



Anna Raevskaia

Kielikaverit: Godot-pelimoottoria käyttävä kaksikielisten lasten mobiili- peli

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknikan tutkinto-ohjelma

Insinöörityö

4.10.2024

Tiivistelmä

Tekijä:	Anna Raevskaia
Otsikko:	Kielikaverit: Godot-pelimoottoria käyttävä kaksikielisten lasten mobiilipeli
Sivumäärä:	33 sivua
Aika:	4.10.2024
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikka
Ammatillinen pääaine:	Ohjelmistotuotanto
Ohjaaja:	Lehtori Vesa Ollikainen

Tämän opinnäytetyön tavoitteena on suunnitella ja kehittää mobiilipeli, joka tarjoaa innostavan ympäristön kaksikielisten lasten kielten oppimiseen ennen koulua.

Suomalais-venäläinen päiväkotikielikeskus "Kalinka" tarjoaa apua pelin kehittämiseen pedagogisesta näkökulmasta ja sen testaamisesta. Keskeisenä tehtävänä on luoda lapsille selkeä käyttöliittymä, minipelit ja ohjeet, joiden avulla lapset voivat opiskella kieliä leikkimielisesti. Tulosten avulla pyritään arvioimaan pelin tehokkuutta kielten oppimisen tukena sekä kehittämään peliä entistä pedagogisemmin ja käyttäjäystävällisemmin.

Työssä esitellyt ratkaisut Godot-pelimoottorin käytöstä tuovat esiin teknologisen osaamisen merkityksen pelisuunnittelussa, mikä tarjoaa tietoa muille opiskelijoille ja pelialan ammattilaisille. Tulokset tuovat esille, miten mobiilipelin suunnittelussa huomioitiin pedagogiset näkökulmat ja miten peli voi toimia tehokkaana työkaluna kielten oppimisen tukena lapsille. Pelin testaus "Kalinka"-päiväkodissa tuottaa tietoa siitä, miten peli vastaanotetaan kohdeyleisössään ja mikä on arvokasta sekä pelisuunnittelijoille että lasten parissa työskenteleville ammattilaisille.

Avainsanat: Godot, mobiilipeli, opetuspelejä, lapset, kaksikielisyys

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Anna Raevskaia
Title: Kielikaverit: Mobile Game for Bilingual Children Using Godot Game Engine
Number of Pages: 33 pages
Date: 4 October 2024

Degree: Bachelor of Engineering
Degree Programme: Information and Communication Technology
Professional Major: Software Engineering
Supervisor: Vesa Ollikainen, Senior Lecturer

The goal of this study was to design and develop a mobile game that provides an engaging environment for bilingual children to learn languages before starting school.

The Finnish-Russian kindergarten "Kalinka" offered assistance in developing and testing the game from a pedagogical perspective. The main task was to create an intuitive and simple user interface, mini-games, and instructions for children, allowing them to learn languages playfully. The aim was to assess the effectiveness of the game in supporting language learning and further develop it to be more pedagogically sound and user-friendly.

The solutions presented regarding the use of the Godot game engine highlight the importance of technological expertise in game design, providing valuable insights to other students and professionals in the gaming industry. The results demonstrate how pedagogical perspectives were considered in mobile game design and how the game can serve as an effective tool for supporting language learning in children. Testing the game in the "Kalinka" kindergarten offered insights into how the game is received by its target audience, which is valuable for game designers and professionals working with children.

Keywords: Godot, mobile game, educational game, children, bilingualism

Sisällys

Lyhenteet

1	Johdanto	1
2	Monikielisyys ja koulutus	1
2.1	Kaksikielisen lapsen kasvattamisen menetelmät	2
2.2	Eurooppalaisen kielitaidon viitekehys (CEFR)	3
2.3	Pelillisuus opetuksessa	4
2.4	Olemassa olevia kielenoppimissovelluksia	5
3	Godot-pelimoottori	8
3.1	Godot-pelimoottorin ominaisuudet	9
3.1.1	Skriptauskielet	9
3.1.2	Solmut, näkymät ja resurssit	10
3.1.3	Signaalit	11
3.1.4	Ryhmät	12
4	Kielikaverit-peli	14
4.1	Pelin tavoitteet	14
4.2	Minipelit	14
4.3	Käytetyt teknologiat	16
5	Kielikaverit-pelin toteutus	16
5.1	Käyttöliittymä	17
5.2	Pukeutumispelin toteutus	18
5.3	Tietovisapelin toteutus	22
5.4	Pelin rakentaminen	26
6	Valmiin pelin arviointi	27
6.1	Pelin testaaminen, lopputulokset ja palautteet	27
6.2	Pelin tulevaisuus	28
7	Yhteenveto	28
	Lähteet	30

Lyhenteet

2P2L: *Two Persons Two Languages*. Kaksi Vanhempaa - Kaksi Kieltä -menetelmä.

CEFR: *The Common European Framework of Reference for Languages*. Eurooppalaisen kielitaidon viitekehys.

mL@H: *Minority Language at Home*. Vähemmistökieli kotona -menetelmä.

OPOL: *One Person One Language*. Yksi henkilö - yksi kieli -menetelmä.

T&P: *Time and place*. Aika ja paikka -menetelmä.

1 Johdanto

Tämän opinnäytetyön tavoitteena on suunnitella ja kehittää mobiilipeli, joka tarjoaa innostavan ympäristön kaksikielisten lasten kielten oppimiseen ennen koulua. Keskeisenä tehtävänä on luoda lapsille selkeä käyttöliittymä, minipelit ja ohjeet, joiden avulla lapset voivat opiskella kieliä leikkimielisesti.

Raportin toisessa luvussa käydään läpi kaksikielistä varhaiskasvatusta. Siinä tutkitaan olemassa olevia pelejä ja sovelluksia kieltenoppimiseen.

Insinööriyössä perehdytään Godot-pelimoottoriin ja sen omaan GDScript-ohjelmointikieleen. Godot on ilmainen ja avoimen lähdekoodin pelimoottori. Se on suunniteltu luomaan sekä kaksi- että kolmiulotteisia pelejä tietokoneelle, verkkoselaimille sekä iOS- ja Android-alustoille.

Päiväkoti Kalinka on kielipäiväkoti, joka on perustettu vuonna 1990. Kalinkassa on äidinkieleltään sekä suomen- että venäjänkielisiä lapsia. Kalinka tarjoaa apua pelin kehittämiseen pedagogisesta näkökulmasta ja sen testaamisesta. Tulosten avulla pyritään arvioimaan pelin tehokkuutta kielten oppimisen tukena sekä kehittämään peliä entistä pedagogisemmin ja käyttäjäystävällisemmin.

2 Monikielisyys ja koulutus

Monikielisyys tarkoittaa sitä, että yksilö osaa useita kieliä sujuvasti. Maailmassa 43 % ihmisistä on kaksikielisiä, 13 % kolmikielisiä ja 3 % puhuu yli kolmea kieltä (1). Tämä tarkoittaa, että maailmassa monikielisyys on yleisempää kuin yksikielisyys.

Tutkimusten mukaan ihmiset, jotka osaavat useampaa kuin yhtä kieltä, pärjäävät kielten opiskelussa paremmin kuin ne, jotka osaavat vain yhden kielen. Lisäksi kaksikieliset lapset eivät kehity hitaammin kuin yksikieliset lapset. (2; 3.)

Lapset oppivat yksinkertaiset sanat ensin. Yleensä nämä sanat ovat lyhyitä ja sisältävät äänneitä, joita on helppo lausua. Kaksikieliset lapset saattavat joskus löytää sanan tai rakenteen, joka on helpompi tai ilmeisempi toisessa kielessä, ja käyttävät sitä helpompaa sanaa tai rakennetta molemmissa kielissä. Esimerkiksi jotkut lapset saattavat vaihtaa venäjänkielisen *еуё* (*yeshcho*) -sanat suomenkieliseen *lisää*-sanaan. Ajan myötä lapsi oppii selviytymään kielivirheistä ja alkaa käyttää oikeaa sanaa jokaisessa kielessä. Toinen tekijä kielen kehityksessä on sanojen toistaminen. Lapsi oppii ensin ne sanat, joita käytetään enemmän. (2.)

Monikielisyyttä voidaan tukea varhaiskasvatuksessa opettamalla lapsille, että kaikki kielet ovat sallittuja ja he saavat käyttää omia kieliään. Lapset voivat käyttää omia kieliään esimerkiksi leikeissä. Lapsilta voi myös kysyä nimityksiä esineille heidän omilla kielillään. (4.)

2.1 Kaksikielisen lapsen kasvattamisen menetelmät

Perhekielen menetelmät määrittävät, miten lapsi opiskelee kieltä: mitä kieltä lapsi puhuu kotona ja julkisesti, kuka puhuu lapselle milläkin kielellä ja onko eri kielille varattu tietty aika ja paikka. Tohtori Ute Limacher-Riebold kuvaa yleisimmin käytettyjä perhekielen menetelmiä.

Yksi henkilö - yksi kieli -menetelmä tunnetaan myös nimellä OPOL (englanniksi *One Person One Language*). Se tarkoittaa metodia, jossa jokainen vanhempi tai hoitaja puhuu eri kieltä lapsen kanssa. (5.) Kielentutkija Annick De Houwerin tutkimuksessa todettiin, että neljännes lapsista, joiden perheet käyttivät OPOL-menetelmää, ei tullut kaksikielisiksi. Nämä lapset puhuvat yleensä vain enemmistökieltä, eli koulutuksen kieltä. Tämä kieli saa heille suuremman merkityksen, jolloin vähemmistökieli ei vaikuta yhtä tärkeältä. (6.) Tätä menetelmää voidaan hyödyntää kaksikielisten lasten kielenopiskelusuovelluksessa siten, että pelissä on kaksi hahmoa, jotka puhuvat eri kieliä. Tällä tavalla jokainen hahmo edustaa yhtä kieltä, mikä auttaa lapsia erottamaan kielet toisistaan selkeämmin.

Vähemmistökieli kotona -menetelmän mukaan molemmat vanhemmat puhuvat kotona samaa vähemmistökieltä. Vähemmistökieli tarkoittaa kieltä, joka ei ole jonkin maan pääkieli. (3.) Tätä menetelmää tunnetaan myös nimellä mL@H (englanniksi *minority Language at Home*). (5.)

Aika ja Paikka -menetelmä, tunnetaan myös nimellä T&P (englanniksi *Time and Place*), keskittyy sovittuun aikatauluun, jossa vanhemmat ja lapset päättävät puhua eri kieliä riippuen joko ajasta, paikasta tai molemmista (5). Tämän strategian suurin etu on, että perheet voivat räätälöidä sen omiin tarpeisiinsa. Huono puoli tässä menetelmässä on se, että jos lapsi kuulee ja käyttää vähemmistökieltä vain tietyissä ajoissa ja paikoissa, hän saattaa olla kehittämättä täyttä sanavarastoa kyseisessä kielessä. (7.) Opettajat voivat hyödyntää tätä menetelmää suunnittelemalla oppitunnit tiettyjen päivien tai kellonaikojen mukaan, jolloin keskitytään yhteen kieleen kerrallaan.

