

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

**Virtuálne výučbové prostredie pre
prírodovedu**

Bakalárska práca

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

**Virtuálne výučbové prostredie pre
prírodovedu**

Bakalárska práca

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

Košice 2026

Milan Halčín

Abstrakt v slovenčine

Táto bakalárska práca sa zaoberá návrhom a implementáciou scenárom riadeného výučbového prostredia vo virtuálnej realite, určeného pre žiakov štvrtého ročníka základnej školy v oblasti prírodovedy. Práca rozširuje existujúci systém LirkisEduVePn vyvíjaný na Katedre počítačov a informatiky Technickej univerzity v Košiciach o nové interaktívne scény a scenáre.

Na základe systematického prehľadu literatúry a analýzy existujúcich riešení boli navrhnuté dve výučbové scény: magnetická scéna, v ktorej žiaci skúmajú vlastnosti magnetizmu prostredníctvom triedenia predmetov, a vesmírna scéna, v ktorej skladajú slnečnú sústavu a súhvezdia. Priebeh oboch scén je riadený scenárom vo forme Petriho siete. Prostredie je realizované pomocou webových technológií A-Frame, Three.js a fyzikálnej knižnice Ammo.js, čo umožňuje jeho spustenie na VR headsete, počítači aj mobilnom zariadení.

Riešenie bolo overené používateľským testovaním so žiakmi štvrtého ročníka a učiteľkami v Základnej škole Gemerská Hôrka, ktoré kombinovalo štandardizované dotazníky (SUS, NASA-TLX, SSQ, IPQ) s objektívnymi údajmi zaznamenanými systémom. Výsledky preukázali dobrú použiteľnosť a vysokú atraktivitu prostredia pre cieľovú skupinu, ako aj jeho potenciál pre praktické nasadenie vo výučbe prírodovedy.

Kľúčové slová v slovenčine

virtuálna realita, prírodoveda, základná škola, imerzívna výučba, Petriho sieť, scenárom riadená výučba, slnečná sústava, magnetizmus

Abstrakt v angličtine

This bachelor thesis deals with the design and implementation of a scenario-driven educational environment in virtual reality, intended for fourth-grade primary school pupils in the field of science. The thesis extends the existing LirkisEduVePn system, developed at the Department of Computers and Informatics of the Technical University of Košice, with new interactive scenes and scenarios.

Based on a systematic literature review and analysis of existing solutions, two educational scenes were designed: a magnetic scene, in which pupils explore the properties of magnetism by sorting objects, and a space scene, in which they assemble the solar system and constellations. The flow of both scenes is controlled by a scenario in the form of a Petri net. The environment is built using the web

technologies A-Frame, Three.js, and the Ammo.js physics library, allowing it to run on a VR headset, a computer, or a mobile device.

The solution was verified through user testing with fourth-grade pupils and teachers at the Gemerská Hôrka Primary School, combining standardized questionnaires (SUS, NASA-TLX, SSQ, IPQ) with objective data recorded by the system. The results demonstrated good usability and high attractiveness of the environment for the target group, as well as its potential for practical deployment in science education.

Kľúčové slová v angličtine

virtual reality, science education, primary school, immersive learning, Petri net, scenario-driven learning, solar system, magnetism

Bibliografická citácia

HALČIN, Milan. *Virtuálne výučbové prostredie pre prírodovedu*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2026. 91 s. Vedúci práce: doc. 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:

Virtuálne výučbové prostredie pre prírodovedu
Virtual Educational Environment for Natural Science

Študent: **Milan Halčín**
Školiteľ: **doc. Ing. Štefan Korečko, PhD.**
Školiace pracovisko: **Katedra počítačov a informatiky**
Konzultant práce:
Pracovisko konzultanta:

Pokyny na vypracovanie bakalárskej práce:

1. Oboznámiť sa so systémom virtuálnych výučbových prostredí so scenármi vo forme Petriho sietí, vytvorenom na domovskom pracovisku.
2. Analyzovať podobné existujúce riešenia a relevantné požiadavky z pedagogickej praxe.
3. Na základe analýzy navrhnúť a implementovať výučbové prostredie zamerané na vybrané oblasti prírodovedy, vrátane scenárov vo forme Petriho sietí.
4. Návrhové a implementačné práce koordinovať s ďalšími riešiteľmi prác na systéme virtuálnych výučbových prostredí.
5. Vypracovať dokumentáciu podľa pokynov vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský
Termín pre odovzdanie práce: 29.05.2026
Dátum zadania bakalárskej práce: 31.10.2025




prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som záverečnú prácu vypracoval samostatne s použitím uvedenej odbornej literatúry, ako aj s využitím nástrojov generatívnej umelej inteligencie v súlade s Metodickým pokynom o záverečných a kvalifikačných prácach a Etickým kódexom študenta Technickej univerzity v Košiciach. Umelá inteligencia bola použitá ako podporný nástroj bez porušenia zásad akademickej etiky a autorskej integrity.

Podakovanie

Na tomto mieste by som sa rád poďakoval svojmu vedúcemu práce doc. Ing. Štefanovi Korečkovi, PhD., za jeho čas, odborné vedenie a pravidelné konzultácie počas riešenia mojej záverečnej práce. Rovnako by som sa rád poďakoval svojim rodičom a priateľom za ich podporu a povzbudzovanie počas celého môjho štúdia. Poďakovanie patrí aj Mykolovi Martyshchukovi za spoluprácu a Bc. Michalovi Sakáčovi za odborné usmernenie pri zaúčaní sa do systému. V neposlednom rade ďakujem pedagogickému zboru Základnej školy Gemerská Hôrka za umožnenie testovania a za ústretovosť pri jeho organizácii, ako aj žiakom, ktorí sa testovania zúčastnili.

Obsah

1	Úvod	1
2	Analýza súčasného stavu	3
2.1	Technická stránka projektu	3
2.2	Aktuálny ponúkaný obsah systému	4
2.2.1	Obsah z pohľadu žiaka	4
2.2.2	Obsah z pohľadu správcu a učiteľa	4
2.3	Predchádzajúce práce	4
2.3.1	Diplomová práca Adama Kašelu	5
2.3.2	Bakalárska práca Lukáša Pisarčíka	5
2.3.3	Bakalárska práca Dmytra Demianenka	6
2.3.4	Bakalárska práca Michala Sakáča	6
2.3.5	Diplomová práca Kristiána Zsigóa	7
3	Analýza a rozbor existujúcich riešení a systematický prehľad literatúry	8
3.1	Formulácia výskumných otázok	8
3.2	Metodika vyhľadávania relevantných výskumov pre túto prácu	9
3.3	Analýza výsledkov dopytov	12
3.4	Vplyv imerzie virtuálneho prostredia na výsledné pochopenie a zapamätanie si preberaného učiva	14
3.5	Rôzne prístupy tvorby výučbových prostredí	14
3.6	Dôležitosť a typy spätnej väzby pri tvorbe výučbových prostredí vo VR	16
3.7	Vplyv predchádzajúcich znalostí študentov na efektívnosť výučby vo virtuálnej realite	16
3.8	Zvedavosť ako silný motivátor pre získanie nových znalostí	18
3.9	Zhrnutie kapitoly	19
4	Návrh používateľského rozhrania	21
4.1	Používateľská analýza a konceptuálny model prostredia	21

4.2	Návrh postupností scenárov	22
4.3	Správna voľba a práca s 3D modelmi	24
4.4	Prototypy scén	24
4.5	Vyhodnotenie prvého prototypu	27
4.5.1	Používateľská spätná väzba prvého prototypu	27
4.5.2	Zmeny vykonané na základe spätnej väzby	28
4.5.3	Rozšírenie scenárov nad rámec prototypov	31
4.6	Riadenie scenárov Petriho sieťami	33
4.6.1	Petriho sieť magnetickej scény	33
4.6.2	Petriho sieť vesmírnej scény	35
5	Implementácia	39
5.1	Podporné komponenty	42
5.1.1	Komponent <i>ammo-init</i>	42
5.1.2	Komponent <i>grabbable-by-parenting</i>	44
5.2	Komponenty magnetickej scény	45
5.2.1	Komponent <i>magnet-box</i>	45
5.2.2	Komponent <i>random-pool-picker</i>	48
5.3	Komponenty vesmírnej scény	51
5.3.1	Komponent <i>orbit-on-collision</i>	51
5.3.2	Komponent <i>proximity-sensor</i>	54
5.3.3	Komponent <i>activate-object-on-event</i>	58
6	Vyhodnotenie	62
6.1	Metodika hodnotenia	62
6.2	Testovacie prostredie VR aplikácie	63
6.3	Respondenti a priebeh testovania	63
6.4	Použitý dotazník	65
6.4.1	Použitelnosť systému	66
6.4.2	Kognitívna a fyzická záťaž	67
6.4.3	Fyzické symptómy pri používaní VR	67
6.4.4	Pocit prítomnosti vo virtuálnom svete	67
6.5	Vyhodnotenie výsledkov dotazníka	68
6.5.1	Demografické údaje respondentov	68
6.5.2	Použitelnosť — System Usability Scale (SUS)	68
6.5.3	Kognitívna záťaž — NASA-TLX	70
6.5.4	Symptómy kybernetickej choroby (SSQ / VRSQ)	72
6.5.5	Pocit prítomnosti vo virtuálnom svete (IPQ)	72

6.5.6	Hlavné nálezy z dotazníkov	74
6.6	Vyhodnotenie zaznamenaných údajov	75
6.6.1	Spracovanie dát	75
6.6.2	Súbor dát	76
6.6.3	Úspešné dokončenie scén	76
6.6.4	Magnetická scéna — triedenie predmetov	77
6.6.5	Vesmírna scéna — umiestňovanie planét	78
6.6.6	Vesmírna scéna — skladanie súhvezdia	79
6.6.7	Využitie nápoved	80
6.6.8	Trvanie relácií	80
6.6.9	Súhrn nálezov zo zaznamenaných údajov	81
6.7	Diskusia výsledkov	82
6.8	Zhrnutie vyhodnotenia	82
7	Záver	85
	Zoznam použitej literatúry	88
	Zoznam príloh	92
A	Použité dotazníky	93
B	Používateľská príručka	105
B.1	Funkcia programu	105
B.2	Inštalácia programu	105
B.2.1	Požiadavky na technické prostriedky	105
B.2.2	Režimy spustenia	106
B.2.3	Spustenie projektu lokálne	107
B.2.4	Vytvorenie administrátorského účtu	109
B.2.5	Import scenára a vytvorenie úlohy	110
B.2.6	Vytvorenie skupiny a priradenie žiakov	111
B.2.7	Produkčné nasadenie na serveri	111
B.3	Použitie programu	112
B.3.1	Prihlasovacia obrazovka	112
B.3.2	Hlavná obrazovka	113
B.3.3	Profil žiaka a spustenie úlohy	113
B.3.4	Ovládanie virtuálnej scény	113
B.3.5	Magnetická scéna	114
B.3.6	Vesmírna scéna	115

B.3.7	Ukončenie a opakovanie úlohy	116
C	Systémová príručka	117
C.1	Funkcia programu	117
C.2	Popis programu	117
C.2.1	Popis riešenia	117
C.2.2	Štruktúra prostredia pre prírodovedu	118
C.3	Popis implementovaných súborov	118
C.4	Popis implementovaných A-Frame komponentov	118
C.5	Využité technológie	120
C.6	Spustenie programu	120
C.7	Nasadenie na server	121
C.7.1	Aktualizácia aplikácie na serveri	121
C.7.2	Konfigurácia servera	122
C.7.3	Konfigurácia nginx	122

Zoznam obrázkov

3.1	Venov diagram výsledkov jednotlivých databáz	12
3.2	Výsledky nájdených dokumentov podľa roku publikácie	13
3.3	Výsledky nájdených dokumentov podľa krajiny pôvodu	13
4.1	Konceptuálny model domény vo forme UML diagramu tried	22
4.2	Prvotný návrh postupností scenárov	23
4.3	Prototyp prostredia s úlohami triedenia magnetických materiálov . . .	25
4.4	Pôvodný model asteroidu pred modifikáciou	25
4.5	Pôvodný model asteroidu po modifikácii	26
4.6	Prototyp prostredia s úlohami slnečnej sústavy a súhvezdí	26
4.7	Vyhodnotenie jednotlivých otázok dotazníka SUS	28
4.8	Neviditeľné bariéry zabraňujúce odletu predmetov zo scény	29
4.9	Informačná tabuľa s tutoriálom ovládania scény	30
4.10	Petriho sieť riadiaca scenár magnetickej scény	34
4.11	Petriho sieť vesmírnej scény	36
5.1	Štruktúra komponentov použitých pre scény a scenáre projektu	40
5.2	Štruktúra modelov, modulov, scenárov a scén projektu	41
5.3	Viditeľný magnetický kváder	46
5.4	Planéta pohybujúca sa po dráhe	54
5.5	Replikácia súhvezdia pomocou 3D tvarov	58
B.1	Dashboard administrátora s možnosťami importu scenára, registrácie scény a vytvorenia úlohy	110
B.2	Prihlasovacia obrazovka aplikácie	112
B.3	Profil žiaka s priradenými úlohami a tlačidlom <i>Start</i>	113
B.4	Informačný panel s prehľadom ovládacích prvkov v scéne	114
B.5	Magnetická scéna – triedenie predmetov pomocou magnetu	115
B.6	Vesmírna scéna – umiestňovanie planét na obežné dráhy	116
B.7	Vesmírna scéna – skladanie súhvezdia podľa náповedy	116

Zoznam tabuliek

3.1	Výskumné otázky vzťahujúce sa k analýze	9
3.2	Dopyty vzťahujúce sa k analýze dát	11
4.1	Tabuľka autorov použitých 3D modelov	24
4.2	Zmeny vykonané v prostredí ako reakcia na pripomienky testerov . .	31
6.1	Demografické zloženie respondentov	68
6.2	SUS skóre žiakov podľa respondenta	69
6.3	SUS skóre učiteliek podľa respondenta	69
6.4	Kognitívna záťaž žiakov (NASA-TLX, škála 1–3, 1 = najhoršia, 3 = najlepšia hodnota)	70
6.5	NASA-TLX skóre učiteliek podľa respondenta (0–100)	71
6.6	Priemerné hodnoty dimenzií NASA-TLX u učiteliek	71
6.7	Symptómy kybernetickej choroby u žiakov (SSQ, počet áno z 10) . . .	72
6.8	Symptómy učiteliek (VRSQ, priemer intenzity 0–3)	73
6.9	IPQ otázky u žiakov (po normalizácii, vyššie = vyšší pocit prítomnosti) .	73
6.10	IPQ u učiteliek — dimenzie (po normalizácii, škála 1–7)	74
6.11	Úspešné dokončenie scén podľa zaznamenaných údajov	77
6.12	Úspešnosť umiestnení v magnetickej scéne podľa respondenta	77
6.13	Úspešnosť umiestnení planét podľa respondenta	78
6.14	Účasť na úlohe skladania súhvezdia	80
6.15	Trvanie relácií podľa scény a skupiny (v minútach)	81
C.1	Prehľad implementovaných A-Frame komponentov	119

Zoznam výpisov

B.1	Konfigurácia súboru <code>docker-compose.yml</code>	107
B.2	Konfigurácia URL pre prístup cez lokálnu sieť	108
B.3	Spustenie projektu s prestavbou obrazov	108
B.4	Spustenie bez prestavby obrazov	109
B.5	Zastavenie projektu	109
B.6	SQL dotaz na povýšenie účtu na administrátora	110
B.7	Štruktúra priečinka scenára vesmírnej scény	110
B.8	Označenie a nahratie Docker obrazu	111
B.9	Spustenie na serveri v režime na pozadí	112
C.1	Spustenie projektu pre lokálny vývoj	120
C.2	Spustenie bez prestavby obrazov	121
C.3	Spustenie na pozadí	121
C.4	Prihlásenie do Docker Hub	121
C.5	Označenie obrazov tagmi	121
C.6	Nahratie obrazov na Docker Hub	122
C.7	Aktualizácia aplikácie na serveri	122
C.8	Konfigurácia <code>docker-compose.yml</code> na produkčnom serveri	122
C.9	Aktivácia konfigurácie <code>nginx</code>	123

1 Úvod

Predmetná práca vychádza zo systému LirkisEduVePn a rozvíja ho. Systém bol vyvíjaný našou katedrou v rámci projektu KEGA č. 048 TUKE-4/2022, ktorý sa zameriaval na vývoj technológií zdieľanej virtuálnej reality vo vzdelávacom procese. Hlavným oporným bodom realizácie boli predchádzajúce práce študentov, ktoré významne prispeli k vývoju doterajšieho projektu.

Motiváciou pre realizáciu tejto práce bola dlhodobá komunikácia so základnou školou v obci Gemerská Hôrka. Konzultácie s touto školou viedli k dohode o vytvorení výučbového prostredia pre žiakov prvého stupňa základnej školy štvrtého ročníka v oblasti neživá príroda a prírodné javy. Základná škola Gemerská Hôrka sa vyznačuje najmä malými triedami, spájanými ročníkmi a vysokým dôrazom na individuálny prístup k žiakom. Z toho dôvodu je vhodným prostredím pre aplikáciu inovatívnych pedagogických metód. Predpokladáme, že implementácia výučbového prostredia do regiónov, v ktorých takáto technológia vo výchovno-vzdelávacom procese ešte nie je integrovaná, prinesie zaujímavé výsledky a cennú spätnú väzbu pre ďalší vývoj.

Prínosom tejto práce je zefektívnenie výučby ľahko prístupnou a voľne dostupnou metódou, ktorá zabezpečí lepšie pochopenie preberaného učiva imerzívnou formou. To zahŕňa zjednodušenú vizualizáciu, plnenie edukatívnych úloh, možnosť aktívneho sa zapájania v zmysle experimentov a pohyb v priestoroch, ktoré sú pre bežného človeka ťažko dostupné alebo úplne nedostupné. Pri tvorbe nášho vzdelávacieho prostredia bolo dôležitým faktorom plnenie jasne stanovených pedagogických cieľov, technické spracovanie a opieranie sa o predchádzajúce skúsenosti našich používateľov s virtuálnou realitou.

Formulácia úlohy

Výsledným produktom tejto práce má byť rozšírenie existujúcej vzdelávacej platformy LirkisEduVePn, a to v podobe nového edukačného prostredia pre žiakov štvrtého ročníka základnej školy. Úlohy zahrnuté vo výučbovom prostredí majú pokrývať časť vzdelávacieho programu „Človek a príroda“ pre prvý stupeň základnej školy štvrtého ročníka, konkrétne tematický celok „Neživá príroda a prírodné javy“. Podobne aj ako ostatné, aktuálne prítomné vzdelávacie prostredia LirkisEduVePn, pre vývoj a zobrazenie scény má byť použitý softvérový rámec A-Frame a scenár úloh definovaný procesným grafom Petriho sieť. Hlavnou úlohou je ponúknuť žiakom alternatívny spôsob výučby ľahko dostupnou formou vo webovej virtuálnej realite. Najdôležitejším obsahovým prvkom je umožnenie interaktívnej manipulácie s predmetmi, ktoré majú edukatívny charakter a s ktorými v bežných podmienkach nie je možné voľne experimentovať. Tvorba výučbového prostredia zo širšej perspektívy bude zahŕňať:

1. pochopenie pedagogických cieľov,
2. navrhnutie interaktívnych úloh,
3. zaradenie priebežnej multimodálnej spätnej väzby,
4. integrácia pedagogických prvkov, napr. automatizovaných testov alebo hodnotiacich úloh
5. nasadenie do vzdelávacieho procesu (ako výhľadový cieľ)

Z technického hľadiska pri návrhu bude potrebné:

1. Vytvorenie alebo použitie 3D modelov edukatívneho charakteru,
2. definovanie scenára na základe procesného grafu (Petriho siete),
3. zobrazenie prostredia a vytvorenie interakcií medzi 3D modelmi,
4. testovanie riešenia v laboratórnych a následne v reálnych podmienkach na konkrétnej škole.

2 Analýza súčasného stavu

Táto kapitola približuje aktuálny stav projektu LirkisEduVePn, ktorý táto práca rozširuje. Najprv sa venujeme technickej architektúre projektu a jeho systémovým súčastiam, následne opisujeme aktuálne ponúkaný obsah z pohľadu jednotlivých používateľských rolí a v závere kapitoly rozoberáme predchádzajúce práce, ktoré k projektu prispeli novými funkcionalitami alebo výučbovými modulmi.

2.1 Technická stránka projektu

LirkisEduVePn je nízkonákladové, voľne dostupné webové riešenie pre výučbu vo virtuálnej realite. Aplikácia pozostáva z backendovej a frontendovej časti, ktoré sú navzájom izolované v kontajneroch prostredia Docker.

Backendovú časť tvorí projekt Maven pre jazyk Java, postavený na softvérovom rámci Spring Boot. Na ukladanie údajov o používateľoch, scénach, scénároch a záznamoch z riešenia úloh slúži databáza PostgreSQL.

Frontendová časť je rozdelená podľa typu obsahu. Dvojrozmerné používateľské rozhranie je postavené na softvérovom rámci Angular, kým trojrozmerné virtuálne scény sú tvorené pomocou A-Frame — softvérového webového rámca s otvoreným zdrojovým kódom, ktorý je postavený nad knižnicou Three.js a na komunikáciu s VR zariadeniami využíva technológiu WebXR. Použitie rámca A-Frame zabezpečuje jednoduchú deklaratívnu HTML syntax pri tvorbe 3D scén.

Ovládanie výučbovej časti aplikácie je flexibilné — je možné pomocou VR okuliarov a ovládačov, klasicky klávesnicou a myšou, ale aj dotykovým ovládaním na smartfóne. Táto viacrežimovosť zvyšuje dostupnosť systému aj v prostrediach, kde nie je k dispozícii VR hardvér.

2.2 Aktuálny ponúkaný obsah systému

Základnou funkciou aplikácie je registrácia a prihlásenie používateľa, pričom systém rozlišuje tri používateľské role — správca, učiteľ a žiak. Rolu používateľovi prideluje správca systému.

V aktuálnom stave systém obsahuje tri použiteľné scény, z toho dve výučbového charakteru a jednu ukážkovú. Jednotlivé scenáre je možné ovládať tak vo virtuálnej realite pomocou VR okuliarov a ovládačov, ako aj konvenčne klávesnicou a myšou. Súčasťou jedného z výučbových scenárov sú aj kvízy zamerané na dejepisné témy.

2.2.1 Obsah z pohľadu žiaka

Žiak po prihlásení spustí jedno z jemu priradených výučbových prostredí a plní v ňom interaktívne úlohy vo virtuálnej realite. Po ukončení relácie si môže zobrazíť históriu svojich riešených scenárov a úloh. Ako doplnkové overenie nadobudnutých znalostí si môže vyskúšať aj niektoré z ponúkaných kvízov.

2.2.2 Obsah z pohľadu správcu a učiteľa

Role učiteľa a správcu dedia všetky funkcie žiaka a navzájom sú si funkčne veľmi podobné — obe pracujú prevažne v sekcii ovládacieho panela. Učiteľ dokáže prideliť úlohu skupine žiakov spojením scenára a scény. Žiakovi sa daná úloha sprístupní v prípade, že je súčasťou skupiny, ktorej bola úloha priradená.

2.3 Predchádzajúce práce

Projekt LirkisEduVePn vznikol v roku 2022 v rámci projektu KEGA č. 048 TUKE-4/2022 a od svojho založenia bol rozvíjaný viacerými študentmi katedry, ktorí ho obohatili buď o nové funkcionality, alebo o nové výučbové moduly. Súčasná podoba systému — s backendom, dvojrozmerným rozhraním pre správu úloh a viacerými výučbovými modulmi — je výsledkom postupného nadväzovania týchto prác. V nasledujúcich podkapitolách rozoberáme príspevky tých autorov, ktorých práca priamo súvisí so systémom LirkisEduVePn ako celkom alebo s tvorbou jeho vzdelávacích modulov, a to chronologicky podľa roku odovzdania.

2.3.1 Diplomová práca Adama Kašelu

Základy projektu LirkisEduVePn položil v roku 2022 Adam Kašela vo svojej diplomovej práci [1], ktorá skúmala možnosti výučbového prostredia riadeného procesným grafom — Petriho sieťou. Analýza literatúry v tejto práci ukázala, že použitie Petriho sietí na riadenie postupnosti úloh je vhodnou a v oblasti tvorby scenárov zaužívanou voľbou. Na základe týchto poznatkov vzniklo pilotné výučbové prostredie, ktorého obsah sa zameriaval na tematický celok predmetu dejepis pre žiakov siedmeho ročníka základnej školy, konkrétne na obdobie habsburskej monarchie.

Prostredie malo podobu múzejnej expozície tvorenej z troch miestností, ktoré obsahovali pätnásť exponátov a tri úlohy. Exponáty bolo možné uchopiť a stlačením tlačidla zobrazíť o nich bližšie informácie. Úlohy zahŕňali kvízové otázky s výberom správnej odpovede alebo presun dobových exponátov na určené miesto. Zobrazovanie scény po technickej stránke zabezpečoval softvérový rámec A-Frame, rovnako ako v tejto práci.

Vyhodnotenie práce prebehlo formou prieskumu použiteľnosti, na ktorom sa zúčastnilo dvadsaťdeväť respondentov, z ktorých osemnásť patrilo do cieľovej skupiny. Z výsledkov prieskumu vyplynul jednoznačný záujem žiakov o interaktívnejšiu formu výučby, pričom respondenti prejavili záujem aj o rozšírenie tohto spôsobu výučby na ďalšie predmety, predovšetkým prírodovedného zamerania. Tento záver je jedným z motivačných východísk aj pre túto prácu, ktorá sa zameriava práve na prírodovedný obsah.

2.3.2 Bakalárska práca Lukáša Pisarčíka

Na Kašelovu diplomovú prácu nadviazal v roku 2023 Lukáš Pisarčík vo svojej bakalárskej práci [2], ktorá sa zamerala na zovšeobecnenie pôvodného riešenia tak, aby bolo použiteľné pre tvorbu ľubovoľných výučbových scenárov, nielen pre konkrétnu múzejnú expozíciu. Súčasťou analýzy bol systematický prehľad literatúry venovaný použitiu Petriho sietí pri tvorbe scenárov virtuálnych prostredí, ktorého výsledky potvrdili vhodnosť tohto prístupu pre vzdelávacie aplikácie.

Hlavným prínosom tejto práce bolo oddelenie definície scenára od konkrétnej scény — Petriho sieť a znenia úloh sú v zavedenom riešení uložené v samostatných súboroch, ktoré sa do scény načítavajú za behu. Tým vzniká univerzálny systém riadenia scén, ktorý je použiteľný pre akýkoľvek výučbový scenár bez nutnosti úprav zdrojového kódu samotnej scény. Pisarčík súčasne navrhol a implementoval mechanizmus zaznamenávania aktivity používateľa, ktorý zachytá-

va interakcie žiaka v scéne a odosiela ich na server pre ďalšie spracovanie. Práca priniesla aj rozšírené ovládanie scény pomocou raycastera pre VR okuliare a podporu viacjazyčnosti scénových popiskov. Tento univerzálny mechanizmus riadenia scén Petriho sieťami je naďalej kostrou systému LirkisEduVePn a využíva ho aj výučbové prostredie navrhnuté v tejto práci.

2.3.3 Bakalárska práca Dmytra Demianenka

Na záznamy aktivity zavedené Pisarčíkom nadviazal v roku 2023 Dmytro Demianenko vo svojej bakalárskej práci [3], ktorá sa venovala spracovaniu týchto údajov a vybudovaniu administratívnej časti systému. Práca priniesla backendovú časť pre zber a analýzu údajov o plnení úloh, pričom zachytáva relevantné údajové body ako čas riešenia, presnosť a jednotlivé interakcie používateľa. Tieto údaje slúžia jednak učiteľovi na sledovanie pokroku žiakov, jednak žiakovi samotnému na získanie spätnej väzby o jeho výkone.

