# Šolski center celje Srednja šola za strojništvo, mehatroniko in medije

# Uporaba obogatene resničnosti pri razvoju reševalnega robota

Raziskovalna naloga

Področje: elektrotehnika, elektronika in robotika

Avtorja:

Rok Vnučec, M-4. c Filip Kokol, M-4. c Mentorja: dr. Matej Veber, univ. dipl. inž. Andro Glamnik, univ. dipl. inž.

Celje, april 2024

#### IZJAVA\*

Mentor Matej Veber v skladu z 20. členom Pravilnika o organizaciji mladinske raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi z naslovom Uporaba obogatene resničnosti pri razvoju reševalnega robota, katere avtorja sta Rok Vnučec in Filip Kokol:

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljene literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalogo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, 8.4.2024

žig šole

Podpis mentorja

Podpis odgovorne osebe

\*

#### POJASNILO

V skladu z 20. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.

# POVZETEK

V raziskovalni nalogi smo se osredotočili na razvoj in vizualizacijo reševalnega robota za Rapid Manufacturing Robo Cup 2024; uporabili smo tehnologije obogatene resničnosti (AR). Cilj naloge je bil podrobno predstaviti sestavo robota v AR, kar bi izboljšalo razumevanje njegove zgradbe in spodbudilo zanimanje za AR-tehnologijo.

Podrobno smo raziskali in opisali osnove VR, vključno z njeno zgodovino, tehnološkim razvojem in praktično uporabo. Prav tako smo raziskali obogateno resničnost in njen razvoj. V okviru tega projekta smo uporabili Microsoft HoloLens 2 za razvoj AR-aplikacije, JigSpace za 3D-demonstracije in programsko orodje CREO za pripravo modelov. Za razvoj aplikacije smo uporabili tudi Unity, pri čemer smo se poglobili v nastavitve projekta, pripravo modela robota in ustvarjanje animacij.

V zaključku naloge smo prikazali rezultate ankete o splošnem razumevanju in zanimanju za AR- in VR-tehnologijo. Naša ugotovitev je, da imata AR in VR velik potencial v izobraževanju in razvoju aplikacij, kar nakazuje možnost širše uporabe teh tehnologij v prihodnosti.

Ključne besede: obogatena resničnost, virtualna resničnost, vizualizacija, razvoj aplikacije

# SUMMARY

In this research project, we focused on the development and visualization of a rescue robot for the Rapid Manufacturing Robo Cup 2024 using augmented reality (AR) technology. The aim of the project was to detail the assembly of the robot in AR to improve the understanding of its structure and generate interest in AR technology.

We thoroughly investigated and described the basics of VR, including its history, technological development and practical application. We also explored augmented reality and its evolution. As part of this project, we used Microsoft Holo Lens 2 for AR application development, JigSpace for 3D demonstrations and CREO software for model creation. We also used Unity for the development of the application, where we dealt with the project settings, preparation of the robot model and the creation of animations.

At the end of the project, we presented the results of a survey on the general understanding and interest in AR and VR technology. We came to the conclusion that AR and VR have great potential for education and application development, which indicates the possibility of wider use of these technologies in the future.

Key words: Augmented reality, virtual reality, visualization, application development.

# KAZALO

1 UVOD	
1.1 HIPOTEZE	1
2 VIRTUALNA RESNIČNOST – VR	2
2.2 UPORABA	5
3 OBOGATENA RESNIČNOST – AR	6
3.1 TEHNOLOGIJA	9
3.1.1 Strojna oprema	9
3.1.2 Prikaz	9
3.2 UPORABA	9
3.3 MICROSOFTHOLO LENS	
4 TEHNOLOŠKE PLATFORME	
4.1 JIGSPACE	
4.1.1 Vnos robota v JigSpace	
4.3 CREO	
4.4 UNITY	
5 UNITY UREJEVALNIK	
5.1 OSNOVE UREJEVALNIKA	
6 NASTAVITVE PROJEKTA	21
6.1 VNOS MRTK-KNJIŽNIC	
6.2 PRIPRAVA MODELA	
6.2.1 Priprava v Creu	
6.2.2 Priprava v FreeCad	
6.2.3 Vnos v Blender	
7 USTVARJANJE ANIMACIJE	
7.1 ANIMATOR	

8 PROGRAMIRANJE	34
8.1 C#	34
8.2 VISUAL STUDIO	35
8.3 OSNOVE SKRIPTIRANJA V UNITY	36
8.4 EKSPLOZIJSKI POGLED ROBOTA IN NASTAVITVE COLIDERJEV	37
8.5 PRIPRAVA KOMPONENT	39
9 GRADNJA IN PRENOS	42
9.1 UNITY BUILD	42
9.2 VISUAL STUDIO PRENOS	44
10 ANKETA	46
11 ANALIZA HIPOTEZ	50
12 MOŽNOSTI NADALJNJE RAZISKAVE	53
12.1 IZBOLJŠANA INTERAKTIVNOST V APLIKACIJI	53
12.2 INTEGRACIJA UMETNE INTELIGENCE (UI)	53
12.3 UPORABNIŠKO TESTIRANJE	53
13 ZAKLJUČEK	54
14 VIRI	55

# KAZALO SLIK

Slika 1: Obogatena resničnost	6
Slika 2:Na sliki vidimo HoloLens 2	11
Slika 3: Glavni meni JigSpacea	
Slika 4: Prikaz 1	
Slika 5: Prikaz 2	
Slika 6: Prikaz 3	
Slika 7: Prikaz 3 v obogateni resničnosti	14
Slika 8: Meni orodij	15
Slika 9: Okno za objavo	15
Slika 10: Osebna nadzorna plošča	16
Slika 11: Prikaz modela v Vuforia View na telefonu	16
Slika 12: Prikaz modela v Vuforia View na HoloLens 2	17
Slika 13: Hierarhija prizora z mešano resničnostjo	
Slika 14: Nastavitve za zamenjavo platforme	
Slika 15: Priporočene programske različice za razvoj mešane resničnosti	
Slika 16: Okno MRTK	
Slika 17: Izbor projekta	
Slika 18: Možnosti za odkrivanje funkcij znotraj MRTK	
Slika 19: Uvoz funkcionalnosti	
Slika 20: Konfigurator projekta MRTK	
Slika 21: Nastavitve za upravljanje vtičnikov	
Slika 22: Projektna validacija	
Slika 23: Menijska vrstica v programski opremi FreeCAD	27
Slika 24: Izvozna okna formata datoteke	
Slika 25: Možnosti izvoza v programu Blender	
Slika 26: Lokacija Unity projekta	
Slika 27: Meni Window	
Slika 28: Animacijsko okno	
Slika 29: Animacija robota	
Slika 30: Animatorjevo stanje in prehodi	
Slika 31: Inspector Animatorjevega prehoda iz prazen v anim1	

Slika 32: Inspector Animatorjevega prehoda iz anim1 v prazen	
Slika 33: Osnovni program	
Slika 34: Metoda "SproziAnimacijo"	
Slika 35: Metoda "ResetirajAnimacijo"	
Slika 36: Metodi "DisableColliders" "EnableColliders"	
Slika 37: Lastnosti glavnega starša	
Slika 38: Meni Mixed Reality Toolkit	
Slika 39: Lastnosti objekta gumb 1	
Slika 40: Lastnosti objekta gumb 2	
Slika 41: Nastavitve gradnje	
Slika 42: Ustvarjena mapa	
Slika 43: Lastnosti omrežnega prenosa	
Slika 44:Meni za gradnjo aplikacije	
Slika 45: Končno sporočilo	45

# KAZALO TABEL IN GRAFOV

Graf 1: Vprašanje 1	54
Graf 2: Vprašanje 2	55
Graf 3: Vprašanje 3	55
Graf 4: Vprašanje 4	
Graf 5: Podvprašanje 4	
Graf 6: Vprašanje 5	
Graf 7: Vprašanje 6	
Graf 8: Vprašanje 7	
Tabela 1: Možne nastavitve	
Tabela 2: Hipoteze	59

# ZAHVALA

Iskreno bi se rada zahvalila najinima mentorjema, Mateju Vebru in Andru Glamniku, ki sta nama odlično odgovorila na vsako zastavljeno vprašanje in naju pripeljala do končane raziskovalne naloge in projekta. Pri veliko zapletih sta nam pomagala in nama dajala rešitve problemov, ki so se ob pisanju pojavili. Brez njiju nama zagotovo ne bi uspelo napisati tako uspešne naloge.

Zahvaljujeva se tudi profesorici slovenščine, Brigiti Renner, ki je najino nalogo jezikovno pregledala.

### 1 UVOD

Namen raziskovalne naloge je vizualizacija reševalnega robota, ki bo tekmoval na Rapid Manufacturing Robo Cup 2024. Njegove sestavne dele smo v obogateni resničnosti razstavili, kar je pomagalo pri različnih predstavitvah robota, hkrati pa je pripomoglo k boljšemu razumevanju njegove sestave. Prav tako smo z nalogo hoteli izboljšati poznavanje obogatene resničnosti.

#### **1.1 HIPOTEZE**

V okviru raziskovalne naloge smo si postavili naslednje hipoteze:

- Robot bo v AR-aplikaciji samodejno navigiral po virtualnih poteh.
- Vključitev robota v AR je preprosta in zahteva minimalno tehnično znanje.
- Uporabnik s HoloLens in drugi s telefonom lahko sočasno in interaktivno delujeta v istem AR-okolju.
- Večina anketirancev bo poznala razliko med obogateno in virtualno resničnostjo.

# 2 VIRTUALNA RESNIČNOST – VR

Virtualna ali navidezna resničnost fizični svet okoli nas popolnoma nadomesti z digitalnim, kar dosežemo z računalniško ustvarjenim (CG) okoljem ali s 360° videoposnetki. Tako so virtualna resničnost lahko samo programi, ki s pomočjo senzoričnih stimulatorjev, kot so slike in zvoki, pretentajo uporabnikove možgane ter popolnoma spremenijo okolje okoli njega, da verjame, da se nahaja v navidezni resničnosti. To dosežemo s pomočjo VR-očal (HMD), ki imajo pred očmi majhne OLED- ali LCD-monitorje in slušalke na ušesih za upodabljanje stereoskopskih grafik ter binauralnih avdiosistemov. Kar pomeni, da naenkrat zaznavamo predvajani video in zvok, ki nam omogoči, da vidimo tridimenzionalno sliko in slišimo z globino in usmerjenostjo. HMD prav tako v realnem času sledi položaju in rotaciji glave za 6 stopenj svobode, kar nam omogoča, da si s pomočjo obračanja glave ogledujemo virtualni prostor, v katerem se nahajamo. Nekateri VR-sistemi imajo celo nadzor gibanja, ki nam omogoča, da se premikamo znotraj sveta. Gibanje lahko kontroliramo s pomočjo ročnih krmilnikov, toda bolj poglobljeni sistemi imajo vsesmerne tekalne steze s podporo za pas. Te uporabniku omogočajo hojo ali tek tako v resničnem kot tudi v virtualnem svetu, ne da bi padli ali se kam zapletli zaradi motenj in nenehnega premikanja.

