

# Programiranje i produkcija zvuka u razvoju video igre

---

**Pejčinović, Matija**

**Undergraduate thesis / Završni rad**

**2023**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Rijeka / Sveučilište u Rijeci**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:195:543858>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-05-18**



Sveučilište u Rijeci  
**Fakultet informatike  
i digitalnih tehnologija**

*Repository / Repozitorij:*

[Repository of the University of Rijeka, Faculty of  
Informatics and Digital Technologies - INFORI  
Repository](#)



Sveučilište u Rijeci – Fakultet informatike i digitalnih tehnologija

Preddiplomski studij Fakulteta informatike i digitalnih tehnologija

Matija Pejčinović

# Programiranje i produkcija zvuka u razvoju video igre

Završni rad

Mentor: doc. dr. sc. Marina Ivašić Kos

Rijeka, 11.06.2023.

Izjavljujem da sam ovaj rad napravio samostalno pomoću dosad stečenih znanja i vještina tijekom studija te korištenjem navedene literature.

Potpis: \_\_\_\_\_

Zahvaljujem se mentorici, izv. prof. dr. sc. Marini Ivašić Kos na savjetovanju, strpljenju i izdvojenom vremenu kako bi što uspješnije dovršio završni rad.

Također želio bih se zahvaliti svim profesorima i asistentima Fakulteta informatike i digitalnih tehnologija u Rijeci na prenesenom znanju koje sam usvojio.

Konačno, zahvalio bih se svojoj obitelji na bezuvjetnom pomaganju te svojim prijateljima i kolegama na nesebičnoj podršci tijekom studiranja.

Matija Pejčinović

# Sažetak

Na početku ovog rada prikazan je povijesni presjek programiranja i produkcije zvuka u računalnim igrama. Središnji dio rada obuhvaća osnove fizike zvuka, važne pojmove potrebne za audio programiranje te detaljno opisuje vještine i potrebna predznanja budućeg audio programera. Naposljetku, na konkretnom primjeru pojašnjen je spoj Google Resonance Audio worklet-a i Unity razvojnog alata.

## Ključne riječi

Audio programiranje, Google Resonance Audio, Unity, zvuk, razvoj zvuka, zvučni efekti, zvučni val, računalna igra, glazbene vještine, producentske vještine, refleksija, reverb.

# Sadržaj

1. Uvod .....	1
1.1. Film - preduvjet za razvoj zvuka u video igricama .....	1
1.2. Razdoblje video igara bez korištenja zvuka .....	3
1.3. Razdoblje video igara s uključenim zvukom .....	4
1.3.1. Sedamdesete godine 20. stoljeća .....	4
1.3.2. Osamdesete godine 20. stoljeća .....	6
1.3.3. Integriranje zvučnih računalnih čipova .....	6
1.4. Pogled u budućnost: Uređaji koji poboljšavaju iskustvo igre .....	9
2. Fizika zvuka .....	11
3. Osnovna znanja potrebna za audio programiranje .....	12
3.1. Glazbene vještine, kreativnost i komponiranje glazbe .....	12
3.2. Znanstveni pristup i inženjersko razmišljanje .....	13
3.3. Producentske vještine .....	13
3.4. Tehničko i računalno znanje .....	14
4. Google Resonance Audio .....	15
4.1. Određivanje horizontalne lokacije izvora zvuka .....	16
4.2. Određivanje vertikalne lokacije izvora zvuka .....	16
4.3. Simulacija zvučnih valova u interakciji sa njihovom okolinom .....	17
4.3.1. Direktan zvuk .....	17
4.3.2. Rana refleksija .....	18
4.3.3. Kasni reverb .....	18
5. Resonance Audio i Unity .....	19
5.1. Unity sučelje .....	21
5.2. Postavljanje Resonance Audio komponenti .....	22
5.3. Vrijeme reverba .....	23
5.4. Programiranje reverba (engl. Reverb Baking) .....	23
5.4.1. Operacije sa reverb sondama (engl. Reverb Probes) .....	24
5.4.2. Mapiranje vizualnih materijala sa sondama i akustičnim materijalima .....	25
5.4.3. Uključivanje specifičnih objekata unutar kalkulacije reverba .....	28
5.5. Dodavanje audio komponenti .....	30
6. Zaključak .....	33
7. Literatura .....	34
8. Popis slika .....	37

# 1. Uvod

Važnu ulogu za potpun doživljaj igranja video igara, stvaranje odgovarajuće atmosfere, pobuđivanje različitih osjećaja i uronjenosti u virtualni svijet ima zvuk, tj. muzika i različiti zvučni efekti u igri.

Pojam muzika uključuje glazbu koja se pojavljuje kao pozadina, tema ili glazba unutar igre poput glazbe na radiju ili glazba koju proizvodi nekakav bend. Zvuk označava zvučne efekte poput eksplozija, kretanja, otvaranja vrata te mnogih drugih. U oba slučaja dizajner zvuka može posegnuti za besplatnim, već postojećim materijalima dostupnim na internetu ili otkupiti autorska prava ukoliko zvuk ili glazba nije besplatna. Također zvučnog materijal može snimiti unutar glazbenog studija. Naracija ili glas se pojavljuje u novije vrijeme u funkciji prepričavanja priče između scena. Poznatije kuće koje se bave proizvodnjom igrica često posuđuju glasove stvarnih glumaca kojima dodatno monetiziraju svoju igru [0].

Zvukovi se u video igrama koriste za stvaranje sveukupnog dojma. Zvuk ili muzika uvelike pridonosi zainteresiranosti korisnika te njegovom fokusu kroz vrijeme koje provodi igrajući igre što bi u današnjici bez zvuka bilo gotovo nezamislivo. Audio programer muzikom unosi emociju te se povezuje s karakteristikama lika, stvara napetost situacije u ključnim trenucima igre dok raznim zvukovima povezuje mozak igrača s koncentracijom [1]. Na taj način dolazi do povećanja adrenalina igrača pa on ulaže maksimalan trud kako bi dovršio trenutni cilj igre.

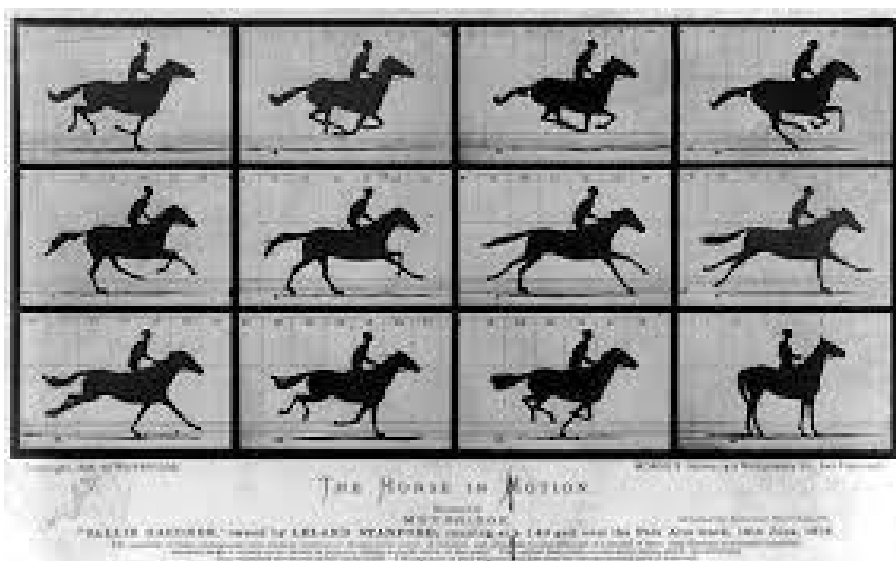
Zvučni efekti mogu biti podijeljeni u različite slojeve ili grupe iako je iz igračeve perspektive tijekom igre zvuk jedinstvena cjelina [2]. Jedni od osnovnih slojeva prema kojima se zvuk grupira su:

- ambijentalni zvuk i zvuk prirode (okoline)
- zvuk izravne povratne informacije igraču (punjenje pištolja, bacanje ručne bombe,...)
- zvukovi sučelja (zvuk klika na gumb ili notifikaciju,...)
- pozadinski ili tematski sloj glazbe.

## 1.1. Film - preduvjet za razvoj zvuka u video igricama

Interaktivni elementi koje pružaju video igre te angažman koji zahtjeva od korisnika potpuno je drukčije iskustvo u odnosu na druge medije. Evolucija računala znatno je utjecala na evoluciju glazbe pa se prema tome glazba u video igrama razvila nakon zvuka u filmu kao mediju.

Godine 1878. Sallie Gardner počeo je eksperimentirati sa slikama u pokretu (Slika 1.) te je javnosti predstavljen Gallop [3].



Slika 1. Sallie Gardner: Gallop - slika u pokretu

Prvi pokušaj sinkronizacije zvuka sa slikom javlja se 1895. godine u naslovu The Dickson Experimental Sound Film [4] kojeg je snimio William Kennedy Laurie Dickson u Edisonovom laboratoriju. Scena iz tog filma prikazana je na slici 2.



Slika 2. Scena iz filma The Dickson Experimental Sound Film

Sinkronizacija glasa, slike, glazbe i boje 1932. godine predstavljena je u crtanom filmu Flowers and Trees [5] producenta Walt Disney-a. Ako pogledamo vremensku lentu, od fotografije u pokretu Gallop do crtanog filma kakvog poznajemo danas trebalo je 54. godine.



## 1.2. Razdoblje video igara bez korištenja zvuka

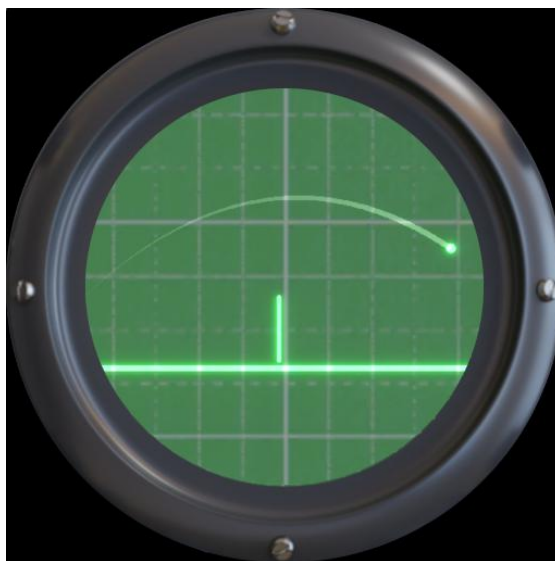
Nakon razvitka filma, uz tehnološki napredak računala dolazi do prosperiteta video igara. Kada pomislimo na prve video igre, zvukovi kojih se prisjećamo prilično su jednostavni. Iako je krenula kao vrlo jednostavna, svega nekoliko različitih frekvencija u službi tona, današnja glazba koju susrećemo u video igrama je stvarna glazba. Ona je komponirana od mnogih talentiranih umjetnika i producenata.

Ako se vratimo u 1947. godinu, prisjetit ćemo se perioda kada glazba u video igricama nije postojala. Naime, te godine se pojavljuje prva elektronička igrica CRTAD (engl. Cathode - Ray Tube Amusement Device) [6] u kojoj smo mogli kontrolirati lasersku zraku na ekranu (Slika 3.).



Slika 3. Prva interaktivna elektronska igra CRTAD, Thomas T. Goldsmith, Estle Ray Mann

Tennis for Two poznatiji pod nazivom Computer Tennis [7] prva je simulacija igre tenisa nastala 1958. godine. Na javnom predstavljanju pokazala je veliki interes javnosti, pogotovo studenata. Igra se prikazuje na osciloskopu [8] te se igra uz pomoć dva aluminijska kontrolera (Slika 4.).



Slika 4. Prikaz igre Tennis for Two na osciloskopu

Zvuk nije postojao niti u prvoj video konzoli razvijenoj 1972. godine pod nazivom Magnavox Odyssey [9]. Iako su igre tog vremena bile bez zvuka, važna je za daljnji razvitak video konzola (slika 5).



Slika 5. Prva video konzola, Magnavox Odyssey

Nakon tog početnog perioda u video igre se počinje uključivati i zvuk. Prvi zvukovi u video igrama bili su prilično jednostavni, sa svega nekoliko različitih frekvenci u službi tona, pa sve do današnjih igara u kojima je glazba posebno prilagođena i komponirana od strane profesionalnih skladatelja i talentiranih umjetnika.

