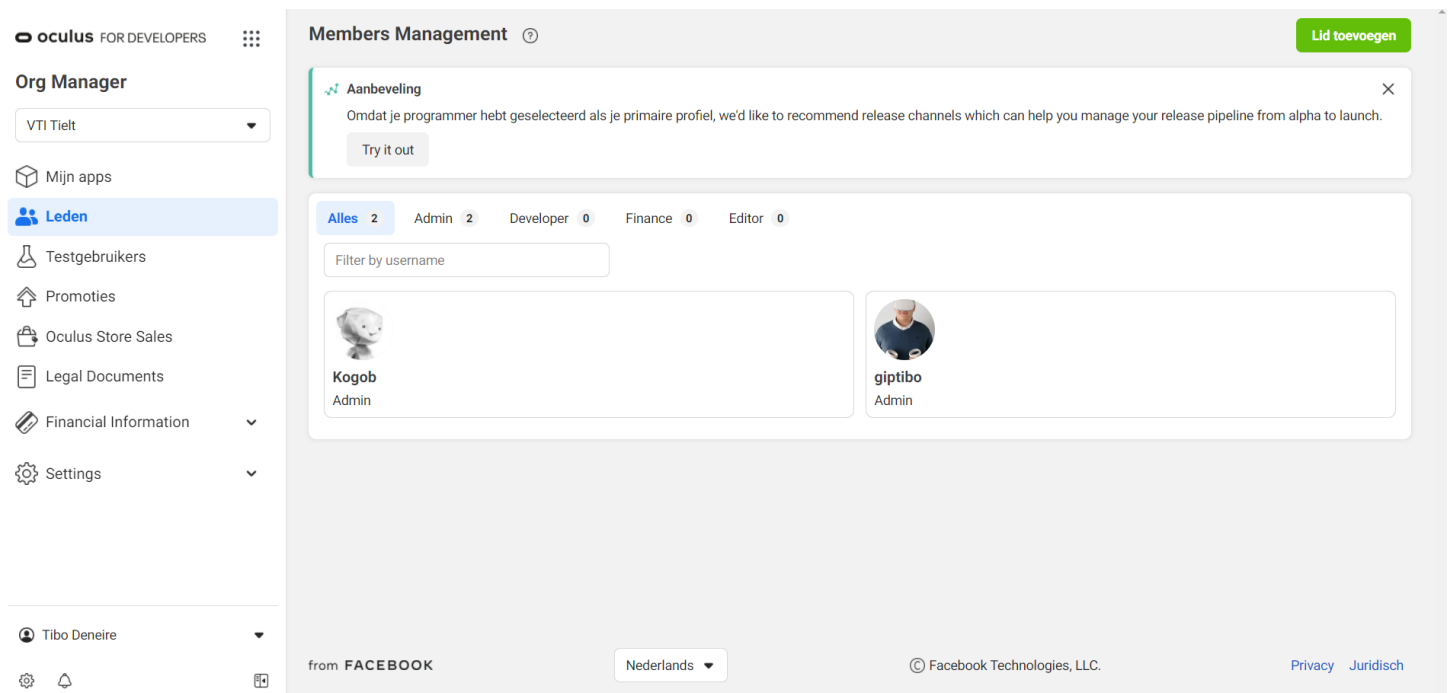
8.1 Oculus Developer Hub downloaden

Als ontwikkelaar hebben we van Oculus een speciale applicatie gekregen om te verbinden met de computer namelijk de Oculus Developer Hub afgekort de ODH. Deze is gratis te downloaden op '<https://developer.oculus.com/downloads/native-android/>'. Sinds 21 oktober is er een nieuwe update dat ervoor zorgt dat je een geverifieerd account moet hebben en dat je lid moet zijn van een organisatie.

8.2 Developer account aanmaken.

Stappenplan voor het aanmaken:

1. Surf naar <https://developer.oculus.com> en log je in met je Facebook account.
2. Ga naar <https://developer.oculus.com/manage/organizations/create> en kies om je GSM nummer toe te voegen of je bankgegevens om te verifiëren.
3. Maak een organisatie maken of je kan toegevoegd worden aan een andere organisatie.



Figuur 9 Oculus for developers organisation

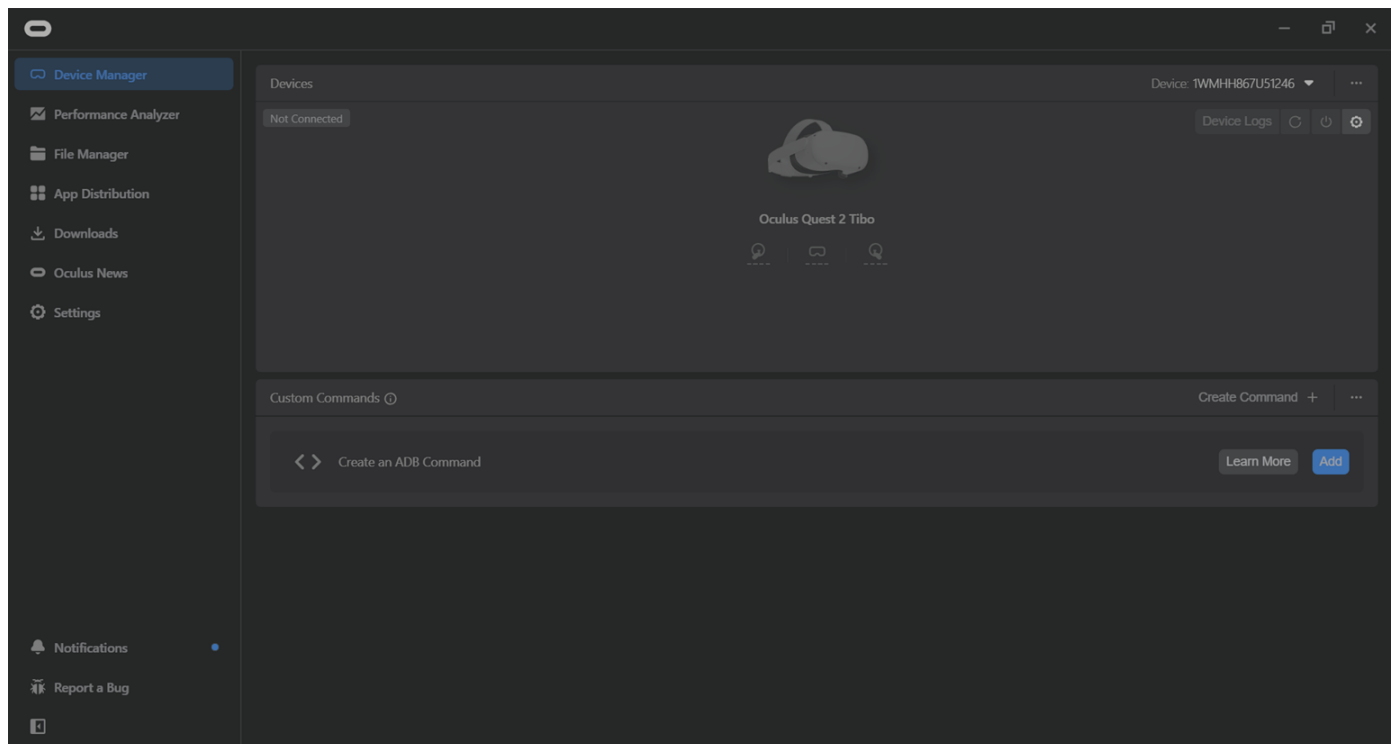
Binnen een organisatie kan je verschillende rollen krijgen zoals admin, devolper, finance en editor. Je kan ook testgebruikers aanmaken en zeggen welke leden aan welke app moeten werken. Ook de vooruitgang kan je allemaal mooi meevolgen. Uiteindelijk is het ook mogelijk om langs deze weg je game te gaan publiceren.

8.3 Developer Mode activeren

Na het maken van een account zal je de Quest 2 opnieuw moeten opstarten en zal je zien dat er nu bij instellingen een extra tab developer staat en op je GSM zal er ook een extra tab ontwikkelaar modus verschijnen. Zorg dat in de Quest 2 de USB-verbinding en de ontwikkelaars modus in de app aan staan.

8.4 Verbinden met de ODH

1. Plug de kabel in de computer en in de headset.
2. Accepteer in de headset dat er bestanden mogen uitgewisseld worden.
3. Open de ODH app op de computer.
4. Log in met je Facebook account.
5. Voeg een nieuw toestel toe.
6. Overloop de stappen en je zal je toestel zien verschijnen bij devices.



Figuur 10 Oculus developer hub

Nu zal je toestel hier elke keer als “connected” worden weergegeven als hij met de kabel verbonden is.

8.5 ODH toepassingen

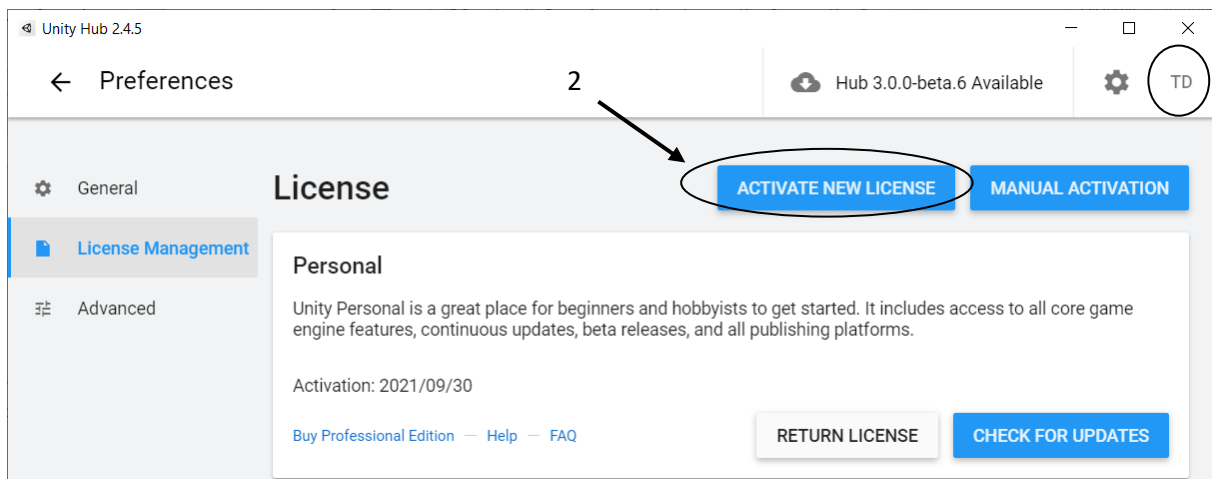
1. Beheren van je organisatie en apps.
2. Bestanden, foto's en video's uploaden, downloaden en beheren.
3. Bijnaam geven aan het toestel (dit is handig wanneer je met meerdere toestellen werkt).
4. Casten en opnemen van video's.
5. Live de performance van het toestel volgen.

