

**Technická univerzita v Košiciach  
Fakulta elektrotechniky a informatiky**

**Používateľský zážitok vo virtuálnych  
prostrediach a jeho vyhodnocovanie**

**Bakalárska práca**

**2023**

**Natanael Prokop**

**Technická univerzita v Košiciach  
Fakulta elektrotechniky a informatiky**

**Používateľský zážitok vo virtuálnych  
prostrediach a jeho vyhodnocovanie**

**Bakalárska práca**

Študijný program: Informatika  
Študijný odbor: 9.2.1. Informatika  
Školiace pracovisko: Katedra počítačov a informatiky (KPI)  
Školiteľ: Ing. Štefan Korečko, PhD.

**Košice 2023**

**Natanael Prokop**

## **Abstrakt v SJ**

Bakalárska práca skúma problematiku nedostatočného používateľského zážitku v experimentálnom nástroji NDMVR. Cieľom práce je zlepšiť používateľský zážitok v tomto nástroji a zlepšiť možnosti interakcie s komponentom pre desktopovú verziu a verziu virtuálnej reality, ako aj overiť vplyv týchto zmien prostredníctvom testovania UX. Na začiatku je vykonaná analýza používaných technológií, prístupov merania a testovania UX, identifikácia najčastejších nedostatkov a chýb znižujúcich UX pri návrhu a tvorbe virtuálnych prostredí. Nadobudnuté poznatky sa následne odzrkadľujú v odhalení základných nedostatkov nástroja NDMVR a pri návrhu riešenia so zreteľom na zlepšenie ovládateľnosti, intuitívnosti, prispôbitelnosti a komfortu pri používaní NDMVR. Po návrhu úprav a ich implementovaní vzniká optimálne a ucelené riešenie novej verzie NDMVR, ktoré prináša nové možnosti používania komponentu, ako aj lepšie prispôbenie prostredia a vizualizácie experimentálnych údajov. Vyhodnotenie sa zameriava na spracovanie výstupu z testovania UX verzie pred vykonanými zmenami a po ich vykonaní. V práci je preukázané výrazne zlepšenie UX modifikovanej verzie NDMVR.

## **Kľúčové slová v SJ**

virtálna realita, rozšírená realita, A-Frame, React, UX, používateľský zážitok, používateľské testovanie, používateľské prostredie, VR scéna, histogram

## **Abstrakt v AJ**

Bachelor thesis examines the issue of inadequate user experience in the experimental tool NDMVR. The aim of the thesis is to improve the user experience in this tool and enhance the interaction capabilities with the desktop and virtual reality components, as well as verify the impact of these changes through UX testing. Initially, an analysis of the used technologies, approaches to measuring and testing UX is conducted, identifying the most common deficiencies and errors that diminish the UX in the design and creation of virtual environments. The acquired knowledge is then reflected in the identification of the fundamental shortcomings of the NDMVR tool and the design of a solution aimed at improving controllability, intuitiveness, adaptability, and usability comfort of NDMVR. After designing the modifications and implementing them, an optimal and integrated solution of the new version of NDMVR is created, which brings new possibilities for using the component, as well as improved adaptation of the

environment and visualization of experimental data. The evaluation focuses on processing the output from UX testing of the version prior to the implemented changes and after their implementation. The thesis demonstrates a significant improvement in the UX of the modified version of NDMVR.

### **Klíčové slová v AJ**

virtual reality, augmented reality, A-Frame, React, UX, user experience, user testing, user environment, VR scene, histogram

### **Bibliografická citácia**

PROKOP, Natanael. *Používateľský zážitok vo virtuálnych prostrediach a jeho vyhodnocovanie*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2023. 75s. Vedúci práce: Ing. Štefan Korečko, PhD.



**TECHNICKÁ UNIVERZITA V KOŠICIACH**  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY  
Katedra počítačov a informatiky

**ZADANIE**  
**BAKALÁRSKEJ PRÁCE**

Študijný odbor: **Informatika**

Študijný program: **Informatika**

Názov práce:

**Používateľský zážitok vo virtuálnych prostrediach a jeho  
vyhodnocovanie**

User Experience in Virtual Environments and its Evaluation

Študent: **Natanael Prokop**

Školiteľ: **Ing. Štefan Korečko, PhD.**

Školiace pracovisko: **Katedra počítačov a informatiky**

Konzultant práce:

Pracovisko konzultanta:

Pokyny na vypracovanie bakalárskej práce:

1. Oboznámiť sa so súčasným stavom softvérového komponentu NDMVR pre vizualizáciu experimentálnych údajov v rozšírenej realite.
2. Analyzovať aktuálne prístupy k vylepšeniu a meraniu používateľského zážitku v 3D virtuálnom prostredí rozšírenej reality.
3. Identifikovať nedostatky komponentu NDMVR z hľadiska používateľského zážitku a to najmä so zreteľom na použitie virtuálno-reálnych prilieb.
4. Navrhnuť riešenie vybraných nedostatkov a implementovať ich do komponentu NDMVR.
5. Implementované riešenie overiť z využitím v analýze identifikovaných prístupov.
6. Vypracovať dokumentáciu podľa pokynov vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský

Termín pre odovzdanie práce: 26.05.2023

Dátum zadania bakalárskej práce: 31.10.2022



prof. Ing. Liberios Vokorokos, PhD.  
dekan fakulty

## **Čestné vyhlásenie**

Vyhlasujem, že som záverečnú prácu vypracoval(a) samostatne s použitím uvedenej odbornej literatúry.

Košice, 13.5.2023

.....

*Vlastnoručný podpis*

## **Podakovanie**

Na tomto mieste by som chcel srdečne poďakovať vedúcemu mojej bakalárskej práce, Ing. Štefanovi Korečkovi, PhD., za jeho vynikajúci prístup, odborné vedenie a neoceniteľnú podporu počas celého procesu vypracovania tejto práce. Rád by som taktiež poďakoval doménovému expertovi RnDr. Martinovi Vaľovi, PhD. za cenné poznatky a skúsenosti. Ich rýchla spätná väzba, prívetivá komunikácia a užitočné rady mi boli pri vypracovaní práce nápomocné a viedli ma k jej úspešnému dokončeniu.

Rovnako by som sa rád poďakoval svojim rodičom a priateľke za ich podporu a povzbudzovanie počas celého môjho štúdia.

V neposlednom rade by som sa rád poďakoval pánom Donaldovi E. Knuthovi a Leslie Lamportovi za typografický systém LATEX, ktorý mi významne pomohol pri písaní práce

# Obsah

---

Úvod	1
<b>1 Analýza prístupov k 3D používateľským rozhraniám a meraniu používateľského zážitku</b>	<b>4</b>
1.1 Merateľnosť používateľskej skúsenosti	4
1.1.1 Forma merateľnosti UX	5
1.2 Úvod do rozšírenej reality	5
1.2.1 Rozšírená realita a termín XR	6
1.3 Používateľská skúsenosť a virtuálny svet	6
1.3.1 Spôsoby použité pre hodnotenie UX	7
1.4 Meranie UX pre imitáciu reálneho prostredia	9
1.4.1 Zhrnutie postupu merania	9
1.5 Meranie použiteľnosti	10
1.5.1 System Usability Scale (SUS)	11
1.6 Meranie zaťaženia	11
1.6.1 Nasa TLX	11
1.6.2 Hodnotenie výsledkov Nasa TLX	14
1.7 Meranie angažovanosti	15
1.7.1 Refined User Engagement Scale (UES-LF)	15
1.7.2 UES Short Form (UES-SF)	15
1.7.3 Výber verzie angažovanosti	16
1.7.4 Hodnotenie UES	17
1.8 UX vo svete VR a jeho testovanie podľa Nielsena	17
1.9 Testovanie rôznych verzií prostredia VR	17
1.10 VR v prehliadači	19
1.11 Výkon a UX	19
1.11.1 Optimálne FPS	19
1.12 Meranie a zlepšenie výkonu v aplikácii VR	20
1.12.1 Postup merania snímkovej frekvencie	21

---

1.13	Zlepšenie UX v prostredí XR . . . . .	22
<b>2</b>	<b>Analýza aktuálnej verzie NDMVR</b>	<b>27</b>
2.1	Predstavenie aplikácie NDMVR . . . . .	27
2.1.1	Aktuálny stavu projektu . . . . .	28
2.1.2	Nedostatky aktuálneho NDMVR . . . . .	29
<b>3</b>	<b>Zmeny vedúce k lepšiemu UX</b>	<b>35</b>
3.1	Nedostatočné možnosti pohybu . . . . .	35
3.1.1	Spôsob odstránenia nedostatočného pohybu . . . . .	35
3.1.2	Odstránenie nedostatkov pohybu . . . . .	37
3.2	Nedostatky NDM tabletu . . . . .	39
3.2.1	Možné vylepšenia NDM tabletu . . . . .	39
3.2.2	Odstránenie nedostatkov tabletu . . . . .	40
3.3	Zmena veľkosti binu a mierka zobrazenia histogramu . . . . .	42
3.3.1	Odstránenie nedostatkov zobrazenia Histogramu . . . . .	43
3.4	Možnosť voľby prostredia . . . . .	45
3.4.1	Návrh scén komponentu . . . . .	45
3.4.2	Implementácia výberu scény . . . . .	46
3.5	Intuitívnejšie zobrazenie dát histogramu . . . . .	51
3.5.1	Návrh zmien zobrazenia histogramu . . . . .	51
3.5.2	Realizácia zmien zobrazenia histogramu . . . . .	51
3.6	Analýza histogramu z výšky . . . . .	52
3.6.1	Návrh možností lietania . . . . .	52
3.6.2	Podporované možnosti lietania . . . . .	53
<b>4</b>	<b>Testovanie vykonaných zmien</b>	<b>54</b>
4.1	Návrh UX testovania . . . . .	54
4.1.1	Návrh a výber metodík testovania . . . . .	54
4.1.2	Vhodní respondenti . . . . .	55
4.1.3	Návrh testovania dvoch verzií projektu . . . . .	56
4.1.4	Scenár testovania . . . . .	57
4.2	Realizácia UX testovania . . . . .	58
4.3	Vyhodnotenie výsledkov UX testovania . . . . .	59
4.3.1	Vyhodnotenie použiteľnosti . . . . .	60
4.3.2	Vyhodnotenie angažovanosti . . . . .	61
4.3.3	Vyhodnotenie zaťaženia . . . . .	64
4.3.4	Vyhodnotenie spätnej väzby respondentov . . . . .	65

<b>5 Záver</b>	<b>67</b>
<b>Literatúra</b>	<b>69</b>
<b>Zoznam príloh</b>	<b>76</b>

# Zoznam obrázkov

---

1.1	Príklad formy merania UX (zdroj: [3]) . . . . .	5
1.2	CUE model (zdroj: [17]) . . . . .	8
1.3	Ukážka meCUE (zdroj: [22]) . . . . .	9
1.4	Virtuálny model imitujúci reálny model budovy(zdroj: [23] ) . . .	10
1.5	SUS dotazník (zdroj: [29] ) . . . . .	12
1.6	Hodnotiaca škála výsledkov SUS(zdroj: [30] ) . . . . .	12
1.7	NASA Task Load Index (NASA-TLX) (zdroj: [34] ) . . . . .	13
1.8	Úroveň zaťaženia (NASA-TLX) (zdroj: [35] ) . . . . .	14
1.9	UES-SF (zdroj: [39] ) . . . . .	16
1.10	Optimálna vzdialenosť pre umiestnenie hologramov od používa- teľa (zdroj: [50] ) . . . . .	23
1.11	Optimálna oblasť pre obsah (zdroj: [51] ) . . . . .	24
1.12	Porovnanie možností zorného poľa VR headsetov (zdroj: [52] ) . .	24
2.1	Používateľské rozhranie NDMVR s náhľadom jednotlivých mož- ností zobrazenia(zdroj: [57]) . . . . .	28
2.2	Náhľad GITLAB data projekcie komponentu NDMVR (zdroj: [57])	29
3.1	Position + orientation in 3D space (Oculus)(zdroj: [58] ) . . . . .	36
3.2	Diagonálny pohyb po osiach na neskôr nami pridanej scéne moon .	36
3.3	Implementovaná zmena témy binov v Oculus zobrazení . . . . .	41
3.4	Minimálne zmenšenie binu, pomocou NDMVR desktop tabletu . .	43
3.5	Pridaná scéna Labs . . . . .	47
3.6	Pridaná scéna Moon . . . . .	48
3.7	Pridaná scéna imitujúca pracovisko Cernu . . . . .	48
3.8	Scéna imitujúca predošlú defaultnú scénu NDMVR . . . . .	48
3.9	Upravená scéna threetowers . . . . .	49
3.10	Jedna z pridanych scén spolu s desktop tabletom slúžiacim pre vý- ber scén . . . . .	50
3.11	Nové usporiadanie scény pri zapnutí NDMVR . . . . .	52

3.12	Podporovaný fly mód v NDMVR . . . . .	53
4.1	Výsledky hodnotenia použiteľnosti metodikou SUS verzií A a B . . .	60
4.2	Výsledky hodnotenia angažovanosti pomocou metodiky UES-SF verzií A a B . . . . .	61
4.3	Výsledky vnímanej použiteľnosti pomocou metodiky UES-SF . . .	62
4.4	Výsledky estetickej príťažlivosti pomocou metodiky UES-SF . . . .	62
4.5	Výsledky sústredenej pozornosti pomocou metodiky UES-SF . . .	63
4.6	Výsledky faktoru odmeny pomocou metodiky UES-SF . . . . .	63
4.7	Výsledky faktoru odmeny pomocou metodiky UES-SF . . . . .	64
4.8	Výsledky bonusového dotazníka . . . . .	65



# Zoznam tabuliek

---

1.1	Vyvážené poradie testovaných verzií (zdroj: [40]) . . . . .	18
4.1	Vyvážené poradie testovaných verzií (naše) . . . . .	56

# Úvod

---

Virtuálne prostredia a rozšírená realita si v oblasti používateľského zážitku (UX) získavajú stále väčšiu pozornosť. S narastajúcim využívaním týchto technológií je kľúčové venovať pozornosť hodnoteniu a zlepšovaniu UX s cieľom optimalizovať interakciu medzi používateľom a virtuálnym prostredím. Bakalárska práca využíva metódy a techniky z oblasti UX designu, psychológie, ergonómie a testovania použiteľnosti. Tieto nástroje umožňujú lepšie porozumieť potrebám a očakávaniam používateľov pri interakcii s virtuálnym prostredím.

Cieľom tejto bakalárskej práce je oboznámiť sa so súčasným stavom softvérového komponentu NDMVR, ktorý slúži na vizualizáciu experimentálnych údajov v rozšírenej realite. V rámci tejto práce analyzujeme existujúce prístupy k zlepšeniu a meraniu optimálneho používateľského zážitku v 3D virtuálnom prostredí rozšírenej reality. Práca prekonáva súčasné nedostatky a prispieva k ďalšiemu rozvoju používateľsky prívetivejších virtuálnych prostredí, ktoré poskytujú používateľom bohatší a pôsobivejší zážitok.

Prvá časť práce sa zaoberá základnými pojmami a technológiami použitými v rámci našej práce. Ďalej analyzuje prístupy k meraniu a testovaniu UX a identifikuje nedostatky a najčastejšie chyby pri návrhu a tvorbe virtuálnych prostredí s nedostatočným UX. Špecificky sa zameriava na 11 užitočných bodov, ktoré vedú k lepšiemu UX a v práci slúžia ako body heuristického hodnotenia komponentu.

Druhá časť práce je venovaná konkrétne projektu NDMVR a jeho súčasnému stavu, s cieľom analyzovať nedostatky aktuálnej verzie. V danej časti používame 11 analyzovaných bodov vedúcich k lepšiemu UX pre ohodnotenie komponentu NDMVR a načrtneme jeho základné nedostatky. Na základe tejto analýzy sú v tretej časti navrhnuté riešenia vybraných nedostatkov a ich implementácia do komponentu NDMVR. Táto časť je zameraná na oblasť používateľského rozhrania, ovládania pohybu, interakcie s virtuálnym svetom a ďalšie aspekty, ktoré majú vplyv na celkový zážitok používateľa. Cieľom návrhu a implementácie zmien je zlepšiť ovládateľnosť, intuitívnosť, a komfort pri používaní NDMVR komponentu. Používateľom prináša nové prvky a možnosti používania komponentu,

ako aj lepšie prispôsobenie prostredia a vizualizácie experimentálnych údajov ich konkrétnym potrebám. Čo prispieva k zvýšeniu spokojnosti a efektivity používateľov.

Posledná časť práce zahŕňa návrh, priebeh a vyhodnotenie merania UX, analyzovanej verzie a nami upravenej verzie NDMVR. Cieľom meraní je overiť a dokázať, že vykonané zmeny skutočne viedli k lepšiemu UX. Overenie implementovaných riešení je uskutočnené prostredníctvom systematickej analýzy existujúcich prístupov a metód merania používateľského zážitku v rámci virtuálnych prostredí uvedených v prvej analytickej časti. Venuje sa nielen objektívnym merateľným faktorom, ale aj subjektívnym pohľadom, skúsenostiam, ako aj navrhnutým zmenám z pohľadu používateľov.

Na záver tejto časti sú reprodukované výsledky merania a dosiahnuté zlepšenia UX upravenej verzie NDMVR oproti predošlej verzii. Výsledky sú prezentované pomocou využitia prehľadných a komplexných grafov, ktoré poskytujú priestor na vyvodenie záverov na základe, ktorých je daná analýza vyhodnotená. Zahrnutá je tiež časť venovaná možnostiach ďalšieho rozvoja a vylepšeniam softvérového komponentu NDMVR s ohľadom na potreby a vyjadrené nedostatky používateľov.

Kapitola 1 tejto práce poskytuje prehľad merania a testovania UX, identifikácie nedostatkov a najčastejších chýb pri tvorbe a návrhu virtuálnych prostredí s nedostatočným UX. Kapitola 2 popisuje nedostatky aktuálnej verzie NDMVR na základe poskytnutého prehľadu. Kapitola 3 navrhuje riešenie týchto nedostatkov a zaoberá sa aj ich implementáciou do aplikácie NDMVR. Kapitola 4 je venovaná návrhu, priebehu a vyhodnoteniu výsledkov UX testovania aktuálnej verzie NDMVR a verzie bez vybraných nedostatkov.

## **Formulácia úlohy**

Prvým krokom bude analýza. Analyzovať budeme zdroje zaoberajúce sa návrhom prostredí virtuálnej reality, ich najčastejšími chybami z pohľadu používateľskej skúsenosti (UX), ako aj meraním používateľskej skúsenosti v použití s rozšírenou realitou.

Po získaní dostatočných znalostí a rozhľadu v danej problematike sa budeme venovať identifikácii nedostatkov komponentu NDMVR z hľadiska UX, predovšetkým sa zameriame na UX v prípade použitia virtuálno-reálnych prílieb.

Získané výsledky sa odzrkadlia pri návrhu riešení vybraných nedostatkov a ich implementácií do komponentu NDMVR.

Na záver našej práce, implementované riešenie overíme prostredníctvom analyzovaného merania používateľskej skúsenosti v použití s rozšírenou realitou. Následne vyhodnotíme nami na základe analýzy navrhnuté a uskutočnené meranie, analyzujeme získane dáta a zaujmeme stanovisko.

Hodnotiť budeme rôzne úrovne používateľskej skúsenosti. Taktiež ponúkame zhodnotenie aktuálne upraveného prostredia a jeho stále pretrvávajúcich nedostatkov, odhalených používateľským testovaním, spolu so zaujatím nášho stanoviska.

# 1 Analýza prístupov k 3D používateľským rozhraniam a meraniu používateľského zážitku

---

Táto kapitola poskytuje prehľad literatúry (angl. literature review, LR) v oblasti experimentov zameraných na meranie používateľského zážitku (angl. User Experience, UX) v rozšírenej realite (angl. Extended reality, XR). Ďalej sa kapitola zameriava na identifikáciu najčastejších nedostatkov a chýb pri návrhu a tvorbe virtuálnych prostredí (angl. Virtual reality, VR), ktoré trpia nedostatočným UX. Tieto informácie majú kľúčový význam pri stanovení a určení výslednej témy, ktorá bude spracovaná v tejto bakalárskej práci.

Táto časť nám poskytne východiskové body pre správnu analýzu nedostatkov VR prostredí a identifikáciu potrebných zmien na ich odstránenie. Úspešnosť týchto akcií chceme neskôr overiť prostredníctvom experimentálneho merania UX.

## 1.1 Merateľnosť používateľskej skúsenosti

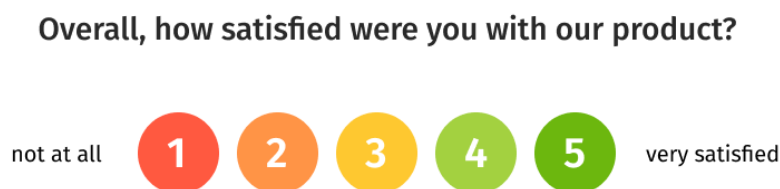
Pred samotnou analýzou prác zameraných na meranie UX v XR sme sa zamerali na otázku, či je vôbec možné merať používateľskú skúsenosť (UX). Odpoveď nám poskytla práca [1]. Práca pozostávala z dvoch štúdií, pričom prvá štúdia bola zameraná na odborníkov zo sveta UX dizajnu.

Výsledky prvej štúdie naznačovali, že mnoho odborníkov malo skeptický postoj k meraniu UX a považovalo to za irelevantné. Ich reakcie sa pohybovali od úsmevu po sarkastické vyjadrenia. Tieto výsledky vzbudili v nás ešte väčší záujem o túto tematiku, pretože naše očakávania a názory sa líšili. Z našej pôvodnej perspektívy sa jednalo o pomerne jednoduchý problém, spočívajúci v testovaní a analýze výsledkov. Preto nás prekvapilo zistenie, že samotná merateľnosť UX môže byť problematická a niektorí ju považujú za neadekvátnu.

Druhá štúdia sa zameriavala na skúmanie postojov v HCI komunitě. Účastníci boli požiadaní, aby zhodnotili svoj postoj k meraniu UX a diskutovali o téme z teoretického, metodologického a praktického hľadiska. Celkovo výsledky naznačili, že postoj k meraniu UX bol pozitívnejší v porovnaní s výsledkami prvej štúdie. Avšak názory na detaily merania UX sa stále líšili. Výsledkom bolo uznanie hodnoty subjektívneho hodnotenia používateľskej skúsenosti a potreby objektívnych meradiel. Tento výskum nás utvrdil v myšlienke, že meranie UX ponúka hodnotnú spätnú väzbu a je realizovateľné. Práca tiež vyzdvihla dôležitosť súhry medzi hodnotením UX a vývojom systému pre jeho ešte efektívnejšie použitie [1].

### 1.1.1 Forma merateľnosti UX

Ďalšou nespornou otázkou je, aké meranie sa považuje za prínosné vo svete UX. Odpoveď nám poskytla práca [2]. Práca odhalila bližší pohľad na merateľnosť UX, nakoľko v štúdiách boli práve zmienky o dotazníkoch používaných na meranie prvkov používateľského zážitku. Používanie týchto populárnych dotazníkov v štúdiách naznačuje, že určitá forma numerických meraní sa považuje za užitočnú, ba dokonca nevyhnutnú [2]. Práca nám ukázala smer a spôsob merania, ktorým by sme sa mohli uberať.



Obr. 1.1: Príklad formy merania UX (zdroj: [3])

## 1.2 Úvod do rozšírenej reality

Predtým, než začneme s podrobnejšou analýzou experimentov zameraných na meranie, zlepšenie a zabezpečenie čo najlepšieho používateľského zážitku (UX), je potrebné sa pozrieť na samotné odvetvie virtuálneho sveta. Toto pochopenie je nevyhnutné pre analýzu nasledujúcich častí tejto bakalárskej práce.

### 1.2.1 Rožšírená realita a termín XR

Termín XR predstavuje kombináciu reálneho sveta s virtuálnym prostredím. Znak „X“ predstavuje premennú pre všetky súčasné a budúce technológie priestorového výpočtu. XR zahŕňa reprezentatívne formy, ako je obohatená realita (AR), zmiešaná realita (MR) a virtuálna realita (VR) [4]. XR je rýchlo rozvíjajúce sa pole, ktoré sa uplatňuje v rôznych oblastiach, vrátane zábavy, marketingu, nenhutelností, školenia a práce na diaľku [5].

**Virtuálna realita (VR)** predstavuje simulovanú skúsenosť, ktorá môže byť podobná alebo úplne odlišná od skutočného sveta. VR sa využíva najmä v zábave, napríklad vo videohrách, ale aj v oblasti vzdelávania [4]. Príkladom je využitie VR technológie na Lekárskej fakulte Univerzity Pavla Jozefa Šafárika v Košiciach, kde sa prostredníctvom Centra simulátorovej a virtuálnej medicíny využíva detailný virtuálny model ľudského tela na tréning ošetrovateľských techník [6].

**Obohatená realita (AR)** je systém, ktorý kombinuje reálny a virtuálny svet, umožňuje interakciu v reálnom čase a presnú 3D registráciu virtuálnych a reálnych objektov [7].

**Zmiešaná realita (MR)** používateľovi poskytuje príležitosť manipulácie s digitálnymi objektmi. Objekty sa správajú podobne ako reálne, hmotné objekty. Praktické využitie MR nachádza práve vo vzdelávaní, najmä pri zameraní na manuálnu manipuláciu. Napríklad v oblasti chirurgie, dizajnu, modelovania a letectva [8].

## 1.3 Používateľská skúsenosť a virtuálny svet

Celom tejto časti je získať skutočný pohľad a inšpiráciu pre meranie používateľskej skúsenosti (UX) v súvislosti s virtuálnou realitou. Chceme zistiť, ako postupovať pri meraní a aké konkrétne oblasti by malo meranie zahrnúť.

Inšpiráciu sme čerpali z práce [9]. V tejto štúdií boli zapojení viacerí účastníci, ktorých prvotnou úlohou bolo spolupracovať v jednom virtuálnom prostredí a dosiahnuť spoločný cieľ. Neskôr sa mali pokúsiť tento cieľ splniť samostatne. Pri meraní UX bol v práci použitý **Pragmaticko/hedonický model UX** od Hassenzahla [10], ktorý umožňuje rozlišovať medzi objektívnymi vlastnosťami produktu a tým, ako ich používateľ vníma [10].

- Podľa tohto modelu sa **pragmatické** vlastnosti týkajú miery, do akej produkt umožňuje dosiahnutie cieľa, tak povediac použiteľnosti a užitočnosti.

- Zatiaľ čo **hedonická** kvalita súvisí s psychologickými potrebami a emocionálnym zážitkom používateľa.

Zvláštnu pozornosť sme venovali kapitole práce [9], ktorá sa zaoberá súvisiacimi konceptmi hodnotenia. Stanney a jeho spolupracovníci [11] uvádzajú, že prítomnosť vo virtuálnom prostredí je jedným z hlavných faktorov tvoriacich presvedčivé zážitky vo virtuálnej realite. Prítomnosť by mala byť súčasťou skúmania UX danej aplikácie. Pokiaľ sa pozrieme na oblasť hedonických kvalít, najmä afektívnych stavov, ukázalo sa, že afektívne reakcie zvyšujú pocit prítomnosti vo virtuálnej realite [12],[13]. Hodnotenie prítomnosti by preto malo zahrňovať aj afektívne merania v rámci hodnotenia UX. Štúdia tiež naznačuje, že prítomnosť je spojená s pragmatickou kvalitou (použitelnosťou a užitočnosťou), neinštrumentálnymi aspektmi (napríklad estetikou) a dôsledkami používania, ako aj s mentálnou a fyzickou ergonómiou (skúsenosťou). Zistili sme, že fyzická záťaž je typickým ukazovateľom fyzickej ergonómie, a preto by mala byť zohľadnená v našom riešení [9].

Autor práce [9] naznačuje, že pri meraní používateľskej skúsenosti vo virtuálnej realite by sme sa mali zamerať na rôzne aspekty, vrátane navigácie, hľadania cesty a manipulácie s objektmi [14]. Dôležitými faktormi sú úroveň zapojenia používateľa (angažovanosť) a výskyt simulačnej nevoľnosti [14], [15].

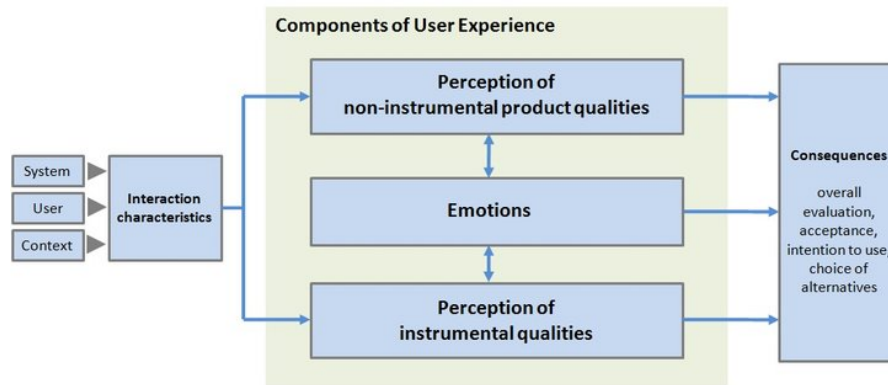
Práca [9] nám poukázala na dôležitosť merania pragmatických a hedonických prvkov používateľskej skúsenosti vo virtuálnej realite. Pričom taktiež odhalila aspekty hodnotenia používateľského zážitku, ktoré boli doteraz prehliadané, ako aj zdravotné problémy spojené s používaním virtuálnej reality a interakciou s ňou.

### 1.3.1 Spôsoby použité pre hodnotenie UX

Záver práce [9] nám podáva podrobný popis použitého spôsobu hodnotenia. Táto časť je pre náš výskum relevantná a môže nám poskytnúť potrebnú inšpiráciu.

Hodnotenie používateľskej skúsenosti bolo realizované pomocou modulov dotazníka meCUE. Tento dotazník je založený na komponentovom modeli používateľskej skúsenosti (CUE model) [16]. V rámci dotazníka sa zohľadňujú kľúčové komponenty používateľskej skúsenosti. Jeho časti sa zaoberajú vnímaním produktu na rôznych úrovniach. K pragmatickým (inštrumentálnym) kvalitám patria použiteľnosť a užitočnosť, zatiaľ čo k hedonickým (neinštrumentálnym) kvalitám patria vizuálna estetika, angažovanosť a status. V rámci hodnotenia sa kládol dôraz aj na emócie, a to pozitívne aj negatívne. Okrem toho sa hodnotili aj





Obr. 1.2: CUE model (zdroj: [17])

dôsledky používania, ako je vernosť produktu, úmysel použitia a celkové hodnotenie skúsenosti.

- Prítomnosť sa merala pomocou nemeckej verzie dotazníka o prítomnosti iGroup (iPQ) [18], ktorý zahŕňa podškály pre všeobecnú prítomnosť, realizmus, zapojenie a priestorovú prítomnosť.
- Sociálna prítomnosť sa merala pomocou modulu Sociálnej prítomnosti v rámci dotazníka o skúsenostiach z hrania hier (GEQ) [19], ktorý zahŕňa psychologické zapojenie, zapojenie do správania a negatívne pocity.
- Na meranie nepohodlia sa používala stupnica nepohodlia podľa modelu, ktorý predpokladá, že nepohodlie je ovplyvnené aspektmi biomechanického dizajnu, ako sú tlakové body [19]. Položky boli upravené, aby odrážali nepohodlie spojené s používaním vybavenia virtuálnej reality, ako sú tlakové body na hlave, namáhanie očí atď.
- Fyzická a psychická záťaž bola meraná jednou položkou, ktorá bola upravená zo škály hodnotiacej skúsenú záťaž (SEA) [20]. Príklad použitej SEA:  
„Cítiš sa byť aktuálne fyzicky napätý?“  
(Originálne znenie: “At the moment, how physically strained do you feel?”)  
„Cítiš sa byť aktuálne psychicky napätý?“  
(Originálne znenie: “At the moment, how mentally strained do you feel?”)  
Odpovede sa mohli pohybovať od „vôbec“ až po „mimoriadne napätý“.  
(Originálne znenie: “not strained at all” to “extremely strained”)
- Afektívny stav účastníkov bol hodnotený počas zážitku a to prostredníctvom plánu pozitívnych a negatívnych afektov (PANAS) [21].

- Okrem toho účastníci hodnotili svoju úroveň pocitu nevoľnosti z vybavenia virtuálnej reality pomocou stupnice Fast Motion Sickness Scale (FMS) [20] pred, počas a po skúsenosti.

The image shows a screenshot of the meCUE interface, which is a Likert scale used for evaluating user experience. The scale consists of seven horizontal rows, each representing a different statement. Each row has seven circular markers (radio buttons) for selection. The statements and their corresponding markers are as follows:

Statement	1	2	3	4	5	6	7
The product is easy to use.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The product is ergonomically designed.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The product makes me feel excited.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The product is stylish.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The design looks attractive.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is quickly apparent how to use the product.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
By using the product, I would be perceived differently.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

The scale is centered around a vertical blue line. The labels for the markers are: strongly disagree, disagree, somewhat disagree, neither agree nor disagree, somewhat agree, agree, and strongly agree.