## 1.3. Razdoblje video igara s uključenim zvukom

### 1.3.1. Sedamdesete godine 20. stoljeća

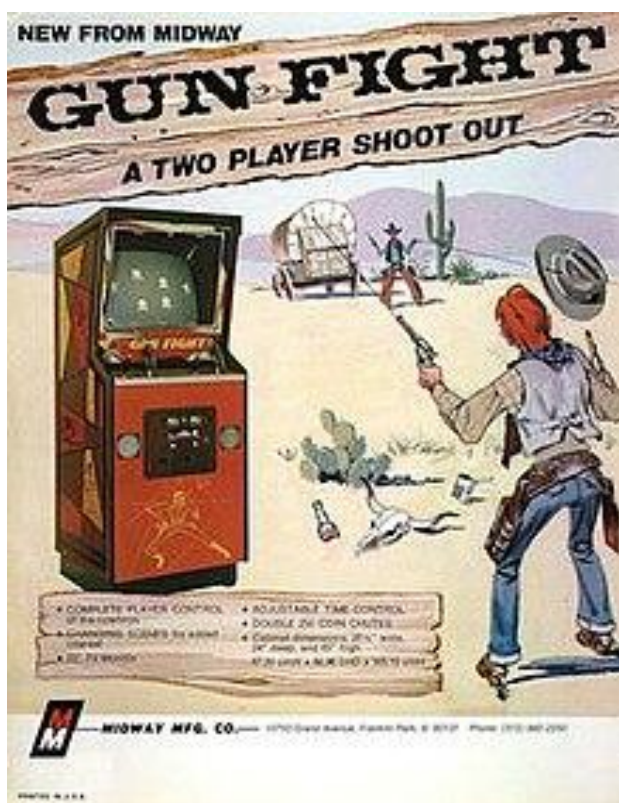
Ranih sedamdesetih godina dolazi do mnogih tehničkih otkrića poput kalkulatora (svojevremeno je koštao kao i polovan automobil) te se u tom razdoblju razvijaju prve video igre sa zvukom.

Prve arkadne, komercijalno dostupne video igre razvijene su na Atari računalima: Computer Space [10] 1971. i Pong [11] 1972. godine (Slika 6.). Naziv Pong dolazi od zvuka loptice koja se odbijala od zapreke te je upravo taj zvuk prvi zvučni efekt u svijetu video igara.



Slika 6. Atari, Pong 1972. prva komercijalna video igra

Godine 1975. izlazi prvi arkadna igra Gun Fight [12] u kojoj su korišteni zvučni efekti te kratku glazbu Funeral march [13] skladatelja Chopina (Slika 7.). Godinu dana nakon na tržište izlazi Breakout [14] unutar kojeg se pojavljuju različiti tonovi ovisno o odbijanju loptice.



Slika 7. Letak arkadne igre Gunfight, prvo pojavljivanje glazbe

Krajem sedamdesetih godina 20. stoljeća audio postaje interaktivniji te povezaniji s korisnikom. Prvi takav primjer je arkadna igra Space Invaders [15] programirana 1978. godine. Audio dizajneri uključuju dotad neiskorištena svojstva tona: promjena tempa glazbe ili zvučnih efekata, promjena tonaliteta te progresije koje su u funkciji stvaranja atmosfere koje motiviraju korisnika na još veći angažman.

### 1.3.2. Osamdesete godine 20. stoljeća

Osamdesete godine osim što donose naslove najpoznatijih arkadnih igara, donose i prvu upotrebu digitaliziranog ljudskog glasa – Vanguard [16] te pojavu glazbe cijelog benda unutar video igre – Journey Escape [17]. Prva najutjecajnija igra koja je postigla globalni uspjeh bila je Pac-Man [18]. Zbog svoje velike popularnosti Pac-man je podigao standarde video igara. Producenti su se pitali na koje sve načine mogu unaprijediti igru te je učiniti zanimljivijom. Upravo iz tog razloga dolazi do potrebe za zasebnim zvučnim procesorom: igra Gyruss [19] 1983. godine koristi pet različitih zvukovnih sekvenci Bachove Toccate i Fuge u d-molu kombiniranih sa zvučnim efektima video igre.

Rastom popularnosti arkadnih igara u javnim prostorima poput kafića i barova, dolazi do sve veće potražnje iskustva igranja video igara u udobnosti svoga doma. Tadašnja računala, namijenjena izričito za poslovne svrhe nisu imala zasebne zvučnike kakve danas postoje. Prvo takvo računalo sa integriranim zvučnikom bilo je IBM 5150 [20] istoimene tvrtke. Unutar računala nalazio se zvučnik koji je producirao jedan zvuk bez mogućnosti kontrole ili puštanja glazbe od strane korisnika. Zvučnik je služio za zvukove računalnog sustava: prilikom paljenja, u slučaju greške, itd. Snalažljivi programeri, mijenjajući frekvenciju tona dobili su melodiju koja je svojevremeno uključena u video igre. Prva takva igra je Alley Cat [21] napravljena 1984. godine od IBM-a.

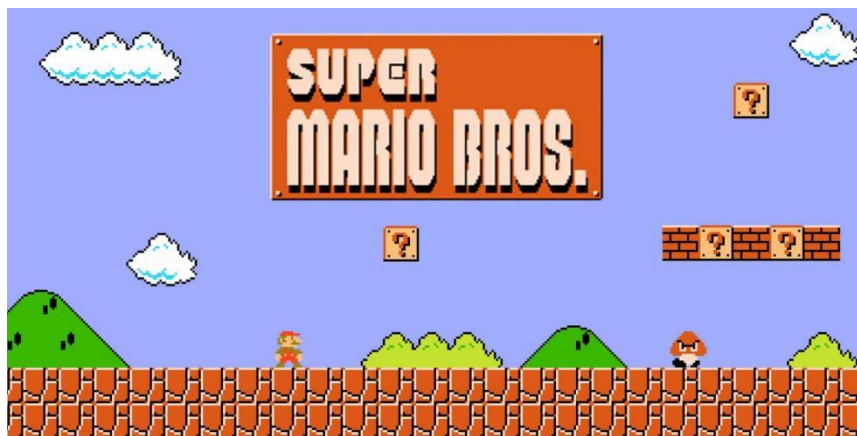
### 1.3.3. Integriranje zvučnih računalnih čipova

PSG (engl. Portable Sound Genetators) označava audio čipove čija je namjena procesiranje glazbe i zvukova na računalu. Jeftini čip AY-3-8910 koji se mogao programirati, također korišten u igri Gyruss, podržavao je izlaz tri kvadratno valnih tonova [22] (periodični oblik nesinusnog zvučnog vala u kojem amplituda alternira na stalnoj frekvenciji između fiksnih minimalnih i maksimalnih vrijednosti) te jedan kanal za šum. Prednost ovog čipa bila je cijena, pristupačnost za jednostavno programiranje te kvalitetan zvuk, korišten je od strane mnogih kompozitora čija je virtuoznost na taj način uključena u svijet video igara.

1982. Commodore 64 [23] je računalo sa integriranim SID (engl. Sound Interface Device) čipom koji ima mogućnost programiranja tri tona čiji su valni oblici programabilni te sadrži potpuni softverski "sintesajzer" (engl. Synthesizer). Sintesajzer je elektronički glazbeni instrument koji proizvodi audio signale na način da generira signale različitih valnih duljina uz pomoć subtraktivnih ili aditivnih sinteza [24]. Također je sadržavao programabilne analogne filtere čija je funkcija eliminacija svih signaa ispod zadane frekvencije.

1983. NES (engl. The Nintendo Entertainment System [25]) sadržavao je 5-kanalni čip:

- 2 promjenjiva kanala kvadratnih valova (engl. Pulse wave channels)
- 1 triangularni val (engl. Triangle wave): kanal koji je obično korišten za bas dionicu
- 1 kanal za bijeli šum (engl. White noise channel) korišten za bubnjeve, prekusiju ili ostale zvukove
- 1 kanal za ostale uzorke (engl. Sample channel) koji služi za snimanje manjih dionica.



Slika 8. Super Mario - zvukovi programirani na NES čipu

Tih pet kanala omogućuje programerima strukturiranje slojeva zvučnih zapisa kao pri produkciji glazbe, npr. možemo prikazati snimanje pjesme u studiju: svaki kanal ima svoju funkciju, tj. jedan kanal prima signal vokala dok drugi primaju signale raznih instrumenata. Spajanjem tih kanala dobiva se zvučna cjelina. Upravo na tom audio čipu, 1985. godine programirana je igra Super Mario [26] (Slika 8.).

Paralelno u svijetu glazbe sve je popularniji zvuk sintesajzera te su mnogi poznati glazbenici tog doba producirali zvuk na legendarnom Yamahinom DX7 sintesajzeru [27] (Slika 9.). To je prvi komercijalno uspješni digitalni sintesajzer pušten u prodaju 1983. godine. Zbog korištenja nove forme sinteze zvuka (modulacijom frekvencije), DX7 mogao je proizvesti mnogo nepredvidivih zvukova. Umjetnici tog vremena imali su puno veću paletu tonova kojima su mogli pristupiti jednostavnim izmjenama algoritama te korištenja raznih filtera nad zadanim instrumentom. Uređaji koji su podržavali FM čipove bili su sve više zastupljeniji. Marble Madness [28] je prva igra u kojoj se koristi FM sinteza 1984. godine.



Slika 9. Yamaha DX7 prospekt

Razdoblje korištenja audio samplera počinje od 1980. godine te traje i danas. Audio sampler omogućuje snimanje određenog zvuka te njegovo ponavljanje [29].



Audio tehnologija počinje se standardizirati krajem osamdesetih, početkom devedesetih godina 20. stoljeća pa se tako u kućna računala ugrađuju zvučne kartice koje sadržavaju set instrukcija tehničkog standarda MIDI (engl. Musical Instrument Digital Interface). Komunikacijski protokol određuje kretanje digitalnog signala koji se može kombinirati sa raznim instrumentima. Do tada nije postojala mogućnost komunikacije elektronskih instrumenata različitih proizvođača. Protokol funkcionira na način da zabilježi poruke i informacije o notama kojima možemo, uz pomoć sintesajzera, mijenjati zvuk.

Prva računalna FPS (engl. First Person Shooter) igra Doom [30] razvijena početkom devedesetih godina 20. stoljeća sadržavala je glazbu dobivenu korištenjem MIDI protokola.

Kako bi mogli zornije usporediti zvuk instrumenata koji je programiran FM protokolom [31] i novijim MIDI protokolom, preslušati ćemo dvije priložene .mp3 datoteke u ovom radu (FM\_Protokol.mp3 i MIDI\_Protokol.mp3).



FM\_Protokol.mp3



MIDI\_Protokol.mp3

MIDI tehnologija zbog svoje versatilnosti bila je odlična podloga za umjetnički izričaj programiranja glazbe no zbog potrebe za sve većom memorijom dolazi do ere CD-a (engl. Compact Disk).

Dinamičko programiranje glazbe susrećemo u samim počecima, smatra se da je taj smjer započeo s igrom Space Invaders. Početkom devedesetih godina 20. stoljeća dinamičko programiranje uz pomoć MIDI tehnologije dovedeno je na novu razinu. Tako su u računalnoj igrici Monkey Island 2 [32] 1991. godine prijelazi igrača između određenih prostorija popraćeni promjenom instrumenata koji sviraju u pozadini. Na primjer, u trenutku kada igrač prijeđe iz vanjskog svijeta u prostoriju, neprimjetnim prijelazom bubnjevi prestanu svirati te se pojavljuju novi instrumenti i drugačija melodija.

Na Nintendovoj konzoli u igri Banjo-Kazooie [33] očit je napredak dinamičkog programiranja na primjeru kada se igrač kreće ispod razine vode, zvukovi koje proizvodi su prigušeni te odaju dojam kao da se igrač uistinu tamo nalazi.

U posljednjih 25 godina interes za glazbom unutar igre raste velikom brzinom. Nefitna organizacija, MAGFest (engl. Music and Gaming Festival) koja promovira edukaciju ljudi o glazbi i povijesti video igara, započela je svoj rad 2002. godine okupljajući 300 sudionika dok je 2020. godine ta brojka iznosila preko 24.000 zaljubljenika u glazbu video igara [34]. Logo MAGFesta prikazan je na slici 10.



Slika 10. MAGFest - nefitna organizacija sa sjedištem u Washingtonu

Ponekad je zvuk dobiven iz samplera nedovoljno realističan u očima kompozitora pa se koriste snimljeni zvuci stvarnih instrumenata ili benda. Danas audio programeri koriste audio sampelere i snimku stvarnog zvuka.

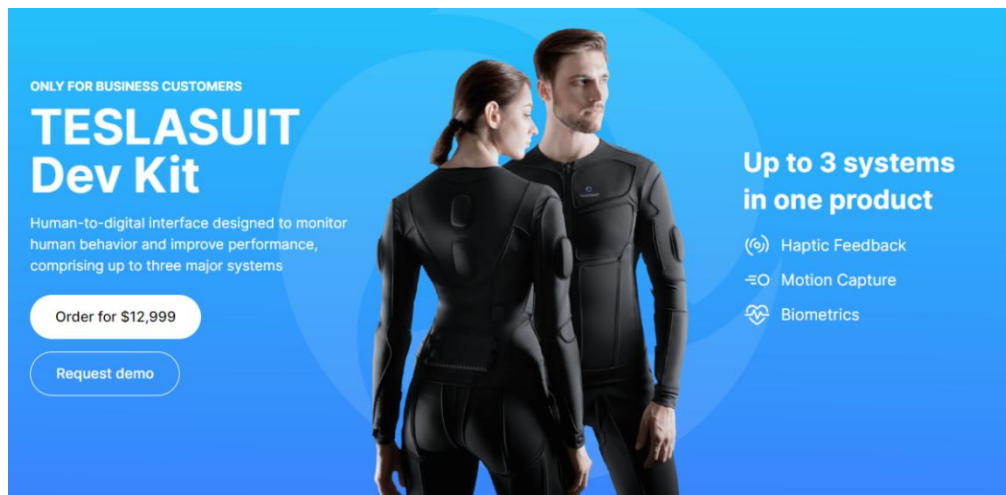
## 1.4. Pogled u budućnost: Uređaji koji poboljšavaju iskustvo igre

Napredak tehnoloških otkrića te sve veća potražnja za dodatnim stimulacijama osjetila igrača potiče razvoj sve većeg broja uređaja (engl. Gadget) koji se koriste prilikom igranja video igara. Svatko tko gleda filmove, sluša muziku, igra VR [35] (engl. Virtual Reality) ili igra računalne igre želi novo iskustvo u virtualnoj stvarnosti. Dosadašnje iskustvo virtualne stvarnosti gameri su doživjeli putem slušalica, VR naočala (Slika 11.) i pomagala koje su mogli koristiti pomoću ruku.



