

Kauno technologijos universitetas

Informatikos fakultetas

Čia rašyti praktikos temą iš programos

Profesinės praktikos ataskaita

|  |  |
| --- | --- |
|  |  |
| Erikas Mackevičius  Studentas | (parašas)  (data) |
|  |  |
| Andrius Paulauskas  Vadovas | (parašas)  (data) |
|  |  |
| Doc. Giedrius Ziberkas  Dėstytojas | (parašas)  (data) |
|  |  |

Kaunas 2022

Turinys

[Įvadas 4](#_Toc92103882)

[1. Analizės dalis 5](#_Toc92103883)

[1.1. Informacija apie įmonę 5](#_Toc92103884)

[1.1.1. Įmonės veiklos sritis, specializacija 5](#_Toc92103885)

[1.1.2. Įmonės vizija 5](#_Toc92103886)

[1.1.3. Įmonės misija 5](#_Toc92103887)

[1.1.4. Kokiais būdais įmonė siekia įgyvendinti savo misiją ir viziją 5](#_Toc92103888)

[1.2. Informacija apie įmonėje naudojamas IT 5](#_Toc92103889)

[1.3. Įmonės veiklos procesai. 5](#_Toc92103890)

[1.3.1. Projekto inicijavimas. 5](#_Toc92103891)

[1.3.2. Projekto koncepcijos analizė. 5](#_Toc92103892)

[1.3.3. Kainos ir investicijų analizė. 5](#_Toc92103893)

[1.3.4. Projektų vykdymas, pristatymas ir dokumentacija. 5](#_Toc92103894)

[1.3.5. Komandinis darbas. 5](#_Toc92103895)

[1.3.6. Inovacijos. 5](#_Toc92103896)

[1.3.7. Produktų vystymo procesai. 5](#_Toc92103897)

[2. Praktikos dalis 6](#_Toc92103898)

[2.1. Techninė užduotis 6](#_Toc92103899)

[2.1.1. Sistemos apibrėžimas 6](#_Toc92103900)

[2.1.2. Bendras veiklos tikslas 6](#_Toc92103901)

[2.1.3. Sistemos pagrįstumas 6](#_Toc92103902)

[2.1.4. Konkurencija rinkoje 6](#_Toc92103903)

[2.1.5. Prototipai ir pagalbinė informacija 6](#_Toc92103904)

[2.1.6. Siūlomas sprendimas 6](#_Toc92103905)

[2.1.7. Sistemos apimtis ir ištekliai, reikalingi sistemai sukurti 6](#_Toc92103906)

[2.2. Reikalavimų specifikacija 6](#_Toc92103907)

[2.2.1. Komercinė specifikacija 6](#_Toc92103908)

[2.2.2. Sistemos funkcijos 6](#_Toc92103909)

[2.2.3. Apribojimai 6](#_Toc92103910)

[2.2.4. Duomenų ir rezultatų specifikacija 6](#_Toc92103911)

[2.2.5. Vartotojo sąsajos specifikacija 6](#_Toc92103912)

[2.2.6. Realizacijai keliami reikalavimai. 7](#_Toc92103913)

[2.2.7. Techninė specifikacija. 7](#_Toc92103914)

[2.3. Sistemos projektavimo eiga ir jos projektas 7](#_Toc92103915)

[2.3.1. Projektavimo valdymas ir eiga 7](#_Toc92103916)

[2.3.2. Projektavimo technologija 7](#_Toc92103917)

[2.3.3. Programavimo kalbos, derinimo, automatizavimo priemonės, operacinės sistemos 7](#_Toc92103918)

[2.3.4. Informacijos srautai 7](#_Toc92103919)

[2.3.5. Sistemos sudėtis 7](#_Toc92103920)

[2.3.6. Komponentai 7](#_Toc92103921)

[2.3.7. Sąsajos įvertinimas pagal vartotojo patyrimą, profesinę terminologiją 7](#_Toc92103922)

[2.3.8. Duomenų kontrolė 7](#_Toc92103923)

[2.4. Testavimo eiga ir rezultatai 7](#_Toc92103924)

[2.5. Dokumentacija 7](#_Toc92103925)

[3. Refleksija 8](#_Toc92103926)

[Išvados 9](#_Toc92103927)

[Literatūra 10](#_Toc92103928)

Įvadas

Šios praktikos užduotis buvo sukurti virtualios realybės simuliaciją, skirtą apmokyti vartotojus aparatinės įrangos apjungimo ypatybių. Sistema buvo orientuota specifiškai į transliavimo aparatinės įrangos scenarijus. Galutinis tikslas buvo turėti veikiančią ir patogią virtualios realybės simuliaciją, kuri leistų vartotojui santykiauti su aplinka ir įvairiais joje esančiais įrenginiais, juos apjungti pagal tam tikrus scenarijus ir pasitikrinti ar apjungimas buvo teisingas. Praktika atlikta Kauno Technologijų Universitete, specifiškai Virtualios ir papildytos realybės laboratorijų skyriui. Kauno Technologijų universitetas yra plačios aprėpties universitetas, glaudžiai bendradarbiaujantis su įvairiomis verslo įmonėmis. Virtualios realybės laboratorijų skyriaus veikla daugiausia susijusi su virtualios ir papildytos realybės įrenginiais, bei virtualių aplikacijų ir projektų kūrimu.

# Analizės dalis

## Informacija apie įmonę

### Įmonės veiklos sritis, specializacija

KTU – lyderiaujantis plačios aprėpties universitetas, glaudžiai bendradarbiaujantis su verslu. Universitete vykdomos įvairių krypčių studijos ir moksliniai tyrimai. Universitetas stengiasi formuoti profesionalius, savo sritį išmanančius ekspertus, kurie turėtų galimybę plėsti savo darbo sritį ir turėti įtaką visuomenėje. Universitetas turi plačius žinių ir resursų šaltinius, prieinamus studentams, suteikia praktikos galimybes, bei skatina verslumą.