Obr. 1.3: Ukážka meCUE (zdroj: [22])

## 1.4 Meranie UX pre imitáciu reálneho prostredia

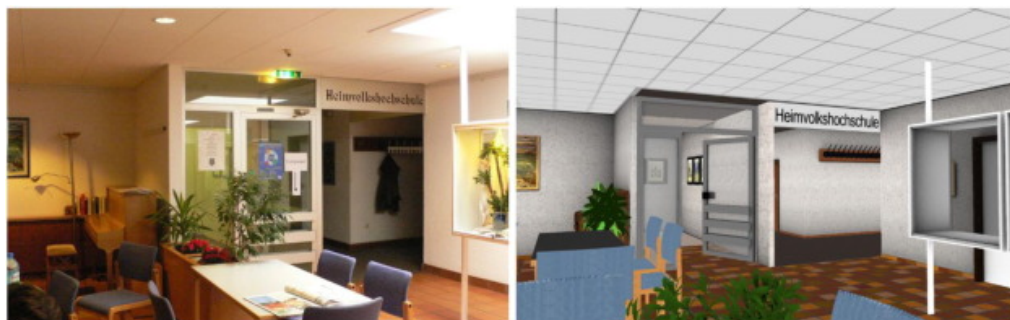
Inšpiráciu pre meranie používateľskej skúsenosti (UX) prostredia, ktoré bolo vytvorené na základe reálnej predlohy a umožňuje jej imitáciu, priniesla práca [23]. Táto práca sa zameriavala na meranie UX prostredia pomerne zaujímavým spôsobom. Motiváciou tejto práce bola skutočnosť, že virtuálna realita (VR) má potenciál byť cenným nástrojom pre štúdium interakcie človeka s prostredím. Príkladom môžu byť štúdie zamerané na použiteľnosť budov a ich hodnotenie pred a po obývaní v architektonickom výskume a praxi.

V práci [23] autori použili metódu porovnávania simulácie a skutočného prostredia na testovanie UX prostredia. Táto myšlienka môže byť pre nás užitočná pri testovaní prostredí, ktoré imitujú reálne priestory.

Práca [23] predstavila viacero metód testovania. Pre nás je prínosným test UX, ktorý pozostával z dvoch štúdií. Tieto štúdie skúmali zhodu skúseností používateľov budovy v skutočnom konferenčnom centre a vysoko podrobnom virtuálnom modeli tej istej budovy. Zaujímavým zistením bolo, že poradie prezentácie (skutočné prostredie pred virtuálnym alebo naopak) môže ovplyvniť subjektívne hodnotenie. Bola upriamená pozornosť na hypotézu (H), že účastníci budú pozitívnejšie hodnotiť virtuálny model budovy, ak im bude predstavený pred skutočnou budovou.

### 1.4.1 Zhrnutie postupu merania

Testovania sa zúčastnilo 23 študentov. Ich úlohou bolo prejsť sedem vopred vyznačených trás v budove. Účastníci boli rozdelení do dvoch skupín: Skupina 1



Obr. 1.4: Virtuálny model imitujúci reálny model budovy (zdroj: [23] )

s počtom 13 osôb a Skupina 2 s počtom 10 osôb. Skupina 1 začínala prehliadku danej trasy v reálnej budove, zatiaľ čo Skupina 2 začínala virtuálnym modelom budovy. Tento rozvrh bol použitý na potvrdenie alebo vyvrátenie hypotézy (H) [23].

Po každej zo siedmich trás účastníci hodnotili prostredie pomocou dotazníka, ktorý bol prispôsobený pre túto štúdiu. Hodnotenie prostredia sa uskutočňovalo v závislosti od podmienok (najprv reálne prostredie, potom virtuálny model alebo naopak). Tento postup sa opakoval, kým účastníci nezažili a neohodnotili všetkých sedem ciest [23].

Na konci celého experimentu účastníci hodnotili svoju celkovú používateľskú skúsenosť. Boli požiadaní, aby pomocou dotazníka ohodnotili podobnosti medzi ich vnímaním skutočných a virtuálnych environmentálnych aspektov na 6-bodovej škále Likertovho typu (od "veľmi podobné" po "vôbec nie") [23]. Tento dotazník bol vytvorený kombinovaním hodnotiacich konceptov z prác [24], [25] a [26].

Práca [23] predstavila spôsob, ktorý by sme mohli zahrnúť do našich experimentov zameraných na UX, najmä pri prostrediach, ktoré imitujú skutočné priestory, napríklad našu fakultu, prípadne konkrétne laboratórium.

## 1.5 Meranie použiteľnosti

V časti 1.3 sme sa zaoberali užitočnými aspektmi hodnotenia UX. Jeden z týchto aspektov je hodnotenie prítomnosti, ktorého meranie je možné realizovať prostredníctvom sledovania použiteľnosti daného VR prostredia/aplikácie. Na základe analyzovaného zdroja [27] sme objavili jednoduchú, overenú a rýchlu metódu sledovania použiteľnosti s názvom System Usability Scale (SUS), ktorú predstavil John Brooke už v 80. rokoch minulého storočia.

### 1.5.1 System Usability Scale (SUS)

Dotazník System Usability Scale<sup>1</sup> slúži na meranie vnímanej použiteľnosti a umožňuje získať adekvátne odpovede na otázky týkajúce sa užitočnosti systému [27]. Je jedným z najznámejších dotazníkov používaných vo výskume UX [27]. Informácia o jeho častom použití nám značne pomohla pri jeho výbere. Chceme mať overený spôsob merania. SUS je post-testovací nástroj, ktorý je poskytnutý účastníkom po skončení celej relácie testovania použiteľnosti. Obsahuje 10 otázok na Likertovej škále [28].

Účastníci majú k dispozícii stupnicu 1 – 5, kde hodnotia svoj súhlas s každým tvrdením týkajúcim sa produktu alebo funkcie v teste. Hodnota 1 znamená úplný nesúhlas, zatiaľ čo hodnota 5 znamená, že účastník úplne súhlasí s daným tvrdením. SUS dotazník poskytuje skóre v rozmedzí od 0 do 100, ktoré reprezentuje použiteľnosť [28]. Je dôležité si uvedomiť, že pomocou SUS nezískame konkrétny problém s naším prostredím alebo jeho funkciou. Slúži nám len ako základ pre zistenie celkovej použiteľnosti nášho produktu [27].

Podľa Jeffa Saura, ktorý uskutočnil rozsiahle porovnanie skóre System Usability Scale (SUS) na rôznych systémoch, je **68 percent** považované za minimálne skóre SUS, ktoré možno kategorizovať ako dobré [28]. Pri tvorbe webových stránok je potrebné dosiahnuť skóre **80 alebo vyššie**, aby sme sa umiestnili v top 10 percentách z hľadiska použiteľnosti, zatiaľ čo skóre **73** nám zaručí len umiestnenie v top 30 percentách [27].

## 1.6 Meranie zaťaženia

Pre meranie fyzickej záťaže ako ďalšieho z aspektov hodnotenia VR prostredia/aplikácie, ako sme opísali v časti 1.3, sme po analýze dostupných metódik hodnotenia zvolili dotazníkovú formu nástroja NASA Task Load Index (NASA-TLX).

### 1.6.1 Nasa TLX

Jedná sa o široko používaný subjektívny a viacrozmerný hodnotiaci nástroj, ktorý sa využíva na hodnotenie vnímanej pracovnej záťaže a je často používaný na posúdenie efektivity úloh, systémov alebo tímov. Okrem toho sa môže zamerať aj na ďalšie aspekty výkonu [31]. NASA-TLX bol vyvinutý skupinou Human Performance Group v Ames Research Center NASA a pozostáva z dvoch častí [32], [33].

---

<sup>1</sup><https://www.nngroup.com/articles/measuring-perceived-usability/>

<b>System Usability Scale Questionnaire</b>	<b>Strongly Disagree</b>	<b>Strongly Agree</b>			
1. I think that I would like to use this product frequently.	1	2	3	4	5
2. I found the product unnecessarily complex.	1	2	3	4	5
3. I thought the product was easy to use.	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this product.	1	2	3	4	5
5. I found the various functions in the product were well integrated.	1	2	3	4	5
6. I thought there was too much inconsistency in this product.	1	2	3	4	5
7. I imagine that most people would learn to use this product very quickly.	1	2	3	4	5
8. I found the product very awkward to use.	1	2	3	4	5
9. I felt very confident using the product.	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this product.	1	2	3	4	5

Obr. 1.5: SUS dotazník (zdroj: [29] )



Obr. 1.6: Hodnotiaca škála výsledkov SUS(zdroj: [30] )

Prvá časť hodnotí celkovú pracovnú záťaž, ktorá je rozdelená do šiestich subjektívnych podškál:

- **Mentálny dopyt** - Udáva, koľko mentálnej aktivity bolo potrebné vyvinúť. Vyjadruje mieru využitia mentálneho aspektu osoby. Bola úloha ľahká alebo náročná, jednoduchá alebo zložitá?

- **Fyzický dopyt** - Udáva, koľko fyzickej aktivity bolo potrebné vyvinúť. Vyjadruje mieru námahy pri fyzickom vykonávaní úlohy.
- **Časová náročnosť** - Podáva informáciu o tom, aký veľký časový tlak osoba pociťovala a aké bolo jej tempo práce. Bolo tempo pomalé alebo rýchle?
- **Celkový výkon** - Hodnotí úspešnosť a spokojnosť s výkonom. Zohľadňuje, do akej miery bola osoba úspešná pri plnení úlohy a ako spokojná bola so svojím výkonom.
- **Úsilie** - Vyjadruje množstvo úsilia, ktoré bolo potrebné vyvinúť. Zohľadňuje náročnosť práce (duševnú aj fyzickú), potrebnú na dosiahnutie určenej úrovne výkonu.
- **Úroveň frustrácie** - Hodnotí podráždenosť, stres a negatívne emócie osoby počas plnenia úlohy v porovnaní so spokojnosťou a uvoľnenosťou.

Podškály sa hodnotia na intervale od 1 do 20, pričom nižšie čísla vyjadrujú nižšiu úroveň a vyššie čísla vyjadrujú vyššiu úroveň záťaže [32].

Order the following variables according to the influence on workload:

Mental demand	Physical demand	Temporal demand	Effort	Performance	Frustration

Evaluate these variables during the monitoring of mission:

Mental demand:

Low	<input type="checkbox"/>	High
-----	--------------------------	------

Physical demand:

Low	<input type="checkbox"/>	High
-----	--------------------------	------

Temporal demand:

Low	<input type="checkbox"/>	High
-----	--------------------------	------

Effort:

Low	<input type="checkbox"/>	High
-----	--------------------------	------

Performance:

Good	<input type="checkbox"/>	Poor
------	--------------------------	------

Frustration:

Low	<input type="checkbox"/>	High
-----	--------------------------	------

Obr. 1.7: NASA Task Load Index (NASA-TLX) (zdroj: [34] )

Druhá časť obsahuje individuálnu váhu týchto šiestich podškál. Táto váha je určená pomocou porovnávania vnímanej dôležitosti. Proces zahŕňa prezentovanie 15 párových kombinácií účastníkom pričom účastníci z každého páru vyberú

škálu, ktorá má najväčší vplyv na pracovné zaťaženie počas analyzovanej úlohy [32].

<b>Workload</b>	<b>Value</b>
Low	0-9
Medium	10-29
Somewhat high	30-49
High	50-79
Very high	80-100

Obr. 1.8: Úroveň zaťaženia (NASA-TLX) (zdroj: [35] )

## 1.6.2 Hodnotenie výsledkov Nasa TLX

Skóre je vynásobené hodnotami stupnice pre danú podškálu a vydelené číslom 15. Existuje aj skrátená verzia nazývaná raw TLX, ktorá zahŕňa len prvú časť merania. Úroveň zaťaženia je možné určiť pomocou obrázka 1.8, ktorý znázorňuje kategorizáciu zaťaženia [32].

Existuje niekoľko spôsobov spracovania NASA-TLX. Zaujímavosťou je, že niektoré metódy môžu ovplyvniť výsledky testu. Jedna štúdia ukázala, že použitie papiera a ceruzky vedie k menšej kognitívnej záťaži v porovnaní so spracovaním informácií na obrazovke počítača [36]. Ďalšia štúdia však zistila, že verzie na počítačových obrazovkách aj na nositeľných zariadeniach účinne zachytávajú relatívne zmeny v pracovnej záťaži [37].

Po analýze sme objavili oficiálnu aplikáciu NASA TLX určenú pre Apple iOS<sup>2</sup>. Táto aplikácia umožňuje zachytenie párových odpovedí, subjektívneho vstupu do podškál a výpočet vážených a nevážených výsledkov. Dôležitou funkciou oficiálnej aplikácie NASA TLX je stupnica subjektívneho analógového hodnotenia (SAER), ktorá poskytuje používateľom podobný zážitok ako pri použití papierovej a ceruzkovej verzie NASA TLX. Žiadna iná počítačová verzia NASA TLX úspešne nezahrnula túto kritickú funkciu na správne zachytenie subjektívneho vstupu používateľa [37], [38].

---

<sup>2</sup><https://apps.apple.com/us/app/nasa-tlx/id1168110608>

## 1.7 Meranie angažovanosti

Aspektom merania UX v časti 1.3 bola úroveň zapojenia používateľa, konkrétne angažovanosť. Informácie sme čerpali zo štúdie [39], ktorá sa zaoberá meraním angažovanosti (user engagement) používateľov v digitálnych technológiách pomocou User Engagement Scale (UES). V rámci tejto práce bola metodika hodnotenia angažovanosti UES zdokonalená a skrátená. Autori práce navrhli a testovali nové, presnejšie verzie formulárov, ktoré by mali poskytnúť nástroj pre meranie angažovanosti používateľov vo virtuálnej realite (VR).

### 1.7.1 Refined User Engagement Scale (UES-LF)

**Rafinovaná verzia UES** bola vytvorená na základe pôvodnej verzie, ktorá sa skladá z 31 položiek, ktoré sa delia do 6 dimenzií: estetická príťažlivosť, sústredená pozornosť a vnímaná použiteľnosť, odolnosti, novosti a pocitu angažovanosti. V Rafinovanej UES dotazníku sa škály odolnosti, novosti a pocitu angažovanosti spojili do štvrtého faktora (Faktor odmeny).

- **Estetická príťažlivosť** - vizuálna príťažlivosť interiéru, rozhrania ("Aesthetic Appeal" ako "AE") - 3 otázky
- **Sústredená pozornosť** - pocit pohltienia do interakcie, pocit straty času v aplikáciách ("Focused Attention" ako "FA") - 3 otázky
- **Vnímaná použiteľnosť** - interakcia, miera kontroly, miera vynaloženého úsilia vzhľadom na použiteľnosť celku ("Perceived usability" ako "PU") - 3 otázky
- **Faktor odmeny** - do akej miery používateľ pociťuje príjemné a uspokojujúce zážitky z interakcie s aplikáciou, sleduje používateľovu ochotu odporučiť aplikáciu ostatným a angažovať sa s ňou aj v budúcnosti. ("Reward factor" ako "RW") - 3 otázky

Každá položka sa meria na škále s 5 bodmi od "rozhodne nesúhlasím" po "plne súhlasím". Počet otázok ostáva rovnaký. Výsledky ukázali, že nová rafinovaná verzia UES má vyššiu spoľahlivosť a presnosť ako pôvodná verzia UES[39].

### 1.7.2 UES Short Form (UES-SF)

**Krátka verzia UES** bola vytvorená na základe rafinovanej verzie a je určená pre použitie v situáciách, kedy je potrebné získať rýchly prehľad o angažovanosti po-



užívateľa s produktom. Výsledky tejto verzie ukázali, že má vysokú spoľahlivosť a presnosť. Dokonca až uchováva presnosť metódy USE-LF. [39].

Skrátený UES (UES-SF) pozostáva z 12 otázok, zahrnutých do spomínaných 4 oblastí záujmu (dimenzií), pričom každá položka sa opäť meria na škále s 5 bodmi od “rozhodne nesúhlasím” po “plne súhlasím” [39].

<b>FA-S.1:</b>	I lost myself in this experience.
<b>FA-S.2:</b>	The time I spent using Application X just slipped away.
<b>FA-S.3:</b>	I was absorbed in this experience.
<b>PU-S.1:</b>	I felt frustrated while using this Application X.
<b>PU-S.2:</b>	I found this Application X confusing to use.
<b>PU-S.3:</b>	Using this Application X was taxing.
<b>AE-S.1:</b>	This Application X was attractive.
<b>AE-S.2:</b>	This Application X was aesthetically appealing.
<b>AE-S.3:</b>	This Application X appealed to my senses.
<b>RW-S.1:</b>	Using Application X was worthwhile.
<b>RW-S.2:</b>	My experience was rewarding.
<b>RW-S.3:</b>	I felt interested in this experience.

Obr. 1.9: UES-SF (zdroj: [39] )

### 1.7.3 Výber verzie angažovanosti

Krátka forma môže byť ideálnejšia v rámci predmetových štúdií, kde účastníci plnia viacero úloh alebo pokusov, alebo porovnávajú dve alebo viac aplikácií [39]. Práve vďaka týmto charakteristikám sme sa rozhodli použiť UES-SF pre účely testovania virtuálnej reality.

Autori tiež porovnali UES-LF a UES-SF s inými existujúcimi metódami merania UX, ako napríklad základným dotazníkom SUS a NASA-TLX. Výsledky ukázali, že UES a UES-SF poskytujú presnejšie merania angažovanosti používateľa vo virtuálnej realite ako ostatné metódy. Preto budeme považovať ich výsledky za dôveryhodnejšie [39].

### 1.7.4 Hodnotenie UES

Skóre by sa malo vypočítať ako priemer pre každú subškálu. Otázky môžu byť jemne prispôsobené do kontextu problému a merania, ale význam a počet otázok by sa nemali meniť. Pre výpočet celkového skóre angažovanosti by sa mal sčítavať priemer každej zo štyroch podškál dlhej formy UES. V prípade UES-SF je možné sčítavať všetky položky a súčet vydeliť dvanástimi, pretože každá subškála SF obsahuje tri položky, a váženie nie je potrebné. Ak sa meranie vykonáva vo viacerých iteráciách, výsledné skóre by malo byť priemerom každej iterácie. Výsledné skóre bude číslo v rozmedzí od 1 do 5, pričom 5 sa považuje za najlepšie hodnotenie a 1 za najhoršie hodnotenie [39].

## 1.8 UX vo svete VR a jeho testovanie podľa Nielsena

V oblasti manipulácie a dizajnu používateľského rozhrania VR aplikácií chýbajú štandardy a konvencie, čo podčiarkuje dôležitosť používateľského testovania<sup>3</sup>.

Podľa Nielsena<sup>3</sup> je dôležité uľahčiť používateľom prácu a dbať na to, že VR predstavuje pomerne nový rozmer, ktorý nie je pre väčšinu používateľov ešte tak bežný. VR prináša nové možnosti, ktoré nám ešte nie sú tak prirodzené. Preto je dôležité zabezpečiť, aby používanie VR bolo intuitívne a používateľom prirodzené, a tým ich viac zapájať do tejto technológie. Treba klásť dôraz na jednoduchosť ovládania, intuitívne rozhranie a dostupné návody, aby sa používatelia cítili pohodlne a mohli plne využívať nové možnosti, ktoré VR ponúka.

Pri získavaní spätnej väzby je potrebné používateľom zadať konkrétnu úlohu, ako napríklad vyhľadanie, objavenie alebo využitie určitého prvku. Ukážka dema a voľného vyskúšania je nepostačujúca. Úlohy by mali byť podobné reálnym situáciám a zohľadňovať očakávanú reakciu<sup>3</sup>.

Získavanie spätnej väzby vo forme dotazníka je lepšie realizovať priamo v prostredí VR, namiesto zloženia headsetu a vyplnenia dotazníka mimo VR prostredia. Manipulácia s headsetom môže byť rušivá pre používateľa a viesť k skresleným výsledkom<sup>3</sup>.

## 1.9 Testovanie rôznych verzií prostredia VR

Táto časť odкрýva, ako správne testovať aktuálnu (nezmenenú) verziu prostredia VR a verziu zahrňajúcu vylepšenia UX.

---

<sup>3</sup><https://www.nngroup.com/videos/virtual-reality-and-user-experience/>

Zdrojom informácií pre nás bola kniha [40], ktorá sa zameriava na proces testovania použiteľnosti s cieľom zlepšenia UX dizajnu. V kapitole 5 v časti Testing Multiple Product Versions sa nachádza relevantná problematika týkajúca sa testovania dvoch alebo viacerých verzií aplikácií a projektov.

V kapitole 5 sa uvádza, že testovanie dvoch verzií aplikácií alebo projektov je užitočné pre zistenie, ktorá verzia funguje lepšie alebo aké sú rozdiely medzi nimi. Je dôležité, aby testovanie prebiehalo s rovnakým súborom používateľov a rovnakými úlohami, aby sme mohli porovnať výsledky.

Použitie rozdielnych respondentov by mohlo spôsobiť problémy, ako sú rozdiely v skúsenostiach používateľov, ktoré by mohli ovplyvniť výsledky testovania. Autor taktiež upozorňuje na problémy spojené s výhodou pre verziu, ktorá je testovaná ako posledná, pretože účastník sa môže naučiť vykonávať úlohy pri používaní prvej verzie a to by mohlo ovplyvniť jeho schopnosť prispôbiť sa druhej verzii, ak je táto diametrálne odlišná. V oboch prípadoch by mohli byť naše výsledky skreslené [40].

**Riešenie**, ktoré navrhuje kniha [40], spočíva v testovaní rovnakých úloh s rovnakými používateľskými skupinami, aby sme mohli adekvátne porovnať výsledky. Je však nevyhnutné vyskúšať každú z verzií postupne vo vyváženom poradí. To znamená, že potrebujeme párnny počet účastníkov, pričom každá verzia sa vykoná rovnaký početkrát na prvej pozícii a na poslednej pozícii, čo zabezpečuje rovnaké podmienky a minimalizuje potenciálne efekty ovplyvnenia. Odporúča sa mať v skupine respondentov minimálne 4 osoby pre dostatočne výpovednú hodnotu.

Skupina respondentov	Poradie testovaných verzií
Ginny	A B
Stephanie	B A
Ken	A B
James	B A

Tabuľka 1.1: Vyvážené poradie testovaných verzií (zdroj: [40])

## 1.10 VR v prehliadači

**Knižnica A-Frame** [41] je open-source webový rámec, ktorý je založený na technológiách WebXR a Three.js. Je určený na vytváranie interaktívnych 3D scén a podporuje rôzne hardvérové zariadenia pre virtuálnu a rozšírenú realitu. Tieto scény môžeme zobrazit pomocou kompatibilných webových prehliadačov, ako napríklad Chrome a Firefox [42]. A-Frame ponúka podporu pre viaceré platformy a je kompatibilný s operačnými systémami iOS, Android a Windows. Je však potrebné brať do úvahy, že výkon webových VR aplikácií môže byť ovplyvnený obmedzeniami GPU a CPU, najmä v prípade aplikácií s viacerými používateľmi [43].

**WebXR**, tiež známy ako WebVR, je aplikačné programovacie rozhranie (API), ktoré umožňuje vývoj VR aplikácií priamo prostredníctvom webového prehliadača [44]. Je to kombinácia dvoch technológií - Webu a VR. Web sa stará o HTML, CSS štýly a JavaScript, zatiaľ čo VR zahŕňa využitie VR zariadení, ako sú napríklad Oculus a Google Cardboard [43]. WebXR je často používaný spolu s WebGL, ktorý je JavaScriptové API na vykresľovanie 2D a 3D grafiky v kompatibilných webových prehliadačoch [45]. Tieto technológie spoločne umožňujú používateľom zažiť VR priamo vo svojom webovom prehliadači. Na rozdiel od VR aplikácií vyvinutých v herných enginech, ako sú Unity a Unreal Engine, ktoré si vyžadujú sťahovanie a inštaláciu na zariadení, WebXR umožňuje prístup k VR aplikáciám priamo a rýchlo prostredníctvom kompatibilného webového prehliadača [43].

## 1.11 Výkon a UX

Táto časť sa zaoberá minimálnymi požiadavkami na plynulý priebeh a pozitívne hodnotenie používateľského zážitku v oblasti VR prostredia, konkrétne zabezpečením stabilnej snímkovej frekvencie [43].

Vykresľovanie grafiky vo VR prostredí vo webových prehliadačoch môže byť náročnou úlohou, najmä pre grafické procesory (GPU). Vyžaduje si výkonnú grafiku a hardvér na spracovanie, aby sa dosiahla stabilná snímková frekvencia. V tejto kapitole sa analyzujú optimálne obnovovacie frekvencie pre riešenia Web VR a možnosti merania a zlepšenia výkonu [46, 47].

### 1.11.1 Optimálne FPS

Frames Per Second (FPS) je dôležitou metrikou vo 3D a VR aplikáciách. Vyššia snímková frekvencia zvyšuje vnímanú kvalitu grafiky a plynulosť používa-

teľského zážitku [43].

Väčšina štandardných prenosných počítačov má obnovovaciu frekvenciu 60 Hz. Ak je snímková frekvencia vyššia ako táto obnovovacia frekvencia, môže dochádzať k trhaniu obrazovky a vizuálnemu oneskoreniu, ak nie je povolená vertikálna synchronizácia<sup>4</sup> (Vsync) [43].

Niektoré herné notebooky poskytujú obnovovaciu frekvenciu až 165 Hz, avšak len niekoľko z nich plne podporuje VR kvôli rozdielom v GPU [43]. Prehliadače bežia pri snímkovej frekvencii 50-60 snímok za sekundu [48]. V dokumentácii A-Frame sa uvádza, že optimálna snímková frekvencia s rozhraním WebVR 1.0 API je 90 snímok za sekundu<sup>5</sup>.

Podľa internetových zdrojov je 60 FPS považovaných za optimálnu snímkovú frekvenciu pre väčšinu ľudí pri hraní hier. Toto platí nielen kvôli plynulosti zobrazovaného obrazu, ale aj vzhľadom na dostupnosť 60Hz monitorov. Získanie GPU, ktoré dokáže dosiahnuť 60 FPS vo videohrách, je v súčasnosti relatívne jednoduché a cenovo dostupné<sup>6</sup>. Pre optimálny zážitok vo VR headsetoch sa však považuje snímková frekvencia 90 FPS za ideálnu<sup>7</sup>. V prípade PC riešenia však môžeme považovať stabilných 60 FPS za dostačujúce.

## 1.12 Meranie a zlepšenie výkonu v aplikácii VR

Inšpiráciou pre nás bola štúdia [43], ktorá sa zaoberá meraním a zlepšovaním počtu snímok za sekundu (FPS) vo virtuálnej realite (VR) pomocou knižnice A-Frame. V tejto štúdií je použitá nová adaptívna schéma rozlíšenia pre VR nazvaná Adaptive Resolution Scheme for VR (ARS-VR). Táto schéma zlepšuje výkon VR z hľadiska snímok za sekundu a latencie snímok na zariadeniach s obmedzenými výpočtovými a grafickými schopnosťami. Algoritmus ARS-VR je implementovaný pomocou JavaScriptu a upravuje rozlíšenie aplikácie vo webovom prehliadači na základe aktuálnej snímkovej frekvencie. Ukážka zdrojového kódu ARS-VR je uvedená v kóde 1.1.

Zdrojový kód 1.1: ARS-VR algoritmus (zdroj: [43] )

```
while browser_window==open do
  Input: framerate; screen_width;
        device_pixel_ratio; window_height
  Output: maxCanvasSize
```

---

<sup>4</sup><https://www.digitaltrends.com/computing/what-is-vsync/>

<sup>5</sup><https://aframe.io/docs/1.3.0/components/stats.html>

<sup>6</sup><https://www.gpumag.com/good-fps-for-gaming/>

<sup>7</sup><https://libguides.bodleian.ox.ac.uk/vr/headsetsvr>

```
Function greatest_common_divisor(w,h):
    var w = screen_width*device_pixel_ratio
    var h = window_height*device_pixel_ratio
    var ratio = screen_width/screen_height
    var gcd = greatest_common_divisor(w, h)
    var num = w/gcd
    var den = h/gcd
    return num, den, ratio, w, h
Function Main:
if ratio <1 then
    maxCanvasSize = {height: h/2, width: w/2}
else
    maxCanvasSize = {height: framerate*den,
width: framerate*num}
    return maxCanvasSize
end
end
```

Cieľom bolo prispôbiť rozlíšenie vykresľovaného obsahu, čím sa mal zvýšiť počet vykreslených snímok za sekundu. V práci bol použitá taktiež ďalšia knižnica, ktorá je postavená na vrchole A-Frame: Networked A-Frame. Tá umožňuje vytváranie šablón pre synchronizáciu polohy avatarov a prehrávanie synchronizovaného videa. Nakoľko sa jedná o aplikáciu umožňujúcu používateľom sledovať film synchronizovaným spôsobom v prostredí virtuálnej kinosály. Vzdialení používatelia sa môžu pripojiť k scéne VR s alebo bez náhlavnej súpravy VR a následne môžu spoločne sledovať film. V danej scéne bola vykonaná analýza výkonu aplikácie s viacerými používateľmi pripojenými prostredníctvom rôznych zariadení. Použitá bola architektúra klient-server. Ako aplikačný server a klient boli použité dva notebooky s príslušnými špecifikáciami uvedenými v práci [43]. Ako klientsky prehliadač bol použitý Chrome. Okrem toho bol na simuláciu klienta z inej siete, ako server použitý smartfón. Jeho špecifikácie môžeme nájsť taktiež v práci [43].

### 1.12.1 Postup merania snímkovej frekvencie

Test v práci [43] prebiehal nasledovne: bol spustený aplikačný server a otvorilo sa viacero okien prehliadača, pričom každé okno zobrazovalo avatar klienta. Aplikácia bežala vo svojom predvolenom rozlíšení po dobu približne 120 sekúnd bez použitia algoritmu ARS-VR. Počas tohto obdobia sa prehralo video vo vnútri

scény a údaje o snímkovej frekvencii sa odosielať na výstup konzoly každú sekundu. Následne boli tieto údaje zhromaždené.

Následne bolo znovu otvorené okno aplikácie, tentokrát s použitím algoritmu ARS-VR. Test bol vykonaný najprv pre 4 používateľov. Podľa grafov v práci [43] sme zistili, že snímková frekvencia v prvom prípade začala dobre, ale niekoľkokrát klesla a mohla dosiahnuť až 50 snímkov za sekundu. V druhom prípade sa snímková frekvencia spustila skvele a zostala stabilná pri 60 snímkoch za sekundu počas celého testu. Rovnaký experiment opakovali pre 6 a 8 používateľov.

Hoci snímková frekvencia vždy klesla pri pridaní nového klienta a ďalší prudký pokles bol spôsobený pri navigácii medzi oknami prehliadača, použitie algoritmu ARS-VR v práci [43] v scenároch so 4, 6 a 8 používateľmi znížilo stratu snímkov za sekundu o 23 percent, 30 percent a 28 percent. Tieto výsledky sú veľmi priaznivé pre našu analýzu, pretože použitie algoritmu ARS-VR by mohlo zlepšiť výsledky meraní aj samotný užívateľský zážitok (UX).

## 1.13 Zlepšenie UX v prostredí XR

Kapitola je venovaná najčastejším nedostatkom a chybám XR aplikácií z pohľadu UX. Cieľom tejto kapitoly je poskytnúť všeobecné odporúčania získané z rôznych zdrojov týkajúcich sa UX dizajnu a VR. Na získanie týchto informácií sme použili prácu [49], ktorá sa zaoberá navrhovaním aplikácií pre nositeľné zariadenia VR ako HMD XR, Microsoft HoloLens, HTC Vive, Oculus Rift a ďalšie, ktoré využívajú stereoskopické zobrazovanie. Táto práca analyzuje 60 webových zdrojov, 1 recenzovaný článok, 1 knihu, 3 nerecenzované akademické zdroje a 3 tradičné zdroje UX. Na základe týchto informácií bol vytvorený súbor pokynov pre maximalizáciu UX a návrhu aplikácií XR, ktoré integrujú rôzne nápady z rôznych komunit XR a UX.

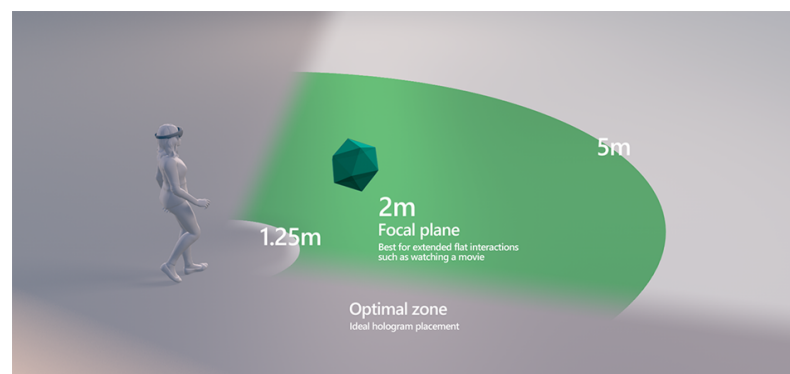
Výstupom práce [49] bolo 11 bodov, na ktoré sme sa rozhodli zamerať.

**Prvý bod** sa zaoberá usporiadaním prostredia XR s cieľom maximalizovať jeho efektivitu. Pri návrhu prostredia je dôležité zohľadniť pohodlné používanie a minimalizáciu námahy zo strany používateľa. Prostredie by malo využívať priestor efektívne a nevyžadovať nadmerný fyzický pohyb. Pri umiestňovaní obsahu treba brať do úvahy fyzické obmedzenia a obmedzené zorné pole používateľov. Vizuálne prvky je vhodné umiestniť do oblastí, kde je možné ich pohodlne sledovať aj po dlhšiu dobu, a interaktívne prvky umiestniť do priestoru, kde je interakcia prirodzená a uvoľnená. Treba tiež dbať na zabránenie preplneniu obsahu a zbytočnému zobrazovaniu informácií. Menej je viac. Zlepšenie efektivity práce s

prostredím je možné dosiahnuť zoskupením podobných predmetov, čo uľahčuje ich nájdenie používateľom.

**Druhý bod** opisuje vytváranie flexibilnej interakcie v XR aplikáciách a dôraz na poskytnutie možností prispôsobenia sa používateľom podľa ich preferencií a pohodlia. Je užitočné mať funkcie, ktoré umožňujú širokému spektru používateľov využívať aplikáciu XR. Keď používateľom umožníte určitú kontrolu nad umiestnením digitálnych prvkov, zlepši sa ich celková spokojnosť a jednoduchosť používania. Aplikácie by tiež mali poskytovať možnosti prispôsobenia pre rôzne úrovne skúseností používateľov - jednoduchšiu interakciu pre začiatočníkov a rýchlejšie akcie pre skúsenejších používateľov. Prispôsobenie sa dá využiť na zvýšenie efektivity, napríklad umiestnením často používaného obsahu do priameho zorného poľa a možnosťou meniť jeho polohu podľa situácie. Faktory ako osobné hranice, fyzické obmedzenia, sociálne aspekty a jas by mali byť flexibilné a prispôsobiteľné.