Slika 11. VR naočale

Danas su sve popularnija interaktivna odijela i majice koje se odjenu baš poput „obične” robe. Funkcioniraju na principu električnih impulsa koji se prenose na tijelo i na taj način vjerno simuliraju udarce. Neki od uspješnih primjera VR majica su Woojer vest edge [36] i TeslaSuit (Slika 12.) [37].



Slika 12. TeslaSuit

Ukoliko se igrač zaželi okušati u vožnji aviona ili letjeti poput ptice poznatim svjetskim gradovima unutar virtualne stvarnosti, odličan primjer VR uređaja koji simulira let je Birdly VR Simulator (Slika 13.): igrač legne na stroj te ruke stavi na sekciju oblikovanu poput krila i krene u virtualni let [38].



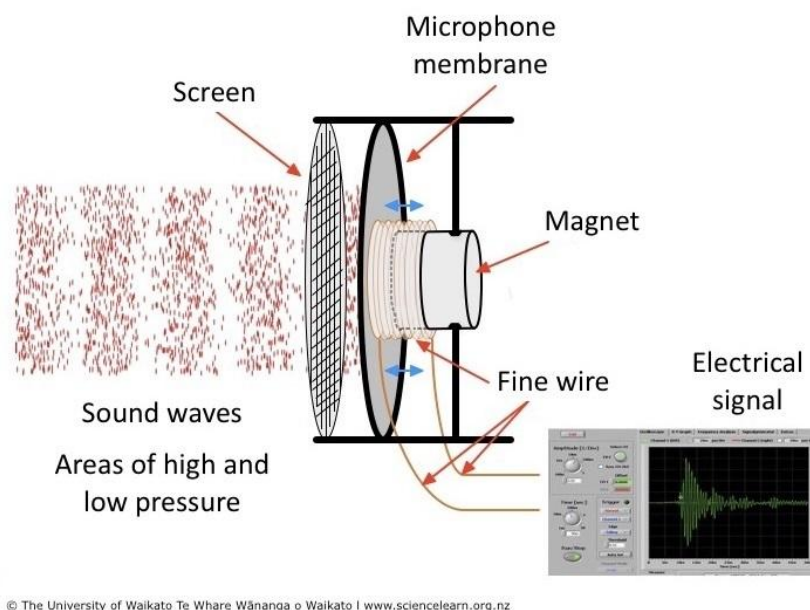
Slika 13. Birdly VR Simulator



## 2. Fizika zvuka

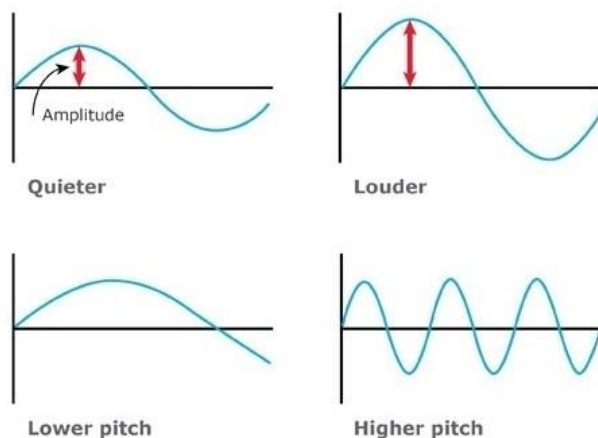
Digitalni zapis zvuka odnosi se na proces pretvorbe zvuka u njegov digitalni zapis. Iako su koncepti i teorija digitalizacije zvuka bili poznate svijetu početkom dvadesetih godina prošlog tisućljeća, složena tehnologija digitalizacije zvuka svoj vrhunac komercijalizacije započinje tehnološkim otkrićima sedamdesetih godina 20. stoljeća.

Kako bi približili osnovna svojstva zvuka moramo poznavati akustiku – grana koja proučava stvaranje, odašiljanje i primanje zvučnih valova. Sva tri svojstva stvaraju fizičke promjene u mediju u kojem se nalaze. Jedan od medija prijenosa zvuka je zrak, kada val putuje zrakom on mijenja njegov pritisak na način da pomiče čestice zraka (Slika 14.). Kako bi vizualno približili princip zvučnih valova, opisati ćemo prijenos energije valova u električni signal [39].



Slika 14. Prijenos zvučnih valova preko mikrofona u električni signal

Jedni od atributa zvučnih valova su glasnoća i frekvencija. O amplitudi ovisi glasnoća valova dok visina tona se mijenja ovisno o frekvenciji zvučnog vala kao što je prikazano na slici 15. [40]



© The University of Waikato Te Whare Wānanga o Waikato | www.sciencelearn.org.nz

Slika 15. Glasnoća i frekvencija zvučnog vala

### 3. Osnovna znanja potrebna za audio programiranje

U svijetu audio programera uobičajeni su timski zadaci međutim cijeli posao programiranja glazbe postaje dinamičniji te se može puno brže napredovati ako svaki član tima ima odgovarajuće znanje o teoriji glazbe i sposobnost produkcije na način da može imitirati cijeli bend ili orkestar. Uz osnovna predznanja o teoriji glazbe, bilo bi poželjno da audio programer zna svirati barem jedan instrument. Bez obzira o kojem se instrumentu radi, programer bi trebao biti dobro upoznat sa sintesajzerom te MIDI klavijaturom koja je jedna od osnovne tehničke opreme uz računalo. Također, dobar audio programer bi trebao poznavati psihologiju uspješnog poslovanja te aktivno pratiti svjetske glazbene trendove i snalaziti se u marketingu. [53]

#### 3.1. Glazbene vještine, kreativnost i komponiranje glazbe

Većina današnjih audio uradaka producirana je u vlastitom domu producenta ili u glazbenom studiju. Već spomenuta MIDI klavijatura ima pristup svim ostalim MIDI instrumentima i softverskim zvukovima dostupnih za komponiranje, iz tog razloga iznimno je korisna. Može se koristiti pri izolaciji ili promjeni glazbenih instrumenata, pri kontroli zvučnih modula, u korištenju samplera i elektroničkih bubnjeva te ostalih virtualnih instrumenata pa čak i za kontroliranje osnovnih radnji pri produkciji zvuka poput pokretanja i zaustavljanja snimanja.

Da bi programer što bolje koristio tehničku opremu, potrebna su znanja o osnovnim svojstvima zvuka, znanje čitanja i pisanja nota, razumijevanje kompleksnijih komponenti zvuka (način odbijanja, koherentnost zvuka<sup>1</sup>, sekvence,...) te visoka kreativnost pri komponiranju glazbe.

Audio programeri prije početka svoje karijere obično komponiraju vlastitu glazbu. Glazba unutar igre je dizajnirana kako bi dočarala specifični trenutak virtualnog svijeta u kojem se igrač nalazi. Svijet glazbe predstavlja mnoštvo različitih stilova. Ako se programer osjeća dovoljno snažnim producentom vezan za jedan stil glazbe, vrijeme je da krene učiti i komponirati glazbu drugih stilova. Igre u većini slučajeva nisu programirane unutar jednog stila te svaki stil ukomponiran u cjelinu igre služi specifičnoj namjeri koja rezultira u igračevom potpunom i jedinstvenom iskustvu.

<sup>1</sup> Koherentnost zvuka [41]: Zvuči li orkestar sa stotinu glazbenika kao jedna cjelina ili kao nepovezan skup instrumenata? Da li veliki koncertni klavir zvuči kao jedan instrument ili su mu tonovi naokolo razbacani i nepovezani? Koherentnost je oznaka i za zvučničku kutiju koja daje koherentan zvuk (tj. zvuk čija je skretnica dizajnirana tako kvalitetno da se ne čuju prelazi na rezovima, tj. zvuk koji je linearan i lišen koloracija).

## 3.2. Znanstveni pristup i inženjersko razmišljanje

Audio programer bi trebao dobro poznavati opremu i softver koji koristi (Slika 16.). Svo uloženo vrijeme prilikom proučavanja opreme uz tehnička predznanja rezultiraju poznavanje svih mana i vrlina opreme koju programer koristi. Uz poznavanje opreme, programer mora moći iskoristiti maksimum zvuka instrumenata kojeg producira uz pomoć EQ (ekvilajzera) i dodatnih efekata. Standardizacija procesa proizvodnje zvuka, npr. prilikom spajanja instrumenata na mikser je jednako bitna kao i fizička urednost koja pridonosi sistematičnosti.



Slika 16. Audio programeri su dobro upoznati s opremom kako bi se njihova kreativnost mogla nesmetano razvijati

## 3.3. Producentske vještine

U karijeri audio programera koji vrhunski vlada znanjem o više instrumenata ili cijelog benda, producentska vještina je jedna od najvažnijih.

Kod produkcije projekata drugih bendova, jednostavnije je ostati objektivan jer audio programer nije emocionalno vezan za glazbu koju producira kao što sam muzičar jest. Taj slučaj donosi programeru bolji fokus na sveukupnu sliku projekta te bolju motivaciju muzičara kako bi kreativno izrazili svoje emocije uz pomoć instrumenata. Obrnut slučaj bio bi kreiranje vlastitog aranžmana glazbene produkcije, tada je moguće da se objektivnost izgubi jer ponekad kreativna linija programera nije uvijek linearna i jasna [53].

S godinama iskustva audio programer može poboljšati ovu vještinu s ciljem uštede vremena prilikom vlastite produkcije zvuka.

Još jedna važna producerska vještina na kojoj audio programer mora raditi je znanje o tome kako će glazba ili zvučni efekti međusobno rezonirati u zvučnom prostoru. Na primjer, ako je odvojena naracija u istom trenutku kad i određena pjesma unutar scene puštena u produkciju, tada programer u završnoj obradi mora predvidjeti i osloboditi frekventni prostor u tom području kako bi pravilno zvučao voice-over (Slika 17.).

Identična pravila vrijede i za zvučne efekte: na primjer, ukoliko je zvučni efekt eksplozije bombe predviđen uz pozadinsku glazbu, tada audio programer prilagođava glazbu da im oduzima niže frekvencije poput basa.

Danas je dosta popularno produciranje zasebnih CD-ova glazbe iz igre. Da bi programer napravio set pjesama korištenih unutar igre, glazba mora biti posebno producirana bez efekata koji se pojavljuju u igri.



Slika 17. Posuđivanje glasa (voice-over) jednom ili više likova u igrici

### 3.4. Tehničko i računalno znanje

Vrlo je bitno neprestano učiti i razvijati nove tehnologije koje prate tehnički razvoj opreme za glazbenu produkciju. Razvoj donosi potrebu za upoznavanjem, održavanjem i otklanjanjem tehničkih poteškoća nastalih prilikom korištenja opreme audio programera. Osnovno tehničko znanje podrazumijeva detekciju neispravnog kabela te mogućnost njegovog lemljenja na licu mjesta, tj. biti u mogućnosti uspješno spojiti više kompliciranih MIDI postavki i omogućiti im da nesmetano rade, ili biti u stanju sinkronizirati različite dijelove opreme i integrirati ih u proces snimanja preko računala te rješavanje mnogih drugih tehničkih poteškoća. Prva riječ u računalnim igrama je računalno i podrazumijeva se da dobar audio programer izvrsno vlada njime i svim njegovim tehničkim i softverskim aspektima. Programi koji služe za sekvenciranje, programi za snimanje više kanalnih multimedijских zapisa, programi za uređivanje zvuka, virtualni instrumenti, dodaci za obradu efekata, softver prilagođen za čitanje i pisanje notnih zapisa, softver za pratnju, sempleri i slični programi glavni su oslonci dobrog studija za programiranje zvuka.



## 4. Google Resonance Audio

Jedni od audio rječnika koji omogućuju iznimno jaku i modernu tehnologiju web i mobilnim programerima su: Google Resonance Audio, The AudioWorklet. Google Resonance Audio je multi-platforman prostorni audio SDK (engl. Software Development Kit) (Slika 18.). SDK donosi više alata uz čiju pomoć možemo programirati aplikacije na raznim platformama operativnih sustava (iOS, Android,...).



Slika 18. Google Resonance Audio

Tehnologija programiranja prostornog zvuka predstavlja veliku važnost na području unaprjeđivanja realističnih iskustava korištenih u AR (Agumented Reality, proširena stvarnost) i VR (Virtual Reality, virtualna stvarnost) aplikacijama te video igricama. AR podrazumijeva projekciju vizualnog sadržaja uz pomoć kamere na mobitelu ili projektora dok je VR produkcija računalne simulacije stvarnog svijeta i uključuje dodatnu opremu (VR naočale, senzori, rukavice,...)[42]. Na slici 19. prikazana je jedna od poznatijih AR mobilnih igara.

