



Osaamista  
ja oivallusta  
tulevaisuuden  
tekemiseen

Veli Oskari Karttunen

# Peli sokealle

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknika

Insinöörityö

21.4.2020

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                        |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|
| Tekijä<br>Otsikko                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Veli Oskari Karttunen<br>Peli sokealle |
| Sivumäärä<br>Aika                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 41 sivua<br>21.4.2020                  |
| Tutkinto                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Insinööri (AMK)                        |
| Tutkinto-ohjelma                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Tieto- ja viestintäteknikka            |
| Ammatillinen pääaine                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Pelisovellukset                        |
| Ohjaaja                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Yliopettaja Kari Aaltonen              |
| <p>Insinööriyössä selvitettiin, minkälaisen pelin voisi suunnitella täysin sokeille pelaajille, kun pelissä täytyy navigoida 3D-ympäristössä. Sitä varten luotiin ongelmanratkontapeli, jota pelataan ensimmäisessä persoonassa. Pelin esittelyversio annettiin näkövammaisten henkilöiden käsiin, testattavaksi.</p> <p>Työtä varten tutkittiin paljon muita audiop pelejä, ja niiden suunnittelun pohjalta rakennettiin yksinkertainen, toimiva peli. Tuloksessa voi 3D-äänen perusteella navigoida yksinkertaisissa sokkeloissa kohti päämäärää.</p> <p>Insinööriyön tulosten perusteella yksinkertaisia 3D-navigointipelejä voi tehdä ilman mitään visuaalista syötettä, käyttämällä oikeita suunnittelutekniikoita. Tämä saattaa olla hyödyllistä tietoa, jos kehitetään peliä näkövammaisille.</p> |                                        |
| Avainsanat                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | pele, sokea, audiopeli, 3D-navigointia |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                                               |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| Author<br>Title                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Veli Oskari Karttunen<br>A game for the blind |
| Number of Pages<br>Date                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 41 pages<br>21 April 2020                     |
| Degree                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Bachelor of Engineering                       |
| Degree Programme                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Information and Communications Technology     |
| Professional Major                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Game Applications                             |
| Instructor                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | Kari Aaltonen, Principal Lecturer             |
| <p>In this thesis we investigate whether or not it is possible to develop a game which involves navigation in a 3D environment for people, who are completely blind. For that purpose we created a first-person puzzle game, of which a demo was distributed to people who are visually impaired, to test the results.</p> <p>We will explain the details, but for the purposes of this thesis we investigated various other audiogames and based on their design, we constructed a simple, functional game. The result produced a game, in which the player can utilize 3D audio to navigate the simple mazes, towards a goal.</p> <p>From this we can conclude that it is indeed possible to create simple 3D navigation games without any visual feedback, by utilizing the right design techniques. This may be useful information, when designing a game for the visually impaired.</p> |                                               |
| Keywords                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | game, blind, audiogame, 3D navigation         |

## Sisällys

### Lyhenteet

|     |                                                          |    |
|-----|----------------------------------------------------------|----|
| 1   | Johdanto                                                 | 1  |
| 2   | Audiopelit                                               | 2  |
| 2.1 | Shades of Doom                                           | 2  |
| 2.2 | Q9 Action Game                                           | 4  |
| 2.3 | Terraformers                                             | 5  |
| 3   | Tutkimusnäkökulmaa ja esikuvallisia vaikuttajia          | 9  |
| 3.1 | Näkövammaisten äänimaailma                               | 9  |
| 3.2 | Artikkeli: kuvattomia 3D-pelejä sokeille pelaajille      | 12 |
| 3.3 | Artikkeli: Audiopelit: Hauskaa kaikille? Kaikille huvia? | 13 |
| 3.4 | Artikkeli: Audiopeljä pelaamassa                         | 14 |
| 4   | Peli sokealle-projektin lähtökohtia                      | 16 |
| 4.1 | Lähtökohtia                                              | 16 |
| 4.2 | Työkalut ja työprosessi                                  | 17 |
| 5   | Pelin toteutuksen vaiheita                               | 18 |
| 5.1 | Pelimaailma                                              | 19 |
| 5.2 | Pelaajien roolihahmo                                     | 19 |
| 5.3 | Varasto keruutoiminnolle                                 | 24 |
| 5.4 | Tason esineitä                                           | 29 |
| 5.5 | Päävalikko                                               | 31 |
| 5.6 | Äänen suunnittelu ja tuottaminen                         | 33 |
| 5.7 | Tasot                                                    | 35 |
| 6   | Testaajien palaute                                       | 37 |
| 7   | Yhteenveto                                               | 39 |
|     | Lähteet                                                  | 42 |

## 1 Johdanto

Videopeleissä visuaalinen syöte on yleisesti erittäin tärkeä. Se kommunikoi pelaajalle hyvin paljon tietoa ja hyvin nopeasti: millaisessa ympäristössä he ovat, paljonko elämäpisteitä heillä on, paljonko on ammuksia, onko vihollisia lähetyvillä, missä kyseiset viholliset ovat ja paljon muuta.

Insinööriyötä varten yritin tehdä pelin täysin sokeille. Päämääränä oli suunnitella peli, joka olisi pelattavissa ilman mitään visuaalista palautetta, ja katsoa, kuinka paljon vuorovaikutusta on mahdollista toteuttaa pelaajille, jotka eivät näe mitään.

Ääniä on jo tyypillisissäkin peleissä käytetty kommunikoidaan tietoa, mutta se harvemmin on ainoa asia, mihin pitää turvautua, joten tässä projektissa kaikkein tärkein asia on ääni. Sen täytyy tarkasti tiedottaa pelaajalle liikkumista, edistymistä, esinevuorovaikutusta ja kaikkea, mitä yleisesti todetaan näköaistilla. Joskus voi olla, että täytyy käyttää sanoin äänitettyjä vihjeitä selittämään mekaaniikkoja ja kontrolleja pelaajalle, mutta mielelläni pyrin käyttämään niitä mahdollisimman vähän.

Listaan ja kuvailen myös muita pelejä, jotka ovat yrittäneet saavuttaa samaa päämäärää. Näistä esimerkeistä tein parhaani oppia audiopelin rakentamisen perusteet.

Lyhyesti kuvailtuna, yritin luoda ongelmanratkontapelin, jossa navigoidaan 3D-ympäristöä ensimmäisessä persoonassa. Tämä ei ole todennäköisesti helpoin lähestymistapa, mutta se on hyvä haaste. Peruslähtökohtana on, että pelaajan täytyy navigoida tason aloituspisteestä loppupisteeseen, joka vie pelaajan seuraavalle tasolle. Esteinä saattaa esiintyä seiniä, ovia, kuiluja, junia tai jotain niiden tyyppistä.

Temaattisesti peli hieman imitoi peliä nimeltä *Portal*. Pelaaja on koehenkilö, jolta on revitty silmät irti ja joka pakotetaan ratkaisemaan monenlaisia testejä. Tosin *Portal*issa ei revitty silmiä irti ja tässä pelissä ei tehdä portaaleja. Mielestäni tämä teema on erityisen sopiva, koska tarkoituksenakin on kokeilla, kuinka paljon vuorovaikutusta peli pystyy toteuttamaan sokean pelaajan kanssa.

## 2 Audiopelit

Vaikka paljon omia ideoita saattaa keksiä siitä, miten kehittää peliä sokeille tällä konseptilla, on hyvä katsoa, mitä muut ovat yrittäneet, koska jos keskittyy vain omiin tekemisiinsä, voi olla, että saattaa osua vaikeuksiin, jotka muut ovat jo ratkaisseet. Audiopelienä on tehty pitkän aikaa, vaikka ne eivät ole vedonneet ihmisiin läheskään yhtä paljon kuin perinteiset audiovisuaaliset pelit. Sokeat pelaajat ovat hyvin pieni yleisö.

Tässä luvussa esittelen lyhyesti muutamaa audiopeliä, joita eräissä artikkeleissa on suositeltu, sekä muutamia kirjoitelmia jotka, saattavat olla hyödyllisiä aiheeseen liittyen.

Monet näistä peleistä ovat hieman vanhoja, mutta uskon, että niistä voi silti oppia paljon.

### 2.1 Shades of Doom

Shades of Doom nimestäkin voi päätellä, että tämä peli on perusteeltaan suunniteltu Doom-pelien pohjalta. Sen on kehittänyt GMA Games vuonna 2005. Audiogames.net-sivuilla mainitaan "release"-kentän kohdalla "2001", mutta kuvauksessa sanotaan, että alkuperäinen versio tuli vuonna 2005. Lähteiden perusteella peli on julkaistu joulukuussa 2005. [1; 2.]

Shades of Doom on ensimmäisen persoonan ammuskelupeli (first-person-shooter), mutta se toimii pelkästään äänellä. Pelissä täytyy navigoida kaikenlaisia sokkeloita tason loppuun pääsemistä varten. Vastaa saattaa tulla erilaisia vihollisia, joita täytyy ampua.

Suunnittelussa on mielestäni käytetty osittain hyviä ideoita, mutta se on ehkä hieman puutteellinen tasojen navigointiin, tai ainakin vähän epäintuitiivinen. Vihollisten ilmestyessä voi tarpeeksi helposti kääntyä niiden suuntaan ja ampua, esineiden luokse käveleminen on helppoa, kun ne tuottavat ääntä, mutta on erityisen vaikeaa tietää, missä orientaatiossa hahmo on, kun kaikilla alueilla ei aina ole äänivertailupistettä.

Seinäänkin osuessa hahmo tekee aika päälleikäyvän huudahduksen ja lopettaa kävelemisen, mikä tekee seinän seuraamisesta hyvin vaikeaa. Ei ole aivan selvää, antaako peli kulkea seinää pitkin vai ei, tai missä kulmassa hahmo päättää pysähtyä seinään.

Askeleet tuottavat kaikuvaa ääntä, jos ei ole seinää joko hahmon vasemmalla tai oikealla puolella, mikä on hyvä idea, mutta se ei mielestäni ole ihan tarpeeksi. Ääni saattaa ilmaista, jos käytävää pitkin kulkiessa tulee kävelyväylä yhdelle puolelle, mutta isommissa huoneissa siitä tulee vähän hyödytön. Jos osuu huoneen seinään eikä ole ääni-vertailupistettä kertomassa, kuinka paljon hahmo on kääntynyt, on aika vaikeaa sanoa, osuuko taas samaan seinään vai olenko nyt kääntynyt kokonaan ympäri ja kävelemässä takaisin sinne mistä tulin.

Pelissä voi käyttää joitain komentoja, jotka antavat pelaajalle äänitetyjä vihjeitä ympäristöstä, mutta tämä on mielestäni vähän epäelegantti ratkaisu. Silloin äänitetyistä kuvailuista tulee ratkaisu joka ongelmalle, mikä johtaa mielestäni liialliseen puheeseen keskittymiseen.

Taulukkoon 1 on koottu pelin hyvä ja huonoja puolia.

Taulukko 1. Pelin hyviä ja huonoja puolia.

| Hyvää                                 | Huonoa                                     |
|---------------------------------------|--------------------------------------------|
| Viholliset ja esineet tuottavat ääntä | Kaikilla alueilla ei välttämättä ole ääntä |
| Hyvä äänen laatu ja äänien valinta    | Seinillä kulkeminen hieman epäselvää       |
| Sopivan vähän kontrolleja             |                                            |

## 2.2 Q9 Action Game

Q9 Action Game-pelin on alun perin kehittänyt Blastbay Studios vuonna 2009. Vuonna 2015 se myi pelin oikeudet toiselle tiimille, joka nähtävästi ei ole kiinnostunut enää myymään peliä itsensä ja tarjoaa sen vain osana 150 dollarin hintaista Leasey-pakettiaan, joka on jonkinlainen tietokonesovellus. Pelistä on yhä saatavilla esittelyversio, jota pelasin tätä työtä varten. [3.]

Q9 Action Game on rinnalta kuvattu 2D-toimintapeli. Siinä pelataan avaruusolentona, joka on haaksirikkoutunut maapallolle, keskelle viidakkoa. Esittelyversiossa on pelattavissa kaksi ensimmäistä tasoa, viidakkotaso ja luolataso. Jokaisen tason lopussa peli antaa pistemäärän pelaajan suorituksesta.

Pelissä voi liikkua vasemmalle tai oikealle, hypätä ja käyttää varustettuja tarvikkeita. Tarvikkeita voi vaihtaa numeronäppäimillä, joihin kuuluu nuija, linko ja energiakilpi. Lingossa on rajoitettu määrä ammuksia, ja energiakilpiä on myös vain muutama.