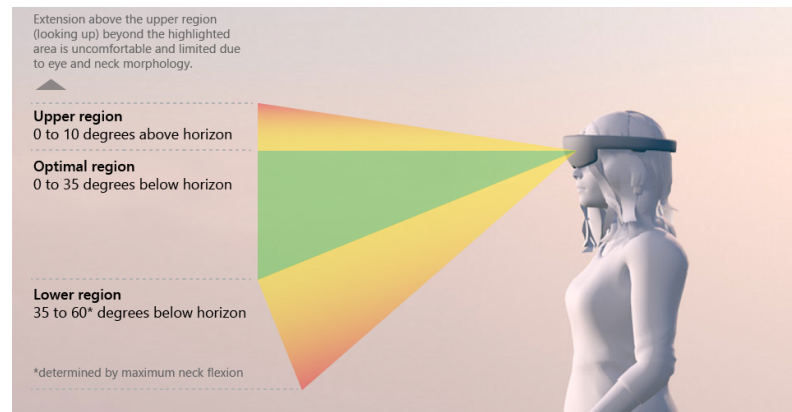
**Tretí bod** odporúča rešpektovať osobný priestor používateľov, pretože ľudia sú fyzicky aj emocionálne citliví najmä v oblasti hlavy. Je vhodné vyhnúť sa umiestneniu objektov príliš blízko k používateľovi a nastaviť ich na pohodlnú vzdialenosť. Používateľom treba umožniť ovplyvňovať svoj osobný priestor a poskytnúť im možnosť meniť alebo upravovať prostredie, ak im nie je príjemné. Je dôležité vyhnúť sa fyzicky namáhavým interakciám, ako je obsah mimo zorného poľa používateľa, ktorý vyžaduje časté otočenie tela alebo hlavy [49]. Microsoft odporúča optimálnu oblasť pre obsah vo výške od 0 do 35 stupňov pod horizontom, ako je znázornené na obr. 1.11. Odporúčaná oblasť uvedená na obr. 1.10 pre obsah hologramov scény je 1,25-5 metrov.



Obr. 1.10: Optimálna vzdialenosť pre umiestnenie hologramov od používateľa (zdroj: [50] )

**Štvrtý bod** hovorí, že menej je viac. Je lepšie mať v prostredí menej prvkov, pretože používateľ si tak viac zapamätá a nedochádza k rozptyľovaniu pozor-

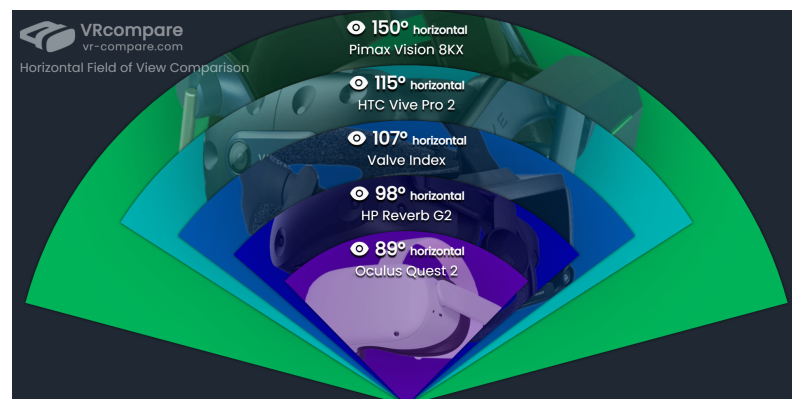




Obr. 1.11: Optimálna oblasť pre obsah (zdroj: [51] )

nosti. XR poskytuje viac virtuálneho priestoru, ale stále existuje limit pre množstvo informácií, ktoré môže používateľ spracovať. Preto je dôležité mať jednoduché a relevantné prvky v prostredí a minimalizovať irelevantné informácie, ktoré by mohli súťažiť o pozornosť používateľa. Treba umožniť používateľom skryť, minimalizovať alebo vypnúť prvky, a tiež mať vizuálne tiché prvky počas nečinnosti, aby neprekážali pri iných úlohách. Dôležité odporúčanie je nezakrývať zrak používateľa virtuálnymi prvkami a minimalizovať hustotu informácií v zornom poli. Je potrebná opatrnosť pri používaní perzistentných displejov (HUD), ktoré môžu zaberáť veľkú časť zorného poľa používateľa.

**Piaty bod** sleduje veľkosť a umiestnenie vizuálnych prvkov, ktoré by mali byť prispôbené hardvéru, na ktorom aplikácia beží. Napríklad, treba brať do úvahy zorné pole VR headsetu. Je užitočné vyzdvihnúť silné stránky daného hardvéru (napr. väčšie zorné pole) a využiť ich vo vývoji aplikácie. Týmto spôsobom minimalizujeme vplyv hardvérových obmedzení na používateľský zážitok (UX). Hardvérové možnosti zorného poľa jednotlivých zariadení VR sú uvedené na obr 1.12.



Obr. 1.12: Porovnanie možností zorného poľa VR headsetov (zdroj: [52] )

**Šiesty bod** hovorí o dôležitosti poskytovania pomoci a navigácie používateľom v XR aplikáciách, nakoľko mnohí z nich nemusia byť zvyknutí na nové spôsoby interakcie. Smerové ukazovatele môžu pomôcť používateľom orientovať sa a vedú ich k dôležitému obsahu v aplikácii. Je dôležité, aby táto pomoc bola jednoduchá a nezaťažovala používateľov zbytočným množstvom navigačných prvkov, ako je napríklad nadmerné množstvo textu. Je tiež dôležité rozdeliť navigáciu na menšie časti a zabezpečiť, aby boli jasné a ľahko pochopiteľné. Používateľov je potrebné informovať nielen o tom, čo môžu robiť, ale aj o obmedzeniach a možných chybách. Pred vykonaním akcie je vhodné upozorniť používateľov na možné nebezpečenstvá a následky.

**Bod číslo sedem** opisuje zlepšenia XR zážitku prostredníctvom zapojenia viacerých zmyslov, ako sú vizuálne, zvukové a naratívne prvky. Priestorový zvuk hrá kľúčovú rolu v tom, že vytvára priestorové vzťahy a prispieva k ponoreniu používateľov do virtuálneho prostredia. Zvukové efekty môžu zvýšiť úroveň ponorenia a poskytnúť pôsobivý zážitok. Čím je zapojenie zmyslov komplexnejšie, tým lepší zážitok je ponúkaný používateľom.

**V bode číslo osem** je zameraný na zlepšenie UX pomocou využitia znalostí z reálneho sveta. Je dobré využiť mentálne modely, ktoré máme ako ľudia na základe predchádzajúcich skúseností. Virtuálne prvky by mali odrážať vlastnosti skutočných objektov, čo môže znížiť kognitívne zaťaženie a vzdelávať používateľov implicitne. Dôležité je tiež spárovať akcie s očakávanými výsledkami, ktoré používatelia očakávajú na základe svojich skúseností. V XR môžu byť očakávania používateľov týkajúce sa detailov interakcií vyššie kvôli ponoreniu a asociáciám so skutočnými zážitkami. Pri používaní prvkov a predmetov z reálneho sveta je dôležité zvážiť ochranu pred zlým použitím a zohľadniť obmedzenia.

**Deviaty bod** hovorí o dôležitosti konzistentnej spätnej väzby pri interakcii s prostredím. Čím dôslednejšia spätná väzba bude používateľovi poskytnutá, tým lepšie bude môcť prijímať informované rozhodnutia. Digitálne prvky nemajú záruku, že sa budú správať rovnako ako skutočné objekty, preto je dôležité implementovať konzistentnú spätnú väzbu, ktorá informuje používateľa o jeho interakciách s prostredím. Každá interakcia by mala mať odlišné stavy, ktoré sa prenášajú pomocou vizuálnej spätnej väzby. Tieto stavy by mali mať konzistentné vlastnosti, čo umožní používateľom rozpoznať ich pri používaní aplikácie. Spätná väzba by tiež mala slúžiť ako forma pomoci pri identifikácii chýb a nežiaducich stavov. Dôležitá je otázka: Je akcia možná? Bola interakcia zaznamenaná? Boli splnené všetky požiadavky na vykonanie akcie? Vďaka adekvátnej spätnej väzbe môžu používatelia zmeniť svoje správanie a poučiť sa zo svojich interakčných chýb.

**Bod číslo desať** je zameraný na dôležitosť udržiavať pocit kontroly u používateľa pri interakcii s aplikáciou v prostredí XR. Používatelia sú zraniteľnejší, keď sú ponorení do virtuálneho prostredia, preto je dôležité vytvoriť u nich pocit dôvery tým, že im umožníme mať kontrolu nad aplikáciou. Aplikácia by nemala vykonávať akcie bez súhlasu používateľa a nemala by predpokladať, čo chce používateľ robiť. Pred vykonaním akcie s následkami je dôležité získať výslovné povolenie od používateľa. Mala by byť poskytnutá jednoduchá možnosť opustiť nechcené situácie a akcie s minimálnym odporom a ťažkosťami zo strany aplikácie. Pocit kontroly sa týka aj ovládania kamery virtuálneho sveta. Používateľ by mal mať kontrolu nad ovládaním kamery, čo mu umožní cítiť sa pod kontrolou a ešte viac sa ponoriť do zážitku ponúkaného aplikáciou.

Umožniť používateľovi skúmať aplikáciu bez obáv odporúča **bod číslo jedenásť**. Ten sa zameriava na spôsob prístupu pokus a omyl, pričom nás informuje o nevyhnutnosti povoliť v maximálnej možnej miere reverzibilné akcie. Nemali by sme však zabudnúť nastaviť ochranu proti potenciálnym chybám. Obe použitia zmiernia úzkosť používateľa zo skúmania aplikácie. V aplikáciách XR nemusia byť tradičné tlačidlá späť alebo klávesové skratky vždy vhodné, a preto je dobré hľadať iné alternatívy, ktoré umožnia jednoduché zrušenie akcií. Umožnenie používateľovi voľne a bez obáv skúmať aplikáciu nakoniec len zlepší celkový používateľský zážitok (UX).

## 2 Analýza aktuálnej verzie NDMVR

---

V tejto kapitole sme sa rozhodli zamerať na softvérový komponent NDMVR (N-DiMensional Virtual Reality). Ten sa pokúsime v krátkosti predstaviť, priblížiť jeho aktuálny stav, ako aj využiť nadobudnuté znalosti z predošlej kapitoly 1 na analýzu nedostatkov, brániacich plnohodnotnému UX.

### 2.1 Predstavenie aplikácie NDMVR

Fyzikálne experimenty zamerané na urýchľovač častíc dokážu generovať veľké množstvo komplexných dát. S týmito dátami následne pracujú fyzici a na základe ich spracovania vyvodzujú závery, rôzne tvrdenia, prehlásenia. Práve tejto problematike sa venuje komponent NDMVR [53], ktorý umožňuje vizualizovať dvoj- a trojrozmerné histogramy (TH2 a TH3) vo webovom VR, vo viere lepšej, rýchlejšej, pohodlnejšej a prínosnejšej analýzy týchto dát. Dáta sú reprezentované vo forme histogramov, ktorých zobrazovacími objektami sú takzvané biny. Význam tejto reprezentácie nám odhalil Karl Pearsna [54], ktorý ju popísal, ako sériu intervalov vytvorenú z množiny hodnôt, ktoré sa neprekrývajú. Pričom biny môžu, ale nemusia byť rovnakej veľkosti. Pri intervaloch rovnakej veľkosti, sa nad intervalom zobrazuje obdĺžnik, ktorého výška zodpovedá frekvencii výskytu daných hodnôt v konkrétnom intervale. NDMVR (N-DiMensional Virtual Reality) bol navrhnutý ako opakovane použiteľný komponent využívajúci softvérové rámce A-Frame [41] a React.js <sup>1</sup> [53]. Ako sme sa už dozvedeli v kapitole 1.10, A-Frame zabezpečuje 3D vizualizáciu a podporu rôzneho hardvéru virtuálnej a rozšírenej reality, zatiaľ čo React.js optimalizuje jeho výkon a umožňuje ho sprístupniť vo forme opakovane použiteľného a prispôsobiteľného komponentu[53]. Naša časť práce nadväzuje na prácu venovanú vizualizácií experimentálnych údajov v zdieleanej rozšírenej realite a jej používateľskému rozhraniu [55], ktorej sa venoval Bc. Martin Fekete. V danej tematike sa rozhodol pokračovať Bc.Lukáš Mikula, ktorý popísal a zdokonalil možnosti vizualizácie experimentálnych údajov vo webo-

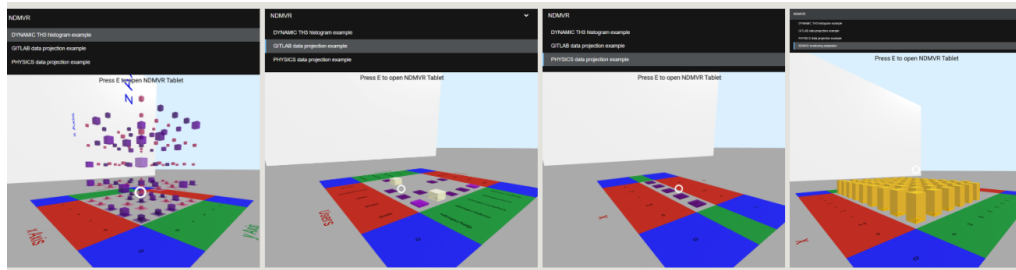
---

<sup>1</sup><https://reactjs.org/>

vej rozšírenej realite [56]. Na interakciu používame buď klasickú zostavu, obrazovku, myš, klávesnicu, alebo stereoskopické zariadenia spolu s ich ovládačmi, ktoré ponúkajú oveľa väčší zážitok. V našej práci sa budeme predovšetkým venovať zobrazeniu pomocou Oculus zariadenia.

### 2.1.1 Aktuálny stavu projektu

Aktuálna verzia NDMVR je dostupná online na [57] a beží výhradne na strane klienta vo webovom prehliadači. Ak sa pozrieme na architektúru NDMVR, môžeme vidieť, že pozostáva z viacerých častí. Top-level časťou je NDMVR – npm balík, ktorý možno začleniť do prakticky akejkoľvek webovej aplikácie, vytvorenej v React.js. Pod touto vrstvou sa nachádza VR scene. Je to časť, ktorá je zodpovedná za vykreslenie 3D scény, ktorá vizualizuje histogramy typu TH2 alebo TH3. Vykreslenie prebieha práve vďaka jej dvom podčasťam a to VR camera a Histogram 3D. Histogram 3D je zodpovedný za VR scénu s histogramom. VR kamera definuje uhol pohľadu prvej osoby (používateľa). Údaje (vizualizované ako histogramy) sa získavajú vo formáte ROOT Data Analysis Framework. Pre tento účel NDMVR používa JSROOT<sup>2</sup> už existujúci rámec pre webovú vizualizáciu vedeckých údajov a ich spracovanie. Je tiež používaný na vykreslenie TH1 histogramu, konkrétneho binu histogramu v 3D scéne.

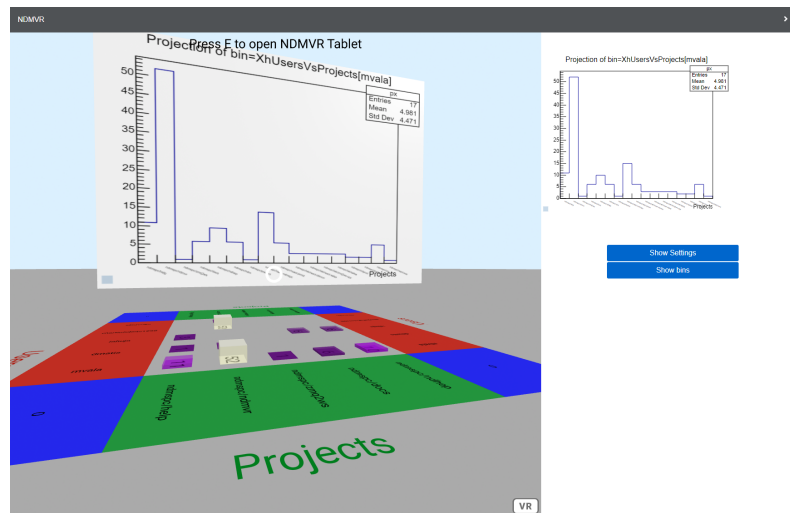


Obr. 2.1: Používateľské rozhranie NDMVR s náhľadom jednotlivých možností zobrazenia (zdroj: [57])

Obrázok 2.1 ukazuje, ako vyzerá používateľské rozhranie NDMVR vo webovom prehliadači pre jednotlivé príklady zobrazenia. Rozhranie má dva režimy zobrazenia, normálny režim a režim VR. Po otvorení webového prehliadača je nám ponúknutý režim normálneho zobrazenia jedná sa o tzv. predvolený režim. Na obrázku 2.2 môžeme vidieť vľavo 3D scénu spolu s histogramom, zatiaľ čo vpravo môžeme vidieť histogram TH1 pre vybraný bin, ktorý je označený zelenou farbou. Na pravej strane máme k dispozícii dve tlačidlá. Tlačidlo Show Settings slúžiace pre nastavenie požadovanej vzorky dát a tlačidlo Show bins, ktoré

<sup>2</sup><https://root.cern.ch/js/>

slúži pre zobrazenie aktuálne zvolených, ako aj posledne zvolených binov. Okrem týchto elementov, sa v pravom dolnom rohu obrazovky vždy nachádza tlačidlo slúžiace pre prepnutie sa do zobrazenia VR. V 3D scéne sa používateľ môže voľne pohybovať, posúvať histogram pozdĺž červenej a zelenej osi, vybrať ľubovoľný bin a zobraziť si jeho TH1. Po vstupe do VR sú používateľovi k dispozícii pomocné panely s podrobnými pokynmi, ako používať vizualizáciu.



Obr. 2.2: Náhľad GITLAB data projekcie komponentu NDMVR (zdroj: [57])

## 2.1.2 Nedostatky aktuálneho NDMVR

Po vyskúšaní zážitku aktuálnej verzie aplikácie NDMVR prostredníctvom Oculusu Meta Quest 2, ako aj po konzultáciách s doménovým expertom a jej autorom RnDr. Martinom Vaľom, PhD., sme zaznamenali hneď niekoľko nedostatkov. Pre adekvátne zhodnotenie týchto nedostatkov sme predovšetkým použili 11 bodov uvedených v kapitole 1.13.

Ak sa pokúsime zhodnotiť prvý bod, ktorý opisuje usporiadanie [49], priestor je rozumne rozložený a priestranný. Používateľ nemá pocit úzkosti a nepohodlia. Avšak, môžeme si všimnúť nejaké neprirodzené prostredie, ktoré pôsobí zmätene a vyvoláva pocit neistoty. Pre zlepšenie UX by bolo vhodné zvoliť prostredie, ktoré simuluje reálny pracovný priestor, prípadne ponúknuť voľbu rôznych prostredí a nechať výber na používateľa. Tieto prostredia by mali tiež obsahovať simuláciu pracoviska, aby sa zabránilo tomuto neprirodzenému efektu. Prvý bod zahŕňal aj zbytočnú fyzickú námahu, ktorá je potrebná pre orientáciu v prostredí a získanie pomoci pri prvom stretnutí s komponentom NDMVR. Stena, na ktorej je vykreslená kombinácia tlačidiel pre zobrazenie manuálu, je za pohľadom používateľa, čo nie je pohodlné. NDMVR podporuje zmeny výšky binov histogramu,

čo je veľmi výhodné, pretože používateľ môže zmenšiť veľkosť binov a tým zväčšiť svoje zorné pole a ušetriť si námahu pri otáčaní hlavy. Nevýhodou je, že rozsah týchto zmien nie je dostatočný na to, aby bol histogram pohodlne analyzovateľný zblízka pri najnižšej výške. Táto najnižšia výška je stále pomerne vysoká, aj keď je nastavená na najnižšiu hodnotu. Napriek tomu prostredie dodržiava zásadu "menej je viac".

Čo sa týka porušenia prvého bodu, dochádza hneď k niekoľkým problémom, ktoré si pre úplnosť zhrnieme:

- Neprirodzené prostredie (pozadie)
- Nepraktické umiestnenie indikátorov pomoci
- Nedostatočný rozsah zmenšenia výšky binov

Po zhodnotení druhého bodu, ktorý opisoval flexibilitu a prispôsobenie aplikácie používateľom, sme identifikovali niekoľko problémov. Jedným z nich je fixné ovládanie, ktoré neumožňuje voľbu ovládania pre ľavákov a prispôsobenie a voľné mapovanie funkcií tlačidiel. Ďalším nedostatkom je absencia možnosti zmeny výšky pohľadu používateľa pre rýchlejšiu analýzu dát, čo je nepodporované v Oculus verzii. Taktiež chýbajú možnosti zmeny prostredia, témy a rozsahu zobrazenia binov, pretože tieto funkcionality buď nie sú podporované prostredníctvom Tabletú, alebo chýbajú v komponente. Ďalej, nie je možné voľne rozmiestniť alebo vybrať vhodné rozloženie prvkov prostredia. Vzhľad scény by tiež mohol byť ešte lepšie prispôsobený potrebám danej domény. Po konzultácii sme dospeli k záveru, že by bolo vhodné zobraziť histogram realistickejším spôsobom, ktorý viac korešponduje so skutočnosťou.

Okrem toho chýba efektívny pracovný režim, ktorý by zahŕňal skratky a gestá, umožňujúce používateľovi vykonávať operácie, ako napríklad zmenu výšky binov, rýchlejšie a pohodlnejšie bez použitia Tabletú. Pre začiatočníkov by bolo prospešné mať k dispozícii krátky tutoriál s odkazmi a návodmi, ktoré im vysvetlia funkcionality počas ich prvých skúseností s aplikáciou.

Konkrétne porušenia a nedostatky druhého bodu sú:

- Fixné ovládanie, ktoré nie je prispôsobiteľné používateľom.
- Chýbajúci režim pre začínajúcich používateľov, napríklad v podobe tutoriálu.
- Absencia režimu pre skúsených používateľov, ktorý by umožňoval pokročilé funkcie, ako napríklad gestá.

- Nedostatočná podpora možností tabletu, ako je zmena témy a rozsahu zobrazenia.
- Chýbajúca možnosť výberu prostredia aplikácie (scény).
- Chýbajúca možnosť zmeny výšky pohľadu používateľa pri Oculus verzii
- Potreba realistickejšieho zobrazenia histogramu.
- Nedostatok prispôbitelnosti a možnosti umiestnenia prvkov prostredia.

Tieto aspekty je dôležité adresovať a vyriešiť, aby sme vytvorili lepšiu a prispôbitelnejšiu používateľskú skúsenosť.

Vyhodnotením tretieho bodu sme dospeli k zisteniu niekoľkých porušení a nedostatkov. Ide skôr o možnosť nepríjemného stretu používateľa s binmi histogramu než o porušenie osobného priestoru. Niektoré biny sa často nachádzajú vo výške hlavy používateľa, čo môže byť nepríjemné, najmä pri určitej vzdialenosti. Avšak používateľ sám určuje svoju polohu voči binom a celkovému histogramu.

V komponente chýbajú zákony fyziky, napríklad nemožnosť prejsť cez plátno, čo nie je podporované. Tento nedostatok ovplyvňuje prirodzenosť a interakciu s prostredím. V prípade histogramu by však táto možnosť mohla byť kontraproduktívna, pretože prechod skrze biny uľahčuje analýzu a pohyb v histograme.

Taktiež by bolo vhodné optimalizovať úvodný zorný uhol používateľa a umiestnenie používateľa samotného na dosiahnutie lepšej pohodlnosti a vhodnejšej vzdialenosti. Aktuálna poloha a zorný uhol môžu byť suboptimálne, čo negatívne ovplyvňuje používateľskú skúsenosť.

Po zhodnotení tretieho bodu sme identifikovali dva problémy:

- Neaplikované zákony fyziky: Implementácia zákonov fyziky, konkrétne pre plátno.
- Umiestnenie základného pohľadu

V súlade s princípom "menej je viac" číslo štyri sme sa rozhodli preskúmať centrum ovládania práce a zobrazenia informácií, ktorým je umiestnený tablet na ľavej ruke používateľa. Pri tomto riešení je potrebné zvážiť aj možnosť umiestnenia tabletu do opačnej ruky, čo v súčasnej verzii nie je podporované. Po nasadení headsetu sme zistili, že okno, ktoré malo slúžiť na zmenu zdroja dát histogramu, nebolo dostatočne popísané (názov funkcionality) a používateľovi nebol jasne vysvetlený jeho účel. Samotné prvky ovládania a ich možnosti by sa mali zdokonaľiť, aby poskytovali prehľadnejší dizajn a prípadne jednoduchšie ovládanie.



Týmto zlepšením sa zabezpečí, že používateľ bude mať väčšiu flexibilitu pri ovládaní aplikácie a zobrazovaní informácií, čo prinesie významné zlepšenie celkovej používateľskej skúsenosti.

Štvrtý bod identifikoval:

- Tablet VR je miestami považovaný za neprehľadný a neintuitívny

Piaty bod sa zameriava na identifikáciu slabých stránok hardvéru, ktoré musíme zohľadniť pri vývoji komponentu pre VR headsety. Naším cieľom je dosiahnuť stabilné a vysoké snímky za sekundu (FPS) vo porovnaní s bežnými aplikáciami. Okrem toho sa snažíme vylepšiť kvalitu prostredia a zobrazovaného kontextu. Je nevyhnutné zohľadniť technické možnosti jednotlivých zariadení a ich silné stránky, aby sme mohli poskytnúť optimálny a stabilný výkon FPS na každom zariadení. Túto časť vylepšenia sme identifikovali v časti 1.12.

Momentálne nemáme presné sledovanie počtu snímok za sekundu, ale vieme, že FPS nie je konštantný a často kolíše, čo spôsobuje trhliny v obraze, najmä pri 360-stupňovom otáčaní. Grafické a estetické prevedenie scény je nedostatočné, čo znižuje celkový dojem. Hoci pracujeme s A-Frame knižnicou, ktorá nám poskytuje obmedzené možnosti, budeme sa snažiť odstrániť tieto nedostatky a zlepšiť grafické zobrazenie scén, aby sme dosiahli lepší celkový dojem aplikácie.

- Nedostatočné grafické a estetické prevedenie scény
- Nedostatočné rozlíšenie pri obrázkoch
- Nekonštantné FPS
- Chýbajúca optimalizácia obrazu pre zariadenia

V komponente NDMVR sme zistili, že chýba úvodný tutoriál a dostatok priestoru na naučenie sa ovládania a funkcionality. To nás priviedlo k bodu číslo 6, ktorý sa zaoberá poskytovaním pomoci a navigácie pri akciách v aplikácii. Chýbajúce šípky, ukazovatele a jasné možnosti znížili užívateľský zážitok, najmä pre menej skúsených používateľov VR. Používatelia sú nútení sa sami učiť a objavovať funkcie aplikácie, čo je pre komponent určený na prácu neefektívne.

Dôležité je mať na pamäti, že VR je pre väčšinu používateľov stále nový rozmer, ktorý predstavuje nové možnosti. Podľa Nielsona v časti 1.8 je dôležité uľahčiť používateľom prácu v VR a venovať pozornosť ich potrebám. Z toho vyplýva aj bod 6, ktorým sa upozorňuje na možné nebezpečenstvá a následky akcií, ktoré

v DMNVR chýbajú. Absencia grafického znázornenia a pomoci pri určitých akciách, ako aj nedostatok inštrukcií pri práci s binom, tabletom alebo v prostredí, je potrebné riešiť.

Okrem toho sme zistili, že výberové tlačidlá použité v komponente sú založené na predvolených tlačidlách systému Oculus Meta Quest 2 VR headsetu. Avšak tieto tlačidlá nie sú pre nezalých používateľov intuitívne. Inštrukcie sú síce dostupné, ale iba po stlačení pomocného tlačidla. To spôsobuje, že používateli sa často cítia dezorientovaní a bez možnosti pomoci. Ovládanie tabletu na ľavej ruke je tiež zle zreteľné a vyžaduje hľadanie.

V záujme zlepšenia celkového užívateľského zážitku je potrebné zabezpečiť lepší tutoriál, jasné navigačné prvky a intuitívne ovládanie, aby sa používatelia mohli ľahšie naučiť a efektívne pracovať s komponentom.

- Chýbajúce grafické znázornenia ako forma pomoci a ukázania možností, prípadných nebezpečných akcií
- Nedostatočné zvýraznenie tlačidiel v prostredí VR pri nasadenom headsete

V komponente NDMVR chýbajú zvukové efekty a hlbšie zapojenie senzitívnych zmyslov prostredníctvom vibrácií. Na základe bodu sedem sme sa rozhodli tieto nedostatky zohľadniť.

- Chýbajúci priestorový zvuk
- Chýbajúce vibrácie

Porušenia sme identifikovali aj v bode osem, ktorý by mal rešpektovať zákonitosti reálneho sveta, pričom jednou z nich je nerelevantný pohyb. V aktuálnej verzii NDMVR je čiastočne implementovaný pohyb, avšak obmedzuje sa iba na pohyb vpred, vzad a doprava, dolava. Pohyb je pevne definovaný a nie je synchronizovaný s uhlom pohľadu používateľa nosiaceho VR headset. Tento problém priamo súvisí s bodom 10 v časti 1.13, ktorý hovorí o dôležitosti poskytnutia kontroly používateľovi. Používateľ je schopný pohnúť sa len vpred a vzad po osi x bez ohľadu na smer, ktorým sa pozerá, čo nezodpovedá prirodzenému správaniu, ktoré by používateľ očakával.

Okrem toho chýba forma ochrany a obmedzenia pred nevhodným použitím, napríklad stanovenie určitej vzdialenosti odchýlky používateľa od úvodnej polohy na mape a znemožnenie opustenia mapy. Nerelevantný pohyb vzhľadom na pohľad používateľa obmedzuje možnosť slobody a rozhodovania v závislosti na pohybe.

- Nedostatočný a nerelevantný pohyb
- Definovaná vzdialenosť pracovného priestoru

Bod číslo 9 sa venuje konzistentnej spätnej väzbe, ktorá je v NDMVR miestami porušená, obzvlášť pri používaní VR tabletu, kedy thumbstick controller reaguje oneskorene alebo preskočí položku zoznamu. Taktiež je nutné zvýšiť citlivosť stlačenia tlačidiel.

- Nedostatky v ovládaní možností výberu vo VR tablete

Tieto spomínané problémy výrazne ovplyvňujú UX a mali by byť riešené pre dosiahnutie lepšej používateľskej skúsenosti. Každý aspekt je dôležitý, či už v tejto práci, alebo pre nedostatok času, v budúcnosti by sa mal adresovať a vyriešiť, aby sme vytvorili lepšiu a prispôsobiteľnejšiu verziu NDMVR.

## 3 Zmeny vedúce k lepšiemu UX

---

V tejto časti práce sa zameriame na návrh a implementáciu našich riešení pre analyzované nedostatky komponentu NDMVR v časti 2.1.2. Pretože naše riešenia budeme implementovať do už vytvoreného a zabehnutého projektu, na ktorom sa podieľali aj autori prác [56] a [55], technológie, ktoré budeme používať, sú fixne dané. Riešenie bude implementované s využitím knižnice JSROOT<sup>1</sup> a webového frameworku A-Frame<sup>4</sup> spolu s ReactJS<sup>3</sup>.

NDMVR je iba nadstavbou celého projektu NDM, ktorý je oveľa rozsiahlejší. Návrh vždy pozostával z analýzy aktuálneho projektu a jeho súčastí, ktoré súvisia s naším problémom. Pri implementácii sme sa snažili dodržiavať existujúce konvencie v projekte NDMVR, vrátane jednotných názvov, umiestnenia funkcií v správnych komponentoch a použitia existujúcich funkcií z projektu NDM core pre vzájomnú komunikáciu medzi rôznymi zobrazeniami.

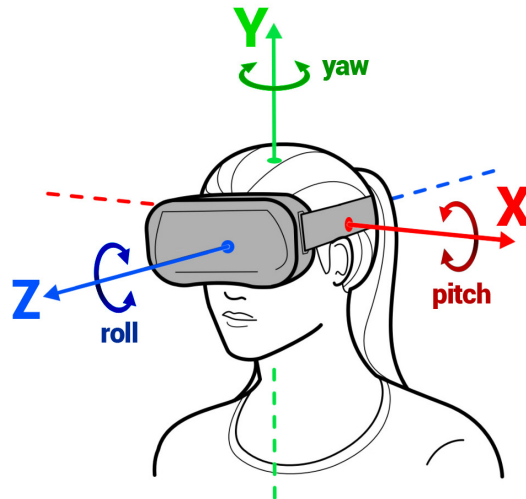
### 3.1 Nedostatočné možnosti pohybu

Jedným z nedostatkov NDMVR bolo nerelatívne pohybovanie, spôsobujúce dokonca pri jeho používaní náznaky kinetózy. Tento problém sme sa rozhodli odstrániť ako prvý, pretože najviac degradoval UX. Venovali sme sa mu najskôr.

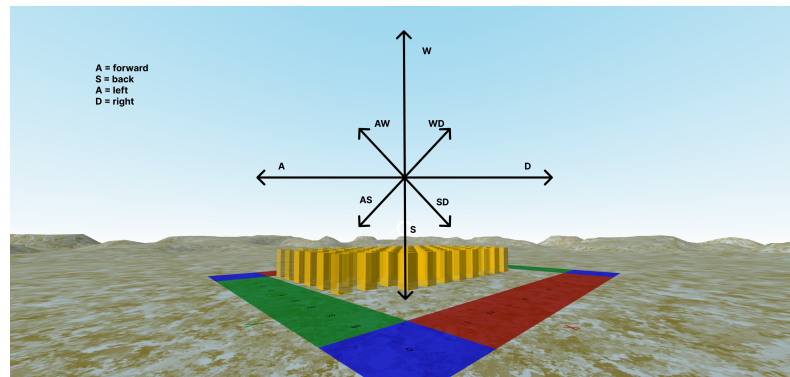
#### 3.1.1 Spôsob odstránenia nedostatočného pohybu

Zamerali sme sa na existujúce riešenia. Zistili sme, že A-Frame nepodporuje pohyb vo stereoskopických zariadeniach, hoci je už podporovaný v PC zobrazení. Pokračovali sme v nedostatočnej implementácii od autora práce [55] a narazili sme na viacero problémov. Komponent `leftOculusController` bol neprehľadný a obsahoval všetky funkcionality ľavého Oculus controllera. Prvým krokom bolo porozumieť samotnému pohybu a jeho reprezentácii pomocou osí. Pomohol nám obrázok 3.1, ktorý ukazuje reprezentáciu osí v trojrozmernom priestore.