### Įmonės vizija

Tarptautinis informacinių technologijų studijas ir mokslinius tyrimus vykdantis centras, pripažintas Baltijos jūros regione. [[1]](#footnote-1)[1]

### Įmonės misija

Teikti moksliniais tyrimais grįstas tarptautinio lygio studijos, prisidedant prie tvaraus visuomenės ugdymo, kuriant bei perduodant žinias ir novatoriškas technologijas darniai valstybės raidai ir inovacijų plėtojimui. [1]

### Kokiais būdais įmonė siekia įgyvendinti savo misiją ir viziją

KTU - rūpestinga bendruomenė, kuri padeda augti vienas kitam akademinėje aplinkoje, plečiant žinias ir žmogaus supratimą. Universitetas palaiko ryšius ir bendradarbiauja su įvairiomis įmonėmis visoje Lietuvoje, į paskaitas dažnai kviečiami įvairių sričių ekspertai. KTU „WANTed“ programa suteikia studentams galimybes rasti praktikos bei darbo vietas, gauti konsultacijas iš įvairių įmonių, planuoti savo studijas atsižvelgiant į ateities karjeros galimybes.

## Informacija apie įmonėje naudojamas IT

KTU svarbu inovacijos, naujausių technologijų kūrimas, įvairių sričių moksliniai tyrimai. Universitetas turi ryšius su įvairiais studijų, mokslo ir verslo centrais, verslo įmonėmis. Dėl šios priežasties universitetas turi prieigą prie naujausių ir moderniausių informacinių technologijų, bei išteklių.

## Įmonės veiklos procesai.

### Projekto inicijavimas.

KTU projektų iniciavimas priklauso nuo jų pobūdžio, bet visais atvejais organizuojami susitikimai su partneriais ir klientais, aptariama su projektu susijusi informacija, tokia kaip projekto laikotarpis, finansavimas, reikalavimai.

### Projekto koncepcijos analizė.

Po pirmųjų susitikimų su klientais atliekama projekto apimties analizė, kurios metu yra įvertinami turimi resursai bei technologijų galimybės sukurti norimą produktą. Vyksta išsamios diskusijos su klientais, sudaroma projekto vizija.

### Kainos ir investicijų analizė.

Universitetas retai kada investuoja į išorinio produkto sukūrimą, nebent projektas to reikalauja.

### Projektų vykdymas, pristatymas ir dokumentacija.

Projektų vykdymas susideda iš etapų:

* Reikalavimų surinkimas ir analizė;
* Programinės įrangos kūrimas;
* Programinės įrangos dokumentacija (kuriama projekto pabaigoje arba iteracijomis);
* Projekto pristatymas užsakovui ir atlikimo įvertinimas.

### Komandinis darbas.

Komandiniam darbui organizuoti naudojami įvairūs planavimo įrankiai. Slack ir Trello dažniausiai naudojami užduočių paskirstymui ir atlikimo sekimui. Projektai dažniausiai saugomi repozitorijose naudojant Github arba Bitbucket.

### Inovacijos.

Inovacijos, jų kūrimas bei skatinimas yra svarbi universiteto dalis. KTU inovacijomis laiko ne tik naujų sprendimų ir technologijų kūrimą, bet ir inovatyvių verslo idėjų skatinimą ir plėtojimą. Pačiame KTU naudojami tik naujausi technologiniai sprendimai ir įranga. Universitetas glaudžiai seka naujų inovatyvių idėjų ir technologijų plėtrą ir stengiasi jas integruoti į savo veiklą. KTU svarbu ne tik naudoti inovatyvius sprendimus, bet ir juos kurti bei plėtoti. KTU Nacionalinis inovacijų ir verslo centras rūpinasi tarptautiniu lygiu pripažintų technologijų kūrimu bei inovatyvius produktus kuriančių įmonių steigimu ir vystymu. Šio centro dėka egzistuoja tokios priemonės kaip KTU Startup Space ir KTU Atviros prieigos centras. Startup Space yra plačiai visuomenei atvira bendruomenė, skatinanti inovatyvių idėjų ir verslo plėtojimą. Atviros prieigos centras yra įrankis, leidžiantis verslo įmonėms ir įstaigoms prieiti KTU mokslininkų teikiamų paslaugų, bei tyrimų informacijos, kuri palengvina naujų technologinių sprendimų kūrimą.

### Produktų vystymo procesai.

Universitete yra taikomi įvairūs produktų vystymo procesai. Tai dažniausiai priklauso nuo projekto pobūdžio ir reikalavimų. Dirbant su europiniais projektais ir valstybiniais užsakymais, dažniausiai taikomi procesai panašūs į krioklio metodą. Jeigu produktai kuriami įmonėms, dažniausiai taikomas iteracinis produkto vystymo modelis. Nepriklausomai nuo metodo, procesai visada įtraukia reikalavimų surinkimą ir analizę, projektavimą, programinės įrangos kūrimą ir jos testavimą. Programinės įrangos palaikymas dažniausiai nėra vykdomas, jeigu sukurtas projektas yra išorinis.

# Praktikos dalis

## Techninė užduotis

### Sistemos apibrėžimas

Su „Unity3D“ varikliu sukurti virtualiosios realybės programą skirtą „Oculus Quest“ platformai. Sistemos turėtų realizuoti mokomuosius scenarijus, kurių metu vartotojas galėtų išmokti teisingai sujungti garso ir vaizdo transliavimo įrangą. Programoje turi būti realizuotas mokomasis scenarijus, kuriame naudotojas turi teisingai paruošti transliavimo įrangą. Šią įrangą turi sudaryti vaizdo kamera, transliavimo pultas, mikrofonas, garso monitorius bei kompiuterinė įranga (kompiuteris, monitorius, įvesties įrenginiai).

