## Documentatie project special input/output

Project Special Input/Output is het project dat wij volgen in de derde periode van ons 2e schooljaar met doel om iets te maken doormiddel van speciale input, denk zo aan voice-recognition of een VR-Project.

Wij hebben ervoor gekozen om doormiddel van het gebruik van de Leap Motion een instrument te kunnen bespelen, zo hebben wij specifiek gekozen voor de Piano.

Dit bestand zal verder de Documentatie bijhouden van alles wat wij zijn tegengekomen, onze eventuele veranderingen, alles dat wij hebben geleerd en natuurlijk de progressie op ons werk.

Team

Job Tol – Klas GD2B

Justin van der Lans – Klas GD2B

***MediaCollege Amsterdam***

## Inleiding

1] Ons Idee

2] Voor en Nadelen (Leap Motion vs Handtracker)

3] Onderzoek Leap Motion

4] Progressie per les

5] Conclusie

6] Bronnen

*Ons Idee*

Zoals wij al eerder hebben benoemd in de inleiding, ons idee is om een instrument te bespelen doormiddel van de Leap Motion, de Leap Motion staat bekent om de handtracking en zo zijn wij al snel gekomen op het idee om het instrument een Piano te maken.

Doordat wij voor dit project geen gebruik mogen maken van Unity zal er op het begin vooral veel worden getest. Engine’s die wij bij aankondiging van het Project op het oog hadden waren Godot Engine en Unreal Engine, verder zouden we kunnen kiezen om een c# of Javascript applicatie te maken.

*Voor en Nadelen (Leap Motion vs Handtracker)*

De keuze om de Leap Motion te gebruiken komt uit een onderzoekje van voor en-nadelen van beide de Leap Motion en Handtracker.

*Voordelen Leap Motion*

- Hoge Sensitivity

- Het komt met gratis SDK (Werkt ook voor android)

- Het is makkelijk te gebruiken

- Het projecteert alle ‘joints’ in je handen waardoor accuraat de beweging meespeelt op het scherm

*Nadelen Leap Motion*

- Werkt niet evengoed op alle PC’s

- Moeite met controle houden

- Het werkt maar voor een korte periode

*Voordelen Handtracker*

- Het pakt gemakkelijk de locatie van de hand

- Werkt op lange duur nog steeds

*Nadelen Handtracker*

- Loopt achter bij snelle bewegingen

- Herkent niet alle joints in de hand

- Herkent al snel de handen niet meer

*Conclusie*

Na het onderzoeken van de voor en-nadelen van beide objecten hebben wij gekozen voor de Leap Motion, onze keuze is gemaakt uit het feit dat wij in de applicatie goed gebruik willen maken van alle posities in onze hand, elke joint die het detecteert kan belangrijk zijn om het zo echt mogelijk te laten lijken dat je daadwerkelijk piano speelt. Verder is de Leap Motion makkelijker te gebruiken en daarnaast is de korte duur geen probleem sinds dit project toch niet zo heel groot wordt.

*Onderzoek Leap Motion*

Nu de keuze naar de Leap Motion is gemaakt is het natuurlijk tijd om verder onderzoek te doen naar het apparaat en daadwerkelijk apps te testen om een goed beeld te krijgen hoe het werkt in game-vorm.

Hierin zijn wij eruit gekomen dat de Leap Motion 4 main-functies heeft.

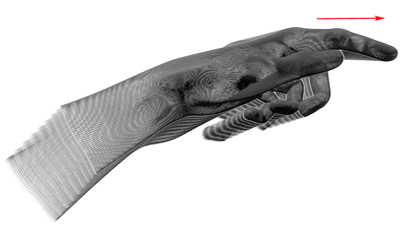
*1] Circle*

**De circle beweging kan je constant blijven doen

met je vinger of met je hand en de Leap Motion zal blijven

updaten totdat je stopt met de beweging te maken

*2] Screen Tap*

Een screen tap kan je doen door de beweging te

maken alsof je een verticaal scherm aantikt.

Er word maar 1 ding toegevoegd per keer dat

je de beweging maakt.

*3] Swipe*

Je kan de Swipe beweging maken met elke vinger

en elke kant op. De swipe beweging is continu bezig,

dus zodra je start blijft hij updaten tot hij stopt.

De beweging stopt zodra je in een andere richting beweegt

of zodra je te sloom beweegt.

*4] Key Tap*

De Tap beweging is gewoon alsof je een piano key indrukt.

De beweging voert maar 1 ding uit met deze beweging.

*Deze 4 functies van de Leap Motion zijn allen essentieel*

*voor gebruik bij ons project.*

*Progressie per les*

Hierin volgt de documentatie van de lessen waarin we aan het project werken.

*Les 1*

Het begon allemaal bij het kiezen tussen de Leap Motion of Handtracker, nadat we alle voor en-nadelen hebben bekeken zijn we bij de Leap Motion uitgekomen. Vrijwel nadat we de Leap Motion in ons handen kregen werden we al snel verwijst naar de website waarin je de SDK moet downloaden.