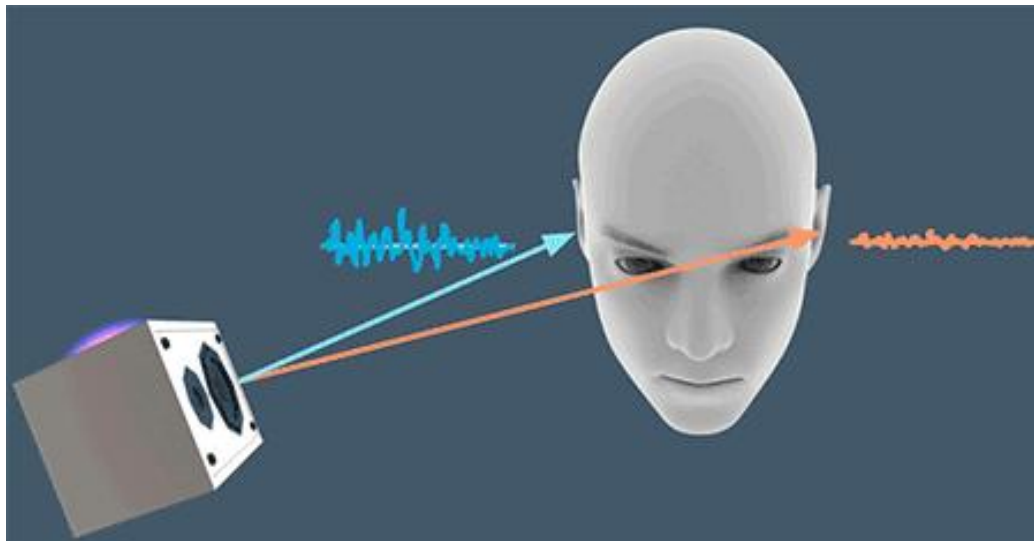


Slika 19. Pokemon GO jedna je od poznatijih AR mobilnih igara

Resonance Audio replicira kako bi se ponašali zvučni valovi u stvarnosti, fokus je na reprodukciji realističnog odbijanja zvučnih valova iz predmeta u prostoru prema ljudskom uhu. Odbijeni zvučni valovi u stvarnom prostoru govore nam gdje se nalaze predmeti koji nas okružuju te se replikacijom tih interakcija kreira iluzija zvukova koji dolaze iz specifičnih smjerova unutar virtualnog svijeta.

## 4.1. Određivanje horizontalne lokacije izvora zvuka

Razlika u vremenu, ITD (engl. Interaural Time Difference), koje je potrebno da dođe u lijevo i desno uho pomaže nam u određivanju horizontalnog položaja nisko frekventnih valova zvuka (Slika 20.). Što je izvor zvuka udaljeniji lijevo ili desno od uha, veća je vremenska razlika.

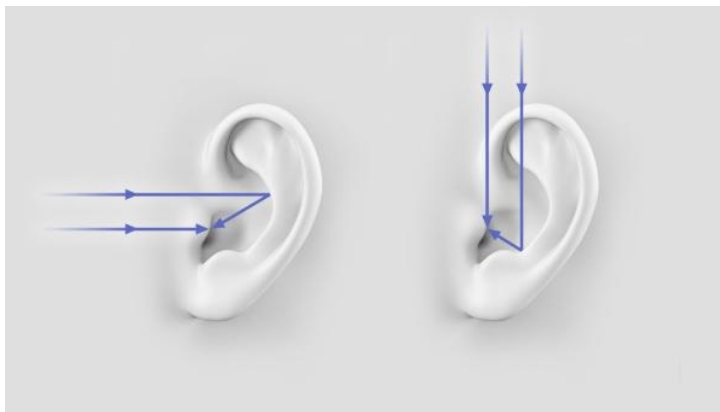


Slika 20. Razlika u vremenu dolaska niskih frekvencija zvuka, ITD

Za razliku od niskih frekvencija, ljudsko uho slušajući zvukove viših frekvencija ne može koristiti razliku u vremenu za određivanje pozicije izvora zvuka. U tom slučaju za određenje horizontalne pozicije koristi se razlike u distribuciji - ILD (engl. Interaural Level Differences). ILD predstavlja razlike glasnoće i frekvencijske distribucije između lijevog i desnog uha. Ako izvor zvuka, na primjer dolazi sa desne strane tijela, čovjek će čuti intenzivniji zvuk u desnom uhu a ne u lijevom zbog slabljenja zvučnih valova kako prolaze kroz glavu.

## 4.2. Određivanje vertikalne lokacije izvora zvuka

Dok nam vrijeme i razlika u intenzitetu zvuka pomažu locirati horizontalnu lokaciju zvuka, druge interakcije sa zvučnim valovima pomažu nam pozicionirati vertikalnu lokaciju. Zvukovi koji dolaze iz različitih smjerova odbijaju se unutar školjke uha na različite načine (Slika 21.). Čovjek koristi te promjene frekvencije (engl. Spectral Effects), nastale prilikom odbijanja za određivanje vertikalne lokacije zvučnog izvora.



Slika 21. Razlike u frekvenciji nastale različitim odbijanjem zvuka otkrivaju horizontalnu poziciju izvora zvuka

### 4.3. Simulacija zvučnih valova u interakciji sa njihovom okolinom

Uz simulaciju interakcije zvučnih valova sa ljudskim uhom, Resonance Audio simulira kretanje i interakciju zvučnih valova u odnosu na okolinu. Relativne promjene u lokaciji izvora zvuka čovjek percipira promjenom pozicije glave. Upravo na toj činjenici Resonance Audio reagira na pokret ljudske glave čuvajući vjerodostojnost lokacije izvora zvuka (Slika 22.). Lokacija je sačuvana uz pomoć senzora koji bilježe kretanje te se ta informacija koristi da bi rotirali zvučnu sferu u suprotnom položaju kretanje ljudske glave. Na taj smo način sačuvali relativnu poziciju predmeta koji projicira virtualni zvuk.

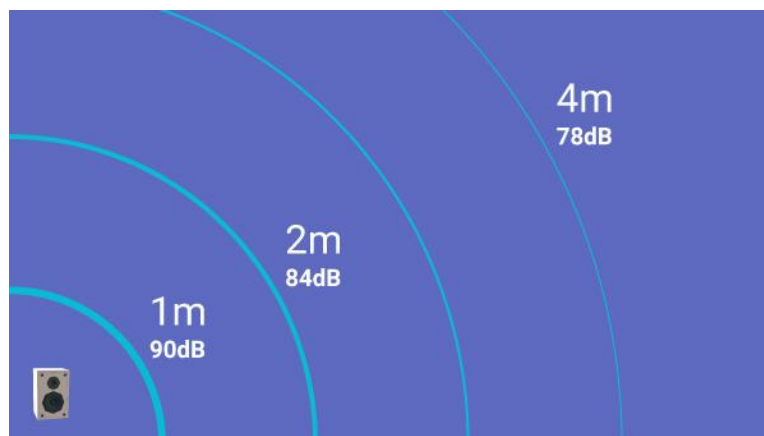


Slika 22. Resonance Audio reagira na promjenu pozicije glave

U stvarnom svijetu zvučni se valovi odbijaju od raznih predmeta na mnogo različitih načina. Takve kretnje kreiraju kompleksan spoj različitih refleksija zvučnih valova. Resonance Audio odvaja taj spoj valova na tri različite komponente: direktan zvuk, rana refleksija, kasni reverb. Kako bi pridodao realizmu, Resonance Audio može simulirati odbijanje zvučnih valova od različitih objekata koji se nalaze između izvora i ljudskog uha. U stvarnosti se više frekvencije više odbijaju nego niže frekvencije te upravo to svojstvo simulira Resonance Audio.

#### 4.3.1 Direktan zvuk

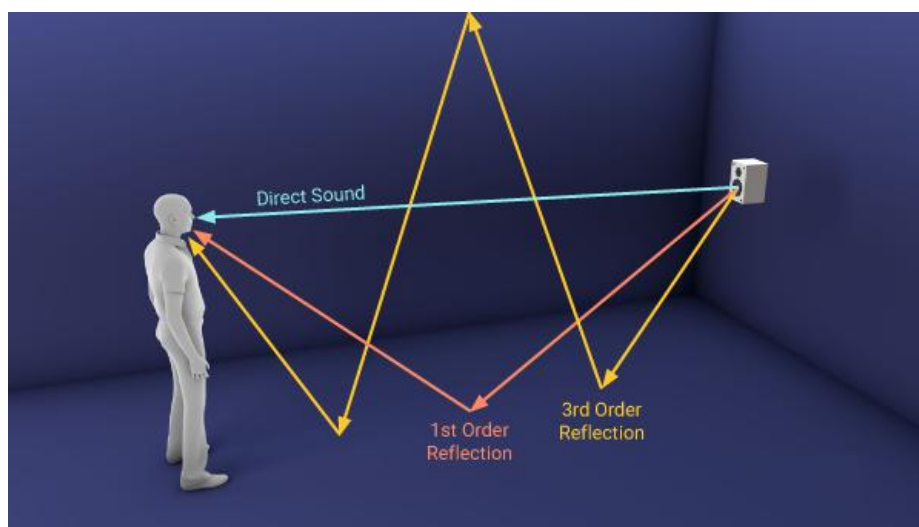
Prvi val koji dođe od izvora zvuka do ljudskog uha naziva se direktan zvuk. Kako se povećava udaljenost od izvora zvuka tako pada intenzitet zvuka, to bi značilo da što smo udaljeniji od izvora zvuka, slabije ćemo ga čuti (Slika 23.).



Slika 23. Utjecaj udaljenosti na intenzitet zvuka

### 4.3.2. Rana refleksija

Prvih par odbijenih zvučnih valova koji dođu do našeg uha nazivaju se rane refleksije zvuka. One djeluju na način da se odbijaju prostorijom te nam daju svojstvo veličine prostorije u kojoj se nalazimo. Resonance Audio bilježi rane refleksije u stvarnom vremenu te simulira izvore za svaku pojedinu refleksiju kao što je prikazano na slici 24.



Slika 24. Resonance Audio simulira odbijene zvukove – rane refleksije

### 4.3.3. Kasni reverb

Kako vrijeme protječe tako se gustoća dolaznih zvučnih valova gomila unutar ljudskog uha do trenutka u kojem se pojedini valovi prestanu razlikovati. Taj fenomen se naziva kasni reverb. Resonance Audio posjeduje ugrađeni reverb engine koji može povezati zvuk sa veličinom prostorije. Ukoliko promijenimo veličinu prostorije ili materijale koji se unutar nje nalaze, reverb engine djeluje tako da u stvarnom vremenu prilagodi zvučne valove novonastalim uvjetima.



## 5. Resonance Audio i Unity

Unity je računalni cross-platform game engine software, primarno dizajniran za stvaranje video igara. Osnovani su ga 2002. godine Nicholas Francis, Joachim Ante i David Helgason u glavnom gradu Danske, Kopenhagenu. U to vrijeme Unity game engine bio je prilagođen samo za izradu igara na platformi OS X. Dok su se ostali game engine software-i fokusirali na najpopularnije proizvođače video igara na konzolama i osobnim računalima, Unity je svoj dio tržišta pronašao na manje zahtjevnijim, jeftinijim, samostalnim produkcijskim kućama te individualnim projektima.

Kako je s vremenom tvrtka razvijala popularnost, došli su u poziciju surađivati s najpoznatijim proizvođačima video igara. Prva verzija platforme objavljena je 2005. godine. Dug je put od stvarne izrade do igranja igre te su se tijekom godina pojavili brojni alati kojima smo mogli razvijati igre. Danas je Unity jedan od najmoćnijih i najpristupačnijih modernih alata te je izvrstan za početnike jer olakšava učenje vještina programiranja 2D i 3D igara. Game engine podržava mnogo različitih programa i rječnika potrebnih za programiranje računalnih, mobilnih te VR i AR (2D ili 3D) igara. Uz razvoj igara, Unity koristi svoj engine u filmskoj, arhitekturnoj, automobilskoj te mnogim drugim industrijama.



Slika 25. Preko 50% mobilnih igrica na svijetu koristi Unity game engine [43]

Navesti ćemo nekoliko uspješnih igara izrađenih u Unity razvojnom okruženju: Fall Guys, razvijen od tvrtke Mediatonic, jedna od najpopularnijih igara za djecu u periodu između 2020. i 2021. godine [44].



Slika 26. Fall Guys – igra za djecu koja je iznimno ocijenjena za kaotičan gameplay i vizualan dojam

Američki game studio Innersloth je 2018. godine objavio online multiplayer igru Among Us. Igra je pokrenuta na više platformi te se može pokrenuti na mnogim uređajima i podržava multiplatformnu igrivost. Platforme koje podržavaju igricu: Microsoft Windows, PlayStation 4/5, Nintendo Switch, Xbox One, Xbox Series X/S, Android, iOS.



Slika 27. Among Us – hit koji je povezao mnoge igrače diljem svijeta

Također Unity vrlo uspješno koristi svoje sposobnosti u svijetu VR i AR stvarnosti. Jedna od najpoznatijih AR igara napravljenih uz pomoć Unity engine-a je Pokémon Go razvijen 2016. godine od tvrtke Niantic u kolaboraciji s Nintendo i Pokémon kompanijom.