VR lahko na splošno razdelimo na tri vrste, glede na njihove stopnje poglobitve. To bi lahko ohlapno razumeli tudi kot ravni dostopnosti, saj so lahko izkušnje VR bolj poglobljene za uporabnikove čute zaradi večje tehnološke vključenosti in običajno več perifernih naprav.

- Nepoglobljena čeprav je izraz bolj tehnične narave in se z izkušnjo navidezne resničnosti ne sklada točno po definiciji, je nepoglobljena virtualna resničnost običajna. Enostaven primer tega je igranje videoiger na televiziji ali računalniškem monitorju, pri čemer ima uporabnik nadzor nad svojim značajem, vendar na koncu sam ni del simulacije. Ločitev virtualnega okolja in uporabnika je tisto, zaradi česar je ta vrsta VR nepoglobljena, zato jo nekateri zavračajo in nasprotujejo razvrščanju nepoglobljenih interakcij med navidezno resničnost.
- Polpoglobljena polpoglobljena virtualna resničnost povezuje neobčutljive in popolnoma poglobljene izkušnje, saj ima globje in zapletenejše prikaze kot prvi, vendar nima dodatne senzorične vpletenosti slednjega. Skladno s tem imajo polpoglobljene interakcije še vedno določeno stopnjo ločenosti od same izkušnje in uporabnika. To

pomeni, da so informacije posredovane preko računalniških monitorjev in zvočnikov ter ne preko VR-očal.

Popolnoma poglobljena – popolnoma poglobljena VR je glavna vrsta navidezne resničnosti. Ta vrsta navidezne resničnosti ustvari popolno potopitev z večjo uporabo specializiranih zunanjih naprav ali tehnologije projekcij in prikazovanj. HMD sledijo gibanju in orientaciji glave, da spremenijo tisto, kar uporabnik vidi v realnem času. To mu omogoča raziskovanje in ogledovanje virtualne pokrajine na skoraj enak način, kot da bi jo opazoval v resničnem svetu. Poglobljene VR-sobe, nasprotno, gradijo okolje okoli uporabnika, ki ga lahko raziskuje brez potrebe po dodatnih zunanjih napravah. Spremljevalni zvok gradi zvočno kuliso in pokrajino, izboljšuje potopitev ter približuje virtualno resničnost k pravi.

Navidezna resničnost uporablja vrsto različnih tehnologij, nekatere so zelo ponovljive in izpopolnjene, druge bolj eksperimentalne ali še v razvoju. Najpogosteje se uporabljajo tiste, ki vključujejo ročne krmilnike in VR-očala, ki spremljajo gibanje uporabnika. Najnaprednejši proizvajalci HMD pripravljajo tehnologijo sledenja očem, ki bodo omogočile VR, da se uči in prilagaja specifični pozornosti oz. osredotočenosti uporabnika. Druge tehnologije, uporabljene v virtualni resničnosti, vključujejo binauralni oz. prostorski zvok, ki ga je mogoče oddati iz katere koli 360° smeri okoli uporabnika, in haptične povratne informacije, ki uporabljajo taktilne povratne informacije, da vzbudijo uporabnikov občutek za dotik na enak način kot za vid in sluh.

VR predstavlja edinstvene izzive in pomisleke v primerjavi s tradicionalnim 2D-oblikovanjem. To zajema tehnične, izkustvene in etične vidike oblikovanja navidezne resničnosti za ustvarjanje poglobljenih, prijetnih in varnih izkušenj virtualne resničnosti.

- Udobje in varnost uporabnika ne sme prihajati do potovalnih slabosti, obremenitve oči in drugih nelagodij z optimiziranimi hitrostmi sličic, zmanjšano zakasnitvijo in funkcijami udobja, kot je npr. teleportacija za premikanje.
- **Izjemen zvok** pozorni moramo biti na 3D prostorski zvok, ki vzbuja in izboljšuje občutek prisotnosti ter potopitve v virtualno okolje.
- Oblikovanje uporabniškega vmesnika (UI) potrebno je oblikovati naravne in spontane uporabniške vmesnike, ki so logično integrirani v VR-okolje, kar zagotavlja jasnost in dostopnost.

- Interakcijski dizajn integriranje naravne in spontane metode interakcije, kot so sledenje rokam ali očem, prepoznavanje kretenj in senzorji gibanja, da se poveča sodelovanje uporabnika.
- Merilo in iskane poti zagotoviti moramo, da aplikacije v navidezni resničnost ohranijo realno merilo in razmerje, da je ustvarjen prepričljiv občutek prisotnosti.
- Optimizacija zmogljivosti kodo je potrebno optimizirati, da zagotovimo gladko delovanje in čim bolj zmanjšamo zahteve glede strojne opreme, s čimer omogočimo dostopnost VR-izkušnje širšemu občinstvu.
- Navigacija in iskanje poti poskušamo ustvariti jasne vizualne namige in orientacijske točke, ki uporabnike vodijo skozi virtualni svet in jim tako zagotovimo enostavno orientacijo.
- Pripovedovanje zgodbe uporabljati moramo različne tehnike pripovedovanja zgodb, uporabnike čustveno pritegnemo in jih vodimo skozi izkušnjo ter spodbujamo globljo povezanost.
- Ustvarjanje vsebin nenehno je potrebno v celoti izkoriščati medij VR in ponujati edinstvene izkušnje, ki jih tradicionalni mediji ne omogočajo, da pridobivamo nove uporabnike.
- Dostopnost in vključevanje upoštevati moramo, da imamo več ciljnih skupin, ki imajo različne preference in si želijo različne funkcije dostopnosti, kot so nastavljiva velikost pisave, glasovni ukazi ali preslikave krmilnikov po meri.
- **Testiranje in povratne informacije** nenehno so potrebna testiranja in zbiranje povratnih informacij uporabnikov, da lahko prepoznamo in odpravimo napake v oblikovanju, težave z udobjem in pomislek glede uporabnosti.
- Povratne informacije o uspešnosti uporabnikom je potrebno zagotoviti povratne informacije o njihovih interakcijah, kot so haptične povratne informacije, vizualni znaki in zvočni odzivi, da povečamo občutek delovanja.

#### **2.2 UPORABA**

Tehnologija virtualne resničnosti se je v preteklih letih močno razširila tako v vsakdanje življenje kot v znanstvene in tehnične panoge, ki nam vsaka na svoj način omogočajo lažje razumevanje, učenje, pridobivanje in preučevanje novih znanj in informacij. Najbolj so se razvile spodaj naštete:

- Igre in zabava VR omogoča poglobljeno, 3D igralno izkušnjo, pri kateri igralci vstopajo v virtualne svetove in interaktivno sodelujejo z okoljem.
- Izobraževanje in usposabljanje VR se uporablja za simulacijo realnih scenarijev, kot so kirurški posegi, pilotiranje letal ali usposabljanje za nujne primere, omogoča varno in nadzorovano učno okolje.
- Medicina uporablja se za terapije, kot so zdravljenje fobij ali posttravmatske stresne motnje, in za rehabilitacije, pri katerih lahko pacienti v virtualnih okoljih izvajajo vaje za obnovitev motoričnih funkcij.
- Arhitektura in oblikovanje arhitekti in oblikovalci uporabljajo VR za vizualizacijo projektov pred izgradnjo, kar omogoča spremembe v zgodnji fazi načrtovanja.
- Uporaba za vojaške namene vojske po vsem svetu uporabljajo VR za usposabljanje vojakov v varnem in kontroliranem okolju, ki simulira bojne razmere.
- **Raziskave in razvoj** znanstveniki uporabljajo VR za vizualizacijo kompleksnih podatkov, kot so molekularne strukture ali meteorološki vzorci.
- **Turizem** VR omogoča "virtualna potovanja" na oddaljene ali nedostopne lokacije, kot so vesolje, globoki ocean ali zgodovinska mesta.
- Film in umetnost VR se uporablja za ustvarjanje poglobljenih filmskih in umetniških izkušenj, pri katerih lahko gledalci "vstopijo" v zgodbo ali umetniško delo. [18], [20],[21]

# 3 OBOGATENA RESNIČNOST – AR

S pomočjo obogatene resničnosti (AR) poskušamo prostor okoli nas interaktivno oplemenititi s pomočjo računalniško ustvarjenih posnetkov (CGI). Z uporabo programske opreme, aplikacij in strojne opreme, kot so AR-očala (HMD), obogatena resničnost z digitalnimi vsebinami obogati resnično okolje in predmete. Lahko jo definiramo kot sistem, ki vključuje tri glavne značilnosti:

- kombinacijo virtualnega in fizičnega sveta,
- interakcije v realnem času,
- natančno 3D-identifikacijo virtualnih in fizičnih objektov.



Slika 1: Obogatena resničnost (Vir: Arch Daily [2])

Informacije, ki jih dobivamo iz okolja preko obogatene resničnosti, so lahko aditivne (dodatek naravnemu okolju) ali destruktivne (prikrivanje naravnega okolja). Njihov namen je izboljšati ali spremeniti resnični svet, namesto da ga preprosto nadomestimo, kot to počnemo z virtualno resničnostjo. Virtualni objekti ali zasloni ležijo nad resničnim svetom s sledenjem položaja, z njimi običajno operiramo prek AR-očal. Obogatena resničnost je povezana z dvema večinoma sopomenskima izrazoma:

- mešana resničnost (MR),
- računalniško posredovana resničnost.

Glavna naloga obogatene resničnosti je povezovanje in bogatenje človekovega dojemanja resničnega sveta z digitalnim. Prvi funkcionalni sistem AR, imenovan Virtual Fixtures, je leta 1992 v laboratorijih USAF Armstrong Labs razvil Louis Rosenberg. Ker je bila 3D-grafika v

zgodnjih 90. letih prejšnjega stoletja prepočasna, da bi predstavila fotorealistično in prostorsko registrirano obogateno resničnost, je Virtual Fixtures uporabil dva fizična robota, ki ju je krmilil polni zgornji eksoskelet telesa, ki ga nosi uporabnik. Za ustvarjanje poglobljene interakcije uporabnika z aplikacijo je bila uporabljena edinstvena konfiguracija optike, ki je vključevala par binokularnih povečeval, poravnanih tako, da je bil uporabnikov pogled na roke robota pomaknjen na položaj njegovih dejanskih fizičnih rok. Rezultat je bila registrirana poglobljena izkušnja, v kateri je uporabnik premikal svoje roke, skozi leče pa je videl, kako se na mestu njegovih rok premikata robota. Sistem je uporabljal tudi računalniško ustvarjanje navideznih oblik, zasnovanih za pomoč uporabniku pri izvajanju resničnih nalog.

Kasneje so se aplikacije AR razširile še na druga področja. Ta tehnologija je še posebej uporabna za vojaške in medicinske namene, saj obe zahtevata veliko osredotočenost in informacije v stresnih scenarijih. Prav tako je ne manjka v izobraževanju, komunikacijah in zabavi.