Kaksi Vanhempaa - Kaksi Kieltä -menetelmää yleensä noudattavat vanhemmat, jotka puhuvat molempia kieliä ja päättävät puhua kumpaakin niistä lapsille. Esimerkiksi yksi vanhemmista puhuu suomea ja ruotsia ja toinen puhuu venäjää ja englantia. Monet vanhemmat valitsevat aluksi OPOL-strategian ja lisäävät muita kieliä myöhemmin. Tämä menetelmä tunnetaan myös nimellä 2P2L (englanniksi *Two Persons Two Languages*). (5.)

2.2 Eurooppalaisen kielitaidon viitekehys (CEFR)

Eurooppalaisen kielitaidon viitekehys (CEFR) on kansainvälinen standardi, jolla kuvataan kielitaitoa. Se jakaa kielitaidon kuuteen tasoon alkaen A1-tasosta C2-tasoon. (8.) Tässä projektissa keskitytään erityisesti A1-tasoon ja sitä edeltävään tasoon.

A1-tasoa edeltävä taso (englanniksi *Pre A1*) tarkoittaa, että opiskelija ei vielä osaa muodostaa lauseita. Tällä tasolla opiskelija osaa käyttää yksinkertaisia tervehdyksiä ja numeroita sekä tehdä valintoja, kuten valita tuotteen, koon tai värin verkkokaupassa visuaalisen tuen avulla. (9.)

A1-taso tarkoittaa kielitaidon alkeita. Tällä tasolla opiskelija osaa kommunikoida yksinkertaisissa tilanteissa ja ymmärtää perusasioita. Opiskelija osaa esimerkiksi käyttää numeroita, kertoa, mikä päivä ja sää tänään on, kysyä ja kertoa kellonajan, kuvailla ihmisiä, esineitä ja paikkoja, kertoa rutiineistaan, ilmaista mieltymyksiään ja kuvata yksinkertaisia toimintoja. (10; 11.)

A1-tason sanastoon kuuluvat esimerkiksi viikonpäivät, kuukaudet ja vuodenajat, numerot ja hinnat, maat ja kansallisuudet, värit, kehon osat, perhe ja ystävät, ruoka ja juomat, kodin huoneet ja huonekalut sekä vaatteet. (10; 11.)

Kieliopin osalta A1-tason opiskelija osaa esimerkiksi käyttää kysymyssanoja, negatiivisia lauseita, persoonapronomineja, imperatiivia eli käskymuotoa, paikallissijoja ja monikon muotoa. Suomen kielen kielioppi sisältää myös verbityyppjä, partitiivin ja genetiivin. Venäjän kielessä on lisäksi substantiivien suku. (10; 11.)

2.3 Pelillisuus opetuksessa

Pelillistäminen tarkoittaa pelimekaniikkojen käyttöä tilanteissa, jotka eivät ole pelejä. Näitä pelimekaniikkoja ovat esimerkiksi pisteet, palkinnot, tulostaulut ja edistymispalkit. (12; 13.)

Pelillisyyden käytöstä lasten opetuksessa voi olla monia etuja. Pelillisuus tekee oppimisesta hauskeempaa ja mielenkiintoisempaa sekä motivoi lapsia parantamaan tuloksiaan. Leikin avulla lapset voivat muistaa opetusmateriaalia paremmin. Pelillisyyden haitat tulee kuitenkin ottaa huomioon. Pelimekaniikat voivat viedä lasten huomion pois itse oppimismateriaalista, jos ne ovat liian häiritseviä tai houkuttelevat liikaa kilpailemaan pisteistä ja palkinnoista. (12; 13.)

2.4 Olemassa olevia kielenoppimisovelluksia

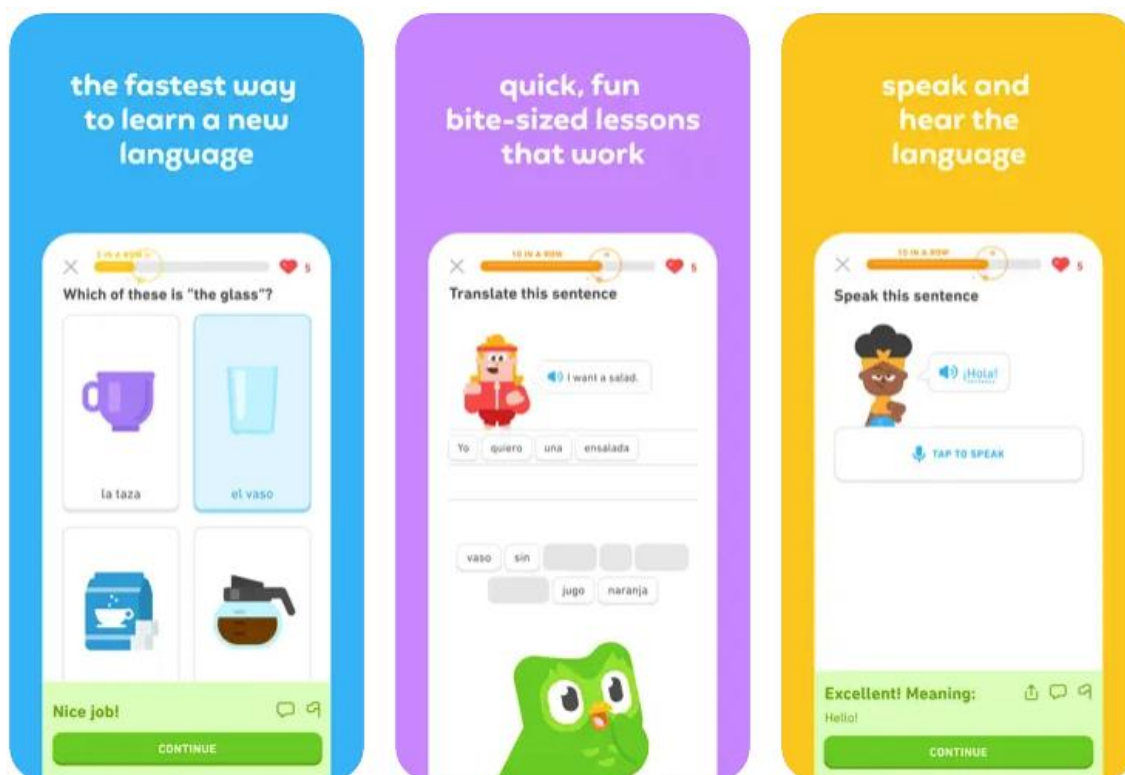
Insinööriyössä tutkittiin myös jo olemassa olevia opetuspelejä. Vaikka löytyy useita kieltenopetuspelejä, ei ole paljon pelejä, jotka on suunniteltu juuri venäjän ja suomen kielen opettamiseen. Useimmat olemassa olevista peleistä keskittyvät yhteen kieleen eivätkä tue kaksikielisyyttä.

Yksi suosituimmista ilmaisista kieltenopiskelusovelluksista on Duolingo (14). Duolingo on Carnegie Mellonin yliopiston professori Luis von Ahnin ja hänen opiskelijansa Severin Hackerin vuonna 2011 perustama mobiilisovellus ja verkkopalvelu. Sovellus tarjoaa kursseja useilla eri kielillä. Sen avulla käyttäjät voivat oppia perussanastoa ja kielioppia. (15.) Duolingo voi pyytää opiskelijaa kääntämään lauseita kohdekielestä äidinkieleensä tai päinvastoin, valitsemaan oikean vaihtoehdon useista annetuista vaihtoehdoista, kuuntelemaan sanan ja kirjoittamaan sen tai täyttämään lauseeseen puuttuvat sanat. Sovellukseen lisättiin uusi ominaisuus, joka mahdollistaa käyttäjien oppia myös puhumista keskustelemalla tekoälyn kanssa. Ominaisuus simuloi keskusteluja eri hahmojen kanssa. Tekoäly mukautuu käyttäjän vastaukseen ja kielitaitoon. (16.) Duolingon avulla käyttäjät voivat seurata edistymistään ja asettaa omia tavoitteita, jotka pitävät heidät motivoituneina jatkamaan kielten opiskelua. Duolingo käyttää viittä periaatetta sovelluspohjaiseen opettamiseen:

1. Käyttäjät aloittavat kielen opiskelun heti käytännön harjoituksilla ilman pitkiä luentoja tai selityksiä. (17.)
2. Duolingo säätää harjoitusten vaikeustason jokaiselle käyttäjälle sopivaksi tekoälyn avulla. (17.)
3. Oppitunnit perustuvat kansainvälisiin standardeihin, kuten eurooppalaiseen kielitaidon viitekehykseen (CEFR), joka varmistaa, että opiskelijat oppivat tärkeät asiat oikeassa järjestyksessä. (8; 17.)
4. Duolingo ylläpitää käyttäjien motivaatiota päivittäisten tavoitteiden, palkintojen ja tulostaulujen avulla. (17.)