Ďalšou otázkou bolo, ako zabezpečiť diagonálny pohyb prostredníctvom dvoch



Obr. 3.1: Position + orientation in 3D space (Oculus)(zdroj: [58] )



Obr. 3.2: Diagonálny pohyb po osiach na neskôr nami pridanej scéne moon

osí, konkrétne osí x a z. Odpoveď sme našli v obrázku 3.2<sup>1</sup>, ktorý znázorňuje reprezentáciu diagonálneho smeru pomocou osí x a z.

Taktiež sme museli zistiť, na akú udalosť reaguje controller pri stlačení thumbsticku, aby sme správne zachytili udalosť pohybu. Po dostatočnom zoznámení sa s týmito problémami sme sa rozhodli doplniť Martinovo riešenie. Chceli sme, aby naše nové riešenie zahŕňalo pohyb po všetkých osiach vrátane správneho smeru stlačenia thumbsticku na Oculus controle. Pridali sme tiež možnosť diagonálneho pohybu.

Pri získavaní relatívneho natočenia voči pohľadu kamery sme narazili na niekoľko problémov. Komponent `leftOculusController` len zachytával udalosti controllera, ktorým bola akcia priradená v súboroch napísaných prostredníctvom Reactu. To sa zdalo zložitejšie a zdĺhavejšie, pretože vytvorenie samotného komponentu A-Frame zahŕňa funkciu `tick`, ktorá by nám uľahčila prácu a zvýšila rých-

<sup>1</sup><https://answers.unity.com/questions/1287860/how-do-i-get-diagonal-3d-movement.html>

losť odozvy. Táto funkcia je volaná pri každom obnovení snímky obrazovky, teda pri každom frame. Pri pohybe by sme tak mohli aktualizovať súradnice kamery každý frame.

V našom riešení reprezentuje kamera pohľad používateľa z prvej osoby. Rozhodli sme sa vytvoriť komponent pohybu pomocou A-Frame a použiť jeho funkciu `tick`, čo by zlepšilo rýchlosť odozvy a samotný zážitok UX. Aj napriek tomu sme zdokonalili Martinovo riešenie a implementovali ho. Zároveň sme sa však rozhodli použiť samotný komponent A-Frame a hľadať ďalšie spôsoby efektívneho a čo najmenej fixného riešenia problému pohybu. Chceli sme, aby pohyb bol voľný a obsahoval čo najmenej obmedzení, na rozdiel od našej pôvodnej implementácie. Naše pôvodné riešenie obsahovalo pevne stanovené podmienky, ktoré zachytávali rôzne polohy thumbsticku a často spôsobovali, že thumbstick sa ocitol v nezadanom stave, čo viedlo k tomu, že nereagoval alebo reagoval pomaly, až po dosiahnutí definovaného stavu. To platilo aj pre fixne daný pohyb po osiach  $x$ ,  $z$  a prípadne  $y$ . Na základe týchto neuspokojivých výsledkov sme vedeli, že toto nie je správny prístup, a najlepším riešením bude komponent, ktorý bude zachytávať pohyb thumbsticku a sám bude mapovať jeho smer. Chceli sme minimalizovať preddefinované stavy a polohy. Preto sme sa rozhodli, že na základe udalosti stlačenia thumbsticku vytvoríme nový vektor pohybu, ktorý bude zahŕňať aj rotáciu objektu. Potom plánujeme tento vektor pripočítať k aktuálnemu pohybu a zmeniť súradnice objektu kamery v NDMVR. Inšpiráciou nám bolo aktuálne riešenie <sup>2</sup>

### 3.1.2 Odstránenie nedostatkov pohybu

Novým riešením problému bolo použitie už existujúceho komponentu,<sup>3</sup> ktorý splňal naše požiadavky. Bol to samostatný komponent A-Framu, využívajúci funkciu `tick` na pravidelnú aktualizáciu polohy kamery. Spolu s vytvorením nového vektora pohybu a zahrnutím orientácie kamery. Tento komponent sme upravili pre naše účely tak, aby získaval aktuálne súradnice entity `cameraWrapper` (`<a-entity id="cameraWrapper">`), ktorá obsahuje pre nás kľúčové objekty ako kamera, hand controllery a VR tablet umiestnený v ľavej ruke používateľa. Entita tiež obsahuje aj laser pre voľbu binu a mnoho ďalších objektov a komponentov, ktoré má k dispozícii používateľ. Všetky objekty entity `cameraWrapper` boli závislé na zmene polohy, a preto bolo nutné pohybovať celou entitou.

<sup>2</sup><https://www.techagekids.com/2022/09/code-3d-vr-halloween-scene-meta-quest-a-frame.html>

<sup>3</sup><https://gftruj.github.io/webzamples/aframe/controls/oculus-thumbstick-controls.html>

Po získaní správnych súradníc objektu sme upravili komponent tak, aby thumbstick ľavej ruky defaultne slúžil pre pohyb v prostredí. Po stlačení gripu a otvorení okien pre voľbu poskytovaných možností sa stáva nástrojom voľby parametrov a ovládania funkcií ponúkaných tabletom. Túto časť sme implementovali tak, že sledujeme stlačenie gripu a aktívne okná tabletu pomocou funkcie `getActiveTool`. Ak je potrebné, aby thumbstick slúžil ako nástroj voľby, je komponent pohybu neaktívny. Ovládanie tabletu a ďalších funkcionalít ľavého ovládača je zahrnuté v komponente **left-controller-logging**. Tým sme zabezpečili, že náš upravený komponent **oculus-thumbstick-movement-controller** slúži výhradne pre pohyb v prostredí a je oddelený od ostatných operácií v NDMVR.

Náš upravený a použitý komponent **oculus-thumbstick-movement-controller** obsahuje množinu vlastností definovaných v objekte schémy. Tieto vlastnosti zahŕňajú zrýchlenie, režim letu, nastavenie horizontálnej a vertikálnej osi a povolenie/zákaz pohybu v určitom smere. Napriek tomu, že zrýchlenie pri desktopovej verzii je 50 a autor komponentu používa zrýchlenie 45, my sme sa rozhodli použiť hodnotu zrýchlenia 40. Tento krok sme urobili s ohľadom na to, že pohyb vo VR môže spôsobovať kinetózu [9], čo môže mať negatívny vplyv na používateľskú skúsenosť (UX). Preto sme komponent defaultne nastavili na zrýchlenie 40, ktoré, podľa našich testov a názoru, poskytuje najpríjemnejší zážitok a minimalizuje príznaky nevoľnosti.

Režim letu je zatiaľ defaultne zakázaný, bude použitý pri ďalšom vylepšení UX v NDMVR. Tento režim umožní sledovanie histogramu z výšky, čo uľahčí jeho analýzu.

Funkcia **init** nastavuje poslucháčov udalostí, konkrétne pohybu thumbstickom a stlačenia gripu. Funkcia **thumbstickMoved** sa pravidelne volá pri pohybe thumbstickom, zatiaľ čo funkcia **tick** aktualizuje polohu vo virtuálnom svete pri každej snímke. Úlohou funkcie **updateVelocity** je vypočítať rýchlosť pohybu kamery na základe polohy thumbsticku a času, ktorý uplynul od poslednej snímky. Funkcia **getMovementVector** vypočíta vektor pohybu na základe získanej rýchlosti a uplynutého času. Na pohyb objektov v trojrozmernom priestore bola použitá knižnica **THREE.js**. Na rotáciu je využitá trieda **THREE.Euler**, ktorá reprezentuje rotácie v trojrozmernom priestore, a triedu **THREE.Vector3** na vytvorenie nového vektora. Následne je aplikovaná rotácia na smer pohybu objektu pomocou metódy **applyEuler()**, ktorá vráti nový vektor pohybu. Výstupom je nový vektor pohybu, ktorý berie do úvahy aktuálnu rotáciu. Funkcia **tick** potom aplikuje nový vektor pohybu na polohu kamery pomocou metódy **applyEuler(object3D.position.add(getMovementVector()))**. Táto metóda neustále pri-

dáva nové súradnice k aktuálnej polohe, čo zabezpečuje plynulejšiu zmenu pozície kamery.

## 3.2 Nedostatky NDM tabletu

Ďalším nedostatkom NDMVR, ktorý sme viackrát spomenuli v kapitole 2.1.2, bol tablet. Používateľovi nedával dostatočnú slobodu, nepodporoval potrebné funkcionality a jeho ovládanie voľby bolo nedostatočné. Okrem toho, mnohé funkcionality, ktoré boli pôvodne podporované, prestali fungovať v dôsledku rôznych aktualizácií systému. V tejto časti sme sa rozhodli pridať možnosť zmeny témy v rámci tabletu, vykonať opravu nefungujúcich funkcií, ako aj drobné zmeny v desktopovej verzii tabletu a zlepšiť možnosť ovládania tabletu prostredníctvom Oculus thumbsticku.

### 3.2.1 Možné vylepšenia NDM tabletu

Po dôkladnom posúdení sme zistili, že funkčná je iba jedna možnosť - zobrazenie základných informácií o binách. V aktuálnej verzii to však nebolo podporované, pretože prebehli veľké zmeny a úpravy v jadre projektu, najmä v komunikácii medzi VR zobrazením a desktopom. To spôsobilo, že niektoré funkcionality prestali fungovať.

Hoci bola implementovaná funkcionality jednotlivých okien, zistili sme, že nefunguje správne. Pri výbere zdroja sa okno zobrazovalo nesprávne a ak výber zlyhal, zostalo prázdne. Predchádzajúca časť, ktorá sa venovala pohybu, obsahovala komponenty pre pohyb a voľbu možností, ako aj ovládanie tabletu. Tieto chyby sme vylúčili. Po zhodnotení sme dospeli k záveru, že chyba je v získaných dátach. Nesprávne dáta mohli spôsobiť nesprávne zobrazenie a nedostupnosť funkcionality zväčšovania a znižovania binov. Opravu týchto chýb sme zahrnuli do implementácie.

Venovali sme sa aj oknu tabletu, ktoré slúži na zmenu veľkosti binov. Po dôkladnom posúdení aktuálneho ovládania sme identifikovali niekoľko nedostatkov. Pohyb thumbsticku bol fixne mapovaný na určitú hodnotu a ovládanie bolo pomalé. Často sa stávalo, že nedosiahneme požadovanú hodnotu a nedostaneme žiadnu reakciu.

Podobné nedostatky sme zaznamenali aj pri výbere možností v oknách tabletu. Rolovacie menu možností často nereagovalo a keď už reagovalo, zmena položky sa prejavila až pri úplnom pohybe thumbsticku nahor alebo nadol. Nedostatkom bol taktiež krátky čas spracovania, ktorý bol nastavený na 100 milisekúnd.



Pri rýchlych pohyboch sa informácia spracovala dvakrát, čo viedlo k posunu o dve položky v zozname namiesto jednej. S cieľom zlepšiť UX chceme pridať funkcionality zmeny témy zobrazenia binov. Táto funkcionality už bola implementovaná v desktopovej verzii, ale vo VR verzii tabletu chýbala. Vytvoríme nové okno VR tabletu, ktoré bude obsahovať preddefinované témy reprezentované názvami. Používateľ bude mať možnosť vybrať tému pomocou ovládania. Pri zmene témy sa vďaka použitiu Reactu a **StoreContextu** okamžite aktualizuje histogram s novou farbou binov, čím sa minimalizuje potreba kompletného obnovovania scény.

### 3.2.2 Odstránenie nedostatkov tabletu

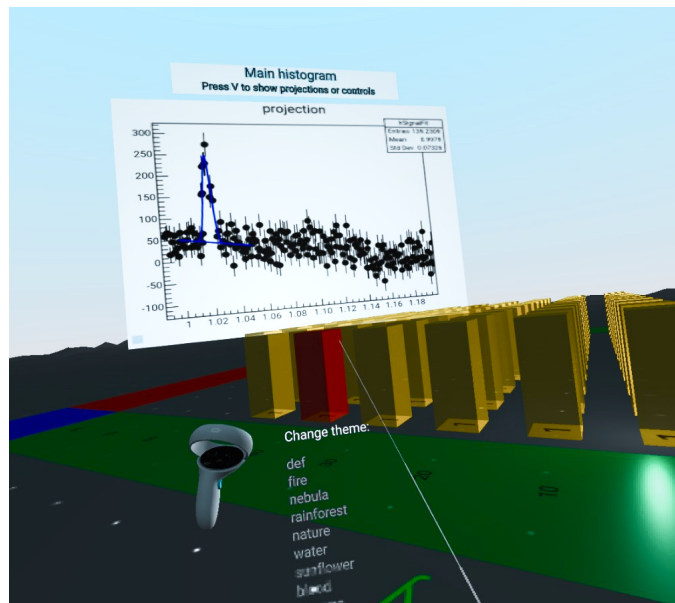
Dôvodom nefunkčnosti okien VR tabletu bola stará implementácia v komponente `NdmVrCamera`, ktorá obsahovala `<StoreContext.Provider>`, ktorého dáta sme získali po zavolaní `useContext(StoreContext) . stores`.

V našej implementácii sme tento problém odstránili. Po získaní správnych dát sme vytiahli už priamo dáta obsahujúce zdroje histogramu, ako aj dáta tém. Tieto dáta sme prihlásili na odber pre získanie aktuálnych informácií o zmenách.

Do funkcie **handleCommunication** sme doplnili funkcionality pre dáta témy. Funkcia zabezpečuje akcie "next" alebo "prev" pre výber možností. List tém prechádzame položku po položke. Ak sa dostaneme na koniec zoznamu, zmeníme index, čo nás vráti opäť na začiatok celého zoznamu. Tým máme zabezpečené dynamické menenie "theme" na základe zadaného príkazu "next" alebo "prev". Téma, na ktorú sa prepneme, sa stáva aktívnou. Príkazy "next" alebo "prev" odchyťme na základe rozhodnutí vykonaných v `OculusControlleri`, kde sme implementovali funkciu **dataThemeChanger()**, ktorá reaguje na pohyb thumbsticku. Pre úplnosť riešenia sme pridali aj funkciu **changeDataTheme()**, ktorá vytvára zobrazený zoznam (pomocou funkcie "map()") s jednotlivými témami a pre každú tému vytvára odkaz (element `<a>`) s názvom témy. Ak je téma aktívna (t.j. zvolená používateľom), názov témy je graficky odlišiteľný (zvýraznený).

Keď používateľ klikne na niektorý z odkazov, zmení sa aktívna téma na zvolenú tému a tým sa zmení aj vzhľad binov histogramu. Zobrazenie okna je implementované ako výpis položiek zoznamu reprezentovaných elementami `a-textu`. Element nám ponúka knižnica `A-Frame`, slúži na zobrazenie a výpis textu v prostredí VR. Toto zobrazenie je identické pre obe okná - zdroje a témy zobrazenia.

Časť implementácie sa sústredila aj na nedostatky a chyby ovládania tabletu. Tie sme odstránili zmenou reakcií na eventy controllera. Zmena položky sa vykoná pri akomkoľvek pohnutí thumbsticku. Je tam minimálne slepé pásmo. Ak sa thumbstick pohne z polohy určujúcej nečinnosť, funkcie pre zachytenie reakcie



Obr. 3.3: Implementovaná zmena témy binov v Oculus zobrazení

implementované v OculusControlleri zareagujú. Tým sme zabezpečili príjemnejšiu spätnú väzbu v podobe prepnutia sa na druhú položku. Časovač, ktorý handluje akciu zmeny elementu zoznamu, sme nastavili na novú hodnotu, aby nedochádzalo k preskočeniu položky zoznamu. Implementáciou takmer celého rozsahu mapovania pohybu thumbsticku po osi y sme pridali možnosť pohodlnejšieho listovania, ako aj pravidelného odchyťavania eventu stlačenia a jeho reakcií odosielaných na spracovanie. Akcie zväčšovania binov histogramu, ako aj listovania sú nastavené tak, aby zmena bola vykonaná pri vychýlení thumbsticku do jeho aktívnej polohy nastavenej na hodnotu vychýlenia väčšiu ako 0.1 a v prípade opačného smeru menšiu ako -0.1. Toto pásmo sme tam ponechali z dôvodu možného problému ponechania palca na thumbsticku pri jeho vrátení do nečinnnej polohy 0. Vzhľadom na prítomnosť palca alebo menšiu ruku s krátkymi svalmi by mohlo dôjsť k vychýleniu, kedy hodnota nebude presne 0, a mohlo by dochádzať k posúvaniu sa v zozname. To by používateľovi pôsobilo rušivo, keďže by aplikácia nespoupracovala. Z tohto dôvodu sme ponechali malé slepé pásmo na intervale (-0.1;0.1).

Oknám tabletu sme pridali názov danej funkcionality, ako aj upravenú viditeľnosť aktuálne zvolenej položky, ako je možné vidieť na obr. 3.3.

### 3.3 Zmena veľkosti binu a mierka zobrazenia histogramu

NDMVR umožňuje zmenu veľkosti binov histogramu a počtu binov v histograme. Tieto možnosti sú dôležité pre analýzu histogramu. Avšak, pôvodne implementované možnosti zmeny veľkosti binov a mierky histogramu boli nedostatočné.

Prvým nedostatkom bolo, že zmena veľkosti binu bola realizovaná pomocou fixnej hodnoty inkrementu alebo dekrementu 0.1. To malo za následok nedostatočný krok, keďže pri priblížení sa k hodnote 0.1 bola jedinou možnou hodnotou pre zmenšenie binu 0. Používateľ stratil pomerne dôležitý rozsah medzi hodnotami 0.1 a 0. Taktiež sme zistili, že po opätovnej zmene veľkosti mohla hodnota binu stať sa zápornou, čo spôsobovalo jeho premiestnenie pod scénu, čo sa prejavilo tým, že malá časť binu sa objavila priamo v strede scény, čo bolo nežiaduce.

Preto chceme pridať nový krok s hodnotou 0.01, ktorý spomalil zmenu veľkosti. Toto riešenie však nebude bolo dostatočné pre headset, kde zmena veľkosti binu je realizovaná prostredníctvom thumbsticku, čo má za následok pomerne veľmi nízku citlivosť. Preto sme zvážili dve riešenia: mapovanie pohybu thumbsticku a použitie rôznych intervalov a skokových hodnôt.

Mapovanie thumbsticku sa ukázalo ako neideálne, preto sme sa rozhodli použiť druhé riešenie s rôznymi intervalmi a skokovými hodnotami. Tým sme dosiahli vyššiu citlivosť a presnosť pri používaní thumbsticku. Thumbstick je možné držať v konštantnej polohe a veľkosť binu sa bude postupne meniť s rôznou rýchlosťou. Po uvoľnení thumbsticku bude bin dosahovať požadovanú veľkosť. Skoková hodnota pri hodnote 0.1 bude nastavená dostatočne citlivo, tak aby bolo ľahké dosiahnuť požadovanú veľkosť binu.

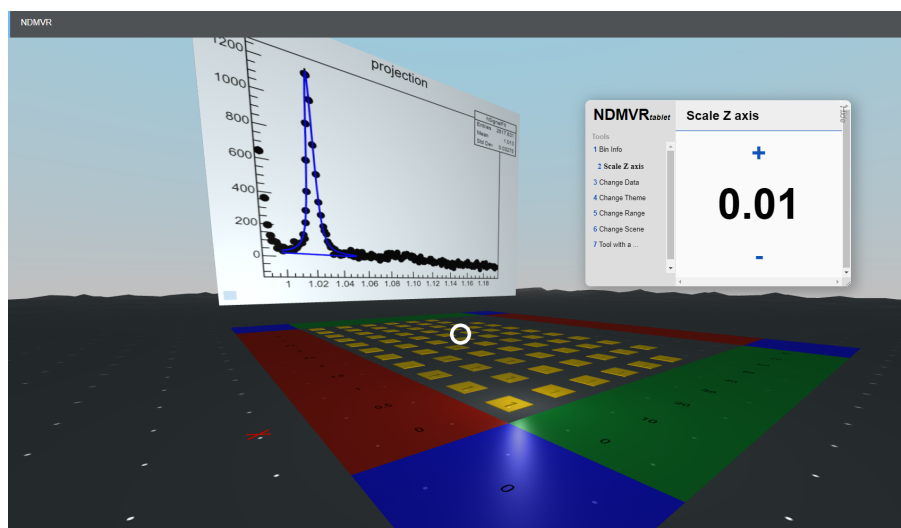
V oboch režimoch zobrazenia budeme kontrolovať minimálnu veľkosť binu a nedovolíme jeho pokles pod hodnotu 0.01, aby sme zabezpečili správne zobrazenie. Úplné zmiznutie binu sme zakázali, aby sme predišli zmäteniu používateľa a znemožneniu analýzy histogramu.

Doplnili sme chýbajúcu možnosť výberu mierky histogramu na VR tablete. Riešenie obsahuje viacero preddefinovaných mierok a nadväzuje na existujúce riešenie v desktopovom zobrazení. Zmena mierky histogramu je možná cez tlačidlo "Show Settings" na pravej strane NDMVR. Toto tlačidlo umožňuje výber preddefinovaných mierok, ktoré menia počet zobrazených binov. Nastavené možnosti mierky histogramu zostávajú nezmenené. Hodnoty mierky sú definované v súbore "ThProjection" a uchovávajú sa v premenných triedy "StoreContext". Predvolene sme po konzultácii s autorom projektu zvolili mierku 10. Nový ná-

stroj na desktopovom tablete bude označený “Scale Change”. Po jeho rozkliknutí je možné vybrať jednu z preddefinovaných mierok, ktorá sa automaticky nastaví. Pre Oculus tablet pridáme nové okno nástrojov s dátami mierky. Ovládanie zostane rovnaké, používa sa grip tlačidlo a thumbstick. Thumbstick má dve polohy - hore a dole, ktorými sa budú meniť položky v zozname.

### 3.3.1 Odstránenie nedostatkov zobrazenia Histogramu

Nedostatky v zmenách veľkosti binov boli odstránené upravením funkcie `scaleZaxis()`, ktorá je zodpovedná za zmenu veľkosti pre desktopový tablet. Funkciu sme rozšírili o sledovanie nasledujúcej hodnoty, aby bolo možné včas upraviť skokovú hodnotu. Sledovanie sme dosiahli odčítaním hodnoty 0.01 od aktuálnej veľkosti binu. Túto hodnotu sme zaokrúhlili na dve desatinné miesta a porovnávali sme ju s hodnotou 0.1, ktorá určovala skokovú hodnotu. Pri zväčšovaní veľkosti sme postupovali rovnako, s použitím znamienka plus. Tým sme zabezpečili správne a včasné nastavenie skokovej hodnoty v oboch prípadoch. Zaokrúhľovanie bolo nevyhnutné, pretože sme zistili, že pri opakovaných zmenách veľkosti sa vyskytuje chyba spojená s nepresnými desatinnými miestami. Zaokrúhlením sme odstránili rozširujúcu sa nepresnosť. Tiež sme do funkcie pridali premennú, ktorá určuje počet desatinných miest pre zobrazenie. Tým sme sa vyhlili zbytočným desatinným miestam, ktoré by mohli zmiatť používateľa. Pri znižovaní veľkosti sme obmedzili minimálnu hodnotu na 0.01.



Obr. 3.4: Minimálne zmenšenie binu, pomocou NDMVR desktop tabletu

Implementácia zmeny veľkosti binov pomocou Oculus ovládača bola realizovaná v funkcii `setScaleForOculus()`. Táto funkcia sledovala hodnotu pohybu

thumbsticku zachytenú v komponente `OculusController`. Na základe tejto hodnoty sa vyhodnotila reakcia - kladná hodnota pre pohyb nahor a záporná hodnota pre pohyb nadol. Informácia bola následne prenesená do `tabletCommunicator`, kde bola odovzdaná funkciou `setScaleForOculus()`. Táto funkcia na základe aktuálneho stavu nastavila skokovú hodnotu. V prípade, že hodnota bola medzi 0,7 a 0,01, bola nastavená na 0,01. V prípade hodnôt medzi 0,01 a 0,001 bola nastavená na 0,001 a hodnotám vyšším ako 0,7 bola priradená hodnota 0,1. Rovnako ako pri desktopovom tablete, aj tu bola stanovená minimálna hodnota.

Pred implementáciou rozsiahlejšej časti nového nástroja tabletu sme sa rozhodli zmeniť predvolenú veľkosť zobrazenia. V komponente `Thprojection`, zodpovednom za správny render histogramu a scény, sme upravili atribút hooku `useState` pre premennú "scale" na hodnotu 10. Tým sme zabezpečili predvolenú veľkosť na požadovanú hodnotu. Táto hodnota bude možné zmeniť len prostredníctvom tabletu alebo tlačidla "Settings". V opačnom prípade bude hodnota stále nastavená na 10 binov.

Následne sme získali dáta zo `storeContextu` pomocou `useContext(StoreContext).stores`. Z týchto dát sme extrahovali informácie o mierke zobrazenia. V našom riešení sme tieto dáta prihlásili na odber podobne ako dáta týkajúce sa témy a zdroja dát. Tým sme získali aktuálne informácie, ktoré sú dôležité pre implementáciu komunikácie medzi tabletom a ďalšími časťami NDMVR. Snažili sme sa používať rovnaké konvencie a metódy ako pri implementácii zmeny témy a zdroja dát, aby sme zabezpečili maximálnu konzistentnosť a jednoduchosť pochopenia. To je dôležité najmä kvôli viacerým vývojárom pracujúcim na vývoji NDMVR.

Implementácia Oculus tabletu zahŕňala funkciu `handleCommunication()`, do ktorej sme pridali funkcionálnosť zmeny mierky zobrazenia. Rovnako ako v predchádzajúcich častiach, táto funkcia obsahovala akcie "next" a "prev", ktoré umožňovali prechod medzi predefinovanými hodnotami mierky zobrazenia. Akcie vyvolané na Oculus ovládači boli zachytené v komponente `OculusController`, kde funkcia `dataScaleChanger()` sledovala udalosti pohybu thumbstickom nahor a nadol a následne informovala `tabletCommunicator` o akcii "next" alebo "prev". Ovládanie thumbsticku a správne zachytenie jeho stavu bolo implementované rovnako ako pri zmene témy a zdroja dát. Tým sa zabezpečilo možnosť výberu bez ohľadu na definovanú silu stlačenia thumbsticku a nepretržité rolovanie možností výberu.

Funkcionálnosť bola tiež pridaná do desktopového tabletu pomocou funkcie `changeDataScale()`. V tejto funkcii sme prechádzali zoznam dát mierky zobrazenia a každý prvok zoznamu sme vykreslili ako "span" element tabletu. V prí-

pade, že bol prvok aktívny, zmenili sme jeho štýl písma, veľkosť, hrúbku a farbu. Po kliknutí na neaktívny prvok mierky zobrazenia sa táto hodnota stala aktívnou a príslušné prvky jej štýlu boli zmenené. Tým sme zabezpečili zobrazenie nového nástroja medzi existujúcimi nástrojmi desktopového tabletu. Toto sme dosiahli pomocou nového prvku, ktorý bol vykreslený ako samostatná funkcia. Štýl a veľkosť zobrazenia boli zachované podľa preddefinovaného dizajnu komponentu DesktopTablet, pričom sme pridali ďalší parameter pre zobrazenie mierky zobrazenia.

Týmto sme skrátili a upravili text tak, aby bol stručnejší a jasnejší.

## 3.4 Možnosť voľby prostredia

V tejto časti sme sa rozhodli zamerať sa na problém nedostatočnej voľnosti a mätkého prostredia pri používaní NDMVR. Aktuálne má používateľ presne definovanú jednu scénu, v ktorej analyzuje histogram a vykonáva všetky s ním spojené operácie. Výrazne by pomohlo odstrániť tento nedostatok možnosťou výberu scény. Používateľovi bude ponúknutá možnosť vybrať si scénu podľa vlastných preferencií, či už zameraných na estetiku alebo praktickosť. Výber scény by mohol uľahčiť analýzu.

### 3.4.1 Návrh scén komponentu

NDMVR obsahuje komponent NdmVrLaboratory, ktorý je zodpovedný za vykresľovanie scény. V tomto komponente bola definovaná použitá scéna prostredníctvom elementov A-Frame scény. Použitú štruktúru sme sa rozhodli zachovať a obohatiť o nové predefinované scény vytvorené z elementov A-Framu. Analýza na interne nám odhalila už existujúci komponent frameworku A-Frame s názvom A-Frame Enviroment, ktorý zabezpečuje vykresľovanie scén, postupným generovaním zvolenej mapy a jej prvkov. Požitím tohto komponentu zabezpečíme, rýchle a dynamické vykreslenie novej scény.

Rozhodli sme sa použiť dynamické vykreslenie scény, nakoľko staticky definované scény by potrebovali rozsiahlejší čas ako aj priestor pri ich implementácii. Projekt NDMVR by obsahoval množstvo zbytočných súborov so staticky definovanými scénami a ich prvkami. Čo by mohlo znížiť pochopiteľnosť a čitateľnosť práce budúcim vývojárom. Nami zvolený komponent priniesol riešenie v podobe jedného súboru, s rozsiahlou funkcionalitou a priamo definovanými šablónami scén.

Pre účely NDMVR chceme použiť predovšetkým vlastné scény, ktorých vytvorenie nám daný komponent, uľahčí, nakoľko obsahuje rôzne parametre prispôsobenia scény. Napríklad typ pozadia, typ povrchu, typ slnečného žiarenia a mnoho ďalších. Pomocou týchto parametrov by sme chceli zhotoviť niekoľko prevažne vlastných scén, prípadne použiť už ponúkané scény, po ich úprave.

Každá scéna, bude musieť obsahovať dostatočne veľký priemer rovného povrchu, pre správne zobrazenie histogramu a jeho plátna. Zasahujúce elementy scény, by mohli narušiť UX, ako aj znemožniť analýzu. Scény by nemali obsahovať prvky, ktoré môžu vyvolávať panický záchvat, alebo spôsobovať akúkoľvek formu šoku. Scény budú volené prostredníctvom tabletu, kde možnosť zmeny, ako aj zobrazenia, bude realizované prostredníctvom už vyššie spomínaných spôsobov v časti 3.2.2, pre zachovanie konzistentnosti ovládania a zobrazenia.

Do výberu scén taktiež zahrnieme aktuálnu scénu, nakoľko scéna je pre aktuálnych používateľov NDMVR už dobre známa. Taktiež by sme chceli zrealizovať scénu simulujúcu prostredie CERNU<sup>4</sup>. Týmto prostredím, by sme odstránili problém opísaný v časti 2.1.2, opisujúci zmätenosť daným prostredím, ako tomu bolo v aktuálnej verzii. Benefitom bude opäť použitie Reactu, ktorý zabezpečí reaktívne zobrazenie zmeny prostredia, teda bez akejkoľvek nutnosti obnovenia stránky.

### 3.4.2 Implementácia výberu scény

Základ nášho riešenia tvorí komponent **aframe-environment-component**, ktorý generuje rozličné scény. Priestor sme pri použití komponentu vymedzili na hodnotu `playArea = 1.3`. Prvá scéna nesie označenie NDMVR, obsahuje pôvodnú defaultnú scénu bez nami vykonaných úprav, túto scénu sme obohatili o podsvietenie podlahy a pridané nasvietenie histogramu.

Následne bola pridaná scéna imitujúca Cern, scéna pozostáva z jednoduchého pozadia tvoreného 360 stupňovým obrázkom rektora Cernu. Pre vytvorenie tejto scény, bolo nutné obohatiť komponent **aframe-environment-component** o funkcionality vytvorenia elementu oblohy obsahujúcej daný obrázok. Funkcionality sme zabezpečili pridaním nového parametra `background` do predefinovaných vzorov scén. Parameter bol definovaný ako reťazec, obsahujúci cestu k súboru. Ten sme ďalej v komponente získali a pridali do atribútu `zdroj` v entite obrázka. Tento parameter je nepovinný a daná funkcia pozadia sa vykoná len v prípade správneho nastavenia parametra na jednotnú farbu (`skyType = color`), farby

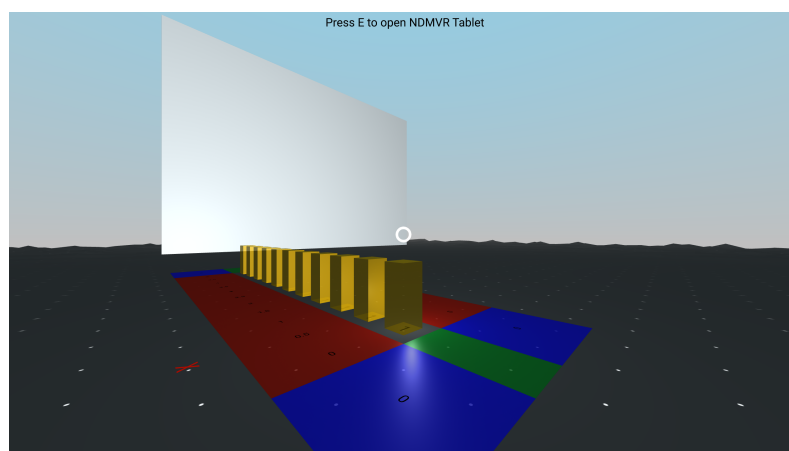
<sup>4</sup><https://home.cern/>

oblohy odporúčame nastaviť na hodnotu “none”. V prípade úpravy pozadia, je možné použiť konkrétne farby, ktoré ovplyvnia pozadia, ktoré poniesie nádych daných farieb.