Slika 28. Pokémon Go - proširena stvarnost, AR



## 5.1. Unity sučelje

Otvorimo li Unity možemo vidjeti prilično jednostavno sučelje. Početna alatna traka (A) (engl. Toolbar) nudi pristup vlastitom Unity korisničkom računu te Unity servisima u oblaku (engl. Cloud Services). Slijedi prozor hijerarhije (B) (engl. Hierarchy window) koji tekstualno reprezentira svaki objekt igre (engl. Game Object) [46] unutar scene (E, D) (engl. Scene) [47]. Koristeći ovaj prozor također možemo vizualno navigirati i uređivati scenu bilo 2D ili 3D, ovisno o projektu na kojem radimo.

Prozor scene prikazuje sve što je vizualno postavljeno unutar igrice. Kako bismo dodali datoteku unutar scene jednostavno ju iz prozora projekt (G) (engl. Project) odvučemo mišem unutar prozora scene te ta datoteka postaje Game Object. Prozor projekt je ujedno i najvažniji prozor u kojem se nalaze sve datoteke u projektu. One mogu biti grafike, razni objekti, fontovi, skripte, zvukovi, animacije i ostalo.

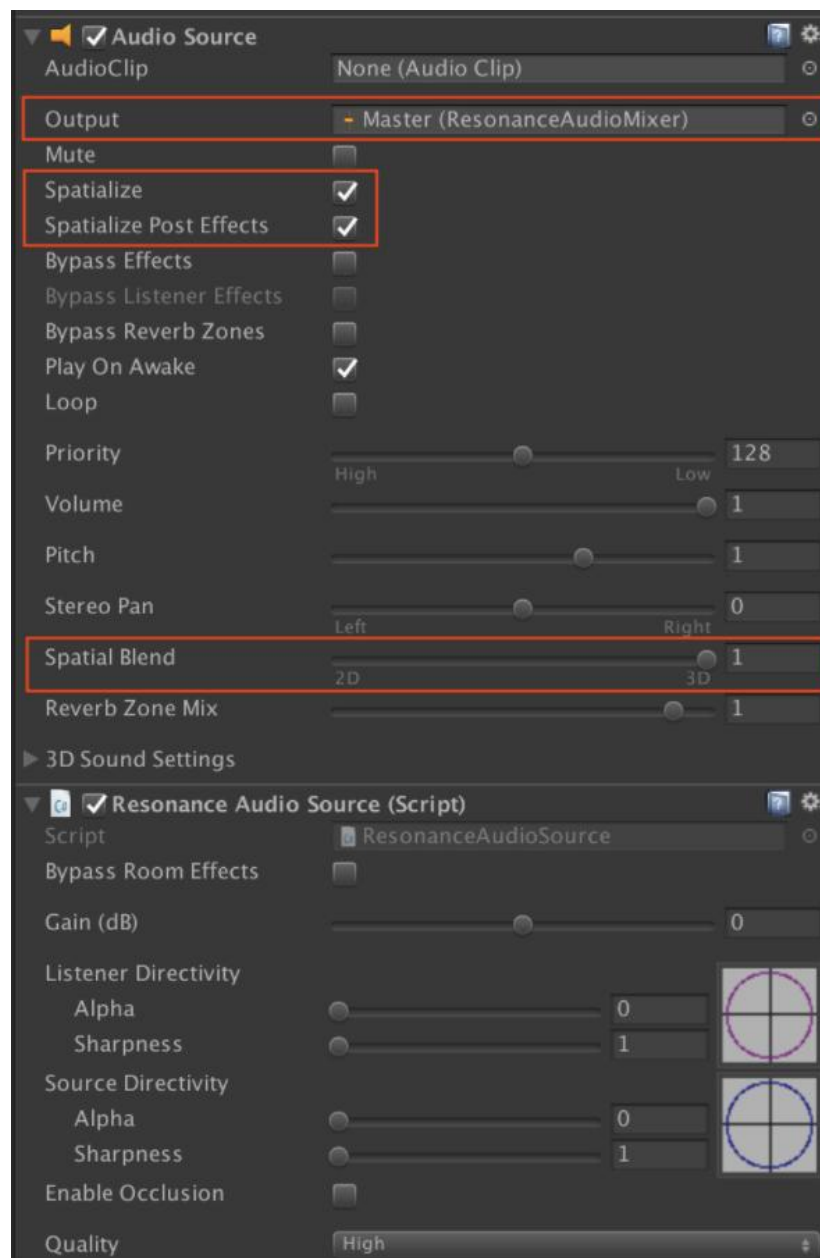
Prozor pregleda igrice (C) (engl. Game view) simulira završni prikaz (engl. Render) igrice. U trenutku kada kliknemo play tipku, simulacija započinje. Prozor inspekcije (F) (engl. Inspector window) prikazuje vlastitu knjižnicu odabranog asseta (engl. Asset) [48] koji su dostupni u trenutnom projektu. Nakon što smo importali (engl. Import) asete unutar projekta, oni se nalaze unutar prozora inspekcije. Statusna traka (H) (engl. Status bar) pokazuje razne obavijesti vezane za Unity procese te donosi brzi pristup raznim alatima i svojstvima.



Slika 29. Unity sučelje

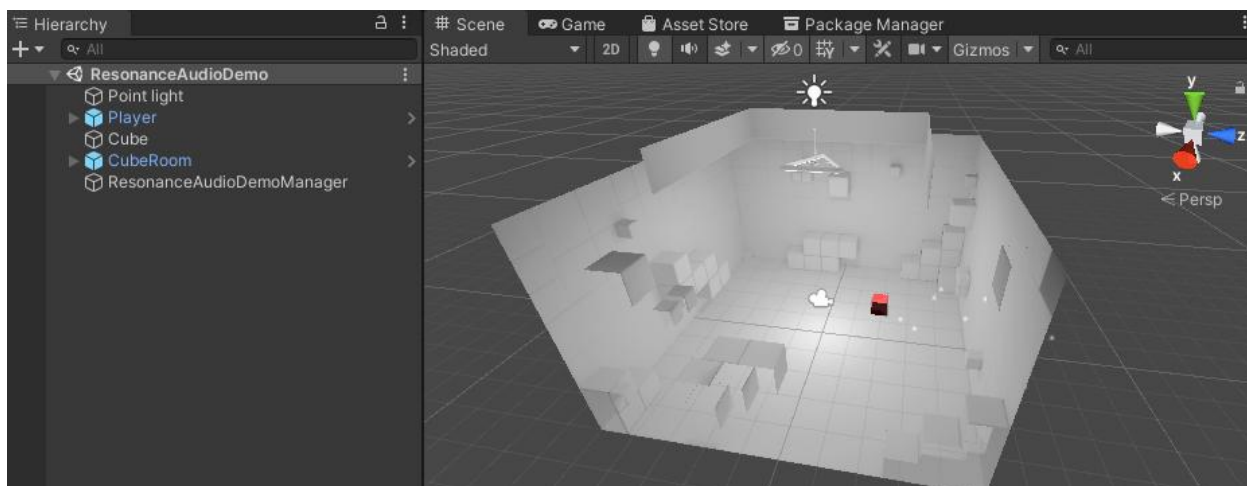
## 5.2. Postavljanje Resonance Audio komponenti

Nakon što je instaliran program Unity te posljednja verzija paketa Resonance Audio for Unity, dostupno na web stranici [49], kreiran je novi 3D projekt te je importan prethodno preuzeti paket Resonance Audio. Uz pomoć glavne trake izbornika odabrali smo Edit --> Project Settings --> Audio te namjestili Spatializer Plugin i Ambisonic Decoder na Resonance Audio. Unutar Main Camera --> Inspector odjeljka, dodali smo skriptu ResonanceAudioListener kako bi naš lik mogao prepoznati u kojoj se prostoriji nalazi. Kreiran je novi GameObjectSource kojem je pridružena skripta ResonanceAudioSource (Slika 30.). Automatski je dodijeljena komponenta Audio Source čiji smo output namjestili na Master polje ResonanceAudioMixer-a. Argument Spatial Blend unutar komponente Audio Source namješten je sa 2D na 3D varijablu. Da bi završili početno postavljanje audio postavki, odabrana je Spatialize opcija unutar Audio Source komponente.



Slika 30. Postavljanje Resonance Audio Source skripte

Iskorišten je već postojeći AudioClip pod nazivom ClapSound te je označen argument Loop kako bi se zvuk pljeskanja ponavljao. Nakon kreiranja i skaliranja novog objekta GameObjectRoom, kojem je dodijeljena skripta ResonanceAudioRoom, efekt reverba se mijenja u odnosu je li igrač u prostoriji ili izvan nje. Kako bismo bolje vizualizirali prostor oko igrača i audio svojstva koja smo postavili, unutar ResonanceAudio --> Demos --> Scenes možemo učitati Resonance Audio Demo scenu kao što je prikazano na slici 31.



Slika 31. Resonance Audio Demo scena

### 5.3. Vrijeme reverba

Vrijeme reverba (engl. Reverbation Time, RT) opisuje vrijeme potrebno zvuku unutar različitih frekvencijskih opsega koje je potrebno za smanjenje do 60 dB u akustičnom okruženju. Na primjer, ako je zvuku u prostoriji potrebno 10 sekundi do utišavanja sa 100 dB na 40 dB, vrijeme reverba je 10 sekundi. Zvuk se neprekidno odbija od raznih podloga unutar prostorije: zid, strop, prozori, stolovi, itd. Reverb se smanjuje u trenutku kada se zvučni valovi odbijaju od podloge koja može apsorbirati zvuk poput zavjesa, stolica pa čak i ljudi.

### 5.4. Programiranje reverba (engl. Reverb Baking)

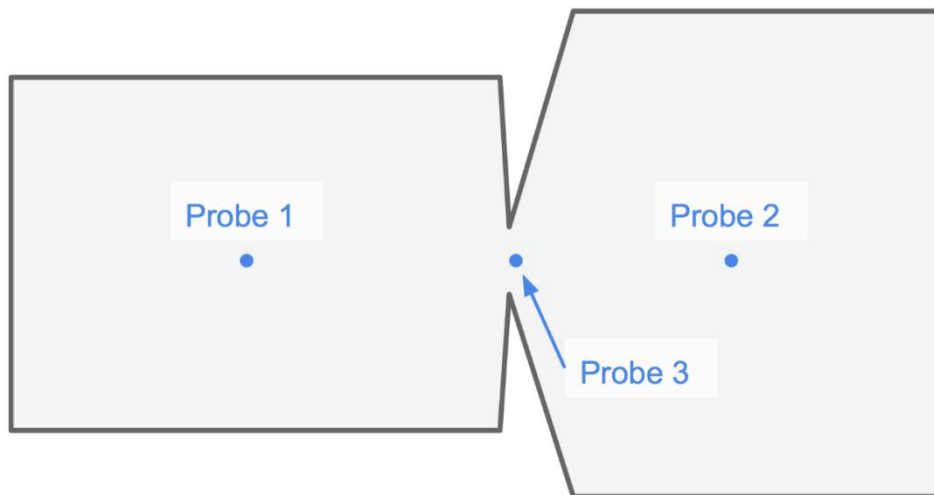
Reverb u glazbi je pojam kojim označavamo pojavu kada se javlja zvuk u prostoru, šaljući zvučne valove u svim smjerovima. Ti se valovi odbijaju od raznih objekata koji su sastavljeni od različitih materijala čija svojstva ne možemo zanemariti. Reverb je sličnog koncepta kao i efekt jeke osim što označava manje vrijeme refleksije potrebno za povratak zvučnih valova do izvora zvuka, najčešće u milisekundama. Manipulacija efektima reverba (elektronski producirani efekt jeke) Resonance Audio paket temelji na geometriji objekata u prostoru. Tehnički je zamišljeno kako se vrlo realističan reverb može omogućiti uz pomoć praćenja nevidljivih zraka koje se odbijaju od objekata u odnosu na statičnu geometriju postavljene scene. Geometry-based reverb, kao takvo softversko rješenje, odlično upotpunjuje Resonance Audio Room skriptu koja djeluje samo na kvadratnim oblicima prostorija u kojima se igrač nalazi.

### 5.4.1. Operacije sa reverb sondama (engl. Reverb Probes)

Još jedno od svojstva Resonance Audio Reverb Baking skripti je djelovanje na više materijala unutar iste prostorije, ne uključujući dodatne skripte, Unity je ograničen na jedan zvukovni materijal po prostoriji. Uz pomoć Resonance Audio Reverb Baking demo scene pokazali smo na koji se način zvuk odbija u različitim prostorijama. Ovako postavljena audio scena odgovara postavki vremena reverba na 60 sekundi, RT60.

Prva prostorija je crkva koja je sastavljena od tri različita dijela: ulaz, svetište i središnji prostor. Svaki od tih dijelova predstavljaju sonde koje se sastoje se od kugle ili kutije. U trenutku ulaska igrača u prostor određene sonde, primjenjuju se zasebna svojstva reverba pa je na taj način odbijanje zvuka različito u svetištu ili u sredini crkve. Kreiran je novi objekt igre kako bi pokazali koje sve mogućnosti nudi Resonance Audio Reverb Probe skripta. Svojstva koja se mogu mijenjati su oblik i veličina sonde, njezina vidljivost (utječe na odbijanje zvuka) i svojstva reverba poput vremena, količine pojačanja (gain) te visine frekvencije (brightness). Gain je izraz za količinu pojačanja primijenjenu na signal bilo kojim procesom koji povećava njegovu snagu. Mjeri se u decibelima, skraćeno dB.