5. Duolingo pyrkii tekemään oppimisesta hauskaa käyttämällä tarinankerrontaa, huumoria ja hauskoja animaatioita. (17.)

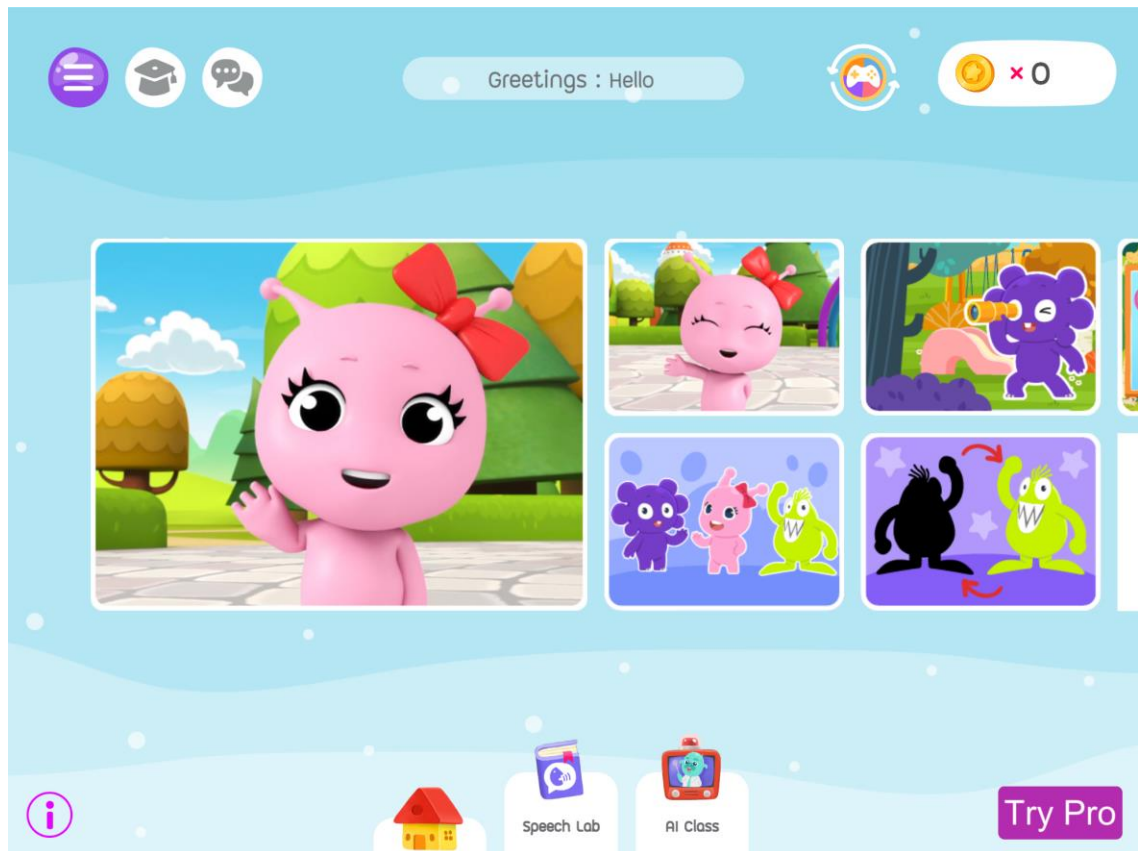
Duolington (kuva 1) helppokäyttöisyys ja pelillistetyt oppitunnit sekä värikäs grafiikka voivat kiinnittää lasten huomion. Jotkut kriitikot kuitenkin väittävät, että liiallinen pelillistäminen voi häiritä kielen oppimista ja heikentää muistamista pitkällä aikavälillä. (12.) Sovellus tukee pääasiassa lukutaidon kehittämistä, joten se ei sovi lapsille, jotka eivät vielä osaa lukea.



Kuva 1. App Store -kaupan mainoskuvat Duolingo-sovelluksesta.

Seuraava tutkittu sovellus on Galaxy Kids (kuva 2), joka on suunniteltu 3–8-vuotiaille lapsille englannin ja kiinan opiskeluun. Kuten Duolingo, se tarjoaa hauskoja aktiviteetteja, eikä perinteisiä oppitunteja. Sovellus keskittyy pääasiassa puhumiseen ja kuunteluun, ei lukemiseen ja kirjoittamiseen, mikä sopii lapsille, jotka eivät vielä osaa lukea. Oppitunnit sisältävät videon, joka esittelee opittavan sanaston, minipelin ja puhelaboration, joka tarkistaa ääntämistä puheentunnistusta käyttämällä. Sovelluksen ilmaisen version oppimismahdollisuudet

ovat kuitenkin rajalliset. Opetus perustuu eurooppalaiseen viitekehykseen (CEFR). Sovellus tarjoa opetusta A1-tasolla ja sitä edeltävällä tasolla. (18; 19.)



Kuva 2. Kuvakaappaus Galaxy Kids -sovelluksesta.

Kolmas tutkittu sovellus on Gus On The Go -sovellus (kuva 3). Se on maksullinen mobiilisovellus, joka on suunniteltu tukemaan 2–5-vuotiaiden lasten monikielisyttä. Sovellus on saatavilla lähes 30 eri kielellä. Sovellus tarjoaa minipelejä, joissa annetaan ohjeita vain opetettavalla kielellä ilman äidinkielen käyttöä. Ohjeissa käytetään myös paljon visuaalisia elementtejä. Minipelit koostuvat taasoista, ja kun pelaaja suorittaa tason, hän saa palkinnoksi pokaalin ja pääsee seuraavalle tasolle. Sovellus opettaa lapsille aiheita, kuten numerot, värit, eläimet, vaatteet ja kulkuvälineet. (20.)



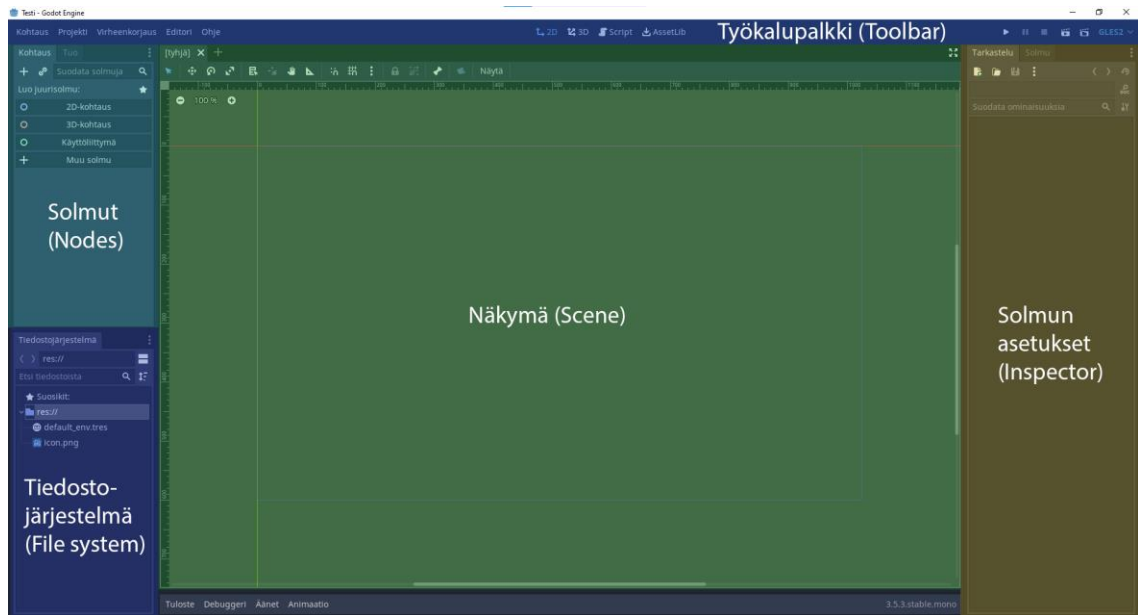
Kuva 3. App Store -kaupan mainoskuvat Gus On The Go -sovelluksesta.

Suunnittelemassamme projektissa päätettiin käyttää Gus On The Go -sovelluksen minipelejen ja palkintojen idea. Tämä toteutus huomioi sen, että päiväkodissa ei ole mahdollisuutta säilyttää henkilökohtaisia palkintoja ja edistymistä, koska kaikki lapset käyttävät yhteistä tablettia. Tällä menetelmällä lapset voivat valita itselleen sopivan vaikeustason. Palkintojen osalta voidaan käyttää positiivista palautetta ja hauskoja animaatioita. Kehittämässämme pelissä tarkoituksena on tukea pääasiassa kuuntelutaitojen kehittämistä, joten pelissä käytetään mahdollisimman vähän tekstiä, kuten Galaxy Kids -sovelluksessa.

3 Godot-pelimoottori

Godot on avoimen lähdekoodin 2D- ja 3D-pelimoottori, joka tarjoaa monipuoliset työkalut pelien kehittämiseen eri alustoille, kuten PC:lle, mobiililaitteille ja pelikonsoleille. (21; 22.)

Godot-pelimoottori (kuva 4) on innovatiivinen ja monipuolinen työkalu pelikehityksessä, joka tarjoaa kehittäjille helppokäyttöisen käyttöliittymän, monipuoliset ominaisuudet ja joustavan lisensointimallin. (21; 22.)



Kuva 4. Editorin näkymä.

Godot-pelimoottorissa on kaksi tasoa: korkean tason pelimoottori ja matalan tason API. Pelinkehittäjä työskentelee korkean tason pelimoottorissa, joka koostuu solmuista ja näkymistä. Matalan tason API sisältää käyttöjärjestelmän (OS) ja palvelimet. (23.)

3.1 Godot-pelimoottorin ominaisuudet

Tässä luvussa käydään läpi Godot-pelimoottorin ominaisuudet, joita hyödynnetään pelin kehityksessä. Näihin ominaisuuksiin kuuluvat käytetyt skriptauskielet, solmut, näkymät, resurssit, signaalit ja ryhmät.

3.1.1 Skriptauskielet

Godot tarjoaa neljä virallista ohjelmointikieltä: GDScript, C#, C ja C++. On mahdollista käyttää useita kieliä yhdessä projektissa. (21; 24.)

GDScript on Godot-pelimoottorin sisäänrakennettu oliopohjainen skriptauskieli. Sen syntaksi on selkeä ja yksinkertainen, mikä tekee siitä sopivan sekä

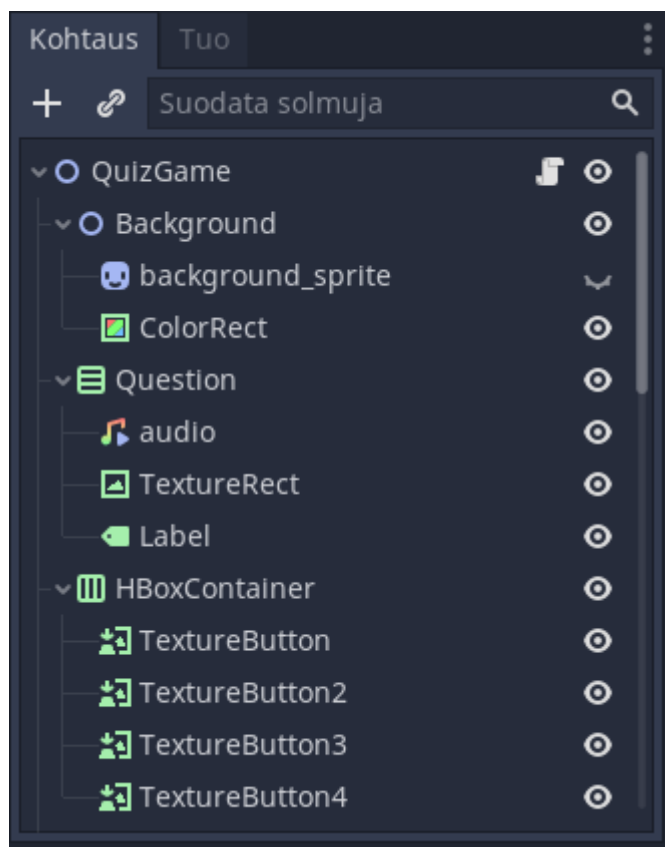
aloittelijoille että kokeneille kehittäjille. GDScript on saanut inspiraationsa Squirrel-, Lua- ja Python-kielistä. (24.)