Delovanje obogatene resničnosti zajema tri glavne korake:

- Zaznavanje in sledenje naprava AR zaznava okolje s kamerami, merilniki pospeška, giroskopi, GPS in celo laserji za sledenje položaja ter orientacije uporabnika in njegove naprave.
- Obdelava in prikazovanje slik sistem analizira podatke senzorjev in identificira predmete ali značilnosti v okolju, ki jih je mogoče povečati. Naprava uporablja algoritme za obdelavo slik, prepoznavanje in sledenje objektov ter okolja v realnem času.
- Upodabljanje in prikaz zadnji korak je ustvarjanje in prikaz računalniško ustvarjene vsebine preko resničnega okolja. Ta korak upodobi in prikaže navidezne predmete v pravilni perspektivi ter položaju glede na zorni kot uporabnika.

Obogatena resničnost se med drugim ukvarja tudi z izboljševanjem naravnih okolij ali situacij in uporabnikom ponuja obogateno izkušnjo dojemanja fizičnega sveta. Ob pomoči naprednih tehnologij AR (dodajanje računalniškega vida, vgradnja AR-kamer v aplikacije za pametne telefone, prepoznavanje okolja in predmetov) postanejo informacije v realnem svetu uporabnika interaktivne in digitalno manipulirane. Uporabljajo se za različne namene, vključno z igranjem iger, vizualizacijo izdelkov, marketinškimi aplikacijami, arhitekturo, oblikovanjem doma, izobraževanjem in industrijsko proizvodnjo. Informacije o okolju in njegovih predmetih se prekrivajo z resničnim svetom, primer tega je videoigra Pokémon GO, v kateri uporabniki v svojih resničnih soseskah iščejo animirane like, ki se prikažejo na njihovem telefonu ali tablici. Prav tako se poglobljene zaznavne informacije včasih kombinirajo z dodatnimi informacijami, kot so rezultati na prenosih športnih dogodkov v živo. To združuje prednosti tehnologije obogatene resničnosti in tehnologije »heads up display« (HUD).

#### **3.1 TEHNOLOGIJA**

#### 3.1.1 Strojna oprema

Komponente strojne opreme so procesor, zaslon, senzorji in vhodne naprave. Sodobne mobilne računalniške naprave, kot so pametni telefoni in tablični računalniki, že vsebujejo te elemente. To so kamera in senzorji mikroelektronskih sistemov (MEMS), natančneje rečeno, to so merilnik pospeška, sistem za določanje položaja (GPS), polprevodniški kompas, giroskop, kar jih naredi primerne za AR-platforme.

#### 3.1.2 Prikaz

Za upodabljanje AR lahko uporabljamo različne platforme, kot so optično projekcijski sistemi, monitorji, mobilne naprave in zasloni, ki se nosijo na telesu. Najpogostejši je naglavni zaslon (HMD). To je naprava, ki je nošena na glavi kot čelada ali trak, preko oči imamo steklo z lečami, kamor je projicirana slika. Tako HMD postavi slike fizičnega sveta in virtualnih predmetov v uporabnikovo vidno polje. Pogosto uporablja senzorje za spremljanje šestih stopenj svobode, ki sistemu omogočajo, da navidezne informacije uskladi s fizičnim svetom in se ustrezno prilagaja glede na uporabnikove premike glave. Upravljamo ga lahko s kretnjami rok, sledenjem oči in z glasovnimi ukazi.

#### **3.2 UPORABA**

Tehnologija obogatene resničnosti se uporablja v številnih panogah:

- Trgovina uporabnik lahko izdelek, preden ga kupi, pogleda v lastnem domu, npr. pohištvo.
- Navigacija pot se lahko projicira v realno okolje s pomočjo dodatnih informacij.
- Gradbeništvo prikaz objektov, hiš itd. v realnem okolju.
- Izobraževanje omogoča interaktivno in individualno učenje za boljše razumevanje zahtevnejše snovi.
- Industrija pri vzdrževanju strojev ponuja serviserjem neposredna navodila in informacije, kar izboljšuje učinkovitost ter zmanjšuje napake.
- Filtri v aplikacijah AR-filtri, ki so na voljo v socialnih medijih in aplikacijah za sporočanje, omogočajo uporabnikom, da spreminjajo ali izboljšujejo svojo, dodajajo zabavne efekte ali kozmetiko.

Igre – zagotavljajo edinstveno igralno izkušnjo, v kateri se digitalni elementi prepletajo z resničnim svetom.

#### **3.3 MICROSOFTHOLO LENS**

HoloLens so očala za ustvarjanje obogatene resničnosti, ki jih je razvilo in proizvedlo podjetje Microsoft. Microsoft je do zdaj izdal dve generaciji tega produkta, HoloLens 1, ki je izšla 30. 3. 2016, in HoloLens 2, 9. 11. 2019. Cena je ob izdaji za prvo generacijo znašala 3000 \$; danes novih na trgu ne moremo več kupiti, cena rabljenih očal pa se giblje med 400 \$ in 800 \$. Cena očal za drugo generacijo pa se je dvignila za 500 \$.

Naš program smo razvijali za drugo generacijo, saj šola ponuja uporabo teh očal in izobraževanje na njih. HoloLens 2 je samostojen holografski računalnik. Prinaša udobnejšo in kakovostnejšo uporabnost kot HoloLens 1. Operacijski sistem, ki deluje na očalih, je »Windows Holographic OS« z lastnostmi »Windows 10«. Za uporabo HoloLensov je prav tako potreben račun, v katerega se moramo prijaviti.

#### Komponente:

- **Vizir** vsebuje vse senzorje, kot so mikrofon, kamere za sledenje oči in zaznavanje okolja, senzor globine ter zaslon, ki se lahko med uporabo dvigne.
- **Gumb za vklop** s pritiskom na gumb se naprava vklopi, ob ponovnem pritisku preide v stanje spanja. Če želimo napravo izklopiti, moramo gumb zadržati za 5 sekund.
- LED-indikatorji 5 LED-indikatorjev nam prikazuje nivo baterije, vsak indikator nam predstavlja 20 % celotne ravni baterije. Baterija je popolnoma napolnjena, ko sveti vseh 5 LED-diod.
- USB Type-C vrata HoloLens 2 je opremljen s kablom USB-C in 18 W polnilnikom. Ta vrata lahko uporabljamo tudi za povezavo z računalnikom in za prenos podatkov.
- Naglavni trak je nastavljiv z nastavitvenim kolescem na zatilju in ga je mogoče tudi sneti z naprave.
- Gumba za svetlost nahajata se na levi strani vizirja, sprednji svetlost zaslona poveča, zadnji jo zmanjša.
- **Gumba za glasnost** nahajata se na desni strani vizirja, sprednji glasnost naprave poveča, zadnji jo zmanjša. [1], [3], [4], [5], [7], [20]



Slika 2:Na sliki vidimo HoloLens 2 (Vir: https://www.fireware.nl/ [10])

# 4 TEHNOLOŠKE PLATFORME

### **4.1 JIGSPACE**

JigSpace je interaktivni 3D-program, ki dovoljuje uporabnikom, da ustvarijo AR-demonstracije in prezentacije na preprost način. Modele, ki so bili vneseni vanj, si lahko ogledamo z napravami, ki delujejo na Androidovem ali Applovem operacijskem sistemu. [9]

#### 4.1.1 Vnos robota v JigSpace

Najprej smo naložili JigSpace na računalnik. Ta je dostopen na strani https://www.jig.space/. Nato smo si v programu ustvarili profil, da smo lahko svoj model robota vnesli v obogateno resničnost. Z njihovim »Free Trial planom« smo imeli dostop do vnosa enega t. i.»Jiga« (modela).

Da smo lahko začeli, smo v desnem zgornjem kotu pritisnili na »Create new Jig«.



Nato nas je program vprašal, ali želimo delati s CAD-modeli, ki so v knjižnici (določeni že vnaprej) ali imamo svojo CAD-datoteko, ki jo bomo sami vnesli v program. Izbrali smo drugo možnost in vnesli datoteko našega robota s formatom STEP. Ta format smo iz ASM-formata spremenili s programom CREO.

Z brezplačnim preizkusom, ki smo ga aktivirali, imamo možnost narediti največ tri slike animacij. Za prvo sliko smo robota pustili v navadni poziciji.



Slika 4: Prikaz 1 (Vir: Osebni arhiv)

Na drugi sliki smo iz robota sneli pokrov in prikazali planetrano gonilo.



Slika 5: Prikaz 2 (Vir: Osebni arhiv)

Za zadnjo sliko smo, prej roko popolnoma odstranili in predstavili notranje komponente robota.



Slika 6: Prikaz 3 (Vir: Osebni arhiv)



Slika 7: Prikaz 3 v obogateni resničnosti (Vir: Osebni arhiv)

#### **4.3 CREO**

Drugi način za postavitev robota v obogateno resničnost je preko programa, v katerem je bil modeliran. V Creu smo generirali QR-kodo, ki je univerzalna za našega robota, preko katere je aplikacija »Vuforia Viewer« postavila robota v resničnost z uporabo kamere. Da smo jo ustvarili, smo v programu Creo odprli zavihek »Tools« in izbrali »Target«.



#### Slika 8: Meni orodij (Vir: Osebni arhiv)

Zatem se nam je ponudila možnost, da izberemo »Thing Mark« ali »Spatial Target«. Če smo izbrali prvo možnost, je naš model deloval le na telefonu. Ob izbiri »Spatial Target« smo lahko model projicirali tudi na HoloLensih. Nato se je, kljub katerikoli izbiri pod 3D-modelom, pojavil »Thing Mark«, ki smo ga lahko poljubno premikali, kar je pozneje vplivalo na postavitev modela v obogateni resničnosti.

Nato smo morali model objaviti. To smo storili s pritiskom na »Publish Model«. Program je zahteval, da se prijavimo v Creo račun in ko smo to storili, smo lahko določili ime in kakovost modela. Prav tako je bilo potrebno izbrati »Thing mark«, ki smo ga želeli objaviti.

Ρ	ublish Experience Model	х
Model Name:	rmrcS	
Viewable quality:	High	Ŧ
Publish To:	Personal	Ŧ
Content:	Current state	•
Target:	ThingMark_1	
	OK Canc	el

Slika 9: Okno za objavo (Vir: Osebni arhiv)

Po nekaj sekundah se je odprlo okno, kar je pomenilo, da je bila objava uspešna. QR-kodo smo pridobili tako, da smo pritisnili na »Manage Model«, kar je odprlo »Design Share Portal«. Tu se nam je prikazalo ime modela, ki smo ga objavili. Pod »Access« smo izbrali »Public«, da bo lahko kdorkoli, ki bo skeniral kodo, videl model. Nato smo pritisnili »Share« in delili QR-kodo.

Personal					
Model Dashboard					
Model Name	Target	Date	Access	Share	Delete
rmrcs		1. 03. 2024	Public 🔽	$\boldsymbol{<}$	圃
nov_robot	[+]	23. 02. 2024	Public 🖌	<	圃

Slika 10: Osebna nadzorna plošča (Vir: Osebni arhiv)

Model lahko nato opazujemo s katerokoli napravo, ki ima nameščeno aplikacijo »Vufira Viewer«.