### Bendras veiklos tikslas

Sukurti programą virtualiai realybei, kurioje būtų realizuojamas transliavimo įrangos komutavimo mokomasis scenarijus. Programa leis paruošti komutavimo scenarijus nereikalaujant fiziškai turėti aparatinės įrangos. Tai padės taupyti kaštus ir tuo pačiu išmokyti vartotoją komutavimo.

### Sistemos pagrįstumas

Poreikis mokyti studentus apie transliavimo įrangą ir jos pajungimo ypatybes yra gana didelis, bet gauti visą įrangą ir ją dedikuoti mokymui yra sudėtinga ir neproduktyvu. VR programos paruošimas leidžia suteikti studentams informaciją apie komutavimą interaktyvioje aplinkoje, bet nereikalauja tikros įrangos.

### Konkurencija rinkoje

* **Unimersiv** – virtualios realybės programa, leidžianti vartotojui pasirinkti vieną iš daugelio interaktyvių mokomųjų simuliacijų. Programos naudotojai gali pasirinkti iš daugelio kategorijų, tokių kaip istorija, kosmosas, anatomija.
* **Veative** – virtualios realybės platforma, turinti vieną iš didžiausių virtualių scenarijų bibliotekų. **Veative** biblioteka šiuo metu turi 700+ scenarijų iš įvairių mokslo šakų, virtualių turų, prezentacijų.
* **Engage** –virtuali platforma, leidžianti organizacijoms kurti virtualias aplinkas ir scenarijus skirtas mokymo tikslams, darbuotojų parengimui ir kitiems scenarijams.

Visos priemonės leidžia vartotojas patirti įvairius mokomuosius scenarijus, bet turi įvairių trūkumų. **Unimersiv** leidžia vartotojui išbandyti įvairius mokomuosius scenarijus, bet suteikia scenarijus tik tam tikrose kategorijose. Šios programos pagrindinis dėmesys skiriamas simuliacijoms istorijos, kosmoso, bei anatomijos kategorijose. **Veative** turi didelį scenarijų pasirinkimą, suteikia vartotojui galimybe patirti scenarijus ir eksperimentus biologijos, chemijos, fizikos bei matematikos srityse. Ši priemonė taip pat suteikia vartotojui galimybe mokytis kalbų naudojant virtualius pokalbių scenarijus, bei aplankyti įvairias pasaulio vietas naudojant turų simuliacijas. **Engage** neturi tam tikrų nustatytų scenarijų, bet suteikia galimybę organizacijoms kurti naujus scenarijus naudojantis jų įrankiu. Tai leidžia kompanijoms kurti įvairius scenarijus pagal paskirtį. Visos šios kompanijos suteikia galimybę vartotojams patirti įvairius mokomuosius scenarijus, bet **Unimersiv** yra apribota kategorijų stokos, **Veative** taip pat ribota kategorijų stokos ir ne visos simuliacijos suteikia vienodą kiekį interaktyvumo. **Engage** scenarijai gali būti įvairių kategorijų, bet jie yra riboti to, jog kiekvienas scenarijus kuriamas naudojant **Engage** įrankiais, todėl jeigu **Engage** sistema neturi tam tikrų funkcijų, nėra galimybės scenarijuose įgyvendinti tam tikrų funkcijų. Kuriama virtualios realybės įrenginių sujungimų sistema susitelkia į vieną kategoriją, prietaisų sujungimą ir validaciją virtualioje erdvėje. Nė viena iš aptartų sistemų nesiūlo būtent tokio tipo scenarijų, **Engage** leistų vartotojui sukurti panašų scenarijų, bet jis būtų ribotas dėl reikiamybės naudoti **Engage** įrankius.

### Prototipai ir pagalbinė informacija

Kuriant programą nenaudojami joki prototipai.

### Siūlomas sprendimas

Sprendimas bus pritaikytas konkretiems scenarijams, kurie skirti mokyti studentus įrangos komutavimo. Programos eigoje, studentai gaus informacijos apie įvairius transliavimo įrenginius, jų tarpusavio sąveiką ir apjungimą.

### Sistemos apimtis ir ištekliai, reikalingi sistemai sukurti

Sistema kuriama vieno programuotojo. Jai skiriama apie 260 valandų. Kūrimui naudojamas paprastas darbinis kompiuteris, pajėgus efektyviai naudoti Unity3D aplikaciją bei Oculus Link. Programos testavimui reikalingi virtualios realybės akiniai Oculus Quest.

1 lentelė Minimalūs Oculus Link reikalavimai

|  |  |
| --- | --- |
| **Komponentas** | **Rekomenduojamos minimalios specifikacijos** |
| Procesorius | Intel i5-4590 / AMD Ryzen 5 1500X |
| Vaizdo plokštė | NVIDIA GeForce GTX 970 / AMD 400 Series |
| Atmintis | 8 GB+ RAM |
| Operacinė sistema | Windows 10 |
| USB jungtys | 1x USB jungtis (rekomenduojama 3.0) |

## Reikalavimų specifikacija

### Komercinė specifikacija

* Projekto užsakovas: Andrius Paulauskas
* Projekto vykdytojas: Erikas Mackevičius
* Produkto vartotojai: dėstytojai ir studentai
* Produkto kūrimo laikotarpis: 2 mėnesiai
* Skiriamas biudžetas: Produkto kūrimui biudžetas neskiriamas
* Pagrindinės priemonės:
  + Kompiuteris su Unity3D ir Oculus programine įranga
  + Oculus Quest virtualios realybės akiniai
  + Oculus Link laidas

### Sistemos funkcijos

Sistemos funkciniai reikalavimai:

* Realizuota virtuali aplinka leidžianti vartotojui judėti, naudoti teleportacijos funkciją ir sąveikauti su objektais;
* Realizuota virtuali įranga su skirtingo tipo jungtimis, bei sistema leidžianti sujungti įrangą tarpusavyje naudojant laidus;
* Realizuota sistema skirta validuoti įrangos sujungimui. Sistema leidžia sudaryti vertinimo kriterijų konfigūraciją, ją validuoja ir sudaro rezultatų suvestinę;
* Sukurti trimačiai aplinkos ir įrangos moduliai;
* Realizuota dalelių sistema;
* Paruošti ir integruoti garso efektai.