Perspektiivin valinta on erikoinen, koska 2D-ympäristö on outo konsepti ainakin sokeana syntyneille ihmisille, tai ainakin 2D-representaatiot 3D-ympäristöistä ja esineistä, mikä ei tässä pelissä ehkä tule ongelmaksi, mutta se on kuitenkin hyvin erilaista sokeiden päivittäiseen perspektiiviin verrattuna.

Äänimaailman perspektiivissä pelaaja liikkuu sivuittain ja kuulee äänet joko oikealta tai vasemmalta, riippuen siitä, kummalla puolella hahmoa äänen lähde on. Eri viholliset tuottavat erilaista ääntä, ja kun ne ovat tarpeeksi lähellä, pelaaja voi hakata ne nuijallaan.

Kaikkein vaikein este, johon pelaamisen aikana törmäsin, olivat kuilut. Kun kuilu alkaa olla lähetyvillä, sen suunnasta kuuluu tuulen ääni, ja sitten kun on aika hypätä, askeleet alkavat tuottaa mutaista ääntä. Jos pelaaja pitää oikeaa nuolinäppäintä pohjassa liikkumaan oikealle ja hyppää, hahmo lopettaa liikkumisen. En ole koskaan pelannut, nähnyt tai kuullut pelistä, joka operoisi tällä tavalla, joten tämä epätavallinen oikku turhautti.



Peli on mielestäni aika hyvä yritys, vaikka siinä onkin parantamisen varaa. Jos hypyn korjaa toimimaan niin kuin kaikissa muissakin 2D-rinnalta kuvaituissa peleissä, peli olisi vielä parempi. Niin kuin aikaisemmin mainitsin, perspektiivi voi olla vähän erikoinen sokeana eläneille, mutta helposti totuttavissa.

Taulukkoon 2 on koottu pelin hyviä ja huonoja puolia.

Taulukko 2. Taulukkoon koottu Q9 Action Gamen hyviä ja huonoja puolia.

| Hyvää                                 | Huonoa                                  |
|---------------------------------------|-----------------------------------------|
| Hyvä äänen laatu ja valinta           | Kuilujen toteutus vähän kömpelö         |
| Tarpeeksi vähän eri kontrolleja       | Hyppy toimii epäintuitiivisella tavalla |
| Erikoinen lähestymistapa audiopelille |                                         |

### 2.3 Terraformers

Terraformers-pelin on kehittänyt ruotsalainen tiimi nimeltä Pin Interactive. Se julkaistiin vuonna 2003 PC:lle. Peli on ensimmäisen persoonan seikkailupeli. Alun perin peli julkaistiin maksullisena, mutta nykyään se on saatavilla kaikille. [4.]

Peli on suunniteltu äänillä pelattavaksi, mutta hyödyntää myös visuaaleja vaihtoehtoisesti, joten se on yksi harvinainen audiopeli, josta voin liittää kuvia (kuva 1). Oletettavasti visuaalinen vaihtoehto on tarkoitettu erittäin huononäköisiä pelaajia varten, mutta he todennäköisesti laittavat kuvasyötteen pois päältä heti, kun he huomaavat, että näkökenttä on niin kapea, että tuntuu siltä, kuin kävelisi kiikarit päässä. Visuaaleihin on panostettu yllättävä määrä vaivaa, kun ottaa huomioon, että peli on suunniteltu sokeille.



Kuva 1. Terraformers-pelin tarinan johdanto videosta [4].

Lyhyesti kuvailtuna pelin tarina on sellainen, että robotit päättivät kapinoida ja ovat vanginneet professori Van Langen hänen laboratorioonsa ja nyt pelaajan täytyy kukistaa robotit ja vapauttaa tämä professori.

En ollut erityisen vaikuttunut pelistä. Pelin navigointijärjestelmä on hyvin monimutkainen, ja se vaatii pelaajaa painamaan mieleen useita eri komentoja, esimerkiksi

- Shift + C = Kompassi päälle/pois päältä.
- M = Toista tutoriaaliviesti, mutta ei viimeisintä viestiä, vaan vain ensimmäinen viesti joka annettiin, kun saapuu huoneeseen. Tämä tarkoittaa, että jotkut ohjeet annetaan vain kerran.
- Shift + B = Sonar päälle/pois päältä (laite, joka kertoo, kuinka kaukana pelaaja on edessä olevasta esineestä), b = kertoo sanoin, mikä esine on edessä.
- V = Kerro pelaajahahmon nykyinen sijainti koordinaateissa.
- S = Tallenna peli.
- J = Visuaalit päälle/pois päältä.

- Z = Tuo lista, jossa on kaikki huoneen esineet, ja jos painat välilyöntiä, saat tietää, kuinka monta yksikköä missäkin ilmansuunnassa esine on (esim: "west 31, north 10").
- END = Sulje lista.
- G = En muista.
- F = Kerro, mikä tavara on kädessä.
- ESC = Sulkee pelin heti ilman mitään vahvistusta.
- Q = Heittää pelaajan takaisin päävalikkon ilman mitään vahvistusta.
- S = Tallenna peli.
- X = Äänitetty huoneenkuvailu.
- Numeronäppäimet asettavat hahmon tiettyyn ilmansuuntaan.
- Välilyönti = Yleinen "käytä asiaa edessä"-nappi.
- Enter = Avaa tavaralaukkulista, käytä hiiren rullaa navigoimaan listaa ja välilyöntiä kaksi kertaa ottamaan kyseinen tavara käteen. Kun olet poissa listasta, käytä tavaraa painamalla välilyöntiä. Maailmassa olevia esineitä ei voi käyttää välilyönnillä, jos on tavara kädessä. En tiedä, miten ottaa tavaraa pois kädestä.

Kuvassa 2 näkyy ampumarata pelin tutoriaalista.



Kuva 2. Ampumarata pelin tutoriaalista [4].

Askeläännet jatkuvat, vaikka kävelee seinään, hyvin paljon ääniä kuuluu joka suunnasta samaan aikaan, tutoriaalin viestit ovat erittäin pitkiä. Tämä kaikki tuottaa mielestäni hyvin raastavan kokemuksen, ja syy, miksi listasin suurimman osan kontrolleista aikaisemmin, oli, että näkökulmani tulisi selväksi.

Näin monen toiminnon vaatiminen ei ole hyvä idea. Uskon, että parasta olisi rajoittaa kontrollit perusliikkumiseen (eteen, taakse, vasen käännös, oikea käännös) ja yleinen "käytä"-nappi. Ehkä voisi olla vielä lisää toimintoja numernäppäimille, mutta siihen vetäisin rajan, varsinkin kun monet tämän pelin toiminnoista olisivat helposti automatisoitavissa. Jos täytyy tuoda enemmänkin toimintoja, ne kannattaisi esitellä pelaajalle hyvin hitaassa tahdissa.

Jos on kärsivällisyyttä, varmaan pärjää, mutta en suosittelisi tätä peliä monille.

Taulukkoon 3 on koottu pelin hyviä ja huonoja puolia.

Taulukko 3. Terraformers-pelin hyvät ja huonot puolet.

| Hyvää                                                            | Hunoa                                                                           |
|------------------------------------------------------------------|---------------------------------------------------------------------------------|
| Vaihtoehtoiset visuaalit, joihin on laitettu jonkun verran työtä | Liian paljon kontrolleja                                                        |
|                                                                  | Ei voi tietää käveleekö seinää päin vain yksinkertaisesti kävelemällä sitä päin |
|                                                                  | Tutoriaali viestit ovat hyvin pitkiä ja niitä on hyvin monta                    |
|                                                                  | Navigointi on hyvin työlästä                                                    |

### 3 Tutkimusnäkökulmaa ja esikuvallisia vaikuttajia

#### 3.1 Näkövammaisten äänimaailma

Sokeutta voi syntyä useanlaisista syistä, ja se voi tulla monessa muodossa: esimerkiksi trakooma, glaukooma, nystagmus, retinitis pigmentosa, sokeritaudista tai perusonnettomuuksista. [17; 18; 19; 20; 21.] Näön puutteen taso voi vaihdella eri variaatioiden takia paljon, mutta niistä suurinta osaa ei todennäköisesti tarvitse ottaa huomioon, jos suunnittelee pelin alusta asti täysin sokeille.

Pelit ovat aika laajalti kommunikointia niiden suunnittelijan ja pelaajan välillä, jotta pelaajat pystyvät mahdollisimman helposti ja intuitiivisesti saamaan otteen pelin pelaamisesta. Itse en ole sokea, joten se on tärkeää, että ainakin yritän tutustua sokeiden maailman hahmotukseen, jotta pystyn kommunikoimaan pelaajille pelin mekaaniikkoja.

Yritin parhaani mukaan löytää tutkimuksia aiheesta, mutta hyvin monet niistä ovat maksullisia tai hyvin vanhoja (muutamit ensimmäisimmät haut Google Scholarissa olivat vuosilta 1950, 1940 ja 1728), joten artikkelien määrä oli aika rajoitettu. Yritin sitten jonkin verran lisätä ymmärrystäni sokeiden ihmisten henkilökohtaisilla kertomuksilla, mutta niiden hyöty voi olla rajallista.

Tommy Edison on sokeana syntynyt henkilö, jolla on YouTube-kanava, jossa hän usein kertoo tarinoita ja omia kokemuksiaan sokeana ihmisenä. Muutamassa niistä hän puhuu omasta ymmärryksestään 2D-maailmasta. Piirtämisen yhteydessä hän on sanonut (käännös englannista): "Piirtäminen on minulle vähän outo juttu. Siis kaikki minun maailmassani on kolmiulotteista. Kaikki mihin kosken tai koen. Joten on hyvin vaikeaa minulle kuvitella miten kolmiulotteisen jutun voi laittaa litteälle paperille. Siinä ei ole mitään järkeä minulle". [9; 10.]

Tommy Edisonin kokemus kertoo, että 2D-representaatiot ovat vieras konsepti, minkä takia itse välttäisin tekemästä 2D-peliä sokeille, mutta mitä olen Q9 Action Gamestä nähnyt, se ei ole silti mahdotonta.

Edisonilta on kysytty myös, millä tavalla hän "visualisoi" esineitä muistista, mihin hän vastasi: "Koska olen ollut syntymästä asti sokea, en oikein visualisoi mitään, koska en tiedä miten nähdä ja alitajuntani ei tiedä miten nähdä. Kun ajattelen jotain, niin arvelen että ajattelen sen ominaisuuksia. Ajattelen mikä se juttu on tai sen muotoa tai sen kokoa. Ajattelen muistoani siitä. Esimerkiksi jos puhut pesäpallosta, muistan miltä sellainen tuntuu". [11.]

Taktiilinen tunne on iso osa sokeiden ihmisten kokemuksesta, jota en tietokonepelillä oikein pysty replikoimaan, joten äänten täytyy toimia senkin edestä. Jos äänien haluaa olevan mahdollisimman nopeasti tunnistettavissa, ne kannattaa perustaa johonkin oikean elämän kokemukseen, jonka pelaaja pystyy hyvällä todennäköisyydellä samaistamaan johonkin yleiseen asiaan. Tähän tekniikkaan myös turvauduttiin aikaisemmin mainitsemissani artikkelissa "Audio Games: Fun For All? All For Fun?". [7.]

Seuraavat tulokset tulevat Joe Hansonilta, joka on tiedekirjoittaja, biologi ja YouTube-henkilö. Hän on tehnyt väitöskirjansa Teksasin yliopistossa Austinissa. Mutta koska hänen videonsa eivät teknisesti ole tieteellisiä artikkeleita, kutsun hänen sanojaan kertomuksiksi. [12.]

Yhdessä videoistaan (How Blind People See With Sound) Hanson kertoo, kuinka ihmisetkin (eikä vain lepakot ja delfiinit) pystyvät hyödyntämään aktiivista kaikuluotausta rajallisessa muodossa. Videossa on mukana myös kaksi sokeaa ihmistä

demonstroimassa tätä kykyä. Molly Burke, tavallinen sokea nainen, ja Brian Bushway, jonka titteli on ”Perceptual Navigation Educator” tai ”aistinavigointikouluttaja”. Bushway on nähtävästi mestari kaikuluotauksessa ja pystyy jopa ajamaan pyörää sokeudestaan huolimatta. Videossa hän opettaa Hansonille ja Burkelle kyvyn perusteita. [13; 14.]