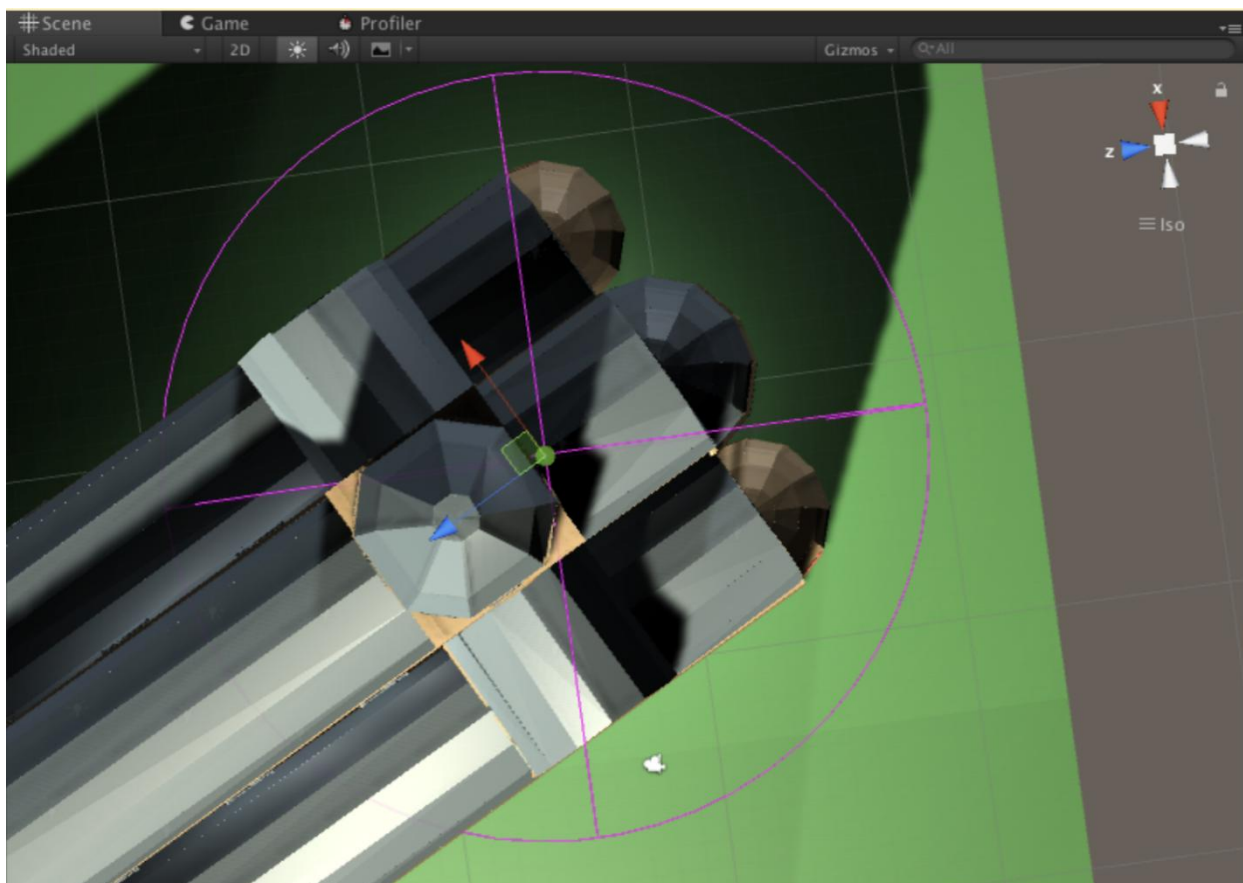
Koristeći se svojstvima akustike određujemo pozicije sonde unutar prostora te na prelasku iz jedne prostorije u drugu. Jasno je da svojstva reverba nisu jednaka na otvorenom prostoru ili u zatvorenoj prostoriji. Prilikom takvog prijelaza korisno je napraviti tri različite sonde: jedna za zvuk otvorenog prostora, jedna za zatvoreni prostor te jedna sonda za prijelazni prostor kao što je prikazano na slici 32. Općenito, što se više očekuje varijacija prostornog reverba, potrebno je više sonde.



Slika 32. Postavljanje reverb sonde vodeći se akustičnim svojstvima

## 5.4.2. Mapiranje vizualnih materijala sa sondama i akustičnim materijalima

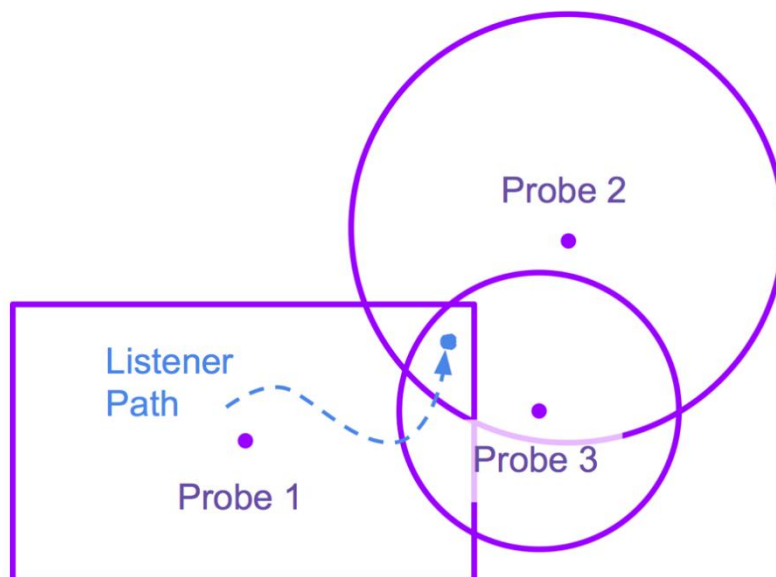
Svojstvo *Only When Visible* promatrane skripte označava ako je igrač u mogućnosti vidjeti objekt sonde, tada se primjenjuju efekti te sonde. Zbog sfernog ili kvadratnog oblika sonde često se dogodi da se prilikom dizajniranja i implementiranja unutar prostorija njihova veličina izlazi izvan veličine prostorije (Slika 33.). U tom se slučaju može dogoditi da se na igraču koji je izvan prostorije u kojoj je zamišljena sonda primjeni svojstvo reverba. Kako bi taj slučaj bio izbjegnuto koristi se svojstvo *Only When Visible* jer igrač ne može vidjeti kroz zidove prostorija iako se nalazi u prostoru sonde.



Slika 33. Moguća pozicija igrača: izvan prostorije, unutar sferne sonde

U našem slučaju ako se igrač nalazi s vanjske strane katedrale, objekt sonde može direktno vidjeti kroz prozor te bi se tada primijenila skripta reverba. Kada želimo izbjeći takav scenarij, postavljamo prozirne object collidere na otvore prozora.

Kako se igrač kreće kroz prostor ulazi u definirana polja sfernih ili kvadratnih sonde. Svaka ima svoja svojstva koja su u interakciji s igračem. Sljedeća slika prikazuje primjenjivanje svojstva sonde s obzirom na kretanju igrača: u čijem polju sonde se igrač nalazi, ta sonda djeluje svojim svojstvima na igrača. Na primjer: igrač se kreće iz prve kvadratne sonde u treću sfernu sondu i u trenutku kada pređe u njezin prostor tada se primjenjuju svojstva treće sonde (Slika 34.). Njegov put neka završi u drugoj sfernoj sondi: trenutak kada je prešao u njeno polje, tada djeluju svojstva druge sferne sonde.



Slika 34. Primjer kretnje igrača kroz kvadratne i sferne sonde

Nakon što smo odredili sonde u prostoru možemo primijeniti mapiranje vizualnih materijala (objekata) s akustičnim materijalima. Općenito je pravilo da objekti koji izgledaju slično imaju sličan akustični materijal povezan na njih. Ukoliko naš projekt zahtjeva drugačije, možemo odvojiti svaki vizualni materijal na više njih te nakon toga mapirati svakog sa svojim zasebnim akustičnim parom. Unutar Reverb Baking prozora, odaberemo object picker [50] te izaberemo ReverbBakingDemoMaterialMap.

Unutar inspector prozora dolazimo do prikaza s lijeve strane svih vizualnih materijala i terena koje su korišteni u projektu (Slika 35.). S desne strane mapiramo vizualni materijal s akustičnim. Kako bi provjerili mapirani akustični materijal unutar scene, možemo odabrati Visualize Mode u Reverb Baking prozoru. Za brisanje svih mapa koristimo opciju resetiraj sve (engl. Reset All) te se svi vizualni materijali mapiraju u prozirno (engl. Transparent) po početnim postavkama.





Slika 35. Mapiranje vizualnih objekata s akustičnim materijalima

Kako bi povezali svojstvo reverba sa sfernim i kvadratnim sondama otvaramo Reverb Probes meni unutar Reverb Baking prozora. Nakon što odaberemo sonde na kojima želimo da se primijeni svojstvo reverba, odaberemo Bake kako bi izvršili proces. Unutar console log prozora trebala bi se pojaviti poruka: "Reverb baking is completed successfully".

Na slici 36. prikazan je Inspector prozor koji detaljno opisuje jednu od sondi. Prikazuje se svojstvo "RT60s for frequency bands (sec)" gdje su raspoređene niske frekvencije s lijeva prema višima s desna. Možemo primijeniti svojstva Gain, Brightness i Time kako bi dodatno prilagodili svojstva reverba za svaku pojedinu sondu. Klikom na Play i krećući se unutar scene pratimo dodatno prilagođene postavke reverba unutar promijenjene sonde.

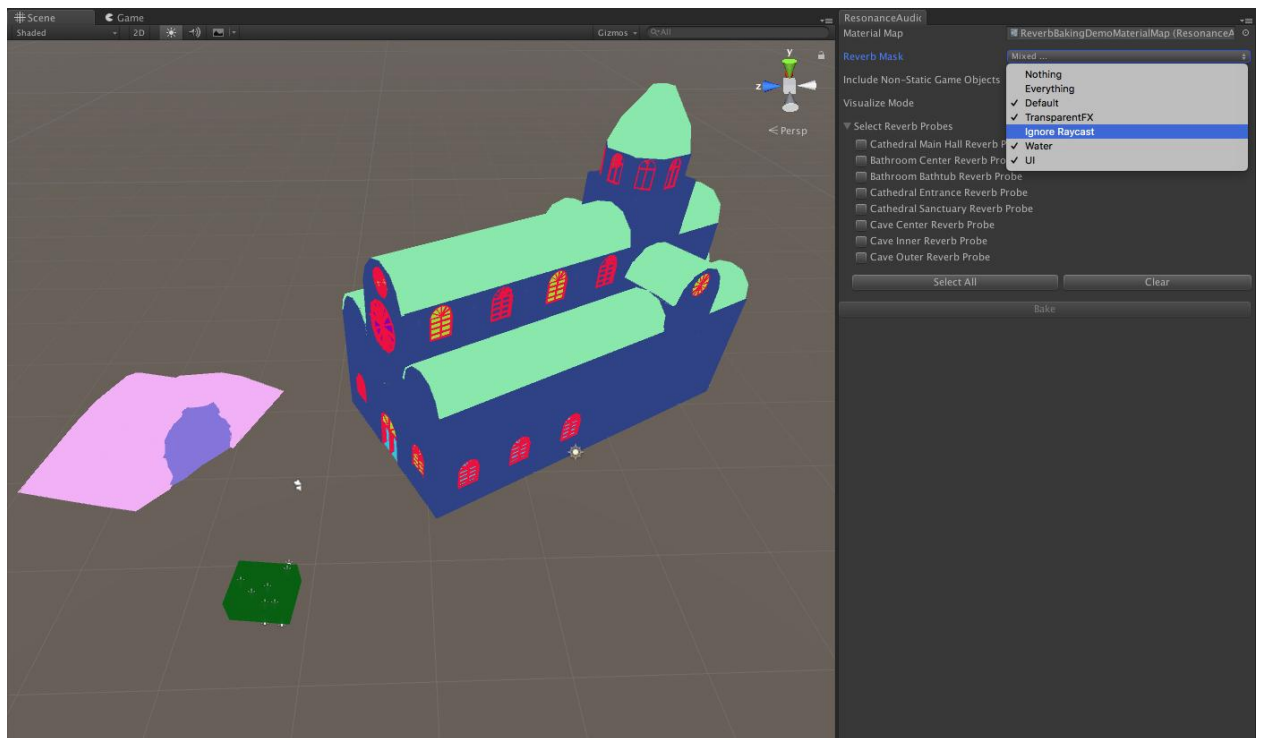


Slika 36. Prilagođavanje reverba unutar sonde

### 5.4.3. Uključivanje specifičnih objekata unutar kalkulacije reverba

U nekim slučajevima ne možemo uključiti svaku geometriju ili objekt u kalkulaciju reverba unutar scene. Na primjer, ako naša scena ima objekt dim, koji ima geometriju u specifičnom obliku, možda ne bismo željeli tu geometriju uključiti u izračun reverba jer ona ne stupa u interakciju sa zvukom. Kako bismo isključili objekte iz kalkulacije, možemo posegnuti dvjema metodama:

1. Isključivanje sloja (engl. Exclude a layer): prvo identificiramo postojeći sloj ili odaberemo novi kojeg ćemo isključiti. Na primjer, unutar naše ReverbBakingDemo scene Terrain game objekt je pridružen Ignore Raycast sloju kao što vidimo na slici 36. Odznačimo taj sloj unutar Reverb Mask svojstva u padajućem izborniku i na taj način ga isključimo iz kalkulacije kao što je prikazano na slici 37.