Všetky ostatné parametre scény, ako svetlo, terén, tieň a mnoho ďalších ostali nezmenené a ich funkcionálnosť je zabezpečená aj pri pozadí vytvorenom z obrázka. Podlaha scény Cern obsahuje štvorce uľahčujúce analýzu a prospievajúce k lepšiemu vizuálnemu vyžitiu. Motív atmosféry, nebol použitý nakoľko je zahrnuté statické pozadie vo forme obrázka.

Podobným spôsobom bola vytvorená scéna nesúca názov Govorun, jej názov je odvodený od prvého hyper-konvergovaneho a 100% kvapalinou chladeného superpočítača (“horúcou vodou”) na svete. Pozadie tejto scény je tvorené práve týmto super výkonným počítačom. Nami pridaná scéna Labs simuluje ideálne laboratórne podmienky, nesvietený histogram, ideálne svetlo a zvolené farby s podsvietením podlahy.

Doplnením danej scény je scéna Labs2, ktorá je obohatená o zmenu farieb a motívu zapadajúceho slnka, taktiež je vypnutá funkcia tieňov, čo by malo zlepšiť výkon na rozdiel od scény Labs. Scéna Moon, zachytáva biele farby a prehľadnosť, jasnosť zobrazenia. Je obohatená o prvky náhodne rozmiestnených kameňov.

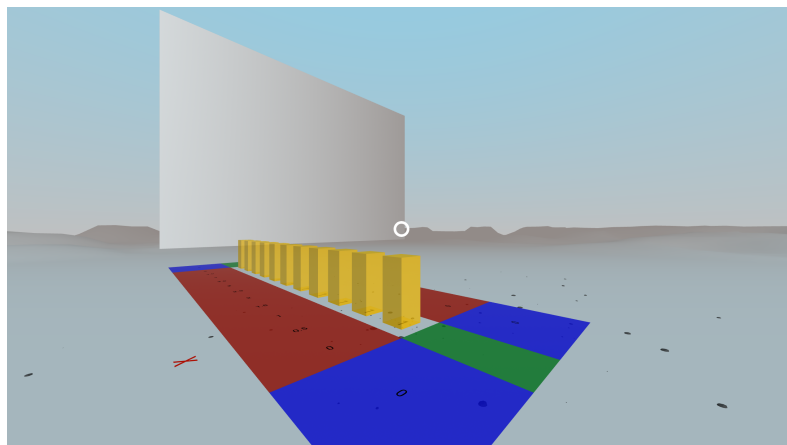


Obr. 3.5: Pridaná scéna Labs

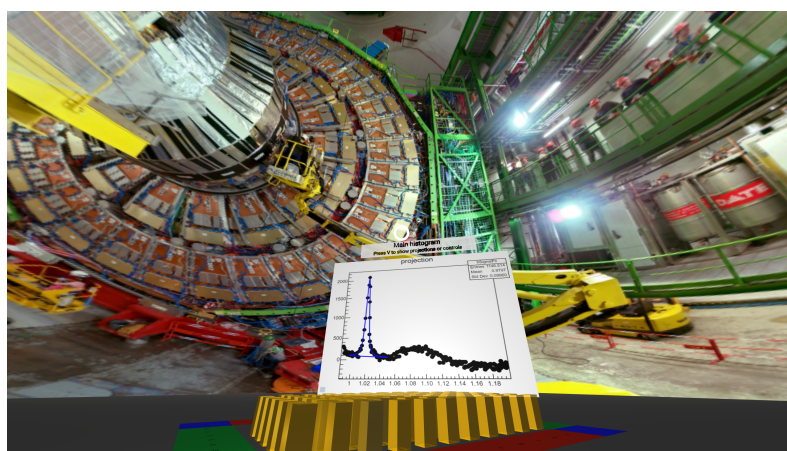
K dispozícii je taktiež scéna imitujúca zasnežený les s prvkami zasnežených stromov, ako aj objektov pozadia a trávy. Pre úplnosť sme pridalí scénu Snow s bielym motívom a štvorcovou podlahou, scéna je jedna z najjasnejších.

Na výber sme ponechali aj scény, ktoré vychádzajú z motívov ponúkaných komponentom, scény však nesú nami vykonané zmeny. Zmeny sú v podobe, random generovaného seedu, ktorý je upravený tak, aby objekty neboli generované v priestore zobrazenia histogramu, pričom počet generovaných objektov bol väčšine znížený, pre účely prázdnej plochy a prehľadnejšieho prostredia, ako aj

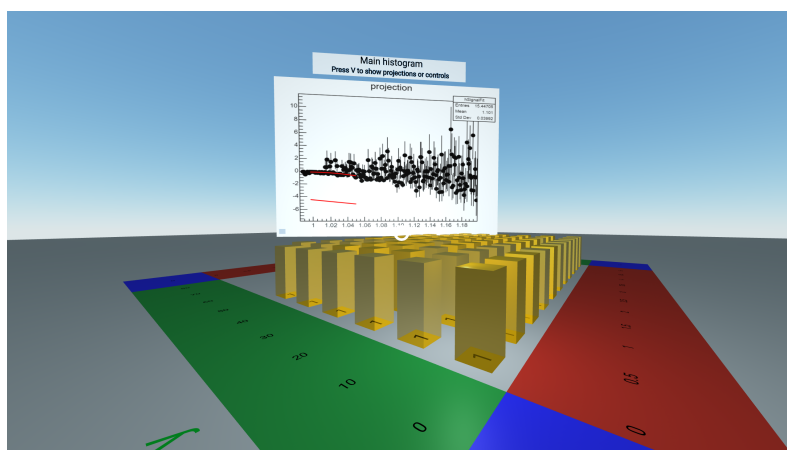




Obr. 3.6: Pridaná scéna Moon



Obr. 3.7: Pridaná scéna imitujúca pracovisko Cernu



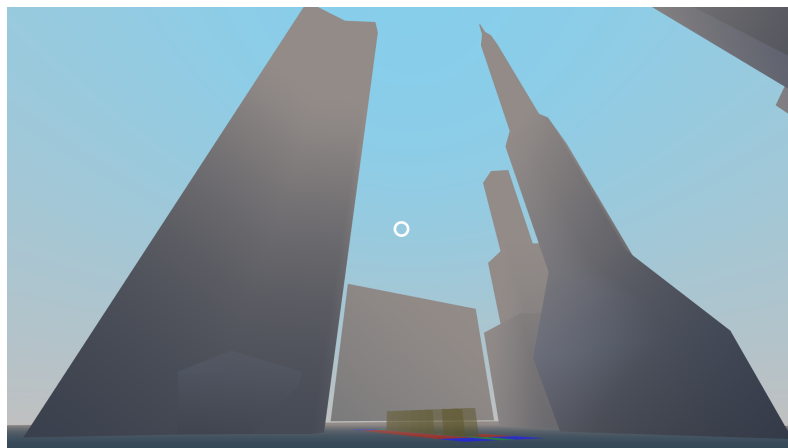
Obr. 3.8: Scéna imitujúca predošlú defaultnú scénu NDMVR

šetrenia výkonu.

Prvky generovania objektov pozadia boli nastavené tak, aby zodpovedali dostatočnej vzdialenosti pre účely zobrazenia väčšieho počtu binov. Mapy majú upravené prvky hmly, farieb, svetla, pre možnosti vyhovujúce analýze dát histogramu.

Všetky funkcionality, ako: pohyb, zmena dát histogramu, jeho témy a rozmedzia, sú podporované pri každom prostredí.

Komponentom zodpovedným za načítanie scény ostal **NdmVrLaboratory**, kde voláme pri tvorbe scény komponent, ktorému je formou props poskytnutý string s názvom prostredia, ten je dosadený ako argument komponentu preset. Spolu s argumentom je možné meniť parametre scény pri volaní komponentu **aframe-environment-component**. Defaultne je nastavná scéna Labs, táto scéna bude zobrazená pri štarte NDMVR, alebo v prípade prázdného props argumentu prostredia. Režimy zobrazenia, kde zmena témy, dát, ako aj mierky zobrazenia spolu so scénou nie je podporovaná, je defaultne nastavená scéna Labs.



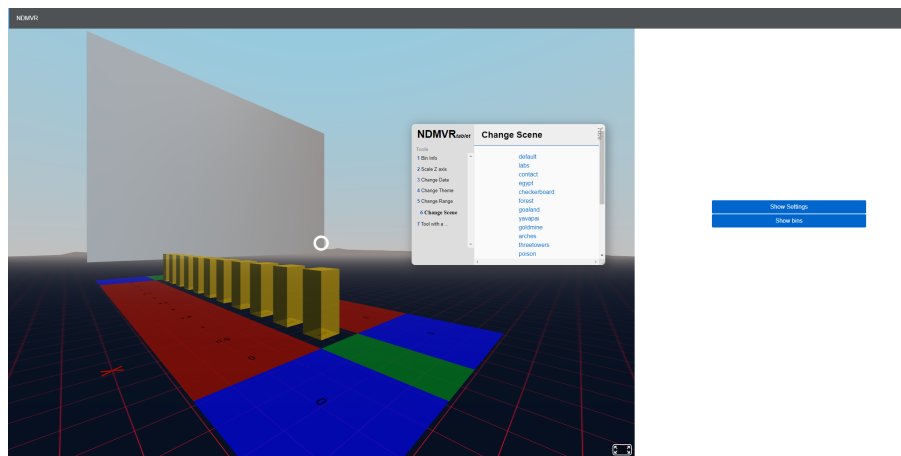
Obr. 3.9: Upravená scéna threetowers

Pre uloženie požadovanej scény a jej reaktívnu zmenu sme použili **StoreContext**, ktorého prvky store a states pre uloženie a načítanie stavu, sme obohatili o údaje nesúce názvy všetkých scén a údaj, obsahujúci aktuálne zvolenú scénu. V časti THprojection, bolo vytvorené pole reťazcov s definovanými názvami scén. Funkcia **setOptions()** prijíma toto pole, ktoré obsahuje zoznam možností/scén, ktoré sa majú zobraziť v aplikácii. Konkrétne, sa táto funkcia sa volá v useEffect hooku a má závislosť na niekoľkých premenných ako sú napríklad: (mainObject?.names, themes, scenes, ranges a views). Keď sa jedna z týchto premenných zmení, funkcia sa spustí a aktualizuje sa stav v aplikácii na základe nových možností scén.

Nato aby sme získali aktuálnu hodnotu scény, sme použili hook **useState()**, ktorý umožňuje uchovať a aktualizovať aktuálnu scénu v aplikácii. Konkrétne, scene je premenná, ktorá uchováva aktuálnu scénu v aplikácii. Počiatočná hodnota stavu je nastavená na reťazec 'labs', ktorý predstavuje názov východzej scény. **ExecutorStore('scene')** vracia aktuálnu hodnotu stavu scény, ktorá bola uložená

v stavovom hooku pomocou funkcie **useExecutorStore**. Túto aktuálne zvolenú scénu dosadíme za props v súbore **NdmVrHistogramScene**, odkiaľ je volaný komponent **NdmVrLaboratory**, nesúci informáciu o aktuálne zvolenej scéne, ktorého úlohu je jej pre nastavenie prostredníctvom komponentu **aframe-environment-component**.

V časti DesktopTabletu, boli pridané funkcie pre zmenu scény prostredníctvom VR tabletu, konkrétne funkcia, ktorá zabezpečuje správne spracovanie eventu stlačenia thumbsticku oculus controllera. Udalosť je zachytená v **leftOculusControlleri** a následne jej sú pridané akcie "next" a "prev" v **OculusControlleri**. Tieto akcie sú odchytené a handlované vo funkcii **handleCommunication** v súbore **DesktopTablet/index**, funkcia prechádza zoznam scén, pričom prednastaví aktuálne zvolenú scénu na novú zvolenú scénu. Na rovnakom mieste bola funkcia **renderOculusTablet** doplnená o zobrazenie možností scén. Funkcia vypíše scény, a označí názov aktuálnej scény. Pre ovládanie prostredníctvom desktopového tabletu, bola pridaná funkcia **changeDataScene**, ktorá zobrazí možnosti scén a po kliknutí na danú scénu zmení jej štýl pre indikáciu aktuálne zvolenej možnosti zoznamu, pričom aktualizuje premennú obsahujúcu aktuálnu scénu.



Obr. 3.10: Jedna z pridaných scén spolu s desktop tabletom slúžiacim pre výber scén

Pri generovaní scén sme narazili na problém zmiznutia pomocnej tabule. Tento problém spočíval v aktuálne nastavenej polohe pomocnej tabule obsahujúcej klávesovú/controllerovú skratku (v závislosti od zobrazenia na danom zariadení), pre zobrazenie možností NDMVR, ako aj aktuálne nastaveného histogram a jeho typ zobrazenia. Tabula bola pôvodne zobrazená v diaľke a jej veľkosť sa zdala byť primeraná, nakoľko jej poloha bola vzdialenejšia ako pohľad kamery a teda aj používateľa. Komponent pre generovanie scén, produkuje scénu menších rozmerov, z dôvodu šetrenia výkonu, čo je nevyhnutné pre udržanie stabilného chodu kom-

ponentu NDMVR. Pomocná tabuľa sa teda nachádzala mimo generovanú scénu, preto ju nebolo možné vidieť.

Tento nedostatok sme odstránili, jej bližšou vzdialenosťou. Usúdili sme, že momentálne NDMVR nepotrebuje tak rozsiahlu scénu, čo by mohlo viesť k potenciálnemu ušetreniu výkonu, oproti predošlej verzii, ktorá obsahovala rozsiahlejšiu scénu. Ušetrený výkon môže byť využitý na ďalšie vyvíjané funkcionality histogramu a jeho zobrazení.

V prípade potreby rozsiahlejšej scény je možné použiť príkaz `scale(x,y,z)` v `NdmVrLaboratory`, ktorým sa daná scéna zväčší.

## 3.5 Intuitívnejšie zobrazenie dát histogramu

V tejto časti sme sa rozhodli zaoberať odstránením nedostatkov nepraktického umiestnenia indikátora pomoci, ako aj informačnej tabule, ktorá bola umiestnená za používateľom a vyžadovala zbytočný pohyb otočenia pri jej použití. Okrem toho boli informácie o zobrazení manuálu pri prvom a úvodnom pohľade skryté, pretože sa nachádzali za používateľom. Ďalším nedostatkom, ktorému sa chceme venovať, je nepraktické a nereálne umiestnenie histogramu.

### 3.5.1 Návrh zmien zobrazenia histogramu

Na dosiahnutie reálnejšieho zobrazenia histogramu budeme musieť zmeniť pozíciu úvodného pohľadu kamery a zrkadlovo otočiť celý histogram. S tým súvisí aj zmena pozície plátna, aby používateľ mohol pohodlne analyzovať dáta histogramu. Pre lepšiu analýzu histogramu sme taktiež posunieme hodnoty na osi  $x$  a  $y$ , aby boli bližšie k hranici skutočnej hodnoty binu pre daný bod. V pôvodnej nezmenenej verzii boli hodnoty umiestnené v strede, čo nezodpovedalo danému bodu. Zlepšiť čitateľnosť osi  $x$  chceme tým, že otočíme jej hodnoty o 90 stupňov, čo z pohľadu používateľa zjednodušuje čítanie. Taktiež sme zmenili polohu informačnej tabule tak, aby bola umiestnená tak, že ju používateľ môže jasne a dostatočne vidieť už pri úvodnom pohľade.

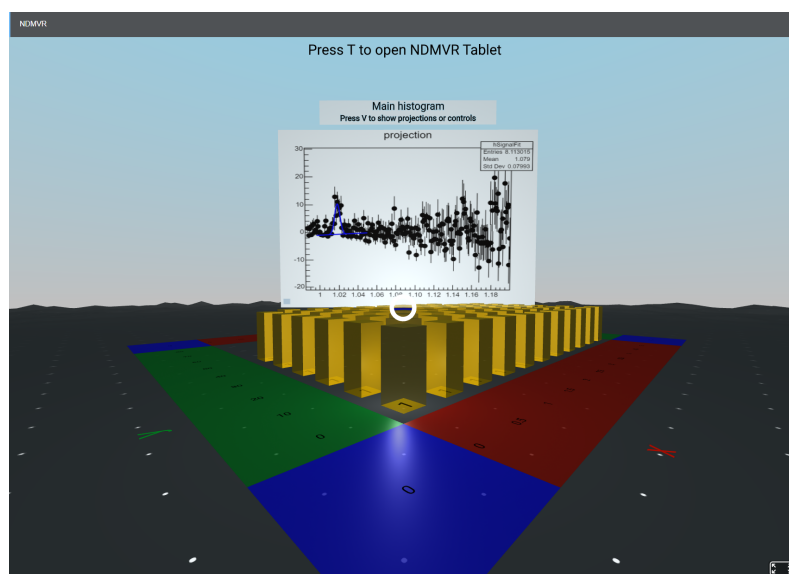
### 3.5.2 Realizácia zmien zobrazenia histogramu

Zmenu zobrazenia histogramu sme realizovali v komponente `ThProjection`, kde sme zmenili súradnice polohy histogramu a jeho osí zrkadlovo otočili. V tomto komponente sme tiež otočili entity osi  $x$  o 90 stupňov pre lepšiu čitateľnosť vzhľa-

dom na pohľad používateľa. Hodnoty osi x sme posunuli od stredu pomocou správneho ofsetu, aby sa ich umiestnenie čo najviac priblížilo skutočnej hodnote v danom bode histogramu. Táto zmena bola vykonaná pre každú entitu osi x pri generovaní histogramu v komponente **NdmVrHistogram**.

Zmenu úvodného pohľadu sme realizovali v komponente **NdmVrCamera**, kde sme zmenili súradnice entity a aj jej úvodnú rotáciu pre kameru, tablet a obe controllery a ďalšie prvky, ktoré používateľ ovláda. Zmenu plátna sme vykonali v **histogramGenerátore**, kde sme definované súradnice generovania plátna nahradili novými a zmenili sme jeho rotáciu pre dosiahnutie lepšieho uhlu pohľadu.

Informačná tabuľa spolu s informáciou o zobrazení manuálu bola umiestnená na vrch plátna, aby ju používateľ mohol ľahko vidieť hneď po spustení aplikácie. Túto zmenu sme realizovali v častiach **histogramGenerator** a **NdmVrhistogram**, pričom zmena polohy a zobrazenia by mala viac pripomínať informačnú tabuľku.



Obr. 3.11: Nové usporiadanie scény pri zapnutí NDMVR

## 3.6 Analýza histogramu z výšky

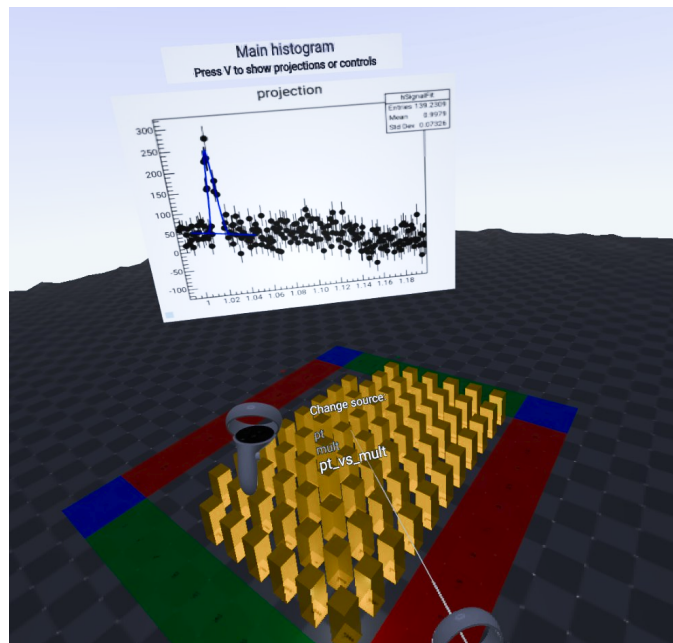
V časti 2.1.2 sme identifikovali nedostatok spočívajúci v obmedzení zmeny výšky pohľadu používateľa v zariadení Oculus. V tejto časti sme sa zaoberali návrhom a implementáciou riešenia tohto nedostatku.

### 3.6.1 Návrh možností lietania

Danú funkcionality nám zabezpečí nami použitý komponent **thumbstickOculusController**, ktorý obsahuje možnosť letu. Jeho ovládanie bude ponechané na

ľavej ruke, aby sme zachovali konzistenciu a ľahšie si zapamätali ovládanie, nakoľko ľavá ruka je zodpovedná za všetky formy pohybu. Pre praktickosť použitia danej funkcionality sme pridali možnosť vrátiť sa späť na zem. Taktiež chceme umožniť pohyb v rovine aj v zmennej výške, tak ako to bolo na zemi. Táto možnosť bude veľmi užitočná pri analýze histogramu z výšky, nakoľko režim letu môže byť menej komfortný a náročný na udržanie rovnakej výšky.

### 3.6.2 Podporované možnosti lietania



Obr. 3.12: Podporovaný fly mód v NDMVR

V komponente `thumbstickOculusController` sme upravili parameter pre režim letu tak, že po stlačení spúšte na oculus kontroléri je prepísaný na hodnotu *true*. Po jeho pustení je zmenený na hodnotu *false*, čo umožňuje pohyb v rovine aj v zmennej výške. Režim letu spracováva aktuálny smer pohľadu a vykonáva pohyb voči nemu, čo znamená, že používateľ letí tam, kam sa pozerá. Pristátie sme umožnili dvojitým kliknutím spúšte, kedy sa neustále mení hodnota y-ovej zložky vektoru, až kým nedosiahne hodnotu y definovanú pri štarte aplikácie, čo určuje výšku entity používateľa. Komponent sme museli opraviť, nakoľko rotácia osi y obsahovala istú odchýlku, ktorá spôsobovala posun pri pohybe dopredu smerom doľava. Chyba bola odstránená pridaním ofsetu, ktorý túto odchýlku vyrovnáva a zaisťuje rovnaký pohyb pri chôdzi aj lete.

## 4 Testovanie vykonaných zmien

---

### 4.1 Návrh UX testovania

Po úspešnom navrhnutí a implementácii zmien v kapitole 3 sme sa venovali hodnoteniu používateľského zážitku (UX). Toto hodnotenie sme realizovali prostredníctvom dvoch UX testovaní. Prvé testovanie sa zameralo na získanie hodnotenia nezmenenej verzie komponentu a jej UX. Druhé testovanie sa zameriavalo na verziu s navrhnutými a implementovanými zmenami. Pre posúdenie efektívnosti týchto zmien bolo nevyhnutné správne otestovať obe verzie a porovnať ich výsledky. Tieto testovania nám poskytnú obraz o tom, do akej miery naše implementované zlepšenia v komponente NDMVR prispeli k zlepšeniu UX.

#### 4.1.1 Návrh a výber metodík testovania

Pri návrhu a výbere metodík testovania sme sa opierali o analyzované práce uvedené v kapitole 1. Aspekty testovania boli zvolené na základe časti 1.3, ktorá poukazuje predovšetkým na tieto 4 aspekty merania UX: užitočnosť, prítomnosť, fyzickú ergonómiu a estetiku. Pre hodnotenie užitočnosti sme použili štandardnú metodiku SUS, ktorá je popísaná v kapitole 1.5.1 a zameriava sa na hodnotenie použiteľnosti a užitočnosti komponentu. Navyše, pre hodnotenie prítomnosti sme využili metodiku UES, ktorá hodnotí zapojenie používateľov. Táto metóda poskytuje presnejšie výsledky ako SUS a je popísaná v kapitole 1.7. Pre hodnotenie fyzickej záťaže sme použili metódu NASA TLX v kapitole 1.6.1, ktorá nám poskytne informácie o zaťažení používateľa. Ako sme sa dozvedeli, hodnotenie fyzickej záťaže hrá dôležitú rolu pri zabezpečení čo najlepšieho používateľského zážitku. Na hodnotenie estetiky sme vytvorili vlastný dotazník, ktorý sa zameriava na dizajn komponentu NDMVR.

Po výbere vhodných metodík hodnotenia sme sa zamerali na formu, v akej budeme metodiky používateľom predkladať. Zistili sme, že elektronická forma je rovnako účinná ako klasická papierová forma, čo sme porovnali v kapitole 1.6.2.



Rozhodli sme sa použiť elektronickú formu, ktorá obsahuje otázky a škálu hodnotenia z papierových foriem. Táto elektronická forma nám umožňuje získať odpovede priamo v prostredí VR s nasadeným headsetom, čo by malo viesť k lepším výsledkom. Pre tento účel sme zvolili nástroj Google Forms<sup>1</sup>, v ktorom budeme realizovať takmer každú metodiku hodnotenia. SUS dotazník sme navrhli na základe analýzy a preložili sme ho do slovenského jazyka. Otázky pre UES-SF sme získali z obr. 1.9 a tiež sme ich preložili do slovenského jazyka. Navyše, pre hodnotenie fyzickej záťaže sme použili oficiálnu aplikáciu NASA TLX pre iOS zariadenia<sup>2</sup>.

Nakoniec sme sa venovali návrhu otázok týkajúcich sa dizajnu. Rozhodli sme sa zamerať na estetickú stránku týchto častí komponentu NDMVR: prostredie, histogram, celkové zobrazenie histogramu a plátno, tablet, pomocný manuál, nápovedy, rozmiestnenie a pocity vyvolané daným dizajnom. Elektronické verzie dotazníkov sú uvedené v prílohe A.

### 4.1.2 Vhodní respondenti

Keďže sa jedná o pomerne špecifickú oblasť, rozhodli sme sa zahrnúť do testovania rôzne skupiny účastníkov, ktorí pracujú s rôznymi formami zobrazovania dát a ich analýzy. Zamierujeme sa na viaceré vekové kategórie, pre ktoré je NDMVR určený. Rozhodli sme sa zahrnúť účastníkov s rôznymi relevantnými charakteristikami.

Na začiatok je dôležité poznamenať, že sme do testovania zahrnuli tri typy respondentov. Prvá skupina sú začiatočníci, ktorí nemajú žiadnu alebo len minimálnu skúsenosť s VR realitou a headsetom. Druhá skupina tvoria pokročilí respondenti, ktorí majú skúsenosť s používaním VR reality a jej základným ovládaním, ale s komponentom NDMVR sa ešte nestretli. Posledná skupina sú experti, ktorí sa aktívne podieľali na vývoji NDMVR alebo budú s aplikáciou úzko spolupracovať v blízkej budúcnosti. Ich odborné znalosti sú veľmi cenné, pretože objavia nedostatky z pohľadu fyzikálnej analýzy a iných aspektov, ktoré ostatným účastníkom môžu uniknúť. Žiadny z týchto testovaných subjektov sa nepodieľal na zmenách smerujúcich k lepšiemu používateľskému zážitku, preto ich názor považujeme za nezaujatý.

Pri výbere všetkých typov respondentov sme si všímali tieto aspekty:

#### 1. Osoba pracuje s dátami

---

<sup>1</sup><https://www.google.com/forms/about/>

<sup>2</sup><https://apps.apple.com/us/app/nasa-tlx/id1168110608>



2. Analyzuje dáta
3. Používa technológie na dennej bázy (PC, smartfón)
4. Používa programy slúžiace pre evidenciu, zobrazenie, porovnávanie, prehľad dát
5. Úlohy, ktoré respondent bežne (v práci/škole) vykonáva vyžadujú zobrazenie obsahu, jeho výsledkov, stanovenia záverov

Zamerali sme sa na vekovú kategóriu od 18 do 50 rokov, nakoľko táto skupina predstavuje potenciálnych používateľov NDMVR. Vzhľadom na to, že prostredie a pokyny v komponente NDMVR sú v anglickom jazyku, vyžadovali sme základnú znalosť anglického jazyka. Odporúčaný počet účastníkov pre testovanie použiteľnosti je 5<sup>3</sup>. Vzhľadom na to, že testujeme dve verzie, vychádzali sme z analýzy časti 1.9, ktorá uvádza minimálny počet 4 pre testovanie dvoch verzií. Pre naše účely sme sa rozhodli pre optimálny počet 8 respondentov. V skupine respondentov budú zastúpení dvaja VR používatelia z každej kategórie (začiatočník, pokročilý), pri kategórii expert budeme mať 4 respondentov.

### 4.1.3 Návrh testovania dvoch verzií projektu

Skupina respondentov	Poradie testovaných verzií
Respondnet 1 (začiatočník)	A B
Respondnet 2 (začiatočník)	B A
Respondnet 3 (pokročilý)	A B
Respondnet 4 (pokročilý)	B A
Respondnet 5 (expert)	A B
Respondnet 6 (expert)	B A
Respondnet 7 (expert)	A B
Respondnet 8 (expert)	B A

Tabuľka 4.1: Vyvážené poradie testovaných verzií (naše)

Potrebuje otestovať dve verzie komponentu NDMVR. Prvá verzia bude zodpovedať analyzovanému aktuálnemu stavu NDMVR komponentu, zatiaľ čo druhá

<sup>3</sup><https://www.nngroup.com/articles/measuring-perceived-usability/>

verzia bude zahŕňať naše implementované zmeny, ktoré majú za cieľ zlepšiť používateľský zážitok (UX). Na tento účel použijeme metódu popísanú v časti analýzy 1.9. Konkrétne plánujeme uskutočniť test na 8 respondentoch, ktorí sú bližšie opísaní v sekcii 4.1.2. Budeme dodržiavať vyvážené poradie testovaných verzií. Navrhli sme vhodný postup, ktorý minimalizuje negatívny vplyv na výsledky testu a je zobrazený v tabuľke 4.1. Po každej verzii respondenti vyplnia nami zvolené hodnotiace metódy, ako je opísané v časti 4.1.1. Odpovede budú zaznamenané priamo v prostredí VR, aby respondenti nemuseli zdvíhať headset a tým nepokazili svoje skúsenosti a dojmy. Čo je dôležité, aby sme predišli nežiaducemu skresleniu výsledkov.

#### 4.1.4 Scenár testovania

V tejto časti sa venujeme návrhu scenára a krokov, ktoré má účastník vykonať, ako aj špecifikácií úlohy a cieľu testovania. (Anglické označenia boli zvolené pre lepšie pochopenie scenára používateľom, nakoľko je program v angličtine)

##### Scenár:

- S1.1. Otvorte VR móde v prehliadači.
- S1.2. Zobrazte ľubovoľný bin.
- S1.3. Odčítajte maximálnu y-ovú hodnotu zobrazeného binu na projekcii.
- S1.4. Prejdite si všetky zobrazenia tabletu na základe poskytnutého manuálu počas UX testovania
- S1.5. Vyberte si z histogramu ľubovoľný bin.
- S1.6. Odčítajte jeho súradnice zobrazené na tablete. Príklad (value:2, x:2, y:1, z:0).
- S1.7. Pomocou tabletu zmeňte výšku binov histogramu na ich minimálnu hodnotu.
- S1.8. Zmeňte výšku binov na zobrazenie väčšie ako je ich minimálna hodnota
- S1.9. Zmeňte zdroj zobrazenia binov
- S1.10. Ak je daná funkcionálna podporovaná, zmeňte tému binov, podľa ľubovoľného výberu, pokúste sa preskúmať čo najviac tém
- S1.11. Ak je daná funkcionálna podporovaná, zmeňte rozsah zobrazenia binov na hodnotu 4 a akúkoľvek ďalšiu ľubovoľnú hodnotu

- S1.12. Ak je daná funkcionálna podpora, pokúste sa zmeniť scénu zobrazenia a preskúmať čo najviac scén
- S1.13. Prejdite sa okolo histogramu, zastavte v mieste, kde bola vaša východzia poloha
- S1.14. Použite fly mód a analyzujte histogram z výšky zodpovedajúcej výške plátna.
- S1.15. Vráťte sa späť na zem
- S1.16. Vypnite fly mod a prejdite sa po zemi

**Špecifikácia:** Scenár sa snaží overiť prvý dojem a intuitívnosť komponentu NDMVR. Jeho cieľom je zistiť, ako rýchlo a efektívne sa rôzni používatelia dokážu zoznámiť s možnosťami, ktoré komponent ponúka, a či sú schopní s ním pracovať. V prvých krokoch sa zameriavame na vstup do systému a zoznámenie sa s histogramom, ktorý slúži na reprezentáciu údajov. Chceme overiť, či je táto operácia v NDMVR jednoduchá a jasná, a či používatelia dokážu správne analyzovať histogram, identifikovať plátno a premietané zobrazenie. Zároveň sa snažíme zistiť, či sú dáta v histogramu pochopiteľné a čitateľné, či sú správne umiestnené a získať spätnú väzbu ohľadom aspektov prostredia, ktoré ovplyvňujú zobrazenie a analýzu dát.

Pozornosť sa taktiež upriamuje na Tablet a jeho ovládanie. Chceme zistiť, či je Tablet prehľadný a či jeho ovládanie je jednoduché pre používateľov, keďže ide o jeden z hlavných ovládacích prvkov. Snažíme sa zistiť, či používatelia dokážu vykonávať rôzne operácie, ktoré Tablet ponúka, bez potreby manuálu alebo iných foriem pomoci.

V posledných krokoch je pozornosť sústredená na pohyb v prostredí. Budeme sledovať pohyb po zemi aj nad ňou. Cieľom je zistiť, či je pohyb intuitívny, príjemný a či sú možnosti pohybu jasné pre používateľov.

## 4.2 Realizácia UX testovania

Na začiatok testovania bolo respondentom predložené prehlásenie o súhlase so spracovaním osobných údajov a potvrdení účasti na testovaní. Dokument sa nachádza v prílohe A. Po jeho vyplnení, boli oboznámení s ovládaním aplikácie NDMVR. Niektorí respondenti boli taktiež požiadaní o súhlas s vyhotovením fotografie počas testovania.

Respondent následne dostal danú verziu, podľa plánu vyváženého poradiu uvedeného v tab. 4.1. Počas celej doby, bola jeho práca vo VR monitorovaná a zaznamenávaná prostredníctvom aplikácie Meta Quest<sup>4</sup>. Respondentovi bol postupne čítaný scenár uvedený v časti 4.1.4, ktorý mal vykonávať krok po kroku. Respondent nahlas komentoval svoje akcie a rozhodnutia. Po dokončení poslednej úlohu nasledovala sada dotazníkov, pričom prvým bol dotazník UES<sup>5</sup>, druhým SUS<sup>6</sup> a tretím nami vytvorený bonusový dotazník<sup>7</sup>. Na záver respondenti vyplnili dotazník Nasa TLX prostredníctvom aplikácie<sup>8</sup>.