Kaikuluotaus on kyky hahmottaa ympäristöä käyttämällä ääntä. Kun äänen vibraatiot osuvat erilaisiin esteisiin, ne tuottavat kaikua. Tämä kaiun muoto voi vaihdella riippuen esteen etäisyydestä, muodosta, materiaalista ja muusta. [15.] Termiä ei tunnettu vielä sata vuotta sitten, ja hyvin pitkään sokeiden ihmisten kyvyn huomata äänettä esteitä luultiin perustuvan iholla tunnusteluun, koska esineiden ajateltiin tuottavan jonkinlaista painetta. [16.] Yleensä näkeville ihmisillä ei ole tapana kiinnittää huomiota näihin kaikuihin, joten heidän aivoillaan on tapana vaimentaa efektiä. Passiivisessa kaikuluotauksessa havaitaan ympäristöä käyttäen muita ympäristön ääniä. Aktiivisessa kaikuluotauksessa haivitsija tuottaa ääniä itse, vaikkapa kieltä naksuttelemalla.

Molly Burkea oli jo valmiiksi koulutettu passiivisessa kaikuluotauksessa, joten hän oppi aktiivisen hieman nopeammin kuin Hanson. He aloittivat pitämällä kulhoa jollain puolella päätä ja päämääränä oli löytää se. Sitten he kokeilivat, miltä ääni kuulostaa seinän vieressä ja oviaukon kohdalla. Näiden testien perusteella näyttäisi siltä, että kaikuluotauksen kyky on mahdollista ihmisille. Se vaatii paljon harjoittelua, mutta ei mahdottoman paljon.

Ihmisten kaikuluotaus tuntuu siltä olevan sokeiden keskuudessa vähemmistössä syystä tai toisesta. Vaikka tämä saattaa olla hyvinkin hyödyllistä tietoa audiopelin rakentamisessa, sitä ehkä kannattaa käyttää lisäominaisuutena, eikä pakollisena pelin edistymistä varten. Shades of Doom yritti hieman hyödyntää kaikuluotauksia askelissa, joita pelaajan täytyi käyttää tunnistaaksensa sivuovia käytävällä.

Todennäköisesti insinööriyössä ei ole aikaa implementoida samanlaista järjestelmää, mutta se olisi kyllä erittäin hyödyllinen huomiokykyisempiä pelaajia varten. Teoriassa olisi varmaan mahdollista myös tehdä peli, joka opettaisi kaikuluotauksia, mutta se on tämän työn tavoitteiden ulkopuolella.

### 3.2 Artikkelit: kuvattomia 3D-pelejä sokeille pelaajille

Seuraavaksi esittelen muutamia artikkeleita, joissa on tutkittu audiopelien ja jotka saattavat olla hyödyllistä luettavaa sellaisten suunnittelussa.

”Video-less’ 3D games developed for blind players” on BBC-artikkeli, joka on julkaistu vuonna 2014. Siinä puhutaan suurimmaksi osaksi ranskalaisesta audiopelistä nimeltä A Blind Legend, joka kertoo tarinaa ritarista, joka on menettänyt näkönsä ja harhailee metsässä yrittämässä löytää vaimonsa kidnappaajat, kun hänen tyttärensä johtaa suuntaa. Peli julkaistiin mobiililaitteilla vuonna 2015 ja Steamissä vuonna 2016. Pelistä on myös esittelyversio saatavilla julkaisijan sivuilla. [5; 6.]

Artikkelissa myös puhutaan kaksikorvaisesta teknologiasta, joka yrittää imitoida luonnollista ihmisen kuulokykyä. Ääntä on nauhoitettu kahdella mikrofonilla, luomaan 3D-audiota. Itselläni ei ole tarvikkeita taikka budjettia tähän tekniikkaan, mutta Unityssa voi tehdä perinteisestikin tallennetusta äänestä 3D-äänenlähteen. Arvelisin, että se tuottaa huonompaa laatua, mutta työskentelen, millä voin.

Kokeilin myös tämän pelin esittelyversiota, mutta en maininnut sitä aikaisempien otsikoiden joukossa, koska siitä ei ole paljoa sanottavaa. Luulisin, että itse peli on huomattavasti monipuolisempi kuin esittelyversio, mutta 0.2-versio vaikutti yksitoikkoiselta vuorovaikutuksen tasolla.

Tyttäreni huutaa ranskaksi jotain. Oletettavasti ”tule tänne”. Kun saavut tarpeeksi lähelle häntä, hän kävelee muualle. Sitten sama tehdään uudestaan. Silloin tällöin ympäristön äänissä kuulee muutoksia, kuten lintujen laulua, joen virtausta tai kärpäsen surinaa, mutta en oikein pelatessa saanut mitään tunnetta edistymisestä tai pelin tilanteen kehittymisestä. Seurasin tyttäätä aika hyvän aikaa, mutta mitään uutta pulmaa ei tuntunut tulevan eteen, joka olisi muuttanut pelaamisen strategiaa millään tavalla. Jos sellaista oli tulossa, siihen kului pitkä aika. Steam-arvosteluissa kyllä puhutaan myös taistelujärjestelmästä, joten oletettavasti se on ominaisuus, joka on vain myytävässä versiossa.

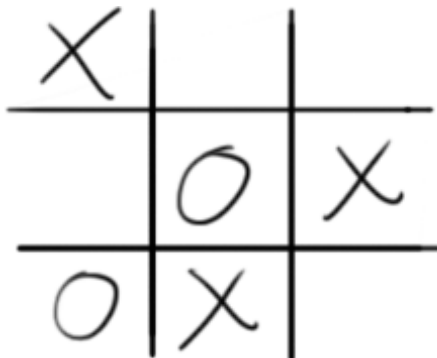


### 3.3 Artikkel: Audiopelit: Hauskaa kaikille? Kaikille huvia?

”Audio Games: Fun for All? All for Fun?” julkaistiin vuonna 2003. Sen ovat kirjoittaneet Sue Targett ja Mikael Fernström. Siinä tutkitaan, olisiko mahdollista luoda viihteellisiä audiopelisiä, jotka eivät käytä yhtään puheen ääntä ja jos sellaiset pelit olisivat hyödyllisiä taitojen kehityksessä tai terapeuttisissa tarkoituksissa. [7.]

Kirjoittajat kehittivät tutkimusta varten kaksi yksinkertaista tietokonepeliä, Tic Tac Toe ja Mastermind.

**Tic Tac Toe**, tunnetaan myös nimellä ”Os & Xs” (”ristinolla” suomeksi), on peli jossa pelaajat vuorotellen asettavat X:n tai O:n 3 x 3 -ruudukolle, kunnes yksi pelaaja onnistuu asettamaan kolme omaa merkkiään jonoon (kuva 3).



Kuva 3. Yleinen visuaalinen representaatio ristinollasta.

Kirjoittajien kehittämässä versiossa valittiin kolme ääntä esittämään kolmea vertikaalista saraketta ja eri sävelkorkeuksilla horisontaalisia sarakkeita. Kun pelaaja teki siirron, peli tuottaisi kaksi ”swish”-ääntä kuvastamaan X-merkkiä. Kun tietokone teki siirron, peli tuottaisi O-tyyppisen äänen, joka vielä kuulosti hieman vihollismaiselta, kuvastamaan vastustavan pelaajan siirtoa. Peli tuottaisi eri äänen riippuen siitä, mikä alkio oli vapaa, vastustajan ottama vai pelaajan ottama, mutta he pitivät yhä sävelkorkeudet samana. Peliä navigoitiin nuolinäppäimillä valitsemaan eri alkiot ja enter-näppäimellä asetettiin X-merkki vapaalle alkioille.

**Mastermindissa** yksi pelaaja valitsee yhdistelmän värejä ja toisen pelaajan täytyy arvata kyseinen väriyhdistelmä muutaman yrityksen sisällä. Joka kerta, kun pelaaja tekee arvauksen, he saavat tietää, kuinka monta väriä olivat ”oikea väri ja oikeassa paikassa”, ”oikea väri mutta väärässä paikassa” ja ”väärä väri”.

Yhdistelmässä on kolme väriä, joiden kuvastamista varten he valitsivat kolme eri ääntä. Pelaaja yksinkertaisesti valitsee enter-näppäimellä, mikä otetaan ensimmäisenä, mikä toisena ja mikä kolmantena. Tämän jälkeen tietokone tiedottaa pelaajalle eri ääniä käyttäen, kuinka oikeassa he olivat.

”Oikea väri ja oikeassa paikassa” esitettiin korkeaäänisen kellon soittolla. ”Oikea väri väärässä paikassa” oli nykäisty naru. ”Väärä väri” oli uudelleenkäytetty O-ääni tekijöiden Tic Tac Toe-projektista.

Testailun jälkeen tultiin siihen tulokseen, että audiopeljä tosiaan voidaan rakentaa käyttämättä ollenkaan puheen ääntä. Vaikka mielestäni kaikki pelilajit eivät siihen aina pysty, tämä kirjoitelma tehokkaasti korostaa äänen suunnittelun tärkeyttä ja kuinka lähestyä sen perusteita.

#### 3.4 Artikkel: Audiopeljä pelaamassa

”Playing Audio-Only Games”-artikkelin ovat kirjoittaneet saksalaiset Niklas Röber ja Maic Masuch, jotka ovat molemmat tohtoritason tutkijoita (kuva 4). [8.]



Kuva 4. Niklas Röber (vasen) ja Maic Masuch (oikea) [8].

Artikkelissa käydään läpi useita eri suunnittelutekniikoita äänipeleille, ja niistä useat muistuttavat aika paljon omia ideoitani, kuten esim. virtuaalisen kepin käyttöä ja ääni-vertailupisteitä. Siellä on myös todettu samanlaista asiaa, mitä huomasin *Shades of Doomia* pelatessa: "...the game (you're making) should never be silent or have quiet spots, as this would hinder in orientation and navigation". (Käännös: "...pelissä (jota olet tekemässä) ei pitäisi koskaan olla äänettä tai hiljaisia paikkoja, koska tämä vaikeuttaisi suunnistamista ja navigointia")

Suositteluja tekniikoita ovat seuraavat:

- 3D-ääni
- palaute esineiden vuorovaikutuksesta
- ympäristön kuvailu (puheella tai ilman)
- ääni-maamerkit
- tiellä edistymisen tiedotus
- tasapaino toiminnon ja estetiikan välillä.

Artikkeli suosittelee käyttämään vain niin monta tekniikkaa kuin tarvitsee ja pitämään äänien määrän rajoitettuna, jotta pelaaja ei hämmenny.

Tämäkin on mielestäni erinomainen artikkeli audiopelien kehittämisen perusteiden tiedostamiseen.

## 4 Peli sokealle–projektin lähtökohtia

### 4.1 Lähtökohtia

Insinöörityön päämäärä oli luoda 3D-ympäristö, jossa täysin sokea ihminen pystyy navigoimaan. Ensimmäisenä askeleena täytyi keksiä, miten liikuttaa hahmo pisteestä A pisteeseen B sellaisella tavalla, että pelaaja pystyy ymmärtämään, mitä tapahtuu.

Tietysti monet vanhat suunnittelutekniikat audiovisuaalisista peleistä ovat hyödyllisiä tässäkin projektissa. Yksi niistä on konseptien esittely yksi kerrallaan, turvallisessa ympäristössä. Tämä tarkoittaisi sitä, että ei kannattaisi laittaa loppupistettä, seiniä, ovia, esineitä ja vihollisia heti alkuun ensimmäiselle tasolle. Sen sijaan kannattaa aloittaa perusteista, joka on tässä tapauksessa käveleminen alkupisteestä loppupisteeseen.

Tulin siihen tulokseen, että tason päämäärän täytyy tuottaa ääntä, jotta pelaaja pystyy suunnistamaan sitä kohti tai ainakin selvittämään, mihin suuntaan hahmo on orientoitunut. Liikkuessa tuotetaan askelten ääniä, jotta pelaaja tietää, että hahmo liikkuu. Ensimmäisessä tasossa on vain loppupiste ja aluetta piirittävät seinät. Täällä pelaajalle kerrotaan, miten liikuttaa hahmoa ja että kyseessä on ensimmäinen taso, mutta päämäärä jätetään arvoitukseksi. Suunnittelussa idea on, että pelaaja kuulee loppupisteen huminan sen suunnasta ja liikkuessaan huomaa, että ääni tulee tietyistä pisteistä. Koska tasolla ei ole mitään muuta äänenlähdettä tai vuorovaikutteisia elementtejä, pelaajan pitäisi teoriassa yrittää lähestyä sitä, minkä jälkeen hän kuulee: "taso 2". Tämä kommunikoi pelaajalle, että edistystä on tapahtunut.