3.1.2 Solmut, näkymät ja resurssit

Solmut ovat Godot-pelimoottorin perusyksiköitä, jotka edustavat erilaisia pelimaailman elementtejä, kuten hahmoja, objekteja, valoja ja äänilähteitä. Jokainen solmu voi sisältää toisia solmuja. (25.) Godotin solmuja voidaan verrata yleisesti käytettävän Unity-pelimoottorin GameObject-objekteihin.

Näkymät ovat solmujen kokoelmia. Niillä on aina yksi juurisolmu. Ne voidaan tallentaa paikalliselle asemalle ja ladata myöhemmin, sekä luoda niin monta instanssia näkymästä kuin halutaan. (25.) Godotin näkymiä voidaan verrata Unity-pelimoottorin Prefabeihin.

Kuva 5 esittää Godot-editorin näkymä tietovisa minipelistä ja sen solmujen koelmasta, jossa QuizGame-solmu on juurisolmu.

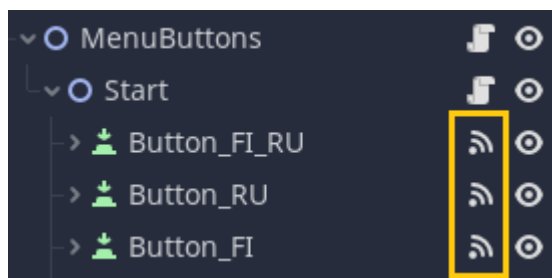


Kuva 5. Godot-editorin näkymä ja sen sisältämät solmut.

Resurssit ovat tietosäiliöitä. Solmut käyttävät resursseissa olevia tietoja. Nämä tiedot voivat olla muun muassa tekstuurit, äänet, 3D-mallit ja kooditiedostot. Resurssi on mahdollista ladata levyltä vain yhden kerran. Jos resurssin kopio on jo muistissa, uudelleenlataus palauttaa aina saman kopion. (26.)

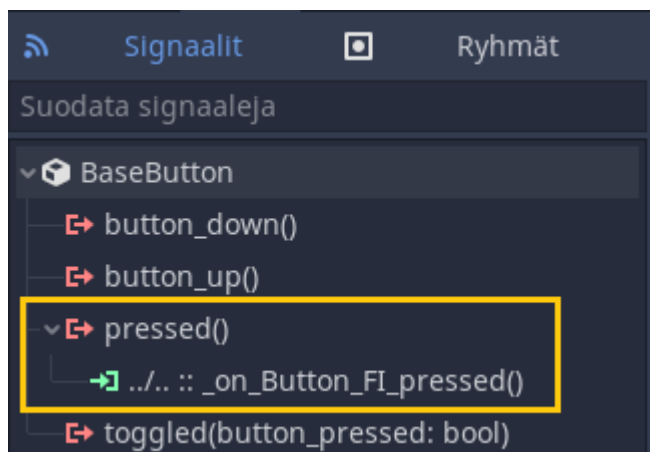
3.1.3 Signaalit

Signaalit toimivat solmujen välisenä viestinvälityksenä aktivoituen tietyistä tapahtumista, kuten painikkeen painalluksesta. Kuva 6 esittää esimerkki signaaleista, jotka on asetettu painikkeisiin.



Kuva 6. Painikkeisiin asetetut signaalit.

Esimerkiksi, kun käyttäjä painaa painiketta, joka on kytketty `_on_button_pressed` -signaaliin (kuva 7), Godot lähettää tämän signaalin ja suorittaa siihen liittyvän metodin (esimerkkikoodi 1). Tämä voi käynnistää tiettyjä toimintoja pelissä, kuten pelitasojen vaihtamisen, pelin tallentamisen tai minkä tahansa muun halutun toiminnon. (27.)



Kuva 7. Pressed-metodi on asetettu solmuun.

```
func _on_Button_FI_pressed():
    Global.change_language("fi")
    get_tree().change_scene("res://Scenes/Home.tscn")
```

Esimerkkikoodi 1. Metodi, joka vaihtaa pelin kielen suomeksi ja siirtyy kotinäky-
mään painikkeen painalluksesta.

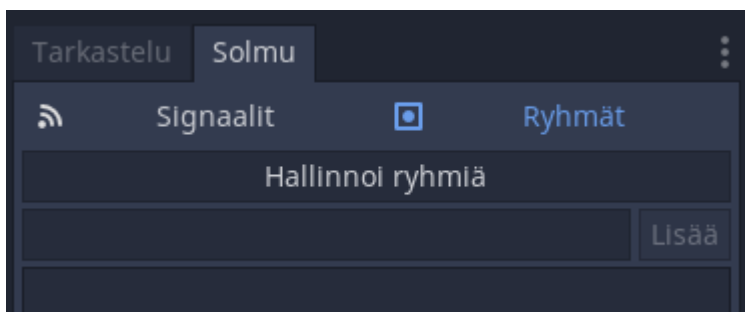
Godot-pelimoottorissa on monia erilaisia signaaleja eri tapahtumien käsittelyyn. Tässä on muutamia esimerkkejä yleisimmistä signaaleista Godot-pelimoottorissa:

- Signaali `_on_button_pressed` lähetetään, kun käyttäjä painaa painiketta.
- Signaali `_on_button_released` lähetetään, kun käyttäjä vapauttaa painikkeen.
- Signaali `_ready` lähetetään, kun solmu on alustettu ja valmis käytettäväksi.
- Signaali `_input` lähetetään, kun käyttäjä antaa syötettä.
- Signaali `_process` lähetetään jokaisessa pelimoottorin päivityksessä.

3.1.4 Ryhmät

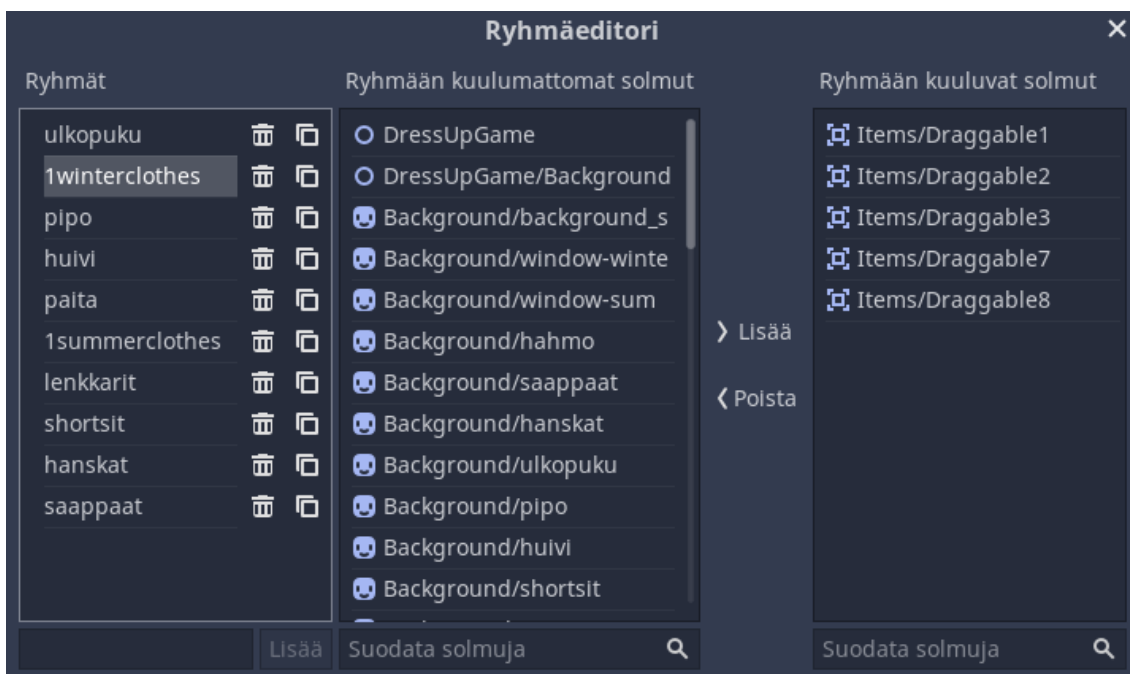
Ryhmät ovat hyödyllinen ominaisuus solmujen järjestämiseen. Ryhmät Godotissa toimivat kuten tunnisteet (englanniksi *tags*) muissa ohjelmistoissa. Ryhmistä

voidaan lisätä tai poistaa solmuja käyttämällä Solmu-ikkunaa editorissa tai `Node.add_to_group()`- ja `Node.remove_from_group()` -komentoja (kuva 8). (28.)



Kuva 8. Solmu-ikkuna editorissa.

Ryhmäeditorin voi avata painamalla "Hallinnoi ryhmiä" -painiketta (kuva 9). Ryhmäeditori sisältää kolme saraketta. Ensimmäinen sarake sisältää luettelon kaikista ryhmistä, jotka ovat nykyisessä näkymässä. Kun ryhmä valitaan tästä luettelosta, toinen sarake näyttää luettelon ryhmään kuulumattomista solmuista ja kolmas sarake näyttää luettelon ryhmään kuuluvista solmuista. (28.)



Kuva 9. Ryhmäeditori-ikkuna.

4 Kielikaverit-peli

4.1 Pelin tavoitteet

Projektin tavoitteena oli luoda perusta pelille, joka on helposti laajennettavissa ja kehitettävissä tulevaisuudessa. Projektia varten päätettiin käyttää minipelejä, koska ne ovat lyhyitä ja keskittymistä vaativia. Lyhyiden pelisessioiden avulla lapset voivat oppia uusia sanoja menettämättä kiinnostustaan. Minipelit voivat sisältää erilaisia tehtäviä ja haasteita, tukea eri oppimistyyliä. Pelin tulisi tarjota lapsille selkeä käyttöliittymä, ohjeita, tavoitteita ja palkintoja, joiden avulla he voivat opiskella kieliä leikkimielisesti.

Kalinkan tavoitteena oli laajentaa lasten sanavarastoa sekä suomeksi että venäjäksi käyttämällä digitaalista työkalua sekä hyödyntää erilaisia oppimismenetelmiä ja -strategioita, jotka tukevat lasten erilaisia oppimistyyliä. Pelin suunnittelussa oli tärkeää ottaa huomioon, mistä 4–6-vuotiaat lapset pitävät. Päiväkoti-ryhmässä järjestetyn kyselyn perusteella selvisi, että suosituimpia aiheita ovat robotit, eläimet ja palomiehet. Näiden aiheiden pohjalta voidaan suunnitella lapsille mielenkiintoisia hahmoja.