### Apribojimai

Sistema kuriama naudoti Oculus Quest ir Oculus Quest 2 operacinėje sistemoje. Darbo aplinkai reikalaujamas darbinis kompiuteris, pajėgus paleisti Unity3D ir Oculus Link. Sistemos testavimui reikalingi Oculus Quest virtualios realybės akiniai. Sistemos kūrimo terminas – 2 mėnesiai. Sistemos kūrimui nėra skiriamas joks biudžetas, todėl kuriant naudojami nemokami įrankiai ir kiti resursai.

### Duomenų ir rezultatų specifikacija

Aplikacijoje duomenys skaidomi į jungties informaciją ir rezultatus. Jungties informacija bus naudojama tikrinant jungčių tipus, jų sujungimo galimybes ir komutacijos įrangos specifikacijas. Rezultatai bus naudojami atvaizduoti vartotojui komutacijos įrangos sujungimo scenarijaus būseną. Jungtys turės duomenų tipus, kuriuose bus apibrėžtos jų specifikacijos. Jungtis gali turėti pavadinimą, tipą, versiją, papildomus apibūdinimus, tokius kaip galimybė praleisti elektrą arba interneto ryšį. Ši informacija bus naudojama vykdant prietaisų sujungimo validavimą. Rezultatai bus apibrėžti papraščiau. Šiuo atveju svarbu žinoti kurie

2 lentelė Jungties laukų duomenų informacija

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Jungties laukų informacija** | | | | | | | | | | | | | |
| **Pavadinimas** | **Tipas** | **Pavyzdys** | | | | | | | | | | | |
| Name | String | HDMI | USB | | | Audio | | | | DC | | | SDI |
| Type | String | Type A | Type B | | 2.5mm | | 3.5mm | | SD | | HD | | 3G |
| Version | Float | 2.0 | | 2.2 | | | | 3.1 | | | | 3.2 | |
| Is powered | Bool | True | | | | | | False | | | | | |
| Has ethernet | Bool | True | | | | | | False | | | | | |
| Cable diameter | Float | 0.644 | | 0.511 | | | | 0.405 | | | | 0.321 | |

3 lentelė Rezultatų laukų duomenų informacija

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Rezultatų informacija** | | | | | |
| **Pavadinimas** | **Tipas** | **Pavyzdys** | | | |
| Task | String | Task 1 | Task 2 | | Task 3 |
| Status | Float | True | | False | |

### Vartotojo sąsajos specifikacija

Projektas kuriamas virtualiai realybei. Esantis virtualioje aplinkoje vartotojas turės rankas, kurios judės naudojant Oculus Touch valdiklius. Vartotojo aplinka bus kambarys su įvairiais prietaisais, kuriuos jis turės sujungti.

Diagram

Description automatically generated with medium confidence

1 pav. Scenarijaus aplinkos prototipas (pavaizduoti ne visi įrenginiai)

Diagram

Description automatically generated

2 pav. Scenarijaus vaizdas iš vartotojo perspektyvos

Vienintelis elementas reikalaujantis UI integracijos išvestyje bus rezultatų atvaizdavimo sistema.

Graphical user interface

Description automatically generated with medium confidence

3 pav. Rezultatų atvaizdavimo UI prototipas

Įvykdžius arba vykdant komutavimo scenarijų, vartotojas galės atidaryti rezultatų lentelę, kurioje bus išvardintos užduotys ir jų atlikimo būsena. Vartotojas turės galimybę bet kurio metu paleisti scenarijų iš naujo.

### Realizacijai keliami reikalavimai.

* Programa turi veikti 72 Hz darbiniu dažniu, kad nesukeltų vartotojui diskomforto;
* Programa turi būti optimizuota ir turėti kuo mažesnį uždelsimą tarp vartotojo veiksmų realybėje ir jų atvaizdavimo virtualioje aplinkoje;
* Programoje neturėtų būti ryškių arba mirgančių vaizdų, tam kad vartotojui nesukeltų diskomforto.

### Techninė specifikacija.

Programai naudoti reikalingi VR akiniai Oculus Quest arba Oculus Quest 2. Norint programą leisti per kompiuterį, reikalinga Oculus programinė įranga, bei įranga leidžianti prijungti VR akinius prie kompiuterio.

## Sistemos projektavimo eiga ir jos projektas

### Projektavimo valdymas ir eiga

Prieš pradedant dirbti prie projekto buvo aptarti sistemos reikalavimai, programinė ir fizinė įranga, kuri bus naudojama jos kūrimui ir testavimui. Sudarytas planas, kuriame atvaizduoti sistemos kūrimo žingsniai ir jiems įgyvendinti skirtas laikas. Projektavimo metu detaliau aptarta sistemos struktūra, planuojamos naudoti duomenų struktūros ir tipai. Baigus projektavimą, pradėta kurti sistema. Sistemos kūrimo metu glaudžiai remtasi sudarytu darbų planu. Sistemos kūrimo pabaigoje atlikti testai, remiantis sudarytais testo scenarijais.