Súčasťou práce je aj dvojrozmerné používateľské rozhranie pre správu systému, ktoré poskytuje rôzne funkcie podľa role používateľa — žiakovi umožňuje spúšťať priradené úlohy a prehliadať si históriu vlastných riešení, učiteľovi prideľovať úlohy skupinám žiakov a sledovať ich výsledky, a správcovi vykonávať administratívne zásahy v systéme. Tieto súčasti, spolu s Petriho sieťami pre riadenie scenárov od Pisarčíka, tvoria základ aktuálnej podoby systému LirkisEduVePn, ako ju opisujú podkapitoly 2.1 a 2.2.

2.3.4 Bakalárska práca Michala Sakáča

Michal Sakáč sa vo svojej bakalárskej práci [4] zaoberal virtuálnym výučbovým prostredím pre skladanie počítačov, určeným pre žiakov strednej školy ako doplnková aktivita k výučbe informatiky. Analýza literatúry v jeho práci zachytila viacero relevantných aplikácií z oblasti pedagogiky aj herného priemyslu, na základe ktorých vznikol výučbový modul pre skladanie počítačov.

Scéna sa odohrávala v kancelárii, ktorá bola pre zlepšenie výkonu systému značne zjednodušená — odstránené boli graficky náročné objekty a tie vypálené do textúr. Výsledná miestnosť obsahovala návod na použitie VR ovládačov a stoly s komponentmi určenými na stavbu počítačov. Úlohy zahŕňali stavbu počítača a výmenu jednotlivých komponentov v rôznych úrovniach náročnosti. Komponent bolo možné nasadiť na správne miesto jeho presunutím na preddefinovaný bod, kde sa objekt automaticky pripojil a natočil do správnej polohy. Manipulácia s objektmi a ich kolízie boli implementované pomocou fyzikálnej

knižnice Ammo.js. Keďže automaticky generované kolízne obaly zložitých modelov boli výpočtovo príliš náročné a znefunkčňovali scénu, autor pristúpil k ručne vytvoreným kolíznym obalom v tvare kvádra.

Výučbové prostredie testovalo dvadsaťosem ľudí, z toho pätnásť použilo VR okuliare. Prieskum fyzickej a psychickej záťaže naznačil, že virtuálne prostredie nebolo pre dobrovoľníkov namáhavé. Pre túto prácu je podstatné, že komponent *attachment-box*, ktorý Michal Sakáč pôvodne vytvoril vo svojej bakalárskej práci [4], v rámci svojej diplomovej práce zovšeobecnil pre použitie vo viacerých scénach, vďaka čomu ho bolo možné integrovať aj do nášho výučbového prostredia (podkap. 4.5.2).

2.3.5 Diplomová práca Kristiána Zsigóa

Kristián Zsigó sa vo svojej diplomovej práci [5] venoval rozšíreniam systému a tvorbe ukážkovej scény, ktorá má slúžiť ako návod pre nových vývojárov projektu. Na základe odporúčaní z odbornej literatúry doplnil systém o sadu nových komponentov a zároveň opravil a vylepšil viaceré existujúce.

Medzi jeho prínosy patrí komponent *autofit-gltf*, ktorý vytvára presnú kolíznú mapu 3D modelov vo formáte GLTF a GLB pre fyzikálny engine Ammo.js. Zsigó tiež upravil komponent *blink-controls* tak, aby znemožnil teleportáciu cez steny, a nahradil zastaraný fyzikálny engine CANON.js za výkonnejší Ammo.js. Pri prehrávaní animácií 3D modelov využil komponent *animation-mixer* zo štandardnej knižnice *aframe-extras*.

Nové komponenty Zsigó demonštroval v ukážkovej scéne, ktorá nemá edukatívny charakter — jej cieľom je ukázať možnosti interakcie vo virtuálnom prostredí. Scéna má podobu upravenej garáže umiestnenej v lesnom prostredí a pozostáva z troch miestností oddelených dverami, ktoré sa otvárajú tlačidlom, pákou alebo zadáním hesla na virtuálnej klávesnici. Jednotlivé miestnosti obsahujú úlohy zábavného charakteru — hádzanie lopty do koša, usporiadanie predmetov podľa farebného spektra a triedenie predmetov podľa farby a tvaru. Prieskum realizovaný v tejto práci bol zameraný predovšetkým na posúdenie toho, do akej miery môže byť ukážková scéna a jej dokumentácia nápomocná pri tvorbe budúcich vzdelávacích modulov.

3 Analýza a rozbor existujúcich riešení a systematický prehľad literatúry

Tvorba správneho výučbového prostredia zahŕňa zložitý a časovo náročný proces. Preto je dôležité zvoliť správne kroky a prvky pri tvorbe takéhoto prostredia. V tejto kapitole sa pozrieme na pohľad rôznych štúdií z hľadiska tvorby korektného a efektívneho virtuálneho výučbového prostredia po obsahovej stránke. Následne bližšie nahliadneme na niektoré existujúce riešenia zameriavajúce sa na tému neživá príroda a prírodné javy. Pre správnu demonštráciu prírodných javov je dôležitá možnosť aktívnej interakcie používateľa s predmetmi a interakcií predmetov medzi sebou vo forme experimentov resp. edukatívnych úloh. Preto podrobne rozoberieme správny výber edukatívnych prvkov a spätnej väzby pri interakcii používateľa s prostredím. Konkrétnu predmetnú otázku zobrazuje tabuľka 3.1.

3.1 Formulácia výskumných otázok

Výskumná otázka sa zameriava na identifikáciu najefektívnejších postupov pri tvorbe výučbových prostredí vo virtuálnej realite, a to z hľadiska obsahového dizajnu, výberu prvkov používateľského rozhrania a foriem spätnej väzby pri interakcii používateľa s objektmi a prostredím. Pri tejto otázke sa kladie dôraz najmä na kognitívnu záťaž študenta a efektivitu osvojovania si poznatkov.

Otázka	Motivácia
O1: Aké postupy treba zvoliť pri tvorbe výučbových prostredí, a čo by malo obsahovať?	M1: Zistiť, aké kľúčové zásady musia byť zohľadnené pri tvorbe efektívneho výučbového prostredia.

Tabuľka 3.1: Výskumné otázky vzťahujúce sa k analýze

3.2 Metodika vyhľadávania relevantných výskumov pre túto prácu

Po úspešnej formulácii potrebných otázok bolo nutné pre zber dát vyhľadanie relevantných odborných prác. Na získanie potrebných informácií boli použité databázové systémy Web of Science a Scopus.

1. *WebOfScience*¹ - platená platforma s prístupom k viacerým bibliografickým a citačným databázam, ktorá poskytuje údaje z rôznych dokumentov. Slúži na sledovanie vedeckého výskumu v zvolenom obore.
2. *Scopus*² - multidisciplinárna abstraktová a citačná databáza s komplexným pokrytím vedeckej, technickej, lekárskej a spoločenskovednej literatúry.

Výsledný dopyt pre vyhľadávanie relevantných zdrojov zachytáva tabuľka 3.2. Pri zadávaní dopytu bolo kľúčové filtrovanie potenciálne nerelevantných výskumov a článkov. Z tohto dôvodu bol do vyhľadávacieho reťazca zakomponovaný logický operátor NOT (resp. AND NOT), ktorého cieľom bolo eliminovať metodologický šum a vyradiť štúdie nespádajúce do presne definovaného rámca tejto práce:

- **Výučba jazykov** („*english teaching*“, „*language learning*“, „*english language teaching*“): Vylúčená preto, lebo jazykové vzdelávanie vo VR sa zameriava na úplne iné typy imerzie a interakcie (napr. konverzačné simulácie) než prírodovedné témy.
- **Špecifické tréningy, medicína a psychológia** („*training*“, „*trauma*“, „*disability*“): Tieto pojmy odkazovali na priemyselné tréningy, chirurgické simulácie či liečbu psychických porúch a traum, ktoré majú diametrálne odlišné pedagogické ciele ako edukácia žiakov primárneho vzdelávania.

¹Definícia WebOfScience: <https://www.elsevier.support/publishing/answer/what-is-web-of-science-and-how-does-it-work>

²Definícia Scopus: <https://www.elsevier.com/products/scopus/content>

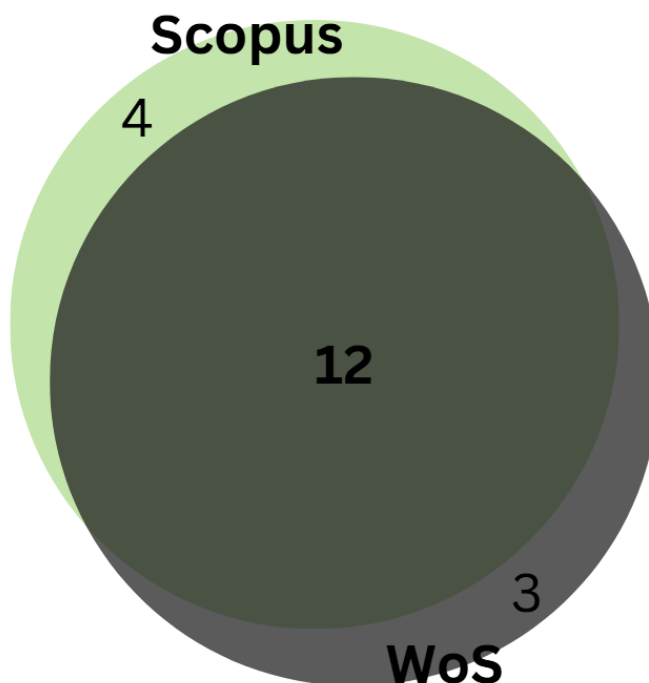
- **Počítačové vedy** („*computer science*“): Vylúčenie článkov zameraných na čistý softvérový vývoj, grafické renderovanie, sieťovú infraštruktúru či optimalizáciu algoritmov virtuálnej reality.
- **Inžinierske vedy a robotika** („*engineering*“, „*robot**“): Vyradenie technicky orientovaných štúdií riešiacich konštrukciu hardvéru, VR okuliarov a robotických systémov, keďže cieľom bolo skúmať didaktický dopad hotového prostredia na žiaka, nie inžiniersky proces vývoja.
- **Biologické a nepríbuzné pojmy** („**sex**“, „*maturity*“): Slúžili ako poistka pred štúdiami zaoberajúcimi sa biologickou dospelosťou, zrelosťou či medicínskym výskumom v oblasti sexuality, ktoré prenikali do výsledkov na základe textovej zhody kľúčových slov.

Databáza	Dopyt
Web of Science	TS=((("virtual reality" OR "VR") AND("immersive learning" OR "immersive environment*" OR "immersive virtual reality") AND ("learning environment*" OR "educational environment*" OR "instructional environment" OR "virtual field trip" OR "learning tasks" OR "education content") AND ("immersive design" OR "instructional design" OR "curriculum design" OR "content design" OR "learning design" OR "science education" OR "elementary school" OR "types of interaction" OR "practices and strategies" OR "instructional media") AND ("learning content" OR "learning objectives" OR "learning outcomes" OR "pedagogical principles" OR "immersion principle" OR "key aspects" OR "educational priorities" OR "feedback" OR "educational practices")) NOT TS=("english teaching" OR "english language teaching" OR "language learning" OR "training" OR "trauma" OR "disability" OR "computer science" OR "engineering" OR "robot*" OR "*sex*" OR "maturity") AND PY=(2012-2026) AND DT=(Article OR Review)
Scopus	TITLE-ABS-KEY(("virtual reality" OR VR) AND ("immersive learning" OR "immersive environment*" OR "immersive virtual reality") AND ("learning environment*" OR "educational environment*" OR "instructional environment" OR "virtual field trip" OR "learning tasks" OR "education content") AND ("immersive design" OR "instructional design" OR "curriculum design" OR "content design" OR "learning design" OR "science education" OR "elementary school" OR "types of interaction" OR "practices and strategies" OR "instructional media") AND ("learning content" OR "learning objectives" OR "learning outcomes" OR "pedagogical principles" OR "immersion principle" OR "key aspects" OR "educational priorities" OR "feedback" OR "educational practices") AND NOT("english teaching" OR "english language teaching" OR "language learning" OR "training" OR "language curricula" OR "trauma" OR "disability" OR "computer science" OR "engineering" OR "robot*" OR "sex*" OR "maturity")) AND PUBYEAR > 2011 AND PUBYEAR < 2027 AND (DOCTYPE(ar) OR DOCTYPE(re))

Tabuľka 3.2: Dopyty vzťahujúce sa k analýze dát

3.3 Analýza výsledkov dopytov

Každý z dopytov bol priebežne niekoľkokrát na základe relevancie upravovaný. Po úspešnej úprave dopytov jednotlivých databáz a ich zadaní sme získali 15 relevantných zdrojov z databázy *WebOfScience*³ a 16 zo *Scopus*⁴.



Obrázok 3.1: Venov diagram výsledkov jednotlivých databáz

Z celkových 31 výsledkov 12 štúdií bolo duplicitných a 7 unikátnych pre jednotlivé databázy a to: 4 rôzne výsledky pre Scopus a 3 pre WebOfScience (viď obrázok 3.1).

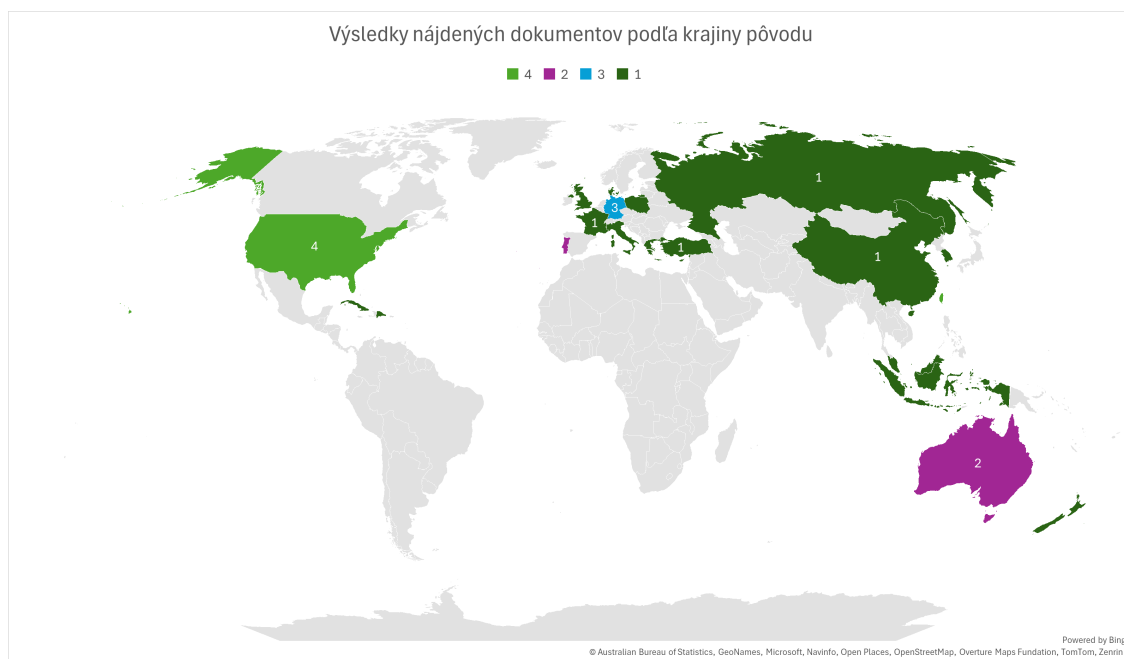
³Definícia WebOfScience: <https://www.elsevier.support/publishing/answer/what-is-web-of-science-and-how-does-it-work>

⁴Definícia Scopus: <https://www.elsevier.com/products/scopus/content>



Obrázok 3.2: Výsledky nájdených dokumentov podľa roku publikácie

Graf zobrazený na obrázku 3.2 zobrazuje nájdené štúdie podľa roku publikácie. Na základe grafu môžeme tvrdiť, že pre našu tému bolo vytvorených 19 relevantných štúdií v rozmedzí rokov 2012 až 2025. Taktiež môžeme deklarovat, že najviac štúdií, ktoré sa zaoberajú našim problémom bolo publikovaných v rokoch 2022 a 2023.



Obrázok 3.3: Výsledky nájdených dokumentov podľa krajiny pôvodu

Obrázok 3.3 ukazuje, že krajiny, ktoré sa najviac zaoberajú tematikou zadanej

v dopyte 3.2 sú USA, menej viditeľný Taiwan a Nemecko. Nižšie zastúpenie iných krajín môže súvisieť s rôznou mierou rozšírenia VR technológií vo vzdelávaní, no naše dáta neumožňujú túto príčinu overiť.

3.4 Vplyv imerzie virtuálneho prostredia na výsledné pochopenie a zapamätanie si preberaného učiva

Dôležitosť imerzie vo virtuálnom prostredí je predmetom mnohých štúdií, ktoré skúmajú do akej miery ponorenosť do prostredia, a či vôbec vplýva na efektivitu výučby. Prieskumy štúdie [6] preukázali, že zvýšenie miery imerzie má pozitívny dopad na zapamätanie si poznatkov či už z krátkodobého alebo dlhodobého hľadiska. Testovanie prebiehalo vo forme virtuálneho výletu najprv s použitím HMD headsetu, pričom testovania sa zúčastnilo 49 náhodne vybraných stredoškôľakov a 53 stredoškôľakov na podobnom prostredí, ale v 2D forme. Témy výučby zahŕňali klimatické zmeny v prostredí. Študenti absolvovali tri testy, a to pred exkurziou, ihneď po exkurzii a následne tri týždne po exkurzii. Výsledky štúdie taktiež zaznamenali významné prepojenie medzi ponorenosťou do prostredia, zvedavosťou a následným dlhodobým zapamätaním si poznatkov. Podobná štúdia zameriavajúca sa na imerziu výučbových prostredí a kognitívnu záťaž študentov [7] na základe výskumov ale zaznamenala, že hoci zvýšenie pocitu prítomnosti výrazne emocionálne angažuje študentov do plnení úloh, môže aj zvýšiť kognitívnu záťaž študenta plniaceho úlohu. Zvýšená mentálna zataženosť môže negatívne vplývať na študentovu schopnosť zapamätania si preberanej látky. Testovanie prebiehalo v dvoch skupinách rovnakého počtu náhodne vybraných študentov. Jedna skupina ($n=42$) plnila úlohy imerzívnou 3D formou a druhá neimerzívnym hypertextovým prostredím. Následne každý zo študentov vykonal test ihneď po dokončení úloh. Je nutné podotknúť, že imerzívne prostredie bolo testované vo forme serióznej hry a nie VR.

3.5 Rôzne prístupy tvorby výučbových prostredí

Pre tvorbu korektného virtuálneho výučbového prostredia je potrebné zohľadniť všetky možné postupy. Štúdia [8] sa zaoberá práve touto problematikou. Cieľom práce bolo vytvoriť systematickú analýzu prístupov pre tvorbu virtuálnych edukatívnych prostredí. Práca analyzovala 219 empirických štúdií z rôznych oblastí

vzdelávania a kurzov. Pri výskume sa identifikovalo deväť dizajnových prvkov súvisiacich s učením ako zvuk, pohyb tela, haptika a pod. VR prístupy boli rozdelené do 5 klastrov:

1. pasívne VR vo forme exkurzií s limitovanou zapojenosťou žiaka do výučby,
2. interaktívne VR, ktoré aktívne zapája žiakov do výučby,
3. role-play, ktorý umožňuje používateľovi vžiť sa do určitej roly,
4. realistické simulácie používané napr. v zdravotníctve,
5. kooperatívne (sociálne) VR založené na rovesníckej spolupráci pri plnení úloh.

Autor štúdie na základe vypracovaného rešeršu dospel k záveru, že správna voľba prístupu k tvorbe virtuálnych edukatívnych prostredí záleží na kontexte a pedagogických cieľoch. Dôležitá je ale kombinácia prvkov interakcie, kontextu a primeranej spätnej väzby. Širší kontext pri používaní IVR nám ponúka aj článok [9], kde autori porovnávajú až 47 štúdií. Pretože všetky predchádzajúce výskumy sa zameriavali na vzdelávacie výsledky, ovplyvňované VR, cieľom tejto štúdie bolo mapovať praktiky a stratégie využívané IVR. Autori identifikovali 45 stratégií a 21 praktík. Nedávna skúsenosť s celosvetovou pandémiou Covid-19 nás prinútila zaviesť dištančné technológie do pedagogickej praxe. Tradičné stratégie a technológie sa museli upraviť alebo nahradiť novými. Odborníci navrhujú zamyslieť sa nad tým, že technológie nemusia vždy fungovať zovšeobecnením, ale navrhujú skúmať kontext konkrétnych podmienok a skupín učiacich sa. Autori analyzovali stratégie, týkajúce sa procesu vyučovania a učenia, pričom vychádzali z Merrillových princípov výučby: študent musí byť zapojený do riešenia, reálnych problémov, aktivuje sa jeho predchádzajúce poznanie, keď je nové poznanie demonštrované, aplikované a integrované do jeho vlastného sveta. Hľadali sa odpovede na otázky: Aké vzdelávacie praktiky, stratégie sú uvádzané v prehľadových štúdiách a aký je základný vzťah medzi nimi. Použitá literatúra sa vyhľadávala uplatnením pojmov: virtuálna realita, rozšírená virtuálna realita a zmiešaná realita. Autori zistili, že väčšina článkov pochádza z obdobia po roku 2013. Následne vytvorili 2 skupiny. Identifikované praktiky boli tematicky zoskupené a vizualizované vo forme sieťového modelu. Zistil sa silný vzťah medzi praktikami a stratégiami, odhalili sa aj miesta s nedostatočným prepojením. Pomohol vytvoriť rámec pedagogických intervencií v imerzívnych vzdelávacích

prostrediach. Štúdia vyzdvihla, že ak chceme využívať IVR ako vhodný pedagogický nástroj vo výchovno-vzdelávacom procese, je nevyhnutné poznať pedagogickú metodológiu, formy, metódy, stratégie a praktiky, ktoré využíva pedagogická veda, aby pôsobenie na žiaka bolo efektívne. Je veľmi dôležité zostavovať vhodný obsah s príslušnou vizualizáciou, aby sme splnili cieľ výchovno-vzdelávacieho procesu.

3.6 Dôležitosť a typy spätnej väzby pri tvorbe výučbových prostredí vo VR

Pri interakcii s prostredím virtuálnej reality a jeho objektmi je otázka spätnej väzby predmetom rôznych štúdií. Práca [10] skúmala ako rôzne typy spätnej väzby pri manipulácii s 3D objektmi v imerzívnej virtuálnej realite ovplyvňujú študentov, konkrétne ich záujem, pochopenie, preferencie a zapamätanie učiva. Pre skúmanie boli vybrané tri typy spätnej väzby a to:

1. vizuálna — zmena rozmerov predmetu pri dotyku,
2. zvuková — vyslovenie názvu predmetu pri dotyku,
3. vizuálna — tvarová deformácia objektu pri dotyku.

Testovanie prebehlo vo forme interakcií s chemickými prvkami. Podobne ako v už spomenutých výskumoch boli študenti rozdelení do dvoch skupín. Jedna skupina zažívala učenie v 2D forme a druhá vo VR. Analýzu tvoril dotazník so sedembodovou Likertovou škálou a štatistické testy. Metriky testovania tvorili záujem, pochopenie, preferencia a zapamätanie. Výsledky analýzy testov a dotazníkov preukázali, že študenti celkovo vnímali VR zážitok lepšie ako 2D. Taktiež z výsledkov vyplýva, že napriek zvýšeným preferenciám študentov v oblasti VR, konečný výsledok zapamätania nebol rozdielny oproti 2D systému. Z hľadiska spätnej väzby sa najefektívnejšia forma javila vizuálna deformácia objektu.

3.7 Vplyv predchádzajúcich znalostí študentov na efektívnosť výučby vo virtuálnej realite

Mnoho štúdií sa už zaoberalo efektívnosťou imerzívneho vzdelávacieho prostredia virtuálnej reality ale predmet skúmania bol rôzny. Medzi inými môžeme

spomenutý výskum [11], zameraný na závislosť efektivity IVRLE od predchádzajúcich vedomostí študentov. Výskum sa uskutočnil na vzorke 14 študentov bakalárskeho stupňa 3. ročníka ošetrovateľstva v rozsahu simulácie štyrikrát počas jedného mesiaca. Komparačnou skupinou bolo skúmanie 57 študentov 1. ročníka ošetrovateľstva rovnakým IVRLE aplikujúc jednorázovo. Tematicky bolo IVRLE zamerané na gestovú interakciu na výučbu predoperačných chirurgických postupov a konceptov ako: rozprestretie mayo podložky, otvorenie chirurgického kontajnera a usporiadanie chirurgických nástrojov na mayo stole. Údaje z prvej fázy boli analyzované podľa neparametrických štatistických testov Friedman a Wilcoxon, resp. údaje druhej fázy pomocou jednovýberového testu. Pearsonov korelačný koeficient poslúžil na zistenie vzťahov medzi dvoma subškálami. Vieme, že získanie zručností v medicínskej oblasti si vyžaduje dlhodobé precvičovanie a viacnásobnú prax. Vzdelávanie VRLE prináša veľa pozitívneho pri získavaní týchto zručností a to: prostredníctvom aktívnej participácie, zážitkového učenia, konštruktívneho hodnotenia, podpory predstavivosti, bezpečného prostredia a poskytnutím realistických vzdelávacích scenárov. Štúdia vyzdvihuje najmä pocit prítomnosti — pocit „byť tam“. Zvyšuje sa motivácia a udržiava sa vyššia úroveň pozornosti. IVRLE je výborným prostriedkom lacnejšieho a opakovateľne použiteľného individuálneho vzdelávacieho prostredia. Výskum sa zameriaval na to, aký je vzťah medzi vnímaným učením, postojom a sebadôverou, ako aj na to, či je vhodnejší IVRLE pre začiatočníkov alebo mierne pokročilých študentov. Čo sa týka IVRLE, kládol sa dôraz na imerziu, prirodzené farby a transparentnosť menu. Študenti podobne ako v našom výučbovom systéme mohli spúšťať scenáre, ukážkové videá a hodnotiť svoj výkon. Pri splnení úloh bolo dôležité: lineárnosť a presné poradie ich plnenia. Zistenia boli nasledovné: z dát, nazbieraných pomocou dotazníka a videozáznamov sa zistilo, že z pohľadu zapojených a pozorovaných vnímali študenti 3. ročníka IVRLE ako efektívny nástroj. Študenti 1. ročníka to hodnotili nižšie. Študenti vyššieho ročníka zaznamenali nárast v oblasti sebadôvery po opakovaných použitíach. Empirické dôkazy hovoria o tom, že IVRLE môže posilniť sebadôveru skúsených študentov a u začiatočníkov existuje silný vzťah medzi postojom, učením a sebadôverou. Hoci sa štúdia zaoberá medicínskym vzdelávaním dospelých, jej zistenie o vzťahu medzi predznanosťami a efektivitou VR výučby je prenosné aj na našu cieľovú skupinu — poukazuje na to, že prostredie musí počítať s rôznou úrovňou vstupných vedomostí žiakov.