4.2 Minipelit

Työn alkuvaiheessa keksittiin erilaiset minipelit ja niiden pelimekaniikat. Pelimekaniikat ovat sääntöjä, jotka ohjaavat pelaajia pelin läpi. Ne määrittävät, miten peli toimii, mitä toimintoja pelaajat voivat suorittaa ja kuinka peli reagoi näihin toimintoihin. (29.) Taulukossa 1 on esitelty alkuperäiset minipelien nimet ja kuvaukset.

Taulukko 1. Minipelit

Pelin nimi	Pelin kuvaus	Opittavat aiheet
Tietovisapeli	Pelissä soitetaan ääninäyte erilaisista esineistä, eläimistä tai toiminnoista, sekä näytetään neljä kuvaa. Lapsen tehtävänä	Eläimet, ammatit, numerot.

Pelin nimi	Pelin kuvaus	Opittavat aiheet
	on tunnistaa ja valita ääninäytettä vastaava kuva.	
Pukeutumispeli	Pelissä lapsille annetaan tehtävä valita oikeat vaatteet, jotka sopivat annettuun säätilaan tai vuodenaikaan. Peliä pelataan valitsemalla erilaisia vaihtoehtoja pelirudulla olevista vaihtoehdoista, kuten vetämällä ja pudottamalla tai napauttamalla sopivaa kuvaketta.	Vaatteet, sää, vuodenaikat.
Esineiden lajittelu	Pelissä näytetään erilaisia esineitä, ja lapsen tehtävänä on lajitella ne oikeisiin kategorioihin molemmilla kielillä.	Värit, muodot, materiaalit.
Viikontähtäjä	Tässä minipelissä pelaajan tavoitteena on järjestää junan vaunut oikeaan järjestykseen. Jokaisessa vaunussa on kirjoitettu yksi viikontähtäjä, kuten maanantai, tiistai ja niin edelleen. Pelaajan on raahattava vaunut oikeaan järjestykseen. Kun pelaaja suorittaa tehtävän, näytetään animaatio, jossa juna lähtee matkaan.	Viikontähtäjät
Muistipeli	Pelissä on kuvakortteja, joissa on erilaisia esineitä tai eläimiä. Lapsen tehtävänä on löytää samanlaiset kortit ja yhdistää ne toisiinsa. Joka kerta kun pelaaja avaa kortin, kuuluu ääninäyte kyseisen kohteen nimestä.	Eläimet, ammatit, numerot.

Projektia varten päätettiin toteuttaa kaksi minipeliä: pukeutumis- ja tietovisapeli. Jokaisessa minipelissä on oma ydinsilmukka (englanniksi *core loop*). Pelin ydinsilmukka on ensisijainen pelimekaniikka. Se koostuu toistuvista toiminnoista, joita käyttäjä suorittaa pelatessaan peliä. Hyvin suunniteltu ydinsilmukka voi pitää pelaajan keskittyneenä pelin tehtäviin samalla tarjoten sopivasti haastetta. (30.)

Pukeutumispeli on yksi minipeleistä, joka on suunniteltu auttamaan lapsia oppimaan sanoja ja käsitteitä liittyen vaatteisiin ja säähän. Pelin ydinsilmukka koostuu neljästä vaiheesta. Ensin pelaaja selvittää, millainen sää on. Tämän jälkeen pelaajan tehtävänä on valita ja pukea hahmolleen sopivat vaatteet raahaamalla

ne hahmon päälle. Peli antaa palautetta siitä, ovatko valitut vaatteet sopivia kyseiseen säähän. Lopuksi, jos vaatteet on oikein valittu, pelaaja pääsee seuraavalle tasolle.

Tietovisapeli on toinen minipeli, joka on suunniteltu auttamaan lapsia oppimaan uusia sanoja ja niiden merkityksiä. Pelin ydinsilmukka koostuu neljästä vaiheesta. Ensin pelaaja kuuntelee sanan opetettavalla kielellä. Sen jälkeen pelaaja voi valita yhden neljästä kuvavaihtoehdosta, joka hänen mielestään vastaa kuultua sanaa. Peli antaa palautetta siitä, oliko vastaus oikein vai väärin. Oikean vastauksen valinnut pelaaja pääsee etenemään seuraavalle tasolle.

4.3 Käytetyt teknologiat

Tässä insinööriyössä kehitettiin kaksiulotteinen (2D) mobiilipeli Android-alustoille käyttäen Godot-pelimootoria. Pelin grafiikka tuotettiin ProCreate-sovelluksella käyttäen iPad-tablettia ja kosketuskynää. Ääninauhoitteet tehtiin Audacity-ohjelmalla.

Peliprojektia varten kokeiltiin erilaisia tekoälysovelluksia, jotka joko tuottavat musiikkia, kuten Suno-sovellus, tai muuntavat tekstiä puheeksi (englanniksi text-to-speech), kuten ElevenLabs, Narakeet ja Genny. Näin peliin luotiin lisää erilaisia ääniä.

Tässä projektissa käytettiin myös GLES2-nimistä grafiikkapohjaista ohjelmointirajapintaa (API), joka tarjoaa paremman suorituskyvyn ja yhteensopivuuden vanhemmilla tai vähemmän tehokkailla mobiililaitteilla. (31.)

5 Kielikaverit-pelin toteutus

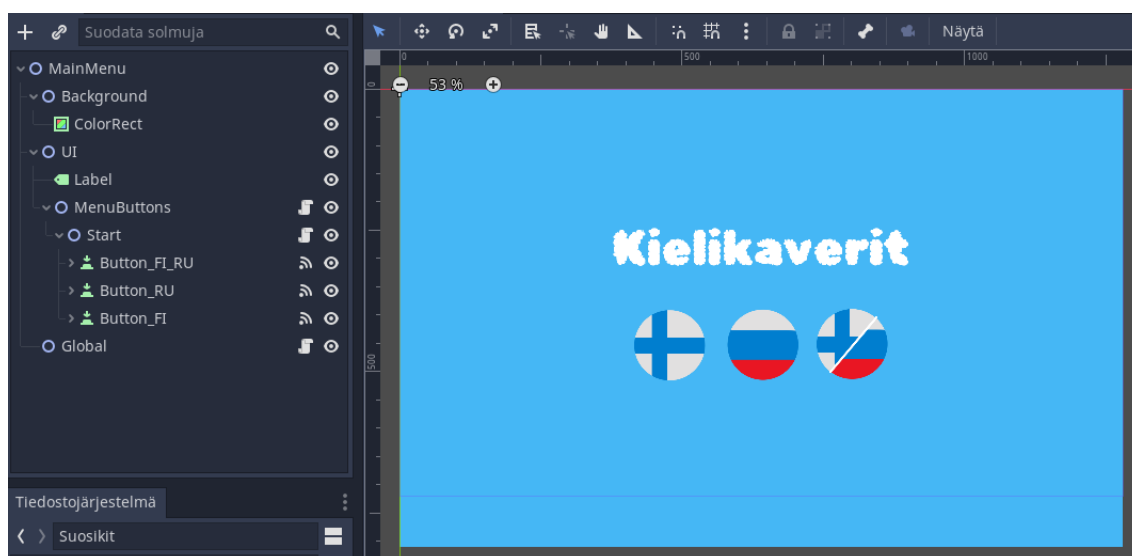
Tässä luvussa käsitellään, miten Kielikaverit-pelin kehityksessä voidaan hyödyntää Godot-pelimootorin editoria ja sen sisäisiä menetelmiä. Pelissä toteutetaan kaksi minipeliä: pukeutumispeli ja tietovisapeli.

5.1 Käyttöliittymä

Käyttöliittymän suunnittelu on ensimmäinen asia, jonka pelaajat näkevät pelissä. Hyvin suunniteltu käyttöliittymä parantaa käyttäjän kokemusta ja kannustaa heitä jatkamaan pelaamista. Esimerkiksi etenemispalkki ja kerätyt pisteet voivat motivoida opiskelijoita pysymään pelissä. (32; 33.)

Tavoitteena oli luoda yksinkertainen käyttöliittymä, jotta opettajat ja lapset osaisivat navigoida helposti. Käyttöliittymän suunnittelussa on otettava huomioon, että lapset eivät välttämättä osaa lukea. Siksi tekstin sijaan tulisi käyttää kuvia tai muita visuaalisia elementtejä, jotka ovat helpommin ymmärrettäviä ja houkuttelevia lapsille.

Pelin aloitusnäkyssä esitellään kolme painiketta, joista kukin edustaa oppimiskieltä: suomi, venäjä ja molemmat samanaikaisesti (kuva 10). Pelaaja voi valita haluamansa kielen näistä painikkeista aloittaessaan pelin.

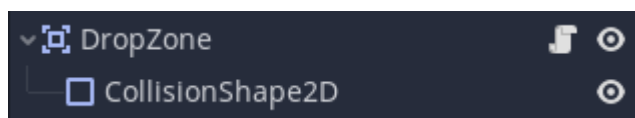


Kuva 10. Godot-editorin näkymä pelin aloitusikkunasta.