lentelė Sistemos kūrimo planas

|  |  |
| --- | --- |
| Laiko tarpas | Planuojami atlikti veiksmai |
| 2022-03-22 – 2022-03-29 | Saveikos ir orientavimosi virtualioje realybėje realizavimas:   * Realizuojama teleportavimo funkcija orientavimuisi virtualioje erdvėje; * Paruošiama bazinė žaidimo objekto realizacija, kuri skirta vartotojui sąveikauti virtualioje realybėje. |
| 2022-03-30 – 2022-04-06 | Virtualiosios įrangos ir komutavimo funkcijų realizacija:   * Realizuojamos transliavimo įrangos jungtys ir jų tipai; * Realizuojama jungčių sujungimo galimybė. |
| 2022-04-07 – 2022-04-14 | Komutavimo rezultato vertinimo realizacija:   * Vertinimo kriterijų konfigūracijos realizacija; * Komutavimo atitikimo konfigūracijai validavimas; * Vertinimo rezultatų suvestinės sudarymas ir atvaizdavimas. |
| 2022-04-15 – 2022-04-21 | Programos turinio paruošimas   * Trimačių modelių aplinkai sukūrimas; * Trimačių įrangos modelių sukūrimas; * Dalelių sistemos sukūrimas; * Garso efektų paruošimas. |
| 2022-04-22 – 2022-04-29 | Sistemos testavimas ir defektų taisymas. |

### Projektavimo technologija

Sistemos kūrimo progresui sekti ir sistemai saugoti buvo naudojama Github platforma. Kadangi sistema kuriama vieno žmogaus, nebuvo reikiamybės naudoti dedikuotų užduočių paskirstymo ir sekimo sistemų, tokių kaip Trello ir Jira. Sistemos kūrimo planas kartu su Github repozitorija suteikė pakankamai informacijos sistemos kūrimo organizavimui.

### Programavimo kalbos, derinimo, automatizavimo priemonės, operacinės sistemos

C sharp, Android, Windows, Oculus quest Sistemos kūrimui naudota Unity programinė įranga, kodas rašomas C# programavimo kalba. Unity paleisti naudotas kompiuteris su Windows 10 operacine sistema. Sistemai paleisti ir testuoti naudotas Oculus Quest su Android operacine sistema. Oculus Link programinė įranga naudota susieti virtualios realybės akinius su kompiuteriu, derinimo tikslais.

lentelė Techninė ir programinė įranga naudota sistemos kūrimui ir testavimui

|  |  |  |
| --- | --- | --- |
| Sistemos kūrimo informacija | | |
| Kategorijos | Kūrimui ir derinimui naudotos priemonės | Sistemos paleidimui ir testavimui naudotos priemonės |
| Techninė įranga | Kompiuteris | Oculus Quest virtualios realybės akiniai |
| Programinė įranga | Unity, Oculus Link | - |
| Operacinė sistema | Windows 10 | Android |
| Programavimo kalba | C# | - |

### Informacijos srautai

Sistemos veikimo metu, pagrindiniai informacijos mainai vyksta vykdant jungčių sujungimą su prietaisais. Jungtys sujungimo metu komunikuoja su prietaisų lizdais, tokiais kaip HDMI, USB, Audio ir t.t. Įvykdžius dviejų prietaisų sujungimą, jungties duomenys yra nusiunčiami prietaiso valdymo klasei ir saugomi tol, kol prietaisai neatjungiami vienas nuo kito. Tuo pačiu metu, jungtis nusiunčia sujungimo garso failą, garso valdymo klasei.

Informacijos srautai:

* **Connector** su **Socket –** sujungiant jungtį su lizdu, vyksta informacijos mainai siekiant validuoti sujungimą
  + **Connector** siunčia **Socket** jungties tipą (HDMI, USB, Audio ir t.t.) su papildoma informacija priklausomai nuo jungties. Jeigu kitas jungties galas jau yra sujungtas su prietaisu, **Socket** taip pat gauna prietaiso identifikacijos informaciją;
  + **Socket** siunčia **Connector** prietaiso, kuriam priklauso jungtis informaciją;
* **Connector** su **SoundManager** – įvykstant sujungimą, jungtis nusiunčia garso valdymo klasei savo sujungimo garso failo informaciją.
* **Socket** su **Device –** sujungus du prietaisus, jų jungtys nusiunčia sujungimo informaciją prietaisų valdymo klasėm. Vėliau ši informacija naudojama scenarijaus įgyvendinimo validacijai.
* **Device** su **SimulationManager** – vykdant scenarijaus įgyvendinimo validavimą, prietaisai siunčia simuliacijos valdymo klasei savo jungčių informaciją.

### Sistemos sudėtis

Sistema susideda iš techninės ir programinės įrangos. Sistemai paleisti naudojami **Oculus Quest** virtualios realybės akiniai su dviem **Oculus Touch** valdikliais. Sistemos programinė įranga yra saugoma .apk faile, kuris naudojamas sistemos instaliacijai.

Table

Description automatically generated with low confidence

4 pav. **.apk failo struktūra**

### Komponentai

Sistemos komponentai:

* **Assets** – pagrindinis sistemos aplankas
  + **EnvironmentAssets** – VR aplinkos detalėms naudojami resursai;
  + **Materials –** Unity medžiagos naudojamos modifikuoti objektų vaizdines savybes;
  + **Oculus** **Hands –** sistemoje naudojamų rankų modelių komponentai
    - **AnimationControllers –** rankoms animuoti naudojami Unity animacijų valdikliai;
    - **Animations –** VR rankų animacijos;
    - **Materials –** VR rankų medžiagos;
    - **Models –** VR rankų modeliai;
    - **Prefabs –** VR rankų Unity objektai, apjungiantys modelius ir animacijas;
  + **Prefabs –** Unity objektai susidedantys iš modelių ir juos valdančio kodo;
    - **Devices –** prietaisų Unity objektai;
    - **Hands –** rankų Unity objektai;
    - **Miscellaneous –** įvairūsnekategorizuoti Unity objektai;
  + **Resources –** jungčių ir prietaisų apjungimo informacijos ruošiniai;
    - **ConnectorData –** prietaisų jungčių tipų ruošiniai;
    - **RequirementData –** apjungimo scenarijų reikalavimų ruošiniai;
  + **Scenes –** Unity scenos;
  + **Scripts –** sistemos kodas;
    - **Connector –** kodas atsakingas už sujungimų valdymą;
    - **ConnectorData –** kodas leidžiantis kurti naujus jungčių tipus;
    - **Device –** kodas atsakingas už prietaisų identifikaciją ir sujungimų saugojimą;
    - **DrawerSetup –** kodas leidžiantis vartotojui sąveikauti su stalčiais;
    - **PhysicsMover –** papildomas kodas naudojamas judinti objektus su Rigidbody komponentu;
    - **HandPresence –** kodas valdantis vartotojo rankas, jų animacijas;
    - **RequirementData –** kodas leidžiantis kurti sujungimo scenarijų ruošinius;
    - **SimulationManager –** kodas atsakingas už scenarijaus užduočių sudarymą, sujungimo validaciją ir scenarijaus valdymą;
    - **Socket –** kodas valdantis jungčių sujungimą su prietaisais;
    - **SoundManager –** kodas atsakingas už garsų paleidimą scenarijaus metu;
    - **TeleportationManager –** kodas atsakingas už teleportacijos užklausų validavimą ir įvykdymą;
    - **Wire –** kodas vykdantis jungčių laido vizualizaciją;
  + **Shaders –** nestandartinės medžiagos;
  + **Sounds –** sistemos garsai;
  + **TextMesh** **Pro –** Unity paketas
  + **Textures –** tekstūros naudojamos sistemoje;
  + **VR** **Controller** **Models –** VR pultelių modeliai;
  + **XR Assets –** Unity XR resursai;
    - **XR –** Unity XR virtualios realybės pagrindinis paketas;
    - **XR** **Action** **Assets –** Unity XR virtualios realybės valdiklių įvedimo ruošiniai;
    - **XRI –** Unity XR virtualios realybės paketas saugantis objektų sąveikos ruošinius;

**Graphical user interface, diagram, application

Description automatically generated**

5 pav. Paketų diagrama atvaizduojanti projekto struktūrą

### Duomenų kontrolė

Duomenų kontrolė sistemoje vyksta vykdant prietaisų sujungimą ir scenarijaus įvykdymo patvirtinimą. Vykstant jungčių sujungimui, jungties duomenys yra nusiunčiami lizdui ir vykdoma validacija. Kiekviena jungtis gali būti skirtingo tipo(HDMI, USB, Audio ir t.t.) bei turėti skirtingas charakteristikas(jungties versija, antgalio tipas, t.t.). Sujungimo metu ši informacija yra palyginama su lizdo informacija, pvz.: Bandant sujungti HDMI jungtį, su USB lizdu, validacija neleis sujungti prietaisų ir jungties duomenys nebus išsaugoti. Sujungus du prietaisus, vykdoma papildoma duomenų validacija, kurios metu nustatoma ar jungtis tarp dviejų prietaisų yra aktyvi. Pvz.: Sujungus du prietaisus naudojant HDMI jungtį, bet su abejais prietaisais naudojant HDMI OUT tipo lizdus, jungčių sujungimas pavyks, bet tarp prietaisų sujungimas nėra veiksmingas. Tokiu atveju tarp prietaisų nevyks duomenų mainai ir jų sujungimo informacija nebus siunčiama simuliacijos valdymo klasei, vykstant scenarijaus įgyvendinimo tikrinimą.

Vykstant scenarijaus įgyvendinimo tikrinimui, visų scenoje esančių prietaisų sujungimo informacija yra siunčiama simuliacijos valdymo klasei. Ši klasė saugo scenarijaus įgyvendinimo kriterijus, su kuriais nustatoma ar prietaisų sujungimai yra pagrįsti.

## Testavimo eiga ir rezultatai

Sistemai testuoti naudotas testavimas scenarijais. Sudaryti scenarijai tikrinti skirtingus sistemos aspektus. Testavimo scenarijaus struktūra:

* **Aprašymas** – aprašomas testuojamas scenarijus;
* **Testo atvejai** – galimi testo rezultatai;

### Teleportacijos funkcionalumas

**Aprašymas:** Testuojamas teleportacijos funkcionalumas virtualioje realybėje. Testuojama teleportacija į galimas zonas, bei sistemos veikimas, vartotojui bandant teleportuotis į draudžiamas zonas, tokias kaip sienos, objektai, durys ir t.t.

**3 lentelė** Teleportacijos funkcionalumo atvejų lentelė

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atvejis | Aprašymas | Žingsniai | Tikėtinas rezultatas | Gautas rezultatas |
| Teleportacija įvyksta | Vartotojas bando nusiteleportuoti į galimą zoną | 1. Valdiklio lazdelė nukreipiama į priekį | Iš pultelio iššauna teleportacijos vietos indikatorius | Iš pultelio iššauna teleportacijos vietos indikatorius |
| 1. Valdiklis nukreipiamas į tinkamą teleportacijai paviršių | Indikatorius nusidažo balta spalva | Indikatorius nusidažo balta spalva |
| 1. Valdiklio lazdelė atleaidžiama | Vartotojas atsiranda norimoje vietoje | Vartotojas atsiranda norimoje vietoje |
| Teleportacija neįvyksta | Vartotojas bando nusiteleportuoti į draudžiamą zoną | 1. Valdiklio lazdelė nukreipiama į priekį | Iš pultelio iššauna teleportacijos vietos indikatorius | Iš pultelio iššauna teleportacijos vietos indikatorius |
| 1. Valdiklis nukreipiamas į netinkamą teleportacijai paviršių | Indikatorius nusidažo raudona spalva | Indikatorius nusidažo raudona spalva |
| 1. Valdiklio lazdelė atleidžiama | Niekas neįvyksta | Vartotojas nusiteleportuoja |

Vykdant teleportacijos funkcionalumo testavimą scenarijais, rasta klaida, dėl kurios bandantis nusiteleportuoti į draudžiamą zoną, teleportacija įvyktų. Klaida buvo ištaisyta.