3.8 Zvedavosť ako silný motivátor pre získanie nových znalostí

Zaujímavou témou pozorovania v štúdiách o používaní IVR bol výskum [12], zameraný na jeho vplyv na učenie sa žiakov v Taiwane. Skúmal sa vplyv použitia pojmových máp ako typu predradeného organizátora na učenie prírodných vied prostredníctvom IVR. Hlavný dôraz sa kládol na úlohu epistemickej zvedavosti žiakov 6. ročníka základnej školy, kde sa na výskume zúčastnilo 74 šiestakov. Dáta sa zbierali formou dotazníkov a vedomostných testov o rastlinách. Výučbovým materiálom VR bola aplikácia "Find the ROOT". Skúmanou oblasťou bola zvedavosť, podmienená afektívnymi faktormi ako sú: pocit prítomnosti, kontrola a aktívne učenie, pozitívne a negatívne emocionálne zapojenie. Žiaci boli náhodne rozdelení do dvoch skupín: s predradeným organizátorom a bez predradeného organizátora (môže to byť napr. úvod, osnova, diagram, pojmová mapa, video). Vieme, že medzi hlavné vzdelávacie prínosy IVR patrí priestorová reprezentácia poznatkov, zážitkové učenie, podpora vnútornej motivácie a transfer poznatkov. Cieľom výskumu bolo zistiť, ako pôsobí predtréninová intervencia (v tomto prípade pojmové mapy) na efektivitu učenia. Ďalším cieľom bolo zistiť faktory, ktoré predikujú učenie (najmä zvedavosť — túžbu po poznaní). Vzdelávanie pozná 4 typy IVR, ktoré im umožňujú navštíviť inak nedostupné miesta a sledovať javy, ktoré v triede sledovať nemôžu:

1. CAVE systémy,
2. náhlavné displeje (HMD),
3. mobilná VR a
4. VR založená na sférickom videu (360°).

Analýza výsledkov dokázala, že žiaci, ktorí mali predradené organizátory dosahovali výrazne lepšie výsledky v porovnaní so žiakmi, ktorí predprípravu nemali. V oboch skupinách ale zvedavosť typu záujmu významne predikovala afektívne faktory. V skupine bez použitia pojmových máp pozitívne emocionálne zapojenie významne predikovalo vedecké poznanie. Zhrnutie výsledkov: predtrénin pomohol žiakom lepšie organizovať poznatky a znížiť kognitívne zaťaženie. Ukázalo sa, že vyšší pocit prítomnosti nie je v priamej úmere s lepšími učebnými výsledkami.

3.9 Zhrnutie kapitoly

Systematický prehľad literatúry priniesol niekoľko zistení, ktoré priamo formujú návrh nášho výučbového prostredia.

Imerzia ako dvojsečný nástroj. Z analyzovaných štúdií vyplýva, že imerzia sama o sebe nie je zárukou efektívnej výučby. Makransky a kol. [6] preukázal pozitívny dopad ponorenosti na dlhodobé zapamätanie poznatkov, no Schrader [7] a Lee a kol. [12] zhodne upozorňujú, že príliš vysoká miera imerzie zvyšuje kognitívnu záťaž a nemusí korelovať s lepšími výsledkami učenia. Pre náš projekt to znamená vedome obmedziť imerzívne prvky na úroveň potrebnú pre zapojenie žiaka, bez toho aby zaťažovali jeho pozornosť na úkor preberanej látky.

Kontext rozhoduje viac než technológia. Systematické prehľady Won [8] a Beck a kol. [9] ukazujú, že neexistuje jeden univerzálne najlepší prístup k tvorbe VR prostredia. Won identifikuje päť klastrov prístupov (pasívne, interaktívne, role-play, simulácie, kooperatívne) a zdôrazňuje, že voľba prístupu musí vychádzať z konkrétneho pedagogického cieľa a vekovej skupiny. Pre cieľovú skupinu žiakov 4. ročníka ZŠ a tematický celok Neživá príroda volíme *interaktívny prístup* s aktívnym zapojením žiaka do plnenia úloh.

Spätná väzba pri manipulácii s objektmi. Štúdia Shin a kol. [10] preukázala, že najefektívnejšou formou spätnej väzby pri interakcii s 3D objektmi je vizuálna deformácia objektu, doplnená auditívnym potvrdením. Toto zistenie priamo ovplyvnilo náš dizajn — v oboch scénach sme zvolili multimodálnu audio-vizuálnu spätnú väzbu, ktorá je v prostredí LirkisEduVePn už zavedeným riešením.

Vplyv predchádzajúcich znalostí. Výskum Tacgin [11] poukazuje na to, že efektivita imerzívneho výučbového prostredia silne závisí od úrovne predznalostí žiaka — skúsenejší žiaci ho hodnotili pozitívnejšie a dosahovali lepšie výsledky. Pre žiakov bez predznalostí je preto kľúčové zabezpečiť podporné mechanizmy, ktoré im umožnia úspešne dokončiť úlohu aj bez predchádzajúcich vedomostí o téme. V našom prostredí túto úlohu plnia voliteľné nápovedy, ktoré sa odhaľujú až na vyžiadanie žiakom.

Predtréning a zvedavosť. Lee a kol. [12] preukázal, že žiaci, ktorí pred VR aktivitou prešli krátkou predtréningovou prípravou vo forme pojmovej mapy, dosahovali výrazne lepšie výsledky než kontrolná skupina. Predtréning pomáha žia-

kom zorganizovať si poznatky a znížiť kognitívnu záťaž pri samotnej VR aktivite. Štúdia zároveň identifikovala zvedavosť ako silný prediktor učebných výsledkov, čo v kontexte našej cieľovej skupiny (žiaci 4. ročníka pri prvom kontakte s VR) predstavuje prirodzene prítomný motivačný faktor.

Zhrnutie pre návrh. Na základe týchto zistení formulujeme nasledovné princípy, ktoré budú riadiť návrh nášho výučbového prostredia:

- zvoliť interaktívny prístup s aktívnym zapojením žiaka, nie pasívnu exkurziu,
- obmedziť imerzívne prvky na úroveň potrebnú pre zapojenie bez nadmernej kognitívnej záťaže,
- poskytnúť multimodálnu audio-vizuálnu spätnú väzbu pri interakcii s objektmi,
- zaviesť voliteľný systém nápod, ktorý vyrovnáva rozdiely v predznalostiach žiakov,
- podporovať prirodzenú zvedavosť cieľovej skupiny hravými prvkami a možnosťou voľby v rámci úlohy.

Tieto princípy tvoria východisko pre návrh konkrétnych scenárov a používateľského rozhrania, ktorému sa venuje kapitola 4.

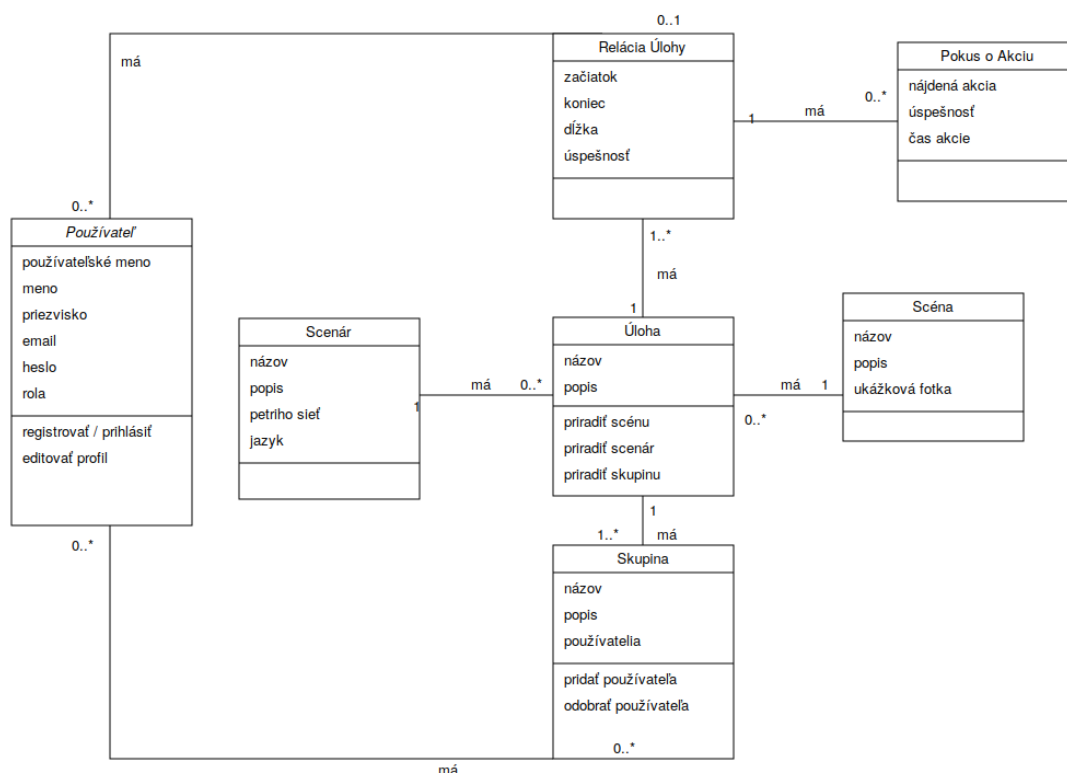
4 Návrh používateľského rozhrania

Vzhľadom na časovú náročnosť tvorby virtuálnych 3D prostredí, bolo používateľské rozhranie vytvárané v 2 iteráciách. Každá iterácia bola následne overená používateľským testovaním. Hlavným formálnym nástrojom na hodnotenie zážitku rozhrania bol štandardný dotazník SUS. Po úspešnej analýze v kapitole 3 sme prišli k záveru, že na základe odporúčaní štúdií sa po obsahovej stránke budeme zameriavať na vytvorenie bezpečného, inak nedostupného priestoru pre žiakov s logickým, lineárne usporiadaným priebehom úloh. Zistili sme taktiež, že imerzívnosť nie je kľúčovým prvkom pri memorizácii učiva a zvyšuje kognitívnu záťaž študenta, ale je do istej miery potrebná. Čo sa týka spätnej väzby sme sa rozhodli zvoliť multimodálnu spätnú väzbu v audio-vizuálnej forme, ktorá bola zaužívaná v už existujúcich riešení tohto projektu. Pred prototypovaním boli vyhotovené náčrty a prvý náhľad ako by mohlo prostredie vyzerieť. Prvý prototyp bol vytváraný priamo vo vývojovom rámci A-Frame a v kombinácii s TypeScriptom, pričom komponenty boli tvorené v JavaScripte. Vývojové prostredie a jeho nástroje boli pevne určené, keďže tento projekt je rozšírením už existujúceho výučbového prostredia vo VR LirkisEduVePn.

4.1 Používateľská analýza a konceptuálny model prostredia

Hlavnou úlohou pri tvorbe virtuálneho edukatívneho prostredia bolo splnenie pedagogických cieľov pre našu tému, zachovanie dostupnosti a kompatibility všetkých modulov systému. Kľúčovou oblasťou bolo zistenie ako korektne tvoriť takéhoto prostredia pre maximálnu efektivitu počas vyučovania. Dôležitá bola aj analýza predošlých rozšírení vývojárov tohto projektu, ale aj pohľad na niektoré komerčné alebo voľne dostupné výučbové systémy. Na základe úzkej spolupráce so Základnou školou Gemerská Hôrka nám bolo navrhnuté riadiť sa pri plnení

cieľov výučby štátnym vzdelávacím programom⁵. Súčasťou analýzy bol aj náhľad do UI prvkov, rôzne typy spätných väzieb pri manipulácii s objektmi a ich správny výber pre náš konkrétny scenár systematickým prehľadom dostupnej odbornej literatúry. Súčasťou výskumu UX bola identifikácia cieľovej domény a konštrukcia výstižnej definície úlohy. Následná voľba primárnej osoby zohráva kľúčovú úlohu pri správnom tvorení projektu ako celku. V našom prípade hlavný používateľ nášho výsledného projektu bude žiak štvrtého ročníka základnej školy. Obsahom nášho výučbového systému, na základe konzultácie s riadením Základnej školy Gemerská Hôrka, bola zvolená tematická oblasť Neživá príroda a prírodné javy. Po následnej analýze domény ako celku bol extrahovaný konceptuálny model, ktorý je zachytený na obrázku 4.1.



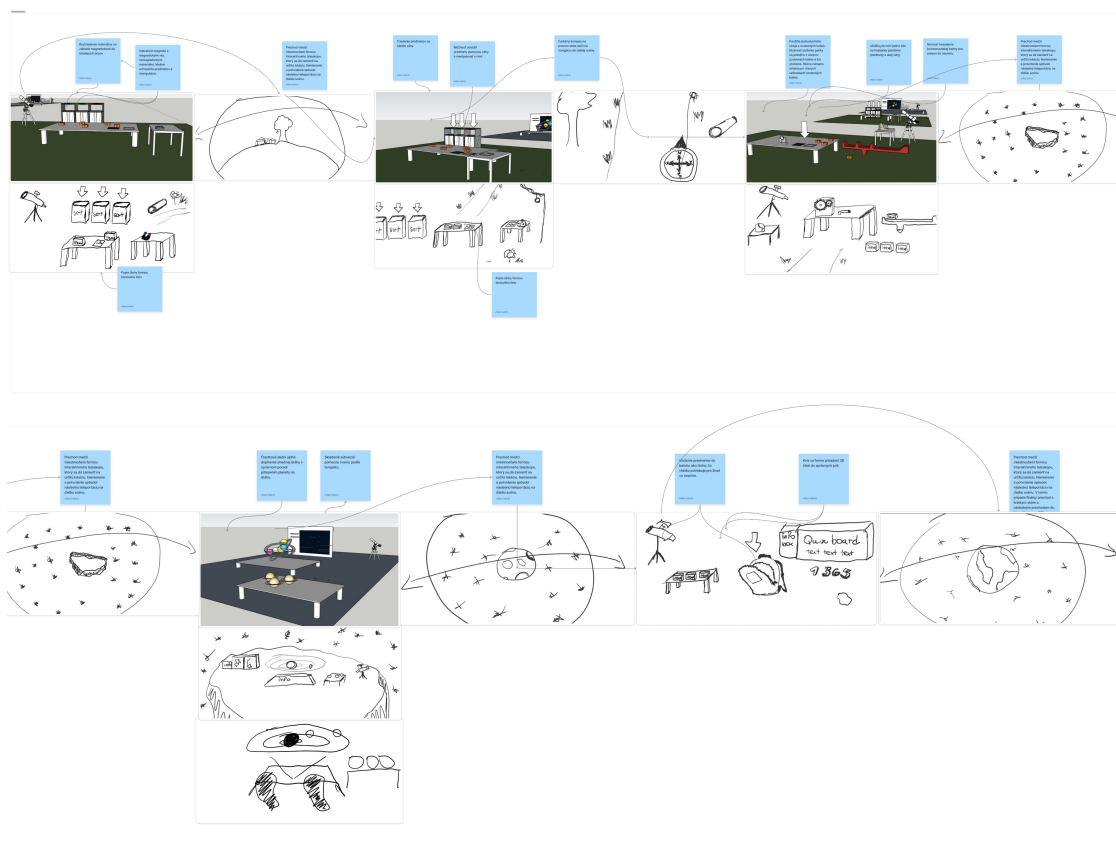
Obrázok 4.1: Konceptuálny model domény vo forme UML diagramu tried

4.2 Návrh postupností scenárov

Prvotnou myšlienkou pri vývoji bola tvorba výučbového prostredia v podobe príbehu, ktorý pokrýva jednotlivé tematické celky štátneho vzdelávacieho programu, konkrétne Neživá príroda a prírodné javy. Príbeh by bol rozdelený do

⁵Štátny vzdelávací program: <https://www.minedu.sk/data/att/dd4/22040.8e1235.pdf>

niekoľkých levelov, obsahujúcich edukatívne úlohy pre žiakov. Prechod medzi miestnosťami by bol možný po splnení niekoľkých podmienok pomocou teleskopu (teleportácie) alebo navigáciou s použitím kompasu. Pilotný nápad pozostával z 5 miestností. Postupnosti a prechod medzi miestnosťami znázorňuje obrázok 4.2. Prvá miestnosť bola zameraná na časť tématického celku magnetizmus. Úlohou žiaka by bolo zoradiť materiály na základe ich magnetickosti. Pre každý z dostupných predmetov by žiak dokázal pomocou magnetu zistiť priložením magnetu k predmetu, či je daný predmet magnetický. Druhá a tretia miestnosť by obsahovala úlohy zamerané na jednoduché stroje a práca s váhou predmetov ich vážením na váhe a presunom na hojdačku alebo páku. Miestnosť 4 tvorí vesmír a obsahoval by úlohu s interaktívnou slnečnou sústavou, konkrétne doplnenie planét do dráh a replikácia súhvezdí na základe vzorov pomocou tvarov. Finálna sekcia by sa odohrávala na mesiaci, kde žiak má za úlohu vybrať všetky predmety potrebné na prežitie vo vesmíre a vyplnenie interaktívneho kvízu, zameraného na fakty vesmíru správnym umiestnením 3D číslic.



Obrázok 4.2: Prvotný návrh postupností scenárov

4.3 Správna voľba a práca s 3D modelmi

Tvorba vo VR zahŕňa prácu s 3D modelmi. Náš výučbový systém musí ale zostať nízkorozpočtový a dostupný. Preto bola voľba správnych nízkopolygónových modelov kľúčová pre zachovanie výkonnosti nášho webového systému. Pre zložitosť a nedostatok skúseností s 3D modelovaním boli takmer všetky 3D modely projektu vybrané z webu Sketchfab, ktorý ponúka široké spektrum bezplatných 3D modelov, určených aj pre komerčné použitie. Každý z modelov bol vybrať a schválený po diskusii s riaditeľkou školy Základná škola Gemerská Hôrka. Zoznam modelov použitých v prototypoch scén znázorňuje tabuľka 4.1.

Autor	Zdroj
NorTEEN_X	Low-Poly Forest
m.altay	Crusader Great Helmet
IronEqual	Magnet from Poly by Google
JDAM	Gold Bar Low-Poly
dayoung.ko	Gallery Stand (White)
harindu fonseka	Low Poly Table
_its_sagar_bro	Low Poly Wooden Box
Binkley-Spacetrucker	Asteroid Diner
Dixept	Lego Bricks
DarrenMcnerney3D	Old Books
Aramuor	Officer / Police Captain / Pilot Hat
Mustafa97	Yatagan Sword (Ottoman)
swittzz	Eifel Munarasy
incg5764	Horseshoe Low Poly
incg5764	Heavy Rusty Anchor
dayoung.ko	Gallery Stand (Black)

Tabuľka 4.1: Tabuľka autorov použitých 3D modelov

4.4 Prototypy scén

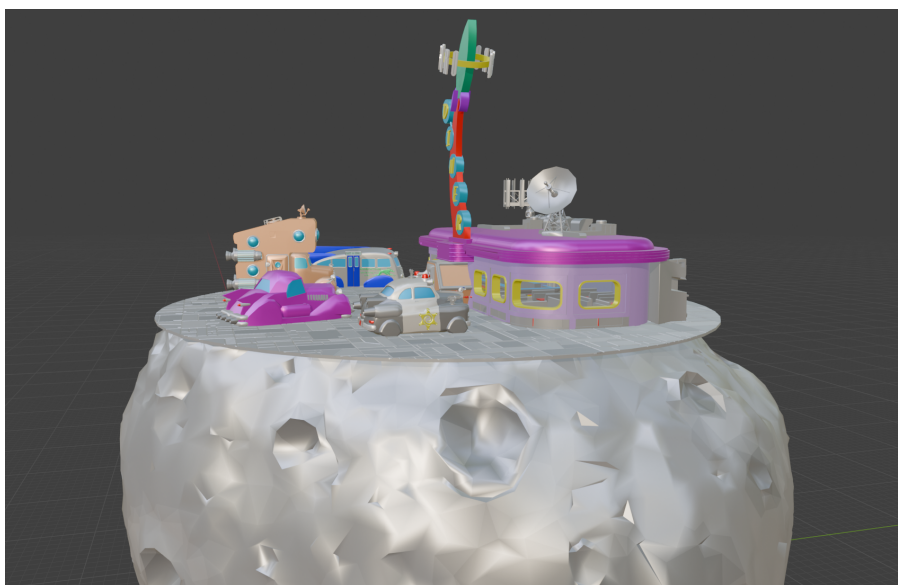
Prvý prototyp po diskusii so správou Základnej škola Gemerská Hôrka bol zameraný na miestnosti 1 a 4 z predošle vytvorenej postupnosti scenárov z podkapitoly 4.2, pretože sa tieto scenáre javili ako potenciálne najhojnejšie v smere spätnej väzby a postrehov vylepšenia výučbových praktík. Každý z výučbových scenárov potrebuje v prvom rade prostredie, v ktorom sa bude odohrávať. Pre miest-

nosť 1 bolo prostredie zvolené v podobe lesa a lúky znázornené na obrázku 4.3. Samotný 3D model lesa nebol modifikovaný, no bol externe obohatený pozadím lesa. Prototyp úlohy prvej miestnosti zahŕňal magnetizmus a používateľ dokázal vyskúšať magnet na dvoch predmetoch a zoradiť ich do jednej z dvoch krabíc. Inštrukcie úlohy boli zapísané na plátne. Po správnom zoradení predmetov bol informovaný používateľ o úspešnom ukončení úlohy vyskakovacím oknom.



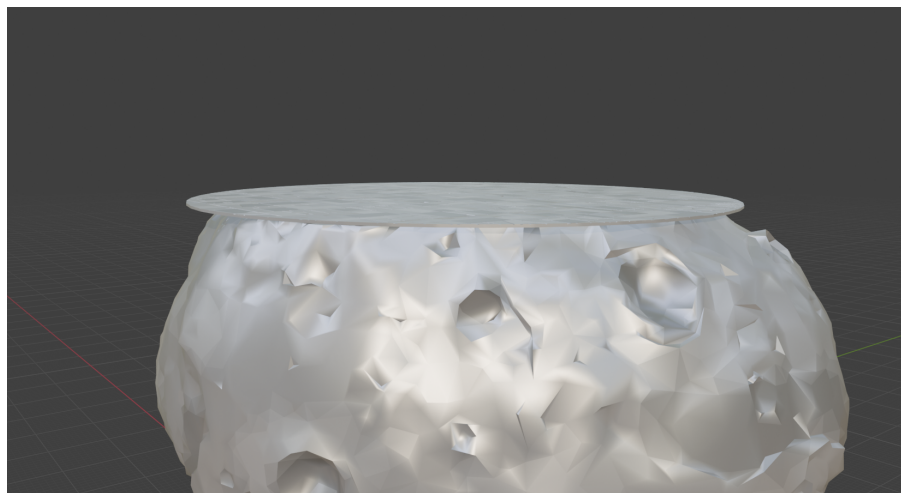
Obrázok 4.3: Prototyp prostredia s úlohami triedenia magnetických materiálov

Druhý prototyp scenára sa odohrával na 3D modeli asteroidu zachytávajúci obrázok 4.4, ktorý bol značne modifikovaný oproti pôvodnej verzii. Počiatočný model asteroidu obsahoval reštauráciu a rôzne modely áut, ktoré boli odstránené, aby ostal iba samotný asteroid a jeho plošina.



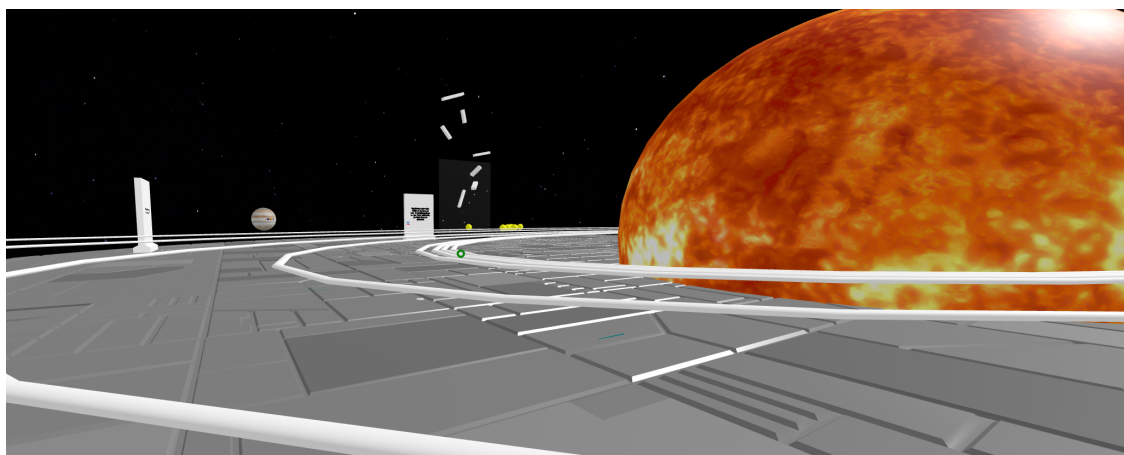
Obrázok 4.4: Pôvodný model asteroidu pred modifikáciou

Nakoľko bol 3D model asteroidu nelogicky rozčlenený do podmodelov, bolo nutné odstraňovať nepotrebné časti po jednotlivých polygónoch. Finálna verzia modifikovaného asteroidu je znázornená na obrázku 4.5.



Obrázok 4.5: Pôvodný model asteroidu po modifikácii

Tento level obsahoval 2 úlohy, ktoré zobrazuje obrázok 4.6. Jedna z týchto úloh bola interaktívna slnečná sústava, v ktorej bolo možné odoberať a pridávať planéty. Za úlohu bolo priradiť všetky planéty slnečnej sústavy na ich správne dráhy. Po umiestnení planéty do konkrétnej dráhy sa začala točiť okolo slnka a aj okolo vlastnej osi. Ďalšou úlohou bolo zostrojiť súhvezdie (v tomto deme Veľký voz) pomocou tvarov, rozhádzaných po asteroide, ktorý bol viditeľný na „oblohe“ ako vzor. Podobne ako pri scenári s magnetizmom, znenia úloh boli zobrazené na plátne formou textu.



Obrázok 4.6: Prototyp prostredia s úlohami slnečnej sústavy a súhvezdí

Na rozdiel od pôvodnej myšlienky vytvorenia prechodov medzi scenármi teleportáciou namierením teleskopu na určitú lokáciu, sme sa nakoniec po internej

diskusii rozhodli, že miestnosti budú tvoriť nezávislé scenáre. Obe zo scenárov bolo možné dokončiť vo VR aj desktopovom móde.

4.5 Vyhodnotenie prvého prototypu

Po úspešnom vytvorení oboch prototypových scenárov nasledovalo používateľské testovanie, ktoré malo za úlohu identifikovať potenciálne nedostatky výučbových scenárov. Testovania sa zúčastnilo 6 dospelých osôb, z toho 1 žena a 5 mužov. Prvá polovica subjektov testovala scenáre na desktopovej verzii a druhá polovica testovala prostredie vo VR na okuliároch Meta Quest 3S. 4 zo všetkých respondentov potvrdili predošlú skúsenosť s podobnými výučbovými scenármi. Každý z testovaných mal za úlohu vžiť sa do role žiaka 4. ročníka základnej školy, ktorému bolo učiteľom pridelených niekoľko úloh z predmetu prírodoveda v našej výučbovej aplikácii. Úlohy zahŕňajú interaktívnu manipuláciu s predmetmi vo VR alebo desktopovej verzii. Používatelia boli taktiež oboznámení s faktom, že nejde o plné verzie jednotlivých scenárov, nejde teda o finálny produkt.