De keuze in SDK bestond uit een V2 versie en een V4 versie, beide voor VR of Desktop, wij waren niet van plan om VR te gebruiken dus we gingen al snel over op Desktop. De V2 versie bleek gericht te zijn op gebruik met Javascript, C# en Python, het ding met de V2 versie is dat het een Legacy Versie is dat betekent dat het veel wordt geüpdate en dus snel ‘outdated’ raakt.

V4 daarentegen is voor Unity en Unreal, dus C# en C++, bij dit project werd gezegd dat het gebruik van Unity verboden is dus de V4 versie viel daardoor wel een beetje af, Toch hebben wij besloten om de V4 versie te downloaden, omdat we snel wilden kijken hoe de Leap Motion werkt in andere apps.

De apps die wij hebben getest in deze les waren ‘Robochess’ en ‘Qubic’, deze keuze is gemaakt, omdat beide games goed gebruik maken van de main functies die de Leap Motion kan gebruiken. Robochess gebruikte het ‘Key Tap’ en ‘Circle’ element heel goed waarnaast Qubic het ‘Screen Tap’ en ‘Swipe’ element goed toepaste.

De Leap Motion hadden we al snel aan de praat, we hebben alleen wat moeite gehad op het begin van het ‘Recalibraten’ van de Leap Motion zelf om het goed te laten werken op onze laptops. Ook waren er wat errors door het spelen van de game waardoor de detectie van de Leap Motion niet zo goed werkte, het ligt waarschijnlijk aan een ‘timing-issue’.

*Les 2*

In deze les hebben wij opnieuw de Leap Motion gepakt om eigenlijk gelijk te testen hoe soepel het deze keer zou gaan, sinds we nu weten hoe het apparaat precies te werk gaat.

Tijdens het ‘Recalibraten’ van het apparaat zijn we erachter gekomen dat er een aantal opties staan in het control panel dat ervoor zorgt dat de Leap Motion beter werkt in omstandigheden met veel fel licht. Ook is er een ‘Low Resource Mode’ dat ervoor zorgt dat het niet gelijk afhaakt als je handen er niet helemaal perfect opstaan, zo krag je een veel soepeler beeld.

Wij begonnen in deze les ook met alles netjes te documenteren en zijn we begonnen aan onze presentatie die we moesten geven om onze voortgang te laten zien.

Als laatste hebben wij nog een half uur besteed om de games die we eerder hebben gedownload te testen nu het apparaat goed werkt, in dit halfuur hadden we 2 kleine errors die beide ontstonden in de eerste vijf minuten, deze errors waren heel klein en zorgde voor geen problemen, een goed resultaat dus om de dag mee te eindigen.

*Les 3*

In deze les hebben we gekeken voor opties om te checken in welke engine we willen gaan werken. Tot nu toe hebben we alleen Unreal Engine uitgeprobeerd want we hadden geen tijd om een andere engine te downloaden. We zijn hierbij op een aantal problemen gekomen zoals dat op 1 van onze computers Unreal Engine constant crashed. Ook hebben we gewerkt aan de presentatie. We hebben veel meer mogelijkheden van de Leap Motion uitgezocht en getest zoals welke bewegingen we kunnen maken en specifieke toepassing in ons eigen project.

*Les 4*

In deze les hebben wij onderzoek gedaan naar het probleem dat de vorige les ontstond, op een van onze laptops bleef Unreal Engine constant crashen en wij hebben dit op allerlei manieren proberen te fixen. Ten eerste probeerden we Unreal Engine te virifiëren om te kijken of er iets verkeerd is gedownload, daarna hebben we geprobeert om de drivers te updaten maar beide stappen kwamen helaas zonder positief resultaat. Als laatste hoop hebben wij nog geprobeert om het Anti-Virus uit te zetten sinds we hadden gelezen dat dat mischien zou werken, maar ook dit kwam zonder positief resultaat.

Zo hebben wij dus een dag meegemaakt dat eigenlijk vrij weinig resultaat opleverde. Wij zijn van plan om in de volgende les mischien te kijken of we een laptop kunnen gebruiken voor Unreal Engine en de andere voor het research naar het maken van een applicatie erin.

*Les 5*

In deze les hebben zijn we erachter gekomen dat we een andere versie van Unreal Engine moesten hebben om de plugin van de Leap Motion goed te laten werken. Nadat dit is gelukt kwam er gelijk goed resultaat uit, de Leap Motion sloot goed aan op Unreal Engine en zijn wij wat gaan kijken naar hoe Unreal Engine zelf nou precies werkt.

*Les 6*

In deze les zijn we vrijwel gelijk Unreal Engine weer ingestapt, na een replica te maken van hoe wij het er ongeveer uit wilden laten zien zagen wij helaas dat Unreal Engine heel traag is, deze engine is echt bestemt voor Goede Laptops en PC’s. We zijn dus al snel verder gaan kijken nu het nog vroeg was in het project, hierin zijn we op GoDot Engine terechtgekomen en hebben we het al heel snel aan de praat gekregen met de Leap Motion. GoDot Engine werkt wel op beide laptops en nog eens heel soepel ook.