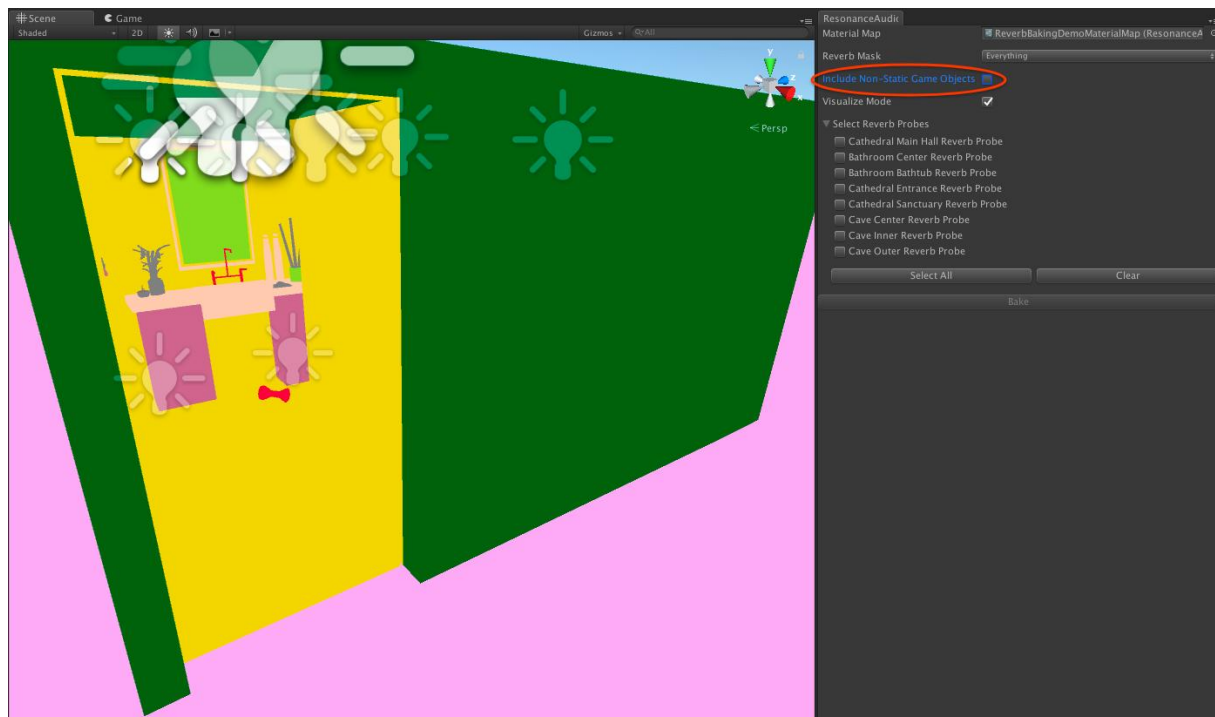


Slika 37. Isključivanje sloja iz kalkulacije reverba

Opcionalno možemo odabrati Everything unutar padajućeg izbornika kako bi uključili sve slojeve u kalkulaciju. Uključeni objekti reflektiraju svojstvo zvuka svojstveno prema mapiranim akustičnim materijalima.

2. Uključivanje/isključivanje nestatičnih objekata igre (engl. Toggle non-static game objects): reverb je unaprijed izračunat te se ne može mijenjati u stvarnom vremenu. Iz tog razloga, možda ne bismo htjeli uključiti u kalkulaciju nestatične objekte poput igrača koji se kreće scenom.

Unutar scene isključimo opciju Include Non-Static Game Objects kako bismo isključili nestatične objekte u kalkulaciju reverba (Slika 38.). Na primjer, unutar demo scene, objekt Bathroom\_door nije statičan te ako je opcija Include Non-Static Game Objects isključena, objekt se ne prikazuje unutar Visualize Mode opcije.

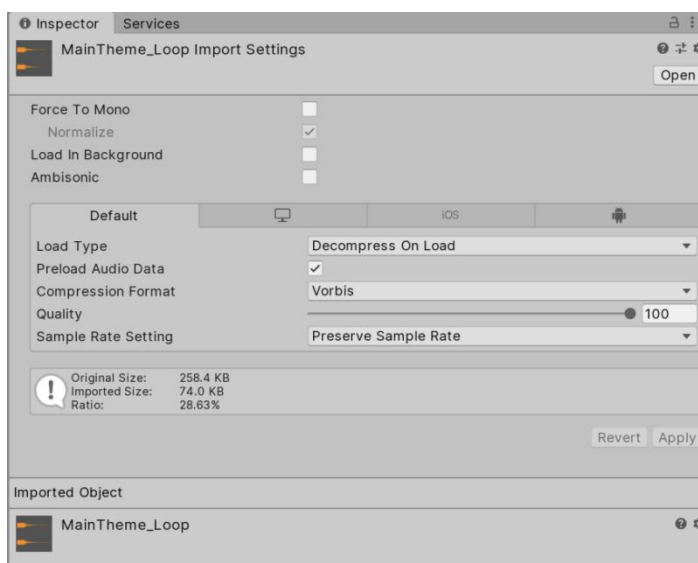


Slika 38. Uključivanje/isključivanje nestatičnih objekata igre

## 5.5. Dodavanje audio komponenti

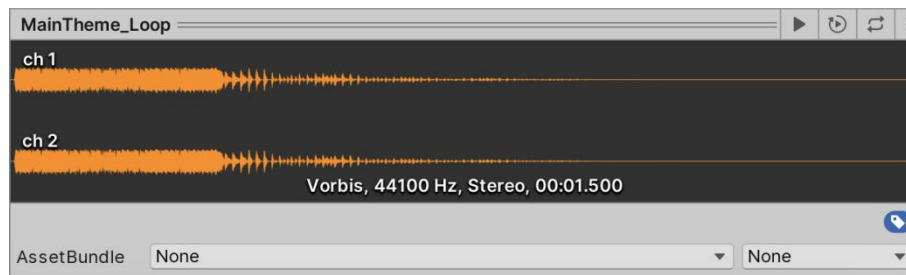
Unity podržava audio datoteke .aiff, .wav, .mp3 i .ogg formata. [54]. Zvukovi korišteni unutar Rada preuzeti su sa web stranice freesound.org.

Nakon što se datoteka zvuka pozadine smjesti unutar projektnog stabla, unutar inspector ploče (engl. Inspector panel) mogu se promijeniti početne postavke ukoliko je potrebno. Na primjer, ako audio datoteka ima dvije odvojene niti za stereo izlaz zvuka, mogu se odabrati opcija Force to Mono u slučaju da odvajanje kanala nije potrebno (Slika 39.).



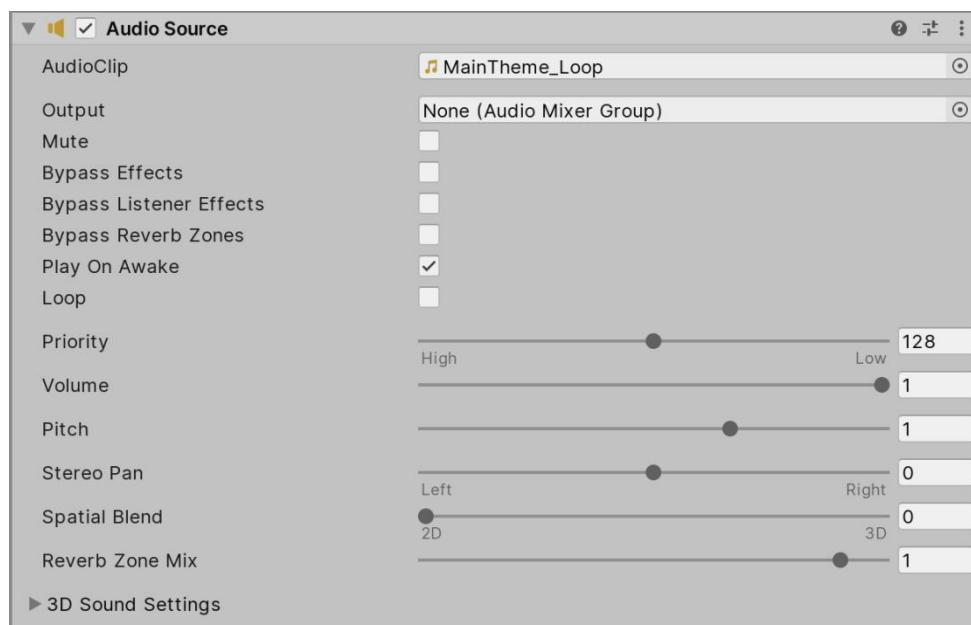
Slika 39. Inspector ploča zvuka pozadine

Na dnu inspector ploče nalazi se prikaz učitane audio datoteke u formi vala ili valova (u ovisnosti o broju kanala). Datoteka se može preslušati pritiskom na gumb play (Slika 40.).



Slika 40. Prikaz audio datoteke u valnom formatu

Sljedeće se dodaje Audio Source komponenta koja pokreće audio datoteku unutar scene. Kako bismo pridružili Audio Source komponentu nekom game objektu unutar scene moramo odabrati GameObject Menu -> Audio -> Audio Source. Kako bi bila pokrenuta, komponenti moramo pridružiti audio datoteku. Komponenta Audio Source odgovorna je za pokretanje i zaustavljanje audio datoteke te za modifikacije ostalih svojstava iste kao što je prikazano na slici 41.



Slika 41. Parametri koje možemo promijeniti unutar Audio Source komponente

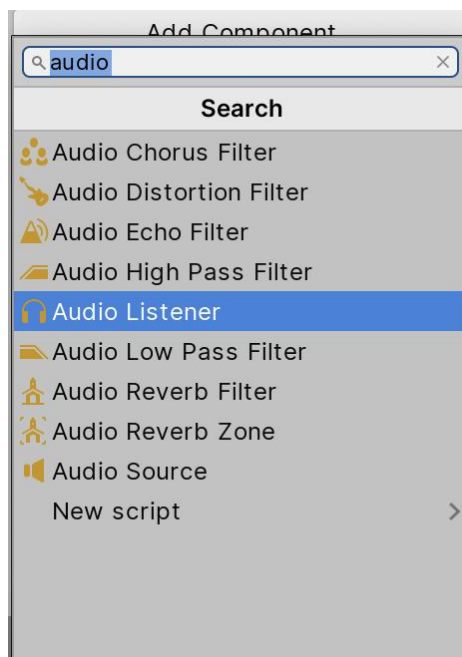
Svojstvo Play On Awake pridruženo je automatski, to znači da će se odabrana audio datoteka pokrenuti u isto vrijeme kada pokrenemo scenu. Ukoliko želimo da se audio datoteka ponavlja neprestano u krug, odabrati ćemo svojstvo Loop. To svojstvo je korisno za razne događaje, na primjer kada želimo integrirati zvuk hodanja - koraci će se čuti sve dokle traje akcija hodanja.

U stvarnom životu zvuk se prenosi uz pomoć vibracija koje djeluju na objekt. Te vibracije dolaze do slušatelja. On može otprilike predvidjeti iz kojeg smjera dolazi zvuk te također može odrediti udaljenost izvora s obzirom na glasnoću i kvalitetu zvuka.

Objekti unutar scene koji se brzo kreću mijenjaju visinu tona, poput bombe koja pada ili policijskog automobila. Također, okruženje u kojem se igrač nalazi utječu na refleksiju (odbijanje) zvuka, na primjer zvuku proizvedenom unutar špilje biti će dodijeljeno svojstvo jeke (engl. Echo) ali istom tom zvuku na otvorenom prostoru nećemo dodijeliti svojstvo jeke.

Kako bismo mogli koristiti audio datoteku unutar scene moramo je povezati sa objektom igre. Objekt igre s povezanim zvukom zovemo audio izvorom - možemo ga zamisliti u ulozi zvučnika. Njegov zvuk "osluškuje" Audio Listener komponenta koja je prema početnim postavkama dodijeljena objektu glavne kamere. Unutar scene može biti samo jedna komponenta Audio Listener pridružena objektu.

Nakon što je kreiran objekt igre, unutar inspektor ploče odaberemo Add Component svojstvo. Zatim odaberemo Audio -> Audio Listener kako bi pridružili Audio Listener komponentu odabranom objektu igre (Slika 42.).



Slika 42. Dodavanje Audio Listener komponente

Audio Listener komponenta ne sadrži nikakva svojstva (Slika 43.), uključena je u glavnu kameru prema početnim postavkama. Ona također može biti pridružena objektu igre koji predstavlja igrača umjesto kamere. Na taj način zvuk će biti pokrenut s obzirom na igrača (umjesto na kameru).



Slika 43. Audio Listener komponenta

Možemo zaključiti da uz par jednostavnih koraka dodajemo audio datoteku u projekt te određujemo izvore koji reproduciraju zvuk unutar scene i određujemo one objekte koji će naposljetku taj zvuk i slušati. Koristeći zvukovne datoteke približavamo virtualnu stvarnost realnosti što uzrokuje povećanje interaktivnosti između igrača i računala.