Sitten tulee ensimmäinen este, seinä. Se on staattinen objekti, joka ei itsenäan tuota ääntä ja jos siihen yrittää laittaa jonkinlaisen kokoaikaisen äänen kertomaan pelaajalle missä se on koko ajan, pelistä tulee todennäköisesti liian äänekäs.

Käsittääkseni tämän ongelman voi ratkaista kahdella eri tavalla (tai molemmilla). Ensimmäisenä, askelten tahti hidastuu silloin, kun hahmo hidastuu, mikä tiedottaa pelaajalle, että nyt ei kuljeta täydellä nopeudella. Toiseksi, kun hahmo liikkuu seinää pitkin, sen suunnasta voi kuulla, kun se harjaa hahmon olkapäätä. Nämä antavat

pelaajalle tiedon, milloin seinään on osuttu, sekä ohjeen käyttää seinää hyväksi navigoinnissa kulkemalla sitä pitkin. Tällöin esteestä tuleekin hyödyllinen työkalu.

Hyvin yleinen este peleissä on myös lukittu ovi ja avain. Yksinkertainen asia, mutta tuo yllättävän paljon vuorovaikutusta. Lukitun oven löytäminen toimii päämääränä, ja sen avaimen tarve rohkaisee pelaajaa tutkimaan tasoa läheisemmin.

Ajattelin, että olisi hyvä idea esitellä oven ja avaimen konsepti loppupisteiden ja seinien jälkeen. Ovi kannattaa asettaa tason alkupisteen ja loppupisteen väliin, jotta se on todennäköisesti ensimmäinen asia, johon pelaaja törmää. Ovi antaa virhe-äänen, joka tiedottaa pelaajalle oven olemassaolosta. Ääni tälle ovelle kannattaa valita harkiten, jotta pelaaja pystyy luonnollisesti ja ajattelematta vaistoamaan, että kyseessä on ovi, joka on lukittu. En tiedä, onko tuottamani ääni tarpeeksi hyvä tähän tarkotukseen, mutta se on hyvin samanlainen muihin ”lukittu ovi” -ääniin, jonka uskon olevan kelvollinen.

Avaimen täytyy tuottaa ääntä, jotta pelaaja pystyy löytämään sen, mutta pitääkseni äänimaailman jonkinlaisessa kurissa, päätin, että olisi hyvä idea rajoittaa esineiden äänietäisyys. Jotta pelaajan ei tarvitsisi sählätä minkään valikon kanssa vain avatakseen oven (niin kuin *Terraformersissa*), se aukeaa heti, kun pelaaja kulkee ovea kohti avaimen keräämisen jälkeen, niin kuin esim. vanhoissa *Doom*-peleissä.

Mielestäni nämä antaisivat projektille hyvän perustan kokeiluvärsiä varten. Vuorovaikutteisia elementtejä ei ole paljoa, mutta niitä voi yhdistellä tuottamaan ainakin muutama taso. Muitakin esteitä ja esineitä minulla oli jo mielessä, mutta päätin aluksi keskittyä näihin, koska muut saattavat tarvita harkintaa siitä, miten esitellä ne pelaajalle.

#### 4.2 Työkalut ja työprosessi

Projektissa käytettiin Unity-pelimoottoria. Peli kehitettiin PC:lle, mutta se on pelattavissa myös ohjaimella. Jouduin välillä työskentelemään kannettavalla tietokoneella ja välillä kotikoneella, ja siksi pidin projektia GitHubissa.

Tehtävien ja päämäärien seuraamista varten olisi varmaan kannattanut käyttää Trelloa tai edes muistilappuja, mutta taisin enimmäkseen pitää kaiken päässäni tai jonkin aikaa joitain asioita listoina tekstitiedostossa.

Äänien nauhoittamista varten käytin puhelintani ja tuottamiseen kaikenlaisia esineitä, mitä löysin kotoani: puulastaa, laseja, kenkiä, vaatteita, pianoa jne. Editoin ääniä käyttämällä ohjelmaa nimeltä Audacity.

Työtä tehtiin aina, kun löysin aikaa töiden jälkeen, noin neljä tuntia päivässä.

Pelin repliikkien toteutuksessa käytettiin ulkopuolista ääninäyttelijää. Lisäsin hänen nauhoituksissa pari äänisuodatinta Audacityllä tekemään siitä hieman robottimaisen.

## **5 Pelin toteutuksen vaiheita**

Halusin pelin sisältävän navigointia 3D-ympäristössä. En ollut ihan varma, kuinka paljon interaktiota voi saada aikaan sokealle sellaisessa tilassa, joten idea oli vähän selvittää sitä tällä projektilla.

Pelimoottoriksi valitsin Unityn, koska minulla oli sen käytöstä huomattavasti enemmän kokemusta muihin pelimoottoreihin verrattuna.

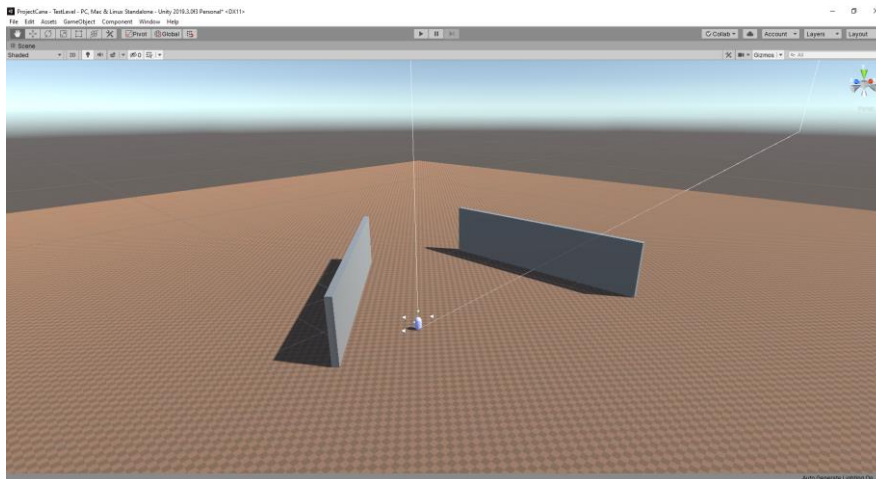
Päätin tuottaa kaiken äänen itse, koska kokemukseni sillä alueella on tähän asti ollut rajoitettua, ja tämä olisi hyvä tilaisuus opetella.

Jokaisen tason päämäärä on päästä sen loppuun. Navigointia varten täytyy kuunnella askelien nopeutta, rapinaa seinistä ja huminaa loppupisteen suunnasta. Kontrollien määrä pidetään hyvin rajoitettuna, jotta pelaajan ei tarvitse huolehtia niiden muistamisesta.

Tätä osiota kirjoittaessa oletan, että lukija ei tunne Unity-pelimoottorin toimintoja erityisen hyvin, joten selitän monia sen ominaisuuksia yksityiskohtaisesti.

## 5.1 Pelimaailma

Unityssa aloitetaan luomalla ”näkyvä”, joka on yksittäinen taso, jossa kaikki objektit, joita aiotaan käyttää kyseisessä tasossa, sijaitsevat. Aloitin luomalla näkymän nimeltä ”TestLevel”, johon laitoin vain kaksi seinää, ja lähdin rakentamaan pelaajan hahmoa (kuva 5).



Kuva 5. Testaustaso.

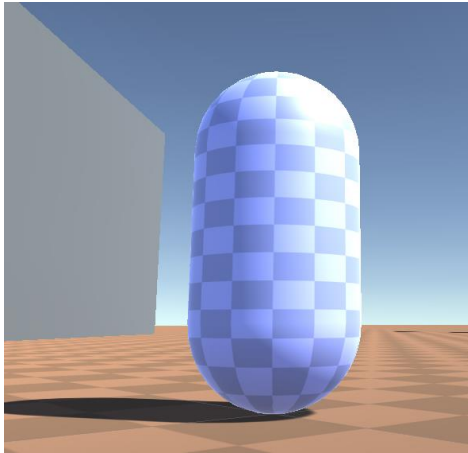
Testitaso oli tarkoitettu vain kokeilemaan navigoinnin perusominaisuuksia. Se ei ole pelattavissa lopullisessa versiossa. Myöhemmin loin näkymät ”Level1”, ”Level2”, ”Level3”, ”Level4” ja ”MainMenu”:n, joita pelissä sitten käytetään.

Unityssa tulee mukana kaikenlaisia ”komponentteja”, joita voi liittää peliobjekteihin. Ne antavat liitetyle objektille erilaisia ominaisuuksia, kuten esimerkiksi fyysistä kollisiota, painovoimaa, näkyvä mesh, äänenlähteitä ja liikkumista. Eri komponentteja voi myös koodata itse C#-kielellä, mikä on aika tarpeellista tällä pelimoottorilla, varsinkin kun monia Unityn omia komponentteja täytyy silti käsitellä koodin puolella, jos niiden halutaan toimivan toistensa kanssa.

## 5.2 Pelaajien roolihahmo

Pelaajan ruumis on yksinkertainen kapseli, johon on liitetty CharacterController-komponentti, joka on Unityn kehittämä järjestelmä. Minun silti täytyi luoda oma

komponentti nimeltä `PCC_PlayerController`, joka katsoo napinpainalluksia ja sitten käyttää `CharacterController`in toimintoja niiden mukaan. Pelaajan hahmo näkyy kuvassa 6.



Kuva 6. Pelaajan hahmo.

Esim. jos pelaaja painaa W-, S- tai ylös/alas-suuntanäppäimiä (kaikki syötteet, jotka on asetettu "Vertical axis" -nimien alle), `PlayerController` yksinkertaisesti kutsuu `CharacterController`in "Move"-funktiota ja antaa sille float-yksiköissä määrän, minkä verran liikkua. Samoin jos pelaaja painaa A-, D- tai vasen/oikea-suuntanäppäimiä (kaikki syötteet jotka on asetettu "Horizontal axis" -nimen alle), pelaaja kääntyy vasempaan ja oikeaan. Esimerkkikoodissa 1 on liikkumis-funktio, jota kutsutaan joka framellä.

```
protected virtual void Movement ()
{
    // Get vertical input, multiply it by speed and delta time and call
    CharacterController's Move-function

    Vector3 moveDirection = new Vector3(0.0f, 0.0f, GetVerticalInput());
    moveDirection *= MovementSpeed;
    moveDirection = this.transform.TransformDirection(moveDirection);

    movementComponent.Move(moveDirection * Time.deltaTime);
}
```

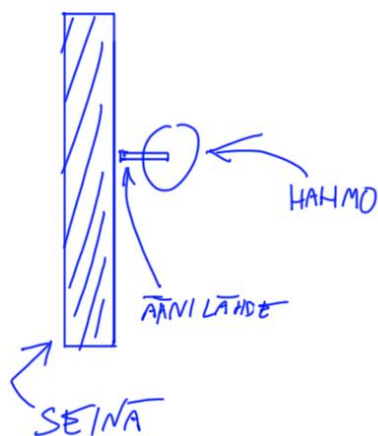
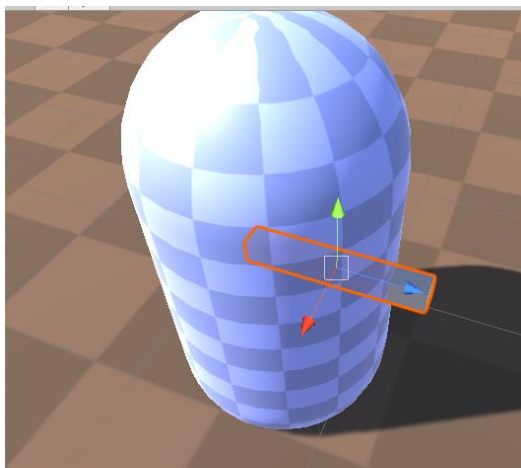
Esimerkkikoodi 1. Liikkumis-funktio, jota kutsutaan joka framellä.



Yritin alun perin luoda oman PCC\_MovementComponentin sen sijaan, että olisin käyttänyt CharacterControlleria, mutta en saanut sen toimintoja senlaatuiseksi kuin halusin ainakaan lyhyellä aikavälillä, päätin käyttää Unityn valmista komponenttia.