*Les 7*

In deze les zijn we gelijk GoDot Engine in gegaan om aan de slag te gaan met ons project, nu we een goede engine hebben gevonden dat goed werkt met de Leap Motion konden we eindelijk een echt begin maken aan ons project. We zijn ten eerste gaan kijken hoe het maken van assets werkt in GoDot, maar hebben al snel gekeken voor een piano asset online die wij zouden kunnen gebruiken, omdat voor ons op het moment een resultaat behalen belangrijker is dan de ‘art’ kant van het project.

Na onderzoek te doen naar het soort scripten in GoDot en te kijken wat je er allemaal mee kan hebben we besloten om de volgende les te kijken of we geluid kunnen produceren door de ‘Key Tap’ toe te passen op de Piano.

*Les 8*

In deze les hebben we wat meer in de Piano asset gekeken, het bleek best wel gek te werken waardoor wij de asset zelf iets moesten aanpassen. Zo hebben wij in Paint3D het bestand proberen over te zetten in een ‘.obj’ bestand. Dit werkte helaas niet waardoor wij Blender hebben moeten downloaden. Nadat we het hebben gedownload vonden we al snel een oplossing op ons probleem en hebben we het bestand succesvol overgezet naar een .obj bestand waardoor we het goed konden implementeren in onze scene. Dit proces heeft helaas best lang geduurt waardoor we geen tijd meer hadden om ons volgende doel te bereiken, dus hebben we deze tijd besteed om meer onderzoek te doen naar het scripten in GoDot.

*Les 9*

Door Problemen met het openbaar vervoer kon één iemand van ons team er helaas niet bij zijn, na wat contact te houden online hebben we snel besloten dat de andere persoon zich zou focussen op de documentatie om dit echt netjes te maken, en hetzelfde met de PowerPoint presentatie. Onder deze lessen hebben we wel zoveel mogelijk contact gehouden en gediscussieerd over het proces en de verwoording in de documentatie.

*Les 10*

In deze les hebben wij meer gewerkt aan onze piano asset in GoDot waardoor we alle scripts klaar hebben staan om vervolgens geluid mee te genereren door een klik op de piano. Volgende les zijn we van plan om ook daadwerkelijk door een klik dat geluid te kunnen genereren.

*Les 11*

In deze les hebben wij gewerkt om het geluid werkend te krijgen bij aanraking van een object, al hoewel dit is gelukt, we kunnen maar één geluid afspelen, het GoDot script is nogal ingewikkeld waardoor we momenteel nog geen weg zien om alle geluiden werkend te krijgen per Piano Tile. Wij hopen dit volgende les (de laatste les) voor elkaar te krijgen

*Les 12*

In onze laatste les hebben wij gewerkt aan: Het afmaken van de presentatie, de documentatie checken en een conclusie trekken en een oplossing proberen te vinden voor ons vorige probleem in het project zelf. Wij hebben helaas het probleem niet kunnen oplossen, de Leap Motion is heel gek geprogrammeerd in GoDot waardoor we niet gemakkelijk kunnen zeggen “bij aanraking, activate child (wat een audioplayer is)”. Heel jammer dus maar wij kunnen dit wel meebrengen in ons leerproces en onze conclusie, we kijken uit naar ons resultaat.

*Conclusie*

Project Input/Output was voor ons beide het eerste project van zijn soort en hierbij hoorde dus gelijk een veel andere aanpak. Er was makkelijk te werken met de Leap Motion, het onderzoek doen ging helemaal prima en hetzelfde kan worden gezegd over de samenwerking.

Wat wij hebben geleerd van dit project is vooral hoe belangrijk onderzoek doen is, bergen aan informatie over de Leap Motion en alle bijbehorende dingen stonden ons te wachten online waar wij maar doorheen moesten zoeken om het deel te vinden waarmee wij kunnen werken.

Ook kwamen wij te weten dat het nooit in één keer, of zelfs meerdere keren goed gaat. Wij hebben continu van Editor moeten wisselen, continu andere SDK’s gedownload om uiteindelijk te komen op een Engine die fijn werkt met de Leap.

En tot slot hebben wij geleerd dat de documentatie bijhouden van een Project echt heel veel nut heeft op het werkproces bij elke les, het was heel fijn om elke keer de gelegenheid te hebben om een soort van ‘daily-standup’ te houden waarin we kort konden bekijken wat we de vorige les hebben gedaan, zo konden we heel snel daaruit verder werken.

Wij zijn ondanks dat het project niet precies ging op de manier waarop wij het wilde hebben, toch trots op ons eindresultaat. Ons doel was om muziek te genereren doormiddel van de Leap Motion, en dat hebben we gedaan.

***- Project Input/Output***

***- Job Tol / Justin van der Lans***

*Bronnenlijst*

<http://blog.leapmotion.com/getting-started-leap-motion-sdk/>

<https://developer.leapmotion.com/documentation>

<https://github.com/leapmotion/LeapUnreal/>

<https://developer-archive.leapmotion.com/documentation/csharp/devguide/Leap_Gestures.html>