9 Nieuwe VR wereld in Unity

Het is dus mogelijk om eigen werelden te maken in VR. Echter moeten we wel rekening houden met een aantal dingen, zoals het uitzicht van je wereld, de belichting, spelbesturing, de interactie tussen een speler en een object, UI (user interface), en nog veel meer. Om dit te kunnen realiseren gebruik ik het programma Unity.

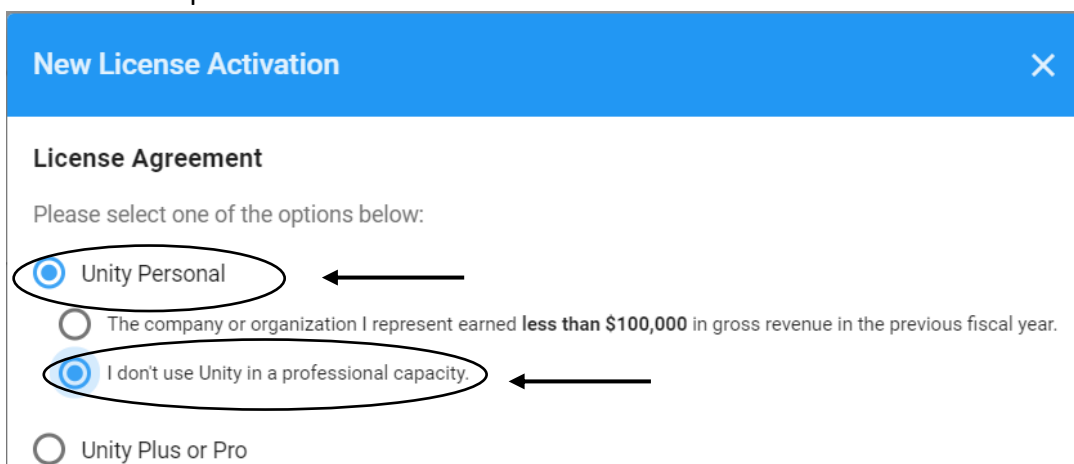
9.1 Installatie Unity

1. Download de Unity Hub <https://unity3d.com/get-unity/download>.
2. Druk op icoon (1) om in te loggen.



Figuur 11 Unity Hub License Management

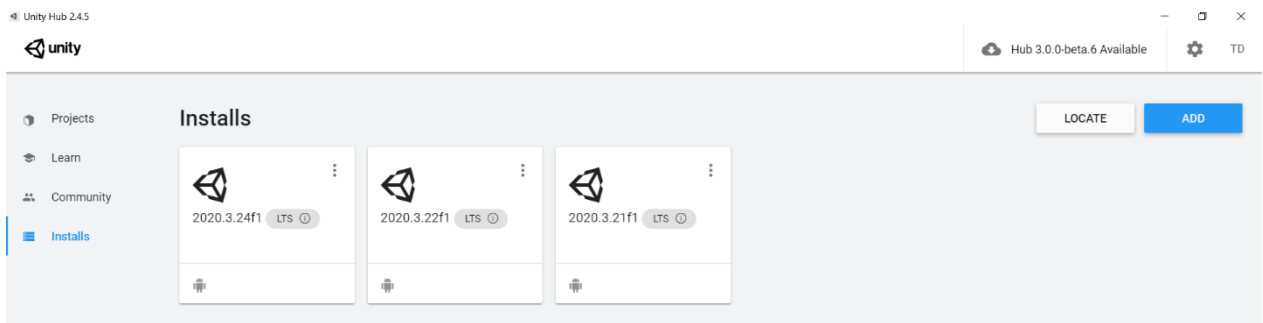
3. Log in met je inloggegevens voor de Quest 2.
4. Klik op “activate new license” (2).
5. Druk op “Unity Personal”.
6. Druk op “I don’t use Unity in a professional capacity”.
7. Druk op done.



Figuur 12 Unity Hub Activate license

Nu is de licentie geactiveerd en kunnen we verder.

1. Klik bij het beginscherm op “Installs” (1).
2. Druk op “Add” en selecteer de recentste versie.



Figuur 13 Unity Hub Install menu

9.2 Algemene setup

Deze setup zal je moeten uitvoeren bij iedere wereld die gemaakt wordt. Het houdt in dat de wereld klaar is om met de Oculus Quest 2 te werken.

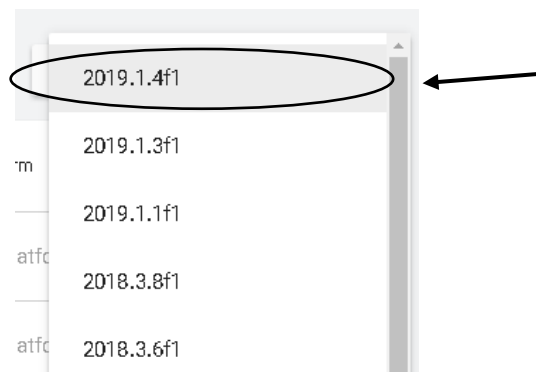
9.2.1 Project aanmaken in Unity

1. Open de Unity hub app.
2. Klik op projects (1).
3. Klik op het pijltje naast new (2).



Figuur 14 Unity Hub Projects menu

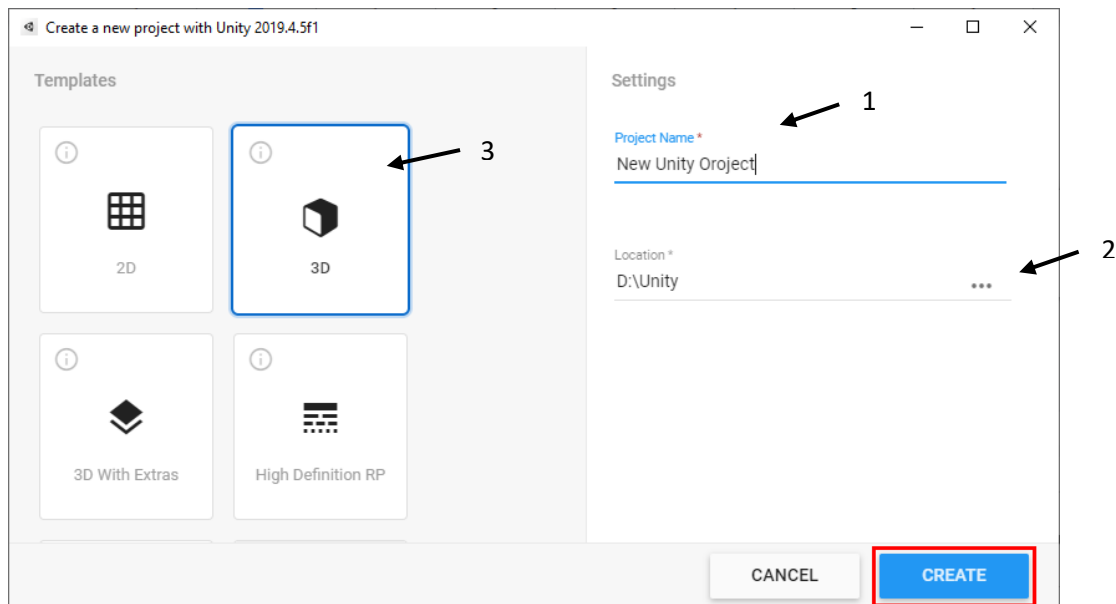
4. Selecteren de recentste versie die we gedownload hebben (dit zal de bovenste zijn).



Figuur 15 Unity Hub Versie menu

5. Er wordt automatisch een project aangemaakt.

6. Kies een naam voor het project (1).
7. Selecteer waar je het wil opslaan (2).
8. Klik dan op 3D (3). De templates zijn er om te kiezen welke app je gaat maken zoals 3D, 2D of als het voor een smartphone is en nog veel meer.
9. Druk op "Create".



Figuur 16 Unity Hub New Project

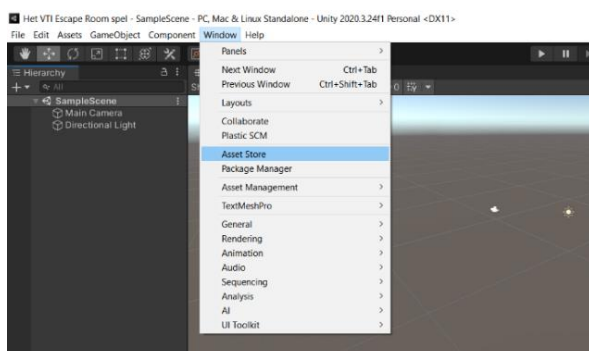
Nu is het project aangemaakt en kunnen we beginnen met het maken van de wereld.

9.2.2 Het project instellen voor de Oculus Quest 2

Omdat Unity een programma is voor veel meer dan VR alleen gaan we nog wat instellingen moeten doen om het te laten werken met de Quest 2. De instellingen die volgen zijn specifiek voor de Quest 2. Bij sommige andere VR brillen van Oculus zal het ook mogelijk zijn om deze instellingen te gebruiken.

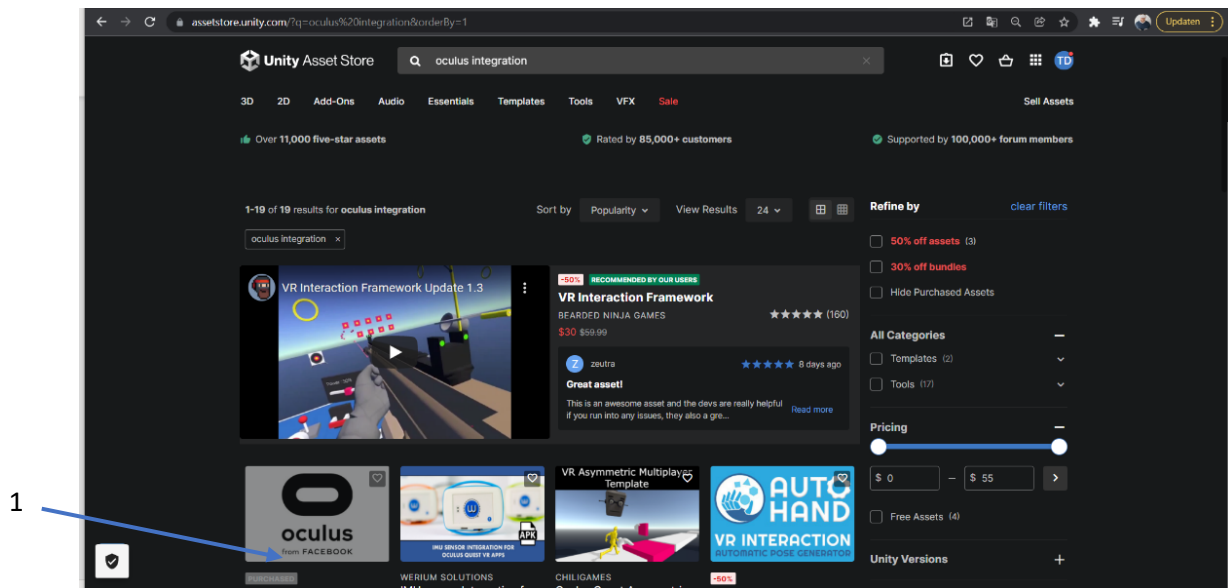
9.2.2.1 Oculus integration

1. Oculus heeft zelf al een integratie gemaakt zodat we gemakkelijk de Oculus toestellen kunnen gebruiken. Om deze te downloaden drukken we eerst op "Window" en daarna op "Asset Store". Dit zal je leiden tot een webpagina.