Kamera on laitettu pelaajan sisälle kuvastamaan ensimmäisen persoonan perspektiiviä. Tällä ei ole paljoa väliä sokealle pelaajalle, mutta kamerassa on kiinni Unityn "AudioListener"-komponentti, mikä tarkoittaa, että kaikki äänet kuuluvat sen omistavan objektin perspektiivistä. Äänet, jotka tulevat kameran oikealta puolelta, kuuluvat oikealta, äänet, jotka tulevat kameran vasemmalta puolelta, tulevat vasemmalta, jne.

Sitten tein tuon vähän huonosti nimetyn "DetectorComponent" in (kuva 7). Sen idea on tuottaa ääntä silloin, kun pelaaja liikkuu seinää vasten, mikä kertoo, missä suunnassa seinä on.



Kuva 7. DetectorComponentin kuvaus.

Kun hahmo osuu seinään, sen sisällä oleva ”kompassi” osoittaa törmäyksen suuntaan ja sen päässä oleva äänenlähde tuottaa sopivaa ääntä. Tämän toiminnon voi todennäköisesti implementoida ilman tämmöistä GameObject-häkkyrää, mutta tämä oli nopea ja helppo toteutus.

Seuraavaksi lähdin tekemään askeljärjestelmää. Sen idea oli tosiaan tuottaa askelääniä hahmon nopeuden mukaan. Mitä nopeammin hahmo liikkuu, sitä useammin askeleet tuottavat ääntä. Sama myös päinvastoin, kun mennään hitaammin. Tämän idea on kertoa pelaajalle, kuinka nopeasti hahmo liikkuu, ja antaa vihje, jos pelaaja juokseekin jotain kohti.

Loin ja lisäsin hahmoon komponenttiluokan nimeltä ”FootstepComponent” tätä ideaa varten (esimerkkikoodi 2). Se myös tuottaa eri äänen joka toisella askeleella imitoidakseen vasemman ja oikean jalan askelia.

```
public class PCC_FootstepComponent : MonoBehaviour
{
    /* [poistettua koodia, lyhentämisen takia] */

    protected virtual void FixedUpdate()
    {
        if (IsCharacterMoving())
        {
            FootStep();
        }
    }

    protected virtual void FootStep ()
    {
        if (!audioSource) return;
        if (audioSource.isPlaying) return;

        footStepTimer.SetDuration(FootStepIntervalCalculation());

        if (!footStepTimer.IsTimerComplete()) return;
        if (rightStep)
        {
            audioSource.PlayOneShot(footStepRight_clip);
        } else
        {
            audioSource.PlayOneShot(footStepLeft_clip);
        }
        rightStep = !rightStep;

        footStepTimer.ResetTimer();
    }
}
```

```

protected virtual float FootStepIntervalCalculation ()
{
    float result = FootStepInterval / (GetCharacterVelocity() + 0.1f);
    return result;
}

protected virtual bool IsCharacterMoving ()
{
    //Debug.Log("Current speed: " + GetCharacterVelocity());
    return GetCharacterVelocity() > footStepSpeedTreshold;
}

protected virtual float GetCharacterVelocity ()
{
    if (!characterController) return 0;
    return characterController.velocity.magnitude;
}
}

```

Esimerkkikoodi 2. FootstepComponent luokka. Se tuottaa askelääniä silloin, kun hahmo liikkuu.

Kuten esimerkistä voi nähdä, yksi hyvin tärkeä osa operaatiota on Timer-luokka (esimerkkikoodi 3), joka tarkistaa, kuinka paljon aikaa on kulunut. Luokan nimeämistavan ero johtuu siitä, että kirjoitin luokan alun perin toista projektiani varten ja toin sen tänne.

```

public class CS_Timer
{
    protected float initiationTime;
    protected float duration;

    // Constructors
    public CS_Timer (float _duration)
    {
        initiationTime = Time.time;
        duration = _duration;
    }

    public CS_Timer (float _duration, float _initiationTime)
    {
        initiationTime = _initiationTime;
        duration = _duration;
    }

    // Tarkistaa onko ajastimen aika kulunut
    public virtual bool IsTimerComplete()
    {
        if ((Time.time - initiationTime) > duration)
        {
            ResetTimer();
            return true;
        }
        return false;
    }

    public virtual void ResetTimer ()
    {
        initiationTime = Time.time;
    }
}

```

```

    }

    // [poistettua koodia, lyhentämisen takia]
}

```

Esimerkkikoodi 3. Aikakello-luokka, jota käytän muutamassa paikassa.

### 5.3 Varasto keruutoiminnolle

Tarvitsin jonkinlaisen tavaroidenkeräysjärjestelmän. Tavarahan pitäisi olla maassa kerättävissä, jolloin se voitaisiin lisätä jonkinlaiseen inventoryyn, jota pelaaja voi käsitellä. Loin kantaluokan nimeltä `PCC_ItemGroundComponent` (esimerkkikoodi 4), josta periytyvät yksityiskohtaiset esinetyypit kuten avaimet ja valkoinen keppi.

```

public class PCC_ItemGroundComponent : MonoBehaviour
{
    public AudioClip BounceAudioClip;

    [Serializable]
    public class TriggerEvent : UnityEvent { }
    public TriggerEvent onPickedUp;

    protected PC_Item thisItem;
    protected AudioSource thisAudioSource;

    protected virtual void Start ()
    {
        //thisItem = null;
        thisAudioSource = GetComponent<AudioSource>();
        if (!thisAudioSource) Debug.LogError("This object is missing an audio source component.");
    }

    public virtual PC_Item GetThisItem ()
    {
        return thisItem;
    }

    public virtual void DestroyItemAfterPickup ()
    {
        onPickedUp?.Invoke();
        Destroy(this.gameObject);
    }

    public virtual void BounceSound ()
    {
        if (!thisAudioSource) return;

        thisAudioSource.PlayOneShot(BounceAudioClip);
    }
}

```

Esimerkkikoodi 4. `ItemGroundComponent` on kantaluokka kaikille maassa oleville tavaroille.

Tein näistä erityyppisistä objekteista myös "prefab"it. Prefabit ovat tallennettuja peliobjekteja, joita voi esimerkiksi helposti vetää listalta pelin näkymään tai luoda pelin aikana käyttäen Unityn Instantiate-funktiota. Tämä tekee mahdolliseksi käsitellä kopioita näistä objekteista tosi helposti.

Tässä välissä saatoinkin tehdä alkeellisen version valkoisesta kepeistä (white cane/probing cane). Tein siitä yksinkertaisen (esimerkkikoodi 5). Jos on seinä tietyn etäisyyden sisäpuolella, se tuottaa "töks"-äänen. Jos se on sen etäisyyden ulkopuolella, se tuottaa "woosh"-äänen.

```
public override void UseThisItem()
{
    base.UseThisItem();
    if (!Owner) return;
    if (!OwnerAudioSource) return;

    // Tarkistaa, jos tagilla sallitut objektit ovat edessä ja lähellä.
    Ja vähän extraa.
    if (GetObjectInRangeAndHandleSomeOtherStuffThatYouReallyReallyReallyShouldntBeDoingInThisFunction(ForwardWallDetectionDistance, Owner.transform))
    {
        OwnerAudioSource.PlayOneShot(GetUseAudioEffect());
        Debug.Log("The cane hits a thing.");
    } else
    {
        OwnerAudioSource.PlayOneShot(GetFailureAudioEffect());
        Debug.Log("The cane hits nothing.");
    }
}

public virtual bool GetObjectInRangeAndHandleSomeOtherStuffThatYouReallyReallyReallyShouldntBeDoingInThisFunction(float distance, Transform ownerTransform)
{
    if (Physics.Raycast(ownerTransform.position, ownerTransform.TransformDirection(Vector3.forward), out rayHitForward, distance))
    {
        Debug.DrawRay(ownerTransform.position, ownerTransform.TransformDirection(Vector3.forward) * rayHitForward.distance, Color.yellow);
        //Debug.Log("Hit: " + rayHitForward.transform.gameObject + "\nDistance: " + rayHitForward.distance);
        if (rayHitForward.transform.gameObject.tag == "Wall")
        {
            return true;
        }
        if (rayHitForward.transform.gameObject.tag == "Interactable")
        {
            PCC_Breakable _breakable = rayHitForward.transform.gameObject.GetComponentInParent<PCC_Breakable>();
            if (_breakable)
            {
                _breakable.DamageThisBreakable(1);
                return true;
            }
        }
    }
}
```

```
    }  
    }  
    return false;  
}
```

Esimerkkikoodi 5. Valkoisen kepin käyttö -funktio ja relevantti boolean.

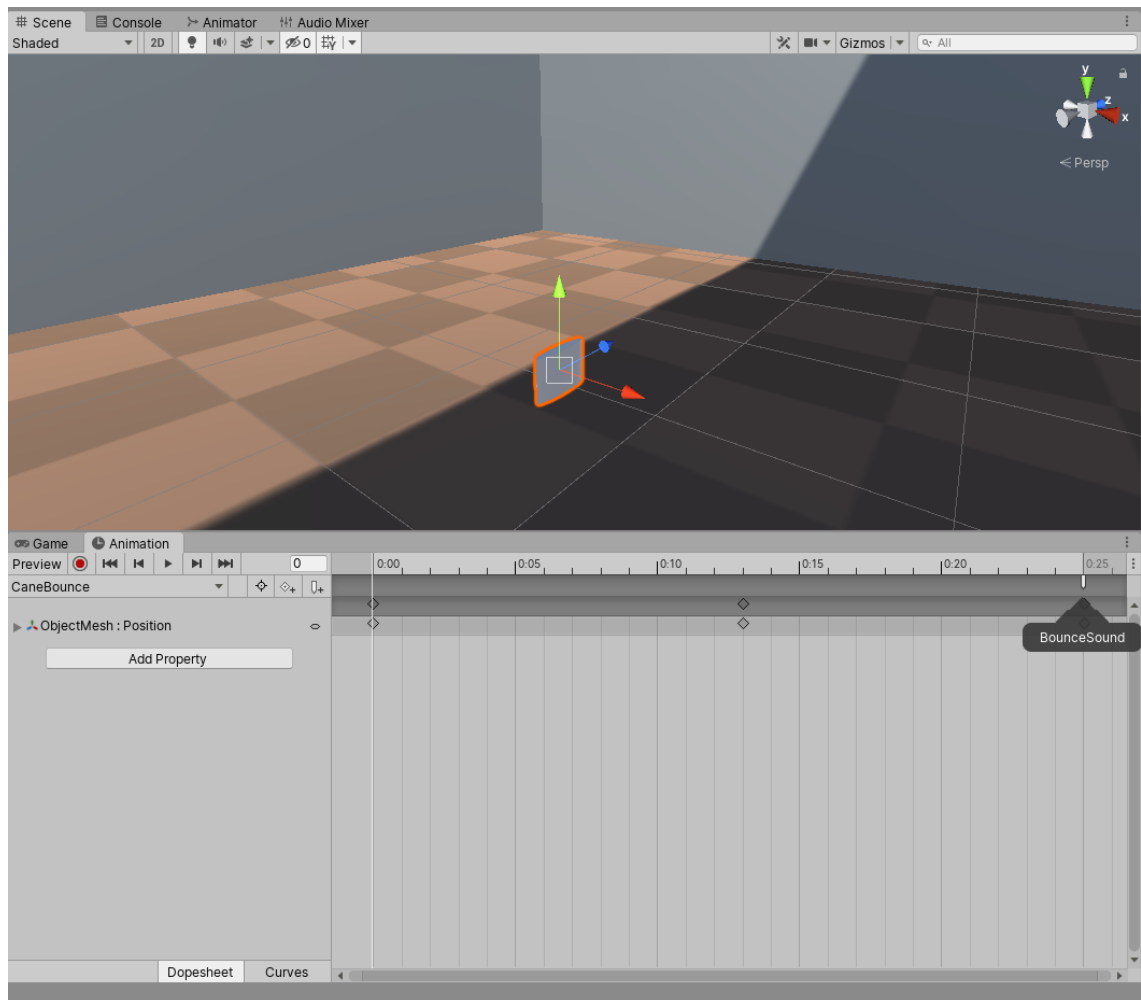
Ideaalisessa visiossa keppi tuottaisi eri ääniä erilaisista pinnoista, mutta voi olla, että sellaista ei ole tässä työssä aikaa toteuttaa.