5.2 Pukeutumispelin toteutus

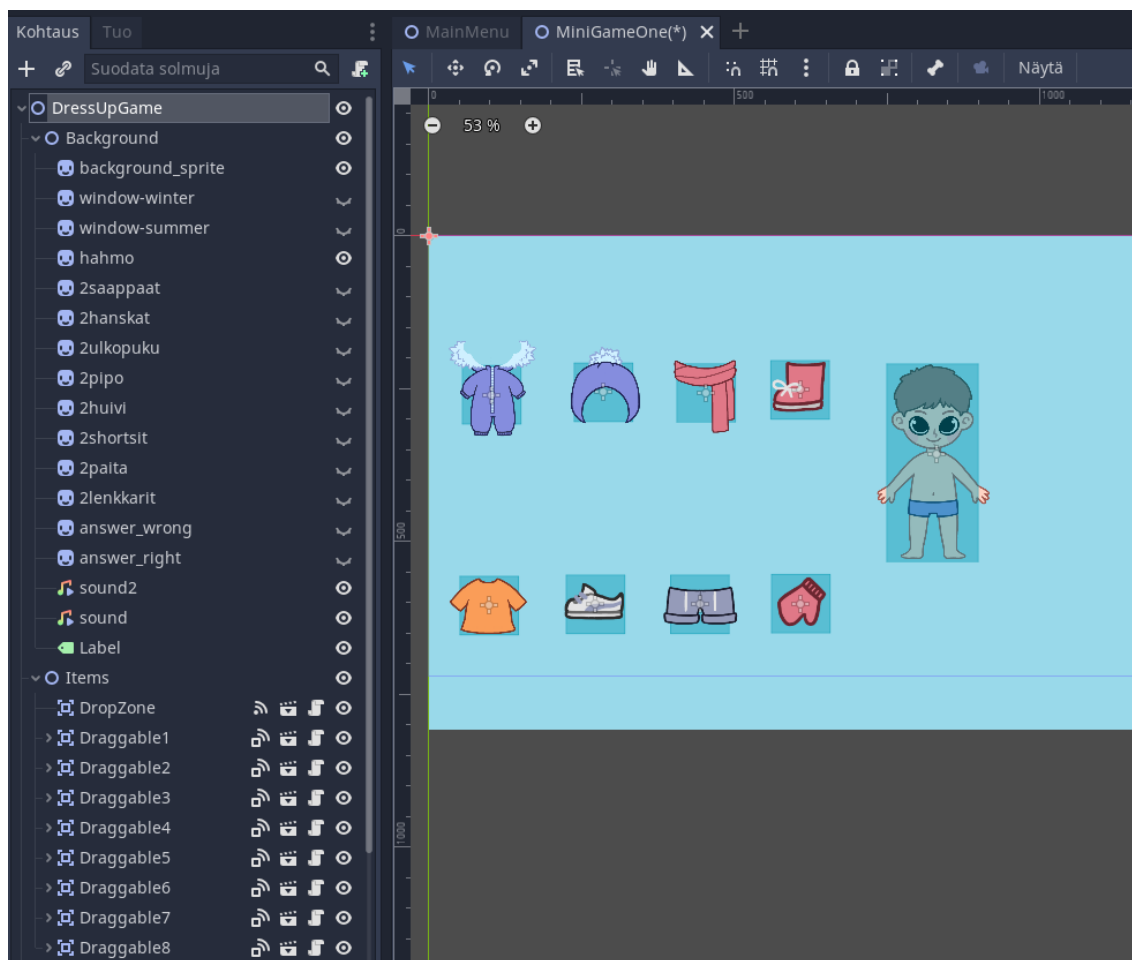
Pelaajan tavoite on pukea hahmo sopiviin vaatteisiin siirtämällä vaatekappaleita sen päälle. Pukeutumispelia varten toteutettiin raahaa ja pudota -toiminto (englanniksi *drag and drop*).

Luodaan kaksi solmua: yksi hahmolle ja toinen vaatekappaleelle. Hahmon ja vaatekappaleiden toteutuksessa käytetään Area2D-solmua, joka sisältää CollisionShape2D-solmun määrittämään 2D-kohteen törmäysalueen. Määritetään kunkin CollisionShape2D-solmun muoto vastaamaan hahmon ja vaatekappaleen muotoja (kuva 11).



Kuva 11. Hahmon solmun sisältö.

Asetetaan hahmo- ja kahdeksan vaatekappalesolmua pukeutumispelin näkymään, joka on esitelty kuvassa 12. Hahmon solmun nimi on DropZone ja vaatekappaleiden solmujen nimet ovat Draggable1, Draggable2 jne. Jokaiselle vaatekappaleelle luodaan myös Sprite-solmu, joka asetetaan sopivasti hahmon päälle. Kaikki nämä Sprite-solmut tehdään näkymättömiksi.



Kuva 12. Godot-editorin näkymä pukeutumispelistä.

Käytetään Area2D-solmua törmäysten havaitsemiseen. Hiiren sijaintia käytetään laskemaan, kuinka paljon vaatekappaleen solmua tulisi liikuttaa. Jos pelaaja koskettaa näyttöä, hiiren sijainti asetetaan uudeksi sijainniksi. Pudotusalue siirtää kaikki törmäävät Area2D-solmut pois näkyvistä ja tekee vaatekappaleen Sprite-solmun näkyväksi, kun pelaaja vapauttaa hiiren painikkeen.

Seuraavana toteutetaan vuodenajan vaihtelua ja sen havaitsemista. Lisätään vaatekappaleiden solmuille kaksi ryhmää: yksi vastaa niiden nimitystä (esimerkiksi ulkopuku, pipo tai paita) ja toinen vuodenaikaa. Ryhmien järjestämisen ja hallinnan helpottamiseksi ryhmän nimen eteen lisätään numero. Tässä on esimerkki käyttämistämme ryhmien nimistä: 1talvi, 1syksy, 2paita, 2pipo.

Esimerkkikoodissa 2 esitetään pudotusalueen skriptin metodi, joka alkaa suorittaa toimintoja, kun se havaitsee Area2D-solmujen törmäyksen. Ensin metodi siirtää törmäävän Area2D-solmun pois näkyvistä. Tämän jälkeen se tarkistaa, kuuluuko kyseinen Area2D-solmu ryhmään, jonka nimi vastaa rand_season-muuttujaa, joka edustaa satunnaisesti valittua vuodenaikaa. Seuraavaksi koodi tarkistaa Area2D-solmun jokaisen ryhmän nimen ja tallentaa sen ryhmän nimen, joka alkaa numerolla "2", muuttujaan "s". Lopuksi metodi tekee saman nimen vaatekappaleen Sprite-solmun näkyväksi.

```
func _input(event):
    if event.is_action_released("ui_touch"):
        for _a in get_overlapping_areas():

            _a.position.x = 1000
            _a.position.y = 1000

            if _a.is_in_group(rand_season):

                var s
                for n in _a.get_groups().size():
                    if (group[n].left(1) == "2"):
                        s = group[n]

                var node = get_node("../..//Background/" + s)
                node.visible = true
```

Esimerkkikoodi 2. Pudotusalueen skriptin metodi alkaa suorittaa toimintoja, kun se havaitsee Area2D-solmujen törmäyksen.

Randomize_items() -metodi sijoittaa vaatekappaleiden solmuja satunnaisesti paikkoihin (esimerkkikoodi 3). Määritellään kaksi taulukkoa: Positions ja Draggables. Positions-taulukko sisältää koordinaatit. Draggables-taulukko sisältää vaatekappaleiden solmujen tunnukset. Sekoitetaan Draggables-taulukko käyttämällä shuffle()-metodia, joka järjestää alkiot satunnaiseen järjestykseen. Käytetään myös randomize()-metodia varmistaaksemme, että alkioiden järjestys ei toistu joka kerta. Seuraavaksi käydään läpi kaikki draggables-taulukon alkiot käyttämällä silmukkaa. Jokaiselle alkioille haetaan solmu tunnukseksi ja asetetaan solmun x- ja y-koordinaatit positions-taulukosta.

```

func randomize_items():
    var positions = [100,200, 233.3,200, 366.6,200, 500,200,
                    100,500, 233.3,500, 366.6,500, 500,500]
    var draggables = [1,2,3,4,5,6,7,8]

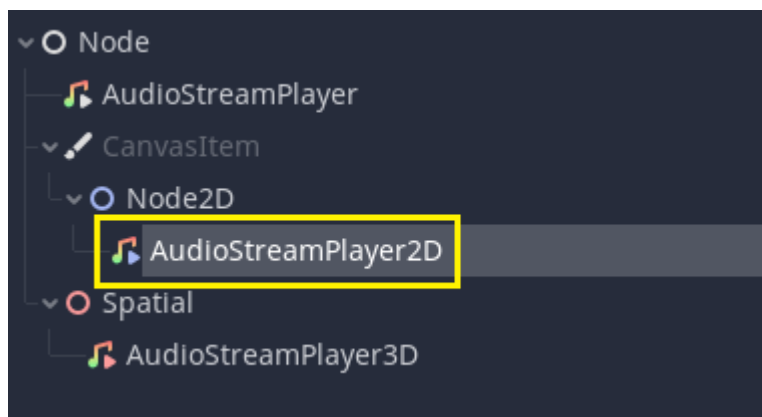
    randomize()
    draggables.shuffle()

    var a = 0
    for n in draggables:
        get_node("../Items/Druggable" + str(n)).position.x =
            positions[a]
        a = a + 1
        get_node("../Items/Druggable" + str(n)).position.y =
            positions[a]
        a = a + 1

```

Esimerkkikoodi 3. Metodi, joka käyttää satunnaisgeneraattoria valitakseen vaa-
tekappaleiden solmujen uuden järjestyksen.

Äänen toteutuksessa käytettiin AudioStreamPlayer2D-solmua, joka mahdollis-
taa äänen soittamisen 2D-peleissä (kuva 13).



Kuva 13. AudioStreamPlayer2D-solmu.

Esimerkkikoodi 4 esittää playSound-metodin, joka lataa ja soittaa ääninauhoi-
teen tiedostojärjestelmästä. Load-metodi lataa ääniresurssin tiedostojärjestel-
mästä. Stream-metodi asettaa ladatun ääninauhoitteen soittimen äänilähteeksi.
Play-metodi soittaa äänen.

```
onready var sound = get_node("../Background/AudioStreamPlayer2D")

func playSound(s):
    var audio = load(s)
    sound.stream = audio
    sound.play()
```

Esimerkkikoodi 4. PlaySound-metodi.

AudioPlayer-metodi soittaa ääninauhoitteet käyttäjän kieliasetuksen perusteella (esimerkkikoodi 5). Luotiin ääninauhoitteita, joiden nimet ovat esimerkiksi: "mekko-fi.mp3" ja "mekko-ru.mp3". Äänitiedostojen nimissä "-fi" tarkoittaa suomenkielistä versiota ja "-ru" venäjänkielistä versiota. Jos käyttäjä on valinnut molemmat kielet samanaikaisesti, soitetaan ensin suomenkielinen ääninauhoite. Odotetaan, kunnes nauhoite on soinnut loppuun käyttämällä yield-metodia, ja sen jälkeen soitetaan venäjänkielinen ääninauhoite.

```
func audioPlayer():

match Global.get_language():
    "fi":
        playSound("res://Audios/" + s + "-fi.mp3")
    "ru":
        playSound("res://Audios/" + s + "-ru.mp3")
    "fi-ru":
        playSound("res://Audios/" + s + "-fi.mp3")
        yield(sound, "finished")
        playSound("res://Audios/" + s + "-ru.mp3")
```

Esimerkkikoodi 5. AudioPlayer-metodi.

5.3 Tietovisapelin toteutus

Tietovisa on yksinkertainen minipeli, jossa pelaajalle näytetään neljä kuvaa ja soitetaan yksi ääninäyte opetettavalla kielellä. Pelaajan tehtävänä on valita oikea kuva, joka vastaa ääninäytettä.

Tietovisaa varten piirrettiin kuvia eri aiheista, kuten vihanneksista, hedelmistä, eläimistä, numeroista ja ammateista (kuva 14).



Kuva 14. Tietovisaa varten piirretyt kuvat.