Dotazníky boli vyplňané pre danú testovanú verziu. Po krátkej pauze, aby respondentovi neprišlo zle z nadmerného používania VR nasledovala ďalšia verzia NDMVR. Postup a scenár bol rovnaký. Po dokončení scenára respondent vyplnil rovnaké dotazníky pozostávajúce z rovnakých otázok, len určené pre druhú verziu. Počas celého testovania boli zapisované rôzne návrhy, reakcie, pripomienky za účelom zlepšenia NDMVR. Po ukončení testovania týchto dvoch verzii, nasledovala krátka diskusia, kde respondent mal možnosť vyjadriť akékoľvek pripomienky, nedostatky, prípadné nápady a možnosti zlepšenia NDMVR, ako aj vyjadriť svoj názor, ktorá verzia bola pre neho osobne viacej obohacujúca, lepšia pre použitie.

### 4.3 Vyhodnotenie výsledkov UX testovania

Pri porovnávaní výsledkov hodnotenia verzií A (bez akýchkoľvek zmien) a B (nami upravená verzia) je dôležité dôkladne zhodnotiť prípadné rozdiely medzi týmito dvoma verziami. Tieto rozdiely môžu poskytnúť cenné informácie o tom, aký vplyv mala naša úprava na používateľský zážitok. V rámci nášho vyhodnotenia testovania sa zameriame na hodnotenie použiteľnosti prostredníctvom metodiky SUS, posúdime celkovú angažovanosť, vnímanú použiteľnosť, estetickú príťažlivosť, faktor odmeny, sústredenú pozornosť. Zhodnotíme výsledky merania zaťaženia pri používaní oboch verzii. Zameriame sa na rozdiely v úrovni frustrácie, stresu a únavy, ktoré môžu byť spojené s danými verziami aplikácie. Analyzujeme diskusiu, spätnú väzbu a odpovede z bonusového dotazníka získaného od respondentov. Preskúmame ich názory, pripomienky a odporúčania s cieľom identifikovať oblasti, ktoré vyžadujú zlepšenie a možné vylepšenia NMDVR na

---

<sup>4</sup><https://apps.apple.com/us/app/meta-quest/id1366478176>

<sup>5</sup><https://forms.gle/4kpr1GVHtshenVPz7>

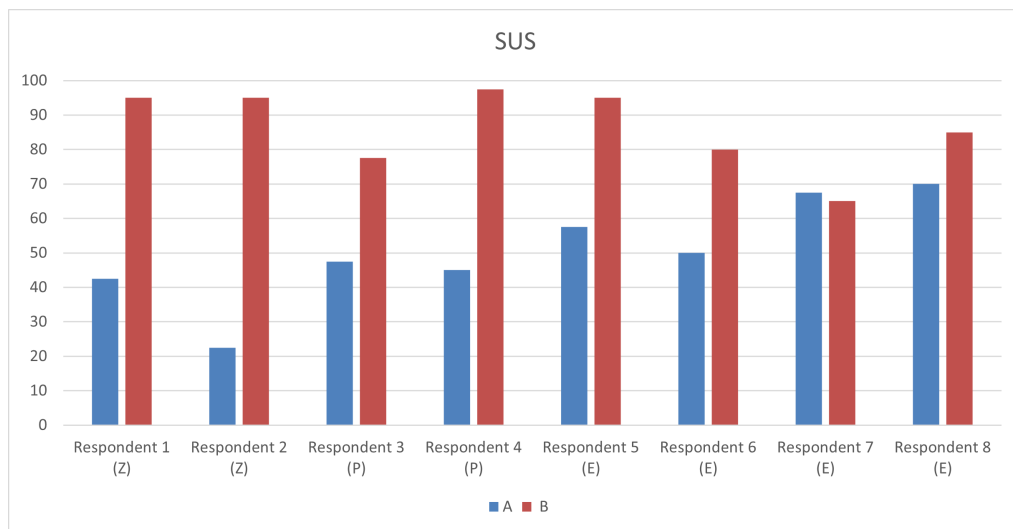
<sup>6</sup><https://forms.gle/mAghKqB7KsPQ4Fb86>

<sup>7</sup><https://forms.gle/dST9qNcNwJxJcZH39>

<sup>8</sup><https://apps.apple.com/us/app/nasa-tlx/id1168110608>

základe získaných výsledkov. Na základe týchto analýz a zhodnotení výsledkov poskytneme závery a odporúčania pre nami upravenú verziu NMDVR. Tieto odporúčania budú zamerané na konkrétne zlepšenia, ktoré môžu zvýšiť použiteľnosť, znižovať zaťaženie a zlepšiť celkový používateľský zážitok ešte rapídnejšie v porovnaní s verziou A.

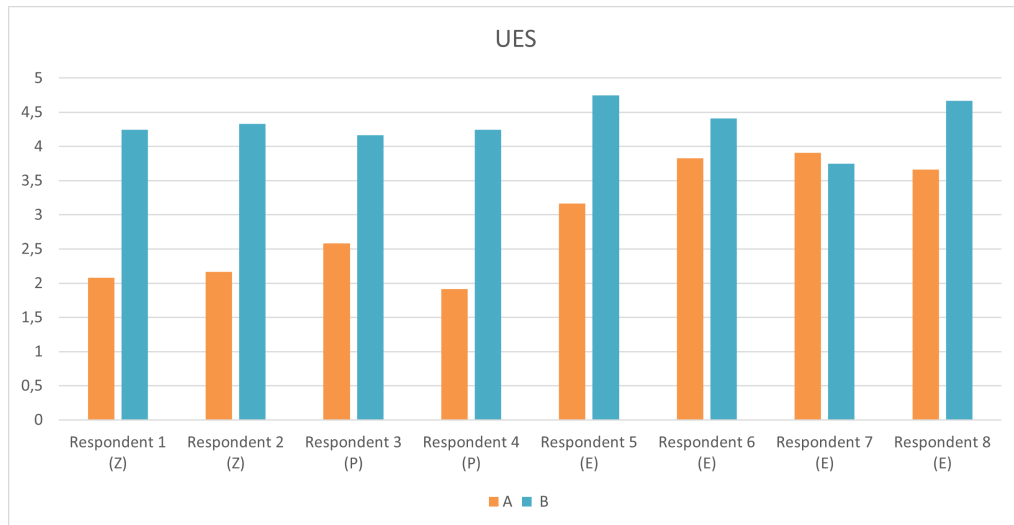
### 4.3.1 Vyhodnotenie použiteľnosti



Obr. 4.1: Výsledky hodnotenia použiteľnosti metodikou SUS verzií A a B

Výsledky hodnotenia použiteľnosti pomocou metodiky SUS sú zobrazené na obrázku 4.1, ktorý zobrazuje výsledky oboch verzií - verzie A a B. Respondenti zaradení do kategórií začiatočník a pokročilý hodnotili verziu B o niečo priaznivejšie než účastníci testovania v kategórii expert. Celková použiteľnosť systému, vypočítaná ako priemer hodnotení jednotlivých respondentov, pre verziu A bola **46,66**. Priemerná hodnota použiteľnosti systému uvedená v časti 1.5.1 bola stanovená na 68. Podľa časti 1.5.2, ktorá sa venuje hodnoteniu SUS, je verzia A zaradená do kategórie "slabá použiteľnosť systému". Verzia B dosiahla skóre **87,91** a patrí do kategórie "výborná použiteľnosť". Je však dôležité zdôrazniť, že vzhľadom na zloženie testovacej skupiny, ktorá zahŕňala nielen expertov, môže tento výsledok byť mierne skreslený. S väčším počtom respondentov, najmä v kategórii expert, by výsledky mohli klesnúť. Napriek tomu sme presvedčení, že verzia B spĺňa požiadavky na priemerný dobre použiteľný systém. Získané hodnotenie verzie B odzrkadľuje lepšiu použiteľnosť systému a teda aj vyšší používateľský zážitok v porovnaní s verziou A.

### 4.3.2 Vyhodnotenie angažovanosti

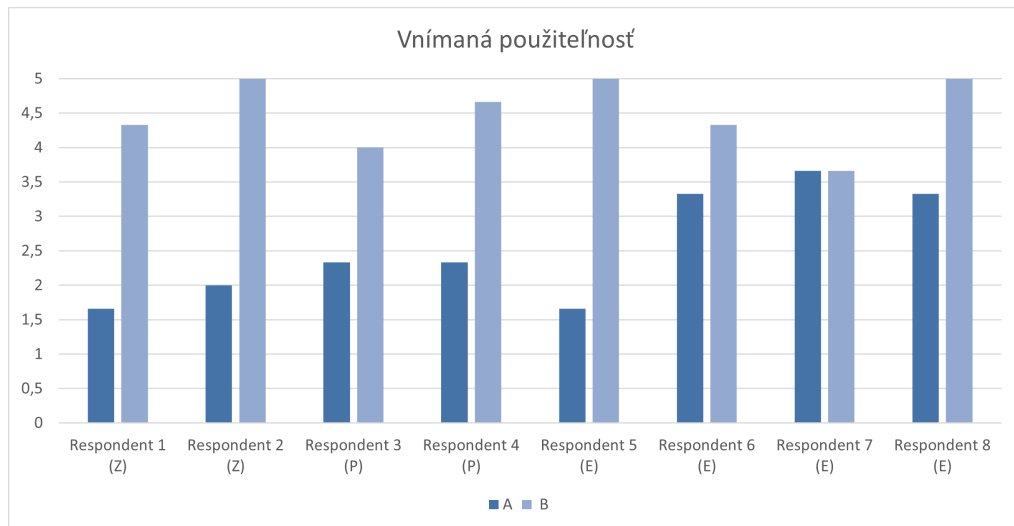


Obr. 4.2: Výsledky hodnotenia angažovanosti pomocou metodiky UES-SF verzií A a B

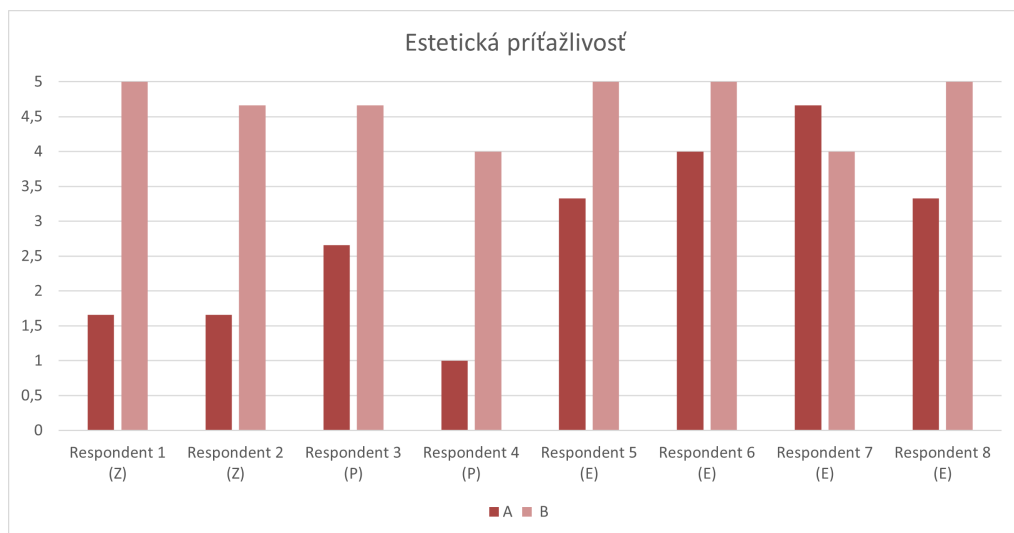
Hodnotenie angažovanosti pomocou metodiky UES-SF, ktorá je opísaná v časti 1.7, je zobrazené na obrázku 4.2. Obrázok zachytáva vývoj celkovej angažovanosti oboch verzií - verzií A a B. Celková angažovanosť pozostáva z oblastí, ako je vnímaná použiteľnosť, estetická príťažlivosť, faktor odmeny a sústredená pozornosť. Verzia A dosiahla skóre celkovej angažovanosti **2,67**. Vzhľadom na výsledky angažovanosti v rozmedzí od 1 do 5, kde 1 predstavuje minimálnu angažovanosť a 5 maximálnu angažovanosť, môžeme kategorizovať angažovanosť verzie A ako slabú. Verzia B dosiahla skóre **4,29**, čo naznačuje veľmi dobrú angažovanosť, keďže sa blíži k maximálnemu skóre 5. Je však potrebné mať na pamäti, že výsledok môže byť mierne skreslený nedostatočným počtom respondentov, najmä v kategórii expert. Verzia B prejavuje vyššiu angažovanosť než verzia A.

Skúmanie jednotlivých oblastí angažovanosti nám poskytuje ešte detailnejšie vyhodnotenie verzií A a B. Prvou skúmanou oblasťou bola **vnímaná použiteľnosť**, ktorá zahŕňa interakciu, mieru kontroly a mieru vynaloženého úsilia v súvislosti s celkovou použiteľnosťou. Na obrázku 4.3 sú zobrazené výsledky vnímanej použiteľnosti pre obe verzie. Vnímaná použiteľnosť verzie A dosiahla hodnotu **2,38**, pričom kategorizácia je vykonávaná v rozmedzí 1-5, čo znamená, že ju môžeme zaradiť medzi slabú použiteľnosť. Verzia B dosiahla skóre **4,49**, čo by sme mohli kategorizovať ako veľmi dobré. Verzia B prejavovala lepšiu vnímanú použiteľnosť než verzia A.

Druhou skúmanou oblasťou bola **estetická príťažlivosť**, ktorá zahŕňa vizuálnu príťažlivosť interiéru a rozhrania. Výsledky merania sú zobrazené na ob-



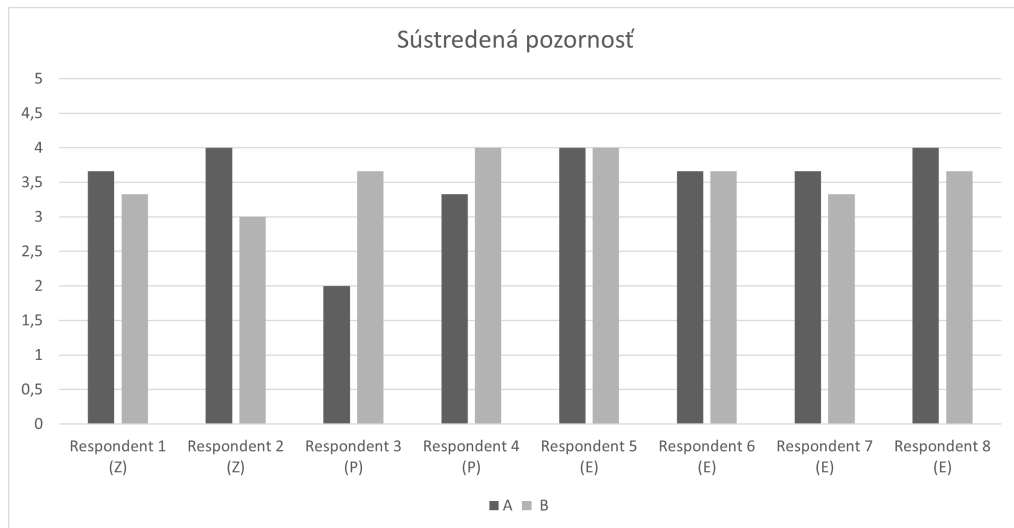
Obr. 4.3: Výsledky vnímanej použiteľnosti pomocou metodiky UES-SF



Obr. 4.4: Výsledky estetickej príťažlivosti pomocou metodiky UES-SF

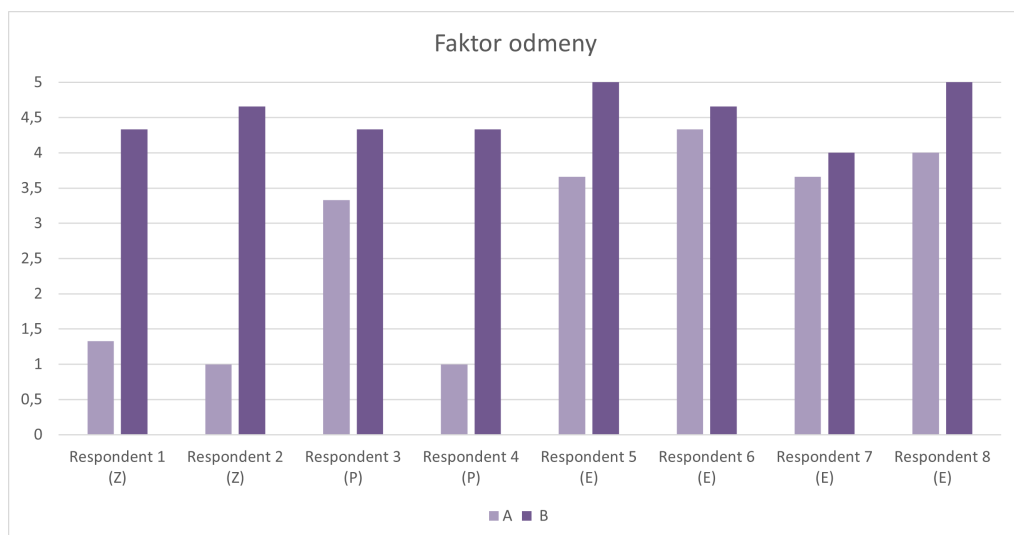
rázku 4.4. Celková estetická príťažlivosť verzie A dosiahla hodnotu **2,44**, čo by sme mohli kategorizovať ako slabú. Verzia B získala hodnotenie **4,63**, čo sa približuje k vynikajúcej estetickej príťažlivosti reprezentovanej číslom 5. Estetická príťažlivosť verzie B môže byť považovaná za veľmi dobrú. Verzia B priniesla lepšiu estetickú príťažlivosť v porovnaní s verziou A.

Tretou skúmanou oblasťou bola **sústredená pozornosť**, ktorá zahŕňa pocity pohltienia do interakcie a pocit straty času pri používaní aplikácie. Výsledky sú zobrazené na obrázku 4.5. Prekvapením bolo, že respondenti často vykazovali nižšiu sústredenú pozornosť pri verzii B. Tento jav môže byť spôsobený viacerými faktormi, ako sú dodatočné funkcionality, ktoré verzia B ponúka. V priebehu testovania bola tiež zaznamenaná komunikácia s osobou vykonávajúcou dohľad,



Obr. 4.5: Výsledky sústredenej pozornosti pomocou metodiky UES-SF

ktorá uistila respondentov, že vykonávajú úlohu správne, a odpovedala na ďalšie otázky a nejasnosti. Okrem toho verzia A mala obmedzené funkcionality a respondenti v nej trávili menej času ako vo verzii B, čo mohlo mierne ovplyvniť ich sústredenú pozornosť. Rovnako dôležité bolo aj poradie, v akom boli verzie testované, čo mohlo ovplyvniť sústredenú pozornosť pri druhej testovanej verzii. Celkovo bola sústredená pozornosť verzie A hodnotená na **3,44**, čo by sme mohli kategorizovať ako priemer. Verzia B získala hodnotenie **3,55**, čo naznačuje mierny nárast sústredenej pozornosti, ale stále by sme ju mohli považovať za priemernú.



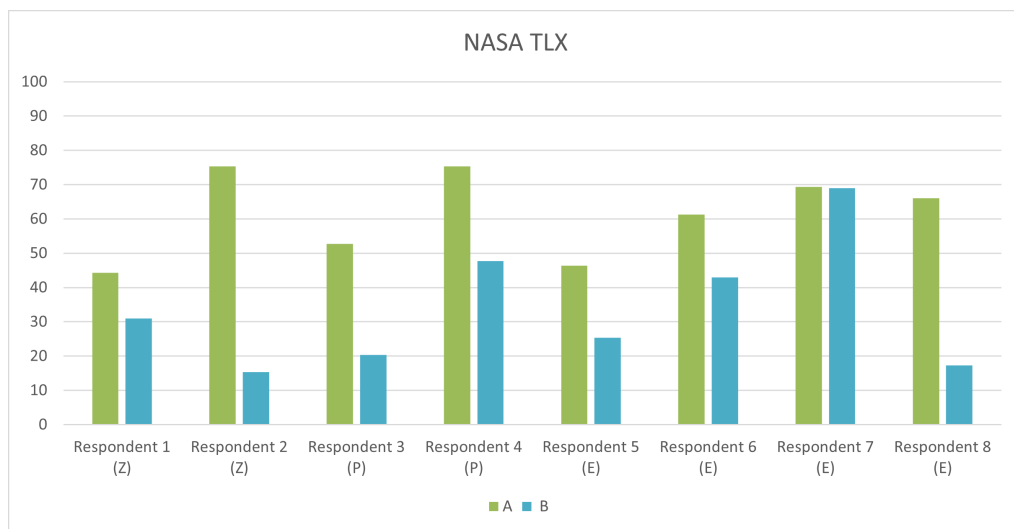
Obr. 4.6: Výsledky faktoru odmeny pomocou metodiky UES-SF

Štvrtou skúmanou oblasťou bola oblasť **faktora odmeny**, ktorá zahŕňa pocity príjemných a uspokojivých zážitkov z interakcie s aplikáciou, ako aj skúmanie ochoty používateľov odporučiť aplikáciu ostatným a angažovať sa s ňou aj v bu-



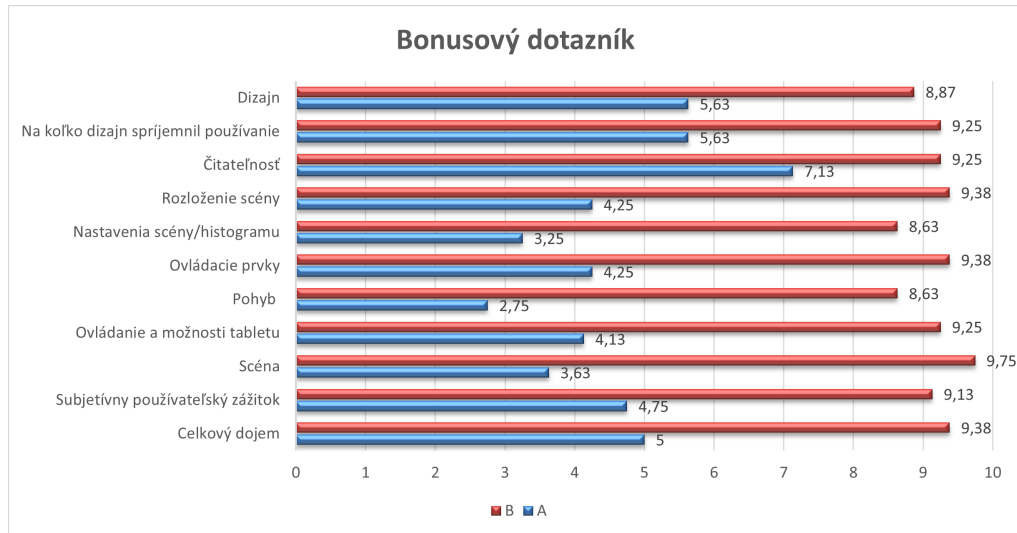
dúčnosti. Merania a ich výsledky sú zobrazené na obrázku 4.6. Celkový faktor odmeny pre verziu A bol **2,14**, čo je kategorizované ako nízke hodnotenie. Verzia B priniesla faktor odmeny s hodnotou **4,49**, čo by sme mohli definovať ako veľmi dobré hodnotenie. Respondenti považovali verziu B za príjemnejšiu a uspokojivejšiu z pohľadu interakcie s aplikáciou. Verzia B by mohla tiež zaujať respondentov natoľko, aby ju odporučili svojim známym a kolegom, čo by veľmi prospelo marketingu a prilákalo viac používateľov k aplikácii NDMVR. Vyššie skóre verzie B nasvedčuje tomu, že respondenti by sa radi s aplikáciou angažovali znova, čo je veľmi dôležitý faktor z pohľadu udržateľnosti používateľov systému.

### 4.3.3 Vyhodnotenie zaťaženia



Obr. 4.7: Výsledky faktoru odmeny pomocou metodiky UES-SF

Hodnotenie zaťaženia bolo uskutočnené pomocou metodiky NASA TLX, ktorá je opísaná spolu s hodnotením výsledkov zaťaženia v časti 1.6. Výsledky sú zobrazené na obrázku 4.7, ktorý zachytáva vývoj celkovej záťaže systému oboch verzií - verzie A a B. Celková záťaž verzie A bola stanovená na **61,52**, čo predstavuje vysokú úroveň zaťaženia. Verzia B dosiahla hodnotenie s nižšou záťažou, s hodnotou **31,94**, čo je kategorizované ako "pomerné vysoké". Je však potrebné zdôrazniť, že pre malú vzorku respondentov môžu výsledky byť mierne skreslené. Na základe vyjadrení respondentov však možno konštatovať, že verzia B prináša nižšiu záťaž ako verzia A, čo sa potvrdzuje aj výsledkami testovania prostredníctvom NASA TLX.



Obr. 4.8: Výsledky bonusového dotazníka

#### 4.3.4 Vyhodnotenie spätnej väzby respondentov

Diskusia odhalila množstvo pozitívnych aj negatívnych aspektov NMDVR, a to pre obe verzie. Niektorí respondenti sa sťažovali na komplikované ovládanie, či už kvôli potrebe zapamätania kombinácií alebo veľkosti rúk, čo im robilo problémy pri držaní všetkých potrebných tlačidiel. Tento problém identifikovali 3 respondenti, ktorí navrhovali riešenie v podobe dotykového tabletu alebo možnosti výberu prostredníctvom kurzora pravej ruky, ak dotykový tablet nie je podporovaný pri používaní NDMVR.

Respondent vo veku nad 50 rokov, zaradený do kategórie expert, jediný vyjadril, že verzia A bola pre neho jednoduchšia, pretože neobsahovala toľko funkcionalít. Jeho názor zohľadnil pri vyplňovaní dotazníkov a brali sme ho do úvahy.

Dvaja respondenti uviedli, že ovládanie tabletu bolo pre nich komplikované v oboch verziách, najmä pri použití prvej testovanej verzie. U pokročilých používateľov však bol systém a jeho funkcie veľmi rýchle a pozitívne prijaté. Diskusia tiež ukázala, že respondenti s predchádzajúcimi skúsenosťami s digitálnymi hrami hodnotili ovládanie jednoduchšie a intuitívnejšie ako respondenti bez takýchto skúseností. Zaujímavosťou je, že pre začiatočníkov, ktorí neboli zvyknutí na ovládanie pomocou joysticku, bolo ovládanie pohybu pomocou thumbsticku trochu problémové a neintuitívne. Avšak po získaní krátkej skúsenosti sa im pohyb s thumbstickom stal bližším a prirodzenejším.

Veľkým pozitívom bolo zistenie, že respondentom, ktorí mali obavy z pohybu vo virtuálnej realite, sa rýchlo naučili ovládať relatívny pohyb verzie B. Naopak, pohyb verzie A takmer u každého spôsoboval náznaky kinetózy a frustrácie pri

jeho používaní. Jeden respondent však po diskusii vyjadril veľmi negatívne vnímanie aj pohybu verzie B, keďže jeho telo bolo nepohyblivé, ale okolie sa menilo. Navrhol prídanie funkcionality pohybu v priestore mapovaného do VR, čo by podľa neho výrazne zlepšilo používateľský zážitok.

5 respondentov výslovne chválilo možnosť meniacej sa rýchlosti zmeny výšky binov, pri ktorej bolo nastavenie správnej veľkosti jednoduché a prispôsobiteľné aj pri nízkych výškach binov. 2 z 4 expertných respondentov uviedli, že systém nie je podľa ich názoru vhodný pre expertnú analýzu a skôr ho vnímajú ako edukačný nástroj pre študentov. Jeden respondent poukázal na nedostatok, a to zmenu rozsahu binov, kde bol pridaný offset 1, čo spôsobovalo zobrazenie jedného dodatočného binu pri každom rozsahu. Taktiež bolo navrhnuté zlepšenie zobrazenia pri zdroji (mult), kde os grafu by mala byť označená zelenou farbou namiesto červenej.

Expertní respondenti veľmi pozitívne hodnotili zmenu rozloženia scény, pretože analýza histogramu bola oveľa intuitívnejšia a zodpovedala skutočnému zobrazeniu histogramu. Jeden respondent vyjadril negatívum týkajúce sa slovného opisu funkcionality zmeny výšky binov, kde sa použilo znamienko "+" medzi jednotlivými ovládacími prvkami, čo ho zmiatlo, keďže hľadal znamienko "+" na ovládači.

Traja respondenti uviedli začínajúce bolesti hlavy z používania VR a preto usúdili, že aplikácia nie je vhodná na dlhé používanie. Spätná väzba ukázala, že vizuálny aspekt bol veľmi pozitívny a zlepšil dojem o NMDVR v prípade verzie B. Zmena scén vyvolala úsmevy na tvárach väčšiny respondentov a bol to najviac dobrovoľne skúmaný prvok NDMVR. Respondenti všetkých kategórií vnímali túto funkciu ako obohacujúcu a zlepšujúcu dojem a používateľský zážitok.

Jednoznačne sa ukázalo, že respondenti vnímali lepší používateľský zážitok v prípade verzie B.

## 5 Záver

---

Táto bakalárska práca sa venuje problematike nedostatočného používateľského zážitku v experimentálnom nástroji NDMVR a jeho optimalizácii s cieľom zlepšiť interakciu a poskytnúť používateľom bohatší a pôsobivejší zážitok z jeho používania. Jej neoddeliteľnou súčasťou je použitie adekvátnych a dôveryhodných metód pre hodnotenie používateľského zážitku NDMVR, nakoľko táto téma nie je zatiaľ dostatočne prebádaná, práca a námet testovania má skôr experimentálny charakter.

Technológie, ktoré boli použité, boli jasne definované, nakoľko komponent NDMVR je súčasťou vývoja už niekoľko rokov a prejdienie na inú technológiu, obsahujúcu viacero možností a prostriedkov ovládania VR, neprichádzalo do úvahy. Použitie technológie A-frame nedovoľovalo vykonanie všetkých nami navrhnutých zmien, ako napríklad dotykové ovládanie tabletu počas používania oculus kontroléra, taktiež ovládanie prvkov tabletu prostredníctvom gest pri primárnom ovládaní Oculus kontrolérom. Napriek tomu sme si definovali jasný cieľ, ktorý sme chceli v práci dosiahnuť.

V práci bola najprv vykonaná analýza existujúcich technológií, metód merania UX a identifikácie nedostatkov, kde boli získané dôležité poznatky o problémoch, s ktorými sa virtuálne prostredia s nedostatočným UX často stretávajú.

Následným krokom bolo odhalenie hlavných nedostatkov nástroja NDMVR a navrhnutie riešení, ktoré by tieto nedostatky odstránili. Nie všetky návrhy boli z dôvodu krátkosti času alebo obmedzení používaných technológií realizovateľné a implementovateľné do verzie komponentu NDMVR. Väčšina riešení bola implementovaná do komponentu NDMVR, ktorá sa aktuálne vyznačuje lepšou ovládateľnosťou, intuitívnosťou, prispôbitelnosťou a pohodlím pri používaní.

Na základe testovania používateľského zážitku pred a po implementácii zmien bolo potvrdené výrazné zlepšenie UX v upravenej verzii NDMVR. Výsledky testovania jasne potvrdili úspech navrhnutých zmien a ich pozitívny vplyv na používateľov. Upravená verzia NDMVR priniesla používateľom nové prvky a možnosti používania nástroja, ktoré sa lepšie prispôbujú ich potrebám a zvyšujú celkovú

spokojnosť a efektivitu.

Okrem vylepšenia samotného UX komponentu NDMVR bola práca zameraná aj na významný aspekt používateľského zážitku vo virtuálnych prostrediach a jeho hodnotenie. Výsledky tejto bakalárskej práce prinášajú novú obohatenú verziu NDMVR, nové poznatky a navrhované metódy overenia UX v rozšírených realitách, čo poskytuje užitočný prínos pre vývoj softvérového komponentu NDMVR a potenciálne aj pre ďalšie oblasti.

Na záver bol ukázaný spôsob testovania UX v systéme VR s pomerne adekvátnymi a analyzovateľnými výsledkami. Týmto spôsobom bolo zabezpečené, že navrhované riešenia prekonávajú analyzované nedostatky a prispievajú k vytvoreniu lepšieho používateľského zážitku vo virtuálnych prostrediach. Výsledky tejto analýzy a overenia zmien budú dôležité pre nasledujúci vývoj a posun komponentu NDMVR smerom vpred.

Na základe vyššie uvedených zistení a výsledkov odporúčame:

- Pokračovanie vo vývoji a optimalizácii komponentu NDMVR na základe navrhnutých zmien a implementovaných vylepšení.
- Ďalšie prebádanie a rozvíjanie metódy merania a hodnotenia používateľského zážitku vo virtuálnych prostrediach, s dôrazom na experimentálne testovanie a zhromažďovanie spätných väzieb od používateľov.
- Vytváranie personalizovateľných možností v rámci NDMVR a podobných nástrojov, ktoré umožnia používateľom prispôbiť si prostredie podľa svojich preferencií a potrieb.
- Kontinuálne zhromažďovanie spätnej väzby od používateľov /potenciálnych používateľov a ich zapojenie do procesu vývoja a testovania, aby sa zabezpečila neustála iterácia a zlepšovanie používateľského zážitku.
- Využívanie nových technológií a vývojových platforiem na zlepšenie a rozšírenie možností NDMVR a podobných nástrojov.