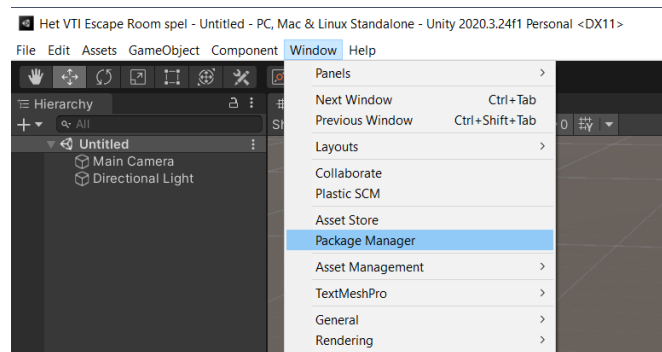
Figuur 17 Unity Window menu

2. Type in de zoekbalk Oculus integration en druk op de eerste asset met het oculus logo (1). Daarna druk je op voeg toe aan mijn assets.



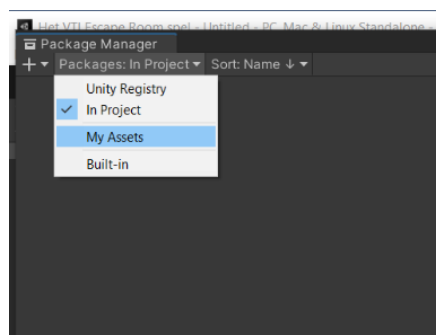
Figuur 18 Asset Store

3. Keer terug naar de Unity editor en druk terug op “Window” en dan “Package Manager”.



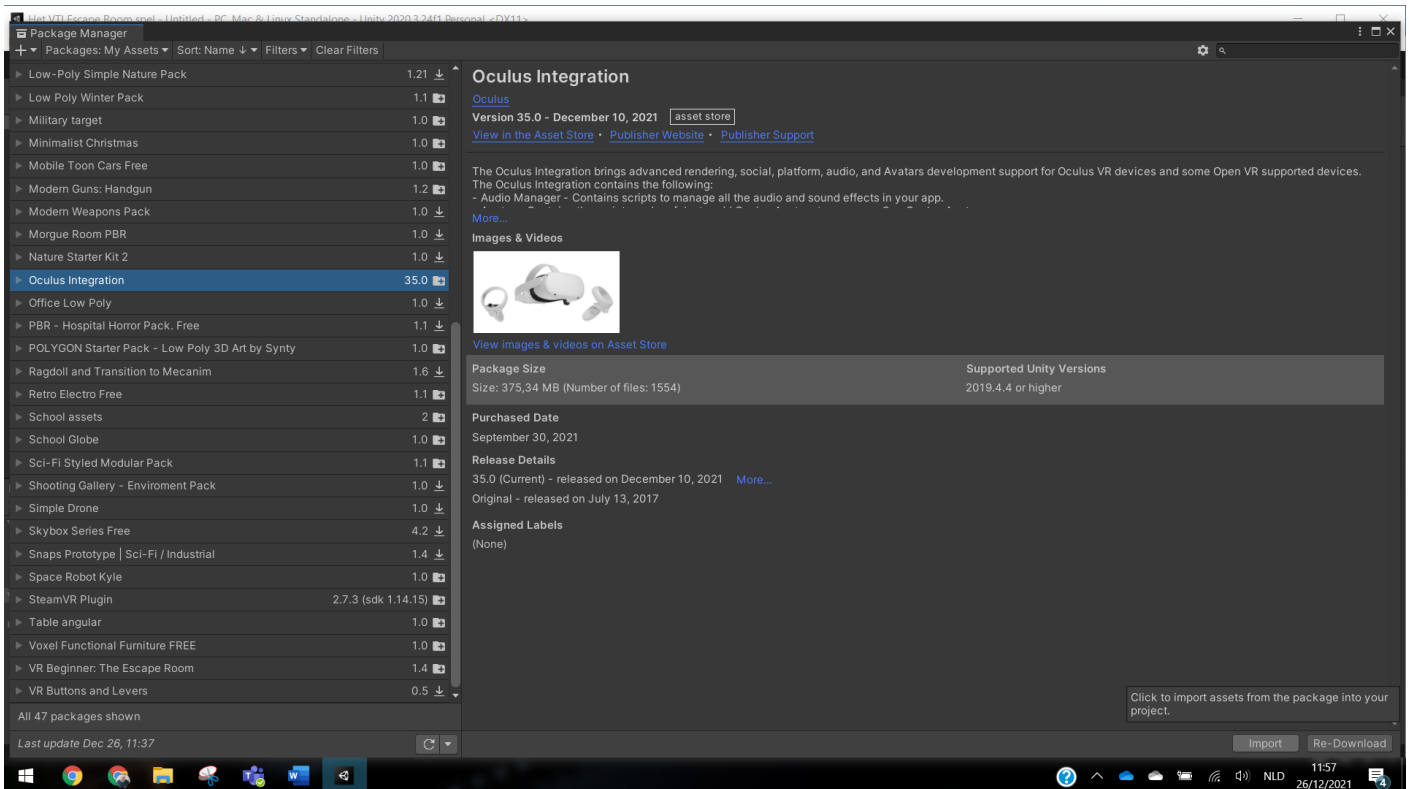
Figuur 19 Unity Window menu

4. Klik op “Packages: In Project” en verander dit in “My Assets” zoals hieronder afgebeeld.



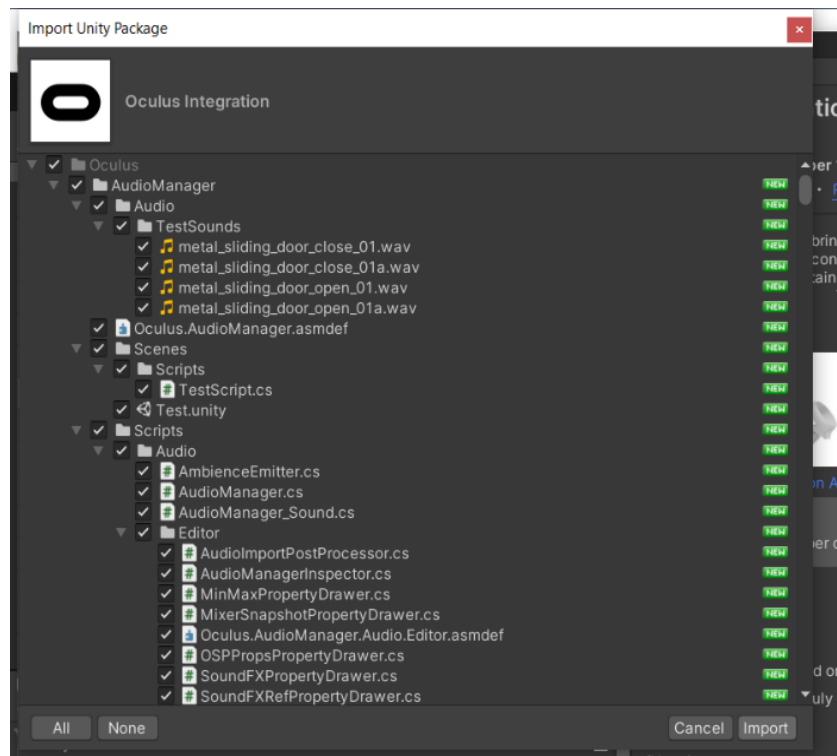
Figuur 20 Unity Package Manager

5. Zoek naar “Oculus Integration” en druk op import.



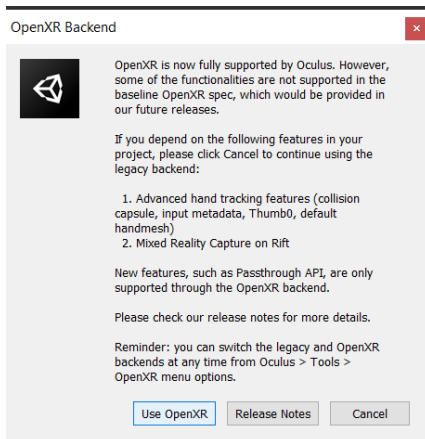
Figuur 21 Unity Package Manager

6. Een pop-up venster verschijnt met alles dat in deze asset zit. kies wat je precies wilt importeren. Van deze asset hebben we alles nodig dus drukken we gewoon nogmaals op import.

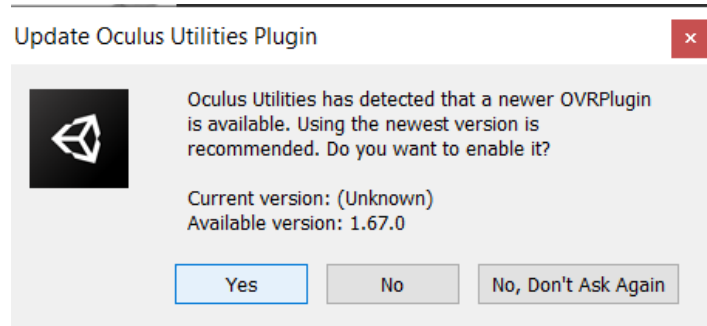


Figuur 22 Oculus Integration

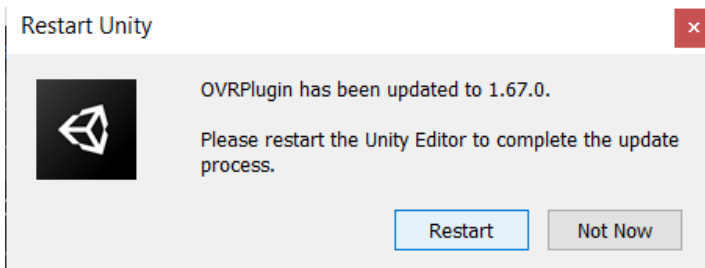
7. Daarna verschijnen een heleboel pop-up vensters met updates en andere dingen die moeten extra geïnstalleerd worden. Volg de stappen zoals hieronder afgebeeld en druk telkens op knop met de blauwe rand.