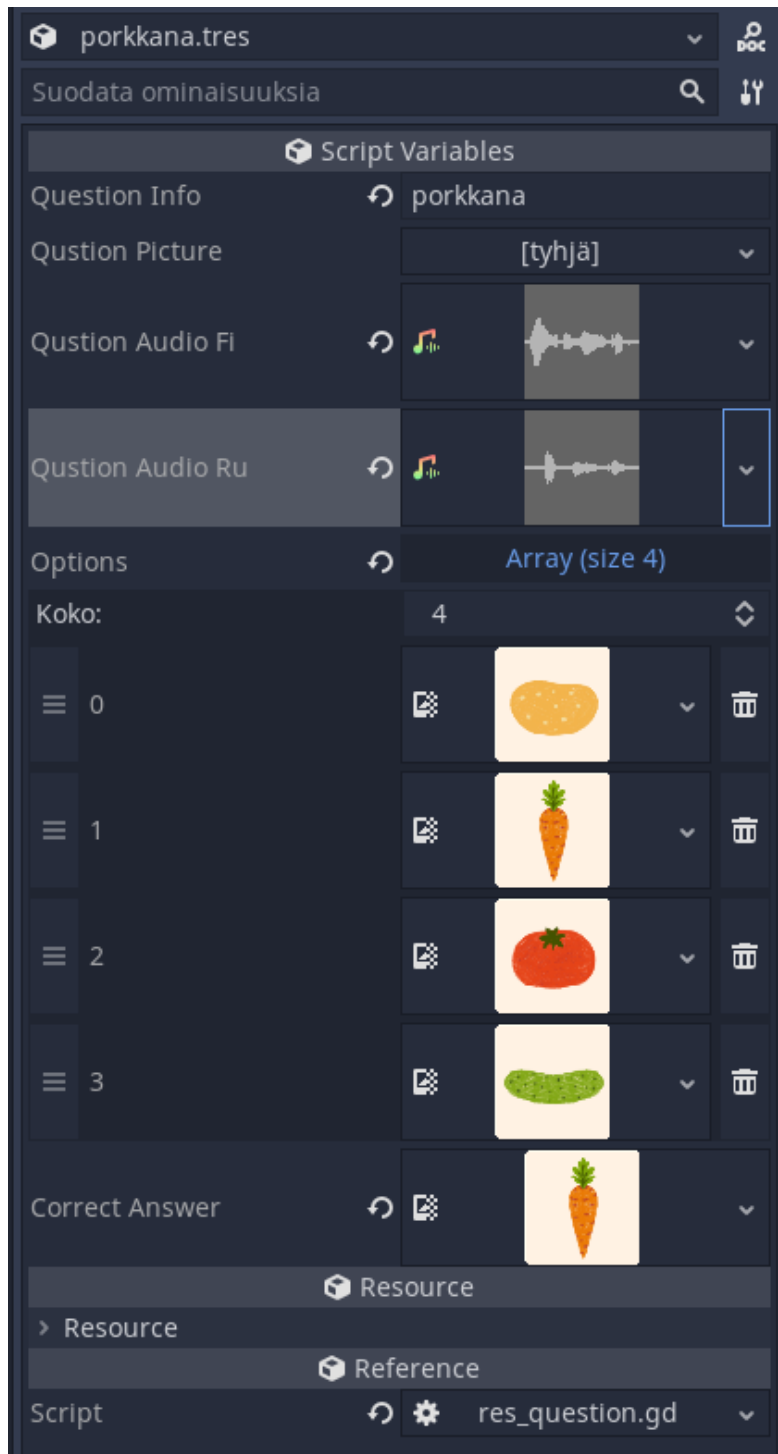
Tietovisan kehityksessä luotiin kaksi skriptiä. Ensimmäisen skriptin nimi on "res_question", joka vastaa yksittäistä kysymystä (esimerkkikoodi 6). Jokainen kysymys sisältää kaksi ääninauhoitetta, toisen suomeksi ja toisen venäjäksi, vastausluettelon sekä oikean vastauksen. Vastausluettelo ja oikea vastaus ovat tekstuurityyppiä. Toisen skriptin nimi on "res_array" (esimerkkikoodi 7). Tämä skripti sisältää taulukon, joka koostuu useista kysymysresursseista. Taulukkoa käytetään resurssien järjestämiseen eri aiheisiin.

```
extends Resource
```

```
export (AudioStream) var qustion_audio_FI
export (AudioStream) var qustion_audio_RU
export (Array, Texture) var answers
export (Texture) var correct_answer
```

Esimerkkikoodi 6. Res_question-skripti, joka sisältää yksittäisen kysymyksen resurssityypin koodin.

Kuvassa 15 on esitelty Godot-editorin ikkuna, jossa on valittu yksi kysymysresurssi eli "porkkana.tres"-resurssi. Tämä resurssi sisältää kysymyksen, johon liittyy neljä kuvaa: peruna, porkkana, tomaatti ja kurkku. Kysymyksen oikea vastaus on porkkana.



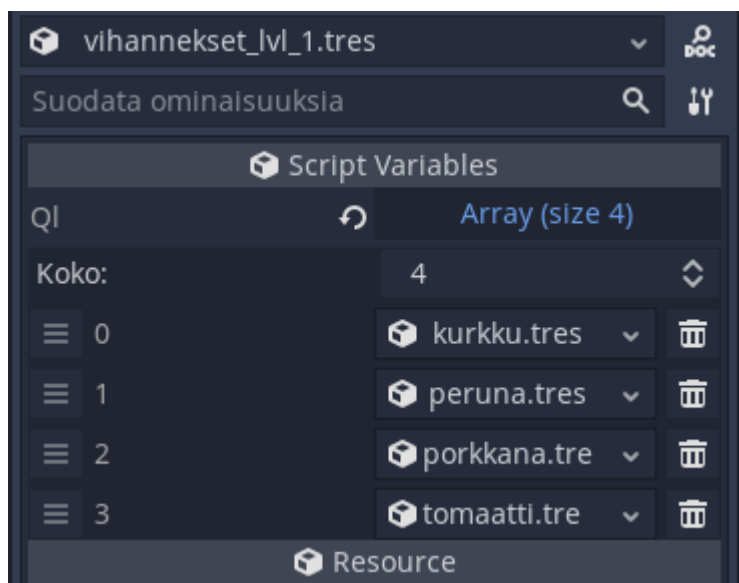
Kuva 15. Tietovisan "porkkana.tres" kysymysresurssi Inspector-ikkunassa.

```
extends Resource
```

```
export (Array, Resource) var question_list
```

Esimerkkikoodi 7. Res_array-skripti, joka sisältää taulukon resurssien järjestämiseen eri aiheisiin.

Kuva 16 esittelee Godot-editorin inspector-ikkuna, jossa on valittu ”vihannekset_lv1_1.tres” resurssi. Tämä resurssi sisältää taulukko, jossa on neljä vihanneksiin liittyvää kysymystä. Taulukon kokoa voidaan muokata suoraan Inspector-ikkunassa.



Kuva 16. Tietovisan ”vihannekset_lv1_1.tres” resurssi Inspector-ikkunassa.

Lisäksi toteutettiin aiheiden ja tasojen vaihtelu. Kuvassa 17 on esitelty tietovisan alussa avautuva ikkuna, jossa näytetään aiheet. Kun pelaaja valitsee haluamansa aiheen, hänelle ehdotetaan valitsemaan taso. Taso-järjestelmä luotiin rajoittamaan tietojen määrää, estämään lasten tylsistymistä sekä lisäämään haastetta.



Kuva 17. Tietovisan aiheet-ikkuna.

Esimerkkikoodi 8 näyttää `open_level`-metodin, jonka avulla voidaan ladata tarvittava taso napin painalluksella tiedostojärjestelmästä.

```
func open_level (button) -> void:
    questions = load("res://Quiz/QL/" + theme + "_lvl_" + button.name
        + ".tres")
```

Esimerkkikoodi 8. `Open_level`-metodi, joka lataa kysymysluettelon napin painalluksella.

5.4 Pelin rakentaminen

Godot-pelimoottorissa on mahdollista luoda kaksi erilaista versiota pelistä, eli buildejä: `debug build` ja `release build`. `Debug build` on tarkoitettu kehitystyöhön ja virheiden vianmääritykseen, kun taas `release build` on tarkoitettu loppukäyttäjille ja julkaistavaksi. `Release build` on optimoitu suorituskyvyn ja tehokkuuden maksimoimiseksi. (34.)

6 Valmiin pelin arviointi

6.1 Pelin testaaminen, lopputulokset ja palautteet

Pelin testaus oli suunniteltu suorittaa muutama kerta "Kalinka"-nimisessä päiväkodissa. Peliä testattiin yhdessä 4–5-vuotiaiden lasten ryhmässä, johon kuului yhteensä 21 lasta. Käyttäjäryhmässä oli 11 äidinkieleltään suomenkielistä lasta, 5 venäjänkielistä lasta ja 5 kaksikielistä lasta.

Testauksen aikana lapset osoittivat suurta kiinnostusta peliä kohtaan ja olivat innokkaita osallistumaan. Palautteen perusteella voitiin kuitenkin todeta, että osa lapsista koki pelin liian helpoksi. Tämä viittaa siihen, että pelin vaikeustaso ei ollut täysin optimaalinen kaikille.

Lapset pitivät erityisesti pelin ääninauhotteista ja animaatioista, joita näytettiin oikeista vastauksista. Nämä elementit toivat peliin lisäiloa ja motivoivat lapsia jatkamaan pelaamista.

Lapset huomasivat pukeutumispelissä virheen. Välillä peli palkitsi pelaajaa, vaikka hän ei ollut pukeutunut kaikkia tarvittavia vaatteita päälle. Lapset pitivät tätä virhettä erittäin hauskana ja kiinnittivät erityistä huomiota juuri pukeutumispeliin. Tämä osoittaa, että peli voi houkutella lapsia ja pitää heidän mielenkiintonsa yllä, mutta samalla on tärkeää huomata, että tämä virhe vähensi heidän kiinnostustaan muihin peleihin. Virheen korjaaminen on olennaista, jotta lapset saavat parhaan mahdollisen pelikokemuksen ja jakavat huomionsa tasaisesti kaikkiin pelin osiin.

Lisäksi opettajat huomasivat, että pelin äänet olivat liian hiljaisia, mikä vaikutti lasten pelikokemukseen. Päiväkodin taustääänet tekivät pelin äänistä vaikeasti kuultavia, mikä haittasi pelaamista.

Jatkokehityksessä on tärkeää ottaa huomioon sekä pelin vaikeustason säätö että äänitasojen parantaminen, jotta peli tarjoaa sopivia haasteita kaikille pelaajille ja miellyttävän äänikokemuksen myös meluisassa ympäristössä.

Ensimmäisen testikerran aikana ilmeni, että yksi suomenkielinen lapsi ei pärjännyt käyttäessään sovellusta pelkästään venäjän kielellä. Tämä tilanne herätti ajatuksen siitä, että peliin voitaisiin lisätä ominaisuus, joka tarjoaa kaksi kieltä samaan aikaan. Tällä tavoin peli ei olisi kokonaan venäjäksi tai suomeksi, vaan siinä ensin sanottaisiin sana suomeksi ja sen jälkeen toistettaisiin se venäjäksi. Tällainen ratkaisu voisi tukea kaksikielisten lasten oppimista entistä paremmin ja tarjota mahdollisuuden kehittää kielitaitoa molemmilla kielillä samanaikaisesti.