# Literatúra

---

1. LAW, Effie Lai-Chong; VAN SCHAIK, Paul; ROTO, Virpi. Attitudes towards user experience (UX) measurement. *International Journal of Human-Computer Studies*. 2014, roč. 72, č. 6, s. 526–541. ISSN 1071-5819. Dostupné z DOI: <https://doi.org/10.1016/j.ijhcs.2013.09.006>. Interplay between User Experience Evaluation and System Development.
2. LAW, Effie Lai-Chong. The Measurability and Predictability of User Experience. In: *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. Pisa, Italy: Association for Computing Machinery, 2011, s. 1–10. EICS '11. ISBN 9781450306706. Dostupné z DOI: [10.1145/1996461.1996485](https://doi.org/10.1145/1996461.1996485).
3. KUČEROVÁ, Markéta. *Measuring UX - a Brief Introduction to Attitudinal Metrics*. [B.r.]. Dostupné tiež z: [https://miro.medium.com/max/1400/1\\*AIDkfxA\\_0U6Tf\\_yNh-ZSxQ.png](https://miro.medium.com/max/1400/1*AIDkfxA_0U6Tf_yNh-ZSxQ.png).
4. GOWNDER, J. P.; VOCE, Christopher; MAI, Michelle; LYNCH, Diane. *Breakout Vendors: Virtual and Augment Reality*. nc., 60 Acorn Park Drive, Cambridge, MA 02140 USA, 2016-05. Tech. spr., 1. Forrester Research. Dostupné tiež z: <https://www.forrester.com/report/Breakout-Vendors-Virtual-And-Augmented-Reality/RES134187>. Získané 13. Oktober 2022.
5. CHUAH, Stephanie Hui-Wen. *Why and Who Will Adopt Extended Reality Technology? Literature Review, Synthesis, and Future Research Agenda*. SSRN Working Paper Series, 2018. Dostupné z DOI: [doi:10.2139/ssrn.3300469](https://doi.org/10.2139/ssrn.3300469).
6. Na LF UPJŠ otvorili simulátorové centrum na výučbu za 2,8 milióna eur. *Tlačová agentúra Slovenskej republiky*. 2021. Dostupné tiež z: <https://www.tasr.sk/tasr-clanok/TASR:2021090900000213>.
7. WU, Hsin-Kai; LEE, Silvia Wen-Yu; CHANG, Hsin-Yi; LIANG, Jyh-Chong. *Current status, opportunities and challenges of augmented reality in education*. Zv. 62. Radarweg 29, 1043 NX Amsterdam, The Netherlands, 2013. Dostupné z DOI: [10.1016/j.compedu.2012.10.024](https://doi.org/10.1016/j.compedu.2012.10.024).

8. MARR, Bernard. The Most Amazing Real-World Examples Of Mixed Reality. 2019. Dostupné tiež z: <https://bernardmarr.com/the-most-amazing-real-world-examples-of-mixed-reality/>.
9. WIENRICH, Carolin; DÖLLINGER, Nina; KOCK, Simon; SCHINDLER, Kristina; TRAUPE, Ole. Assessing User Experience in Virtual Reality – A Comparison of Different Measurements. In: MARCUS, Aaron; WANG, Wentao (ed.). *Design, User Experience, and Usability: Theory and Practice*. Cham: Springer International Publishing, 2018, s. 573–589. ISBN 978-3-319-91797-9.
10. LAW, E.; VERMEEREN, A.; HASSENZAHN, M.; BLYTHE, (EDS.), M. Towards a UX manifesto. *COST294-MAUSE Affiliated Workshop*. 2007, roč. 2, s. 205–206.
11. STANNEY, Kay M; KENNEDY, Robert S; DREXLER, Julie M. Cybersickness is not simulator sickness. In: *Proceedings of the Human Factors and Ergonomics Society annual meeting*. SAGE Publications Sage CA: Los Angeles, CA, 1997, zv. 41, s. 1138–1142. Č. 2.
12. BAÑOS, R.M.; BOTELLA, C.; ALCANIZ, M.; LIAÑO, V.; GUERRERO, B.; REY, B. Immersion and Emotion: Their Impact on the Sense of Presence. *CyberPsychology & Behavior*. 2004, roč. 7, č. 6, s. 734–741. Dostupné z DOI: 10.1089/cpb.2004.7.734. PMID: 15687809.
13. RIVA, Giuseppe; MANTOVANI, Fabrizia; CAPIDEVILLE, Claret Samantha; PREZIOSA, Alessandra; MORGANTI, Francesca; VILLANI, Daniela; GAGGIOLI, Andrea; BOTELLA, Cristina; ALCANIZ, Mariano. Affective interactions using virtual reality: the link between presence and emotions. *Cyberpsychology & behavior*. 2007, roč. 10, č. 1, s. 45–56.
14. STANNEY, Kay M; MOURANT, Ronald R; KENNEDY, Robert S. Human factors issues in virtual environments: A review of the literature. *Presence*. 1998, roč. 7, č. 4, s. 327–351.
15. WIENRICH, C; NOLLER, F; THÜRING, M. Design principles for VR interaction models: an empirical pilot study. In: *Virtuelle und erweiterte Realitäten. 14. Workshop der GI-Fachgruppe VR/AR*. 2017, s. 162–171.
16. MINGE, Michael; RIEDEL, Laura. meCUE-Ein modularer Fragebogen zur Erfassung des Nutzungserlebens. In: *Mensch & Computer*. 2013, s. 89–98.

17. *CUE model*. [B.r.]. Dostupné tiež z: [http://mecue.de/\\_\\_\\_\\_impro/1/Homepage%20Content/english/CUE-Model.jpg?etag=%2213968-535743aa%22&sourceContentType=image%2Fjpeg&ignoreAspectRatio&resize=734%2B325&extract=0%2B5%2B734%2B320&quality=85](http://mecue.de/____impro/1/Homepage%20Content/english/CUE-Model.jpg?etag=%2213968-535743aa%22&sourceContentType=image%2Fjpeg&ignoreAspectRatio&resize=734%2B325&extract=0%2B5%2B734%2B320&quality=85).
18. SCHUBERT, Thomas; FRIEDMANN, Frank; REGENBRECHT, Holger. The experience of presence: Factor analytic insights. *Presence: Teleoperators & Virtual Environments*. 2001, roč. 10, č. 3, s. 266–281.
19. IJSSELSTEIJN, WA; KORT, YAW de; POELS, K. *Game Experience Questionnaire. FUGA The fun of gaming: Measuring the human experience of media enjoyment GAME*. Technische Universiteit Eindhoven, 2013.
20. KESHAVARZ, Behrang; HECHT, Heiko. Validating an efficient method to quantify motion sickness. *Human factors*. 2011, roč. 53, č. 4, s. 415–426.
21. KROHNE, Heinz W; EGLOFF, Boris; KOHLMANN, Carl-Walter; TAUSCH, Anja et al. Untersuchungen mit einer deutschen version der "positive and negative affect schedule" (PANAS). *Diagnostica-Gottingen*. 1996, roč. 42, s. 139–156.
22. *CUE model*. [B.r.]. Dostupné tiež z: [http://mecue.de/\\_\\_\\_\\_impro/1/onewebmedia/Startbild.jpg?etag=%22fba0-5357d603%22&sourceContentType=image%2Fjpeg&ignoreAspectRatio&resize=1001%2B277&extract=0%2B0%2B1001%2B277&quality=85](http://mecue.de/____impro/1/onewebmedia/Startbild.jpg?etag=%22fba0-5357d603%22&sourceContentType=image%2Fjpeg&ignoreAspectRatio&resize=1001%2B277&extract=0%2B0%2B1001%2B277&quality=85).
23. KULIGA, S.F.; THRASH, T.; DALTON, R.C.; HÖLSCHER, C. Virtual reality as an empirical research tool — Exploring user experience in a real building and a corresponding virtual model. *Computers, Environment and Urban Systems*. 2015, roč. 54, s. 363–375. ISSN 0198-9715. Dostupné z DOI: <https://doi.org/10.1016/j.compenvurbsys.2015.09.006>.
24. KORT, Yvonne AW de; IJSSELSTEIJN, Wijnand A; KOOIJMAN, Jolien; SCHURMANS, Yvon. Virtual laboratories: Comparability of real and virtual environments for environmental psychology. *Presence: Teleoperators & Virtual Environments*. 2003, roč. 12, č. 4, s. 360–373.
25. WESTERDAHL, Börje; SUNESON, Kaj; WERNEMYR, Claes; ROUPÉ, Mattias; JOHANSSON, Mikael; ALLWOOD, Carl Martin. Users' evaluation of a virtual reality architectural model compared with the experience of the completed building. *Automation in construction*. 2006, roč. 15, č. 2, s. 150–165.



26. FRANZ, Gerald; WIENER, Jan M. From space syntax to space semantics: a behaviorally and perceptually oriented methodology for the efficient description of the geometry and topology of environments. *Environment and planning b: planning and design*. 2008, roč. 35, č. 4, s. 574–592.
27. LAUBHEIMER, Page. Beyond the NPS: Measuring Perceived Usability with the SUS, NASA-TLX, and the Single Ease Question After Tasks and Usability Tests. 2018. Dostupné tiež z: <https://www.nngroup.com/articles/measuring-perceived-usability/>.
28. BROOKE, John. SUS: a retrospective. *Journal of Usability Studies*. 2013, roč. 8, s. 29–40.
29. BANGOR, Aaron; KORTUM, Philip; MILLER, James. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Studies*. 2009, roč. 4, č. 3, s. 114–123.
30. *What's the System Usability Scale(SUS)*. [B.r.]. Dostupné tiež z: <https://blog.hubspot.com/hs-fs/hubfs/System%20Usability%20Scale%20Scoring.png?width=598&name=System%20Usability%20Scale%20Scoring.png>.
31. COLLIGAN, Lacey; POTTS, Henry; FINN, Chelsea; SINKIN, Robert. Cognitive workload changes for Nurses transitioning from a legacy system with paper documentation to a commercial Electronic Health Record. *International Journal of Medical Informatics*. 2015, roč. 84. Dostupné z DOI: 10.1016/j.ijmedinf.2015.03.003.
32. HART, S. G. NASA Task Load Index (TLX). Volume 1.0; Paper and Pencil Package. In: 1986.
33. HART, Sandra G.; STAVELAND, Lowell E. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In: HANCOCK, Peter A.; MESHKATI, Najmedin (ed.). *Human Mental Workload*. North-Holland, 1988, zv. 52, s. 139–183. Advances in Psychology. ISSN 0166-4115. Dostupné z DOI: [https://doi.org/10.1016/S0166-4115\(08\)62386-9](https://doi.org/10.1016/S0166-4115(08)62386-9).
34. ROLDÁN GÓMEZ, Juan; PEÑA-TAPIA, Elena; MARTÍN-BARRIO, Andrés; OLIVARES-MENDEZ, Miguel; CERRO, Jaime; BARRIENTOS, Antonio. Multi-Robot Interfaces and Operator Situational Awareness: Study of the Impact of Immersion and Prediction. *Sensors*. 2017, roč. 17, s. 1720. Dostupné z DOI: 10.3390/s17081720.

35. PRABASWARI, Atyanti; BASUMERDA, Chancard; UTOMO, Bagus. The Mental Workload Analysis of Staff in Study Program of Private Educational Organization. *IOP Conference Series: Materials Science and Engineering*. 2019, roč. 528, s. 012018. Dostupné z DOI: 10.1088/1757-899X/528/1/012018.
36. NOYES, Jan; BRUNEAU, Daniel. A self-analysis of the NASA-TLX workload measure. *Ergonomics*. 2007, roč. 50, s. 514–519. Dostupné z DOI: 10.1080/00140130701235232.
37. MACH, Sebastian; GRÜNDLING, Jan; SCHMALFUSS, Franziska; KREMS, Josef. How to Assess Mental Workload Quick and Easy at Work: A Method Comparison. In: 2018. ISBN 978-3-319-96067-8. Dostupné z DOI: 10.1007/978-3-319-96068-5\_106.
38. [B.r.]. Dostupné tiež z: <https://apps.apple.com/us/app/nasa-tlx/id1168110608>.
39. O'BRIEN, Heather L.; CAIRNS, Paul; HALL, Mark. A practical approach to measuring user engagement with the refined user engagement scale (UES) and new UES short form. *International Journal of Human-Computer Studies*. 2018, roč. 112, s. 28–39. ISSN 1071-5819. Dostupné z DOI: <https://doi.org/10.1016/j.ijhcs.2018.01.004>.
40. RUBIN, J.; CHISNELL, D.; SPOOL, J. *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests, 2nd Edition*. Wiley, 2008. ISBN 978-0-470-18548-3.
41. *A-frame.io-documentation-site*. [B.r.]. Dostupné tiež z: <https://aframe.io/docs/1.3.0/introduction/>. dostupné: 21.1.2023.
42. *A-frame.io-documentation-browsers*. [B.r.]. Dostupné tiež z: <https://aframe.io/docs/1.3.0/introduction/vr-headsets-and-webxr-browsers.html#what-browsers-support-vr>. dostupné: 21.1.2023.
43. PATHAK, Rishabh; SIMISCUKA, Anderson Augusto; MUNTEAN, Gabriel-Miro. An Adaptive Resolution Scheme for Performance Enhancement of a Web-based Multi-User VR Application. In: *2021 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. 2021, s. 1–6. Dostupné z DOI: 10.1109/BMSB53066.2021.9547069.
44. *WebXR-documentation-site*. [B.r.]. Dostupné tiež z: <https://immersiveweb.dev/>. dostupné: 21.1.2023.
45. *WebGL-documentation-site*. [B.r.]. Dostupné tiež z: <https://get.webgl.org/>. dostupné: 21.1.2023.

46. PARTHASARATHY, Venkatakrisnan; SIMISCUKA, Anderson Augusto; O'CONNOR, Noel; MUNTEAN, Gabriel-Miro. Performance Evaluation of a Multi-User Virtual Reality Platform. In: *2020 International Wireless Communications and Mobile Computing (IWCMC)*. 2020, s. 934–939. Dostupné z [DOI: 10.1109/IWCMC48107.2020.9148390](https://doi.org/10.1109/IWCMC48107.2020.9148390).
47. ABBAS, Sami; SIMISCUKA, Anderson Augusto; MUNTEAN, Gabriel-Miro. A Platform Agnostic Solution for Inter-Communication between Virtual Reality Devices. In: 2019, s. 189–194. Dostupné z [DOI: 10.1109/WF-IoT.2019.8767230](https://doi.org/10.1109/WF-IoT.2019.8767230).
48. GUNKEL, Simon; PRINS, Martin; STOKKING, Hans; NIAMUT, Omar. WebVR meets WebRTC: Towards 360-degree social VR experiences. In: *2017 IEEE Virtual Reality (VR)*. 2017, s. 457–458. Dostupné z [DOI: 10.1109/VR.2017.7892377](https://doi.org/10.1109/VR.2017.7892377).
49. VI, Steven; SILVA, Tiago Silva da; MAURER, Frank. User Experience Guidelines for Designing HMD Extended Reality Applications. In: *Human-Computer Interaction – INTERACT 2019: 17th IFIP TC 13 International Conference, Paphos, Cyprus, September 2–6, 2019, Proceedings, Part IV*. Paphos, Cyprus: Springer-Verlag, 2019, s. 319–341. ISBN 978-3-030-29389-5. Dostupné z [DOI: 10.1007/978-3-030-29390-1\\_18](https://doi.org/10.1007/978-3-030-29390-1_18).
50. MICROSOFT. Optimal distance for placing holograms from the user. In: [b.r.]. Dostupné tiež z: <https://learn.microsoft.com/en-us/windows/mixed-reality/design/images/distanceguiderendering-950px.png>.
51. MICROSOFT. Optimal region for content. In: [b.r.]. Dostupné tiež z: <https://learn.microsoft.com/en-us/windows/mixed-reality/design/images/optimal-field-of-view-2.png>.
52. VR-COMPARE.COM. Horizontal field of view comparison. In: [b.r.]. Dostupné tiež z: <https://i.redd.it/26o3a9tcu9c81.png>.
53. KORECKO, Stefan; VAĽA, M.; FEKETE, M. VISUALIZATION OF EXPERIMENTAL DATA IN WEB-BASED VIRTUAL REALITY. In: 2021, s. 149–152. Dostupné z [DOI: 10.54546/MLIT.2021.42.97.001](https://doi.org/10.54546/MLIT.2021.42.97.001).
54. PEARSON, Karl. Contributions to the Mathematical Theory of Evolution. II. Skew Variation in Homogeneous Material. *Philosophical Transactions of the Royal Society of London. A* [online]. 1895, roč. 186, s. 343–414 [cit. 2023-01-22]. ISSN 02643820. Dostupné z: <http://www.jstor.org/stable/90649>.

55. FEKETE, Martin. Vizualizácia experimentálnych údajov v zdieľanej rozšírenej realite a jej používateľské rozhranie. In: Technická univerzita v Košiciach Fakulta, 2021.
56. MIKULA, Lukáš. Možnosti vizualizácie experimentálnych údajov vo webovej rozšírenej realite. In: Technická univerzita v Košiciach Fakulta, 2022.
57. NDMVR Demonstrator homepage. In: [b.r.]. Dostupné tiež z: <https://ndm.spc.gitlab.io/ndmvr>. dostupné: 19.2.2023.
58. *Position + orientation in 3D space*. 2015. Dostupné tiež z: <http://blog.dsky.co/tag/hmd/>. dostupné: 19.3.2023.

# Zoznam príloh

---

**Príloha A** CD médium – záverečná práca v elektronickej podobe,

**Príloha B** Používateľská príručka

**Príloha C** Systémová príručka

Obsah prílohy A:

- Adresár src - obsahuje projekt NDMVR so všetkými zdrojovými súbormi
- Adresár doc - obsahuje túto prácu, používateľskú príručku, anonymizované výsledky celého používateľského testovania a informovaný súhlas vo formáte PDF
- Adresár tex - obsahuje zdrojové kódy dokumentov v adresári doc

**Technická univerzita v Košiciach  
Fakulta elektrotechniky a informatiky**

**Používateľský zážitok vo virtuálnych  
prostrediach a jeho vyhodnocovanie**

**Používateľská príručka**

**2023**

**Natanael Prokop**

## Inštalácia a spúšťanie projektu NDMVR

Nutné súčasti projektu, ktoré je potrebné nainštalovať pred spustením projektu NDMVR:

- Npm
- Node.js
- Ngrok

Pre inštaláciu samotného nástroja je nevyhnutné:

1. Skopírujte celý obsah adresára src z prílohy B do počítača
2. Otvorte nový terminál v hlavnom adresári projektu
3. Zadajte príkaz `npm run dev`

V prípade ak je proces spustenia úspešný, v terminálovom okne adresára sa zobrazí výpis:

```
VITE v4.0.2 ready in 13373 ms
→ Local:   http://127.0.0.1:5173/
→ Network: use --host to expose
→ press h to show help
```

Spustenie NDMVR prostredníctvom Oculusu:

1. Otvorte aplikáciu ngrok
2. V aplikácii ngrok zadáme príkaz `ngrok http [číslo portu]`
3. Skopírujeme novú vygenerovanú link adresu pod ktorou aktuálne beží naša verzia projektu (adresa je zvýraznená na obrázku)

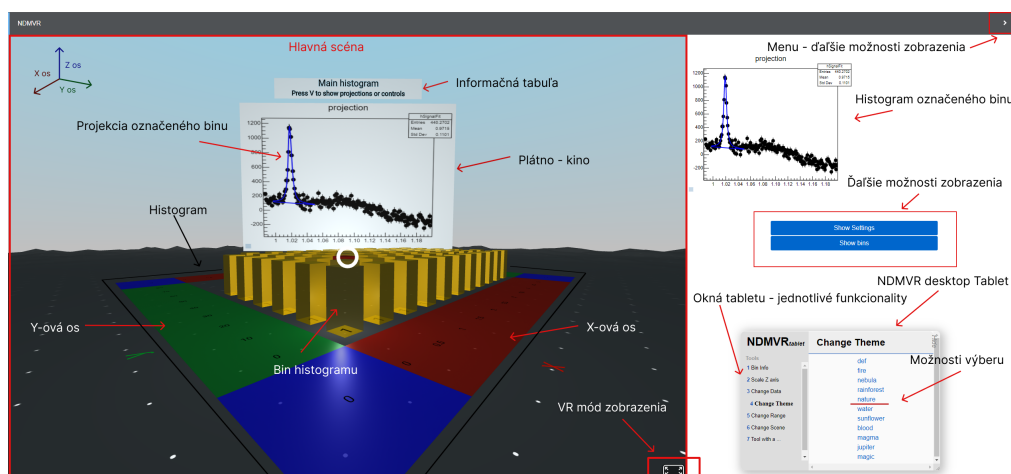
V prípade ak je proces spustenia úspešný, v aplikácii ngrok sa zobrazí výpis: Aktuálne je možné projekt otvoriť vo webovom prehliadači prostredníctvom

```
Session Status      online
Account             ██████████
Version             3.3.0
Region              Europe (eu)
Latency              -
Web Interface       http://127.0.0.1:4040
Forwarding           https://53d1-88-212-36-72.ngrok-free.app -> http://localhost:5173

Connections         ttl    opn    rt1    rt5    p50    p90
                   0      0      0.00  0.00  0.00  0.00
```

localhostu. Vďaka použitiu ngrok vieme k projektu NDMVR pristúpiť aj skrze zariadenie Oculus Quest 2. Nevyhnutné je pripojené v tej istej sieti ako počítač. Pre spustenie nástroja v zariadení Oculus Quest 2 stačí použiť link z aplikácie ngrok ktorý je zvýraznený na obrázku. Pre vypnutie programu je možné použiť klávesovú skratku Ctrl + C v oboch oknách terminálu/aplikácie ngrok.

## Základného zobrazenie NDMVR a jeho možnosti



Menu NDMVR ponúka zobrazenia:

- DYNAMIC TH3 histogram example
- GITLAB data projection example
- PHYSICS data projection example
- NDMVR monitoring adaptation
- NDMVR Shared Projection example

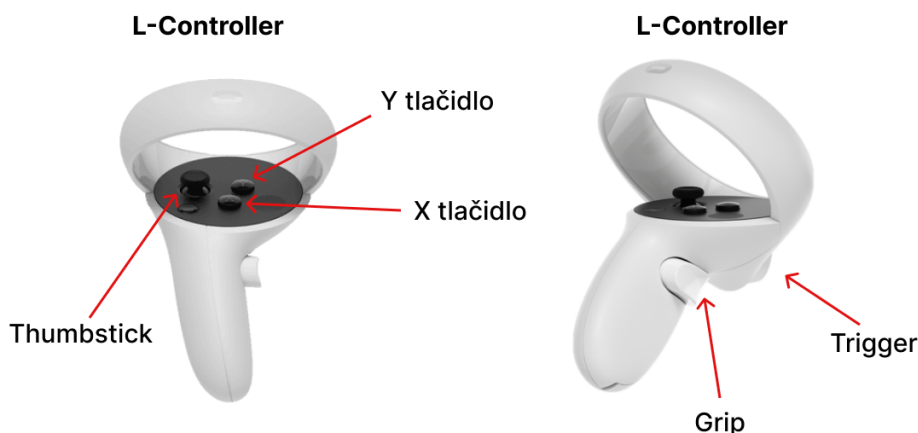
Zobrazenie PHYSICS data projection example/GITLAB data projection example má k dispozícii najširší výber možností Tablet. Pri skúšaní nástroja odporúčame použiť jedno z týchto zobrazení.

## Ovládanie NDMVR - Oculus verzia

Pre správne zobrazenie verzie NDMVR pre Oculus Quest 2 je nevyhnutné vstúpiť do režimu VR. Interakcia s tabletom je navrhnutá tak, aby držanie tlačidla Grip aktivovalo jeho ovládacie prvky, preto je pre akékoľvek ovládanie tabletu nevyhnutné držať tlačidlo grip stlačené.



## Oculus Controller Controls



### A. Ovládanie tabletu:

**Grip + Y** = nasledujúce okno/nástroj tabletu

**Grip + X** = predchádzajúce okno/nástroj tabletu

**Grip + Thumbstick Up** = Zmena výšky binov/zmena aktívneho zobrazenia

**Grip + Thumbstick Down** = Zmena výšky binov/zmena aktívneho zobrazenia

### B. Ovládanie pohybu:

**Thumbstick** = relatívny pohyb voči pohľadu používateľa (v jednej rovine)

**Thumbstick + Trigger** = fly mode relatívny voči pohľadu používateľa

**DoubleClick Trigger** = Automatické pristátie

## Ovládanie NDMVR - Desktop verzia

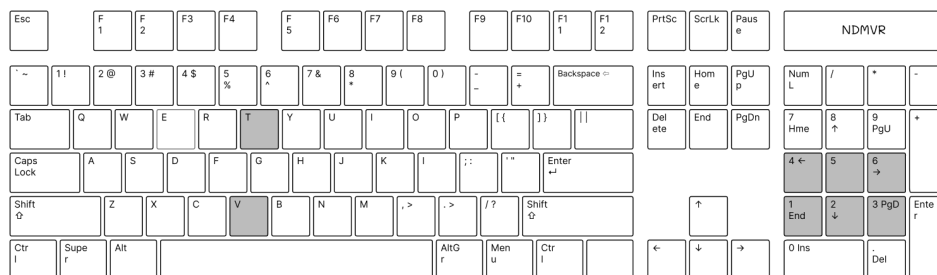
Klávesové skratky pre NDMVR budú funkčné pri vypnutom VR móde, len v prípade nachádzajúceho sa kurzora na hlavnej scéne.

**Tablet komponent** je možné otvoriť a zatvoriť pomocou tlačidla 'T'.

**Klávesové skratky a možnosti zobrazenia** sú k dispozícii po stlačení klávesy 'V'.

**Označenie binu** je možné realizovať kliknutím na daný bin prostredníctvom kurzora scény. Následne sa jeho 2D projekcia zobrazí na komponente plátna, ako aj v bočnom paneli stránky v prípade vypnutého VR módu.

## Desktop Controls



**Zmena výšky binu, témy, rozsahu zobrazenia, ako aj dát zobrazenia** je možná prostredníctvom tabletu, po kliknutí na dané okno tabletu s uvedenou funkčnosťou/stlačenie alternatívnej numerickej skratky, stačí zvoliť preferovanú možnosť z ponuky možností na pravej strane tabletu. V prípade zmenšenia veľkostí, stlačiť adekvátne znamienko operácie zmenšenia/zväčšenia.

Desktop verzia Tabletú nie je podporovaná pre iné zariadenia ako oculus v režime VR módu.

## Spustenie developerského zobrazenia

Najjednoduchší spôsobom je použitie aplikácie Meta Quest Developer Hub. Ten zahŕňa verziu pre windows zariadenia <sup>1</sup> a mac zariadenia <sup>2</sup>.

Pre spustenie konzoly je nutné <sup>3</sup>:

### 1. Nastavenie MQDH a Oculusu:

1. Otvorte aplikáciu MQDH.
2. Vyberte správnu cestu k ADB (Android Debug Bridge).
3. Prihláste sa pomocou účtu vývojára.
4. Použite overený účet, prípadne ho overte (to umožní režim vývojára).
5. Pripojte Oculus k počítaču pomocou kábla USB-C.
6. Po zobrazení výzvy na Oculusu pre získanie prístupu k údajom kliknite na položku - Povolit.
7. V aplikácii MQDH, zvolte pridať nové zariadenie.
8. Vyberte svoje zariadenie (Ndm Spc).

<sup>1</sup><https://developer.oculus.com/downloads/package/oculus-developer-hub-win/>

<sup>2</sup><https://developer.oculus.com/downloads/package/oculus-developer-hub-mac/>

<sup>3</sup><https://developer.oculus.com/documentation/web/browser-remote-debugging/>

9. Uistite sa, že režim vývojár je zapnutý

2. Možnosti pripojenia:

1. Prostredníctvom USB pripojenia
2. Prostredníctvom WiFi pripojenia

(a). ADB prostredníctvom USB (rýchlejšie):

1. (V prehliadači Chrome): Otvorte na počítači: `chrome://inspect/#devices`.
2. (V prehliadači Chrome): Vyberte: objaviť zariadenia USB
3. (V prehliadači Chrome): Vyberte: predvolenú možnosť
4. (V prehliadači Chrome): Zo zoznamu kariet otvorených na zariadení Oculus, vyberte kartu s názvom NDMVR
5. (V prehliadači Chrome): následne stlačte tlačidlo 'inspect' danej karty NDMVR, pre otvorenie okna s vývojárskou konzolou

(b). ADB prostredníctvom Wifi (pomalšie):

1. Zapnite ADB cez Wifi v MQDH
2. Pridajte nový príkaz ADB (`adb shell ip route`) do MQDH alebo cmd
3. (V prehliadači Chrome): Otvorte: `chrome://inspect/#devices`
4. (V prehliadači Chrome): Vyberte: Objaviť ciele siete
5. (V prehliadači Chrome): Zadajte IP adresu z vytvoreného príkazu ADB a predvolený port: 5555
6. (V prehliadači Chrome): Zo zoznamu kariet otvorených na zariadení Oculus, vyberte kartu s názvom NDMVR
7. (V prehliadači Chrome): následne stlačte tlačidlo 'inspect' danej karty NDMVR, pre otvorenie okna s vývojárskou konzolou

**Technická univerzita v Košiciach  
Fakulta elektrotechniky a informatiky**

**Používateľský zážitok vo virtuálnych  
prostrediach a jeho vyhodnocovanie**

**Systemová príručka**

**2023**

**Natanael Prokop**

## Komponent <oculusThumbstickController>

### Funkcionalita:

Komponent 'thumbstickOculusController' je zodpovedný za ovládanie pohybu v aplikácii pomocou thumbsticku oculusu. Tento komponent využíva knižnicu A-Frame a importuje funkciu 'getActiveTool' z komponentu 'Tablet'.

### Schéma:

- **acceleration:** Typ 'number', predvolená hodnota 40. Určuje zrýchlenie pohybu.
- **fly:** Typ 'boolean', predvolená hodnota false. Označuje, či je povolený režim letu.
- **horizontalAxis:** Typ 'string', hodnoty 'x', 'y', 'z', predvolená hodnota 'x'. Určuje horizontálnu os pohybu.
- **verticalAxis:** Typ 'string', hodnoty 'x', 'y', 'z', predvolená hodnota 'z'. Určuje vertikálnu os pohybu.
- **enabled:** Typ 'boolean', predvolená hodnota true. Označuje, či je povolené ovládanie pohybu.
- **adEnabled:** Typ 'boolean', predvolená hodnota true. Označuje, či je povolený pohyb v osiach 'x' a 'z'.
- **adInverted:** Typ 'boolean', predvolená hodnota false. Určuje, či je invertovaný smer pohybu v osiach 'x' a 'z'.
- **wsEnabled:** Typ 'boolean', predvolená hodnota true. Označuje, či je povolený pohyb v ose 'y'.
- **wsInverted:** Typ 'boolean', predvolená hodnota false. Určuje, či je invertovaný smer pohybu v ose 'y'.

### Výstup:

Zmena polohy entity reprezentujúcej používateľa.

**Ponúkané možnosti:**

1. Možnosť *fly* ovplyvňuje režim letu. Ak je táto možnosť zapnutá, pohyb bude povolený aj vo vertikálnom smere, čo umožňuje let. Ak je táto možnosť vypnutá, pohyb bude obmedzený iba na horizontálne osi.
2. Možnosť *adInverted* umožňuje invertovať smer pohybu v osiach 'x' a 'z'. Ak je táto možnosť zapnutá, smer pohybu bude invertovaný. Ak je táto možnosť vypnutá, pohyb bude v smeroch 'x' a 'z' bežný.
3. Možnosť *wsInverted* umožňuje invertovať smer pohybu v ose 'y'. Ak je táto možnosť zapnutá, smer pohybu v osi 'y' bude invertovaný. Ak je táto možnosť vypnutá, pohyb v ose 'y' bude bežný.

**Jednotlivé funkcie:**

- **init:** Táto funkcia sa volá pri inicializácii komponentu. V tejto funkcii sa inicializujú premenné, nastavujú počiatočné hodnoty a pridávajú event listeners na rôzne udalosti (napríklad 'gripdown', 'gripup', 'triggerdown', 'triggerup' a 'thumbstickmoved').
- **update:** Táto funkcia sa volá pri aktualizácii komponentu. V tejto funkcii sa získava referenčný objekt scény (cameraWrapper) a ukladá sa do premennej sceneElement.
- **tick:** Táto funkcia sa volá každý frame (obnovovaciu frekvenciu určuje hardvér). V tejto funkcii sa aktualizuje rýchlosť pohybu a pozícia objektu na základe týchto údajov.
- **updateVelocity:** Táto funkcia slúži na aktualizáciu rýchlosti pohybu. Na základe údajov z ovládača (thumbstick) a nastavených parametrov (ako je zrýchlenie, invertovanie osí) sa upravuje rýchlosť pohybu v horizontálnych a vertikálnych osiach.
- **getMovementVector:** Táto funkcia slúži na výpočet vektora pohybu objektu na základe rýchlosti a rotácie kamery. Vektory sú transformované vzhľadom na rotáciu kamery a aplikuje sa ich smer pohybu.
- **thumbstickMoved:** Táto funkcia sa volá pri udalosti 'thumbstickmoved' (pohyb thumbsticku). Získava informácie o pohybe thumbsticku a ukladá ich do premennej tsData.

- **remove:** Táto funkcia sa volá pri odstránení komponentu. V tejto funkcii sa odstraňuje event listener pre udalosť 'thumbstickmoved'.

**Použitie:**

Komponent je možné použiť s defaultným nastavením zavolaním *oculus-thumbstick-movement-controller = ''*. Komponent je použitý v komponente *NdmVrCamera* pre ľavý controller oculusu.

**Komponent <NdmVrTablet>**

**Subor:** `src/lib/components/Tablet/index.jsx`

**Funkcionalita:**

Tento súbor je súčasťou komponentu s názvom **DesktopTablet**. Komponent je implementovaný v Reacte a slúži na zobrazenie a interakciu s tabletom v prostredí prehliadača.

**Základné informácie:**

Komponent zobrazuje rôzne nástroje (*tools*) a umožňuje interakciu s nimi na tablete. Na základe vybranej položky v danom nástroji sa menia aktívne zdroje dát, témy, rozsahy a scény. Komunikácia medzi tabletom a aplikáciou prebieha pomocou objektu *tabletComunicator*, ktorý je inštanciou triedy *Subject* z knižnice *rxjs*.

**Vstup:**

Vstupy komponentu sú rôzne premenné a stavy, ako napríklad *activeToolLocal*, *activeSource*, *activeTheme*, *activeRange*, *activeScene*, *binInfo*, *scaleCoeficient*, *activeDevice*, *dataSources*, *dataThemes*, *dataRange*, *dataScene*.