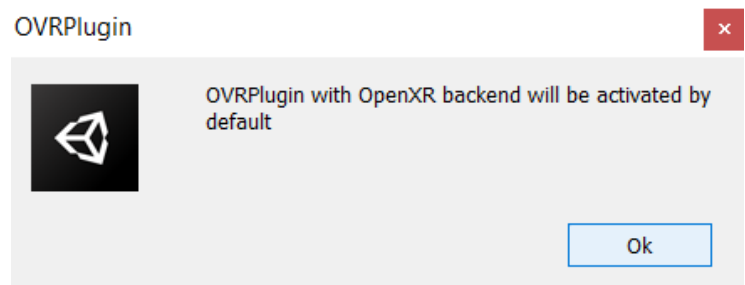
Figuur 23 Unity OpenXR Backend



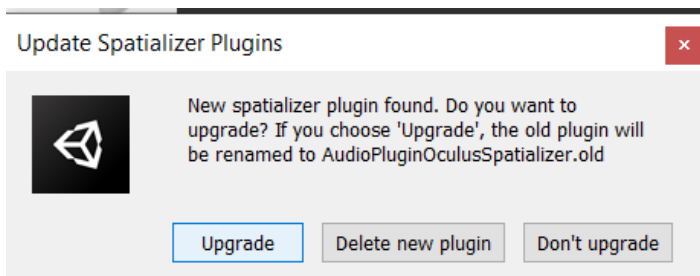
Figuur 24 Unity Update Oculus Utilities Plugin



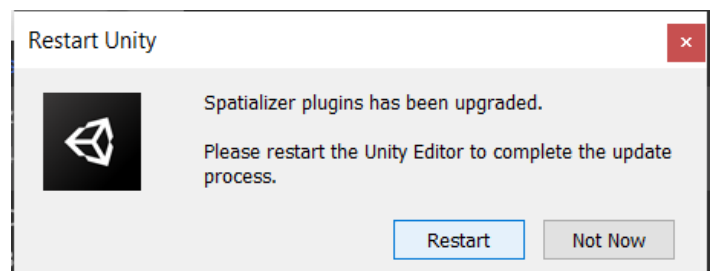
Figuur 25 Unity Restart Unity



Figuur 26 Unity OVRPlugin



Figuur 27 Unity Update Spatializer Plugins



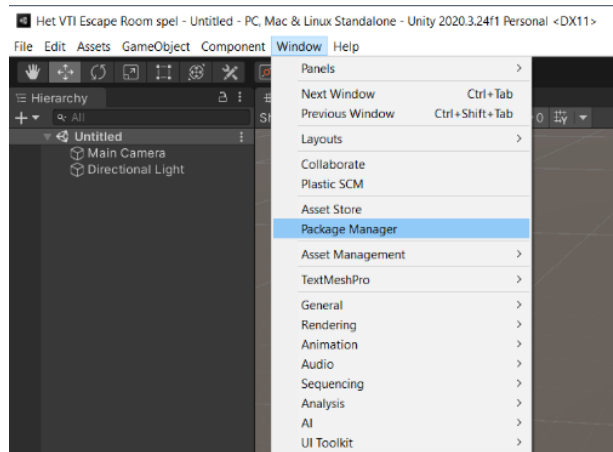
Figuur 28 Unity Restart Unity

8. Na deze stappen zal Unity herstarten en is de Oculus Integration geïmporteerd. Dit kan even duren.

9.2.2.2 XR Interaction Toolkit

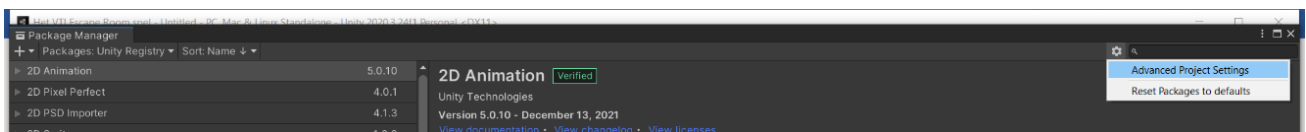
Nu gaan we de XR interaction toolkit installeren. Deze hebben we nodig om de interactie tussen objecten en de controllers te kunnen waarmaken.

1. We gaan naar “Window” en drukken we op “Package Manager”.



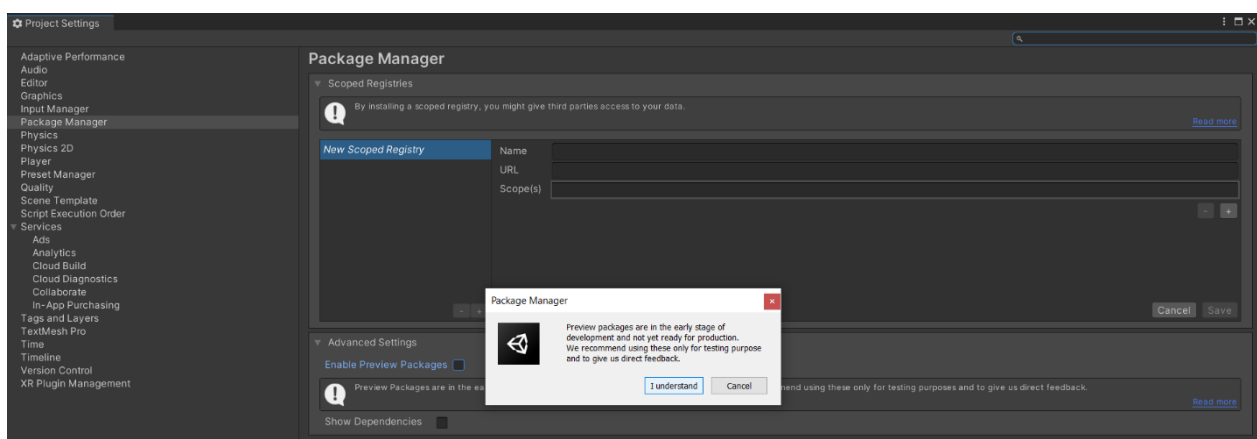
Figuur 29 Unity Window menu

2. Klik op “Packages” en verander dit in “Unity registry”.
3. Druk op het tandwielte en selecteer “Advanced Project Settings”.



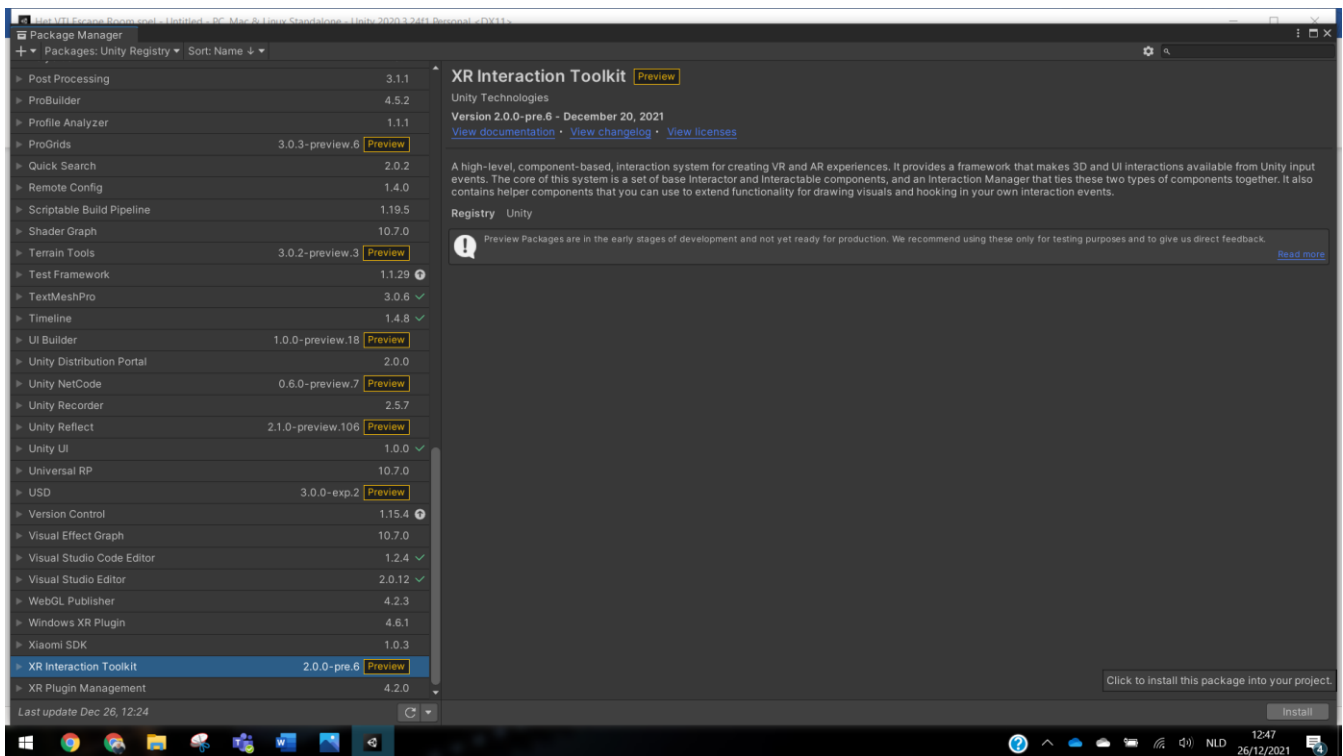
Figuur 30 Unity Package Manager

4. Druk op “Enable Preview Packages”.
5. Selecteer dan “I understand”. We doen dit omdat de XR Interaction Toolkit nog in een Beta versie staat, maar dat is voor ons geen probleem.



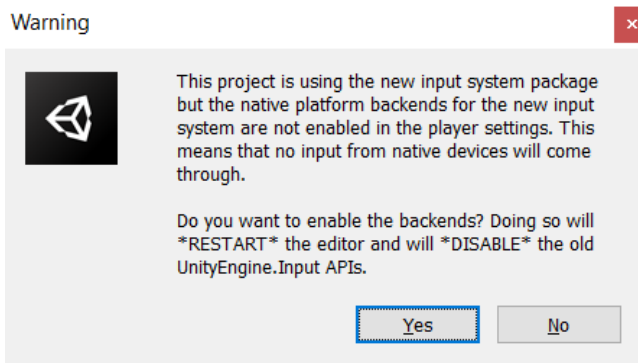
Figuur 31 Unity Project settings

6. Sluit het venster en keer terug naar de “Package Manager”.
7. Scrol naar beneden tot we “XR Interaction Toolkit” tegen komen.
8. Druk op import.