Slika 11: Prikaz modela v Vuforia View na telefonu (Vir: Osebni arhiv)



Slika 12: Prikaz modela v Vuforia View na HoloLens 2 (Vir: Osebni arhiv)

#### **4.4 UNITY**

Unity je priljubljen igralni pogon in platforma za razvoj, ki ga pogosto uporabljajo tako neodvisni razvijalci kot večja razvojna podjetja. Predstavljen in izdan je bil junija 2005 na Apple Worldwide Developers Conference kot igralni pogon za Mac OS X. Pogon se je kasneje postopoma razširil, da podpira različne namizne, mobilne, konzolne in razširjeno resničnostne platforme. Še posebej priljubljen je za iOS in Androidov razvoj mobilnih iger, ker je enostaven za uporabo tako za začetnike, priljubljen pa je tudi med neodvisnimi razvijalci. Pogon se lahko generira za ustvarjenje tako tridimenzionalnih (3D) in dvodimenzionalnih (2D) iger kot interaktivnih simulacij in drugih izkušenj.

Glavne prednosti in značilnosti so:

- Večplatformskost Unity omogoča razvoj iger in interaktivnih aplikacij za široko paleto platform, vključno z osebnimi računalniki, konzolami, mobilnimi napravami in spletnimi brskalniki.
- Uporabniški vmesnik ima priročen in intuitiven grafični vmesnik, ki olajša razvoj iger in aplikacij.
- **Programski jezik** uporablja C# kot glavni programski jezik, ki je znan po svoji moči in prilagodljivosti.
- Grafična zmogljivost ponuja napredne grafične zmogljivosti, ki so primerne tako za 2D- in 3D-igre, vključno z osvetlitvijo, senčenjem in fizikalnimi učinki.
- Fizikalni motor vključuje močan fizikalni motor, ki omogoča realistično simulacijo gibanja in trkov.
- **Skupnost in podpora** Unity ima obsežno skupnost razvijalcev in bogat nabor učnih virov, vključno z dokumentacijo, vodiči, videoposnetki in forumi.
- Licenčna politika Unity ponuja različne licenčne možnosti, od brezplačne različice za neodvisne razvijalce in majhne studie, do naprednejših, plačljivih različic za večja podjetja.

Unity je izjemno vsestranski in uporabniku prijazen igralni pogon, ki omogoča razvijalcem vseh spretnostnih ravni, da svoje vizije oživijo v interaktivnih izkušnjah. Zaradi svoje zmožnosti večplatformskega razvoja in velikega nabora funkcij je postal ključno orodje za inovacije v industriji videoiger in širše. Unity ni le orodje za ustvarjenje iger, ampak platforma, ki presega industrijo, spodbuja kreativnost in omogoča razvoj aplikacij, ki spreminjajo svet interaktivne zabave, izobraževanja in simulacij. [16]

#### **5 UNITY UREJEVALNIK**

Da smo lahko uporabljali Unity urejevalnik, smo si najprej ustvarili Unity ID, ki nam je omogočil dostop do vseh verzij Unity urejevalnika in njihovih predlog (templates) za ustvarjanje aplikacij. Račun smo si ustvarili na njihovi spletni strani https://id.unity.com/.

Nato smo naložili Unity Hub, ki je aplikacija, preko katere smo lahko nalagali prej omenjene verzije Unity urejevalnika. Unity Hub smo uporabljali tudi za pregledno shranjevanje projektov, ki smo jih ustvarjali z Unity. V Unity Hubu smo lahko opazili tudi »Learn« okno, v katerem smo našli različne vaje (vadnice), ki nas spoznajo z Unity urejevalnikom, in okno »Community«, preko katerega smo lahko dostopali do različnih blogov, trgovine s sredstvi, ki pripomorejo k lažjemu razvoju aplikacij, debat drugih uporabnikov ipd.

Preden smo lahko naložili Unity urejevalnik in ga začeli uporabljati, smo morali aktivirati licenco. Osebna licenca je za razvijalce v Unityju brezplačna. Aktivirali smo jo tako, da smo se v Unity Hubu prijavili s svojim Unity ID, nato kliknili na svojo profilno ikono in izbrali »Manage license«. To nas je popeljalo do okna z naslovom »License«, kjer smo lahko upravljali z licencami. Da smo dobili osebno licenco, smo kliknili »Add« in nato pravilno izbrali.

Za razvijanje aplikacije na HoloLensih je bilo zelo pomembno, da so bile verzije vseh programov in knjižnic, ki smo jih uporabili za razvoj in prenos aplikacije, med seboj kompatibilne. Za naš projekt smo uporabili verzijo 2022.3.20f1 LTS, z dodanimi moduli za »Universal Windows Platform Build Support«, »WebGL Build Support«, »Windows Build Support (IL2CPP)« in Visual Studio verzije 2022. LTS (Long Term Support) verzija Unityja je zagotavljala stabilnost in dolgoročno podporo, "f" in "a" oznake pa so označevale finalne in alfa različice posodobitev.

### 5.1 OSNOVE UREJEVALNIKA

Za razvoj aplikacije smo uporabljali ključne elemente, gradnike in funkcionalnosti. Najpomembnejši so bili:

- Game Object je osnovna enota vsega, kar je v aplikaciji. To so lahko 3D-modeli, svetlobni viri ipd. Sam po sebi nima funkcionalnosti in je v prostoru samo prisoten, dokler mu ne dodamo komponent.
- **Kamera** je virtualno okno, skozi katerega igralec vidi svet v igri. Kako deluje, vidimo ob simulaciji aplikacije.
- Animator Controller in Animation Animator Controller je komponenta, ki upravlja z animacijami, ki so posnete in povezane z Game Objekti. Animacija je niz sličic, ki vsebujejo gibanje objekta (v našem primeru 3D-modela).
- Skripte so kode, zapisane v C#, in jih lahko dodamo kateremukoli Game Objectu ter mu tako dodamo lastnosti, ki so napisane v programu skripte.
- **Colliders** so komponente, ki določajo fizične meje objekta za zaznavanje dotikov (vsak del robota ima svoj Collider).
- Inspector je orodje v urejevalniku, ki omogoča ogled in urejanje lastnosti izbranega Game Objekta ali komponente.
- Scene predstavlja vizualno okolje, v katerem predstavljamo Game Objekte.
- Hierarhija je razdelek znotraj urejevalnika, ki prikazuje vse objekte v trenutni sceni, v drevesu podobnem prikazu. To omogoča organiziranje objektov, hitro navigacijo med njimi in urejanje njihovih odnosov (npr. objekti lahko postanejo "otroci" drugih objektov (staršev), kar pomeni, da sledijo njihovim spremembam).



Slika 13: Hierarhija prizora z mešano resničnostjo (Vir: Osebni arhiv)

### **6 NASTAVITVE PROJEKTA**

Ko smo naložili želeno verzijo Unity urejevalnika, smo ustvarili nov projekt s klikom na »New Project«. S tem smo prišli do določanja, kje na računalniku smo želeli, da je projekt shranjen, kako smo ga poimenovali in na kakšni predlogi smo želeli ustvarjati. Za razvoj aplikacije, ki bo delovala v obogateni resničnosti, smo izbrali 3D-predlogo, kar omogoča postavljanje tridimenzionalnih modelov v okolje.

Ob prvem zagonu projekta smo v Unityju morali spremeniti »Build settings«, torej nastavitve, ki določajo, za katero platformo bo aplikacija zgrajena. Najprej smo v »Filu« pritisnili na »Build Settings«. Po privzetih nastavitvah smo opazili, da so bile nastavljene za razvoj aplikacij za sisteme Windows, Mac in Linux. To smo spremenili tako, da smo pod »Platform« izbrali »Universal Windows Platform« in pritisnili »Switch Platform«.



Slika 14: Nastavitve za zamenjavo platforme (Vir: Osebni arhiv)

### 6.1 VNOS MRTK-KNJIŽNIC

»Mixed Reality Toolkit« (MRTK) je programski modul ali knjižnica, ki jo vodi Microsoft, in je namenjena pospeševanju razvoja aplikacij med različnimi platformami mešane resničnosti v Unityju. MRTK zagotavlja sistem za vhodne naprave različnih platform in podpira gradnjo modelov za prostorsko interakcijo ter uporabniške vmesnike. Ta orodja omogočajo hitrejši razvoj simulacij neposredno v urejevalniku, kar pomeni, da lahko razvijalci takoj opazijo spremembe in prilagoditve. Ta funkcionalnost je še posebej koristna pri interativnem procesu oblikovanja in razvoja aplikacij za mešano resničnost.

Deluje na številnih napravah, npr.:

- Microsoft HoloLens 2
- Meta Quest
- Windows Mixed Reality
- SteamVR
- Oculus Rift on OpenXR

Software	Version	Notes
Microsoft Visual Studio @	2019 Community edition or greater	Recommend Visual Studio 2022
Unity	2020.3.35+ or 2021.3.4+	Recommend using an LTS release, make sure to check out known issues in certain Unity versions
Mixed Reality Feature Tool for Unity <sup>교</sup>		Used to acquire MRTK3 packages
Mixed Reality OpenXR Plugin		Install via Mixed Reality Feature Tool

Slika 15: Priporočene programske različice za razvoj mešane resničnosti (Vir: Osebni arhiv)

Najprej smo naložili MRTK iz Microsoftovega centra za prenose na https://www.microsoft.com/en-us/download/. Če se nam je datoteka naložila v .zip formatu, jo je bilo potrebno ekstrahirati in zagnati. Ob zagonu se nam je prikazalo okno, na katerem smo videli verzijo MRTK-ja; tam smo pritisnili start.



Slika 16: Okno MRTK (Vir: Osebni arhiv)

Nato smo izbrali pot, kjer je bil shranjen naš projekt. Tako je lahko program vnesel želene funkcije v Unity urejevalnik.



Slika 17: Izbor projekta (Vir: Osebni arhiv)

Prva funkcija, ki je bila potrebna za razvoj naše aplikacije, se je nahajala pod zavihkom *Mixed Reality Toolkit*, imenovana *Mixed Reality Toolkit Foundation*, druga pa pod zavihkom *Platform Support*, z imenom *Mixed Reality OpenXR Plugin*. Obe smo obkljukali in pritisnili »Get features«.

83	Micro	soft Mixed Reality Feature Tool		- 🗆 ×
	Di	scover Features		_
	ŧ	Azure Mixed Reality Services (0 of 7)		Select All
	Ŧ	Experimental (0 of 4)		Select All
	Ŧ	Mixed Reality Toolkit (1 of 10)		Select None
	Ŧ	MRTK3 (0 of 15)		Select All
	Ŧ	Platform Support (1 of 5)		Select None
	Ŧ	Spatial Audio (0 of 1)		Select All
	Ŧ	World Locking Tools (0 of 4)		Select All
	_	, , , , , , , ,		
I	Last upo	lated 26. 02. 2024 17:51:00		
	Ü	? Show preview releases	Go Back	Get Features

Slika 18: Možnosti za odkrivanje funkcij znotraj MRTK (Vir: Osebni arhiv)

Na naslednji strani se nam je prikazal seznam izbranih funkcij in seznam funkcij, ki smo jih morali zaradi odvisnosti od izbranih še dodati. Najprej smo pritisnili na »Validate«, kar je preverilo, če je bilo po nalaganju funkcij vse ustrezno. Nato smo pritisnili import, kar je naložilo funkcije v naš projekt.