Sitten loin PCC\_InventoryComponentin. Vaikka kyseistä komponenttia tässä vaiheessa on suunniteltu käyttämään vain pelaaja, päätin luoda sen omana erillisenä asiana, jotta sitä voisi mahdollisesti uudelleenkäyttää muissakin asioissa, jos niin halutaan.

Kun hahmo, jolla on InventoryComponent-luokka liitettynä, kävelee tämän objektin yli, siitä kutsutaan GetThisItem-funktiota, joka lisää non-monobehaviour-version kyseisestä tavarasta Inventory-listaan, minkä jälkeen kutsutaan DestroyItem-funktiota, joka tuhoaa sen pelimaailmassa olevan fyysisen representaation.

Non-monobehaviour-versiossa tavarasta määritetään sen eri käyttöominaisuuksia. Esimerkiksi PC\_Key sisältää salasanan, jota ovet tarkistavat silloin, kun pelaaja kävelee ovia kohti, ja PC\_Canessa on funktio nimeltä "UseThisItem", joka tuottaa siihen määritetyn toiminnon.

Vaikka peli on tarkoitettu täysin sokeille, päätin hyödyntää BounceSound():n kutsumista varten animaatiokomponenttia, joka kutsuu kyseistä funktiota aina tietyssä kohtaa animaatiotaan (kuva 8).



Kuva 8. Animaatioikkuna. Oikeassa reunassa näkyy, milloin funktiota kutsutaan.

Pomppauksen visuaali on tarpeeton, mutta se tekee toteutuksesta helpon. Ainoa koodipohjainen toteutus, jonka keksin, olisi käyttää taas aikakello-luokkaani, mutta tämä on huomattavasti yksinkertaisempi.

Ovea ja lukkoa varten loin eri komponenttiluokat siltä varalta, että haluan lisätä lukon mahdollisesti jonkinlaiseen muuhun objektiin, joka ei ole ovi (esimerkkikoodi 6).

```
public class PCC_Lock_Base : MonoBehaviour
{
    // Lista avaimenrei'istä, jos halutaan että lukossa on enemmän kuin yksi.
    public List<KeyHole> KeyHoles = new List<KeyHole>();
    protected bool DoorUnlocked = false;
    public delegate void UnlockAttempted();
    public UnlockAttempted OnUnlockAttempt;
```

```

[System.Serializable]
public class KeyHole
{
    public bool KeyHoleUnlocked;
    public string Password;
}
// Yrittää avata oven annetuilla avaimilla.
public virtual void AttemptUnlock (List<PC_Key> _keys)
{
    if (KeyHoles.Count <= 0)
    {
        SetDoorUnlock(true);
    }

    TryKeys(_keys);
    if (AreAllKeyholesUnlocked())
    {
        SetDoorUnlock(true);
    }

    OnUnlockAttempt?.Invoke();
}
// Ottaa sisään listan avaimia ja tarkistaa onko niiden joukossa sopivia
protected virtual void TryKeys (List<PC_Key> _keys)
{
    for (int i = 0; i < KeyHoles.Count; i++)
    {
        for (int j = 0; j < _keys.Count; j++)
        {
            if (KeyHoles[i].Password == _keys[j].GetPassword())
            {
                KeyHoles[i].KeyHoleUnlocked = true;
            }
        }
    }
}
// Tarkistaa onko kaikki avaimenreiät avattuja.
public virtual bool AreAllKeyholesUnlocked () { /* Lyhennetty */ }
// Booleaani erikseen, jos halutaankin avata ovi vaikka avaimenreikiä
saattaakin olla yhä auki.
public virtual bool IsUnlocked ()
{ return DoorUnlocked || AreAllKeyholesUnlocked(); }
}

```

Esimerkkikoodi 6. Lukkokomponentti. Se avautuu, kun sille syötetään oikeat avaimet.

Ovi ottaa sisään listan avaimia, käy ne kaikki läpi ja avaa kaikki avaimenreiät, joihin löytyi sopiva avain. Kun kaikki avaimenreiät ovat auki, ovi nousee. Esimerkkikoodissa 7 on maassa olevan avaimen luokka.

```

public class PCC_KeyGround : PCC_ItemGroundComponent
{
    public string Password;

    protected override void Start()
    {
        base.Start();
        thisItem = new PC_Key(Password);
    }
}

```



```

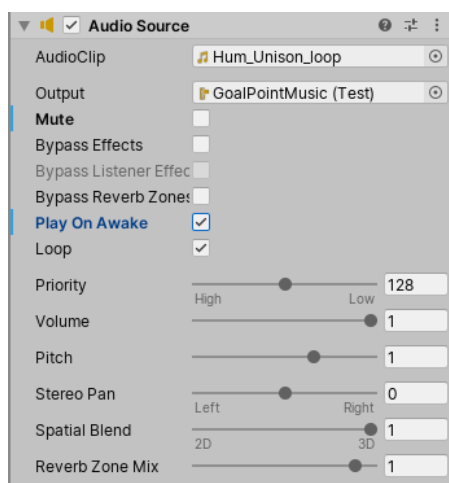
public virtual string GetPassword ()
{
    return Password;
}
}

```

Esimerkkikoodi 7. Maassa olevan avaimen luokka.

#### 5.4 Tason esineitä

Maalipiste on iso valkoinen pallo, josta pelaaja voi kävellä läpi. Lisäsin siihen Unityn oman AudioSource-komponentin (kuva 9), joka tekee sen omistavasta objektista äänenlähteen. Niitäkin pitää useimmiten käsitellä koodissa itse, jos haluaa äänien vastaavan pelin aikana, mutta aikaisessa testausvaiheessa riitti, että valitsin komponentin AudioClip-muuttujaksi jonkin placeholder-äänien, laitoin päälle "Play On Awake" ja "Loop", Spatial Blend = 1, sitten testasin sitä pelissä yrittämällä kävellä sen ääntä kohti.



Kuva 9. AudioSource-komponentti.

Laitoin siihen placeholder-ääneksi musiikkia, ja sitten kokeilin navigoida sen kohdalle äänen perusteella. Löysin perille.

Myöhemmin tein loppupistettä varten PCC\_SceneTransitionTrigger-komponentin, joka lataa uuden näkymän, kun pelaaja kävelee maalipisteen rajojen sisäpuolelle.

Ladattavan näkymän nimi määritetään editorissa, jotta ei tarvitse luoda omaa komponenttia joka tason siirrokselle.

Replikkien pelissä implementoimista varten päätin tehdä nopean ratkaisun, koska siinä vaiheessa oli kiire saada peli esiteltävään muotoon. Tein prefabin nimeltä "Object\_Speaker", johon on kiinnitetty luomani "Announcement Component", joka pystyy jonottamaan audioklippejä ja soittamaan ne (esimerkkikoodi 8).

```
public class PCC_AnnouncementComponent : MonoBehaviour
{
    protected List<AudioClip> announcementQueue = new List<AudioClip>();

    protected virtual void Update()
    {
        HandleAnnouncementQueue();

        if (queuedLastEntry && !audioSource.isPlaying)
        {
            onAnnouncementQueueEnded?.Invoke();
            queuedLastEntry = false;
        }
    }

    protected virtual void HandleAnnouncementQueue()
    {
        if (announcementQueue.Count <= 0 || audioSource.isPlaying) return;
        audioSource.PlayOneShot(announcementQueue[0]);
        RemoveAnnouncementAtIndex(0);
    }

    public virtual bool IsAnnouncementActive ()
    {
        return announcementQueue.Count > 0 || audioSource.isPlaying;
    }

    public virtual bool QueueAnnouncement(AudioClip announcement)
    {
        if (!audioSource) return false;
        AddAnnouncement(announcement);
        return true;
    }

    public virtual bool QueueAnnouncements(List<AudioClip> announcements)
    {
        if (!audioSource) return false;
        for (int i = 0; i < announcements.Count; i++)
        {
            AddAnnouncement(announcements[i]);
        }
        return true;
    }
}
```

Esimerkkikoodi 8. AnnouncementComponentin lyhennetty versio.

Näitä komponentteja hallinnoi yleensä joko LevelController, joka hoitaa kaikenlaisia operaatioita tason alussa, tai AnnouncementTriggerListener, jonka tarkoitus on

kuunnella tapahtumia ja soittaa kuulutuksensa, kun ne tapahtuvat. Esimerkiksi, kun pelaaja ottaa tavaran maasta, se kutsuu delegaattia, jota AnnouncementTriggerListener on kuuntelemassa.

Object\_Speakerit tuottavat 3D-ääntä, ja ne on lajiteltu ympäri tasoja. Ne tuottavat myös hieman venttiilimäistä ääntä koko ajan antaakseen pelaajille lisää pieniä vertailupisteitä navigoidessa.

## 5.5 Päävalikko

Päävalikkoa varten loin oman näkymän, joka ei vielä luo pelaajaa. Yleensä valikot ovat hyvin visuaalinen konsepti, ja tämäkin on lähtökohtaisesti rakennettu visuaalisella tavalla. Siinä on kolme vaihtoehtoa listattu rivissä, ylhäältä alas: "New Game", "Audio Test" ja "Exit Game". Kun pelaaja käyttää horisontaalisita syötettä, näiden valintojen sivulla oleva valitsin liikkuu ylös ja alas. Kun valinta vaihtuu, kuuluu repliikki, joka kertoo, mikä vaihtoehto on valittuna. Esimerkkikoodissa 9 nähdään valiko syötteen tarkistusta.

```
bool bReceivedAxisInput = false;
void LateUpdate()
{
    if (Input.GetButtonDown("Cancel"))
    {
        ToggleMenu();
    }

    // Do not check for these inputs, if the menu is not open
    if (!bMenuOpen) return;

    if (Input.GetButtonDown("Submit"))
    {
        SelectorComponent.SelectCurrentIndex();
    }

    if (GetVerticalInput() == 0 && bReceivedAxisInput) bReceivedAxisInput
= false;

    if (GetVerticalInput() != 0 && !bReceivedAxisInput)
    {
        bReceivedAxisInput = true;
        if (GetVerticalInput() > 0) SelectorComponent.NextOption();
        if (GetVerticalInput() < 0) SelectorComponent.PreviousOption();
    }
}
```

Esimerkkikoodi 9. Valikon syötteen tarkistusta.

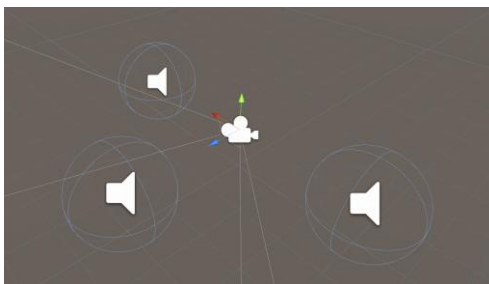
MainMenu-objektia käytetään myös MainMenu- näkymän ulkopuolella, osana pelaajan prefabia, jotta pelaaja voi pelin aikana halutessaan tuoda sen esiin (kuva 10).



Kuva 10. Päävalikko, kun se on näkyvillä.

”New Game” aloittaa uuden pelin. ”Audio Test” tekee äänitestin, joka pyörittää kolme eri repliikkiä kolmesta eri suunnasta. Repliikit sanovat: ”tämän äänen pitäisi tulla vasemmalta”, ”tämän äänen pitäisi tulla oikealta” ja ”tämän äänen pitäisi tulla edestä”. ”Exit Game” poistuu ohjelmasta.

MainMenu-scenen kameran eteen, oikealle puolelle ja vasemmalle puolelle on asetettu objekteja, jotka myös sisältävät Announcement-komponentin (kuva 11). Valikkovalintoja varten käytetään enimmäkseen edessä olevaa kuuluttajaa, mutta äänitestiä varten käytetään myös vasenta ja oikeaa.



Kuva 11. Päävalikon kamera ja sen kaiuttimet.

Äänitestin idea on tarkistaa äänien suunnan toimivuus, koska se on hyvin tärkeää tietoa pelaajalle. Jos äänet tulevat väärästä suunnasta tai jos pelaajat eivät pysty kertomaan, mistä suunnasta mikäkin ääni tulee, voi olla, että pelaajat joutuvat selvittämään ongelmaa.

## 5.6 Äänen suunnittelu ja tuottaminen

Ensimmäinen ääni, jonka tuotin oli askeleet. Yritin saada ne kuulostamaan mahdollisimman geneerisiltä kengän askelilta. Käytin puhelintani nauhoittamaan itseäni tallomassa kengät jalassa kotonani. Yritin hieman poistaa taustaääniä sulkemalla tietokoneen ja ikkunan ja vetämällä johdon irti jääkaapista, mutta minulla ei ollut saatavilla oikeaa nauhoituskoppia, hyviä varusteita tai kokemusta äänen tuotosta, mikä voi vähän kuulua laadussa.