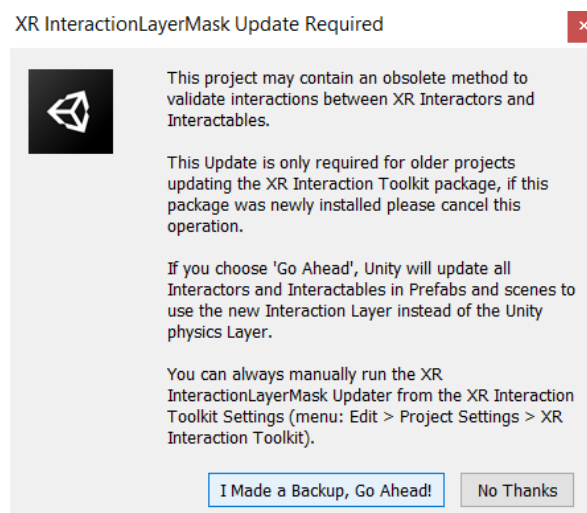


Figuur 32 Unity Package Manager

9. Druk op “Yes”.
10. Klik op “I made a backup, go ahead!”.



Figuur 33 Unity Warning



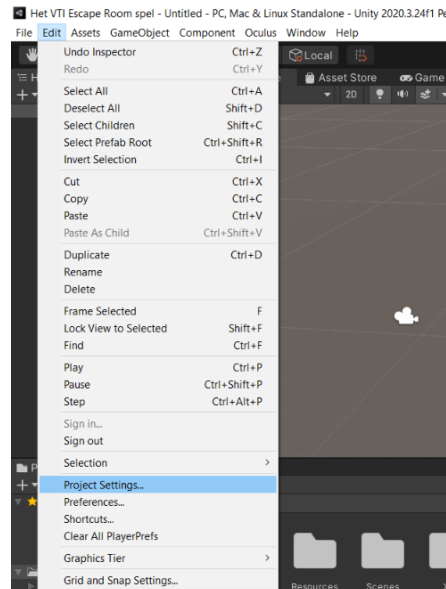
Figuur 34 Unity Update Required

Nu is de “XR Interaction Toolkit” geïnstalleerd en klaar voor gebruik.

9.2.2.3 XR Plugin Management

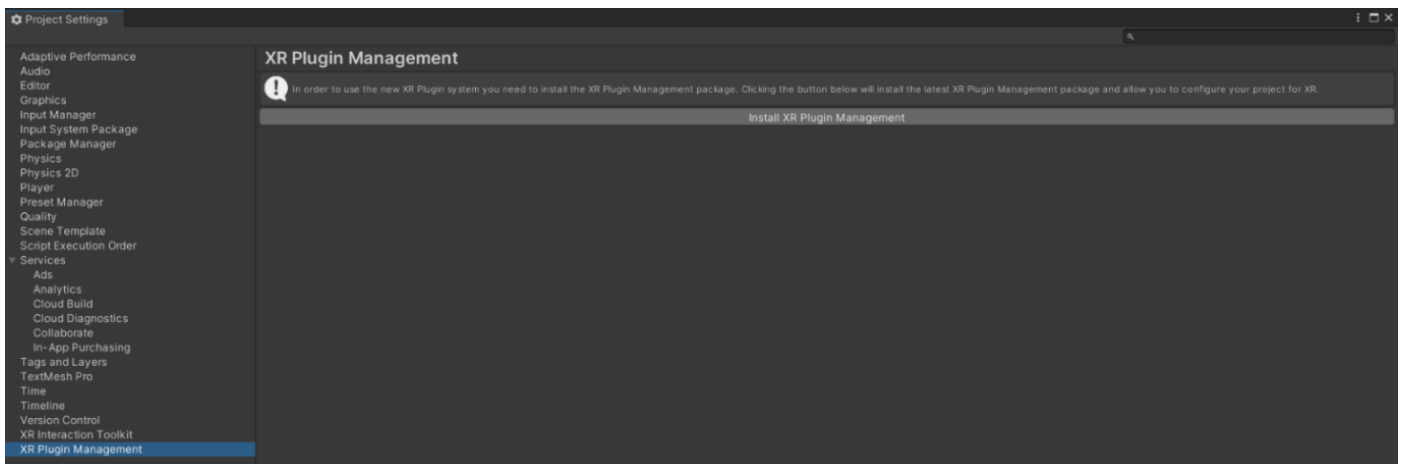
Met de XR Plugin Management zorgen we ervoor dat de app die we maken werkt op de Oculus.

1. Ga naar “Edit” en druk vervolgens op “Project Settings”.



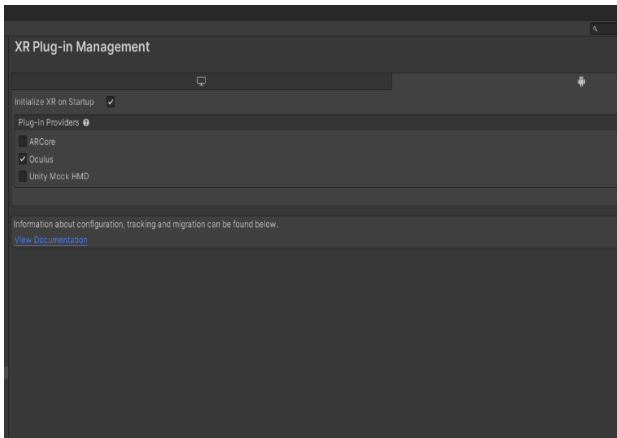
Figuur 35 Unity Edit Menu

2. Klik op “XR Plugin Management” en druk op “install XR Plugin Management”.

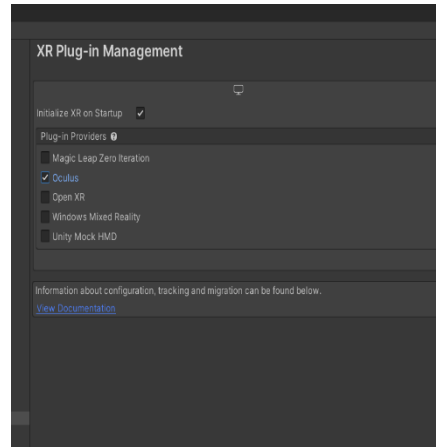


Figuur 36 Unity Project Settings

3. Omdat onze VR bril werkt met Android klikken we op Android en dan vinken we Oculus aan.
4. Klik op het computer scherm en vink Oculus aan.



Figuur 37 Unity XR Plug-in Management

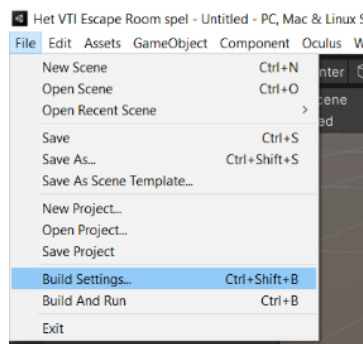


Figuur 38 Unity XR Plug-in Management

9.2.2.4 Switch to Android platform

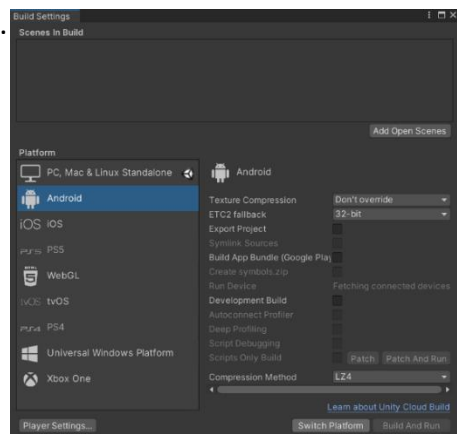
Omdat de Quest 2 met Android werkt, zullen we het upload platform moeten veranderen. Dit is vooral noodzakelijk om de app te kunnen uploaden naar de Quest 2 en om de apk (applicatie = leesbaar bestand van de app voor de Quest 2) te kunnen maken.

Eerst gaan we naar file en naar “build settings”.



Figuur 39 Unity File Menu

Vervolgens klikken we op Android en klikken we op “Switch Platform”. Nu is ons project klaar om te beginnen.



Figuur 40 Unity Build Settings

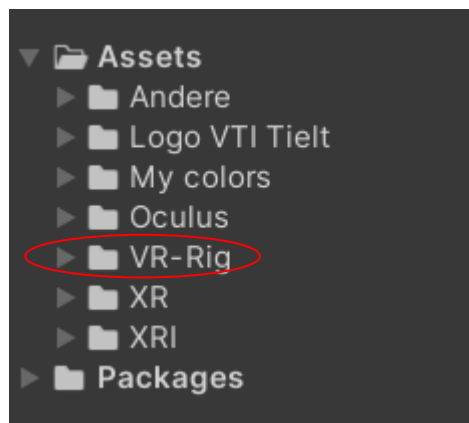
9.3 VR-Rig

Alle bestanden die gebruikt worden zijn te downloaden op mijn GitHub pagina, <https://github.com/Tibo-Deneire/Gip-Tibo>.

De VR-Rig is het belangrijkste element dat een VR developer nodig heeft. De VR-Rig zorgt ervoor dat de bewegingen in de echte wereld gekoppeld worden met deze in de virtuele wereld. De VR-Rig gaan we dus gaan gebruiken voor de handen in te stellen, teleportatie, locomotion (rondlopen) en nog voor veel meer.

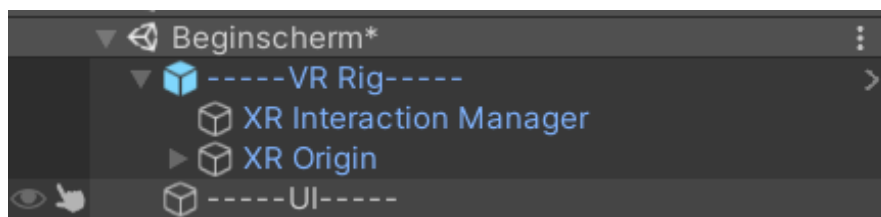
Ik heb de VR-Rig zelf gemaakt en ter beschikking gesteld op GitHub.

1. We starten met implementeren van het mapje VR-Rig. Dit doen we door het mapje in de Asset folder te slepen. Als dit goed gelukt is, zal het onder de Assets zitten.



Figuur 41 Unity Asset Folder

2. Open de map en sleep de prefab in de scène.



Figuur 42 Unity Scene overzicht

Nu hebben we de VR-Rig geïmplementeerd in de VR-Rig zitten ook al de fysieke handen ingewerkt. Hierna kan je zelf aan de slag met het maken van jouw wereld.

10 Oculus Link

Oculus Link maakt het mogelijk om aan PC gaming te doen. Op die manier zal je de game niet meer op de Quest 2 laten draaien, maar wel op je PC. De Link kabel maakt het ook mogelijk als developer om je applicaties in Unity live te testen en aan te passen.