### Objektų manipuliavimo funkcionalumas

**Aprašymas:** Testuojama galimybė manipuliuoti objektus virtualioje realybėje. Vartotojas bando paimti ir judinti šalia esančius objektus. Taip pat testuojamas tolimo paėmimo funkcionalumas, leidžiantis vartotojui paimti tolimiau esančius objektus nereikalaujant daug pastangų.

**4 lentelė** Objektų manipuliavimo funkcionalumo atvejų lentelė

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atvejis | Aprašymas | Žingsniai | Tikėtinas rezultatas | Gautas rezultatas |
| Vartotojas paima ir manipuliuoja šalia esantį objektą | Vartotojas bando paimti objektą, naudodamas savo virtualią ranką ir jį judinti virtualioje realybėje. | 1. Vartotojas užveda ranką ant objekto | Niekas neįvyksta | Niekas neįvyksta |
| 1. Vartotojas sugniaužia ranką | Objektas atsiranda vartotojo rankoje | Objektas atsiranda vartotojo rankoje |
| 1. Vartotojas judina ranką | Objektas juda su vartotojo ranka | Objektas juda su vartotojo ranka |
| 1. Vartotojas atleidžia ranką | Objektas iškrenta iš vartotojo rankos | Objektas iškrenta iš vartotojo rankos |
| Vartotojas paima ir manipuliuoja toli esantį objektą | Vartotojas bando paimti objektą, naudodamas tolimų objektų paėmimo funkciją ir jį judinti virtualioje realybėje. | 1. Vartotojas nutaiko ranką į toliau esantį objektą | Iš rankos iššauna baltas paėmimo indikatorius | Iš rankos iššauna baltas paėmimo indikatorius |
| 1. Vartotojas sugniaužia ranką | Objektas atsiranda vartotojo rankoje | Objektas atsiranda vartotojo rankoje |
| 1. Vartotojas judina ranką | Objektas juda su vartotojo ranka | Objektas juda su vartotojo ranka |
| 1. Vartotojas atleidžia ranką | Objektas iškrenta iš vartotojo rankos | Objektas iškrenta iš vartotojo rankos |

Įvykdžius objektų manipuliavimo scenarijus nerasta klaidų.

### Prietaisų sujungimo funkcionalumas

**Aprašymas:** Testuojamas prietaisų sujungimo funkcionalumas virtualioje realybėje. Prietaisai sujungiami tarpusavyje, tikrinamas jungčių atjungimas ir kiti aspektai.

**5 lentelė** Prietaisų sujungimo funkcionalumo atvejų lentelė

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atvejis | Aprašymas | Žingsniai | Tikėtinas rezultatas | Gautas rezultatas |
| Sujungiami prietaisai | Vartotojas bando sujungti du prietaisus naudojant jungtis | 1. Vartotojas paima jungties galą ir sujungia jį su laisvu prietaiso lizdu | Jungtis susijungia su lizdu ir pasigirsta sujungimo garsas | Jungtis susijungia su lizdu ir pasigirsta sujungimo garsas |
| 1. Vartotojas paima kitą jungties galą ir sujungia jį su laisvu kito prietaiso lizdu | Jungtis susijungia su lizdu, pasigirsta sujungimo garsas, vartotojas mato, jog tarp dviejų prietaisų yra ryšys | Jungtis susijungia su lizdu, pasigirsta sujungimo garsas, vartotojas mato, jog tarp dviejų prietaisų yra ryšys |
| Atjungiama jungtis | Vartotojas bando ištraukti jungtį iš prietaiso lizdo | 1. Vartotojas užveda ranką ant jungties įkištos į prietaiso lizdą | Niekas neįvyksta | Niekas neįvyksta |
| 1. Vartotojas sugniaužia ranką ir ištraukia jungtį | Jungtis išsitraukia iš lizdo ir juda su vartotojo ranka | Jungtis išsitraukia iš lizdo ir juda su vartotojo ranka |
| 1. Vartotojas atleidžia ranką | Jungtis iškrenta ir vartotojo rankos, ji nėra sujungta su prietaisu | Jungtis iškrenta ir vartotojo rankos, ji nėra sujungta su prietaisu |
| Nepavyksta įkišti jungties į lizdą | Vartotojas bando įkišti jungtį į užimtą arba netinkamo tipo lizdą | 1. Vartotojas paima jungties galą | Jungtis atsiranda vartotojo rankoje | Jungtis atsiranda vartotojo rankoje |
| 1. Vartotojas bando įkišti jungtį į lizdą | Niekas neįvyksta | Niekas neįvyksta |

Įvykdžius prietaisų sujungimo scenarijus nerasta klaidų.

### Scenarijaus valdymo funkcionalumas

**Aprašymas:** Testuojamas prietaisų sujungimo funkcionalumas virtualioje realybėje. Prietaisai sujungiami tarpusavyje, tikrinamas jungčių atjungimas ir kiti aspektai.