## 6. Zaključak

Nakon što je prikazan povijesni razvoj zvuka u računalnim igricama od vremena prije računala do danas, opisuju se osnovni pojmovi fizike zvuka. Da bi bolje razumjeli svojstva i pojam audio programiranja, središnji dio rada detaljnije opisuje znanja i vještine koje bi trebao imati dobar audio programer. Slijedi objašnjenje sučelja Unity razvojnog alata. Kako bi fokus bio na programiranju zvuka unutar Unity razvojnog okruženja uz pomoć Google Resonance Audio workleta, vizualna grafika i objekti korišteni u radu preuzeti su sa interneta. Taj dio rada opisuje i neka važna svojstva Google Resonance Audio workleta koja su primjenjena na konkretnom primjeru. Naposljetku je prikazano kako se postavljaju audio komponente u računalnoj igri u Unity alatu.

Cilj rada je potaknuti čitateljevo zanimanje za danas izuzetno traženom te specifičnom granom programiranja: spoj umjetnosti (glazbe) s računalnim znanjem.

## 7. Literatura

- [1] [Mrežno]. Available: <https://gamingkk.com/sound-affect-gaming-performance/>. [Pokušaj pristupa lipanj 2023].
- [2] [Mrežno]. Available: <https://www.bluelabellabs.com/blog/do-users-play-better-with-music-sound-fx/>. [Pokušaj pristupa lipanj 2023].
- [3] [Mrežno]. Available: [https://en.wikipedia.org/wiki/The\\_Horse\\_in\\_Motion](https://en.wikipedia.org/wiki/The_Horse_in_Motion). [Pokušaj pristupa lipanj 2023].
- [4] [Mrežno]. Available: [https://en.wikipedia.org/wiki/The\\_Dickson\\_Experimental\\_Sound\\_Film](https://en.wikipedia.org/wiki/The_Dickson_Experimental_Sound_Film). [Pokušaj pristupa lipanj 2023].
- [5] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Flowers\\_and\\_Trees](https://en.wikipedia.org/wiki/Flowers_and_Trees). [Pokušaj pristupa lipanj 2023].
- [6] [Mrežno]. Available: <https://www.historyofinformation.com/detail.php?entryid=3573>. [Pokušaj pristupa lipanj 2023].
- [7] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Tennis\\_for\\_Two](https://en.wikipedia.org/wiki/Tennis_for_Two). [Pokušaj pristupa lipanj 2023].
- [8] [Mrežno]. Available: <https://enciklopedija.hr/natuknica.aspx?ID=45650>. [Pokušaj pristupa lipanj 2023].
- [9] [Mrežno]. Available: [https://americanhistory.si.edu/collections/search/object/nmah\\_1302004](https://americanhistory.si.edu/collections/search/object/nmah_1302004). [Pokušaj pristupa lipanj 2023].
- [10] [Mrežno]. Available: <https://thereader.mitpress.mit.edu/before-pong-there-was-computer-space/>. [Pokušaj pristupa lipanj 2023].
- [11] [Mrežno]. Available: <https://www.ponggame.org/>. [Pokušaj pristupa lipanj 2023].
- [12] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Gun\\_Fight](https://en.wikipedia.org/wiki/Gun_Fight). [Pokušaj pristupa lipanj 2023].
- [13] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Funeral\\_march](https://en.wikipedia.org/wiki/Funeral_march). [Pokušaj pristupa lipanj 2023].
- [14] [Mrežno]. Available: <https://elgoog.im/breakout/>. [Pokušaj pristupa lipanj 2023].
- [15] [Mrežno]. Available: <https://freeinvaders.org/>. [Pokušaj pristupa lipanj 2023].
- [16] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Vanguard\\_\(video\\_game\)](https://en.wikipedia.org/wiki/Vanguard_(video_game)). [Pokušaj pristupa lipanj 2023].
- [17] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Journey\\_Escape](https://en.wikipedia.org/wiki/Journey_Escape). [Pokušaj pristupa lipanj 2023].
- [18] [Mrežno]. Available: <https://www.thoughtco.com/pac-man-game-1779412>. [Pokušaj pristupa lipanj 2023].
- [19] [Mrežno]. Available: <https://arcadeologia.es/en/machines/gyruss-25.html>. [Pokušaj pristupa lipanj 2023].



- [20] [Mrežno]. Available: [https://www.ibm.com/ibm/history/exhibits/pc25/pc25\\_intro.html](https://www.ibm.com/ibm/history/exhibits/pc25/pc25_intro.html). [Pokušaj pristupa lipanj 2023].
- [21] [Mrežno]. Available: <http://www.hardcoregaming101.net/alley-cat/>. [Pokušaj pristupa lipanj 2023].
- [22] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Square\\_wave](https://en.wikipedia.org/wiki/Square_wave). [Pokušaj pristupa lipanj 2023].
- [23] [Mrežno]. Available: [https://americanhistory.si.edu/collections/search/object/nmah\\_334636](https://americanhistory.si.edu/collections/search/object/nmah_334636). [Pokušaj pristupa lipanj 2023].
- [24] [Mrežno]. Available: [https://en.wikipedia.org/wiki/Additive\\_synthesis](https://en.wikipedia.org/wiki/Additive_synthesis). [Pokušaj pristupa lipanj 2023].
- [25] [Mrežno]. Available: <https://www.britannica.com/technology/Nintendo-console>. [Pokušaj pristupa lipanj 2023].
- [26] [Mrežno]. Available: [https://www.youtube.com/watch?v=9bB7r0M9kg&ab\\_channel=iacobelli7](https://www.youtube.com/watch?v=9bB7r0M9kg&ab_channel=iacobelli7). [Pokušaj pristupa lipanj 2023].
- [27] [Mrežno]. Available: <https://retrosynthads.blogspot.com/search?q=dx7>. [Pokušaj pristupa lipanj 2023].
- [28] [Mrežno]. Available: <https://www.arcade-history.com/?n=marble-madness&page=detail&id=1559>. [Pokušaj pristupa lipanj 2023].
- [29] [Mrežno]. Available: [What Is A Sampler?](#). [Pokušaj pristupa lipanj 2023].
- [30] [Mrežno]. Available: <https://www.britannica.com/topic/Doom>. [Pokušaj pristupa lipanj 2023].
- [31] [Mrežno]. Available: [\[SoundBattle\] Doom: Sound Blaster vs UltraSound vs Sound Canvas](#). [Pokušaj pristupa lipanj 2023].
- [32] [Mrežno]. Available: <https://www.superjumpmagazine.com/the-history-of-monkey-island/>. [Pokušaj pristupa lipanj 2023].
- [33] [Mrežno]. Available: <https://www.inverse.com/gaming/banjo-kazooie-rare-oral-history>. [Pokušaj pristupa lipanj 2023].
- [34] [Mrežno]. Available: <https://www.magfest.org/>. [Pokušaj pristupa lipanj 2023].
- [35] [Mrežno]. Available: <https://virtualspeech.com/blog/history-of-vr>. [Pokušaj pristupa lipanj 2023].
- [36] [Mrežno]. Available: <https://history-computer.com/the-5-best-haptic-vests-for-gaming-today/>. [Pokušaj pristupa lipanj 2023].
- [37] [Mrežno]. Available: <https://teslasuit.io/>. [Pokušaj pristupa lipanj 2023].
- [38] [Mrežno]. Available: <https://aixr.org/insights/human-flight-virtual-reality>. [Pokušaj pristupa lipanj 2023].

- [39] [Mrežno]. Available: <https://www.sciencelearn.org.nz/resources/2814-sound-visualising-sound-waves>. [Pokušaj pristupa lipanj 2023].
- [40] [Mrežno]. Available: <https://www.sciencelearn.org.nz/resources/2814-sound-visualising-sound-waves>. [Pokušaj pristupa lipanj 2023].
- [41] [Mrežno]. Available: [http://www.audiopuls.hr/clanci/audio\\_rjecnik/02.html](http://www.audiopuls.hr/clanci/audio_rjecnik/02.html). [Pokušaj pristupa lipanj 2023].
- [42] [Mrežno]. Available: <https://sopa.tulane.edu/blog/whats-difference-between-ar-and-vr>. [Pokušaj pristupa lipanj 2023].
- [43] [Mrežno]. Available: <https://unity.com/>. [Pokušaj pristupa lipanj 2023].
- [44] [Mrežno]. Available: <https://www.create-learn.us/blog/top-games-made-with-unity/>. [Pokušaj pristupa lipanj 2023].
- [45] [Mrežno]. Available: <https://docs.unity3d.com/Manual/CreatingScenes.html>. [Pokušaj pristupa lipanj 2023].
- [46] [Mrežno]. Available: <https://docs.unity3d.com/Manual/class-GameObject.html>. [Pokušaj pristupa lipanj 2023].
- [47] [Mrežno]. Available: <https://docs.unity3d.com/Manual/CreatingScenes.html>. [Pokušaj pristupa lipanj 2023].
- [48] [Mrežno]. Available: <https://docs.unity3d.com/2021.3/Documentation/Manual/AssetWorkflow.html>. [Pokušaj pristupa lipanj 2023].
- [49] [Mrežno]. Available: <https://github.com/resonance-audio/resonance-audio-unity-sdk/releases>. [Pokušaj pristupa lipanj 2023].
- [50] [Mrežno]. Available: <https://docs.unity3d.com/2021.3/Documentation/Manual/search-advanced-object-picker.html>. [Pokušaj pristupa lipanj 2023].
- [51] Hocking, Joseph (2015.) *Unity in action*
- [52] Gibson Bond, Jeremy (2022.) *Introduction to Game Design, Prototyping and Development: From Concept to Playable Game with Unity and C#*
- [53] Aaron Marks (2017.) *Aaron Mark's Complete Guide to Game Audio for Composers, Sound Designers, Musicians and Game Developers*
- [54] [Mrežno]. Available: <https://learn.unity.com/tutorial/working-with-audio-components-2019-3#5f8fa275edbc2a284332bcea>. [Pokušaj pristupa lipanj 2023].

## 8. Popis slika

Slika 1. Sallie Gardner: Gallop - slika u pokretu .....	2
Slika 2. Scena iz filma The Dickson Experimental Sound Film .....	2
Slika 3. Prva interaktivna elektronska igra CRTAD, Thomas T. Goldsmith, Estle Ray Mann .....	3
Slika 4. Prikaz igre Tennis for Two na osciloskopu .....	3
Slika 5. Prva video konzola, Magnavox Odyssey .....	4
Slika 6. Atari, Pong 1972. prva komercijalna video igra .....	5
Slika 7. Letak arkadne igre Gunfight, prvo pojavljivanje glazbe .....	5
Slika 8. Super Mario - zvukovi programirani na NES čipu .....	7
Slika 9. Yamaha DX7 prospekt .....	7
Slika 10. MAGFest - neprofitna organizacija sa sjedištem u Washingtonu .....	8
Slika 11. VR naočale .....	9
Slika 12. TeslaSuit .....	10
Slika 13. Birdly VR Simulator .....	10
Slika 14. Prijenos zvučnih valova preko mikrofona u električni signal .....	11
Slika 15. Glasnoća i frekvencija zvučnog vala .....	11
Slika 16. Audio programeri su dobro upoznati s opremom kako bi se njihova kreativnost mogla nesmetano razvijati .....	13
Slika 17. Posuđivanje glasa (voice-over) jednom ili više likova u igrici .....	14
Slika 18. Google Resonance Audio .....	15
Slika 19. Pokemon GO jedna je od poznatijih AR mobilnih igara .....	15
Slika 20. Razlika u vremenu dolaska niskih frekvencija zvuka, ITD .....	16
Slika 21. Razlike u frekvenciji nastale različitim odbijanjem zvuka otkrivaju horizontalnu poziciju izvora zvuka .....	16
Slika 22. Resonance Audio reagira na promjenu pozicije glave .....	17
Slika 23. Utjecaj udaljenosti na intenzitet zvuka .....	18
Slika 24. Resonance Audio simulira odbijene zvukove – rane refleksije .....	18
Slika 25. Preko 50% mobilnih igrica na svijetu koristi Unity game engine [43] .....	19
Slika 26. Fall Guys – igra za djecu koja je iznimno ocjenjena za kaotičan gameplay i vizualan dojam .....	19
Slika 27. Among Us – hit koji je povezao mnoge igrače diljem svijeta .....	20
Slika 28. Pokémon Go - proširena stvarnost, AR .....	20
Slika 29. Unity sučelje .....	21
Slika 30. Postavljanje Resonance Audio Source skripte .....	22
Slika 31. Resonance Audio Demo scena .....	23
Slika 32. Postavljanje reverb sonde vodeći se akustičnim svojstvima .....	24
Slika 33. Moguća pozicija igrača: izvan prostorije, unutar sferne sonde .....	25
Slika 34. Primjer kretanja igrača kroz kvadratne i sferne sonde .....	26
Slika 35. Mapiranje vizualnih objekata s akustičnim materijalima .....	27
Slika 36. Prilagođavanje reverba unutar sonde .....	28
Slika 37. Isključivanje sloja iz kalkulacije reverba .....	29
Slika 38. Uključivanje/isključivanje nestatičnih objekata igre .....	30
Slika 39. Inspector ploča zvuka pozadine .....	30
Slika 40. Prikaz audio datoteke u valnom formatu .....	31
Slika 41. Parametri koje možemo promijeniti unutar Audio Source komponente .....	31
Slika 42. Dodavanje Audio Listener komponente .....	32
Slika 43. Audio Listener komponenta .....	32