4.5.1 Používateľská spätná väzba prvého prototypu

Každý z testerov najprv splnil scenár s magnetizmom a následne scenár vo vesmíre. Prvý scenár splnili úspešne všetci respondenti. Pri prvej úlohe vytkli tester najmä problém s navigáciou, nedostatok sprievodného textu alebo šípok. Kritizovali ostré farby, zlý úchyt predmetov, prepádávanie sa objektov pod zem, nemožnosť si predmety pritiahnuť k sebe. Privítali by systém inventáru, sprievodnú tabuľu s tutoriálom na ovládanie a jasné označenie krabíc pri triedení magnetických materiálov. Pri scenári vo vesmíre prekážali testerom niektoré už spomenuté chyby, ale vadila im aj fyzika spájania súhvezdia, umiestnenie slnečnej sústavy, nízka úroveň rýchlosti pohybu a považovali úlohy za pracné až ťažkopádne. Privítali by pridanie opisu objektov pri ich držaní, guľovú bariéru, ktorá zamedzí odlietavanie predmetov zo scény a možnosť použitia oboch rúk vo VR móde. Ani jeden z respondentov, ktorý plnil scenáre vo VR móde, nedokončil úspešne druhú úlohu so skladaním súhvezdí, čo môže indikovať vysokú náročnosť úlohy. Každý zo subjektov po ukončení úloh vyplnil štandardný 10-bodový dotazník *System Usability Scale (SUS)*⁶. Pričom po celkovom výpočte SUS skóre bolo 85,5, z čoho vyplýva nadpriemerné hodnotenie respondentov a naznačuje dobrú použiteľnosť a potenciál vyvíjaného softvéru. Výsledné vyhodnotenie jednotli-

⁶Doplňujúce info k metóde SUS: <https://measuringu.com/sus/>

vých otázok znázorňuje obrázok 4.7. Na otázku či by respondenti používali tento systém často, vyšiel priemer: 3,1. Či je systém zbytočne komplikovaný - bol hodnotený s priemerom 2,6. S priemerom odpovedí 3,3 skončila otázka, či sa systém používal ľahko. Na otázku, či by potrebovali respondenti technickú podporu, aby mohli systém používať, v priemere odpovedali hodnotou 1,8. Integrovanosť rôznych funkcií dostalo v priemere ohodnotenie 3,6. Ohodnotenie v priemere 1,5 dostala odpoveď, či obsahoval systém príliš veľa nezrovnalostí. 3,8 bol priemer pri odpovedi, ako rýchlo by sa dokázali používatelia systém ovládať. 2,3 bol priemer odpovede na otázku, či sa danému respondentovi ťažkopádne ovládal systém. Otázka "Cítil som sa sebavedomo pri používaní systému" skončila s priemerom 4,3, čo je veľmi pozitívne. Posledná otázka o náročnosti príprav, ktoré potrebovali respondenti pred absolvovaním ovládania systému skončila tiež pozitívnym ohodnotením s priemerom 1,5. Znamená to nenáročnosť príprav pred používaním systému. Väčšina odpovedí sa pohybovala okolo priemerného ohodnotenia. Dve posledné otázky boli naklonené skôr k pozitívnejšiemu názoru. Všetky spätné väzby, hodnotenia, pohľady, videozáznamy, výpočty a vyhodnotenia boli zachytené prostredníctvom Figjam projektu⁷.

Časová pečiatka	Myslím, že by som tento systém používal/a často.	Systém mi prišiel zbytočne komplikovaný.	Myslím, že sa tento systém používal ľahko.	Myslím, že by som potreboval/a technickú podporu aby som dokázal/a používať tento systém.	Myslím, že všetky rôzne funkcie systému boli dobre integrované.	Myslel/a som, že systém obsahoval príliš veľa nezrovnalostí.	Myslím, že by sa ľudia tento systém naučili používať veľmi rýchlo.	Prišlo mi veľmi ťažkopádne, používať tento systém.	Cítil/a som sa veľmi sebavedomo pri používaní tohto systému.	Potreboval/a som sa naučiť mnoho vecí pred používaním tohto systému.
2025/12/15 5:16:22 PM SEČ	2	3	4	3	2	1	2	3	4	2
2025/12/15 5:17:17 PM SEČ	3	1	4	1	4	2	5	1	5	1
2025/12/15 5:18:51 PM SEČ	5	3	3	3	5	1	5	1	5	1
2025/12/15 9:06:54 PM SEČ	3	4	2	1	4	1	3	4	4	1
2025/12/15 10:33:21 PM SEČ	2	1	4	1	3	3	3	4	5	1
2025/12/15 11:31:07 PM SEČ	4	4	3	2	4	1	5	1	3	3

Obrázok 4.7: Vyhodnotenie jednotlivých otázok dotazníka SUS

4.5.2 Zmeny vykonané na základe spätnej väzby

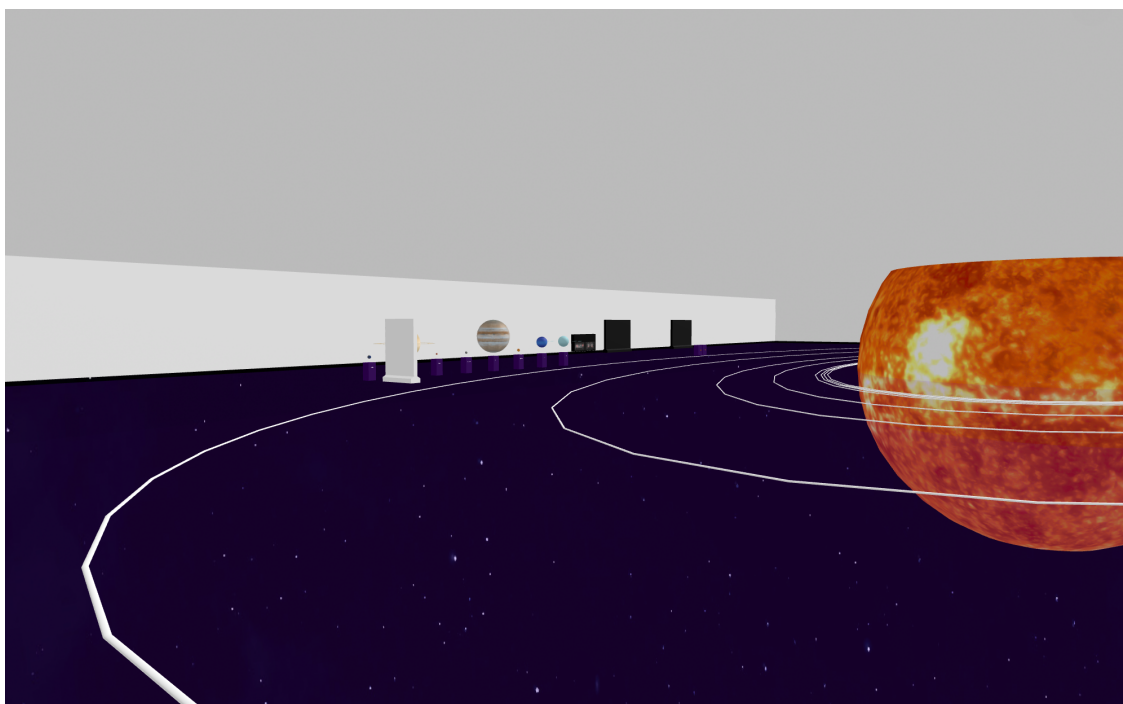
Identifikované nedostatky boli systematicky vyhodnotené a podľa ich závažnosti boli pred ďalšou fázou vývoja vykonané konkrétne dizajnové aj technické úpravy oboch scenárov. Niektoré zmeny boli implementované priamo v rámci tej-

⁷Figjam projekt: <https://www.figma.com/board/rwSv5TZtrFsr2CtxEyLvXE/UX-2-3-Ha%C4%8Din-Milan?node-id=253-2718&t=E8nWJCSWABghdxVz-1>

to práce, iné prostredníctvom komponentov vyvinutých v paralelne realizovanej diplomovej práci Michala Sakáča [4], pričom v oboch prípadoch ide o reakciu na konkrétne pripomienky testerov.

Na pripomienku problémov s navigáciou v prostredí scény bolo upravené počiatočné umiestnenie hráča – pôvodne vzdialenejšia pozícia bola posunutá bližšie k aktívnej oblasti úloh, čím sa znížil čas potrebný na zorientovanie sa v scéne a žiak je hneď po spustení v centre diania. Súvisiacu pripomienku k pomalému pohybu v prostredí sme riešili dvojstupňovo – úpravou mierky a veľkosti objektov v scéne, ako aj zvýšením rýchlosti pohybu hráča. Tým bola minimalizovaná potreba dlhých presunov medzi jednotlivými prvkami úlohy.

Problém s prepádávaním objektov pod terén bol riešený pridaním jednoduchých neviditeľných kolíznych kvádrov tvoriacich spodnú hranicu scény. Tieto kvádre slúžia ako fyzikálna bariéra, ktorá zachytáva všetky objekty bez ohľadu na ich tvar a tým zabraňuje strate predmetov mimo hrateľnej oblasti. Podobne bola na základe pripomienky o odlietavaní predmetov mimo scénu pridaná boxová bariéra ohraničujúca pracovný priestor scény, ktorú znázorňuje obrázok 4.8. Hoci respondenti pôvodne navrhovali guľovú bariéru, kvádrová alternatíva bola zvolená ako výpočtovo úspornejšia s rovnakým praktickým efektom v kontexte úloh s lokalizovanou pracovnou plochou.

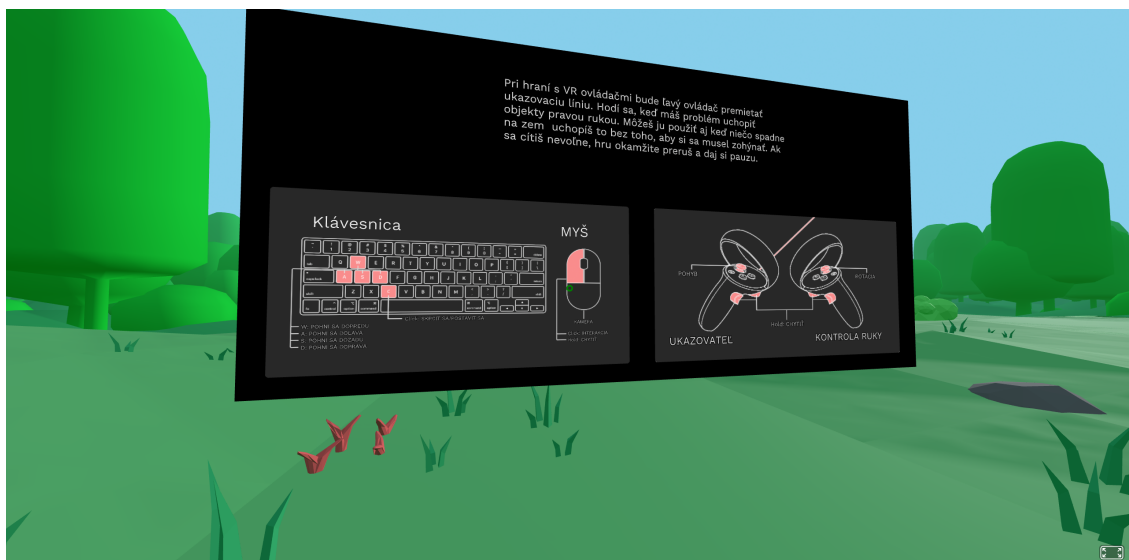


Obrázok 4.8: Neviditeľné bariéry zabráňujúce odletu predmetov zo scény

Možnosť pritiahnutia predmetu k sebe a celkové vylepšenie ovládania ucho-

povania bolo realizované prostredníctvom komponentu *grabbable-by-parenting* vyvinutého Michalom Sakáčom (popísaného v podkapitole 5.1.2), ktorý okrem priameho uchopenia umožňuje aj zmenu vzdialenosti uchopeného objektu od kamery pomocou kolieska myši. Týmto sa odstránil viacnásobne kritizovaný problém zlej manipulácie s predmetmi.

Pre lepšiu orientáciu používateľa v úlohách bola do scén zaradená sprievodná tabuľa s tutoriálom na ovládanie a navigáciu, ktorej pridanie inicioval prvotný prototyp tejto práce, no jej finálnu podobu zabezpečuje opätovne použiteľný prvok z práce Michala Sakáča [4]. Tabuľa viditeľná na obrázku 4.9 zobrazuje schému ovládacích prvkov pre desktopový aj VR režim a zostáva viditeľná počas celého priebehu scenára, čím poskytuje žiakovi referenciu k základným interakciám aj v prípade, ak si na ovládanie nespomenie v priebehu úlohy.



Obrázok 4.9: Informačná tabuľa s tutoriálom ovládania scény

V scéne lesa boli ďalej krabice na triedenie magnetických materiálov vizuálne odlíšené textovou, čo žiakovi okamžite signalizuje, kam má daný typ predmetov vkladať.

V scéne slnečnej sústavy bola fyzika spájania jednotlivých tvarov pri replikácii súhvezdia prepracovaná na základe komponentu *attachment-box* (rovnako z práce Sakáča [4]), ktorý zabezpečuje deterministické pripojenie tvaru na vopred určený bod bez závislosti na presnosti nárazu. Tým sa eliminoval problém, kedy sa jednotlivé tvary nepripájali korektne alebo sa od seba odrážali. Pre splnenie pripomienky k chýbajúcim opisom objektov boli planéty doplnené o samostatné podstavce s textovými popismi, ktoré poskytujú žiakovi informačnú spätnú väzbu o pomenovaní telesa.

Súhrn všetkých vykonaných zmien voči pôvodnému prototypu uvádza tabuľka 4.2.

Pripomienka testerov	Vykonaná zmena
Problém s navigáciou v prostredí	Posunutie počiatočnej pozície hráča bližšie k úlohám
Prepadávanie objektov pod zem	Pridanie neviditeľných kolíznych kvádrov tvoriacich spodnú bariéru
Nemožnosť pritiahnuť predmety k sebe	Použitie komponentu <i>grabbable-by-parenting</i> [4]
Chýbajúci tutoriál ovládania	Pridanie sprievodnej tabule s návodom
Nejasné označenie cieľových krabíc	Vizuálne odlíšenie krabíc farbou a piktogramami
Problémová fyzika spájania súhvezdia	Použitie komponentu <i>attachment-box</i> [4]
Nízka rýchlosť pohybu v scéne	Úprava mierky scény a zvýšenie rýchlosti pohybu hráča
Chýbajúce opisy objektov pri ich držaní	Pridanie planét na podstavce s textovými popismi
Odlietavanie predmetov mimo scény	Pridanie boxovej bariéry ohraničujúcej pracovný priestor

Tabuľka 4.2: Zmeny vykonané v prostredí ako reakcia na pripomienky testerov

4.5.3 Rozšírenie scenárov nad rámec prototypov

Po vyhodnotení prvého prototypu a vykonaní zmien reagujúcich na pripomienky testerov boli oba scenáre dodatočne upravené a rozšírené o prvky, ktoré neboli súčasťou pôvodného prototypového zadania. Cieľom týchto úprav bolo jednak doladiť vizuálnu stránku prostredia na základe konzultácie s vedúcim práce, a jednak obohatiť scenáre o pedagogicky a herne hodnotnejší obsah, ktorý lepšie reflektuje zámer štátneho vzdelávacieho programu pre tematický celok Neživá príroda a prírodné javy.

Úprava povrchu asteroidu.

Na základe konzultácie s vedúcim tejto práce bol povrch asteroidu, na ktorom sa odohráva scéna slnečnej sústavy, nahradený priehľadnou sklenenou plochou s jemným fialovým odtieňom. Pôvodná detailná textúra povrchu asteroidu pô-

sobila vizuálne rušivo a odpútavala pozornosť žiaka od samotných úloh. Nový povrch ponecháva dominantnou v scéne slnečnú sústavu a samotné súhvezdie, čím sa vizuálna hierarchia scény podriaďuje jej výučbovému obsahu. Priehľadnosť plochy zároveň poskytuje hráčovi otvorený výhľad do okolitého vesmíru, čo posilňuje dojem voľného priestoru a podporuje atmosféru scény.

Hľadanie magnetu pred triedením predmetov.

V pôvodnej verzii prototypu boli všetky predmety triedené žiakom rozmiestnené v scéne hneď od jej spustenia. Tento prístup však neviedol žiaka k aktívnemu preskúmaniu prostredia a celá úloha sa redukovala na mechanické triedenie viditeľných objektov. V upravenej verzii preto musí žiak najprv samotný magnet nájsť v scéne a až po jeho uchopení sa v prostredí naraz objavia všetky predmety určené na triedenie. Pre samotné hľadanie magnetu bola žiakovi pridaná navádzacia guľôčka inšpirovaná detskou hrou „zima – teplo“ – guľôčka mení svoju farbu plynulo medzi modrou a červenou v závislosti od vzdialenosti od ukrytého magnetu, pričom modrá farba reprezentuje vzdialenú pozíciu („zima“) a červená farba blízku pozíciu („teplo“). Tým je žiakovi v rámci úlohy ponúknutá hravá forma navigácie, ktorá nie je založená na priamom textovom pokyne ani na šípke, ale na intuitívnej priestorovej spätnej väzbe.

Toto rozšírenie je technicky realizované kombináciou komponentov *proximity-sensor* (popísaného v podkapitole 5.3.2) pre samotnú navigáciu k magnetu a komponentov *activate-object-on-event* (popísaného v podkapitole 5.3.3) a *random-pool-picker* (popísaného v podkapitole 5.2.2) pre objavenie predmetov po nájdení magnetu. Po nájdení magnetu sa vyšle udalosť, ktorá spustí náhodný výber predmetov z pripravenej zásoby a každý vybraný predmet sa následne aktivuje na vopred určenej pozícii. Týmto spôsobom žiak pri každom opakovaní úlohy nepracuje s rovnakou skupinou objektov, čo posilňuje jeho aktívne uvažovanie nad fyzikálnou vlastnosťou predmetov namiesto memorovania konkrétneho riešenia.

Voľba súhvezdia v scéne slnečnej sústavy.

Pôvodný prototyp scény vesmíru obsahoval pevne určené súhvezdie Veľký voz, ktoré mal žiak z dostupných tvarov zostaviť. Vo finálnej verzii bola scéna rozšírená o možnosť výberu medzi dvoma súhvezdiami – *Orion* a *Ursa Major* (Veľký voz). Žiakovi je tým ponechaná čiastočná kontrola nad priebehom úlohy, čo podporuje pocit autonómie a zvyšuje jeho zaangažovanosť vo výučbovom procese. Z pedagogického hľadiska zároveň výber medzi viacerými súhvezdiami umožňuje žiakovi pri opakovanom prechode scénou prejsť úlohou viackrát s odlišným cieľom, čím sa rozširuje obsahové pokrytie tematického celku.

Voliteľné odhaľovanie vzoru súhvezdia ako forma nápovedy.

Pôvodne bol vzor súhvezdia v scéne premietnutý ako stálo viditeľný obraz na pozadí. Tento prístup však nezohľadňoval rôznu úroveň znalostí žiakov – tí, ktorí súhvezdie nepoznali, sa museli riadiť vzorom mechanicky, kým tí, ktorí ho poznali, mali úlohu triviálne jednoduchú. Vo finálnej verzii preto vzor súhvezdia nie je predvolene viditeľný a žiak ho môže odhaliť ako voliteľnú nápovedu prostredníctvom interaktívneho prvku v scéne. Aj toto rozšírenie je realizované pomocou komponentu *activate-object-on-event*, ktorý vzor zobrazí v okamihu, keď si žiak nápovedu vyžiada. Tento prístup zodpovedá zisteniam z analýzy literatúry (kapitola 3), ktoré poukazujú na to, že efektivitu výučby zvyšuje možnosť individuálneho prispôsobenia úrovne podpory podľa predchádzajúcich znalostí žiaka.

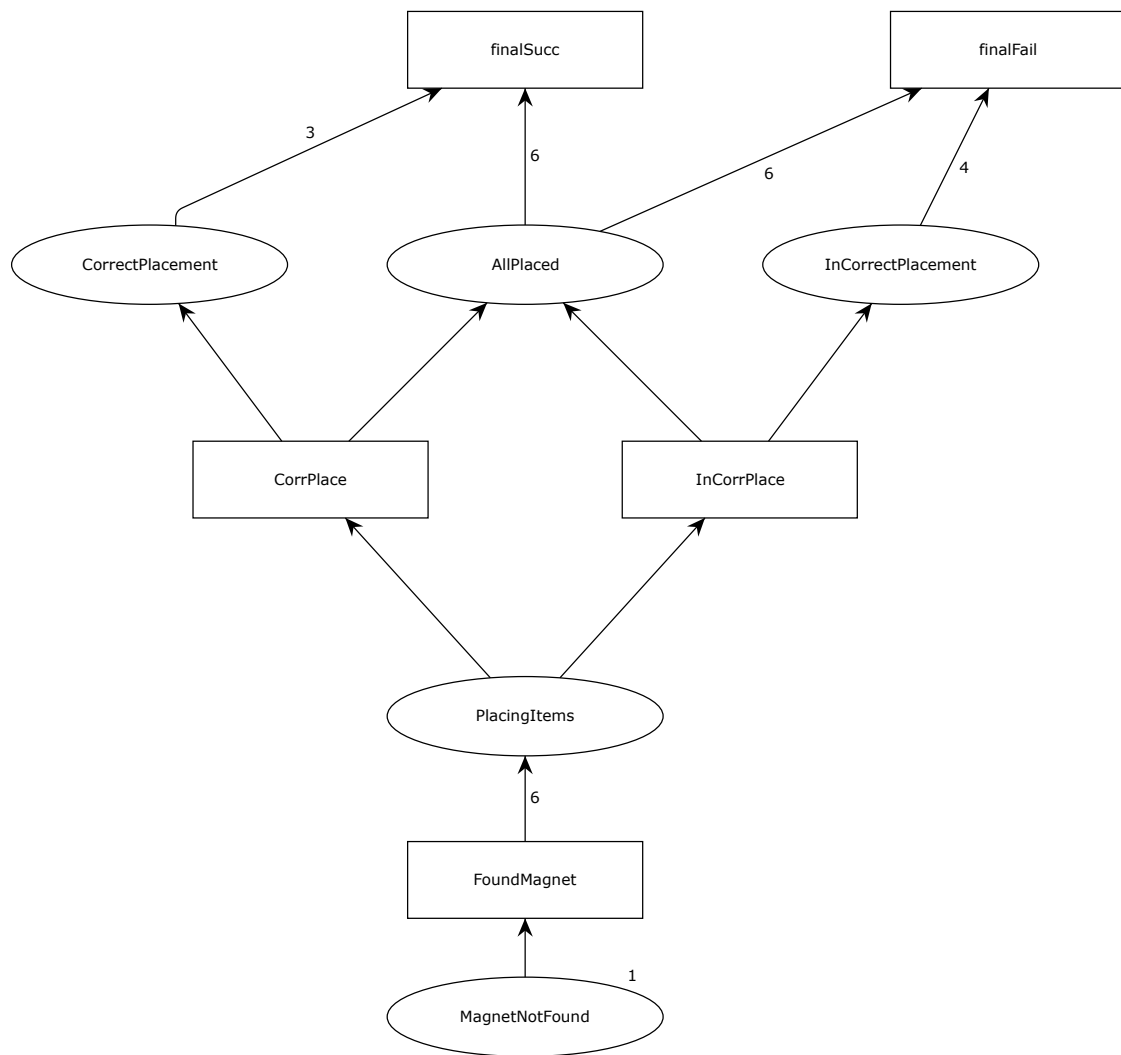
4.6 Riadenie scenárov Petriho sieťami

Postupnosť úloh, podmienky ich úspešného splnenia a vyhodnotenie scenára sú v prostredí LirkisEduVePn riadené prostredníctvom *Petriho sietí*. Petriho sieť poskytuje formálny matematický model, v ktorom *miesta* (*places*) reprezentujú jednotlivé stavy scenára, *prechody* (*transitions*) zodpovedajú akciám žiaka v scéne a *tokeny* v miestach vyjadrujú počet objektov v príslušnom stave. Každá zaznamenaná akcia žiaka v scéne (uchopenie predmetu, umiestnenie planéty, stlačenie tlačidla) v zázname z riešenia úloh zodpovedá odpáleniu konkrétneho prechodu v Petriho sieti. Petriho siete pre obe scény boli vytvorené v nástroji **CPN Tools**⁸ vo formáte CPN ML.

4.6.1 Petriho sieť magnetickej scény

Petriho sieť riadiaca magneticú scénu je znázornená na obrázku 4.10. Pozostáva z piatich miest a piatich prechodov.

⁸Dostupné na: <https://cpntools.org>



Obrázok 4.10: Petriho sieť riadiaca scenár magnetickej scény

Počiatočný stav. Petriho sieť v počiatočnom stave obsahuje jeden token v mieste `MagnetNotFound`, ktoré reprezentuje fakt, že žiak ešte nenašiel magnet v scéne. Predmety určené na triedenie nie sú v tejto fáze ešte v scéne aktivované.

Nájdenie magnetu. Prvým prechodom je `FoundMagnet`, ktorý sa odpáli v okamihu, keď žiak pomocou navádzacej guľôčky nájde a uchopí magnet (popísané v podkap. 4.5.3). Tento prechod odoberie token z miesta `MagnetNotFound` a vloží šesť tokenov do miesta `PlacingItems`, ktoré reprezentuje šesticu predmetov pripravených na triedenie. Súčasne sa v scéne aktivuje výber predmetov pomocou komponentu *random-pool-picker* (podkap. 5.2.2).

Triedenie predmetov. Z miesta `PlacingItems` vedú oblúky do dvoch prechodov:

- `CorrPlace` reprezentuje správne umiestnenie predmetu do magnetického koša podľa jeho fyzikálnej vlastnosti. Po odpálení sa token presunie do miesta `CorrectPlacement` a zároveň do miesta `AllPlaced`, ktoré počíta všetky umiestnené predmety bez ohľadu na správnosť.
- `InCorrPlace` zodpovedá nesprávnemu umiestneniu. Token sa presunie do miesta `InCorrectPlacement` a rovnako do `AllPlaced`.

Žiak môže tieto prechody opakovať toľkokrát, koľko je predmetov v scéne (šesť), pričom každý prechod zaznamenáva cieľový objekt interakcie.

Vyhodnotenie úspechu. Po umiestnení všetkých šiestich predmetov sa miesto `AllPlaced` naplní šiestimi tokenmi, čo umožní odpáliť jeden z dvoch finálnych prechodov:

- `finalSucc` sa odpáli, ak miesto `CorrectPlacement` obsahuje aspoň tri tokeny — teda žiak správne zatriedil aspoň polovicu predmetov. Tento prechod ukončí scenár ako úspešný.
- `finalFail` sa odpáli, ak miesto `InCorrectPlacement` obsahuje aspoň štyri tokeny — teda žiak väčšinu predmetov zatriedil nesprávne.

Prahové hodnoty (3 pre úspech, 4 pre neúspech) sú zakódované vo váhe vstupných oblúkov finálnych prechodov, čím sa elegantne realizuje testovacia podmienka bez dodatočnej logiky.

4.6.2 Petriho sieť vesmírnej scény

Petriho sieť vesmírnej scény je výrazne rozsiahlejšia než magnetická — obsahuje 13 miest a 15 prechodov.