Slika 19: Uvoz funkcionalnosti (Vir: Osebni arhiv)

Ko smo odprli Unity urejevalnik, nam je le-ta sporočil, da se bo ponovno zagnal, da bi lahko integriral vse dodane funkcije. Po ponovnem zagonu se je prikazalo okno z imenom »MRTK Project Configurator«. Ta konfigurator nam je pomagal nastaviti parametre, ki so potrebni za kasnejše nalaganje aplikacije na HoloLens.

Najprej smo pritisnili na »Unity OpenXR plugin (recommended)«. To nas je pripeljalo v okno »Project Settings« pod zavihek »XR Plug-in Management«. Tam smo obkljukali »OpenXR« in »Windows Mixed Reality feature group«. Ta korak je bil ključen za zagotovitev, da bo naša aplikacija pravilno delovala na platformi HoloLens, saj so ti nastavitveni parametri omogočali ustrezno podporo za funkcije, ki jih potrebuje mešana resničnost.

MRTK Project Configurator		x
Welcome to MRTK!		
This configurator will go through some settings		
XR Pipeline Setting - Enabling the XR SDK Pip	eline	
To build applications for AR/VR devices you ne the XR SDK pipeline there are two categories o	ed to enable an XR pipeline. Please make sure you are targetir f provider plugins:	ng the desired build target before proceeding. With
Unity OpenXR plugin (recommended)		
Choose this if you want to embrace the new in Windows Mixed Reality headsets, Quest 1 and	dustry standard and easily support a wide range of AR/VR dev 2, and other conformant runtimes. The Unity OpenXR Plugin w	ices in the futurel Officially supports HoloLens 2, ill be installed.
Built-in Unity plugins (non-OpenXR) Choose this if your application needs to suppo Unity XR Management Plugin will be installed i	rt platforms beyond HoloLens 2 and Windows Mixed Reality he f not already.	adsets (e.g. Oculus/Magic Leap headsets). The
For more information, please click on the Learn		
	Learn More	
Not ready to setup the project now?		
Skin This Sten	Skin Satun Until Next Sassion	Always Skin Satun
Skip Tills Step	Skip Setup Ontil Next Session	Always Skip Setup

Slika 20: Konfigurator projekta MRTK (Vir: Osebni arhiv)

🌣 Project Settings					: 🗆 ×
Adaptive Performance Audio Editor	XR Plug-in Manaq	gement			
Graphics	<b>P</b>	0.00 0.00	÷.	Ü	4
Input Manager	Initialize XR on Startun				
Input System Package Memory Settings					
Mixed Reality Toolkit	Plug-in Providers 😲				
Build Settings					
Package Manager Physics	OpenXR @ A				
	<ul> <li>Windows Mixed Reali</li> </ul>	ty feature group 😧			
Player Preset Manager	Holographic Remotin	g remote app feature group (	9		
Quality	XR Simulation				
Scene Template	Unity Mock HMD				
Script Execution Order Services					
Tags and Layers					
TextMesh Pro	Information about configurati	ion, tracking and migration ca	n be found below.		
Timeline					
UI Builder					
Version Control Visual Scrinting					
V XR Plug-in Management					
OpenXR					
XR Simulation					

Slika 21: Nastavitve za upravljanje vtičnikov (Vir: Osebni arhiv)

Pomembno je, da smo odpravili napake, ki so nastale zaradi izbranega vtičnika.To smo storili tako, da smo pritisnili na trikotnik, ki se je pojavil v prejšnjem koraku, kar nas je pripeljalo pod zavihek *»Project validation«*. Tam smo odpravili napake, tako da smo pritisnili na *»Fix all«*. Kljub temu je ostala napaka, ki smo jo morali popraviti sami.

Project Settings					:
Adaptive Performance Audio Editor Graphics	Project Validation Scene Validation Selected Profiles	<b>n</b> <sup>-</sup> um Off			
Input Manager	Validation Rules				
Memory Settings			+	ê	4
Mixed Reality Toolkit     Build Settings	Issues (1) of Checks	(10)		Show all	lanore build errors
Package Manager Physics	✓ [OpenXR] The OpenXR	package has been updated a			
Physics 2D Ployer		Package Settings asset has	duplicate settings and must be	regenerated.	
Preset Manager Quality	[OpenXR] At least one the Features menu.				it in Edit
Scene Template Script Execution Order Services	<ul> <li>[OpenXR] The only sta operating systems are</li> </ul>	ndalone targets supported are not supported at this time.			
Tags and Layers TextMesh Pro					
Time Timeline					
Ul Builder Version Control	<ul> <li>[OpenXR] [Optional] Sydeprecated in a future</li> </ul>	vitch to use InputSystem.XR.F release.			
Visual Scripting VIsual Scripting XR Plug-in Management	<ul> <li>[OpenXR] [Optional] Synthetic base code dependence</li> </ul>	vitch to use StickControl thun	hbsticks instead of Vector2Cor	trol, but may break existing pro	jects Fix
Privšitas Privšitas 20 Privšitas 20 Player Preset Manager Quality Scene Template Script Execution Order Services Tags and Layers TextMesh Pro Time Timeline Ul Builder Visual Scripting * XR Plug-in Management OpenXR Project Validation XR Simulation	control, such as acting buttons.	as both a combined 2D vecto	r, two independent axes or a fo	our-way Dpad with 4 independe	ent
	<ul> <li>[Mixed Reality Feature complete the update.</li> </ul>				
	<ul> <li>[Mixed Reality Feature performance costs for ]</li> </ul>	s] "Run in Background" is not KR Unity apps.			

Slika 22: Projektna validacija (Vir: Osebni arhiv)

Napako smo odstranili tako, da smo pod zavihkom »OpenXR« v oknu »Enabled Interaction Profiles« dodali »Hand Interaction Profile« in »Eye Gaze Interaction Profile«. S tem smo določili vnose, ki so ustvarili interakcijo z aplikacijo. [6], [8], [11], [12], [15]

#### **6.2 PRIPRAVA MODELA**

#### 6.2.1 Priprava v Creu

Da smo lahko model robota naložili v Unity, je bilo potrebno njegov format spremeniti, da bo kompatibilen z urejevalnikom. Unity podpira <u>.fbx</u>, <u>.dae</u>, .3ds, .dxf, .obj formate. Format .asm je datoteka, pisana v nizkem nivoju programskega jezika, ki se imenuje »asembly language«. Vsebuje zaporedje operacij, ki generirajo objektno kodo.

Ta format smo v Creu spremenili v format .stp, ki je 3D CAD (Computer-Aided Design) datoteka, uporabljena za menjavo modelov med CAD in CAM (Computer-Aided Manufacturing) aplikacijami (razlika med obema je, da CAD uporabljamo za razvoj dizajna modela, CAM pa za proizvodnjo enakega modela).

#### 6.2.2 Priprava v FreeCad

Da smo lahko model vnesli v program Blender, ki ga Unity podpira in čigar modele lahko naloži, smo .stp format morali spremeniti še enkrat v .gltf (GL Transmission Format). Ta prav tako shranjuje informacije o 3D-modelih, ampak v načinu JSON-formata. Uporaba tega minimizira velikost 3D-sredstev in čas procesiranja, ki je potreben za »razpakiranje« vseh sredstev. To smo storili s prenosom programa FreeCAD in uvozom .stp datoteke robota.



Slika 23: Menijska vrstica v programski opremi FreeCAD (Vir: Osebni arhiv)

Nato smo datoteko izvozili, tako da smo izbrali celoten model, pod zavihkom datoteka pritisnili na izvozi in za vrsto datoteke izbrali .gltf format.



Slika 24: Izvozna okna formata datoteke (Vir: Osebni arhiv)

#### 6.2.3 Vnos v Blender

Tako je bil 3D-model pripravljen za uvoz v Blender. Na meniju smo izbrali »File«, zavihek »Import« in .gltf format ter uvozili naš model.



Slika 25: Možnosti izvoza v programu Blender (Vir: Osebni arhiv)

Shranili smo ga kot .blend vrste na želeno lokacijo in ga kopirali v mapo »Asetss«, ki se je nahajala v našem Unity projektu. [13]

Elements (D:) > Unity projects > final > Assets	;		
↑ Ime	Datum spremembe	Vrsta	Velikost
animations	4. 03. 2024 19:41	Mapa z datotekami	
MixedRealityToolkit.Generated	4. 03. 2024 11:34	Mapa z datotekami	
MRTK	1.03.202404:33	Mapa z datotekami	
Scenes	4. 03. 2024 19:41	Mapa z datotekami	
	1.03.202415:07	Mapa z datotekami	
TextMesh Pro	1.03.202404:35	Mapa z datotekami	
XR	4. 03. 2024 11:43	Mapa z datotekami	
📄 animations.meta	1.03.202404:38	Datoteka META	1 KB
MixedRealityToolkit.Generated.meta	1.03.202404:33	Datoteka META	1 KB
MRTK.meta	1.03.202404:33	Datoteka META	1 KB
rmrc.blend.meta	29. 02. 2024 21:09	Datoteka META	3 KB
🔊 robot.blend	29, 02, 2024 21:08	Blender 4.0	157.938 KB

Slika 26: Lokacija Unity projekta (Vir: Osebni arhiv)

## 7 USTVARJANJE ANIMACIJE

Za ustvarjanje animacije modela v eksplozijskem pogledu smo pritisniti na zavihek »Window«, možnost »Animation« ali pa uporabili bližnjico s pritiskom na »Ctrl+6«.



Slika 27: Meni Window (Vir: Osebni arhiv)

To nam je odprlo okno z imenom »Animation«. Nato smo izbrali celotnega robota in ustvarili nov posnetek. Zatem smo začeli s snemanjem animacije, in sicer s pritiskom na rdeč krog v odprtem oknu.



Slika 28: Animacijsko okno (Vir: Osebni arhiv)

Potem smo izbrali posamezne dele, za katere smo želeli, da se med animacijo med seboj ločijo in jih razporedili po abscisni osi. Abscisna os prikazuje čas, na kateri je število 1:00 enako 1 sekundi.



Slika 29: Animacija robota (Vir: Osebni arhiv)

#### 7.1 ANIMATOR

V mapi »Assetss« smo nato ustvarili novo komponento »Animation Controller« in vanjo vnesli našo animacijo. Ta se je samo delno povezala s stanjem »Entry«. To povezavo smo uničili, saj bi se animacija v nasprotnem primeru aktivirala takoj ob zagonu aplikacije. Ustvarili smo novo prazno stanje in naredili povezavo, ki teče iz »Entry« v prazno stanje. Nato smo stanje »Any State« povezali z našo animacijo (anim1) in stanji prezen ter anim1 povezali med seboj.



Slika 30: Animatorjevo stanje in prehodi (Vir: Osebni arhiv)

Sledilo je ustvarjanje dveh parametrov, ki bosta pozneje klicana v skripti za zamenjavo stanja. Prvi parameter smo poimenovali »Explode« in ga vnesli v pogoje povezave, ki poteka iz stanja prazen, ko je robotovo celotno stanje v anim1, kar ga razstavi. Tako smo si pripravili pogoj, ki bo ob njegovem priklicu zamenjal stanje robota.