Tarvitsin myös jonkinlaiset jatkuvat äänet tavaroille, jotta pelaaja pystyisi löytämään ja ottamaan ne. Jokaisella tavaratyyppillä kannattaisi olla oma ääni, jotta ne pystyy erottamaan toisistaan.

Pelin sisäinen selitys tälle jatkuvalla äänelle on, että tässä testilaitoksessa, missä pelaaja on, tapahtui jonkinlainen sci-fi-tyyppinen onnettomuus ja nyt kaikki esineet pomppivat loputtomiin lattioilla.

Keppiä varten käytin kokkailuun tarkoitettua puulastaa. Kokeilin tökkiä sillä kaikkea, mitä omistan, kun yritin etsiä sopivaa ääntä, ja tulin siihen tulokseen, että seinä oli paras. Avainta varten aluksi harkitsin tavallisen avaimen tökkimistä seinään, mutta se ei tuottanut erityisen selvää ääntä, ja päätin sen sijaan tökkiä vesilasiasia voiveitsellä. Ei ehkä paras mahdollinen päätös, mutta se toimii toistaiseksi.

Tarvitsin myös äänet, kun pelaaja ottaa kyseisen tavaran maasta. Tätä varten ajattelin, että raapaiseva ääni olisi kaikkein luonnollisin, joten käytin aikaisemmin mainittua puulastaa keppiä varten ja kolikkoa avainta varten ja raapaisin niillä seinää.

Ovea varten tarvitsin virhe-äänien silloin, kun ovi on lukittu, ja aukaisuäänien silloin, kun pelaaja avaa sen. Virhe-ääntä varten nauhoitin kaksi nuottia pianostani ja lisäsin efektejä Audacityllä, jotta se kuulostaisi hieman sci-fimaiselta.

Kun ovi aukeaa, se tuottaa aika kivimäisen äänen. Tämä ei ole ehkä ideaalinen ääni, mutta koska tuotan kaiken itse, piti löytää jotain, mikä olisi helposti käsillä. Otin pari hyvänkokoista kiveä, hieroin niitä toisiinsa ja lisäsin suodattimia editoinnissa, jotta se kuulostaisi syvemmältä.

Maalipisteiden humina oli vähän erilainen tapaus. Alun perin laitoin placeholder-ääneksi musiikkikappaleen "Collector Base", joka on alun perin pelistä Mass Effect 2. Omat musiikintuotostaitoni ovat hyvin rajoitetut, mutta olen kyllä harjoitellut pianon soittamista.

Yritin silti tuottaa hyvin syvä-äänistä huminaa imitoidakseni alkuperäistä placeholder-musiikkia. Ajattelin, että syvämpi ääni olisi teoriassa hyvä jatkuvalla, kokoaikaisella äänelle, joka ei ikinä lopu, koska uskon, että se häiritsee aisteja huomattavasti vähemmän kuin keski- tai korkeaääniset nuotit, mutta on silti huomattavissa.

Object\_Speaker tuottaa venttiilimäistä ääntä. Sitä varten nauhoitin tavallista venttiiliaukkoa, mutta se oli raakana hieman epätaisainen ääni, joten poistin siitä vahvempia piikkejä editoinnissa.

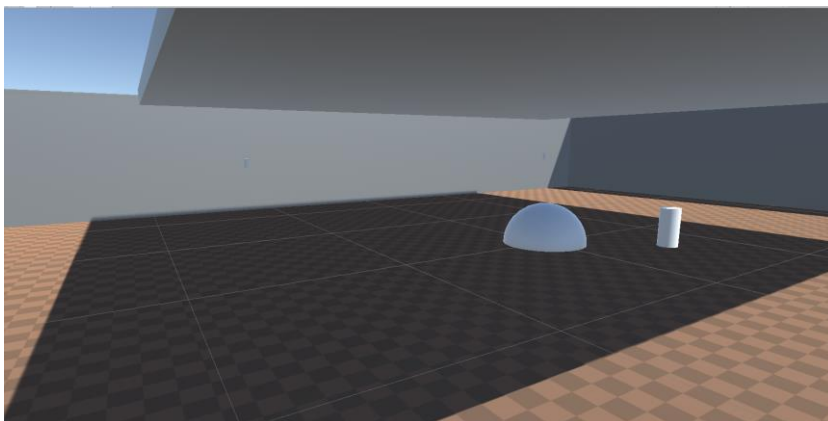
Päävalikkoa ja joitakin perusmekaniikkoja varten oli pakko äänittää sanallisia ohjeita. Alun perin suunnittelin äänittäväni ne itse, mutta tyttöystäväni, Sanna Nousiainen, onnistui löytämään ääninäyttelijäksi pyrkivän henkilön, joka suostui tekemään palveluksen ja äänittämään muutaman repliikin ilmaiseksi. Hän ei kertonut oikeaa nimeään, mutta kutsuu itseään "Mämmi"ksi. Teknisesti hän ei nauhoittanut kovin montaa repliikkiä, mutta ne on jaettu 35 osaan, joita käytetään pelissä.

Hän äänitti neljä kontrollien ohjeistusrepliikkiä: päävalikon navigoinnille, liikkumiselle ja kepin käytölle, sitten kuusi "flavor"-repliikkiä, jotka antavat peliin tunnelmaa ja 11 repliikkiä päävalikon valinnoille ja 8 tasojen numeron kuuluttamiselle.

Ideana oli imitoida Portal-pelien GLaDOSia, joka on psykopaattinen tekoäly, joka hallinnoi kaikkia Aperture-tiedelaitoksen toimintoja, ja tämänkin pelin tekoälyhahmo on hieman kylmä pelaajaa kohtaan ja huomattavasti enemmän kiinnostunut testamaan pelaajan kykyä ratkaista ongelmia, mitä yritin kuvastaa esimerkiksi repliikillä: "Sokeuden simuloimiseksi olemme ottaneet vapauden poistaa silmäsi. Pyydämme anteeksi mahdollisia hankaluuksia mitä tämä saattaa tuottaa. Voitte välittää valituksen Valitusten Silppuri Osastolla kun olette suorittaneet kurssin."

## 5.7 Tasot

Insinöörityönä tehty peli toimii tasojen logiikalla, jossa seuraava taso on haastavampi kuin edellinen. Mutta päätin ottaa taas oppia Portal-peleistä ja kohdella jokaista tasoa eräänlaisena tutoriaalina ja pulmana samaan aikaan. Kuvassa 12 näkyy taso 1.

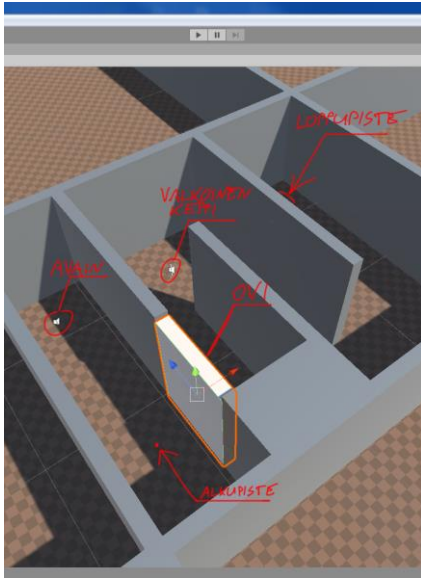


Kuva 12. Taso 1.

Ensimmäisessä tasossa kerrotaan, miten liikkua. Tämän jälkeen pelaajan täytyy itse keksiä päämäärä. Vihje on annettu sillä, että maalipiste tuottaa kokoaikaista huminaa, joka tulee tietystä suunnasta. Muita esteitä ei ole pelaajan ja maalin välissä.

Toisella tasolla pelaajan ja maalin välissä on seinä. Kun pelaaja kulkee suoraan kohti maalia, hän osuu seinään ja hänen pitäisi huomata askelten hitaus ja rapina sitä kohti kävellessä. Täällä pelaajan täytyy navigoida kulkemalla seinää pitkin ja päästä lopulta maaliin.

Kolmannella tasolla (kuva 13) tavataan ovi ja avain. Koska ovi on aloituspisteen ja loppupisteen välissä, pelaajan pitäisi teoriassa osua siihen ensimmäisenä.



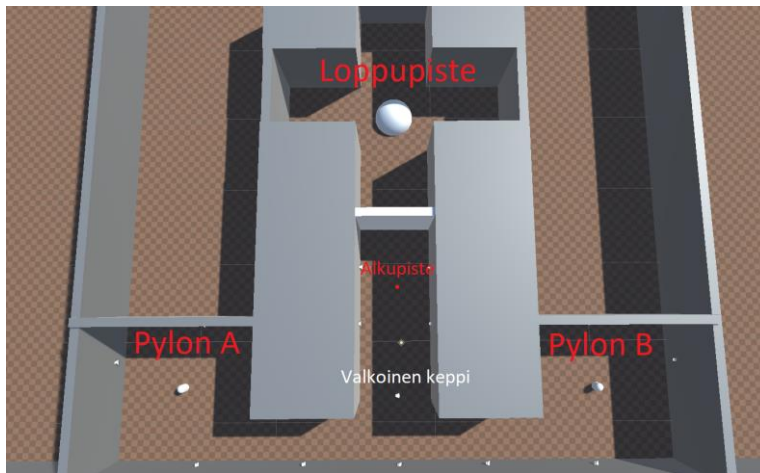
Kuva 13. Taso 3.

Kun pelaaja osuu oveen, kuuluu virhe-ääni, jonka idea on antaa pelaajalle vihje, että tässä on kyseessä ovi, joka on lukittu.

Neljäs taso oli koottu hieman hätiköidysti. Halusin vielä yhden tason, ennen kuin lähetin latauslinkin testaajille, joten rakensin parissa tunnissa tämän.

Neljännellä tasolla (kuva 14) on kaksi suuntaa, joihin pelaajan täytyy kulkea. Molemmissa suunnissa on Object\_Pylon, joita pelaajan täytyy tökkäistä kepillä, jotta ovi loppupisteeseen aukeaa.





Kuva 14. Taso 4.

Loppupisteen ääntä ei kuule, ennen kuin molemmat pylonit on suljettu. Pelaajan täytyy navigoida käyttäen lähettyvillä olevia venttiilejä. Ennen kuin pelaaja saapuu risteykseen, hänen pitäisi kuulla valkoisen kepin ääni ja kävellä sitä kohti.

Kun pelaaja lähestyy pylonia, siitä kuuluu hyvin erikoinen humina. Mitään vihjeitä ei ole annettu pelaajalle, mitä ne ovat tai mitä niillä on tarkoitus tehdä, joten on vaikeaa sanoa, onnistuuko pelaaja keksimään, että objektia täytyy tökkäistä kepillä, mutta se on ainoa objekti, jota pelaajalle annetaan, joten pelaaja teoriassa keksii ratkaisun.

Huoli saattaa ehkä olla koko venttiili-idean kanssa. Luulisin, että usean pienemmän äänelähteen seuraaminen ei toimi ihan yhtä hyvin kuin yhden päämäärän, mutta sitä tässä testataankin.

## 6 Testaajien palaute

Siinä vaiheessa, kun lähetin pelin esittelyversion vapaaehtoisille testaajille, joihin sain yhteyttä Näkövammaisten Liiton kautta, minulla oli noin neljä tasoa valmiina, joista viimeinen oli ehkä vähän liian nopeasti koottu yhteen. Näkövammaisten Liiton kautta testaajiksi ilmoittautui yksitoista henkilöä, joista neljä vastasi ja joista kolme antoi rakentavaa palautetta ja luvan käyttää palautetta tässä raportissa. Puolet palautteesta saapui vasta maaliskuun loppupuolella, joten en ehtinyt kehittämään uutta esittelyversiota palautteen pohjalta, mutta kommentit, jotka sain, olivat seuraavat:

#### Testaajapalaute 1:

Ensimmäinen testaaja oli hyvä ystäväni, Jose (en laskenut häntä yhdeksi ilmoittautujista). Hän ei ole sokea, mutta ajattelin, että oli hyvä antaa jonkun testata peli ainakin yhden kerran läpi, ennen kuin lähdin jakamaan linkkiä vapaaehtoisille. Hän tuntui sisäistävän pelin idean aika nopeasti. Hän huomasi, että seiniä vasten kävellessä syntyy ääni ja askelten nopeus riippui siitä, millaisessa kulmassa hahmo käveli seinää päin, mikä antoi hänen navigoida huoneita aika hyvin.