10.1 Pc vereisten

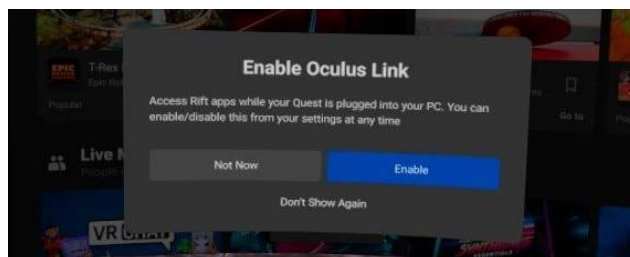
Om de Oculus Link te gebruiken moet je pc aan een aantal voorwaarden voldoen.

- Processor: Intel i5-4590/AMD Ryzen 5 1500X of hoger.
- Grafische kaart: GeForce Nvidia 9 series en hoger (zie uitzonderingen op site Meta).
- Geheugen: 8GB of meer RAM.
- Besturingssysteem: Windows 10.
- USB-poorten: 1x USB C-poort of 1x USB-poort.

10.2 Setup

Om de Oculus Link te gebruiken hebben we ook de Oculus app nodig. Die is te downloaden in de apps store voor de computer.

1. Open de Oculus app op de computer.
2. Start de Quest 2 op, kies je account en maak je speelveld.
3. Plug de Oculus Link kabel in je grafische kaart en in de Quest 2.
4. Als er een melding voor Oculus Link komt in je headset druk op enable.



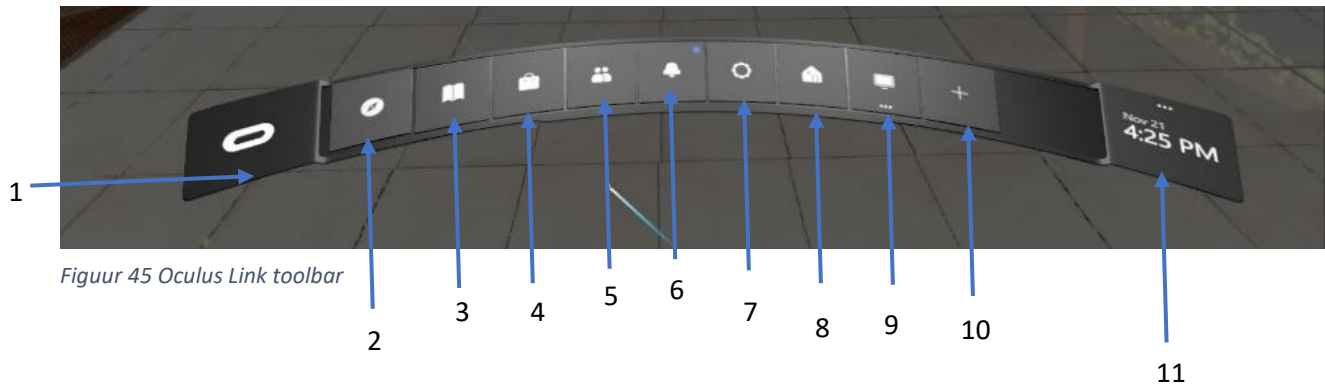
Figuur 43 Oculus Link melding

5. Als dit gelukt is krijg je dit scherm te zien.



Figuur 44 Oculus home omgeving

10.3 Toolbar



Figuur 45 Oculus Link toolbar

1. Oculus knop: kan je de Oculus Link uitschakelen.
2. Explore: om dingen op te zoeken.
3. Bibliotheek: waar je de games terug vindt.
4. Store: apps kopen.
5. Sociaal: vrienden toevoegen en party's maken.
6. Meldingen.
7. Instellingen.
8. Home: je gaat naar je home omgeving.
9. Monitor: hier kan je de beeldschermen van je computer overnemen.
10. Plusteken: selecteer deze als je extra dingen aan je toolbar wilt hangen zoals Unity.
11. Datum en tijd.

10.4 SteamVR

Steam VR is een online game platform die speciaal ontworpen is voor VR games. In Steam VR kunnen er applicaties gekocht worden, die nuttig zijn voor in de les. Zo bestaan er applicaties waar je zelf elektronische schakelingen kan bouwen en simuleren. Een andere mogelijkheid is een applicatie die het menselijk lichaam volledig toont. Je kan ook alleen de spieren gaan bekijken en nog veel meer.

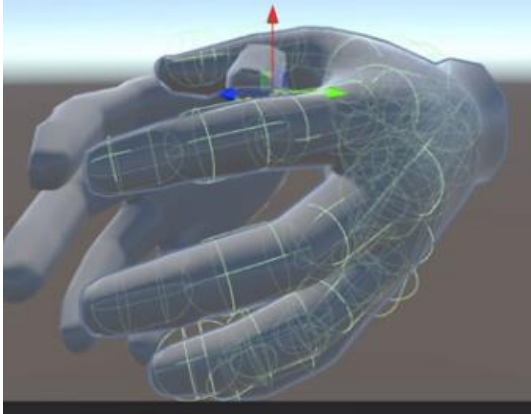


Figuur 46 Steam VR

11 Realisaties

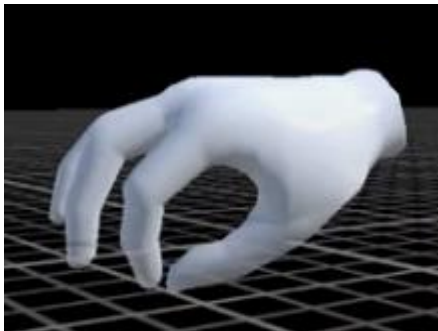
11.1 Handsen

Ik heb ervoor gezorgd dat we fysieke handen kregen in de game. Dit is handig als we een drukknop willen indrukken of als we objecten willen verplaatsen zonder vast te nemen. Om dit te realiseren moest ik colliders rond mijn volledige hand zetten.



Figuur 47 Unity Hand with colliders

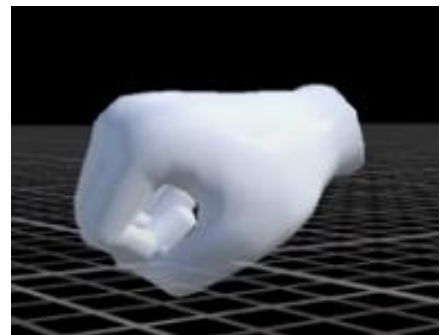
Ik heb er ook voor gezorgd dat de handen 3 poses konden aannemen. De standaard volledige hand, een vuist en een grijp pose. Deze heb ik nu gekozen en gemaakt met de animator functie van Unity. Het is dus ook mogelijk om andere poses aan te nemen.



Figuur 49 Unity Hand Pinch



Figuur 48 Unity Full Hand

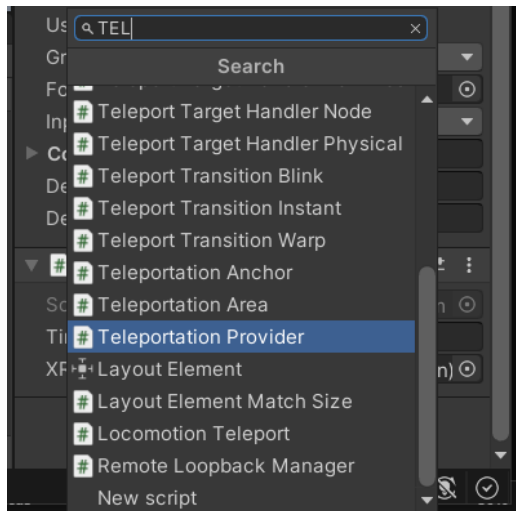


Figuur 50 Unity fist

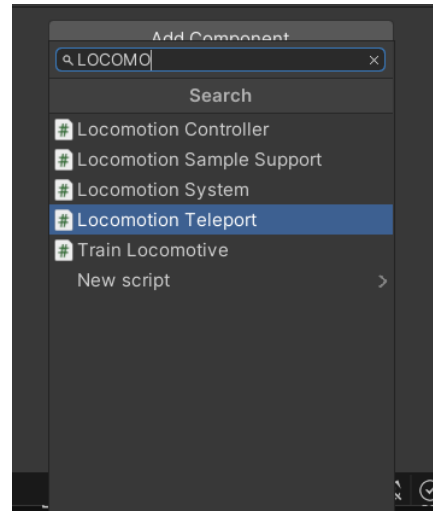
11.2 Teleportatie

Om zich te verplaatsen in de game kunnen we gebruik gaan maken van teleportatie. Dit houdt in dat we van het ene punt naar het andere punt gaan. We kunnen een speler vrij laten teleporteren, maar die plaatsen kunnen we ook vastleggen.

1. We starten met een Locomotion Systeem toe te voegen, door naar de VR-Rig te gaan en onderaan op “Add Component” te klikken.
2. Type “Locomotion System” en voeg het toe.
3. Klik op “Add Component” en type “Teleportation Provider”.

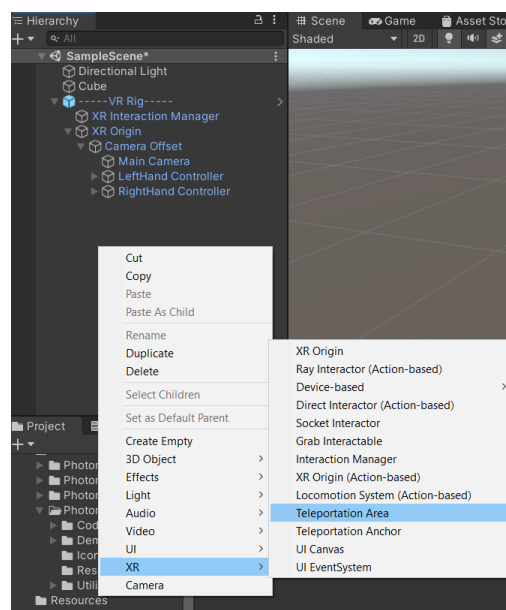


Figuur 52 Unity Teleportation Provider



Figuur 51 Unity Locomotion System

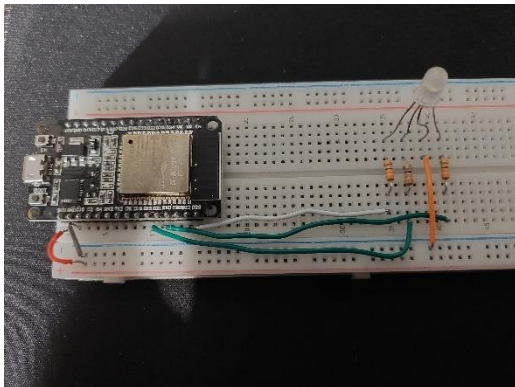
4. Nu hebben we ervoor gezorgd dat we kunnen teleporteren, maar nu moeten we ook nog een teleportatie gebied toevoegen zodat we niet buiten het gebied kunnen teleporteren.
5. Voeg een Teleportation Area toe door in de scène op de linkermuisknop te drukken en dan op “XR” en vervolgens op “Teleportation Area”. Nu kunnen we gaan teleporteren.