Inspector		а:
prazen -> anim1		
Animator I ransitionBase		
Transitions	Solo	Mute
prazen -> anim1		
		•
prazen -> anim1		
Has Exit Time 🗸		
▶ Settings		
Cannot preview transition: source state does not have motion		
Conditions		
= Explode •		
	+	

Slika 31: Inspector Animatorjevega prehoda iz prazen v anim1 (Vir: Osebni arhiv)

Drugi parameter smo poimenovali »Reset« in ga vnesli v povezavo iz stanja anim1 v stanje prazen. Ta pogoj bo postavil robota v prvotno stanje.

Inspector	а:
<ul> <li>anim1 -&gt; prazen</li> <li>1 AnimatorTransitionBase</li> </ul>	‡÷
Transitions	Solo Mute
anim1 -> prazen	
T	\$
≥nim1_> prazon	
Has Exit Time	
► Settings	
,0:10 10:20 11:00 11:10 11:20 12:00 12:10 12:20	3:00  3:10
anim1	
prazen	
Conditions	
= Reset	
	+

Slika 32: Inspector Animatorjevega prehoda iz anim1 v prazen (Vir: Osebni arhiv)

### **8 PROGRAMIRANJE**

Unity uporablja za pisanje skript, ki določajo delovanje aplikacije, programski jezik C#. Te skripte pišemo v programu Visual Studio, ki je integrirano razvojno okolje (IDE). Unity in Visual Studio sta integrirana, kar pomeni, da lahko razvijalci preprosto preklapljajo med IDE in Unity urejevalnikom.

#### 8.1 C#

C# (izgovarja se kot "C sharp") je zasnovan za splošno uporabo in preprosto razumevanje ter omogoča izdelavo široke palete aplikacij, vključno z namiznimi aplikacijami, spletnimi storitvami, mobilnimi aplikacijami in igrami. Njegova sintaksa je podobna Javi in C++.

Ko pišemo program v C#, je pomembno, da poznamo ključne sintakse tega jezika. V C# se spremenljivke deklarirajo z navedbo njenega tipa (int, bool, string, float), ki mu sledi ime spremenljivke. Npr. » **int number;** « deklarira spremenljivko »number« tipa »int«. Komentarji se uporabljajo za dokumentiranje kode in so neizvedljivi. Enovrstični komentarji se začnejo z //, večvrstični pa z /\* in končajo z \*/. V C# je potrebno konec stavka označiti s podpičjem. Stavki so združeni z zavitimi oklepaji. [17]

#### **8.2 VISUAL STUDIO**

Visual Studio je razširjeno in močno integrirano razvojno okolje (IDE), ki ga je razvil Microsoft. Je ključno orodje za programerje, ki razvijajo aplikacije za Windows, spletno in mobilno platformo. Njegove glavne značilnosti so:

- Podpora za več jezikov Visual Studio podpira številne programske jezike, vključno s C#, VB.NET, C++, F#, JavaScript, Python idr., kar omogoča razvoj v različnih tehnologijah.
- **Razširljivost** ponuja bogat nabor vtičnikov in razširitev, ki jih lahko uporabniki dodajajo za izboljšanje funkcionalnosti ter za prilagoditev delovnega okolja.
- **Razhroščevanje in diagnostika** Visual Studio ima napredne možnosti za odpravljanje napak in diagnostiko, kar olajša iskanje ter odpravljanje težav v kodi.
- Integracija z Microsoftovimi tehnologijami tesno je integriran z drugimi Microsoftovimi izdelki in storitvami, kot so Azure, SQL Server, Office idr., kar omogoča enostaven razvoj ter uvajanje aplikacij v ekosistemu Microsoft.
- Vgrajena orodja za razvoj spletnih in mobilnih aplikacij vključuje orodja za razvoj spletnih aplikacij (kot je ASP.NET) in mobilnih aplikacij (z uporabo Xamarin).
- Različice na voljo je v več različicah, vključno z brezplačno različico Visual Studio Community, pa tudi plačljivimi različicami, kot sta Professional in Enterprise, ki nudita dodatne funkcije.
- Integracija z različnimi platformami za nadzor različic podpira integracijo z Git, TFS in drugimi orodji za nadzor različic, kar omogoča enostavno upravljanje projektov in sodelovanje v ekipi.
- Intuitivno uporabniški vmesnik dobro zasnovan in prilagodljiv uporabniški vmesnik, ki olajša navigacijo ter izboljšuje produktivnost razvijalcev.
- **Razvojno okolje, ki temelji na projektnih mapah** omogoča enostavno upravljanje ter organizacijo kod in datotek v projektih.
- **Podpora za oblačne storitve** omogoča razvoj in uvajanje aplikacij v oblaku, zlasti z integracijo s platformo Microsoft Azure.

Visual Studio je zelo priljubljen med razvijalci zaradi svoje vsestranskosti, močnih funkcij in tesne integracije z drugimi Microsoftovimi izdelki. [14], [19]

#### 8.3 OSNOVE SKRIPTIRANJA V UNITY

Ko smo ustvarili novo skripto v Unity urejevalniku, se nam je odprlo razvojno okolje v Visual Studiu. Osnovni program je viden na sliki 32.



Slika 33: Osnovni program (Vir: Osebni arhiv)

Direktiva »Using« je omogočala uporabo tipov, ki so definirani v določenem imenskem prostoru. V osnovnem programu uporabljene direktive so:

- »System.Collections«, ki je vključeval različne strukture podatkov, ki jih je možno uporabljati za shranjevanje in manipulacijo zbirke podatkov.
- »System.Collections.Generic« omogoča dostop do generičnih različic kolekcijskih razredov.
- »UnityEngine« omogoča dostop do Unityjevih razredov in komponent, ki so potrebni za interakcijo z igralnim okoljem (transformacije, fizika predmetov, rendering ...).

Osnovna metoda v programu se imenuje »public class NewBehavioutScript: MonoBehaviour« in je definicija javnega razreda, v tem primeru poimenovanega »NewBehavioutScript«. S tem razredom lahko uporabnik doda skripto kateremukoli objektu v Unityjevem urejevalniku.

Funkcija »Start« v Unity okolju je posebna metoda, ki se izvede enkrat ob začetku igre ali ko je igralni objekt prvič ustvarjen. Prvi del imena metode, »void«, nakazuje, da metoda ne vrača nobene vrednosti po svojem izvajanju. To pomeni, ko kličemo »Start«, ne pričakujemo, da bi dobili katero koli povratno informacijo ali rezultat, kot bi to storili pri funkciji, ki vrača npr. celoštevilsko vrednost (»int«). Metoda »Start« je v bistvu namenjena začetni nastavitvi ali inicializaciji stanja objekta, kot so nastavljanje začetnih spremenljivk, zagotavljanje, da so potrebni viri najdeni, ali izvajanje začetnih izračunov, ki so potrebni pred začetkom igre. [19]

#### 8.4 EKSPLOZIJSKI POGLED ROBOTA IN NASTAVITVE COLIDERJEV

Omenjeni program je bil uporabljen za aktivacijo animacije, ki je eksplozijski pogled modela. Ta razporedi dele robota po prostoru, da se bolje vidijo, in sicer ob pritisku na gumb 1 in za vrnitev modela v izvirno pozicijo ob pritisku na gumb 2.

Spremenljivke, ki so uporabljene v programu, so »robotAnimator«, ki je referenca za prej predstavljeno Animator komponento v Unity urejevalniku, in »animationLenght«, ki je časovna dolžina animacije, predstavljena v sekundah. V programu smo ustvarili svoje metode, ki odločajo o ravnanju programa. Metoda »SproziAnimacijo« se aktivira ob pritisku gumba 1, naredi pa naslednje:

- Omogoči Animator komponento.
- Sproži animacijo »Explode« z uporabo ukaza »SetTrigger«.
- Zamrzne animacijo na zadnji sliki, kar zadrži model v eksplozijskem pogledu.
- Kliče metodo »EnableColliders«, kar omogoči vse Colliderje, ki so bili za začetku programa onemogočeni. To dovoljuje dogodke ob dotiku posameznih delov robota.



Slika 34: Metoda "SproziAnimacijo" (Vir: Osebni arhiv)

Drugo metodo smo poimenovali z »ResetToNormalViewAndDisableColliders«. Ta metoda se aktivira ob pritisku na gumb 2 in:

- Ponovno omogoči »Animator« komponento.
- Sproži animacijo »Reset« z uporabo ukaza »SetTrigger«.
- Kliče metodo »DisableColliders«, kar onemogoči možnost dotika s posameznimi deli.

18	public void ResetirajAnimacijo()
19	{ {
20	<pre>robotAnimator.enabled = true;</pre>
21	<pre>robotAnimator.SetTrigger("Reset");</pre>
22	<pre>DisableColliders();</pre>
23	3

Slika 35: Metoda "ResetirajAnimacijo" (Vir: Osebni arhiv)

Metodi »DisableColliders« in »EnableColliders« vpogledata v vse Colliderje na delih robota z ukazom »GetComponentsInChildren<Collider>()« in jih glede na potrebo omogočita ali onemogočita.



Slika 36: Metodi "DisableColliders" "EnableColliders" (Vir: Osebni arhiv)

# 8.5 PRIPRAVA KOMPONENT

Po pripravljenem programu smo v hierarhiji izbrali glavnega starša našega robota in v njegovem inspektorju dodali ustvarjeno skripto z imenom »razdren\_pogled«. V okno »Robot animation« smo vnesli »Animator« našega robota.

Inspector		а	:
mrc_robot		atic	
Tag Untagged   Layer Default			
🔻 🙏 Transform	0		
Position X 0 Y 0.914 Z 2	2		
Rotation X 0 Y -90 Z 0	)		
Scale 🔍 X 1 Y 1 Z 1			
▶ 😚 🗹 Box Collider	0		
🕨 井 🔽 Object Manipulator (Script)	0		
🕨 🗯 🗹 NearInteractionGrabbable	0		
# Constraint Manager (Script)	0		
► ≻ ✓ Animator	0		
# Explode View (Script)	0		
Script I razdren_pogled			۲
Robot Animator >> rmrc_robot (Animator)			$\odot$
Animation Length 3			
▶ # ✓ Disable Colliders (Script)	0		
Add Component			

Slika 37: Lastnosti glavnega starša (Vir: Osebni arhiv)

V sceno smo dodali dva gumba, ki odločata o poziciji robota. Najprej smo pod zavihkom »Window« izbrali možnost »Toolkit«, nato »Toolbox«.

Mixe	d Reality Window Help			
	Toolkit	>	Utilities	>
	Project Validation Settings	>	Toolbox	
	Remoting	>	Add to Scene and Configure	
			Help	>
*			MRTK Figma Bridge	>
- ¢‡⇒			MSBuild	>
5			Show version	
2			Show version	

Slika 38: Meni Mixed Reality Toolkit (Vir: Osebni arhiv)

To je odprlo okno »MRTK Toolbox«, ki je MRTK orodna vrstica, v kateri imamo že vnaprej pripravljene igralne objekte, kot so različne vrste gumbov, drsnikov in ostalih uporabniških vmesnikov. Izbrali smo poljubna gumba in ju vstavili v sceno. V inspektorju, ki pripada enemu od ustvarjenih gumbov, ki smo ga poimenovali gumb 1, smo v komponento »Interactable«, ki določa katera skripta bo aktivirana ob pritisku, pod možnost »Events«, vstavili starša našega robota in v izbirni vrstici izbrali »ExplodeView«, nato pa še »SproziAnimacijo«.