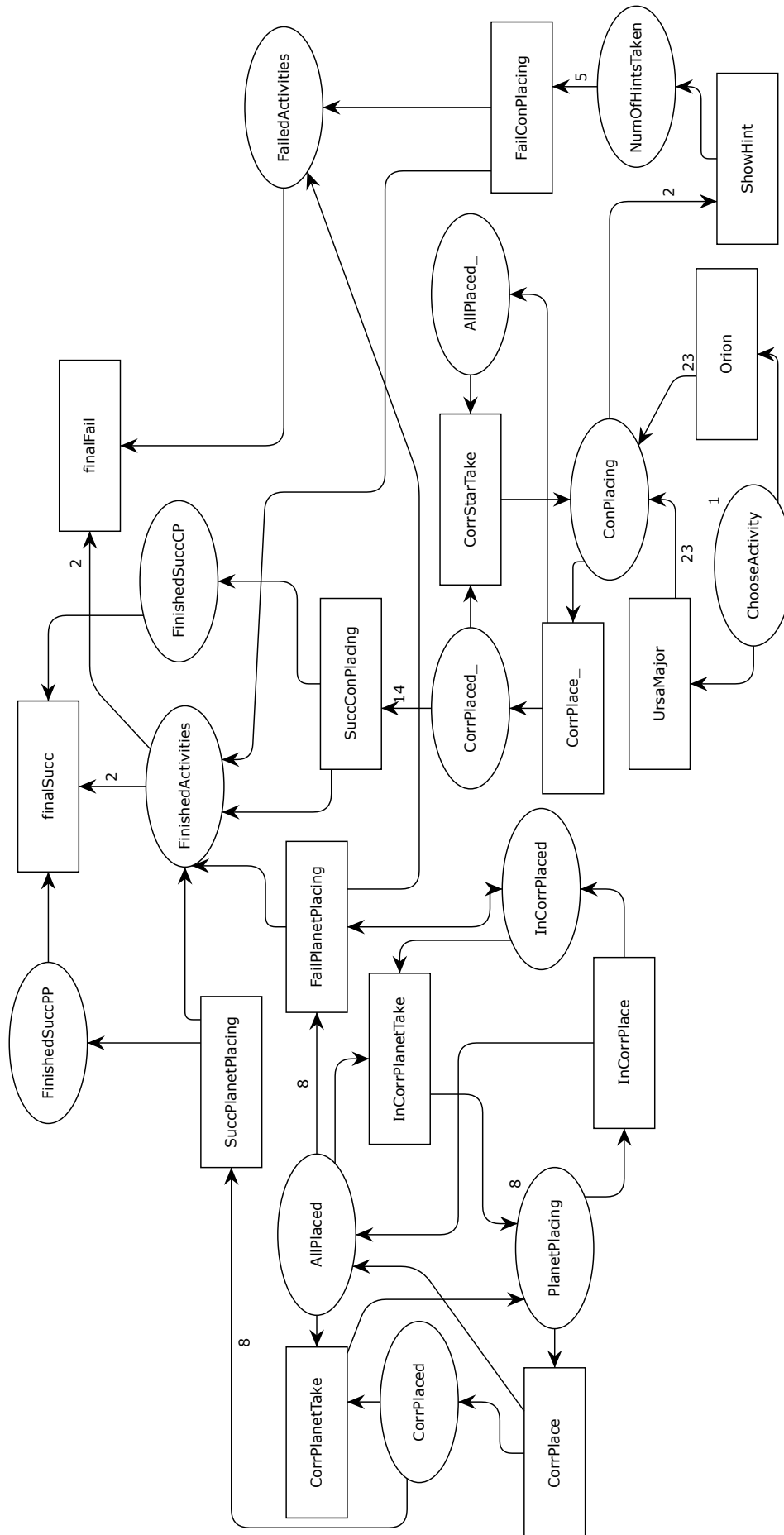
Dôvodom je, že scéna pozostáva z viacerých na seba nadväzujúcich úloh: umiestňovania planét, voľby súhvezdia, skladania súhvezdia a podporného systému nápoved.

Štruktúru siete znázorňuje obrázok 4.11.

Úloha umiestňovania planét

Sieť začína s ôsmimi tokenmi v mieste `PlanetPlacing`, ktoré reprezentuje osem planét pripravených na umiestnenie na obežné dráhy.

Žiak pri pokuse umiestniť planétu odpáli jeden z dvoch prechodov:



Obrázok 4.11: Petriho sieť vesmírnej scény

- CorrPlace pri správnom umiestnení planéty na zodpovedajúcu dráhu — token sa presunie do miesta CorrPlaced a zároveň do AllPlaced.
- InCorrPlace pri nesprávnom umiestnení — token sa presunie do miesta InCorrPlaced a tiež do AllPlaced.

Po umiestnení môže žiak planétu z dráhy zobrať späť pomocou prechodov CorrPlanetTake (zobratie planéty zo správnej dráhy — rozhodenie správneho stavu) alebo InCorrPlanetTake (zobratie z nesprávnej dráhy — oprava chyby). Oba prechody vrátia token späť do miesta PlanetPlacing, čím sa scenár stáva iteratívnym — žiak môže svoje umiestnenia opakovane korigovať. Keď sa miesto AllPlaced naplní ôsmimi tokenmi (všetky planéty umiestnené), môže sa odpáliť prechod SuccPlanetPlacing (ak je miesto CorrPlaced úplné s ôsmimi tokenmi) alebo FailPlanetPlacing (ak je v mieste InCorrPlaced aspoň jeden token). Tieto prechody následne presúvajú automaticky scenár do miesta FinishedActivities alebo FailedActivities.

Voľba a skladanie súhvezdia

Voľba súhvezdia je riadená miestom ChooseActivity, ktoré obsahuje jeden token od začiatku scenára. Žiak voľbou medzi súhvezdiami odpáli buď prechod UrsaMajor alebo Orion, čím presunie 23 tokenov do miesta ConPlacing. Hodnota 23 zodpovedá celkovému počtu interaktívnych prvkov potrebných pre dokončenie súhvezdia — 7 hviezd, 7 spojnic a ďalšie kontrolné akcie. Pre samotné skladanie súhvezdia slúžia prechody CorrPlace_ (umiestnenie hviezdy na správnu pozíciu, ktorá je jediná možná) a CorrStarTake (zobratie hviezdy späť pre opätovné umiestnenie). Tokeny sa pri správnom umiestnení presúvajú do miest AllPlaced_ a CorrPlaced_. Pri dosiahnutí 14 tokenov v CorrPlaced_ (čo zodpovedá kompletnému súhvezdiu) sa odpáli prechod SuccConPlacing, ktorý označí úlohu súhvezdia za úspešne dokončenú.

Systém nápoved

Pre úlohu skladania súhvezdia je v sieti integrovaný aj systém nápoved. Z miesta ConPlacing vedie oblúk s váhou 2 do prechodu ShowHint, ktorý odpáli pri stlačení tlačidla nápovedy (buttonHint1 pre Veľký voz, buttonHint2 pre Orión). Prechod preniesie tokeny do miesta NumOfHintsTaken, ktoré počíta využité nápovedy. Ak ich počet dosiahne hodnotu 5, prechod FailConPlacing prepne úlohu súhvezdia do neúspešného stavu — tento mechanizmus odrádza žiakov od nadmerného spoliehania sa na nápovedy.

Finálne vyhodnotenie

Scenár sa ukončuje jedným z dvoch finálnych prechodov:

- `finalSucc` sa odpáli, ak boli obe úlohy (planéty aj súhvezdie) splnené úspešne — teda miesta `FinishedSuccPP` a `FinishedSuccCP` obsahujú tokeny.
- `finalFail` sa odpáli, ak aspoň jedna z úloh skončila neúspechom — teda miesto `FailedActivities` obsahuje token.

Vďaka tejto štruktúre dokáže sieť rozlíšiť čiastočné úspechy (napríklad správne planéty, ale neúspešné súhvezdie) a poskytnúť pre každú úlohu samostatné vyhodnotenie zaznamenané počas riešenia úloh.

5 Implementácia

Táto kapitola sa zaoberá implementáciou z technickej stránky. Väčšina komponentov pracuje alebo úzko spolupracuje s fyzikálnou knižnicou *Ammo.js*⁹. Kapitola podrobne približuje jednotlivé komponenty, ktoré prispievajú k tvorbe scény a k jej fungovaniu prostredníctvom scenára. V najbližších podkapitolách budú komponenty rozdelené na základe ich použitia v jednotlivých scénach s ich podrobne popísanou rolou a funkcionalitou. Súborovú štruktúru a umiestnenie všetkých komponentov zobrazuje obrázok 5.1. Modely, moduly a zložky jednotlivých scenárov a scén projektu sú pre prehľadnosť uvedené samostatne na obrázku 5.2. Komponenty a súbory zvýraznené **červenou farbou** predstavujú komponenty a moduly vytvorené v rámci tejto práce, ostatné boli prevzaté z existujúceho riešenia.

⁹Knižnica *Ammo.js*: <https://github.com/c-frame/aframe-physics-system/blob/HEAD/AmmoDriver.md#ammo-body>

Obrázok 5.1: Štruktúra komponentov použitých pre scény a scenáre projektu

```

\LirkisEduVePn\LirkisEduVePn-UI\src\app\components\tasks>
├── entities
├── js
│   ├── components
│   │   ├── keyboard
│   │   ├── ammoCollisionTransition.component.js
│   │   │   ├── ammoCollisionTransition.component.spec.ts
│   │   │   ├── ammoInit.component.js
│   │   │   ├── ammoPhaseShiftHands.component.js
│   │   │   ├── attachmentBox.component.js
│   │   │   ├── betterMouseEvents.component.js
│   │   │   ├── blinkControl.component.js
│   │   │   ├── buttonControl.component.js
│   │   │   ├── crouch.component.js
│   │   │   ├── enableTextOnMarking.component.js
│   │   │   ├── grabbableByParenting.component.js
│   │   │   ├── handCollisionBodies.component.js
│   │   │   ├── handGrab.component.js
│   │   │   ├── handPointFingertipTouch.component.js
│   │   │   ├── handPointShoot.component.js
│   │   │   ├── handTeleport.component.js
│   │   │   ├── handTrackingBridge.component.js
│   │   │   ├── magnet.component.js
│   │   │   ├── mobileViewHandler.component.js
│   │   │   ├── objectProximitySensor.component.js
│   │   │   ├── onEventActivateObject.component.js
│   │   │   ├── petriNetSim.component.js
│   │   │   ├── phaseShiftHands.component.js
│   │   │   ├── planetOrbit.component.js
│   │   │   ├── pn-transition-handler.component.js
│   │   │   ├── randomPoolPicker.component.js
│   │   │   ├── raycasterForControllerButtonPressed.component.js
│   │   │   ├── richtClickStopsMouseLookControls.component.js
│   │   │   ├── sceneAmmoCollisions.component.js
│   │   │   ├── sceneLanguage.component.js
│   │   │   ├── showFinishInfo.component.js
│   │   │   ├── textMarkingUpdate.component.js
│   │   │   ├── thrownBallChecker.component.js
│   │   │   ├── timer.component.js
│   │   │   ├── toggleLabelVisibility.component.js
│   │   │   ├── transitions.ts
│   │   │   ├── updateInput.component.js
│   │   │   ├── userDashboard.component.js
│   │   │   ├── utilities.js
│   │   │   ├── visibilityBasedOnPetriNet.component.js
│   │   │   └── vrDetectionForRaycasting.component.js

```

Obrázok 5.2: Štruktúra modelov, modulov, scenárov a scén projektu

```
\LirkisEduVePn\LirkisEduVePn-UI\src\app\components\tasks>
```

```
├── js
│   ├── models
│   │   ├── actionTypes.enum.js
│   │   ├── sceneEvent.enum.js
│   │   └── typedefs.js
│   └── modules
│       ├── cpnLoader.mjs
│       ├── cpnLoader.spec.ts
│       ├── petriNet.mjs
│       ├── petriNet.spec.ts
│       ├── petriNetLoader.mjs
│       ├── petriNetLoader.spec.ts
│       ├── serverLogger.mjs
│       ├── serverLogger.spec.ts
│       ├── userActivityLogger.spec.ts
│       └── userActivityLogger.ts
├── scenarios
│   ├── elementary_school
│   │   ├── lang
│   │   │   ├── sceneDescription.en.json
│   │   │   └── sceneDescription.sk.json
│   │   └── nature_test.cpn
│   └── space_solar_system
│       ├── lang
│       │   ├── sceneDescription.en.json
│       │   └── sceneDescription.sk.json
│       └── space_solar_system.cpn
└── scenes
    ├── elementary_school
    │   ├── scripts
    │   │   └── sceneScript.ts
    │   ├── inanimate_nature.component.css
    │   ├── inanimate_nature.component.html
    │   ├── inanimate_nature.component.ts
    │   └── inanimate_nature.spec.ts
    └── space_solar_system
        ├── scripts
        │   └── sceneScript.ts
        ├── space_solar_system.component.css
        ├── space_solar_system.component.html
        ├── space_solar_system.component.spec.ts
        └── space_solar_system.component.ts
```

5.1 Podporné komponenty

Projekt LirkisEduVePn je dlhodobou rozvíjanou softvérovou platformou, na ktorej v aktuálnom akademickom roku paralelne pracovali viacerí študenti katedry. Implementácia tejto práce nadväzuje okrem komponentov vyvinutých v predchádzajúcich prácach (popísaných v podkapitole 2.3) aj na komponenty pochádzajúce z paralelne realizovanej diplomovej práce Michala Sakáča, ktorá v čase písania tejto práce ešte nebola formálne odovzdaná. Pre úplnosť technického popisu sú v tejto sekcii stručne predstavené tie z nich, ktoré sú podstatné pre pochopenie funkčnosti komponentov navrhnutých v rámci tejto práce. Pôvodné autorstvo je v každej podsekcii explicitne uvedené.

5.1.1 Komponent *ammo-init*

Komponent *ammo-init* bol vyvinutý Michalom Sakáčom v rámci jeho aktuálne riešenej diplomovej práce nadväzujúcej na bakalárku [4]. Vzhľadom na to, že jeho diplomová práca v čase písania tejto práce ešte nebola formálne odovzdaná, popis komponentu vychádza z analýzy zdrojového kódu projektu. Komponent rieši špecifický problém poradia inicializácie fyzikálnych telies v knižnici *Ammo.js*, ktorý vzniká pri kombinácii dynamicky načítavaných modelov a hierarchicky zaradených entít s fyzikálnym správaním.

Princíp fungovania

Bežný postup priradenia komponentov *ammo-body* a *ammo-shape* priamo v HTML šablóne nezohľadňuje skutočnosť, že niektoré entity sú načítavané asynchrónne a v okamihu inicializácie scény ešte nemusia mať dostupnú geometriu. Komponent *ammo-init* tento problém rieši odložením priradenia fyzikálnych komponentov až do okamihu, keď je entita pripravená, prípadne až po dokončení inicializácie inej referenčnej entity v scéne.

Komponent rozlišuje dva typy inicializácie podľa toho, či ide o entitu reprezentovanú modelom vo formáte GLTF. V prípade GLTF modelu komponent počká na udalosť *model-loaded*, ktorá signalizuje, že model je úplne načítaný a jeho geometria je dostupná pre výpočet kolízneho tvaru. Ak je model v okamihu inicializácie už načítaný, komponent priradí fyzikálne komponenty okamžite bez čakania na túto udalosť. Čakanie na úplné načítanie modelu je dôležité najmä pri použití parametra *fit: all* komponentu *ammo-shape*, ktorý vyžaduje prístup k mesh objektu modelu pri svojom vytvorení.

Druhým mechanizmom je možnosť explicitne podmieniť inicializáciu jednej entity dokončením inicializácie inej entity v scéne. Tento mechanizmus rieši situ-

áciu, kde je dynamické fyzikálne telo zanorené v hierarchii pod iným fyzikálnym telesom (napríklad objekt umiestnený na stole). Bez správneho poradia inicializácie by sa pozícia podradenej entity nesprávne interpretovala ako absolútna namiesto relatívnej voči rodičovi, čo by viedlo k jej okamžitému prepadnutiu cez rodičovské teleso v scéne.

Parametre komponentu

1. typ modelu – *isGLTF*: { *type*: 'boolean', *default*: false } – príznak signalizujúci, že entita využíva model vo formáte GLTF a inicializácia musí prebehnúť až po jeho úplnom načítaní; prípustné hodnoty *true* a *false*
2. entita inicializácie – *initializeAfter*: { *type*: 'selector' } – voliteľná referencia na entitu, ktorej dokončenie inicializácie je nevyhnutné pred priradením fyzikálnych komponentov tejto entite; prípustnou hodnotou je platný CSS selektor existujúcej entity v scéne

Implementačné detaily

Konfigurácia samotných komponentov *ammo-body* a *ammo-shape* sa neprenáša cez schému komponentu *ammo-init*, ale je zapísaná v atribútoch *data-ammo-body* a *data-ammo-shape* entity v HTML šablóne. Komponent tieto reťazce parsuje pomocou *AFRAME.utils.styleParser*, čím prevedie zápis vo formáte CSS-podobných pravidiel na objektovú reprezentáciu vhodnú pre priradenie komponentov. Pre špecifický prípad parametra *heightfieldData*, ktorý očakáva pole polí čísel a nedá sa korektne reprezentovať v textovom zápise, je implementované dodatočné spracovanie hodnoty cez *JSON.parse*.

Po priradení komponentov *ammo-body* a *ammo-shape* entite komponent sleduje udalosť *componentinitialized* a v okamihu, keď sú obe komponenty inicializované, vyšle vlastnú udalosť *ammo-init-finished* s identifikátorom entity. Tým je umožnené reťazenie inicializácií prostredníctvom parametra *initializeAfter* medzi viacerými entitami.

Použitie v scéne

V rámci tejto práce sa komponent využíva pri postupnom zobrazovaní objektov scény pomocou komponentu *activate-object-on-event* (viď podkapitola 5.3.3), kde zabezpečuje korektnú reinicializáciu fyzikálneho správania entity v okamihu jej zobrazenia. Komponent sa zároveň využíva v scéne slnečnej sústavy pri inicializácii planét umiestňovaných na obežné dráhy, kde poradie inicializácie zabezpečuje korektné fyzikálne správanie planét voči ich rotátorom.

5.1.2 Komponent *grabbable-by-parenting*

Komponent *grabbable-by-parenting* bol vyvinutý Michalom Sakáčom v rámci jeho aktuálne riešenej diplomovej práce nadväzujúcej na bakalárku [4]. Vzhľadom na to, že jeho diplomová práca v čase písania tejto práce ešte nebola formálne odovzdaná, popis komponentu vychádza z analýzy zdrojového kódu projektu. Komponent predstavuje alternatívu ku komponentu *grabbable* z knižnice *aframe-super-hands-component*¹⁰, pričom mení spôsob uchopenia objektu z prepočtu pozícií pomocou fyzikálnych síl na priamu zmenu rodičovskej entity.

Princíp fungovania

Pri uchopení objektu komponent zmení rodiča uchopenej entity na entitu reprezentujúcu ruku používateľa, čím sa pohyb a rotácia objektu automaticky synchronizujú s pohybom ruky bez potreby explicitnej aktualizácie pozície v každom snímku. Takýto prístup eliminuje typické artefakty fyzikálne riadeného uchopenia – napríklad oneskorenie objektu za rukou, jeho odrazy od kolíznych telies alebo nestabilné správanie pri rýchlych pohyboch.

Súčasne s pripnutím k ruke komponent prepne fyzikálne telo entity do režimu *kinematic*, čím sa zabráni tomu, aby na uchopený objekt naďalej pôsobila gravitácia alebo iné kolízne sily. Pri uvoľnení sa entita vráti k pôvodnému rodičovi a fyzikálne telo sa prepne späť do režimu *dynamic*, čím sa obnoví bežné fyzikálne správanie.

V desktopovom režime komponent rozširuje uchopovací mechanizmus o možnosť meniť vzdialenosť uchopeného objektu od kamery pomocou kolieska myši, prípadne objekt rotovať pohybom myši pri stlačení pravého tlačidla. Tieto rozšírenia sú implementované cez globálne poslucháče udalostí myši a sú aktívne iba počas aktívneho uchopenia.

Parametre komponentu

1. minimálna vzdialenosť – *minDistance*: { *type*: 'number', *default*: 0.2 } – najmenšia povolená vzdialenosť uchopeného objektu od kamery v desktopovom režime; kladná hodnota v metroch, menšia než *maxDistance*
2. maximálna vzdialenosť – *maxDistance*: { *type*: 'number', *default*: 3 } – najväčšia povolená vzdialenosť uchopeného objektu od kamery; kladná hodnota v metroch, väčšia než *minDistance*

¹⁰Knižnica *aframe-super-hands-component*:
aframe-super-hands-component

<https://github.com/c-frame/>

3. citlivosť kolieska – *scrollMultiplier*: { *type*: 'number', *default*: -0.0004 } – koeficient prevodu hodnoty kolieska myši na zmenu vzdialenosti uchopeného objektu; znamienko hodnoty určuje smer pohybu (záporná predvolená hodnota zabezpečuje, že rolovanie kolieskom nahor objekt približuje)
4. citlivosť rotácie – *rotationMultiplier*: { *type*: 'number', *default*: 0.01 } – koeficient prevodu pohybu myši na zmenu rotácie uchopeného objektu; kladná hodnota

Použitie v scéne

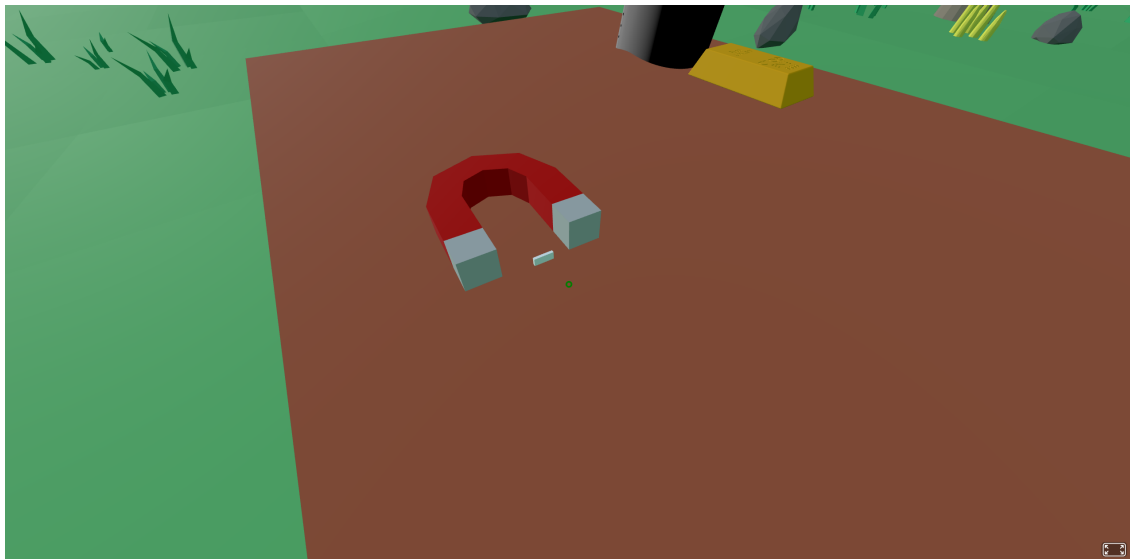
V tejto práci sa komponent využíva pri všetkých interaktívnych objektoch, s ktorými žiak fyzicky manipuluje – pri uchopovaní magnetu v magnetickej scéne, planét pri ich umiestňovaní na obežné dráhy a tvarov pri skladaní súhvezdia. V scenároch, kde je entita zobrazovaná postupne pomocou komponentu *activate-object-on-event* (viď podkapitola 5.3.3), je atribút *grabbable-by-parenting* pri skrytí entity dočasne odstránený a pri opätovnom zobrazení obnovený, čím sa zabraňuje uchopeniu neviditeľných objektov.

5.2 Komponenty magnetickej scény

Táto sekcia popisuje dizajn a funkcionálnosť jednotlivých komponentov prislúchajúcich magnetickej scéne. Následne znázorní použitie komponentov a aktív danej scény a ich interakcie s inými komponentmi a prostredím.

5.2.1 Komponent *magnet-box*

Univerzálny komponent *magnet-box* zo širšieho pohľadu, ako už názov napovedá, slúži na simuláciu magnetu reálneho sveta. Z užšieho pohľadu je reprezentovaný neviditeľným 3D boxom, ktorý je možné pripnúť na akýkoľvek predmet, čo znázorňuje obrázok 5.3. Tento prístup umožňuje oddeliť vizuálnu reprezentáciu magnetu od jeho fyzikálneho správania, čím je možné komponent priradiť ľubovoľnému 3D modelu bez nutnosti úpravy jeho geometrie.



Obrázok 5.3: Viditeľný magnetický kváder

Princíp fungovania

Komponent v každom snímku scény (metóda *tick*) prechádza zoznam okolitých objektov označených ako magnetické a na základe ich vzdialenosti od magnetu na ne aplikuje príťažlivú alebo odpudivú silu. Priťahovanie predmetov magnetom sprostredkujú fyzikálne sily knižnice *Ammo.js*¹¹ aplikované na telo *ammoBody*, a to konkrétne funkcia *applyCentralForce(ammoForce)*, kde *ammoForce* predstavuje vektor sily pôsobiacej na okolitý predmet. Smer vektora je daný rozdielom svetových pozícií magnetu a cieľového objektu, jeho veľkosť závisí od nastavenej sily, vzdialenosti a útlmovej krivky.

Magnet pôsobí výlučne na objekty, ktoré obsahujú atribút *magnetic* nastavený na hodnotu *true*. Tento prístup zabezpečuje, že komponent nezasahuje do správania ostatných fyzikálnych telies v scéne a žiak môže s nemagnetickými predmetmi voľne manipulovať bez nežiaducich vedľajších efektov.

Parametre komponentu

Komponent ponúka viacero konfiguračných parametrov, ktoré umožňujú prispôbiť silu a charakter pôsobenia magnetu konkrétnemu scenáru:

1. sila magnetu – *strength*: { *type*: 'number', *default*: 15 } – základná veľkosť pôsobiacej sily pred aplikovaním útlmu; kladná hodnota
2. rádius pôsobenia – *radius*: { *type*: 'number', *default*: 4 } – maximálna vzdialenosť, pri ktorej magnet ešte ovplyvňuje cieľový objekt; kladná hodnota v metroch

¹¹Knižnica *Ammo.js*: <https://github.com/c-frame/aframe-physics-system/blob/HEAD/AmmoDriver.md#ammo-body>

3. polarita – *polarity*: { *type*: 'number', *default*: 1 } – určuje smer pôsobenia magnetu; očakávané hodnoty 1 (priťahovanie) a -1 (odpuďzovanie), pričom hodnota zároveň lineárne škáluje veľkosť výslednej sily
4. útlm sily – *falloff*: { *type*: 'number', *default*: 1.5 } – exponent, ktorým sa sila zoslabuje so vzdialenosťou; kladná hodnota (vyššia hodnota znamená výraznejší pokles sily so vzdialenosťou)
5. minimálna vzdialenosť – *minDist*: { *type*: 'number', *default*: 0.2 } – spodná hranica vzdialenosti vstupujúcej do výpočtu, ktorá zabraňuje extrémnym hodnotám sily pri tesnom priblížení; kladná hodnota v metroch
6. maximálna sila – *maxForce*: { *type*: 'number', *default*: 100 } – horná hranica sily pôsobiacej na jeden predmet, ktorá chráni simuláciu pred numerickými nestabilitami; kladná hodnota
7. tlmenie rýchlosti – *damping*: { *type*: 'number', *default*: 0.99 } – koeficient útlmu lineárnej rýchlosti predmetu pri tesnom priblížení k magnetu; prípustný rozsah (0,1), kde hodnota 1 znamená žiadne tlmenie a hodnota 0 úplné zastavenie pohybu

Implementačné detaily

Pri inicializácii komponent automaticky priradí svojej entite fyzikálne telo *ammo-body* typu *kinematic* s aktivovaným príznakom *disableDeactivation*, čím sa predchádza automatickému uspaniu fyzikálneho enginu pri dlhšej nečinnosti telesa. Kolízny tvar je zjednodušený na kváder pomocou *ammo-shape* typu *box*, čo znižuje výpočtovú náročnosť oproti generovaniu kolízneho tvaru z geometrie modelu.

Po inicializácii scény komponent zhromaždí zoznam všetkých magnetických objektov pomocou selektora [*magnetic*="true"] a uloží si ich do interného poľa *targets*. Týmto spôsobom sa zoznam získava len raz pri spustení scény, čím sa eliminuje opakované dotazovanie DOM v každom snímku. Pre potreby prenosu vektorových hodnôt do prostredia knižnice *Ammo.js* sú vytvorené dva opakovane používané objekty typu *btVector3* (*_ammoForce* a *_ammoVel*), čím sa predchádza alokáciám nových objektov v každom snímku a znižuje sa záťaž na garbage collector prehliadača. Pri zničení komponentu sú tieto natívne objekty explicitne uvoľnené volaním *Ammo.destroy()*, aby nedošlo k úniku pamäte mimo JavaScriptový heap.