Päiväkodin opettajat halusivat ottaa pelin aktiivisesti käyttöön. Opettajien mukaan peli auttaa kannustamaan ja herättämään lasten kiinnostuksen kielen opiskeluun digitalisen työkalun avulla.

6.2 Pelin tulevaisuus

Peliä käytetään Kalinkan opetuksessa digitaalisen tuen välineenä. Kehittäjän tavoitteena on kehittää ja laajentaa peliä tulevaisuudessa lisäämällä uusia minipelejä, saavutuksia ja palkintoja sekä parantaa pelin kuvia, äänitteitä ja ääniefektejä. Suunnitelmissa on myös lisätä puheentunnistus, mikä voi olla haastavaa erityisesti lasten puheen tunnistamisen tarkkuuden vuoksi.

7 Yhteenveto

Opetuspelit ovat tehokas tapa yhdistää oppiminen ja viihde, ja niitä voidaan käyttää erilaisten taitojen kehittämiseen, kuten kielten oppimiseen.

Insinööriyössä toteutettiin yksinkertainen mobiililaitteille suunnattu kielioppimispelejä käyttämällä Godot-pelimootoria. Peli on suunnattu 3–6-vuotiaille lapsille. Pelin tavoitteena oli herättää lasten kiinnostus kielten oppimiseen hausalla ja vuorovaikutteisella tavalla. Peli koostuu useista minipeleistä, joista tähän mennessä on kehitetty tietovisa ja pukeutumispeli.

Insinööryön projektissa tutustuttiin Godot-pelimoottoriin ja arvioitiin sen soveltuvuutta kieltenopetuspelin toteuttamiseen. Godot on ilmainen ja avoimen lähdekoodin pelimoottori, joka soveltuu hyvin 2D-mobiilipelien kehittämiseen.

Insinööryön projektin alkuvaiheessa kehitys eteni hitaasti, koska pelimoottori ei ollut tuttu. Kuitenkin ajan myötä työskentely nopeutui, kun pelimoottorin ominaisuuksiin ja työkaluihin tutustuttiin paremmin. Tällainen peli vaatii runsaasti grafiikkaa ja äänitiedostojen luomista, mikä toi mukanaan lisähaasteita kehitysprosessiin.

Kokonaisuutena projekti osoitti, että mobiilipelimarkkinoilla löytyy useita kieltenopetuspelejä, mutta ei ole paljon pelejä, jotka on suunniteltu juuri venäjän ja suomen kielen opettamiseen. Markkinoilla on tarve pelistä, joka tukisi kaksikielisyttä ja olisi erityisesti suunniteltu näiden kielten oppimiseen.

Peli auttaa kaksikielisessä ympäristössä kasvavia lapsia laajentamaan sanavarustoaan sekä suomen että venäjän kielillä. Peli tarjoaa lapsille mahdollisuuden toistaa sanoja useita kertoja, mikä tukee oppimista ja vahvistaa muistamista. Varhaislapsuudessa kielen oppiminen on erityisen tehokasta, ja tämän pelin avulla lapset voivat syventää ymmärrystään molemmista kielistä leikkillisessä ympäristössä.

Lähteet

- 1 Multilingual People. 2018. Verkkoaineisto. Ilanguages. <<https://ilanguages.org/bilingual.php>>. Luettu 8.5.2024.
- 2 Döpke Susanne. Children in more than one Language: Language Development in Bilingual Children. Verkkoaineisto. <https://arts.unimelb.edu.au/__data/assets/pdf_file/0010/3546379/Language_development.pdf>. Luettu 5.3.2024.
- 3 Vähemmistökieli. 2022. Verkkoaineisto. Wikipedia. <<https://fi.wikipedia.org/wiki/V%C3%A4hemmist%C3%B6kieli>>. Luettu 10.6.2024.
- 4 Monikielisyyden tukeminen varhaiskasvatuksessa. Verkkodokumentti. Opetushallitus. <https://www.oph.fi/sites/default/files/documents/monikielisyyden_tukeminen_varhaiskasvatuksessa.pdf>. Luettu 5.3.2024.
- 5 Limacher, Ute. Family Language Strategies. Verkkoaineisto. Multilingual Families. <https://multilingual-families.com/family-language-strategies/#Time_and_Place>. Luettu 5.3.2024.
- 6 De Houwer, Annick. Why do so many children who hear two languages speak just a single language? Verkkoaineisto. <https://www.researchgate.net/publication/341121459_Why_do_so_many_children_who_hear_two_languages_speak_just_a_single_language>. Luettu 7.3.2024.
- 7 Harrison, Haley. Which Family Language Strategy Works for You? <<https://www.spanish.academy/blog/which-family-language-strategy-works-for-you/>>. Luettu 5.3.2024.
- 8 International language standards. 2024. Verkkoaineisto. Cambridge. <<https://www.cambridgeenglish.org/exams-and-tests/cefr/>>. Luettu 27.2.2024.
- 9 Common European framework of reference for languages: learning, teaching, assessment. 2018. Verkkoaineisto. Council of Europe. <<https://rm.coe.int/cefr-companion-volume-with-new-descriptors-2018/1680787989>>. Luettu 10.6.2024.
- 10 Study English level A1: Level descriptors & Study targets (CEFR A1). EnglishRadar. <<https://www.englishradar.com/english-levels/english-level-a1/>>. Luettu 12.6.2024.

- 11 Gehring, Sonja & Heinzmann, Sanni. 2010. Oppikirja. Suomen mestari. Finn Lectura. Luettu 10.3.2024.
- 12 Mogavi RH, Guo B, Zhang Y, Ul-Haq E, Hui P, Ma X. 2022. When Gamification Spoils Your Learning: A Qualitative Case Study of Gamification Misuse in a Language-Learning App. <https://www.researchgate.net/publication/359578682_When_Gamification_Spoils_Your_Learning_A_Qualitative_Case_Study_of_Gamification_Misuse_in_a_Language-Learning_App>. Luettu 5.3.2024.
- 13 Leppänen, Sami; Lozano, Carmelo Enrico; Hayes, Gene. Verkkoaineisto. What are the potential drawbacks of using gamification for e-learning? LinkedIn. <<https://www.linkedin.com/advice/0/what-potential-drawbacks-using-gamification-e-learning/>>. Luettu 10.6.2024.
- 14 Leading language learning apps worldwide in January 2024, by downloads. 2024. Verkkoaineisto. Statista. <<https://www.statista.com/statistics/1239522/top-language-learning-apps-downloads/>>. Luettu 20.4.2024.
- 15 Duolingo. 2024. Verkkoaineisto. Wikipedia. <<https://en.wikipedia.org/wiki/Duolingo>>. Luettu 15.4.2024.
- 16 Parker, Henry. 2024. Verkkoaineisto. Can I use ChatGPT to practice a new language? Duolingo Blog. <<https://blog.duolingo.com/chatbot-language-practice/>>. Luettu 20.4.2024.
- 17 The Duolingo Method for App-based Teaching and Learning. 2023. Verkkoaineisto. Duolingo. <https://duolingo-papers.s3.amazonaws.com/reports/Duolingo_whitepaper_duolingo_method_2023.pdf>. Luettu 20.4.2024.
- 18 Galaxy Kids - Learning English. 2024. Verkkoaineisto. Google Play. <<https://play.google.com/store/apps/details?id=com.Robotgalaxykids.ga2&hl=fi>>. Luettu 20.7.2024.
- 19 Our Learning Path. Verkkoaineisto. Galaxy Kids. <<https://galaxykids.ai/curriculum/learning-path/>>. Luettu 20.7.2024.
- 20 Gus on the Go app review. 2018. Verkkoaineisto. Les Petits Tigres. <<https://www.lespetitstigres.com/post/gus-on-the-go-app-review>>. Luettu 20.7.2024.
- 21 Godot (game engine). 2024. Verkkoaineisto. Wikipedia. <[https://en.wikipedia.org/wiki/Godot_\(game_engine\)](https://en.wikipedia.org/wiki/Godot_(game_engine))>. Luettu 27.2.2024.

- 22 Introduction. Verkkoaineisto. Godot Docs. <<https://docs.godotengine.org/en/stable/about/introduction.html#about-godot-engine>>. Luettu 27.2.2024.
- 23 Using SceneTree. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/3.5/tutorials/scripting/scene_tree.html>. Luettu 10.4.2024.
- 24 Scripting languages. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/stable/getting_started/step_by_step/scripting_languages.html>. Luettu 27.2.2024.
- 25 Nodes and Scenes. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/stable/getting_started/step_by_step/nodes_and_scenes.html>. Luettu 27.2.2024.
- 26 Resources. Verkkoaineisto. Godot Docs. <<https://docs.godotengine.org/en/stable/tutorials/scripting/resources.html>>. Luettu 7.3.2024.
- 27 Using signals. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/stable/getting_started/step_by_step/signals.html>. Luettu 28.2.2024.
- 28 Groups. Verkkoaineisto. Godot Docs. <<https://docs.godotengine.org/en/stable/tutorials/scripting/groups.html>>. Luettu 6.3.2024.
- 29 The Ultimate Beginner's Guide To Game Mechanics. 2023. Verkkoaineisto. Juego Studio. <<https://www.juegostudio.com/blog/the-ultimate-beginners-guide-to-game-mechanics>>. Luettu 10.4.2024.
- 30 Lovato, Nathan. 2017. How To Perfect Your Game's Core Loop. Verkkoaineisto. Game Analytics. <<https://gameanalytics.com/blog/how-to-perfect-your-games-core-loop/>>. Luettu 11.4.2024.
- 31 Differences between GLES2 and GLES3. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/3.5/tutorials/rendering/gles2_gles3_differences.html>. Luettu 6.3.2024.
- 32 Soyluççek, Seza. 2012. Graphic Design on Educational Computer Games. Verkkoaineisto. <https://www.researchgate.net/publication/271609827_Graphic_Design_on_Educational_Computer_Games>. Luettu 11.4.2024.
- 33 Anugerah, Yope & Budiyanto, Cucuk. 2017. The Design of Children Educational Game Interface: Review of the Literature. Verkkoaineisto.

<https://www.researchgate.net/publication/321844483_The_Design_of_Children_Educational_Game_Interface_Review_of_the_Literature>. Luettu 11.4.2024.

- 34 Exporting for Android. Verkkoaineisto. Godot Docs. <https://docs.godotengine.org/en/stable/tutorials/export/exporting_for_android.html>. Luettu 29.2.2024.