							Ъ	;
gumb1							atic	_
Tag Untagged		▼ Lay	er Del	fault				
🔻 🙏 🛛 Transform						0		
Position		x -0.319	Y -0	.05	ΖC	0.041	500	09
Rotation		X O	Y 0		ΖC			
Scale	8	X 5	Y 3.	75	Z 1			
🕨 🍞 🖌 Box Collider						0		
🕨 # 🗹 PressableButton	HoloLer	าร2				0		
# PhysicalPressEv	entRout	er				0		
🔻 🗰 🖌 Interactable						Ø		
General				Doci	umer	itatio		
States		& FocusPress	TouchG	FrabStates	(Stat	es)		⊙
Enabled		~						
Input Actions		Select						
Is Global								
Voice Command		Select						
Selection Mode		Button						
Profiles								
Target		Q CooltCoultI o	hol				പ	
► Theme			aaltSa	viti abol (Ti	ame			•
	4011010	graphicoattono	centoa	yncabel (H	ieme			
Target								-
▶ Theme	\$%Pres	sableButtonFror	ntPlate	(Theme)				0
Target		⊕UlButtonSqu	arelco	n			0	
▶ Theme	ℜPres	sableButtonIcor	n (Ther	ne)				0
		Add Profile						
▼ Events								
OnClick ()								
Runtime Only	▼ Exp	lodeView.Sproz	iAnima	cijo				
rmrc_robot (Explc	0							
						+	-	
Deceivers								
- Receivers								
🕨 텍 🖌 Audio Source						0		
🕨 🗯 🗹 NearInteractionT	ouchab	le				0	같	
🕨 🛱 🗹 Button Config He	elper (So	cript)				0	1	
		Add Component						

Slika 39: Lastnosti objekta gumb 1 (Vir: Osebni arhiv)

Enako smo storili za gumb 2, le da smo namesto »SproziAnimacijo« izbrali »ResetirajAnimacijo«.



Slika 40: Lastnosti objekta gumb 2 (Vir: Osebni arhiv)

### **9 GRADNJA IN PRENOS**

#### 9.1 UNITY BUILD

»Unity build« je način programske opreme, ki združuje vse »Assetse«, »Scripte« ... v skupno izvršljivo datoteko, kar nam je omogočalo hitro nalaganje aplikacij na različne platforme. »Unity build« zagotavlja boljše delovanje, saj optimizira kodo, tako da odstrani odvečne podatke, ki za program niso potrebni, in združuje povezane funkcije v samostojna navodila, ki se jim reče »chunks«. Te procesorji izvršijo veliko bolje. To pripomore k hitrejšemu nalaganju in izboljša hitrost sličic ob uporabi aplikacije.

Končano aplikacijo smo uvozili na Hololense, tako da smo v Unityjevih »build settingsih« nastavili pravilne parametre, ki so ustrezali napravi, na katero bomo aplikacijo naložili in vnesli potrebno.

Architecture	Omogoča izbiro ciljne CPU-arhitekture za našo gradnjo.
Build Type	Izberemo vrsto gradnje programa v Visual Studiu.
Target SDK Version	To kaže na verzijo kompleta za razvoj programske opreme
	(SDK), na katero cilja aplikacija.
Minimum Platform Version	Najnižja verzija platforme, na kateri lahko aplikacija teče.
Visual Studio Version	Verzija Visual Studia, v katerem bomo gradili.
Build and Run on	Ta možnost vam omogoča izbiro, kje graditi in izvajati
	projekt.
Build configuration	Določa vrsto konfiguracije gradnje, kot sta Debug ali Release
	(razhroščevanje ali izdaja).
Device Portal Adress	Lokalni naslov za portal naprave.
Device Portal Username	Uporabniško ime za prijavo v portal naprave.
Device Portal Password	Polje za geslo za portal naprave.

#### Tabela 1: Možne nastavitve



Slika 41: Nastavitve gradnje (Vir: Osebni arhiv)

Ob pritisku na »build« se nam je odprl raziskovalec, v katerem smo določili, v kateri mapi se bo želena aplikacija zgradila. Ob koncu njene gradnje, kar lahko traja nekaj minut, se nam je izbrana mapa odprla. V primeru, ko smo razvijali aplikacijo za Hololense, so bile v mapi tri dodatne mape, ki so vsebovale komponente aplikacije (assets ...), ».prop« datoteko in »ime\_projekta.sln«. To je bila datoteka, ki smo jo morali zagnati, če smo želeli aplikacijo naložiti na Hololense. ».lns« (Visual Studio solution file) je bila datoteka, ki je vsebovala besedilne informacije, ki jih je program uporabljal za iskanje in nalaganje parametrov, nato se je odprl v Visual Studiu.

>	Bements (D:) > builds-unity > holo1.4 >						
ti 🖈	↑ Ime	Datum spremembe	Vrsta	Velikost			
*	build	27. 02. 2024 16:39	Mapa z datotekami				
	Holo-1	27. 02. 2024 16:46	Mapa z datotekami				
	II2CppOutputProject	27. 02. 2024 16:46	Mapa z datotekami				
na	Holo-1.sin	27. 02. 2024 16:37	Visual Studio Solu	8 KB			
	UnityCommon.props	27. 02. 2024 16:37	Project Property File	1 KB			

Slika 42: Ustvarjena mapa (Vir: Osebni arhiv)

#### 9.2 VISUAL STUDIO PRENOS

Preden lahko aplikacijo prenesemo, moramo na računalniku in na Hololensih v nastavitvah vklopiti način za razvijalce. To nam dovoljuje nameščanje aplikacij z naprave a na napravo b. Visual studio nam omogoča izbiro načina nalaganje aplikacije, in sicer jo lahko naložimo preko kabla ali preko internetne povezave. V prvem primeru morajo biti Hololensi vklopljeni v računalnik, za način nalaganja izberemo »device«. Za nalaganje preko internetne povezave je pogoj, da sta obe napravi, torej računalnik, preko katerega nalagamo, in Hololensi, na istem omrežju ter da se nobena izmed naprav med procesom ne zaklene, saj bi to povzročilo napako in prenehalo s prenosom. Nato si nadenemo Hololense in izgovorimo »what is my ip adress«. To nam bo prikazalo lokalni naslov Hololensov v našem omrežju. Na meniju Visual Studia poiščemo zavihek »Project« in v njem izberemo možnost »Properties«. Odprlo se nam bo okno, v katerem navigiramo do zavihka »Debugging«; v prostor ob »Machine name« vpišemo lokalni naslov Hololensov in potrdimo vnos s tipko »V redu«. Tako bo Visual Studio vedel, na katero napravo nalaga.

nfiguration:	Active(Release)	V Platform: A	ctive(ARM64)	~	Configuration Manager
Configurat Genera Debug	tion Properties I ging	Debugger to launch: Remote Machine			
N C/C++	Directories	Launch Application	Yes		
<ul> <li>b Linker</li> </ul>		Allow Local Network Loopback	Yes		
<ul> <li>Manife</li> </ul>	st Tool	Debugger Type	Native	Only	
> XML D	ocument Generator	Machine Name	192.16	8.1.19	
Browse	Information	Authentication Type	Univer	sal (Unencrypted Protor	col)
▷ Build E	vents	Deploy Visual C++ Debug Runti	ime Librar No		
Custon	n Build Step	Amp Default Accelerator	WARP	software accelerator	
Code A	Analysis	Package Layout Path			
		Advanced Remote Deployment	Туре Сору Т	o Device	
		Package Registration Path			
		Remove Non-Layout Files from	Device No		
		Command Line Arguments			
		Launch Application Specifies whether to launch my app	lication immedia	tely or wait to debug m	y application when it start

Slika 43: Lastnosti omrežnega prenosa (Vir: Osebni arhiv)

Nato smo nastavili pravilne parametre za brezžično nalaganje aplikacije. To smo storili tako, da smo\_pritisnili »release« in izbrali procesor Holo Lens ARM64 ter ob trikotniku »Remote machine«, zaradi prenosa preko omrežja. Ob pripravi vseh parametrov smo aplikacijo prenesli ob kliku na zavihek »Build« in možnost »Deploy Solution«.



Slika 44:Meni za gradnjo aplikacije (Vir: Osebni arhiv)

Ob končanem prenosu se nam je v terminalu izpisalo, da je bil prenos uspešen.



Slika 45: Končno sporočilo (Vir: Osebni arhiv)

### **10 ANKETA**

V sklopu raziskovalne naloge smo opravili tudi anketo o splošnem razumevanju obogatene in virtualne resničnosti. V anketi, ki jo je rešilo 239 anketirancev, od tega 51 % žensk in 49 % moških, smo največ odgovorov dobili iz starostne skupine do 20 let (66 %), 20–40 let jih je bilo 11 %, 19 % vprašanih je bilo starih 40–60 let, najmanj odgovorov pa smo dobili od anketirancev, starejših od 60 let (4 %).

Sledi predstavitev rezultatov ankete.

1. Ali ste že slišali za pojem obogatena resničnost (AR) ali virtualna resničnost (VR)?



Graf 1: Vprašanje 1

Večini oseb, ki so oddale odgovor, sta obogatena in virtualna resničnost znani, medtem ko je več kot tretjini znana samo virtualna resničnost.

2. Kakšna je razlika med obogateno (AR) in virtualno (VR) resničnostjo?



Graf 2: Vprašanje 2

Pravilen odgovor je bil tretji (AR-tehnologija vizualizacijo sveta okoli nas izboljša, VR pa popolnoma nadomesti.); pravilno je odgovorilo več kot dve tretjini vseh anketirancev.



3. Kaj od naštetega spada med izkušnjo obogatene resničnosti?

Pravilni odgovori so bili Snapchat filter, igranje igre Pokemon GO in projiciranje hitrosti na vetrobransko steklo. V tem primeru je prav tako večina vprašanih izbirala pravilne odgovore.

4. Ali ste že kdaj uporabljali kakšno napravo za obogateno (AR) ali virtualno (VR) resničnost?



Graf 4: Vprašanje 4

Anketirancem, ki so odgovorili z da, smo nato zastavili še podvprašanje, katero napravo so uporabljali in največ jih je uporabljalo očala znamke Oculus Quest (26 %).



Graf 5: Podvprašanje 4

5. Katera uporaba AR/VR-tehnologije vas najbolj zanima?





Anketirance je najbolj zanimala uporaba AR- in VR-tehnologije v igrah (28 %), najmanj pa v medicini in zdravstvu (9 %).

6. Ali menite, da AR/VR-tehnologija ponuja edinstvene izobraževalne možnosti?



Dve tretjini anketirancev menita, da tehnologija obogatene in virtualne resničnost ponuja edinstvene izobraževalne možnosti.



7. Ali bi morala biti AR/VR-tehnologija dostopnejša v šolah?

Graf 8: Vprašanje 7

Malo manj kot dve tretjini anketirancev je v anketi podalo mnenje, da sta AR- in VR-tehnologija premalo dostopni. Verjetno je to posledica dejstva, da je slednja precej draga.