V rámci snímkovej metódy *tick* komponent najprv získa svoju aktuálnu pozíciu vo svetovom priestore a následne pre každý magnetický cieľ vykoná nasledujúce kroky. Najprv overí, či sa cieľový objekt nachádza v rámci nastavené-

ho rádiusu pôsobenia. Ak je vzdialenosť väčšia ako rádius, objekt sa preskočí a tým sa minimalizuje počet zbytočných výpočtov pre vzdialené telesá. V opačnom prípade sa zo vzdialenosti vypočíta efektívna vzdialenosť, ohraničená zdola parametrom *minDist*, čo zabraňuje deleniu hodnotou blízkou nule pri výpočte sily.

Veľkosť pôsobiacej sily je určená vzťahom $strength/vzdialenosť^{falloff}$, čo zodpovedá fyzikálnemu modelu klesajúcej sily so vzdialenosťou. Výsledná hodnota je následne orezaná hornou hranicou *maxForce* a vynásobená polaritou, ktorá určuje smer pôsobenia. Smerový vektor sily sa získa rozdielom pozície magnetu a cieľa, normalizáciou a vynásobením vypočítanou veľkosťou. Vektor je následne prevedený do reprezentácie *btVector3* a aplikovaný na fyzikálne telo cieľového objektu volaním *applyCentralForce*.

Pri tesnom priblížení cieľového objektu k magnetu (ak vzdialenosť menšia ako dvojnásobok *minDist*) komponent dodatočne aplikuje útlm na lineárnu rýchlosť tohto objektu. Aktuálna rýchlosť sa vynásobí koeficientom *damping* a nastaví sa späť na fyzikálne telo. Tento mechanizmus tlmi prudké oscilácie, ktoré by inak vznikali pri vysokých silách v blízkosti magnetu, a zabezpečuje plynulé priliepanie predmetov bez zbytočného odrazu.

Použitie v scéne

Komponent bol využitý v rámci magnetickej scény, kde slúži ako základný interaktívny prvok pre úlohu triedenia magnetických materiálov, popísanú v podkapitole 4.2. Žiak prostredníctvom magnetu testuje jednotlivé predmety rozmiestnené v prostredí, pričom magnetické predmety sú k magnetu pritiahnuté a nemagnetické zostanú nehybné. Na základe tejto interakcie následne žiak triedi predmety do dvoch krabíc podľa ich magnetických vlastností. Vďaka konfigurovateľným parametrom je možné správanie magnetu jemne doladiť tak, aby bola interakcia pre žiaka 4. ročníka základnej školy zrozumiteľná a vizuálne čitateľná, no zároveň zachovala dojem reálneho fyzikálneho javu.

5.2.2 Komponent *random-pool-picker*

Komponent *random-pool-picker* slúži na náhodný výber podmnožiny entít zo zadaného súboru kandidátov a na zaslanie aktivačnej udalosti každej z vybraných entít. Z pohľadu scenára plní funkciu randomizéra, ktorý umožňuje variabilitu obsahu scény medzi opakovanými spusteniami úlohy bez nutnosti vytvárať pre každý variant osobitnú scénu alebo scenár. Tým je do scenára zabudovaný prvok premenlivosti, ktorý zvyšuje znovuhrateľnosť úlohy a obmedzuje možnosť mechanického zapamätania si jediného správneho riešenia.

Princíp fungovania

Komponent sa pri inicializácii zaregistruje ako poslucháč definovanej udalosti na úrovni scény. Pri jej zachytení komponent vyhladá v scéne všetky entity zodpovedajúce zadanému CSS selektoru, náhodne ich premieša a z premiešaného zoznamu vyberie nakonfigurovaný počet kandidátov. Pre každú vybranú entitu komponent následne vyšle samostatnú aktivačnú udalosť, ktorej názov je tvorený konkatenáciou predpony a identifikátora danej entity.

Počet vybraných entít je možné určiť dvoma spôsobmi. V jednoduchšom prípade sa použije pevná hodnota parametra *pickCount*. Ak je však nastavený parameter *petriNetPlace*, počet sa určí dynamicky podľa aktuálneho značenia príslušného miesta v Petriho sieti – teda podľa počtu tokenov, ktoré sa v danom mieste nachádzajú v okamihu spustenia výberu. Tým je počet vybraných objektov priamo previazaný so stavom scenára, čo umožňuje, aby napríklad počet zobrazených predmetov zodpovedal hodnote, ktorú definuje samotná Petriho sieť riadiaca scénu.

Tento mechanizmus je navrhnutý tak, aby spolupracoval s komponentom *activate-object-on-event* (viď podkapitola 5.3.3), pri ktorom je každá entita prepojená s konkrétnou aktivačnou udalosťou. Týmto spôsobom je zachované oddelenie zodpovednosti medzi výberom množiny aktivovaných objektov a samotným zobrazením každého z nich – komponent *random-pool-picker* určuje ktoré entity sa aktivujú, kým komponent *activate-object-on-event* určuje ako sa aktivovaná entita zobrazí. Vďaka tomu je možné randomizér použiť na ľubovoľnú množinu objektov v scéne bez akýchkoľvek úprav samotného komponentu.

Komponent štandardne zabezpečuje, aby k výberu došlo iba raz počas životnosti scény, čím sa zabraňuje opakovanému výberu pri viacnásobnom prijatí tej istej udalosti. V prípade potreby (napríklad pri scenári, kde sa úloha v rámci jednej relácie opakuje viackrát s novým náhodným výberom) je možné toto obmedzenie vypnúť.

Komponent je deklarovaný ako *multiple: true*, čo umožňuje jeho viacnásobnú inštanciu na tej istej entite s rôznymi konfiguráciami. Týmto spôsobom je možné v jednom riadiacom uzli scény definovať viacero nezávislých skupín náhodne vybraných objektov, pričom každá skupina reaguje na vlastnú spúšťačiu udalosť a používa vlastný selektor kandidátov.

Parametre komponentu

1. názov udalosti – *listenEvent*: { *type*: 'string', *default*: 'activate' } – udalosť, ktorej prijatie na úrovni scény spustí náhodný výber kandidátov

2. predpona aktivačnej udalosti – *wakeUpPrefix*: { *type*: 'string', *default*: 'WakeUp_' } – reťazec, ktorý je pri tvorbe názvu aktivačnej udalosti spojený s identifikátorom vybranej entity
3. selektor kandidátov – *selector*: { *type*: 'string', *default*: "" } – CSS selektor, ktorým sa určuje množina entít vstupujúcich do náhodného výberu; prípustnou hodnotou je platný CSS selektor
4. počet vyberaných entít – *pickCount*: { *type*: 'number', *default*: 3 } – pevný počet entít vyberaných z množiny kandidátov; celé kladné číslo (pri hodnote menšej alebo rovnej nule sa výber nevykoná); použije sa iba vtedy, ak nie je nastavený parameter *petriNetPlace*
5. miesto Petriho siete – *petriNetPlace*: { *type*: 'string', *default*: "" } – voliteľný názov miesta v Petriho sieti, ktorého aktuálne značenie (počet tokenov) určuje počet vyberaných entít; prípustnou hodnotou je názov existujúceho miesta v Petriho sieti; ak je zadáný, má prednosť pred parametrom *pickCount*
6. jednorazový výber – *once*: { *type*: 'boolean', *default*: true } – príznak, ktorý určuje, či má komponent ignorovať opakované prijatia spúšťajúcej udalosti po prvom úspešnom výbere; prípustné hodnoty true a false

Implementačné detaily

Náhodné premiešanie kandidátov je realizované algoritmom Fisher-Yates, ktorý postupne prechádza zoznamom od konca po začiatok a každý prvok zamení s náhodne vybraným prvkom z dosiaľ nespracovanej časti. Tento algoritmus zaručuje rovnomerné rozdelenie pravdepodobností všetkých možných permutácií zoznamu a má lineárnu časovú zložitosť, čo je pri typických veľkostiach zoznamu kandidátov v rámci scény (jednotky až desiatky entít) zanedbateľná záťaž. Premiešanie sa vykonáva nad kópiou zoznamu, čím sa zabráni nežiaducej modifikácii pôvodného poradia entít v DOM.

Previazanie s Petriho sieťou je realizované cez udalosť *petri-net-prepared*, ktorú komponent zachytáva na úrovni scény a prostredníctvom ktorej získa referenciu na pripravenú Petriho sieť. Samotné určenie počtu vyberaných entít zabezpečuje metóda *_resolvePickCount*: ak parameter *petriNetPlace* nie je nastavený, vráti pevnú hodnotu *pickCount*; v opačnom prípade vyhledá v Petriho sieti príslušné miesto metódou *findPlace* a vráti jeho aktuálne značenie. Metóda je navrhnutá defenzívne – ak je názov miesta zadáný, ale Petriho sieť ešte nie je pripravená alebo sa dané miesto v sieti nenájde, metóda zaznamená varovanie do konzoly a vrá-

ti hodnotu signalizujúcu, že k výberu nemá dôjsť, čím sa predíde nesprávnemu výberu nad neúplným stavom scenára.

Pred odoslaním aktivačnej udalosti komponent overí, že vybraná entita má priradený identifikátor. Entity bez identifikátora nemôžu byť aktivované, keďže názov udalosti je zložený z predpony a identifikátora; v takom prípade komponent zaznamená varovanie do konzoly a entitu preskočí. Tento defenzívny prístup uľahčuje odhalenie chýb v konfigurácii scény, kde napríklad omylom vznikne kandidátska entita bez priradeného ID.

Pre korektnú spoluprácu s ostatnými komponentmi systému sú obsluhy udalostí uložené v interných referenciách (*_onEvent* pre spúšťaciu udalosť a *_onPetriNetPrepared* pre prípravu Petriho siete), čo umožňuje ich spoľahlivé odregistrovanie pri zničení komponentu prostredníctvom metódy *remove*.

Použitie v scéne

Komponent sa využíva v rámci scény magnetizmu pri úlohe triedení predmetov. Žiakovi je v každej relácii náhodne vybraná podmnožina tvarov z dostupnej zásoby. Tým je zabezpečené, že žiak pri opakovanom riešení úlohy nepracuje vždy s tou istou skupinou objektov, čo podporuje aktívne uvažovanie nad úlohou namiesto memorovania pevného poradia tvarov. Vďaka kombinácii s komponentom *activate-object-on-event* sa každý vybraný tvar zobrazí v predpísanej pozícii a stáva sa interaktívnym pre žiaka, kým ostatné kandidátske tvary zostávajú skryté a do úlohy nevstupujú.

5.3 Komponenty vesmírnej scény

5.3.1 Komponent *orbit-on-collision*

Komponent *orbit-on-collision* slúži na simuláciu obehu telesa po kruhovej dráhe pri kolízii s určeným objektom. Z pohľadu fungovania komponent sleduje kolízne udalosti na objekte dráhy a pri kontakte s povoleným telesom ho priradí k orbite, po ktorej následne automaticky obieha. Tento prístup umožňuje vytvoriť ilúziu dynamickej slnečnej sústavy, v ktorej žiak aktívne určuje, ktorá planéta patrí na ktorú dráhu, namiesto pasívneho sledovania vopred animovaného modelu.

Princíp fungovania

Geometricky je dráha reprezentovaná entitou kruhového tvaru, ktorej polomer je odčítaný priamo z atribútu *geometry* pri inicializácii komponentu. Tým je zabezpečené, že vizuálna reprezentácia dráhy v scéne a trajektória pohybu obiehajúce-

ho telesa sa zhodujú bez nutnosti duplicitného definovania polomeru.

Pohyb telesa po dráhe je zaručený priamou aktualizáciou súradníc v každom snímku pomocou parametrickej rovnice kružnice, pričom výsledná pozícia je transformovaná do svetového priestoru pomocou matice *matrixWorld* a následne prevedená do lokálneho priestoru rodiča rotátora. Tento dvojstupňový prevod je nevyhnutný preto, že entita reprezentujúca planétu je zanorená v hierarchii scény a jej lokálne súradnice nezodpovedajú priamo svetovým súradniciam dráhy.

Pre samotné pripnutie telesa na dráhu komponent využíva zovšeobecnený komponent *attachment-box*, ktorého autorom je Michal Sakáč vo svojej práci [4]. Zovšeobecnením tohto komponentu bolo možné jeho opätovné použitie v novom kontexte obehu planét, bez nutnosti duplikácie logiky pripájania a odpájania objektov.

Parametre komponentu

Konfigurácia komponentu umožňuje prispôbiť rýchlosť a cieľ obehu konkrétnemu scenáru a obmedziť, ktoré objekty môžu byť na dráhu prijaté:

1. cieľový rotátor – *rotator*: { *type*: 'selector' } – entita, ktorej pozícia je v každom snímku aktualizovaná podľa parametrickej rovnice kružnice; prípustnou hodnotou je platný CSS selektor existujúcej entity
2. pripájací box – *attachmentBox*: { *type*: 'selector' } – referencia na pomocnú entitu s komponentom *attachment-box*, ktorá zabezpečuje fyzické pripnutie a odpojenie planéty; prípustnou hodnotou je platný CSS selektor existujúcej entity
3. povolené ID objektov – *allowedIDs*: { *type*: 'array' } – zoznam identifikátorov objektov, ktoré môžu byť na danú dráhu pripojené, čo umožňuje viazať konkrétnu planétu na konkrétnu dráhu; prípustnou hodnotou je zoznam identifikátorov existujúcich entít v scéne
4. rýchlosť pohybu – *speed*: { *type*: 'number', *default*: 0.002 } – prírastok uhla v radiánoch pridávaný v každom snímku, čím sa určuje uhlová rýchlosť obehu telesa; kladná hodnota (záporná hodnota by obrátila smer obehu)

Implementačné detaily

Pri inicializácii si komponent uloží polomer kruhovej dráhy z geometrie vlastnej entity, vytvorí pomocný vektor *position* pre opakovanú aktualizáciu pozície a inicializuje uhol obehu na nulu. Použitie jedného opakovane zdieľaného vektora namiesto vytvárania nového v každom snímku znižuje záťaž na garbage collec-

tor prehliadača, čo je pri pravidelne volanej metóde *tick* podstatné pre plynulosť scény.

Pripnutie telesa na dráhu prebieha reaktívne na základe udalosti *collidestart*, ktorú vyšle fyzikálny engine pri kontakte dvoch kolíznych telies. V obsluhu udalosti komponent najprv overí tri podmienky: či je identifikátor kolidujúceho objektu súčasťou zoznamu *allowedIDs*, či dráha aktuálne nemá pripojený iný objekt a či kolidujúci objekt už nie je v stave *orbiting* z inej dráhy. Tento trojstupňový filter je nevyhnutný preto, že fyzikálny engine môže v jednom snímku vyhodnotiť kolíziu jedného telesa s viacerými dráhami súčasne, čo by bez kontroly stavu viedlo k nedeterministickému správaniu.

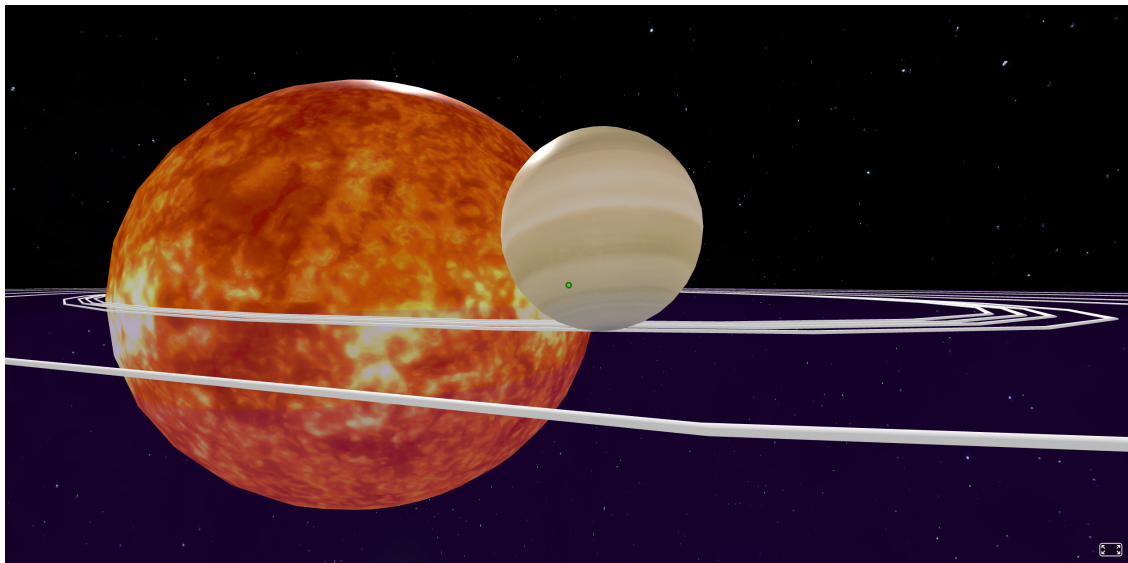
Aby sa predišlo kolíznym podmienkam pri súčasnom dotyku viacerých objektov s dráhou, komponent využíva stavový mechanizmus A-Frame. Priradený objekt nadobúda stav *orbiting*, ktorý zabraňuje jeho opakovanému zachyteniu iným kolíznym objektom dovtedy, kým žiak planéty z dráhy explicitne neodoberie. Po splnení podmienok komponent deleguje samotné fyzické pripnutie na metódu *attach* komponentu *attachment-box* a aktivuje animáciu obiehajúcej entity nastavením atribútu *animation* na hodnotu *enabled*. Súčasne sa registruje jednorazový poslucháč udalosti *dettached*, ktorý pri odobratí planéty z dráhy animáciu pozastaví, odstráni stav *orbiting* a uvoľní internú referenciu na pripojený objekt. Použitie jednorazového poslucháča (príznak *once: true*) zabezpečuje, že po každom cykle pripojenia a odpojenia sa poslucháč automaticky odstráni a nehromadí sa pri opakovaných interakciách.

V rámci snímkovej metódy *tick* komponent v každom snímku zvýši aktuálny uhol obehu o hodnotu *speed* a vypočíta novú pozíciu telesa pomocou parametrickej rovnice kružnice v rovine X-Y. Vypočítaný vektor pozície sa najprv transformuje do svetového priestoru aplikovaním matice *matrixWorld* entity dráhy, čím sa zohľadní jej aktuálne umiestnenie a rotácia v scéne. Následne sa svetová pozícia prevedie do lokálneho priestoru rodiča rotátora pomocou metódy *worldToLocal*, čo zabezpečí korektné zobrazenie aj v prípade, že je rotátor zanorený v hierarchii scény. Týmto prístupom je komponent funkčný nezávisle od toho, ako hlboko v hierarchii scény sa entity nachádzajú, a nevyžaduje špecifickú štruktúru rodičovských uzlov.

Použitie v scéne

Komponent bol použitý v rámci scény slnečnej sústavy, kde predstavuje dráhy jednotlivých planét obiehajúcich okolo Slnka. Žiak môže planétu uchopiť a umiestniť ju na konkrétnu dráhu, po ktorej sa planéta začne pohybovať (viď obrázok 5.4). Vďaka parametru *allowedIDs* je každá dráha viazaná na konkrétnu planétu. Tento

komponent teda priamo podporuje úlohu priradovania planét na správne miesta v slnečnej sústave spomínanú v kapitole 4, konkrétne podkapitole 4.2.



Obrázok 5.4: Planéta pohybujúca sa po dráhe

5.3.2 Komponent *proximity-sensor*

Komponent *proximity-sensor* slúži na detekciu blízkosti zdrojových objektov voči senzoru a na vykonanie definovaných akcií pri vstupe alebo výstupe objektu z daného polomeru. Z architektonického pohľadu ide o všeobecný komponent navrhnutý tak, aby ho bolo možné využiť v rôznych scenároch – od jednoduchého spustenia animácie pri priblížení žiaka, cez vizuálne zvýraznenie objektu zmenou farby podľa vzdialenosti, až po vyslanie udalosti, na ktorú reagujú iné komponenty v scéne. Tým komponent poskytuje jednotný mechanizmus priestorovej spätnej väzby a umožňuje znižovať mieru explicitnej interakcie potrebnej zo strany žiaka.

Princíp fungovania

Komponent v pravidelných časových intervaloch porovnáva aktuálnu pozíciu senzora s pozíciami všetkých nakonfigurovaných zdrojových objektov a podľa zmeny stavu (vstup do polomeru, zotrvanie v ňom, výstup z polomeru) spúšťa nakonfigurované akcie. Kontrola vzdialenosti neprebíha v každom snímku, ale v nastaviteľnom časovom intervale pomocou interného časovača, čím sa znižuje výpočtová záťaž scény. Vzdialenosť sa vypočítava v svetovom priestore pomocou metódy *getWorldPosition*, čo zaručuje korektný výsledok aj v prípade, že senzor alebo zdrojový objekt sú zanorené v hierarchii scény a ich lokálne súradnice nezodpovedajú svetovým.

Komponent podporuje tri typy akcií, ktoré je možné kombinovať podľa potreby konkrétneho scenára. Akcia *animation* spustí alebo pozastaví animáciu objektu pri zmene jeho stavu vzhľadom na polomer senzora. Akcia *color* plynule mení farbu objektu na základe jeho vzdialenosti od senzora pomocou predgenerovanej farebnej palety interpolovanej medzi hodnotami *colorStart* a *colorEnd*. Akcia *event* vyšle definovanú udalosť na zdrojový objekt pri jeho vstupe alebo výstupe z polomeru, čím umožňuje voľnú väzbu s ďalšími komponentmi v scéne bez priameho prepojenia.

Parametre komponentu

Konfigurácia komponentu pokrýva tri oblasti – výber zdrojových objektov a frekvenciu kontroly, voľbu typu akcie, a parametre špecifické pre vizuálnu spätnú väzbu prostredníctvom farby:

1. zdrojové objekty – *sources*: { *type*: 'selectorAll', *default*: null } – selektor identifikujúci entity, ktorých vzdialenosť od senzora je sledovaná; v prípade prázdnej hodnoty komponent sleduje samú entitu, ku ktorej je pripnutý; prípustnou hodnotou je platný CSS selektor
2. polomer detekcie – *radius*: { *type*: 'number', *default*: 5 } – maximálna vzdialenosť od senzora, do ktorej sa zdrojový objekt považuje za prítomný v detekčnej zóne; kladná hodnota v metroch
3. interval kontroly – *checkInterval*: { *type*: 'number', *default*: 200 } – minimálny časový rozostup medzi dvoma kontrolami v milisekundách, ktorý umožňuje optimalizovať pomer medzi presnosťou detekcie a výpočtovou záťažou; kladná hodnota v milisekundách
4. typ akcie – *action*: { *type*: 'array', *default*: ['animation'] } – pole definujúce, ktoré akcie sa pri zmene stavu vykonávajú; prípustné hodnoty sú *animation*, *color* a *event*, ktoré možno ľubovoľne kombinovať
5. počiatočná farba – *colorStart*: { *type*: 'color', *default*: '#0000ff' } – farba zodpovedajúca minimálnej vzdialenosti od senzora pri akcii *color*; prípustnou hodnotou je platný farebný kód (napríklad hexadecimálny zápis)
6. koncová farba – *colorEnd*: { *type*: 'color', *default*: '#ff0000' } – farba zodpovedajúca hraničnej vzdialenosti rovnajúcej sa nastavenému polomeru; prípustnou hodnotou je platný farebný kód
7. počet krokov palety – *colorSteps*: { *type*: 'number', *default*: 32 } – počet diskrétnych odtieňov, ktoré sú vopred vypočítané medzi počiatočnou a koncovou

farbou; celé číslo ≥ 2 (nižšie hodnoty sú automaticky zvýšené na 2)

8. udalosť pri vstupe – *emitEnter*: { *type*: 'string', *default*: " } – názov udalosti, ktorá sa vyšle na zdrojový objekt v okamihu jeho vstupu do detekčnej zóny
9. udalosť pri výstupe – *emitExit*: { *type*: 'string', *default*: " } – názov udalosti vysielanej pri opustení detekčnej zóny

Implementačné detaily

Pri inicializácii komponent najprv overí platnosť nastavenia parametra *action*, ktorý musí obsahovať aspoň jednu z podporovaných hodnôt. V prípade neplatného nastavenia komponent zaznamená chybu do konzoly a nastaví interný príznak *_valid* na hodnotu *false*, čím sa následne vypne celá logika komponentu vrátane snímkovej metódy *tick*. Tento defenzívny prístup zabraňuje tichému zlyhaniu konfigurácie a uľahčuje odhalenie chýb pri integrácii komponentu do nových scén.

Pre evidenciu aktuálne detegovaných objektov komponent využíva dátovú štruktúru *Set*, do ktorej ukladá referencie na zdrojové entity nachádzajúce sa v detekčnej zóne. Použitie množinovej štruktúry namiesto poľa zaručuje konštantnú časovú zložitosť pri overovaní príslušnosti, čo je pri pravidelných kontrolách v rámci snímkovej metódy podstatné. Farebná paleta používaná akciou *color* je vypočítaná raz pri inicializácii a opätovne prepočítaná pomocou metódy *update* iba v prípade, že sa skutočne zmenila niektorá z farebných hodnôt alebo počet krokov. Tým sa zabraňuje zbytočným prepočtom pri zmenách iných parametrov komponentu.

V rámci snímkovej metódy *tick* komponent inkrementuje vnútorný časovač o uplynulý čas snímku a kontrolu vzdialeností vykoná iba v prípade, že časovač prekročil hodnotu parametra *checkInterval*. Po vykonaní kontroly sa časovač nastaví späť na nulu. Tento mechanizmus efektívne oddeľuje frekvenciu detekcie od snímkovej frekvencie scény, čo je dôležité najmä na slabších zariadeniach, kde sa snímková frekvencia môže výrazne meniť.

Samotná kontrola spočíva v získaní svetových pozícií senzora a všetkých zdrojových objektov, výpočte ich vzdialenosti a porovnaní s polomerom detekcie. Komponent rozlišuje medzi tromi situáciami: vstupom do zóny (objekt je dnu, no v predchádzajúcom cykle nebol), výstupom zo zóny (objekt už nie je dnu, ale v predchádzajúcom cykle bol) a zotrvaním v zóne. Pri zmene stavu sa volá metóda *_onStateChange*, ktorá podľa nakonfigurovaných akcií reaguje príslušným spôsobom – spustí alebo pozastaví animáciu, obnoví pôvodnú farbu objektu pri jeho výstupe zo zóny, alebo vyšle definovanú udalosť. Pri zotrvaní v zóne komponent

v prípade aktivovanej akcie *color* priebežne aktualizuje farbu objektu podľa jeho aktuálnej vzdialenosti od senzora.

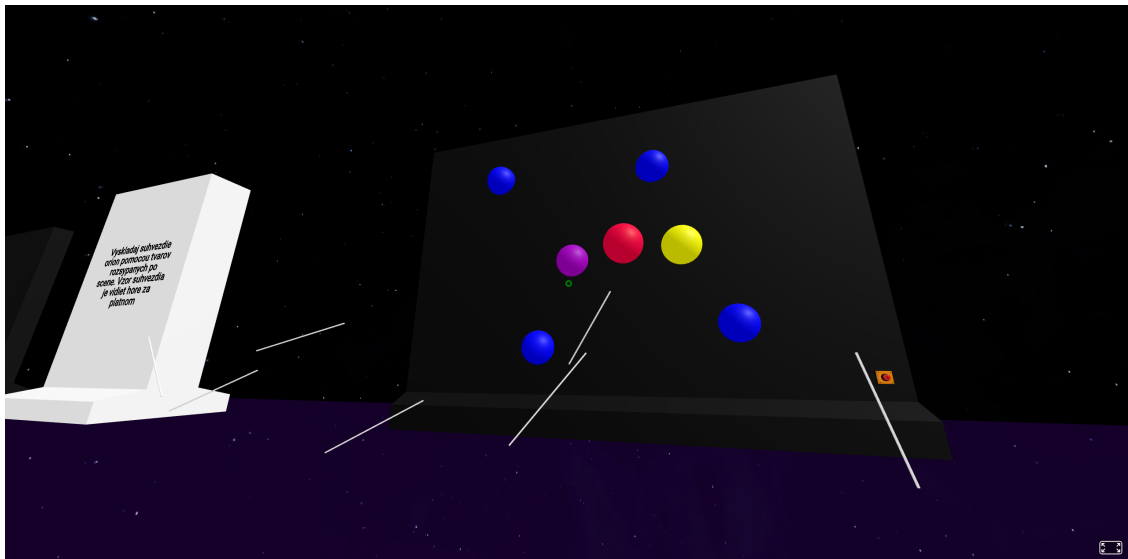
Mapovanie vzdialenosti na farbu prebieha lineárnou interpoláciou v rozsahu od nuly po polomer detekcie. Vypočítaný interpolačný koeficient je následne premietnutý na index v predgenerovanej palete, čo je výpočtovo efektívnejšie ako interpolácia farieb v každom snímku, no vďaka dostatočne vysokému počtu krokov palety je rozdiel pre používateľa vizuálne nepostrehnuteľný.