Hän saattoi mahdollisesti olla hieman hämmentynyt, kun hän osui lukittuun oveen ja kuuli tuottamani virhe-äänen, ja kommentoi: *"vissiin kävelen pianon päällä nyt?"* Kuitenkin hän siirtyi ovesta muualle ja kuuli avaimen kilinän. Sitten hän jatkoi tason loppuun ilman mitään erityisempiä vaikeuksia. Tämän jälkeen kehitin vielä hätiköidysti kootun neljännen tason, jota en ehtinyt esitellä Joselle.

Joka tapauksessa hän tuntui aika nopeasti ymmärtävän, miten navigoida pelissä.

#### Testaajapalaute 2:

Juho sanoi, että peli oli vaikeudeltaan noin 3/10, mutta turhautui ja lopetti neljännessä tasossa, jolloin tuli liian monta ääntä, joiden tarkoitusta hän ei ymmärtänyt. Hän kommentoi, että pelissä on hyvin vähän ohjeita, eikä aloittaessa tiennyt, mitä piti tehdä, mutta nähtävästi hän tästä huolimatta onnistui keksimään, että on tarkoitus kulkea ääntä kohti.

Hän myös teki ehdotuksia, kuten esimerkiksi, että kannattaa selittää tason päämäärä sen alussa, ehdotti pisteytymekaniikkaa ja dynaamisempia elementtejä, kuten liukuportaita, liukuovia, korkeuseroja ja pudotuksia. Valkoisesta kepeistä hän sanoi, että sen käyttö ei ihan samaistunut arkielämän käyttöön.

#### Testaajapalaute 3:

Samilla oli nähtävästi vaikeuksia äänenvoimakkuuden kanssa, josta olen vähän yllättynyt, mutta se olisi ehkä korjattavissa, jos olisin ehtinyt lisäämään peliin äänenvoimakkuus vaihtoehtoja.

Kuitenkin hän kommentoi että peli on ”yksinkertaisesti liian hankala” ja neuvoi monipuolistamaan pelin äänimaailmaa antamalla enemmän vihjeitä, jos pelaaja on ollut tietyn aikaa hukassa, tai nostettavien esineiden lähetyvillä pyytämään pelaajaa ottamaan objekti.

Hän myös ehdotti navigoinnin lukitsemista 90 asteen kulmiin tai lisäämään jonkinlaisen tavan tiedottamaan pelaajaa ilmansuunnista, koska hän koki vaikeuksia löytää oikeaa orientaatiota.

Testaajapalaute 4:

Kosse myös raportoi äänenvoimakkuuden vähäisyydestä. Selvästi tämä on ongelma, jota kannattaisi mahdollisessa jatkokehityksessä katsoa. Hän sanoi, että vaikka äänet saattavat olla laadultaan hyväksyttäviä, esineet, askeleet, seinästä aiheutunut ääni ja keppi olivat liian hiljaisia loppupisteeseen verrattuna, joka saattaisi joskus jopa peittää muita ääniä. Hän myös suositteli antamaan käyttäjän säätää eri äänien voimakkuutta, joka on tosiaan hyvä idea.

Hän kommentoi, että peliä oli kohtalaisen vaikea navigoida ilman koordinaatteja, jolloin ainoa keino tutkia alueita oli harhailta päättömästi tai seurata seiniä, mikä vaikutti hänestä epäkäytännöllisiltä vaihtoehdoilta. Tämä oli vähän omatuntoa pistävä kommentti, koska seinät tosiaan olivat suunniteltu navigointitapa. Mutta hän ei ainakaan kutsunut sitä mahdottoman vaikeaksi ja arvioi vaikeusasteeksi noin 6/10.

## 7 Yhteenveto

Kuten kehitysprojekteissa yleensäkin, tehtyjen valintojen ja ohjelmakoodin uudelleentarkastelu kannattaa tehdä esittelyversiovaiheen päätteeksi. Myös peli-idean monipuolistaminen kohti julkaistavaa versiota etenee esimerkiksi uusien dynaamisten elementtien lisäämisellä, kuten murskaimet, kuilut ja viholliset.

Jotkut testaajista sanoivat, että sokean on liian vaikeaa vaigoida pelissä ja että peli ehkä tarvitsisi enemmän ohjeistusta. Tämä voi olla totta, mutta on vaikeaa kuvitella, kuinka paljon enempää ohjeistusta voin antaa paljastamatta tason ratkaisua. Peli jo

kertoo, miten ohjata hahmoa ja miten käyttää esineitä, joten ainoat asiat, mitkä jäävät pelaajalle mietittäväksi, ovat tasojen ratkaisut ja niihin liittyvät elementit. Ajastetut vihjeet saattaisivat auttaa asiaa jonkin verran.

Luulen, että suurin osa ongelmista olisi ratkaistavissa, jos äänien suunnittelu olisi parempi, mikä tekisi joistakin elementeistä intuitiivisempia, mutta rajoitetulla ajalla ja varusteilla sellaisten suunnittelu ja tuottaminen oli vaikeaa.

Voi myös olla, että koska sokeiden pelaajien määrä on hyvin pieni, heidän yleinen pelilukutaitonsa saattaa olla huomattavasti pienempi, mikä selittäisi, miksi persoonallinen kaverini onnistui navigoimaan ilman ongelmia.

Myös testaaajien raportoima äänenkorkeusongelma kannattaisi korjata säätövaihtoehdoilla. Vaikka mielestäni äänien tasapaino vaikutti hyvältä, se olisi hyvä antaa käyttäjälle säädettäväksi. Ainoa syy, miksi sellaista ei ollut jo esittelyversiossa, oli se, että aika jäi rajoitetuksi. Äänenlähteet on kyllä jaettu eri ääniryhmille, mutta koodillinen toteutus ei sinne ehtinyt.

Hyvänlaatuisen äänen tuottaminen oli vaikeaa myös tarvikkeiden ja kokemuksen puutteen vuoksi, vaikka näinkin tämän projektin tilaisuutena oppia. Helmikuun loppupuolella 2020 aloitin työt Housemarquella, ja jouduin tekemään 8 tuntia töitä joka arkipäivä tutkintoni lisäksi, mikä saattoi vaikuttaa laatuun. Suurin osa ilmoittautuneista testaaajista ei antanut palautetta, mikä vaikeuttaa pelin ongelmien tunnistamista ja vaikuttaa raportin sisällön määrään.

Parhaani yritin, mutta kuten usein tunnen omista peleistäni, mielestäni jäi parantamisen varaakin.

Insinööriyössä tutkittin, onko mahdollista kehittää sokeille peliä, jossa täytyy navigoida 3D-ympäristöä. Vaikka pelit eivät ainakaan nykyään vielä pysty tuottamaan taktiillia tunnetta, luulen, että pelkästään ääniä käyttämällä tämä päämäärä olisi saavutettavissa. Tätä on vaikea arvioida, koska vain muutama testaaaja antoi minkäänlaista palautetta johon perustaa projektini onnistuneisuutta taikka epäonnistuneisuutta, mutta sanoisin, että vaikka pelille on tässä pohja, olisi äänen suunnittelussa parantamisen varaa.

Olen oppinut tämän projektin aikana pelien suunnittelua, koodaamista ja äänen tuottamista. Aihetta oli kiinnostavaa tutkia, ja se on antanut uutta perspektiiviä siihen, mitä peleillä voi vielä tehdä ja missä alueissa tutkintojen määrä on ollut peliteollisuudessa yleisesti rajoitettua.

Audiopelit ovat ainakin nykyään hyvin pieni osa koko peliteollisuuden rinnalla. Toivottavasti löydökseni auttavat aiheesta kiinnostuneita. Pelistä ei tullut niin valmista kuin halusin, mutta se on käyttökelpoinen pohja, josta voi jatkaa kehitystä, ja se voi toimia esimerkkinä muille peleille.

## Lähteet

1. Shades of Doom. 2005. Verkkoaineisto. Gmagames. <<http://www.gmagames.com/sod.html>>. Luettu 3.1.2020.
2. Shades of Doom. 2005. Verkkoaineisto. Audiogames. <<https://audiogames.net/db.php?action=view&id=shadesofdoom>>. Luettu 3.1.2020.
3. Q9 Action Game. 2009. Verkkoaineisto. Audiogames. <<https://audiogames.net/db.php?action=view&id=Q9%20action%20game>>. Luettu 3.1.2020.
4. Terraformers. 2003. Verkkoaineisto. Audiogames. <<https://audiogames.net/db.php?action=view&id=terraformers>>. Luettu 5.1.2020.
5. Brennan, Claire. 2014. 'Video-less' 3D games developed for blind players. Verkkoaineisto. BBC. <<https://www.bbc.com/news/technology-28757186>>. 19.8.2014. Luettu 14.1.2020.
6. A Blind Legend. 2016. Verkkoaineisto. Steampowered. <[https://store.steampowered.com/app/437530/A\\_Blind\\_Legend/](https://store.steampowered.com/app/437530/A_Blind_Legend/)>. Luettu 14.1.2020.
7. Targett, Sue & Fernström, Mikael. 2003. Audio games: Fun for all? All for fun! Verkkoaineisto. Smartech. <<https://smartech.gatech.edu/handle/1853/50444>>. Päivitetty 9.7.2003. Luettu 20.1.2020.
8. Röber, Niklas & Masuch, Maic. 2005. Playing Audio-only Games: A compendium of interacting with virtual, auditory Worlds. Verkkoaineisto. Simon Fraser University. <<http://summit.sfu.ca/item/243>>. 16.3.2005. Luettu 24.1.2020.
9. Edison, Tommy. 2011. Verkkoaineisto. YouTube. <<https://www.youtube.com/user/TommyEdisonXP/about>>. Kanava luotu 8.10.2011. Luettu 30.11.2016.
10. Edison, Tommy. 2013. Can Blind People Draw? Verkkoaineisto. YouTube. <<https://www.youtube.com/watch?v=P1IY6plQKGI>>. 2.7.2013. Luettu 30.11.2016.
11. Edison, Tommy. 2017. Does A Blind Person Visualize Things In Their Mind? Verkkoaineisto. YouTube. <<https://www.youtube.com/watch?v=gCwZCaawKNk>>. 8.3.2017. Luettu 8.3.2017.

12. Dr. Joe Hanson. Verkkoaineisto. Comscicon. <<https://comscicon.com/Joe-Hanson>>. Luettu 20.2.2020.
13. Bushway, Brian. Verkkoaineisto. Brianbushway. <<https://www.brianbushway.com/about/>>. Luettu 20.2.2020.
14. Burke, Molly. 2014. Verkkoaineisto. YouTube. <<https://www.youtube.com/user/MollyBurkeOfficial/about>>. Kanava luotu 11.7.2014. Luettu 20.2.2020.
15. Hanson, Joe. 2019. How Blind People See With Sound. Verkkoaineisto. YouTube. <<https://www.youtube.com/watch?v=08smCjKWNL0>>. 10.8.2019. Luettu 20.2.2020.
16. Kolarik, Andrew; Cirstea, Silvia; Pardhan, Shahina & Moore, Brian. 2014. A Summary of research investigating echolocation abilities of blind and sighted humans. Verkkoaineisto. Zenodo <<https://zenodo.org/record/853235#.XoN9e4gzaUI>>. 1.4.2014. Luettu 22.2.2020.
17. Trakooma. 2020. Verkkoaineisto. World Health Organization. <<https://www.who.int/news-room/fact-sheets/detail/trachoma>>. 1.2.2020. Luettu 12.2.2020.
18. Matti Seppänen. 2018. Glaukooma. Verkkoaineisto. Duodecim terveyskirjasto. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=dlk00452](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=dlk00452)>. 27.2.2018. Luettu 12.2.2020.
19. Nystagmus. Verkkoaineisto. American Optometric Association. <<https://www.aoa.org/patients-and-public/eye-and-vision-problems/glossary-of-eye-and-vision-conditions/nystagmus>>. Luettu 12.2.2020.
20. Retinitis pigmentosa. 2019. Verkkoaineisto. National Eye Institute. <<https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/retinitis-pigmentosa>>. 10.7.2019. Luettu 12.2.2020.
21. Matti Seppänen. 2018. Diabeettinen retinopatia. Verkkoaineisto. Duodecim terveyskirjasto. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=dlk00826](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=dlk00826)>. 22.2.2018. Luettu 12.2.2020.