Kot vidimo iz odgovorov, je večina anketirancev poznala osnovne podatke o obogateni in virtualni resničnosti. Analiza ankete je pokazala, da vprašani AR- in VR-tehnologijo najraje uporabljajo za zabavo in menijo, da bi dobro prispevala k izobraževalnim možnostim, a je slabo dostopna.

### **11 ANALIZA HIPOTEZ**

Tabela 1: Hipoteze

Hipoteze	Stanje
Robot v AR-aplikaciji bo samodejno navigiral po virtualnih poteh.	Ni potrjeno.
Vključitev robota v AR je preprosta in zahteva minimalno tehnično znanje.	Potrjeno.
Uporabnik s HoloLens in drugi s telefonom lahko sočasno ter interaktivno delujeta v istem AR-okolju.	Ni potrjeno.
Večina anketirancev bo poznala razliko med obogateno in virtualno resničnostjo.	Potrjeno.

#### Robot v AR-aplikaciji bo samodejno navigiral po virtualnih poteh.

Pred razvijanjem aplikacij v Unity smo hoteli robota navigirati po prostoru s štirimi gumbi, da bi lahko s pomočjo vizualizacije videli, kako se bo vozil. To bi precej pripomoglo k razvoju dejanskega robota, saj bi s spremembo v aplikaciji že pred 3D-tiskom nove verzije spremljali, kako ta deluje v okolju. Spoznali smo, da je zaradi izdelave in modela koles samostojna vožnja robota v okolju presegla naše trenutne tehnične zmogljivosti ter obseg raziskave.

#### Vključitev robota v AR je preprosta in zahteva minimalno tehnično znanje.

Ta hipoteza je potrjena, saj kljub temu da se z vnosom objektov v obogateno resničnost prej nismo srečali, je bil ta postopek po nekaj prebranih dokumentacijah in pregledanih videih precej

jasen. Aplikaciji, kot sta Jig Space in Vufora View, dovoljujeta vnos 3D-modelov v obogateno resničnost brez kakršnegakoli predznanja. Unity zahteva boljše tehnično razumevanje in programerske veščine, še posebej pri težjih nalogah, kot sta animacija in interakcija robota v AR. Proces vključevanja in animiranja robota v Unity je bil zato zahtevnejši in daljši.

# Uporabnik s HoloLens in drugi s telefonom lahko sočasno in interaktivno delujeta v istem AR-okolju.

To hipotezo smo zavrgli zaradi pomanjkanja dokumentacije na internetu in zaradi zahtevnosti. Poskušali smo jo uresničiti, da bi uporabnik s prostorskimi sidri (spatial achors), da bi bil uporabik HoloLensa gostitelj prostora, uporabnik mobilne aplikacije pa bi se na isto omrežje z njo prijavil in opazoval dogajanje.

### Večina anketirancev bo poznala razliko med obogateno in virtualno resničnostjo.

Anketa je razkrila, da anketiranci načeloma prepoznavajo razliko med AR in VR, ampak je prav tako pokazala, da je razumevanje razlike med tema dvema tehnologijama različno. Rezultati nakazujejo, da je potrebno nadaljnje izobraževanje in ozaveščanje o teh dveh tehnologijah. To bi pomagalo pri jasnejšem razumevanju njunih razlik in uporabe.

# 12 MOŽNOSTI NADALJNJE RAZISKAVE

### 12.1 IZBOLJŠANA INTERAKTIVNOST V APLIKACIJI

V prihodnje bi lahko aplikacijo izboljšali z razvojem naprednejše metode interaktivnosti robota v obogateni resničnosti. Dodali bi lahko informacijska okna, ki bi se pokazala v primeru, da uporabnik izbere enega izmed delov robota; tako bi razumevanje le-tega še izboljšali.

#### 12.2 INTEGRACIJA UMETNE INTELIGENCE (UI)

Z integracijo UI v AR-aplikacijo bi lahko robot postal bolj avtonomen, se bolje prilagajal in odzival na uporabniške interakcije. UI bi lahko omogočil naprednejše funkcije, kot so prepoznavanje objektov, učenje iz interakcij in celo prilagajanje vedenja robota glede na uporabnikove preference ali zgodovino interakcij.

### 12.3 UPORABNIŠKO TESTIRANJE

Obsežno uporabniško testiranje in zbiranje informacij bi pripomoglo k nadaljnjemu razvoju aplikacije. S tem bi lahko zbrali informacije o uporabniški izkušnji, odkrili težave ali pomanjkljivosti in se osredotočili na izboljšanje aplikacije.

# 13 ZAKLJUČEK

Tema raziskovalne naloge je v nas vzbudila veliko zanimanje, saj mladi o vplivu razširjene resničnosti na razvoj premalo razmišljajo in se ne zavedajo njenih potencialov ter vseh načinov, kako nam lahko izboljša vsakdan.

Skozi raziskovanje smo se srečali s kar nekaj težavami, kot je npr. nekompatibilnost verzij Unity urejevalnika. Naloženih jih je bilo kar nekaj, preden smo aplikacijo končno lahko prenesli na Holo Lense. Prav tako se je bilo potrebno bolj spoznati s programom C#, ki ga v šoli nismo obravnavali. K sreči je zelo podoben programskemu jeziku C++, ki se uporablja v programiranju mikrokontrolerja Arduino.

Glede na analizo ankete se za obogateno in virtualno resničnost zanima precej ljudi. Zanimajo jih predvsem igre, vendar je popularno tudi izobraževanje in virtualna potovanja. Pokazalo pa se je tudi, da vedno ne znajo prepoznati, kaj točno so izkušnje razširjene resničnosti.

Pri raziskovanju naloga ni bila preveč zahtevna, saj je na spletu veliko pomoči za programiranje in snovi, povezane z raziskovalno nalogo. Pogosto je bilo potrebno pogledati malo globje kot le na prvo povezavo, ki je bila podana. Pri raziskovanju so se sproti pojavljale tudi nove informacije, razmišljanja in postavljala nova vprašanja.

Razvoj tehnologije razširjene in virtualne resničnosti bo imel pomemben vpliv na prihodnost. V izobraževanju bodo te tehnologije omogočile interaktivno učenje s 3D-modeli. V poslovnem svetu bodo AR in VR izboljšale sodelovalne procese in omogočale učinkovitejša usposabljanja. V zdravstvu bodo obogatena in virtualna resničnost pripomogli k napredku v kirurškem usposabljanju in daljinskem zdravljenju. Soočiti se bo potrebno tudi z izzivi, kot so zasebnost, kibernetska varnost in etična vprašanja ter prav tako zagotoviti široko dostopnost tehnologij, da ne bo prišlo do prevelike tehnološke neenakosti. Skupno gledano bodo AR in VR obogatile našo interakcijo z digitalnim svetom ter igrale ključno vlogo v nadaljnjem tehnološkem napredku.

#### **14 VIRI**

- [1] AR (Agumented reality) [Članek na spletu], Interaction Design Foundation © 2023, pridobljeno 23. 2. 2024 s https://www.interaction-design.org/literature/topics/augmented-reality.
- [2] Augumented reality [Fotografija na spletu], pridobljeno 15. 2. 2024 s https://www.ecodico.net/wp-content/uploads/2024/02/technology-IT-services-1024x679.jpg
- [3] *AR (Augmented Reality) in njena uporaba* [Članek na spletu], pridobljeno 15. 2. 2024 s https://www.pomagalnik.com/marketing/ar/ar-augmented-reality-in-njena-uporaba/.
- [4] *Agumented reality*, Investopedia, pridobljeno 15. 2. 2024 s https://www.investopedia.com/terms/a/augmented-reality.asp,
- [5] *Agumented reality*, Wikipedija, pridobljeno 15. 2. 2024 s https://en.wikipedia.org/wiki/Augmented\_reality.
- [6] "Getting Started with Unity for Hololens2 | Tutorial" [Video datoteka], pridobljeno 16.2. 2024 s https://www.youtube.com/watch?v=dOsYerpKloY.
- [7] *How Agumented Reality Works*, pridobljeno 7. 2. 2024 s https://computer.howstuffworks.com/augmented-reality.htm.
- [8] How To Setup HoloLens for Unity development in 2022 [Video datoteka], pridobljeno
   19. 2. 2024 s https://www.youtube.com/watch?v=75\_i\_csnA5U
- [9] Jig Space, pridobljeno 18.2.2024 s https://golden.com/wiki/Jigspace-NMGKJPY
- [10] Microsoft HoloLens 2 [Fotografija na spletu], Fireware © 2022, pridobljeno
   15. 2. 2024 s https://www.fireware.nl/wp-content/uploads/2022/06/021-021-001 Microsoft-HoloLens-2.jpg
- [11] Mixed Reality Toolkit 3: Dokumentacija za razvijalce [Učbenik na spletu], Microsoft © 2024, pridobljeno 19. 2. 2924 s https://learn.microsoft.com/enus/windows/mixed-reality/mrtk-unity/mrtk3-overview/
- [12] Navodila za zagon in izdelavo aplikacij v Unity [Učbenik na spletu], Unity Technologies © 2024, pridobljeno 17. 2. 2024 s https://docs.unity3d.com/2021.3/Documentation/Manual/30\_search.html?q=build+and +run

- [13] Stp to Blender CAD conversion using FreeCAD free 3D software [Video datoteka], pridobljeno 24. 2. 2024 s https://www.youtube.com/watch?v=j8WefhRmeQ8.
- [14] Upravljanje virov: Uvoz in konfiguracija Usposabljanje [Učbenik na spletu], Microsoft © 2024, pridobljeno [datum pridobitve] s https://learn.microsoft.com/enus/training/modules/learn-mrtk-tutorials/1-4-configure-resources.
- [15] Uporaba Visual Studia za uvajanje in odpravljanje napak Mešana resničnost [Učbenik na spletu], Microsoft © 2024, pridobljeno [datum pridobitve] s https://learn.microsoft.com/en-us/windows/mixed-reality/develop/advancedconcepts/using-visual-studio?tabs=hl2#deploying-an-app-to-the-hololens-2-emulator.
- [16] Unity (igralni pogon) [Članek na spletu], Wikipedia © 2024, pridobljeno[datum prid15.
- [17] Uvod v C# [Učbenik na spletu], W3Schools © 2023, pridobljeno [datum pridobitve] s https://www.w3schools.com/cs/cs\_intro.php,
- [18] Virtual reality, Wikipedija, pridobljeno 22. 2. 2024 s https://en.wikipedia.org/wiki/Virtual\_reality
- [19] Visual Studio [Učbenik na spletu], Microsoft © 2024, pridobljeno 8. 2. 2024 s https://learn.microsoft.com/en-us/visualstudio/get-started/visual-studio-ide?view=vs-2022
- [20] What is extended reality? (XR, VR, AR, MR) [Video datoteka], pridobljeno
   18.2.2024 s https://www.youtube.com/watch?v=O6KWWLnoFfI,
- [21] *What is virtual reality?*, Techtarger, pridobljeno 22. 2. 2024 s https://www.techtarget.com/whatis/definition/virtual-reality.