Figuur 53 Unity Teleportation Area

11.3 Connectie met de buitenwereld

Om de connectie met de buitenwereld te tonen heb ik een opstelling gemaakt met 3 knoppen in de virtuele wereld een RGB led gekoppeld aan een ESP 32 in de echte wereld.



Figuur 55 ESP met RGB



Figuur 54 Unity kamer met drukknoppen

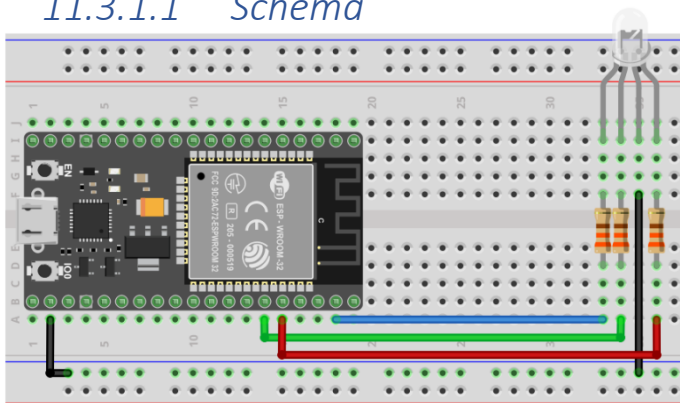
De communicatie tussen de twee gebeurt via een webserver. Vanuit mijn virtuele wereld wordt er een Json naar een web applicatie gestuurd aan de hand van een HTTP Post request. De Esp gaat dan met een HTTP Get request de waarde gaan opvragen.



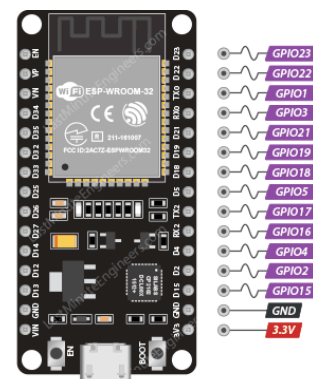
Figuur 56 Communicatie Schema

11.3.1 ESP met RGB Led

11.3.1.1 Schema



Figuur 57 Schema ESP en RGB Led



11.3.1.2 Programma

Het programma voor de ESP is beschikbaar als bijlage. Het is geschreven in de Arduino taal C++ met het programma Arduino IDE.

Het programma zal een HTTP GET Request uitvoeren aan de site <http://gip20212022.gobbin.be/api/tibo/knop/1>. Er zal een JSON teruggegeven worden, en deze JSON wordt omgezet naar variabelen. Afhankelijk van wat deze variabelen zijn zal de led al dan niet branden.

11.3.2 Virtuele wereld

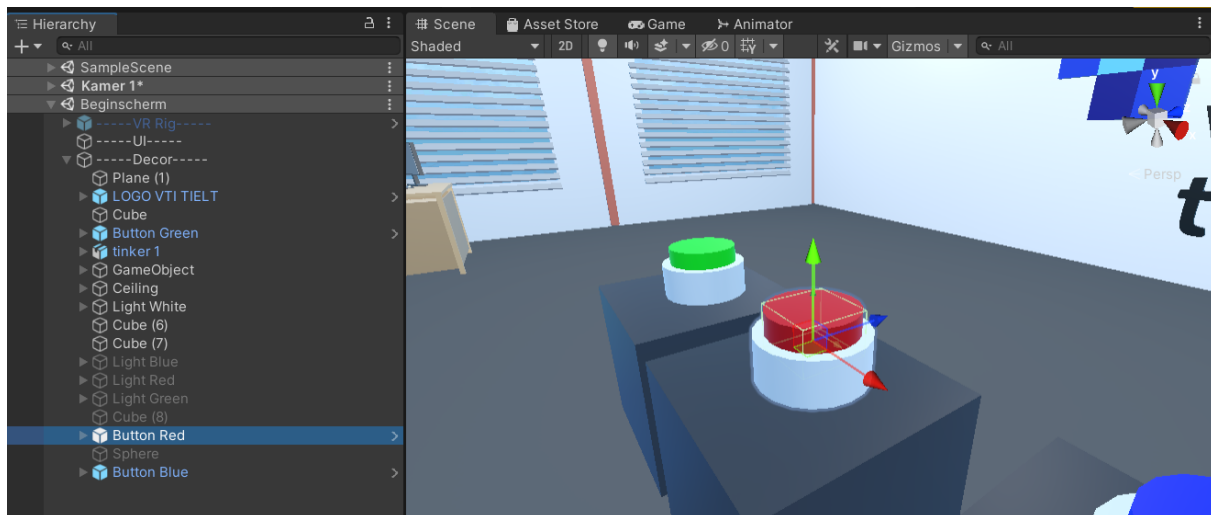
11.3.2.1 Programma

Het programma is achteraan te bezichtigen, het gaat een http post request doen met de waardes die hij eerst in een Json gaat omzetten.

11.3.2.2 Knoppen

Het bestand (prefabs, scripts, collors) voor de knoppen zijn ook terug te vinden op mijn GitHub pagina.

1. Importeer de map “drukknoppen” in de Assets vervolgens slepen we de prefab in de hiërarchie.



Figuur 58 Unity Drukknoppen

2. Verplaats de prefab naar een locatie volgens keuze. Er kunnen andere kleuren aan de knop gegeven worden door een ander materiaal erop te slepen.

De drukknoppen zijn geprogrammeerd zodat ze actief worden vanaf dat de ondergrens wordt getriggerd. Deze blijft actief zolang het percentage van de bovengrens niet overschreden wordt.

11.4 Wegwijs in de virtuele wereld

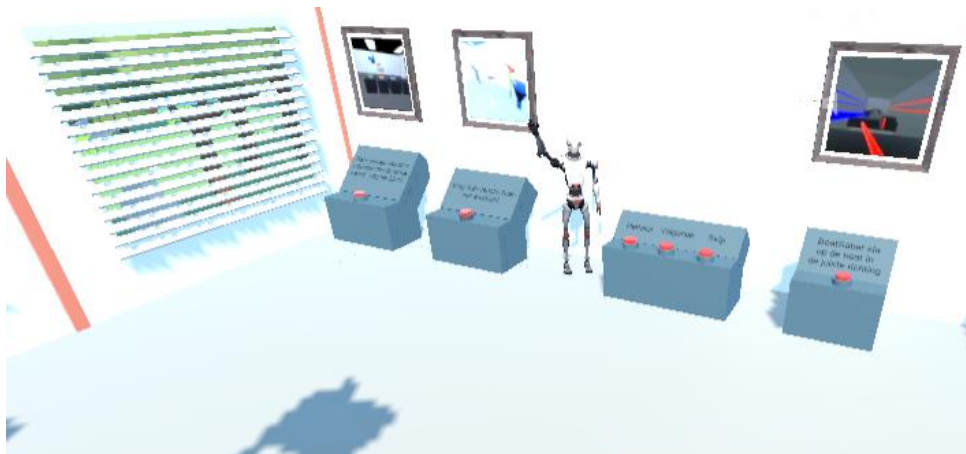
Omdat VR voor heel wat mensen nieuw is, heb ik een applicatie ontwikkeld waar men stap voor stap leert kennis maken met Virtual Reality. De spelers zullen leren hoe ze moeten omgaan met de controllers of met de handen.

De speler komt eerst terecht in een klaslokaal waar een robot hun zal toespreken en wat uitleg geeft.



Figuur 59 Unity intro

Vervolgens gaan ze naar de main hub, daar zal je kunnen kiezen uit een paar werelden. Elke wereld heeft iets speciaals zoals een reactie baan, knoppen die verbonden zijn met de buitenwereld en nog veel meer. Er is ook een mogelijkheid om eerst kennis te maken met VR en dus een uitleg te krijgen over hoe de controllers werken en hoe je je kan verplaatsen.



Figuur 60 Unity Main Hub

11.5 Beeldmateriaal

Van mijn realisaties heb ik ook beeldmateriaal, al dit beeld materiaal zal ook terug te vinden zijn op mijn GitHub pagina (link <https://github.com/Tibo-Deneire/Gip-Tibo>) onder het mapje Beeldmateriaal.

12 Besluiten

Ik vond dit een zeer leuke en uitdagende GIP. Ik heb af en toe wel eens problemen tegen gekomen. Zo heb ik software updates moeten uitvoeren bij de ODH en heb ik een tijd lang niet kunnen uploaden naar mijn Quest 2 en kon ik niet casten. Halverwege mijn GIP heeft Oculus ook nog wat aanpassingen gedaan, zodat ik in Unity andere commando's moest gebruiken bv: VR-Rig werd XR-Origin. Daar heb ik wat tegenslagen tegengekomen.

Op de vraag wat kan een leerling van de 3^{de} graad elektriciteit-elektronica zelf in VR, heb ik lang over moeten nadenken. Want als je een VR game moet maken, moet je met heel veel rekening houden. Als je een VR wereld vanaf nul moet maken heb je ook enorm veel tijd nodig. Als je hier mee begint moet je wel beginnen met een basis Unity VR. Als je die basis niet hebt, zal het zeer moeilijk zijn om juist programma's te schrijven.

Het naar buiten brengen van info uit de virtuele wereld of naar binnenbrengen, is wel een stuk makkelijker en lijkt heel erg op de Arduino code (C++). Ik heb er nu een RGB Led mee laten besturen, maar er zijn zeker nog meer leuke toepassingen mogelijk.

Het was voor mij de eerste keer dat ik in aanraking kwam met VR. Door deze ervaringen van het afgelopen jaar, heb ik ook besloten om verder te studeren in een richting waar VR ook aan bod zal komen.

Conclusie: het is mogelijk voor een 3^{de} graad EE om aan de slag te gaan met VR mits er voldoende tijd kan aan besteed worden. De uitwisseling van info naar en van VR is mogelijk en niet zo heel moeilijk. Het was een zeer leuke en interessante GIP waarbij er genoeg uitdagingen waren.

13 Prijslijst



Grote Hulststraat 28

8700 Tielt

Tel. 051 40 05 68

benodigdhedenlijst

Artikelbenaming	Aantal	Eenheidsprijs(€)	Totale prijs(€)
Oculus			
Oculus Quest 2 128GB	1	349,00 €	349,00 €
Oculus Link Kabel	1	99,00 €	99,00 €
Quest 2 Elite band met batterij*	1	109,00 €	109,00 €
Quest 2 opbergkoffer*	1	49,00 €	49,00 €
Componenten			
ESP 32	1	12,50 €	12,50 €
Jumper kabels 40stk	1	4,50 €	4,50 €
RGB Led	1	0,75 €	0,75 €
330 Ohm weerstand	3	0,10 €	0,30 €
Breadbord	1	4,85 €	4,85 €

* extra (niet verplicht)

Totaal	470,90 €
Totaal*	628,90 €

14 Begrippenlijst

EE: Elektriciteit-Elektronica, de richting die ik volg

VR: Virtual Reality,

XR: Extended Reality

AR: Augmented Reality,

ODH: Oculus developer hub, platform voor developers waar we de Quest 2 kunnen beheren

Prefab: is een item dat al klaar gemaakt is, als dat om een object gaat dan zal dat de vorm en de kleuren al bevatten. De prefab kan ook al instellingen bevatten of scripts. Deze kan je herkennen aan de blauwe kubussen.