**Výstup:**

Výstupom komponentu je vizuálna reprezentácia interaktívneho tabletu.

**Importy:**

- **React:** hlavná knižnica na prácu s React komponentami a stavmi.
- **useState:** hook pre správu stavov komponentu.
- **useRef:** hook pre získanie referencie na element v DOM.

- **useEffect**: hook pre spracovanie vedľajších efektov v komponente.
- **useContext**: hook pre prístup k hodnotám zo *StoreContext*.
- **Subject**: trieda pre vytvorenie a správu objektu na komunikáciu medzi komponentmi.
- **StoreContext**: kontext pre prístup k hodnotám z kontextu *StoreContext*.
- **aframe**: knižnica pre prácu s A-Frame, frameworkom na vytváranie virtuálnej reality v prehliadači.
- **histogramTH2Service**, **histogramTH1Service** - služby pre správu histogramov.

### Jednotlivé funkcie:

- Funkcia **DesktopTablet**: Toto je hlavná funkcia komponentu, ktorá sa exportuje. Obsahuje v sebe všetky ostatné funkcie, efekty a logiku komponentu. Hlavným účelom tejto funkcie je vykresliť a spracovať interakcie s obsahom tabletu.
- Funkcia **handleComunication**: Táto funkcia slúži na spracovanie komunikácie medzi rôznymi časťami aplikácie. Vstupom je objekt **data**, ktorý obsahuje rôzne typy dát a príkazov. Funkcia na základe hodnôt v objekte **data** vykonáva príslušné akcie, ako zmena aktívneho nástroja, zmena zdroja dát, zmena tém, zmena rozsahu alebo zmena scény.
- Funkcia **renderOculusTablet**: Táto funkcia slúži na vykreslenie obsahu a rozhrania tabletu, ak je používateľ na zariadení Oculus. Je to blok switch príkazov, ktorý vracia rôzne UI komponenty na základe hodnoty premennej **activeToolLocal**. Jednotlivých prípady:
  1. Prípad 0: Vráti komponent **<a-text>**, ktorý zobrazuje informácie o vybranom bin, vrátane jeho hodnoty, x-koordináty, y-koordináty a z-koordináty.
  2. Prípad 2: Vráti komponent **<a-text>**, ktorý poskytuje inštrukcie na zmenu vizuálnej výšky binov pomocou gripu a joysticku na ľavom ovládači.
  3. Prípad 1: Vráti komponent **<a-box>**, ktorý reprezentuje tablet. Má špecifické vlastnosti ako farbu, veľkosť, rotáciu a pozíciu.



4. Prípád 3: Vrátí skupinu komponentov `<a-text>`, ktoré reprezentujú dostupné zdroje dát. Každý `<a-text>` zobrazuje názov zdroja. Aktívny zdroj je zvýraznený inou farbou.
  5. Prípád 4: Vrátí skupinu komponentov `<a-text>`, ktoré reprezentujú dostupné témy dát. Každý `<a-text>` zobrazuje názov témy. Aktívna téma je zvýraznená inou farbou.
  6. Prípád 5: Vrátí skupinu komponentov `<a-text>`, ktoré reprezentujú dostupné rozsahy dát. Každý `<a-text>` zobrazuje hodnotu rozsahu. Aktívny rozsah je zvýraznený inou farbou.
  7. Prípád 6: Vrátí skupinu komponentov `<a-text>`, ktoré reprezentujú dostupné scény dát. Scény sú rozdelené do viacerých stĺpcov na základe hodnoty `scenesPerColumn`. Každý `<a-text>` zobrazuje názov scény. Aktívna scéna je zvýraznená inou farbou.
  8. Predvolený prípad: Vrátí komponent `<a-text>`, ktorý oznamuje, že vybraný nástroj nemá rozhranie pre Oculus Tablet.
- Funkcia **handleActiveToolChange**: Táto funkcia slúži na zmenu aktívneho nástroja na tablete. Na základe vstupnej hodnoty `tool` mení hodnotu premennej `activeTool`.
  - Funkcia **renderDefaultTablet**: Táto funkcia slúži na vykreslenie obsahu a rozhrania tabletu, ak používateľ nie je na zariadení Oculus. Opäť na základe hodnoty `activeToolLocal` vykresľuje rôzne prvky a texty na tablete.
  - Funkcia **subscribeToComunicator**: Táto funkcia sa volá pri inicializácii komponentu a slúži na predplatné komunikátoru `tabletComunicator`. V podstate sa nastavuje odber, aby komponent počúval na zmeny v komunikátore a reagoval na ne.
  - Funkcia **setScaleForOculus**: Táto funkcia slúži na zmenu mierky (škály) zobrazenia v prostredí Oculus. Na základe hodnoty `scale` mení koeficient mierky a aktualizuje ju v reálnom čase.
  - Funkcia **getBinInfoTool**: Táto funkcia slúži na zobrazenie informácií o binách (jednotkách) v tabuľke. Ak je aktívne zariadenie desktop, funkcia vykresľuje informácie o hodnote, x-ovej pozícii, y-ovej pozícii a z-ovej pozícii. Vrátí komponent s tabuľkou obsahujúcou tieto informácie. Funkcia má vstup bez parametrov a výstupom je komponent s tabuľkou informácií o binoch.

- Funkcia **scaleZaxis**: Táto funkcia slúži na zmenu hodnoty a mierky osi Z. V závislosti od aktuálneho **scaleCoeficient** sa mení hodnota **jumpValue** a **fixed**. Následne sa vykresľuje komponent obsahujúci tlačidlá pre zväčšenie a zmenšenie hodnoty **scaleCoeficient** a zároveň sa mení mierka histogramových objektov. Funkcia má vstup bez parametrov a výstupom je komponent s tlačidlami pre zmenšenie a zväčšenie hodnoty **scaleCoeficient**.
- Funkcia **changeDataSource**: Táto funkcia slúži na zmenu zdroja dát. Vykresľuje komponent s možnosťami zmeny aktívneho zdroja dát na základe dostupných **dataSources**. Ak je aktívny zdroj rovnaký ako **activeSource**, komponent je zvýraznený. Má vstup bez parametrov a výstupom je komponent s možnosťami zmeny zdroja dát.
- Funkcia **changeDataTheme**: Táto funkcia slúži na zmenu témy dát. Vykresľuje komponent s možnosťami zmeny aktívnej témy na základe dostupných **dataThemes**. Ak je aktívna téma rovnaká ako **activeTheme**, komponent je zvýraznený. Má vstup bez parametrov a výstupom je komponent s možnosťami zmeny témy dát.
- Funkcia **changeDataRange**: Táto funkcia slúži na zmenu rozsahu dát. Vykresľuje komponent s možnosťami zmeny aktívneho rozsahu na základe dostupných **dataRange**. Ak je aktívny rozsah rovnaký ako **activeRange**, komponent je zvýraznený. Má vstup bez parametrov a výstupom je komponent s možnosťami zmeny rozsahu dát.
- Funkcia **changeDataScene**: Táto funkcia slúži na zmenu scény dát. Vykresľuje komponent s možnosťami zmeny aktívnej scény na základe dostupných **dataScene**. Ak je aktívna scéna rovnaká ako **activeScene**, zvýrazní sa. Má vstup bez parametrov a výstupom je komponent s možnosťami zmeny scény dát.

Každá z týchto funkcií slúži na vykreslenie konkrétnej časti rozhrania tabletu a zároveň reaguje na interakcie používateľa a mení hodnoty príslušných premenných, ktoré ovplyvňujú vykreslenie a správanie sa komponentu.

## Komponent <NdmVrLaboratory>

### Funkcionalita:

Komponent **NdmVrLaboratory** slúži na zobrazenie virtuálneho laboratória v rozhraní VR pomocou prostredia vytvoreného v komponente **a-frame environment**.

Vlastnosť **preset** určuje, aké prostredie sa má zobrazíť. Môže byť nastavená na rôzne predvolené hodnoty scén alebo sa dá meniť pomocou **props**.

### Importy:

- **React**: hlavná knižnica na prácu s React komponentami a stavmi.
- **useContext**: hook pre prístup k hodnotám zo *StoreContext*.
- **StoreContext**: kontext pre prístup k hodnotám z kontextu *StoreContext*.
- **aframe**: knižnica pre prácu s A-Frame, frameworkom na vytváranie virtuálnej reality v prehliadači.

### Funkcia `NdmVrLaboratory`:

- Funkcia prijíma **props** ako parameter.
- Premenná **preset** je inicializovaná hodnotou **props?.scene** alebo hodnotou "labs", ak vlastnosť **scene** neexistuje v **props**.
- Komponent vráti fragment `<React.Fragment>`, ktorý obsahuje `<a-entity>` komponent s nastavením prostredia.
- Vlastnosť **environment** komponentu `<a-entity>` je nastavená na **preset**, ktorý je odvodený z **props** alebo predvolene nastavený na "labs".

### Komentáre:

Sú uvedené niektoré zmeny a alternatívne nastavenia pre komponent `<a-entity>` prostredia. Komentáre označené –MAPS—popisujú rôzne predvolené hodnoty pre vlastnosť **preset** v závislosti na scéne.

Komponent `<aframe-environment-component>`:

`<aframe-environment-component>` je jedným z mnohých komponentov dostupných v knižnici A-Frame.

Tento komponent umožňuje jednoduché nastavenie prostredia (scény) pre virtuálnu realitu.

Poskytuje rôzne predvolené hodnoty pre nastavenie scény vrátane textúr, farieb neba a horizontu, osvetlenia, tieňovania a ďalších vlastností.

Pomocou atribútu **preset** je možné vybrať predvolené prostredie zo zoznamu dostupných možností (napríklad "default", "forest", "japan", "starryätd.).

Okrem **preset** je možné použiť aj ďalšie atribúty, ako je nastavenie textúry zeme, farby horizontu, farby oblohy, zobrazenie tieňov, nastavenie štýlu a ďalšie.

## Komponent <aframe-environment-component>

### Funkcionalita:

Komponent <aframe-environment-component> je užitočný pre rýchle nastavenie základného prostredia pre VR scény. Umožňuje vytvárať pôsobivé a realistické vizuálne prostredia s minimálnym úsilím a kódom.

### Schéma:

Vlastnosti (Schéma) komponentu zahŕňajú možnosti ako aktívne prostredie, predvolená scéna, semeno, typ oblohy, osvetlenie, tieňovanie, farby a ďalšie.

Hlavné nastaviteľné vlastnosti:

- **active:** Určuje, či je prostredie aktívne alebo nie.
- **preset:** Určuje preddefinovaný vzhlad scény. Môže byť nastavený na hodnoty ako "none" (žiadny preset), "default" (predvolený preset), "forest", "egypt", "checkerboard" a ďalšie.
- **seed:** Určuje hodnotu semena pre generovanie náhodných prvkov v scéne.
- **skyType:** Určuje typ neba v scéne. Môže byť nastavený na "none" (žiadne nebo), "color" (jednofarebné nebo), "gradient" (farebný prechod) alebo "atmosphere" (atmosférický efekt).
- **background:** Určuje obrázok alebo farbu použitú ako pozadie scény.
- **skyColor:** Určuje farbu neba.
- **horizonColor:** Určuje farbu horizontu.
- **lighting:** Určuje typ osvetlenia scény. Môže byť nastavený na "none" (žiadne osvetlenie), "distant" (vzdialené svetlo) alebo "point" (bodové svetlo).
- **shadow:** Určuje, či sa majú generovať tieň v scéne.
- **shadowSize:** Určuje veľkosť tieňov.
- **lightPosition:** Určuje pozíciu svetla v scéne.
- **fog:** Určuje hodnotu hmlistosti v scéne.
- **flatShading:** Určuje, či sa má použiť ploché zatiernenie (flat shading).
- **playArea:** Určuje veľkosť plochy pre hráča v scéne.

- **ground**: Určuje typ povrchu scény. Môže byť nastavený na "none" (žiadny povrch), "flat" (plochý povrch), "hills" (pahorky), "canyon" (kaňon), "spikes" (ostne), "noise" (šum).
- **groundYScale**: Určuje výšku povrchu scény.
- **groundTexture**: Určuje textúru použitú na povrchu scény. Môže byť nastavený na "none" (žiadna textúra), "checkerboard" (šachovnica), "squares" (štvorce), "walkernoise" (walkernoise textúra).
- **groundColor**: Určuje farbu povrchu scény.
- **groundColor2**: Určuje druhú farbu povrchu scény (používa sa pre niektoré typy povrchov).
- **dressing**: Určuje typ ozdôb v scéne. Môže byť nastavený na "none".

#### Defaultne nastavené scény:

Komponent obsahuje aj prednastavené hodnoty pre niektoré typy prostredí, napríklad "default", "labs", "ndmvrätd". Tieto hodnoty sa použijú, ak nie sú špecifikované iné hodnoty pre komponent.

#### Jednotlivé funkcie:

- **Funkcia init** - Nicializuje niektoré premenné a objekty:
  - Premenná **environmentData** je objekt, ktorý uchováva údaje o prostredí.
  - Premenná **STAGE\_SIZE** má hodnotu 200 a predstavuje veľkosť pódia.
  - Objekt **assets** obsahuje rôzne položky reprezentujúce objekty, ktoré môžu byť vkladané do scény. Jednou z nich je napríklad pole **arches**, ktoré obsahuje viacero objektov reprezentujúcich oblúky. Tieto objekty majú rôzne vlastnosti, ako napríklad typ "mesh" (sieť), súradnice vrcholov a steny.
- Táto časť kódu slúži na nastavenie počiatočných hodnôt a na načítanie počiatočných zdrojov pre ďalšie časti programu.
- **Funkcia update** - Metóda, ktorá sa volá pri aktualizácii prostredia. Vykonáva rôzne operácie na základe parametrov prostredia a aktualizuje vizuálne prvky prostredia. Nastavenie sky parametrov, background, ground a ďalších parametrov schemy komponentu.

- **Funkcia `getFogColor`** - Metóda, ktorá vráti farbu mlhy na základe typu prostredia a intenzity.
- **Funkcia `logPreset`** - Slúži na výpis parametrov prostredia.
- **Funkcia `dumpParametersDiff`** - Slúži na výpis rozdielov medzi aktuálnymi parametrami prostredia a predvolenými hodnotami.
- **Funkcia `random`** - Generuje náhodné číslo na základe zadaného semena.
- **Funkcia `updateGround`** - Aktualizuje geometriu a vzhľad povrchu prostredia.
- **Funkcia `updateDressing`** - Stará sa o aktualizáciu obliekania (dressing) v scéne. Vytvára objekty (kocky, pyramidy, valce alebo iné) na základe nastavených parametrov prostredia. Objekty sú pridané do kontajnera `e` a pridené ako mesh (objekt siete) k prvku s id `dressing`.
- **Funkcia `createStars`** - Vytvára hviezdy v scéne. Vytvára bufferovú geometriu a materiál pre hviezdy a pridáva ich ako mesh k prvku s id `stars`.
- **Funkcia `setStars`** - Slúži na nastavenie počtu hviezd, ktoré majú byť viditeľné v scéne. Parametrom `e` sa určuje počet hviezd a je obmedzený na rozsah od 0 do 2000.
- **`AFRAME.registerShader("skyshader")`** - Táto časť kódu registruje vlastný shader s názvom "skyshader". Shader obsahuje rôzne uniformné premenné a funkcie, ktoré sa používajú na vykreslenie sférického prostredia s nebeskou textúrou.
- Komponent umožňuje vytvorenie a nastavenie rôznych vlastností a parametrov scén.

## Komponent `<leftOculusController>`

### Funkcionalita:

Komponent zachytáva jednotlivé udalosti ľavého controllera a následne ich odsiela do komponentu `Pculus Controller` pre detailnejšiu funkcionality, obsluhu daného stavu, ako aj jeho reakcií. Nastavovaný časový limit pre opätovné stlačenie thumbsticku.

**Jednotlivé funkcie:**

- Metóda **init** inicializuje komponent a nastavuje niektoré premenné. Zoznam týchto premenných:
  - **gripActive** je logická premenná, ktorá určuje, či je tlačidlo "grip" na ovládači aktívne (stlačené).
  - **sourceSlot** je logická premenná, ktorá určuje, či je zdrojový slot aktívny.
  - **speed** je číselná premenná, ktorá určuje rýchlosť pohybu.
  - **toolChangeInProgress** je logická premenná, ktorá určuje, či prebieha zmena nástroja.

tiež pridáva poslucháčov udalostí pre ovládač Oculus. Tu je zoznam týchto udalostí:

- **gripdown** a **gripup** naslúchajú na stlačenie a uvoľnenie tlačidla "grip" na ovládači.
- **thumbstickdown** a **thumbstickmoved** naslúchajú na stlačenie a pohyb joysticku na ovládači.
- **ybuttondown**, **xbuttondown**, **ybuttonup** a **xbuttonup** naslúchajú na stlačenie a uvoľnenie tlačidiel "Yä "X" na ovládači.

V metóde **init** sú definované rôzne podmienky a funkcie, ktoré sa vykonávajú na základe aktívneho nástroja na ovládači. Tieto funkcie sú implementované v module **oculusController**. Niektoré z týchto funkcií zahŕňajú zmenu mierky histogramu, zmenu zdroja, zmenu témy, zmenu rozsahu a zmenu scény.

- Metóda **update** je volaná pri aktualizácii komponentu a volá funkciu **oculusUpdateCameraReference**, ktorá aktualizuje referenciu na kameru.

**Komponent <oculusController>****Funkcionalita:**

Obsluha funkcií controllera, predovšetkým možností výberu, definovaná citlivosť spolu so slepou zónou thumbsticku.

## Jednotlivé funkcie:

Implementované boli funkcie, ktoré pracujú s histogramom a slúžia na zmenu rôznych parametrov a nastavení. Opis jednotlivých funkcií:

- **histogramScaleChanger**: Táto funkcia slúži na zmenu skalovania binov v histograme. Zmena sa vykonáva na základe udalostí z thumbsticku (joysticku). Ak je thumbstick stlačený smerom nahor (**event.detail.y > 0.1**), funkcia posiela signál na ďalšiu akciu s parametrom **scale** nastaveným na hodnotu  $-0.1$  (negatívna zmena). Ak je thumbstick stlačený smerom nadol (**event.detail.y < -0.1**), funkcia posiela signál s parametrom **scale** nastaveným na hodnotu  $0.1$  (pozitívna zmena).
- **dataSourceChanger**: Táto funkcia slúži na zmenu zdroja dát (*data source*) pre histogram. Opäť sa vykonáva na základe udalostí z thumbsticku. Ak je thumbstick stlačený smerom nahor, funkcia posiela signál na ďalšiu akciu s parametrom **source** nastaveným na hodnotu **'next'** (posun na ďalší zdroj). Ak je thumbstick stlačený smerom nadol, funkcia posiela signál s parametrom **source** nastaveným na hodnotu **'prev'** (posun na predchádzajúci zdroj).
- **dataThemeChanger**: Táto funkcia slúži na zmenu témy dát (*data theme*) pre histogram. Podobne ako predchádzajúce funkcie, zmena sa vykonáva na základe udalostí z thumbsticku. Ak je thumbstick stlačený smerom nahor, funkcia posiela signál na ďalšiu akciu s parametrom **theme** nastaveným na hodnotu **'next'** (posun na ďalšiu tému). Ak je thumbstick stlačený smerom nadol, funkcia posiela signál s parametrom **theme** nastaveným na hodnotu **'prev'** (posun na predchádzajúcu tému).
- **dataRangeChanger**: Táto funkcia slúži na zmenu rozsahu dát (*data range*) pre histogram. Opäť sa vykonáva na základe udalostí z thumbsticku. Ak je thumbstick stlačený smerom nahor, funkcia posiela signál na ďalšiu akciu s parametrom **range** nastaveným na hodnotu **'next'** (posun na ďalší rozsah). Ak je thumbstick stlačený smerom nadol, funkcia posiela signál s parametrom **range** nastaveným na hodnotu **'prev'** (posun na predchádzajúci rozsah).
- **dataSceneChanger**: Táto funkcia slúži na zmenu scény dát (*data scene*) pre histogram. Podobne ako predchádzajúce funkcie, zmena sa vykonáva na základe udalostí z thumbsticku. Ak je thumbstick stlačený smerom nahor,



funkcia posiela signál na ďalšiu akciu s parametrom **scene** nastaveným na hodnotu **'next'** (posun na ďalšiu scénu). Ak je thumbstick stlačený smerom nadol, funkcia posiela signál s parametrom **scene** nastaveným na hodnotu **'prev'** (posun na predchádzajúcu scénu).

## Komponent <HistogramGenerator>

### Funkcionalita:

Slúži na vygenerovanie histogramu TH1, TH2, TH3.

### Jednotlivé funkcie:

Opísané sú len modifikované funkcie.

- Funkcia **createZLabels** vytvára popisy a osi pre os  $z$ . Prijíma niekoľko parametrov, ako sú **elements**, **normalAxis**, **reversedAxis**, **maxLength**, **maxYLength**, **maxZLength**, **zLabels**, **zTitle** a **count**. Vytvára viacero entít osí a priradzuje ich do poľa **normalAxis** a **reversedAxis**. Popisy osí  $z$  sú umiestnené na základe poskytnutých parametrov.
- Funkcia **createYLabels** vytvára popisy a osi pre os  $y$ . Prechádza hodnotami osi  $y$  a vytvára pre každú hodnotu príslušnú entitu osi. Popisy osí  $y$  sú umiestnené na základe poskytnutých parametrov.
- Funkcia **createXLabels** vytvára popisy a osi pre os  $x$ . Podobne ako predchádzajúca funkcia, prechádza hodnotami osi  $x$  a vytvára príslušné entity osí. Popisy osí  $x$  sú umiestnené na základe poskytnutých parametrov.
- Funkcia **createEnvironment** je zodpovedná za vytvorenie prostredia pre trojrozmernú vizualizáciu. Nastavuje nadpis pre vizualizáciu a umiestňuje ho na základe poskytnutých parametrov.

Funkcie slúžia taktiež pre zmenu polohy histogramu, plátna a informačnej tabule.

## Komponent <THProjection>

### Funkcionalita:

Komponent THProjection je zodpovedný za zobrazenie a manipuláciu s rôznymi projekciami histogramov. Využíva knižnicu React a rôzne komponenty a funkcie z balíčkov **@patternfly/react-core** a **@ndmspc/react-ndmspc-core**.

### Ponúkané možnosti:

- Komponent zobrazuje sidebar, ktorý obsahuje komponent **NdmVrHistogramScene** z knižnice **@ndmspc/react-ndmspc-core**. Tento komponent zobrazuje 3D scénu s histogramom a umožňuje interakciu s ním.
- Komponent umožňuje výber rôznych parametrov pre zobrazenie histogramu, ako je téma, scéna, rozsah a zobrazenie.
- Komponent poskytuje možnosť otvoriť a načítať súbor s histogramom.
- Komponent zobrazuje projekciu histogramu v samostatnom paneli.
- Komponent umožňuje vykonávanie rôznych príkazov a manipuláciu s projekciami histogramu.

Okrem toho kód obsahuje rôzne premenné a funkcie, ktoré slúžia na správu stavu a manipuláciu s dátami.

### Deklarované premenné:

- **let store**: Deklaruje objektovú premennú **store** s vlastnosťami **source**, **theme**, **scale**, **view** a **scene**. Týmto vlastnostiam sú priradené hodnoty vrátené funkciou **executorStore**.
- Deklarácie konštánt:
  - **const styles**: Deklaruje objektovú premennú **styles** s CSS štýlmi pre rôzne elementy.
  - **const projectionPanelIds**: Deklaruje pole **projectionPanelIds** s viacerými ID panelov.
  - **const views**: Deklaruje pole **views** s rôznymi možnosťami zobrazenia.
  - **const themes**: Deklaruje pole **themes** s rôznymi možnosťami témy.
  - **const scenes**: Deklaruje pole **scenes** s rôznymi možnosťami scény.
  - **const ranges**: Deklaruje pole **ranges** s rôznymi možnosťami rozsahu.

### Použité funkcie:

- **readProjectionFromFile**: Táto funkcia slúži na čítanie dát zo súboru v špecifickom formáte a následné vykreslenie histogramových projekcií. Funkcia prijíma nasledujúce parametre: **filename** (názov súboru), **basePath** (základná

cesta), **ix** a **iy** (indexy), **projName** (názov projekcie) a **targetElmIds** (identifikátory cieľových elementov). Využíva sa načítanie súboru, čítanie objektu zo súboru, čistenie existujúcich elementov, vykreslenie histogramových projekcií a ukladanie vytvorených painterov.

- **useExecutorStore**: Táto funkcia je vlastný React hook, ktorý slúži na správu stavových hodnôt a funkcií. Využíva sa na sledovanie a aktualizáciu stavu v rámci komponentu.
- **createTH1Projection**: Táto funkcia slúži na vytvorenie histogramovej projekcie v 1D. Pri vytváraní projekcie sa berú do úvahy rôzne parametre, ako je osa, zvolené hodnoty a cieľové elementy.
- **displayImageOfProjection**: Táto funkcia slúži na zobrazenie vykreslených histogramových projekcií. Vyžaduje identifikátor cieľového elementu a parametre pre zobrazenie.
- **openFile**: Táto funkcia slúži na otvorenie súboru na základe zadaného názvu. Vracia objekt reprezentujúci otvorený súbor.
- **cleanup** a **redraw**: Tieto funkcie slúžia na čistenie a aktualizáciu elementov po vykreslení projekcií.
- Komponent **THProjection**: Tento funkcionálny komponent je hlavným komponentom kódu. Využíva React hooky na správu stavu a vykonávanie vedľajších efektov. Komponent renderuje komponent **Sidebar** s obsahom, vrátane **NdmVrHistogramScene** a ďalších komponentov na základe stavu a poskytnutých dát. Kód tiež obsahuje niekoľko **useEffect** hookov na spracovanie vedľajších efektov na základe zmien v závislostiach a **useState** hooky na správu stavu komponentu. Táto funkcia je React komponent, ktorý zobrazuje 3D vizualizáciu histogramových projekcií. Komponent využíva rôzne stavy a efekty na správne načítanie a zobrazenie dát. Vstupné parametre komponentu zahŕňajú názov súboru (**filename**). Komponent obsahuje rôzne štýly a identifikátory, definuje reakciu na zmeny vstupných parametrov a vykonáva potrebné operácie, ako je otvorenie súboru, čítanie štítkov, čítanie objektov, získavanie údajov o projekciách, vykresľovanie projekcií a správu zmien.

**Jednotlivé funkcie komponentu THprojection:**

- **handleChange:** Funkcia, ktorá sa volá pri zmene výberu zdroja (**source**). Aktualizuje vybraný zdroj vo vnútri stavu komponentu **THProjection**.
- **handleThemeChange:** Funkcia, ktorá sa volá pri zmene výberu témy (**theme**). Aktualizuje vybranú tému vo vnútri stavu komponentu **THProjection**.
- **handleSceneChange:** Funkcia, ktorá sa volá pri zmene výberu scény (**scene**). Aktualizuje vybranú scénu vo vnútri stavu komponentu **THProjection**.
- **handleRangeChange:** Funkcia, ktorá sa volá pri zmene výberu rozsahu (**range**). Aktualizuje vybraný rozsah vo vnútri stavu komponentu **THProjection**.
- **handleChangeProj:** Funkcia, ktorá sa volá pri zmene výberu panelu projekcie (**projection panel**). Aktualizuje identifikátor kontajnera pre daný panel projekcie vo vnútri stavu komponentu **THProjection**.
- **handleChangeView:** Funkcia, ktorá sa volá pri zmene výberu zobrazenia (**view**). Aktualizuje vybrané zobrazenie vo vnútri stavu komponentu **THProjection**.
- **handleViewChange:** Funkcia, ktorá sa volá pri zmenách v zobrazení a vykonáva príslušnú logiku pre daný záznam. Vracia farbu na základe podmienok.
- **handleClick:** Funkcia, ktorá sa volá pri kliknutí na daný záznam. Aktualizuje vybraný záznam vo vnútri stavu komponentu **THProjection** a vykoná príslušné akcie s projekciami.
- **handleExecuteCommand:** Funkcia, ktorá sa volá pri vykonaní príkazu. Vykoná daný príkaz nad projekciami a aktualizuje stav komponentu **THProjection**.
- **renderHistogramProjections:** Funkcia, ktorá renderuje projekcie histogramov.
- **render:** Funkcia, ktorá renderuje komponent **THProjection**.

## Komponent <NdmVrHistogramScene>

### Funkcionalita:

NdmVrHistogramScene je React komponent, ktorý vytvára VR scénu s potrebnými komponentmi pre vizualizáciu histogramových dát. Tento komponent prijíma niekoľko vstupných parametrov, vrátane dát pre vizualizáciu, stavových premenných a úložísk pre rôzne funkcie. Celkovo **NdmVrHistogramScene** vytvára VR scénu s potrebnými komponentmi a spravuje interakciu a vizualizáciu histogramových dát v tomto prostredí.

### Použité funkcionality a komponenty v NdmVrHistogramScene:

- **NdmVrLaboratory**: Tento komponent predstavuje laboratórium virtuálnej reality, ktoré zabezpečuje správu a interakciu so scénami. Závisí od **states.scene**, ktorý je stavovou premennou a reprezentuje aktuálnu scénu.
- **NdmVrCamera**: Komponent, ktorý zabezpečuje kamery pre VR scénu.
- **NdmVrHistogram**: Komponent, ktorý zobrazuje histogramy na VR scéne.
- **DesktopTablet**: Komponent, ktorý sa zobrazuje len v prípade, že nie je pripojený headset pre VR zariadenia.

Okrem týchto komponentov, **NdmVrHistogramScene** tiež obsahuje niekoľko ďalších funkcionalít a vlastností:

- Využíva kontext **StoreContext**, ktorý poskytuje úložisko pre dáta, stavy a úložiská pre rôzne funkcie.
- Využíva **binDataDistributor** na distribúciu udalostí a dát z komponentu. Tento distribútor spracováva prichádzajúce dáta a vykonáva príslušné akcie na základe typu udalosti (kliknutie, dvojkliknutie, hover).
- Využíva hook **useEffect** na sledovanie zmien vstupných parametrov a na zaregistrovanie odberu dát z **binDataDistributor**.
- Používa komponenty **a-scene** a **a-assets** z knižnice A-Frame, ktoré poskytujú základnú infraštruktúru pre VR scénu.

### Implementácia zmeny scény:

- **StoreContext.Provider** slúži na poskytnutie hodnôt z kontextu **StoreContext** pre vnorené komponenty. V **value** sú definované rôzne vlastnosti (**data**, **stores**, **states**), ktoré sú prenesené ako hodnoty kontextu. **data** obsahuje identifikačné a údajové informácie o histogramoch. **stores** obsahuje úložiská pre rôzne aspekty vizualizácie. **states** obsahuje stavové premenné, vrátane témy, rozsahu, scény a ďalších.
- **<NdmVrLaboratory>** je komponent, ktorý zabezpečuje samotnú laboratóriu pre vizualizáciu histogramu. Ako vlastnosť **scene** prijíma hodnotu z **states.scene**, ktorá definuje aktuálnu scénu. Tento komponent je neskôr zodpovedný za spracovanie a zobrazenie samotných vizualizácií na základe poskytnutých dát a stavu.

### Komponent **<NdmVrHistogram>**

#### Funkcionalita:

React komponent s názvom **NdmVrHistogram** je zodpovedný za vytvorenie histogramu v rámci prostredia VR. Tento komponent sa skladá z niekoľkých častí kódu:

#### Importy:

Komponent importuje potrebné závislosti, ako React, niektoré vlastné súbory a knižnicu **'aframe'**, ktorá poskytuje možnosti pre vytvorenie 3D scény v prehliadači.

#### Vstup:

Komponent je definovaný ako funkcia, ktorá neprijíma žiadne argumenty.

#### Výstup:

Komponent vracia komponenty z knižnice **'aframe'**, ktoré tvoria 3D scénu histogramu.

#### Opis komponentu a jeho súčastí:

- **Hlavné telo komponentu:** Komponent používa React hooky ako **useState** a **useEffect** na správu stavu a životného cyklu komponentu.

- **Získanie hodnôt z kontextu:** Komponent využíva **useContext** hook na získanie hodnôt zo **StoreContext**. Tieto hodnoty sa priradzujú do rôznych premenných, ako je **selectedView**, **onViewChange**, **onProgress**, **section**, **range**, **theme** a **selectedBins**.
- **Vytvorenie inštancie HistogramReactFactory:** Komponent vytvára inštančiu triedy **HistogramReactFactory** pomocou **useState** hooku a poskytuje jej potrebné vstupné hodnoty, ako je **id**, **histogram**, **section**, **range**, **theme**, **selectedBins**, **onViewChange** a **onProgress**.
- **Správa stavu komponentu:** Komponent využíva rôzne **useEffect** hooky na aktualizáciu inštancie **HistogramReactFactory** v reakcii na zmeny vstupných hodnôt. Napríklad, ak sa zmení **histogram**, **section**, **range**, **theme**, **selectedBins**, **onViewChange** alebo **selectedView**, tak sa aktualizujú prvky histogramu.
- **Odoberanie odberu:** Komponent využíva **useEffect** hook na odber odberu z rôznych služieb (napríklad **histogramTH1Service**, **histogramTH2Service**, **histogramTH3Service**). Tento odber sa inicializuje po vykreslení histogramu a vytvorení inštancie **HistogramReactFactory**.
- **Vykresľovanie histogramu:** Ak sú dostupné prvky histogramu (**elements.bins**), komponent vykresľuje jednotlivé prvky pomocou značky **<a-box>** a **<a-text>**. Tieto prvky majú rôzne atribúty, ako je **id**, **key**, **class**, **material**, **binth**, **geometry**, **animation** a ďalšie. Komponent tiež vykresľuje ďalšie prvky, ako sú popisky, názvy osí a plochy pod histogramom.
- **Vykresľovanie náhradného obsahu:** Ak nie sú dostupné prvky histogramu (**elements.bins**), komponent vykresľuje jednoduchý náhradný obsah v podobe **<a-box>** elementu, ktorý sa animuje.