Osobitnú pozornosť si vyžaduje uchovanie pôvodnej farby objektu pred prvou modifikáciou a jej obnovenie pri výstupe zo zóny. Komponent podporuje dva spôsoby reprezentácie materiálu: priamo na úrovni objektov *three.js* (cez *userData* materiálu) a na úrovni A-Frame entity prostredníctvom komponentu *material* (cez *dataset* HTML elementu). Pôvodná farba sa zaznamená pri prvej modifikácii a využíva sa pri obnove pôvodného vzhľadu objektu po jeho opustení detekčnej zóny.

Špecifickou výzvou je situácia, keď sa jeden objekt v rámci jedného snímku nachádza v dosahu viacerých senzorov súčasne. V takom prípade by sa farba objektu menila podľa toho, ktorý senzor naposledy zapísal hodnotu, čo by viedlo k nedeterministickému vizuálnemu výsledku. Komponent túto situáciu rieši časovým oknom v dĺžke 50 milisekúnd, počas ktorého si pamätá vzdialenosť od najbližšieho senzora a zmenu farby povolí iba vtedy, ak je aktuálny senzor bližšie než predošlý zápis v rámci toho istého okna. Tým je zaručené, že farba objektu vždy odráža vplyv najbližšieho senzora.

Použitie v scéne

Komponent nachádza uplatnenie ako pomocný prvok pri tvorbe interaktívnych scén, kde je žiaduce reagovať na pohyb žiaka alebo ním držaného predmetu v priestore. V rámci scény slnečnej sústavy bol komponent použitý pri úlohe replikácie súhvezdia, kde žiak pomocou priblíženia uchopeného tvaru k cieľovému bodu vidí jeho farebné zafarbenie odzrkadľujúce vzdialenosť k cieľu (ako vidieť na obrázku 5.5), čím získava intuitívnu priestorovú spätnú väzbu o správnosti svojho postupu. Vďaka kombinovateľnosti akcií je možné rovnaký mechanizmus využiť aj pre iné typy spätnej väzby v budúcich scénach – napríklad automatické spustenie animácie objektu v momente, keď sa k nemu žiak priblíži, alebo vyslanie udalosti spúšťajúcej iné komponenty v scéne. Tým komponent umožňuje poskytovať žiakovi spätnú väzbu o jeho pohybe v priestore bez nutnosti explicitnej interakcie.



Obrázok 5.5: Replikácia súhvezdia pomocou 3D tvarov

5.3.3 Komponent *activate-object-on-event*

Komponent *activate-object-on-event* slúži na riadené zobrazovanie objektov v scéne v reakcii na definovanú udalosť. Z pohľadu scenára plní funkciu spínača, ktorý umožňuje, aby objekt v scéne fyzicky existoval, no zostal pre žiaka neviditeľný a neinteraktívny kým nenastanú podmienky vyplývajúce z postupu úlohy. Tento prístup umožňuje pripraviť všetky aktíva scény vopred a ich postupné odhaľovanie riadiť scenárom úlohy bez nutnosti dynamického vytvárania a odstraňovania entít za behu, čo je zo strany výkonu prehliadača výrazne výhodnejšie.

Princíp fungovania

Komponent pri inicializácii skryje entitu, ku ktorej je pripnutý, a zaregistruje sa ako poslucháč definovanej udalosti na úrovni scény. Pri zachytení tejto udalosti sa entita zobrazí, voliteľne sa premiestni na novú pozíciu, rotáciu a mierku, a v prípade potreby sa jej znova priradia fyzikálne a interakčné komponenty. Po uplynutí nastavenej doby zobrazenia sa entita opäť skryje a v prípade potreby sa obnoví do svojho pôvodného stavu.

Komponent rozlišuje dva režimy správania. V základnom režime je každá entita aktivovateľná iba raz a opätovné prijatie udalosti je ignorované, čo zodpovedá scenárom typu jednorazového odhalenia objektu (napríklad zobrazenie kompasu po splnení predošlej fázy úlohy). V režime s povolenou opakovateľnosťou je možné entitu aktivovať a deaktivovať opakovane, čo umožňuje napríklad cyklické zobrazovanie nápoedy alebo viacnásobné použitie tej istej entity v rôznych fázach scenára.

Pre prepojenie s ostatnými komponentmi systému je možné v rámci udalosti

odovzdať konkrétnu pozíciu, rotáciu a mierku zobrazenia. Ak nie sú v udalosti uvedené, použijú sa hodnoty z konfigurácie komponentu. Ak nie sú nastavené ani v konfigurácii, entita sa zobrazí vo svojej pôvodnej polohe. Tým je komponent flexibilný voči rôznym scenárom – od vopred určeného zobrazenia v pevnej pozícii až po dynamické umiestnenie podľa kontextu inej udalosti scenára.

Parametre komponentu

Konfigurácia komponentu pokrýva tri oblasti – definíciu udalosti a režimu opakovania, počiatočné nastavenie pozičných parametrov pri zobrazení a parametre určujúce inicializáciu fyzikálneho tela:

1. názov udalosti – *listenEvent*: { *type*: 'string', *default*: 'activate' } – udalosť, ktorej prijatie na úrovni scény spustí aktiváciu entity
2. typ modelu – *isGltf*: { *type*: 'boolean', *default*: false } – príznak určujúci, či ide o entitu reprezentovanú modelom vo formáte GLTF, čo ovplyvňuje spôsob inicializácie fyzikálneho tela; prípustné hodnoty true a false
3. entita inicializácie – *initializeAfter*: { *type*: 'selector', *default*: null } – referencia na entitu, ktorej dokončenie inicializácie podmieňuje vytvorenie fyzikálneho tela aktivovaného objektu; prípustnou hodnotou je platný CSS selektor existujúcej entity
4. doba zobrazenia – *duration*: { *type*: 'number', *default*: 0 } – časový limit zobrazenia v milisekundách, po uplynutí ktorého sa entita automaticky skryje; hodnota ≥ 0 , pričom hodnota 0 znamená neobmedzené zobrazenie
5. opakovateľnosť – *allowRepetition*: { *type*: 'boolean', *default*: false } – príznak povoľujúci opakovanú aktiváciu entity tou istou udalosťou; prípustné hodnoty true a false
6. pozícia zobrazenia – *showPosition*: { *type*: 'vec3', *default*: { *x*: 0, *y*: 0, *z*: 0 } } – voliteľná pozícia, na ktorú sa entita premiestni v okamihu aktivácie; nulový vektor {0, 0, 0} je interpretovaný ako nenastavená hodnota (ponechá sa pôvodná pozícia)
7. rotácia zobrazenia – *showRotation*: { *type*: 'vec3', *default*: { *x*: 0, *y*: 0, *z*: 0 } } – voliteľná rotácia entity v okamihu aktivácie; nulový vektor je interpretovaný ako nenastavená hodnota
8. mierka zobrazenia – *showScale*: { *type*: 'vec3', *default*: { *x*: 0, *y*: 0, *z*: 0 } } – voliteľná mierka entity v okamihu aktivácie; nulový vektor je interpretovaný ako nenastavená hodnota (ponechá sa pôvodná mierka)

9. obnova pri skrytí – *restoreOnHide*: { *type*: 'boolean', *default*: true } – určuje, či sa pri skrytí entity obnoví jej pôvodná pozícia, rotácia a mierka, alebo či zostane v poslednom aktívnom stave; prípustné hodnoty true a false

Implementačné detaily

Pri inicializácii komponent v prvom rade vytvorí kópie pôvodnej pozície, rotácie a mierky entity, ktoré sú následne využívané pri obnove počiatočného stavu. Klonovanie hodnôt namiesto uchovania priamej referencie je nevyhnutné preto, že vektorové atribúty A-Frame entity sú spravované ako mutovateľné objekty – bez kópie by sa pri zmene pozície menili aj uložené pôvodné hodnoty a obnova by stratila zmysel. Súčasne sa entita skryje nastavením atribútu *visible* na hodnotu *false*, čím sa pripraví do počiatočného neviditeľného stavu.

Obsluha udalosti je implementovaná ako jediná funkcia uložená do internej referencie *_onEvent*, čo umožňuje jej spoľahlivé odregistrovanie pri zničení komponentu prostredníctvom metódy *remove*. Tým sa predchádza úniku pamäte v scénaroch s opakovaným vytváraním a ničením entít. Po prijatí udalosti komponent najprv overí príznak *_activated*; ak entita už bola aktivovaná a režim opakovateľnosti nie je povolený, udalosť je ignorovaná. Následne sa zruší prípadný bežiaci časovač skrytia, čo zaisťuje, že nová aktivácia v priebehu predošlého zobrazenia korektne predĺži životnosť entity, namiesto toho, aby ju nepredvídateľne skryla v dôsledku staršieho časovača.

Pri určovaní cieľovej pozície, rotácie a mierky komponent uplatňuje trojstupňovú prioritu hodnôt: najprv hľadá hodnoty v dátach prijatej udalosti, potom v konfigurácii komponentu a v konečnom dôsledku ponechá pôvodný stav. Pre rozlíšenie nastavenej a predvolenej hodnoty v konfigurácii sa využíva pomocná metóda *_isNonZeroVec3*, ktorá považuje vektor s nulovými zložkami za nenastavený. Toto rozhodnutie je pragmatické – A-Frame neumožňuje rozlíšiť medzi explicitne nastavenou nulovou hodnotou a chýbajúcou hodnotou, a v praxi je nulová pozícia, rotácia alebo mierka zriedkavo používaná ako zámerná hodnota.

Pri aktivácii komponent v prípade potreby znovu inicializuje fyzikálne telo entity prostredníctvom komponentu *ammo-init* (popísaného v podkapitole 5.1.1). Tento krok je netriviálny preto, že fyzikálne komponenty knižnice *Ammo.js* sú zviazané s konkrétnym stavom entity v čase ich vytvorenia a nedokážu sa automaticky synchronizovať s následnými zmenami pozície alebo geometrie. Komponent preto entite priradí komponent *ammo-init* s príslušnou konfiguráciou, ktorá zohľadňuje, či ide o GLTF model a či má byť inicializácia oneskorená až po dokončení inicializácie inej entity scény. Oneskorená inicializácia je dôležitá v scénaroch, kde aktivovaná entita závisí od iného fyzikálneho objektu (napríklad ro-

dičovskej platformy), ktorý ešte nemusí byť plne pripravený.

Komponent súčasne pri zobrazení obnovuje atribút *grabbable-by-parenting* (popísaný v podkapitole 5.1.2) z internej cache *_grabbableCache*, ak bola predtým naplnená. Cieľom je zachovať schopnosť uchopenia entity žiakom v opakovaných cykloch zobrazenia a skrytia, keďže pri skrytí je tento atribút z entity odstránený, aby žiak nemohol manipulovať s neviditeľným objektom.

Skrytie entity prebieha buď automaticky po uplynutí nastavenej doby *duration* prostredníctvom časovača, alebo je v prípade nulovej doby ponechané na riadenie inou logikou scenára. V metóde *_hide* sa entita najprv skryje, následne sa odstránia všetky fyzikálne komponenty (*ammo-body*, *ammo-shape*, *ammo-init*), a v prípade potreby sa atribúty pozície, rotácie a mierky obnovia do pôvodného stavu. Odstránenie fyzikálnych komponentov pri skrytí je dôležité preto, že kolízne telesá knižnice *Ammo.js* pôsobia nezávisle od vizuálnej viditeľnosti entity – bez ich odstránenia by žiak mohol s neviditeľným objektom kolidovať alebo ho nechtiac uchopiť.

Použitie v scéne

Komponent sa využíva ako univerzálny mechanizmus na postupné odhaľovanie objektov v scéne podľa priebehu scenára riadeného Petriho sieťou. V rámci scény vo vesmíre sa využíva napríklad pri odhaľovaní tvarov pre úlohu replikácie súhvezdia. Vďaka kombinácii s mechanizmom udalostí Petriho siete je možné scenár jednoducho rozšíriť o ďalšie fázy bez zásahu do logiky komponentu samotného – stačí v scéne pridať novú entitu s týmto komponentom a v scenári definovať udalosť, ktorá ju aktivuje. Komponent tým prispieva k oddeleniu definície scenára od samotnej implementácie scény, čo je v súlade s celkovou architektúrou systému LirkisEduVePn založenou na riadení úloh prostredníctvom procesných grafov.

6 Vyhodnotenie

Táto kapitola sa zaoberá vyhodnotením riešenia navrhnutého a implementovaného v predošlých kapitolách. Vyhodnotenie pozostáva z dvoch fáz, ktoré sa od seba líšia metodikou aj cieľovou skupinou. Prvá fáza, popísaná v podkapitole 4.5, predstavovala formatívne testovanie prototypov scén so šiestimi dospelými respondentmi za účelom identifikácie použiteľnostných problémov pred ďalším vývojom. Druhá fáza, ktorej sa venuje táto kapitola, predstavovala overovacie testovanie finálnej verzie riešenia v reálnych podmienkach s cieľovou skupinou – žiakmi 4. ročníka Základnej školy Gemerská Hôrka. Cieľom tejto fázy bolo overiť funkčnosť výučbového prostredia v autentickom prostredí výučby a získať spätnú väzbu od skutočných používateľov, pre ktorých je systém určený.

6.1 Metodika hodnotenia

Vyhodnotenie druhej fázy testovania bolo postavené na kombinácii dvoch komplementárnych zdrojov dát – objektívnych a subjektívnych údajov.

Objektívne údaje sú merané rovnako pre každého respondenta nezávisle od jeho výpovede. Patria sem údaje zaznamenané samotným systémom počas riešenia úloh – jednotlivé akcie žiakov v rámci scenára, vrátane úspešných a neúspešných pokusov, sekvencií prechodov Petriho siete, použitia náповed a doby trvania jednotlivých relácií.

Subjektívne údaje sú tie, ktoré o sebe poskytnú sami používatelia. Patria sem výsledky dotazníka, ktorý žiaci vyplnili po ukončení testovania za asistencie vyučujúcich.

Objektívne údaje umožňujú hodnotenie použiteľnosti scenárov nezávisle od spätnej väzby respondentov, kým dotazník zachytáva subjektívnu skúsenosť žiaka, ktorá je z pohľadu pedagogickej použiteľnosti rovnako podstatná. Pre obohatenie kvantitatívnych dát boli počas testovania zaznamenávané aj voľné pozorovania a otázky kladené žiakom v priebehu úloh.

Vzhľadom na veľkosť vzorky (15 respondentov) má druhá fáza povahu ove-

rovacieho testovania s cieľovou skupinou, nie štatisticky zovšeobecniteľného sumatívneho merania. Výsledky preto slúžia primárne ako orientačná spätná väzba od reálnych používateľov a ako podklad pre ďalší vývoj prostredia, nie ako absolútne hodnotenie použiteľnosti systému.

6.2 Testovacie prostredie VR aplikácie

Testovanie prebiehalo v priestoroch Základnej školy Gemerská Hôrka dňa 27. apríla 2026. Pre účely testovania bola pripravená dvojica zariadení – VR headset Meta Quest 3S pre imerzné použitie systému a osobný počítač s monitorom pre desktopový režim. Obe zariadenia mali prístup k tej istej webovej inštancii systému LirkisEduVePn, čo umožňuje plynulý prechod medzi režimami pre prípad potreby (napríklad pri prejavoch kybernetickej nevoľnosti).

Systém bol nasadený na lokálnom serveri v rámci školskej siete, ku ktorému sa obe zariadenia pripájali cez lokálnu IP adresu. Lokálne nasadenie zabezpečilo nezávislosť testovania od kvality školského pripojenia na internet a zároveň minimalizovalo latenciu pri komunikácii so serverom. Obráz z VR headsetu bol počas celého priebehu testovania prenášaný (*cast*) na samostatný väčší monitor, čo umožnilo sledovať, čo žiak v scéne práve robí, bez potreby nasadiť si vlastný headset. Tento postup je zaužívaný v používateľskom testovaní VR systémov a uľahčuje priebežnú diagnostiku problémov, na ktoré môže žiak naraziť.

Joysticky boli zaistené nastaviteľnými remienkami proti pádu pri prudkých pohyboch. Pre testovanie boli pripravené fiktívne používateľské účty pre každého žiaka, ktoré umožnili samostatné záznamy pre každú reláciu.

6.3 Respondenti a priebeh testovania

Respondenti.

Testovania sa zúčastnilo desať žiakov 4. ročníka Základnej školy Gemerská Hôrka, ktorí boli vybraní v spolupráci s pedagogickým zborom školy. Vekovo sa žiaci pohybovali v rozmedzí 10 – 11 rokov. Cieľová skupina pochádzala prevažne zo sociálne znevýhodneného prostredia, čo predstavuje špecifický pedagogický kontext, v ktorom inovatívne digitálne výučbové prostriedky zatiaľ nie sú bežne dostupné. Pre žiakov tak išlo o prvý kontakt s technológiou virtuálnej reality v školskom prostredí.

Okrem žiakov bolo do testovania zapojených aj 5 učiteliek vrátane riaditeľky školy, ktoré asistovali pri vyplnení záverečného dotazníka a poskytli pedagogickú

spätnú väzbu na použiteľnosť systému.

Organizácia testovania.

Testovanie prebiehalo individuálne, v dvojiciach zložených zo žiaka s lepším a žiaka so slabším prospechom. Toto dvojčkové usporiadanie bolo zvolené z dvoch dôvodov – jednak umožnilo efektívne využitie testovacieho času (ovládanie systému bolo demonštrované obom žiakom naraz), jednak vytvorilo prirodzenú sociálnu situáciu, v ktorej sa žiaci mohli vzájomne pozorovať a podporovať. Testovanie prebiehalo počas riadnej výučby a celkovo trvalo približne 9 hodín.

Predtestovacie otázky.

Pred samotným spustením testovania boli každému respondentovi položené štyri voľné otázky, ktoré slúžili na identifikáciu predchádzajúcich skúseností a potenciálnych zdravotných rizík:

1. *Je ti pravidelne nevoľno?* – otázka identifikujúca predispozíciu na kybernetickú nevoľnosť
2. *Máš predošlú skúsenosť s digitálnym výučbovým prostredím?* – otázka mapujúca digitálnu gramotnosť
3. *Máš predošlú skúsenosť s VR?* – otázka mapujúca skúsenosť s konkrétnou technológiou
4. *Do akej miery si zvedavý/á, ako funguje výučba vo VR?* – otázka mapujúca úroveň motivácie

Respondenti boli zároveň informovaní o tom, že kedykoľvek v priebehu testovania môžu požiadať o prestávku alebo ukončenie testovania, ak by im začalo byť nevoľno.

Priebeh sedenia.

Po prihlásení do svojich fiktívnych účtov bol žiakom nasadený VR headset s remienkami na ovládače. Počas celého sedenia prebiehala komunikácia s respondentom a v priestore bola zachovaná fyzická prítomnosť pre prípad potreby.

Sedenie začínalo scenárom magnetizmu (magnetická scéna). Žiaci boli oboznámení so základným ovládaním a navigáciou pomocou navádzacej guľôčky (komponent *proximity-sensor*, popísaný v podkapitole 5.3.2), ktorá funguje ako interaktívny tutoriál pre nájdenie magnetu v prostredí. Tento prvok zároveň slúžil ako jemné oboznámenie s mechanikou uchopenia objektov pomocou joystickov. V prípade, že žiak narazil na nejasnosť alebo sa v scenári zasekol, mohol o tom kedykoľvek informovať vedúceho sedenia.

V prípade, že žiakovi začalo byť počas sedenia vo VR nevoľno, bol presunutý k osobnému počítaču a úlohu dokončil v desktopovom režime. Tento postup bol zámerne zvolený, aby žiak nezostal s pocitom nedokončenej úlohy alebo neúspechu, ktorý by mohol negatívne ovplyvniť jeho ďalšie skúsenosti s výučbovými technológiami.

Po dokončení scenára magnetizmu sa žiaci presunuli na druhý scenár vo vesmírnej scéne (slnečná sústava). Pri úlohách, ktoré niektorým žiakom trvali výrazne dlhšie, ako bolo časovo plánované, im bola položená otázka, či môžu prejsť k ďalšej úlohe. Tento postup bol zvolený s ohľadom na obmedzený celkový čas testovania a na únavu mladších respondentov.

Posttestovacie otázky.

Po skončení oboch scenárov boli respondentom položené voľné otázky zamerané na ich subjektívnu skúsenosť:

1. *Je ti nevoľno? Ak áno, ako presne ti je nevoľno?* – identifikácia symptómov kybernetickej nevoľnosti
2. *Páčilo sa ti to?* – celkové subjektívne hodnotenie
3. *Naučil/a si sa niečo?* – mapovanie pedagogického prínosu z pohľadu žiaka
4. *Vedel/a by si si predstaviť, že by si toto opakoval/a a používal/a v škole?* – mapovanie ochoty opätovného použitia
5. *Čo ti na systéme vadilo?* – identifikácia použiteľnostných problémov z pohľadu žiaka

Po týchto otázkach žiaci pristúpili k vyplneniu štruktúrovaného dotazníka popísaného v sekcii 6.4.

6.4 Použitý dotazník

Pre subjektívne hodnotenie skúseností žiakov bol použitý dotazník zložený zo štyroch tematických okruhov, ktoré pokrývajú odlišné dimenzie používateľskej skúsenosti vo virtuálnej realite. Každý okruh vychádza z medzinárodne uznávaného nástroja výskumu používateľskej skúsenosti – konkrétne System Usability Scale [13], NASA Task Load Index [14], Simulator Sickness Questionnaire [15] a Igroup Presence Questionnaire [16].

Slovenský preklad otázok.

Slovenské znenie otázok pre okruhy NASA-TLX, SSQ a IPQ bolo prevzaté zo štú-

die Korečka, Mathesa a Sobotu [17], ktorá v rámci výskumnej činnosti pracoviska Katedry počítačov a informatiky Technickej univerzity v Košiciach publikovala overené slovenské preklady etablovaných dotazníkov pre meranie používateľskej skúsenosti vo virtuálnej realite.

Adaptácia SUS pre detskú vekovú skupinu.

Pôvodný SUS dotazník je v odbornej literatúre etablovaným nástrojom merania použiteľnosti, no jeho originálne formulácie sú primárne určené pre dospelých používateľov a v prípade detí mladšieho školského veku ich kognitívna náročnosť presahuje primeranú úroveň. Pre prvý okruh dotazníka bola preto využitá detská adaptácia SUS publikovaná Putnam a kol. [18] v rámci konferencie CHI 2020, validovaná pre deti vo veku 7 až 11 rokov. Vzhľadom na to, že detská adaptácia SUS bola pôvodne publikovaná v anglickom jazyku, bola na základe anglických formulácií vytvorená vlastná slovenská verzia, pričom kontextové odkazy na hernú aplikáciu boli nahradené odkazmi na VR výučbové prostredie a plnenie úloh.

Expertné posúdenie vekovej primeranosti.

Vzhľadom na to, že okruhy NASA-TLX, SSQ a IPQ neexistujú vo validovanej detskej adaptácii porovnateľnej s adaptáciou SUS od Putnam a kol., bola pre tieto okruhy vykonaná redukcia položiek na vekovo primeranú podmnožinu na základe expertného posúdenia pedagogickým zborom Základnej školy Gemerská Hôrka. Z pôvodných sád otázok boli vybrané iba tie, ktorým žiaci 4. ročníka podľa posúdenia pedagógov rozumejú a sú schopní na ne zmysluplne odpovedať. Otázky, ktoré boli identifikované ako kognitívne náročné alebo abstraktné nad rámec vekových schopností respondentov, boli z dotazníka vyradené. Tento postup zodpovedá metodike expertného posúdenia obsahovej validity dotazníka pri výskume s deťmi.

Asistované vyplnenie.

Vyplnenie samotného dotazníka prebehlo formou asistovaného vyplnenia za prítomnosti vyučujúcich, ktorí žiakom podľa potreby vysvetľovali jednotlivé otázky a pomáhali pri interpretácii formulácií. Tento prístup zodpovedá zaužívanej praxi v pedagogickom výskume s mladšími respondentmi a zabezpečuje, že odpovede odrážajú skutočný zážitok žiaka, nie nepochopenie znenia otázky.

Plná verzia použitého dotazníka so všetkými otázkami a možnosťami odpovedí je uvedená v prílohe A.

6.4.1 Použiteľnosť systému

Prvý okruh dotazníka pozostával z trinástich otázok zameraných na subjektívne hodnotenie použiteľnosti systému, ktoré vychádzajú z dotazníka System Usabili-

ty Scale [13] v detskej adaptácii podľa Putnam a kol. [18]. Otázky 1 – 10 zodpovedajú pôvodným SUS položkám preformulovaným do detsky primeraného jazyka a sú formulované striedavo pozitívne a negatívne, čo umožňuje výpočet štandardizovaného SUS skóre v rozsahu 0 – 100. Otázky 11 – 13, doplnené v rámci detskej adaptácie, sú vyhodnocované samostatne ako miera zábavnosti, ochoty pokračovať a ochoty odporučiť aplikáciu rovesníkom. Všetky otázky boli hodnotené na 5-bodovej Likertovej škále od 1 (vôbec nesúhlasím) po 5 (úplne súhlasím).

6.4.2 Kognitívna a fyzická záťaž

Druhý okruh otázok hodnotil kognitívnu a fyzickú záťaž pri plnení úloh a vychádzal z dotazníka NASA Task Load Index [14]. Z pôvodných šiestich dimenzií záťaže (mentálna náročnosť, fyzická náročnosť, časový tlak, vnímaný úspech pri plnení úloh, vynaložené úsilie a frustrácia) boli do dotazníka zaradené tie, ktoré boli expertným posúdením pedagogického zboru identifikované ako vekovo primerané pre žiakov 4. ročníka. Odpovede boli zaznamenávané na 3-bodovej škále, ktorá je pre detskú cieľovú skupinu zrozumiteľnejšia ako pôvodná 7-stupňová škála originálneho NASA-TLX.

6.4.3 Fyzické symptómy pri používaní VR

Tretí okruh sa zameriaval na identifikáciu fyziologických prejavov kybernetickej nevoľnosti a vychádzal z dotazníka Simulator Sickness Questionnaire [15]. Z pôvodného zoznamu symptómov boli do dotazníka zaradené tie, ktoré boli expertným posúdením pedagogického zboru identifikované ako vekovo primerané a zrozumiteľné pre žiakov 4. ročníka – únava, námaha očí, ťažkosti so zaostréním, bolesť hlavy, tlak v hlave, rozmazané videnie, závraty pri zatvorených očiach a točenie hlavy.

Žiaci pri tomto okruhu označovali, ktoré z uvedených symptómov pociťovali počas alebo po používaní VR. Vzhľadom na deklaratívnu povahu otázok je formát binárny, čo znižuje kognitívnu záťaž pri vyplňaní a zodpovedá pôvodnému návrhu nástroja.