Materials: dit zijn de verschillende kleuren/materialen je een object kan geven. Die worden cirkelvormig weergegeven.

Scènes: je kan verschillende scènes maken om een onderscheid te maken van bepaalde plaatsen bv: je kan een scène maken voor het beginscherm en dan nog een scène voor je huis. Je kan de verschillende scènes herkennen in de hiërarchie doordat het Unity logo ernaast staat.

Colliders: ze zorgen ervoor dat het object fysiek wordt. Deze hebben we ook nodig wanneer we objecten willen vast nemen. We hebben ook verschillende soorten colliders zoals box, sphere, capsule, circle, ...

15 Figuurlijst

Figuur 1 Soorten virtuele technologieën.....	11
Figuur 2 AR/VR/MR	12
Figuur 3 Oculus Quest 2 VR Bril.....	13
Figuur 4 Oculus Quest 2	14
Figuur 5 Toolbar	16
Figuur 6 Virtual Environment	17
Figuur 7 Hands tracking.....	17
Figuur 8 Passthrough.....	18
Figuur 9 Oculus for developers organisation	19
Figuur 10 Oculus developer hub	20
Figuur 11 Unity Hub License Management	21
Figuur 12 Unity Hub Activate license	21
Figuur 13 Unity Hub Install menu.....	22
Figuur 14 Unity Hub Projects menu	22
Figuur 15 Unity Hub Versie menu	22
Figuur 16 Unity Hub New Project.....	23
Figuur 17 Unity Window menu	23
Figuur 18 Asset Store.....	24
Figuur 19 Unity Window menu	24
Figuur 20 Unity Package Manager.....	24
Figuur 21 Unity Package Manager.....	25
Figuur 22 Oculus Integration	25
Figuur 23 Unity OpenXR Backend.....	26
Figuur 24 Unity Update Oculus Utilities Plugin	26
Figuur 25 Unity Restart Unity	26
Figuur 26 Unity OVRPlugin	26
Figuur 27 Unity Update Spatializer Plugins	26
Figuur 28 Unity Restart Unity	26
Figuur 29 Unity Window menu	27
Figuur 30 Unity Package Manager.....	27
Figuur 31 Unity Project settings	27
Figuur 32 Unity Package Manager.....	28
Figuur 33 Unity Warning	28
Figuur 34 Unity Update Required.....	28
Figuur 35 Unity Edit Menu	29
Figuur 36 Unity Project Settings.....	29
Figuur 37 Unity XR Plug-in Managment	30
Figuur 38 Unity XR Plug-in Managment	30
Figuur 39 Unity File Menu	30
Figuur 40 Unity Build Settings	30
Figuur 41 Unity Asset Folder	31
Figuur 42 Unity Scene overzicht.....	31
Figuur 43 Oculus Link melding	32

Figuur 44 Oculus home omgeving	32
Figuur 45 Oculus Link toolbar	33
Figuur 46 Steam VR	33
Figuur 47 Unity Hand witch colliders	34
Figuur 48 Unity Full Hand	34
Figuur 49 Unity Hand Pinch	34
Figuur 50 Unity fist	34
Figuur 51 Unity Locomotion System	35
Figuur 52 Unity Teleportation Provider	35
Figuur 53 Unity Teleportation Area	35
Figuur 54 Unity kamer met drukknoppen	36
Figuur 55 ESP met RGB	36
Figuur 56 Communicatie Schema	36
Figuur 57 Schema ESP en RGB Led	36
Figuur 58 Unity Drukknoppen	37
Figuur 59 Unity intro	38
Figuur 60 Unity Main Hub	38

16 Programma's

16.1 Programma ESP met RGB Led

```

/*
Module: ESP32 Wroom
pinout: LED_Groen = 19;
        LED_Blaauw = 22;
        LED_Rood = 21;
        POT = 2

*/

////////// INCLUDE //////////
#include <WiFi.h>
#include <HTTPClient.h>
#include <Arduino_JSON.h>

////////// CONST CHAR //////////
const char *ssid = "Casa 45";
const char *password = "45454545";

////////// INT //////////
int LED[3]{19, 22, 21};
int welkeknop;
int httpResponseCode[3];

////////// BOOL //////////
bool ledstate;

////////// STRING //////////
String payload[3];
String serverNameKnop[3] = {"http://gip20212022.gobbin.be/api/tibo/knop/1", "http://gip2021
2022.gobbin.be/api/tibo/knop/2", "http://gip20212022.gobbin.be/api/tibo/knop/3"}; //Your Dom
ain name with URL path or IP address with path
String serverPath[3];

////////// UNSIGNED LONG //////////
unsigned long lastTime = 0;
//unsigned long timerDelay = 600000; // Timer set to 10 minutes (600000)
unsigned long timerDelay = 5000; // Set timer to 5 seconds (5000)

////////// SETUP //////////
void setup()
{
    Serial.begin(115200); // Start Serial monitor
    for (int i = 0; i <= 2; i++) // set all leds as output
    {
        pinMode(LED[i], OUTPUT);
        Serial.println("led");
        Serial.println(i);
    }
}

```

```

WiFi.begin(ssid, password); // Start wifi
Serial.println("Connecting");
while (WiFi.status() != WL_CONNECTED) //Connect to wifi
{
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.print("Connected to WiFi network with IP Address: ");
Serial.println(WiFi.localIP()); // print IP adress

Serial.println("Timer set to 5 seconds (timerDelay variable), it will take 5 seconds before publishing the first reading.");
}

////////// LOOP //////////
void loop()
{
    if ((millis() - lastTime) > timerDelay)
    {
        if (WiFi.status() == WL_CONNECTED) //Check WiFi connection status
        {
            HTTPClient http; // Start the http server client
            for (int i = 0; i <= 2; i++)
            {
                serverPath[i] = serverNameKnop[i]; // link the URL to a path. here you can add something else to your URL
                http.begin(serverPath[i].c_str()); // Your Domain name with URL path or IP address with path
                httpResponseCode[i] = http.GET(); // GET request

                if (httpResponseCode[i] > 0) // Check if something came in or not
                {
                    Serial.print("HTTP Response code: ");
                    Serial.println(httpResponseCode[i]);
                    payload[i] = http.getString(); // Get the data
                    Serial.println(payload[i]);

                    JSONVar myObject = JSON.parse(payload[i]); // put the json into a variable

                    if (JSON.typeof(myObject) == "undefined")// JSON.typeof(jsonVar) can be used to get the type of the var
                    {
                        Serial.println("Parsing input failed!");
                        return;
                    }

                    Serial.print ("JSON.typeof(myObject) = ");

```



```

        Serial.println(JSON.typeof(myObject)); // prints: object

        if (myObject.hasOwnProperty("welkeKnop")) // myObject.hasOwnProperty(key) checks
if the object contains an entry for key
        {
            Serial.print("myObject[\"welkeKnop\"] = ");

            Serial.println((int)myObject["welkeKnop"]);
            welkeknop = (int)myObject["welkeKnop"]; // put the number of button into the va
riable
        }

        if (myObject.hasOwnProperty("statusKnop"))
        {
            Serial.print("myObject[\"statusKnop\"] = ");

            Serial.println((bool)myObject["statusKnop"]);
            ledstate = (bool)myObject["statusKnop"]; // put the ledstate into the variable

            Serial.println();
        }
        else
        {
            Serial.print("Error code: "); // error trying to connect
            Serial.println(httpResponseCode[i]);
        }
    }
    digitalWrite(LED[i], ledstate); // set on/off the good led
}

    lastTime = millis();
}
}
}

```

16.2 Programma knoppen in Unity

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Events;
using UnityEngine.Networking;
using System.IO;

```

```
public class PhysicsButtonBlue : MonoBehaviour
{
    [SerializeField] private float threshold = .1f;
    [SerializeField] private float deadZone = .025f;

    private bool _isPressed;
    private Vector3 _startPos;
    private ConfigurableJoint _joint;
    private string json;
    private bool gestuurd = false;

    public int welkeKnop;
    public bool statusKnop;

    public UnityEvent onPressed, onReleased;
    public AudioSource source;
    public AudioClip fireSound;

    void Start()
    {
        _startPos = transform.localPosition;
        _joint = GetComponent<ConfigurableJoint>();
    }

    // Update is called once per frame
    void Update()
    {
        if (!_isPressed && GetValue() - threshold > 0)
            Pressed();
        if (_isPressed && GetValue() - threshold <= 0)
            Released();
    }

    private float GetValue()
    {
        var value = Vector3.Distance(_startPos, transform.localPosition) /
        _joint.linearLimit.limit;

        if (Mathf.Abs(value) < deadZone)
            value = 0;

        return Mathf.Clamp(value, -1f, 1f);
    }

    private void Pressed()
    {
        _isPressed = true;
        onPressed.Invoke();
        //source.PlayOneShot(fireSound);
        Debug.Log("pressed");

        if (gestuurd == false)
        {
            Save1();
            StartCoroutine(Upload());
            gestuurd = true;
        }
        else if (gestuurd == true)
        {
            Save2();
        }
    }
}
```



```

        StartCoroutine(Upload());
        gestuurd = false;
    }
}
private void Released()
{
    _isPressed = false;
    onReleased.Invoke();
    Debug.Log("Released");
}

IEnumerator Upload()
{
    WWWForm form = new WWWForm();
    using (UnityWebRequest www =
UnityWebRequest.Post("http://gip20212022.gobbin.be/api/tibo/knop", json))
    {
        www.uploadHandler = new
UploadHandlerRaw(System.Text.Encoding.UTF8.GetBytes(json));

        www.SetRequestHeader("Content-Type", "application/json");
        yield return www.SendWebRequest();

        if (www.result != UnityWebRequest.Result.Success)
        {
            Debug.Log(www.error);
        }
        else
        {
            Debug.Log("Upload complete!");
        }
    }
}
[System.Serializable]
private class JSONObject
{
    public int welkeKnop;
    public bool statusKnop;
}
void Save1 ()
{
    JSONObject stuurjson = new JSONObject();
    stuurjson.welkeKnop = 1;
    stuurjson.statusKnop = true;
    json = JsonUtility.ToJson(stuurjson);
    Debug.Log("json");
    Debug.Log(json);
}
void Save2()
{
    JSONObject stuurjson = new JSONObject();
    stuurjson.welkeKnop = 1;
    stuurjson.statusKnop = false;
    json = JsonUtility.ToJson(stuurjson);
    Debug.Log("json");
    Debug.Log(json);
}
}

```