**6 lentelė** Scenarijaus valdymo funkcionalumo atvejų lentelė

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atvejis | Aprašymas | Žingsniai | Tikėtinas rezultatas | Gautas rezultatas |
| Scenarijus įvykdytas sėkmingai | Vartotojas teisingai įvykdęs scenarijų vykdo validaciją | 1. Vartotojas nueina prie scenarijaus reikalavimų ekrano | Vartotojas mato scenarijaus reikalavimus | Vartotojas mato scenarijaus reikalavimus |
| 1. Vartotojas sujungia prietaisus pagal reikalavimus | Nieko neįvyksta | Nieko neįvyksta |
| 1. Vartotojas nuspaudžia tikrinimo mygtuką esantį reikalavimų lentelėje | Lentelėje atsiranda sėkmingo scenarijaus įvykdymo tekstas | Lentelėje atsiranda sėkmingo scenarijaus įvykdymo tekstas |
| Scenarijus įvykdytas nesėkmingai | Vartotojas vykdo scenarijaus validaciją, neįvykdęs arba nepilnai įvykęs prietaisų sujungimą | 1. Vartotojas neteisingai sujungia prietaisus | Niekas neįvyksta | Niekas neįvyksta |
| 1. Vartotojas nuspaudžia tikrinimo mygtuką esantį reikalavimų lentelėje | Lentelėje atsiranda nesėkmingo scenarijaus įvykdymo tekstas | Lentelėje atsiranda nesėkmingo scenarijaus įvykdymo tekstas |
| Scenarijus pradedamas iš naujo | Vartotojas bando pradėti scenarijų iš naujo | 1. Vartotojas įvykdo prietaisų sujungimus ir kitus pakeitimus virtualioje erdvėje | Vartotojas mato pakeitimus virtualioje erdvėje | Vartotojas mato pakeitimus virtualioje erdvėje |
| 1. Vartotojas nueina prie scenarijaus reikalavimų ekrano | Niekas neįvyksta | Niekas neįvyksta |
| 1. Vartotojas nuspaudžia scenarijaus perkrovimo mygtuką | Vartotojas atsiranda pradinėje pozicijoje, visi objektai yra savo pradinėse pozicijose, prietaisai nėra sujungti | Vartotojas atsiranda pradinėje pozicijoje, visi objektai yra savo pradinėse pozicijose, prietaisai nėra sujungti |

Įvykdžius scenarijaus valdymo funkcionalumo scenarijų nerasta klaidų.

## Dokumentacija

## Sistemos reikalavimai

Sistemai paleisti reikalingi Oculus Quest arba Oculus Quest 2 virtualios realybės akiniai, bei jų valdikliai. Norint sistemą paleisti per kompiuterį, papildomai reikalingas Oculus Link kabelis ir Oculus programinė įranga.

## Sistemos naudojimas

Sistema paleidžiama ją pasirenkant Oculus programų kataloge ir nuspaudžiant atidarymo mygtuką. Programai atsidarius, vartotojas atsiranda uždarame kambaryje su įvairiais prietaisais:

* Kompiuteris
* Monitorius
* Televizorius
* HDMI signalo daliklis
* WEB kamera
* Klaviatūra
* Kompiuterio pelė
* 3 mikrofonai
* SDI šakotuvas

Vartotojas gali judėti kambaryje naudodamas teleportacijos funkciją. Teleportacija vykdoma nuspaudžius kairio arba dešinio valdiklio lazdelę į šiaurės poziciją. Įvykdžius šį veiksmą virtualioje erdvėje atsiras teleportacijos indikatorius, kurį vartotojas gali naudoti vietos pasirinkimui. Pasirinkus galimą teleportacijos vietą, indikatorius nusidažys balta spalva. Norėdamas patvirtinti teleportaciją, vartotojas turi atleisti valdiklio lazdelę.

Shape

Description automatically generated with low confidence

6 pav. Teleportacijos vietos indikatorius

Vartotojas gali paimti ir judinti įvairius objektus, prietaisus bei jungtis užvesdamas ant jų ranką ir nuspausdamas šoninį valdiklio mygtuką. Vartotojas gali įkišti jungtis į prietaisų lizdus, jas paimdamas ir nuvesdamas prie lizdo. Vartotojas gali atjungti prietaisus paimdamas įkištas jungtis ir atitraukdamas jas nuo prietaiso.

A picture containing indoor

Description automatically generated

7 pav. Vartotojo rankos virtualioje erdvėje

A computer on a table

Description automatically generated with low confidence

8 pav. Virtualiame kambaryje esantys įvairūs prietaisai

Vartotojo tikslas yra sujungti visus kambaryje esančius objektus pagal tam tikrą scenarijų. Scenarijaus reikalavimai gali būti matomi ant kambario sienos, esančios priešais vartotoją, prasidėjus scenarijui. Šalia reikalavimų yra trys mygtukai, Validate, Restart ir Exit. Validate leidžia vartotojui patikrinti ar įvykdyti visi scenarijaus reikalavimai. Reset mygtukas leidžia vartotojui perkrauti scenarijų iš naujo. Exit mygtukas uždaro programą.

Graphical user interface

Description automatically generated

9 pav. Scenarijaus valdymo ir reikalavimų ekranas

# Refleksija

(Praktikos ataskaitos neatsiejama dalis (kaip ataskaitos skyrius ar ataskaitos priedas) yra studento refleksija, išdėstoma dienoraščio, veiklos refleksijos, savęs įsivertinimo ar kita forma, pildoma periodiškai praktikos laikotarpiu. Refleksijoje studentas aprašo, kaip jam sekėsi atlikti užduotis (klausimai, sunkumai, kliūtys, įspūdžiai ir pan.), identifikuoja praktinės veiklos sąsajas su teorinėmis žiniomis, įvertina savo pažangą, pateikia tolesnius profesinio ir asmeninio tobulėjimo poreikius, apibendrina praktikos vertę.)

Išvados

Literatūra

<https://ktu.edu>

<https://apc.ktu.edu>

<https://startupspace.ktu.edu>

<https://nivc.ktu.edu>

(literatūros sąrašas pateikiamas pagal LST ISO 690:2010 standarto reikalavimus <https://www.ebooks.ktu.lt/einfo/1466/rasto-darbu-rengimo-metodiniai-nurodymai/>)

1. [1] Kauno technologijos universitetas. (2022). [žiūrėta 2022-03-10]. Prieiga per internetą <https://ktu.edu/universitetas/> [↑](#footnote-ref-1)