6.4.4 Pocit prítomnosti vo virtuálnom svete

Štvrtý okruh dotazníka hodnotil subjektívny pocit prítomnosti vo virtuálnom svete (*presence*) a vychádzal z dotazníka Igroup Presence Questionnaire [16]. Z pôvodných položiek IPQ boli do dotazníka zaradené tie, ktoré boli expertným posúdením pedagogického zboru identifikované ako zrozumiteľné pre žia-

kov 4. ročníka. Vybrané otázky pokrývali viaceré dimenzie pocitu prítomnosti – pocit bytia v scéne, pocit obklopenia virtuálnym svetom, pocit aktívneho konania v ňom, vnímanie reálneho prostredia počas používania a vnímanú realističnosť virtuálneho sveta.

6.5 Vyhodnotenie výsledkov dotazníka

Po ukončení testovania VR aplikácie vyplnili respondenti štyri okruhy dotazníka opísané v kap. 6.4. Žiaci 4. ročníka ($n=10$) vyplňali detskú verziu prispôsobenú podľa Putnam a kol. [18] so zjednodušenými škálami a piktogramami palcov, kým učiteľky ($n=5$) absolvovali plnú verziu pre dospelých vychádzajúcu zo slovenských prekladov Korečka a kol. [17]. Vyplnené odpovede sme digitalizovali do tabuľkového procesora, kde sme automaticky prepočítali SUS skóre podľa Brookeho vzorca [13] a normalizovali IPQ otázky s otočenou polaritou. Výsledky oboch skupín uvádzame paralelne.

6.5.1 Demografické údaje respondentov

Tabuľka 6.1: Demografické zloženie respondentov

	Žiaci 4. ročníka	Učiteľky
Počet	10	5
Vek (priemer, rozsah)	10,1 (10–11)	—*
Pohlavie	7 chlapcov, 3 dievčatá	5 žien

* Učiteľky vek pri testovaní neuviedli; nie je preto dostupný.

Vzorka žiakov bola tvorená 10 deťmi 4. ročníka Základnej školy v Gemerskej Hôrke s priemerným vekom 10,1 roka (rozsah 10–11). Pohlavné zloženie bolo 7 chlapcov a 3 dievčatá. Vzorku dospelých tvorilo 5 učiteliek tej istej školy.

6.5.2 Použitelnosť — System Usability Scale (SUS)

Žiaci. Priemerné SUS skóre žiakov dosiahlo **75,0** ($SD = 12,4$; rozsah 52,5–95,0). Podľa Bangorovej krivkovej kategorizácie [19] to zodpovedá písmenu *C+* až *B* a slovne hodnoteniu „good“. V porovnaní s normou priemerných SUS skóre v literatúre (približne 68 bodov [20]) je tento výsledok mierne nadpriemerný.

Štyria žiaci (4/10) dosiahli hodnotenie „Vynikajúce“ ($SUS \geq 80$) a iba jeden žiak (P02) skončil v pásme „Slabé“. Pri rozhovore s P02 sa ukázalo, že mal počas

Tabuľka 6.2: SUS skóre žiakov podľa respondenta

Respondent	SUS skóre	Hodnotenie
P05	95,0	Vynikajúce (A)
P09	87,5	Vynikajúce (A)
P08	82,5	Vynikajúce (A)
P06	80,0	Vynikajúce (A)
P07	77,5	Dobré (B)
P00	75,0	Dobré (B)
P04	70,0	Dobré (B)
P01	67,5	Priemerné (C)
P03	62,5	Priemerné (C)
P02	52,5	Slabé (D)
Priemer	75,0	Dobré (B)

testovania problém s ovládaním pohybu vo virtuálnom prostredí, čo zodpovedá jeho nízkemu skóre.

Doplňkové otázky Q11–Q13. K štandardnej desaťotázkovej SUS sade pridali Putnam a kol. [18] tri otázky merajúce zábavnosť a motiváciu pokračovať: ochotu plniť úlohy doma (Q11), ochotu pokračovať pri viac času (Q12) a ochotu odporučiť aplikáciu kamarátom (Q13). Tieto otázky nevstupujú do klasického SUS výpočtu, ale sú vyhodnocované samostatne. Priemer odpovedí žiakov na týchto troch otázkach dosiahol **4,73 z 5**, čo predstavuje *veľmi silnú emocionálnu väzbu k aplikácii*. Sedem z desiatich žiakov označilo všetky tri otázky maximálnou hodnotou.

Učiteľky. Priemerné SUS skóre učiteľiek dosiahlo **78,0** (SD = 13,2; rozsah 57,5–92,5). Tabuľka 6.3 zobrazuje výsledky podľa respondenta.

Tabuľka 6.3: SUS skóre učiteľiek podľa respondenta

Respondent	SUS skóre	Hodnotenie
P13	92,5	Vynikajúce (A)
P10	85,0	Vynikajúce (A)
P14	80,0	Vynikajúce (A)
P12	75,0	Dobré (B)
P11	57,5	Slabé (D)
Priemer	78,0	Dobré (B)

Štyri z piatich učiteliek hodnotili použiteľnosť ako dobrú až vynikajúcu. Učiteľka P11 ($SUS = 57,5$) bola výrazný outlier oproti zvyšku skupiny — pri rozhovore uviedla, že počas testovania mala technické problémy s headsetom a interakciou s kontrolérmi, čo zodpovedá jej nízkemu skóre. Porovnanie medzi skupinami ukazuje, že použiteľnosť bola pre obe vekové kategórie hodnotená podobne (75,0 oproti 78,0), pričom učiteľky mali len mierne vyšší priemer.

6.5.3 Kognitívna záťaž — NASA-TLX

Žiaci. Detská verzia NASA-TLX bola zjednodušená na 3-bodovú škálu s jednotnou orientáciou, kde **1 = najhoršia hodnota** (najvyššia záťaž, resp. najnižšia úspešnosť) a **3 = najlepšia hodnota** (najnižšia záťaž, resp. najvyššia úspešnosť). Žiak pri každej otázke vyberal z troch možností (napríklad „náročné – priemerne – nenáročné“ alebo „málo úspešní – priemerne – veľmi úspešní“). Tabuľka 6.4 zobrazuje priemery jednotlivých dimenzií.

Tabuľka 6.4: Kognitívna záťaž žiakov (NASA-TLX, škála 1–3, 1 = najhoršia, 3 = najlepšia hodnota)

Dimenzia	Priemer
1. Vedomostná náročnosť	2,30
2. Manuálna náročnosť	2,80
3. Časový nátlak	2,80
4. Úspešnosť (3 = veľmi úspešní)	2,40
5. Úsilie	2,30
6. Frustrácia	2,50

Keďže na tejto škále vyššia hodnota znamená nižšiu záťaž, výsledky ukazujú, že žiaci vnímali úlohy ako *prevažne nenáročné*. Najvyššie hodnoty dosiahli manuálna náročnosť a časový nátlak (oba 2,80), čo znamená, že práve tieto aspekty ich zaťažovali najmenej — manipulácia s objektmi aj tempo plnenia úloh im teda vyhovovali. Mierne nižšie, no stále v pásme nízkej záťaže, boli vedomostná náročnosť a úsilie (po 2,30). Hodnota frustrácie 2,50 sa nachádza bližšie ku koncu škály označujúcemu „vôbec“, čo naznačuje nízku úroveň frustrácie počas plnenia úloh. Pri otázke úspešnosti (kde 3 = veľmi úspešní) dosiahli žiaci 2,40, teda sa hodnotili ako *priemerne až dosť úspešní* — napriek tomu, že objektívne mali pri planetárnej úlohe ťažkosti (kap. 6.6). Tento rozdiel medzi vnímanou a skutočnou úspešnosťou je sám o sebe zaujímavým zistením: žiaci úlohu nevnímali ako neúspech, hoci poradie planét často neuhádli. Celkovo nízka vnímaná záťaž ladí

s vysokým hodnotením použiteľnosti aj zábavnosti aplikácie.

Učiteľky. Dospelá verzia NASA-TLX používa štandardnú 21-bodovú škálu 0–100 v krokoch po 5 podľa Hart a Staveland [14]. Pri otázke výkonu sa hodnota invertuje (vyšší výkon = nižšia záťaž). Súhrnné TLX skóre je nevážený priemer šiestich dimenzií.

Tabuľka 6.5: NASA-TLX skóre učiteliek podľa respondenta (0–100)

Respondent	TLX skóre	Záťaž
P10	8,3	Nízka
P14	15,0	Nízka
P13	18,3	Nízka
P12	36,7	Stredne nízka
P11	53,3	Stredná
Priemer	26,3	Stredne nízka

Priemerná kognitívna záťaž 26,3 spadá do pásma „stredne nízka“. Tri z piatich učiteliek hodnotili záťaž ako *nízku* ($TLX \leq 20$). Najvyššiu záťaž pociťovala P11 ($TLX = 53,3$), čo koreluje aj s jej najnižším SUS skóre — pravdepodobne v dôsledku tých istých technických problémov pri ovládaní.

Pri jednotlivých dimenziách (tab. 6.6) si učiteľky najmenej zaťažujúco vnímali fyzickú náročnosť, časový tlak a frustráciu (každá s priemerom 16,0). Najvýraznejšie boli invertovaný výkon a úsilie (po 44,0). Hodnota 44,0 pre invertovaný výkon znamená, že učiteľky sa hodnotili ako pomerne úspešné v plnení úloh (44 zo 100 v invertovanej forme zodpovedá 56 zo 100 v pôvodnej škále výkonu).

Tabuľka 6.6: Priemerné hodnoty dimenzií NASA-TLX u učiteliek

Dimenzia	Priemer
Mentálne zaťaženie	22,0
Fyzické zaťaženie	16,0
Časový nátlak	16,0
Výkon (invertovaný)	44,0
Úsilie	44,0
Frustrácia	16,0

6.5.4 Symptómy kybernetickej choroby (SSQ / VRSQ)

Žiaci — SSQ. V detskej verzii sme použili zjednodušený binárny formát — žiak označil, či daný symptóm pociťoval (1) alebo nepociťoval (0). Tabuľka 6.7 sumarizuje výskyt jednotlivých symptómov.

Tabuľka 6.7: Symptómy kybernetickej choroby u žiakov (SSQ, počet áno z 10)

Symptóm	Počet (n=10)	Percento
Námaha očí	4	40 %
Rozmazané videnie	4	40 %
Točenie hlavy	4	40 %
Bolesť hlavy	3	30 %
Únava	2	20 %
Ťažkosti so zaostrením	1	10 %
Závraty pri zatvorených očiach	1	10 %
Tlak v hlave	0	0 %

Najčastejšie symptómy boli *vizuálneho charakteru* — námaha očí, rozmazané videnie a točenie hlavy (každý u 4 detí, t. j. 40 %). Toto je v súlade s literatúrou o VR pre deti [15], ktorá identifikuje vizuálnu únavu ako primárny symptóm pri kratších VR reláciách. Žiadny žiak neoznačil tlak v hlave a viaceré závažnejšie symptómy boli zriedkavé. Trvanie VR relácie bolo priemerne 13–15 minút (kap. 6.6), čo je v bezpečnom pásme.

Učiteľky — VRSQ. Dospelá verzia používa VRSQ podľa Kim a kol. [21] so 4-stupňovou škálou intenzity (0 = žiadny, 3 = vážny).

U učiteliek dominovali *vertigo* (4/5 pociťovali, priemer 1,00) a *námaha očí* (3/5, priemer 1,00). Intenzita symptómov bola prevažne na úrovni „mierna“ (hodnota 1). Žiadna učiteľka neoznačila „vážny“ symptóm. Vertigo a všeobecný diskomfort u učiteliek v porovnaní so žiakmi možno vysvetliť dlhším priemerným trvaním učiteliek vo vesmírnej scéne (25 min oproti 12 min u žiakov, kap. 6.6).

6.5.5 Pocit prítomnosti vo virtuálnom svete (IPQ)

IPQ podľa Schuberta [16] obsahuje 14 položiek v štyroch dimenziách: všeobecná prítomnosť (G), priestorová prítomnosť (SP), zaangažovanosť (INV) a realita (REAL). Pre slovenský preklad Korečka a kol. [17] sme reverzne skórovali otázky SP2, SP3, INV1 a INV3, kde je polarita formulácie opačná voči koncep-

Tabuľka 6.8: Symptómy učiteliek (VRSQ, priemer intenzity 0–3)

Symptóm	Priemer	Pocítovali (n=5)
Vertigo (točenie hlavy)	1,00	4/5
Námaha očí	1,00	3/5
Všeobecný diskomfort	0,80	2/5
Bolesť hlavy	0,40	2/5
Rozmazané videnie	0,40	1/5
Závraty pri zatvorených očiach	0,40	1/5
Únava	0,20	1/5
Ťažkosti so zaostrením	0,20	1/5
Tlak v hlave	0,20	1/5

tu prítomnosti. Vyššia hodnota po normalizácii teda vždy znamená vyšší pocit prítomnosti.

Žiaci. Detská verzia obsahuje 9 položiek prebraných z dotazníka Korečka a kol. so zjednodušenými škálami (5-bodová Likertova pre väčšinu, 3-bodová pre dve otázky). Otázky 3 (vnímam iba obrázky) a 7 (reálnosť VR) sme reverzne skórovali. Tabuľka 6.9 uvádza normalizované priemery.

Tabuľka 6.9: IPQ otázky u žiakov (po normalizácii, vyššie = vyšší pocit prítomnosti)

Otázka	Škála	Priemer
Q1: Pociť „bol som tam“	1–5	4,50
Q2: VR svet ma obklopuje	1–5	4,20
Q3: Iba obrázky (otočená)	1–5	3,90
Q4: Naozaj konám vo VR	1–5	3,70
Q5: Cítil som sa prítomný	1–5	4,10
Q6: Vnímanie reality okolo seba	1–3	1,90
Q7: Reálnosť VR sveta (otočená)	1–3	1,50
Q8: Zhoda s reálnym svetom	1–3	2,00
Q9: VR skutočnejší ako reálny	1–5	3,30

Najsilnejší pocit prítomnosti žiaci vyjadrili pri otázkach merajúcich *priestorovú prítomnosť* — pocit „bol som tam“ (Q1 = 4,50) a obklopenia virtuálnym svetom (Q2 = 4,20). Otázka Q5 (cítil som sa prítomný) dosiahla 4,10. Žiaci po normalizácii súhlasili, že nevnímali aplikáciu ako „iba obrázky“ (Q3 = 3,90) — to potvrdzuje, že VR pôsobila prevažne pohlcujúco.

V dimenzii *realita* ($Q7 = 1,50$ na škále 1–3) však žiaci virtuálny svet nepovažovali za príliš reálny, hoci stredná hodnota $Q9$ (VR skutočnejší ako reálny svet) bola 3,30. Tento výsledok ukazuje, že žiaci VR vnímali ako presvedčivý priestor, ale jeho „realnosť“ v zmysle vernosti reálnemu svetu skôr odmietali.

Učiteľky. Dospelá verzia obsahuje plný 14-položkový IPQ s reverzným skórovaním otázok 3, 4, 7 a 9.

Tabuľka 6.10: IPQ u učiteliek — dimenzie (po normalizácii, škála 1–7)

Dimenzia	Priemer	SD
G — Všeobecná prítomnosť	6,00	1,00
SP — Priestorová prítomnosť	6,24	0,78
INV — Zaangažovanosť	4,50	0,56
REAL — Realita	3,60	0,45

Najvyšší pocit prítomnosti učiteľky vyjadrili v dimenziách *všeobecná prítomnosť* ($G = 6,00$ zo 7) a *priestorová prítomnosť* ($SP = 6,24$). To znamená, že učiteľky pociťovali silný pocit byť „v inom svete“ a obklopenia virtuálnym prostredím. *Zaangažovanosť* ($INV = 4,50$) bola na strednej úrovni a *realita* ($REAL = 3,60$) bola najnižšia. Tento vzor je typický pre VR aplikácie zamerané na obsah s ilustračnou grafikou, kde priestorová ponornosť funguje, ale fotorealistický pocit chýba [16].

Porovnanie oboch skupín ukazuje konzistentný profil: *vysoká priestorová prítomnosť, mierna zaangažovanosť a relatívne nízky pocit reálnosti*. Toto je v súlade s charakterom prostredia LirkisEduVePn, ktoré sa zameriava na edukačnú zrozumiteľnosť scén, nie na fotorealizmus.

6.5.6 Hlavné nálezy z dotazníkov

Vyhodnotenie štyroch okruhov dotazníka prinieslo nasledovné zistenia:

1. **Dobrá použiteľnosť pre obe skupiny.** Žiaci aj učiteľky hodnotili použiteľnosť ako „dobrú“ ($SUS = 75,0$ a $78,0$). Hodnota presahuje literárny priemer 68 bodov, čo radí aplikáciu medzi nadpriemerne použiteľné systémy.
2. **Silná emocionálna väzba detí k aplikácii.** Doplnkové otázky Q11–Q13 zachytávajúce zábavnosť a motiváciu dosiahli priemer 4,73 z 5. Sedem z desiatich detí označilo všetky tri otázky maximálnou hodnotou, čo poukazuje na vysokú atraktivitu prostredia pre cieľovú skupinu.

3. **Nízka kognitívna záťaž u oboch skupín.** U učiteliek priemerné TLX skóre 26,3 (stredne nízka záťaž) a u žiakov priemery 2,3–2,8 na škále, kde 3 = najnižšia záťaž, zhodne naznačujú, že úlohy boli pre respondentov primerané až nenáročné. Aj hodnota frustrácie u žiakov (2,50) sa nachádza v pásme nízkej frustrácie, čo ladí s vysokým hodnotením zábavnosti aplikácie.
4. **Outliery v oboch skupinách.** Žiak P02 (SUS = 52,5) a učiteľka P11 (SUS = 57,5) boli výrazne pod priemerom svojich skupín. V oboch prípadoch to korelovalo s rozhovormi po testovaní, kde obaja uviedli technické ťažkosti pri ovládaní.
5. **Symptómy boli prevažne mierne.** U žiakov dominovali vizuálne symptómy (námaha očí, rozmazané videnie, točenie hlavy — každý u 40 %), u učiteliek vertigo (4/5) a námaha očí (3/5). Intenzita bola na úrovni „mierna“. Žiadny respondent nevykazoval vážny symptóm.
6. **Vysoká priestorová prítomnosť, nižšia realita.** V IPQ dosiahli obe skupiny vysoké hodnoty v dimenziách všeobecnej a priestorovej prítomnosti (učiteľky $G = 6,00$ / $SP = 6,24$; žiaci $Q1 = 4,50$ / $Q2 = 4,20$). Nižšie hodnoty v realite ($REAL = 3,60$ u učiteliek, $Q7 = 1,50$ u žiakov) odrážajú nefotorealistickú grafiku prostredia, ktorá je vedome zvolená v prospech edukačnej zrozumiteľnosti.

Tieto zistenia z dotazníkov *korelujú s analýzou zaznamenaných údajov* (kap. 6.6) a spoločne tvoria základ pre diskusiu a odporúčania v kap. 6.7.

6.6 Vyhodnotenie zaznamenaných údajov

Okrem subjektívneho hodnotenia prostredníctvom dotazníka sme z testovania získali aj objektívne údaje zaznamenané systémom prostredia LirkisEduVePn počas riešenia úloh. Tento systém pri každom významnom úkone v scéne vytvára záznam, ktorý obsahuje typ akcie, časovú značku, identifikátor relácie a stav príslušného prechodu Petriho siete. Z týchto dát sme získali objektívny pohľad na to, ako respondenti v aplikácii reálne pracovali — nezávisle od ich subjektívnej výpovede.

6.6.1 Spracovanie dát

Pri analýze sme zohľadnili nasledovné špecifiká systému záznamu:

- Pri akcii FoundMagnet v magnetickej scéne sa zaznamenáva každé chytenie magnetu do ruky, vrátane opakovaných. Túto akciu sme preto použili ako indikátor miery experimentovania s magnetom.
- Používateľ P00 neabsolvoval magnetickú scénu — pre túto scénu nemá zaznamenanú žiadnu reláciu. Vo vesmírnej scéne však jeho údaje k dispozícii sú a vstupujú do príslušných analýz.
- Respondentka P12 má zaznamenané dve relácie — jednu v magnetickej a jednu vo vesmírnej scéne — ktoré obe vstupujú do analýzy.

6.6.2 Súbor dát

Systém zaznamenal počas testovania spolu **31 relácií** od 15 respondentov — 21 relácií od 10 žiakov 4. ročníka (P00–P09) a 10 relácií od 5 učiteliek (P10–P14). Z toho **14 relácií** prebehlo v magnetickej scéne a **17 relácií** vo vesmírnej scéne. Spolu sa zaznamenalo **732 udalostí**, z toho **351** v magnetických reláciách a **381** vo vesmírnych.

Najpočetnejšou zaznamenanou akciou bolo chytenie interaktívneho magnetu do ruky (FoundMagnet, 279 výskytov vrátane opakovaných), čo zodpovedá experimentálnej povahe magnetickej scény. Umiestnenia objektov do cieľových pozícií tvorili druhú najväčšiu skupinu — 56 správnych a 9 nesprávnych umiestnení predmetov v magnetickej scéne (CorrPlace, InCorrPlace) a 54 správnych a 45 nesprávnych umiestnení planét vo vesmírnej scéne. Pri skladaní súhvezdí systém zaznamenal 108 umiestnení hviezd (CorrPlace_) a 55 zobrazení nápovedy (ShowHint). Tieto záznamy tvoria objektívny základ pre analýzu v nasledujúcich podkapitolách.

6.6.3 Úspešné dokončenie scén

V zaznamenaných údajoch sa úspešné dokončenie scény manifestuje finálnou akciou `finalSucc`, kým ukončenie scenára bez splnenia úloh akciou `finalFail`. Tabuľka 6.11 zobrazuje, koľko respondentov z každej skupiny dokončilo jednotlivé scény podľa zaznamenaných údajov.

Z tabuľky je zrejmé, že *všetkých päť učiteliek úspešne dokončilo magnetickú scénu*, kým u žiakov to boli iba 2 z 10. Vo vesmírnej scéne sa žiadnemu žiakovi nepodarilo dosiahnuť úspešné ukončenie, kým 2 z 5 učiteliek áno. Tu treba uviesť dôležité obmedzenie: *záznam nezachytil všetky reálne dokončenia* — podľa pozorovania vyučujúcich pri testovaní úlohu reálne dokončili aj P05 a P08, ale finálna

Tabuľka 6.11: Úspešné dokončenie scén podľa zaznamenaných údajov

Skupina	Magnet		Vesmír	
	finalSucc	finalFail	finalSucc	finalFail
Žiaci (n=10)	2 (P06, P07)	0	0	1 (P01)
Učiteľky (n=5)	5 (všetky)	0	2 (P13, P14)	1 (P11)

akcia `finalSucc` sa v ich zázname nenachádza. Naša analýza dokončenia sa tým opiera o zaznamenané údaje, ktoré pôsobia ako spodný odhad.

6.6.4 Magnetická scéna — triedenie predmetov

V magnetickej scéne mali respondenti za úlohu roztriediť predmety do dvoch magnetických košov podľa toho, či sú magnetické. Tabuľka 6.12 zobrazuje úspešnosť umiestnenia podľa respondenta.

Tabuľka 6.12: Úspešnosť umiestnení v magnetickej scéne podľa respondenta

Respondent	Rola	Správne	Nesprávne	Spolu	Úspešnosť
P01	žiak	4	0	4	100 %
P02	žiak	1	1	2	50 %
P03	žiak	0	1	1	0 %
P04	žiak	2	0	2	100 %
P05	žiak	4	1	5	80 %
P06	žiak	5	1	6	83 %
P07	žiak	5	1	6	83 %
P08	žiak	4	0	4	100 %
P09	žiak	4	1	5	80 %
P10	učiteľka	5	1	6	83 %
P11	učiteľka	6	0	6	100 %
P12	učiteľka	5	1	6	83 %
P13	učiteľka	5	1	6	83 %
P14	učiteľka	6	0	6	100 %
Žiaci (spolu)		29	6	35	83 %
Učiteľky (spolu)		27	3	30	90 %

Pri magnetickej scéne dosiahli žiaci 83 % správnych umiestnení (29 z 35 pokusov), učiteľky 90 % (27 z 30). Tento rozdiel je malý, čo naznačuje, že koncept magnetizmu a samotná interakcia s VR rozhraním sú pre štvrtákov primerane navrhnuté. Päť respondentov dosiahlo 100 % úspešnosť (P01, P04, P08 zo žiakov

a P11, P14 z učiteliek).

Manipulácia s magnetom. Akcia FoundMagnet predstavuje chytenie magnetu do ruky a tým ukazuje, koľkokrát respondent magnet zobral. Zatiaľ čo žiaci urobili priemerne **21,3 chytení na reláciu**, učiteľky iba **17,4**. Extrémne hodnoty sa vyskytli u žiakov P05 (59 chytení) a P08 (57), ako aj u učiteliek P11 (29) a P13 (27). Naopak veľmi nízke číslo (1 chytenie) sa vyskytlo u P02, P07 a P14 — títo respondenti pravdepodobne magnet zobrali a držali ho po celý čas riešenia úlohy. Pomer chytení magnetu k umiestneniam predmetu bol u žiakov priemerne 5,5 : 1, u učiteliek 2,9 : 1. Žiaci s magnetom viac experimentovali, čo je pri prvom kontakte s VR prirodzené.

6.6.5 Vesmírna scéna — umiestňovanie planét

V prvej časti vesmírnej scény mali respondenti umiestniť 8 planét na 8 obežných dráh okolo Slnka. Tabuľka 6.13 zobrazuje úspešnosť umiestnenia podľa respondenta.

Tabuľka 6.13: Úspešnosť umiestnení planét podľa respondenta

Respondent	Rola	Správne	Nesprávne	Spolu	Úspešnosť
P00	žiak	0	4	4	0 %
P01	žiak	1	2	3	33 %
P02	žiak	4	3	7	57 %
P03	žiak	0	8	8	0 %
P04	žiak	1	1	2	50 %
P05	žiak	3	6	9	33 %
P06	žiak	2	2	4	50 %
P07	žiak	1	4	5	20 %
P08	žiak	0	7	7	0 %
P09	žiak	4	4	8	50 %
P10	učiteľka	6	3	9	67 %
P11	učiteľka	8	0	8	100 %
P12	učiteľka	8	1	9	89 %
P13	učiteľka	8	0	8	100 %
P14	učiteľka	8	0	8	100 %
Žiaci (spolu)		16	41	57	28 %
Učiteľky (spolu)		38	4	42	90 %

Najsilnejší nález celého testovania je výrazný rozdiel medzi žiakmi a učiteľ-

kami pri planétach. Žiaci dosiahli iba **28 % správnych umiestnení**, kým učiteľky **90 %** — rozdiel 62 percentuálnych bodov. Traja respondenti zo žiakov (P00, P03, P08) mali **0 % správnych umiestnení**, kým tri z učiteliek (P11, P13, P14) mali **100 %**. Tento dramatický rozdiel jasne ukazuje, že úloha umiestnenia planét vyžaduje predznalosti o slnečnej sústave, ktoré žiaci 4. ročníka ešte nemajú dostatočne osvojené.

Učiace sa správanie. V zaznamenaných údajoch sa rozlišujú dve špecifické akcie: InCorrPlanetTake (zobratie planéty z nesprávnej dráhy, teda oprava chyby) a CorrPlanetTake (zobratie planéty zo správnej dráhy, teda rozhodenie správneho stavu). Žiaci urobili **21 opráv** a iba **5 chybných rozhodnutí**, čo dáva pomer 4,2:1. Učiteľky mali 2 opravy a 1 rozhodenie (pomer 2:1, avšak na malom vzorku). Tento výsledok naznačuje, že *scenár úspešne podporuje samokorektívne učenie* — žiaci po nesprávnom umiestnení často zistia chybu a planétu znovu zoberú na opravu.

Najproblematickejšie planéty. Pri analýze cieľových objektov pri chybných akciách (InCorrPlace a InCorrPlanetTake) sme sledovali, ktoré planéty boli najčastejšie zamieňané. Najviac chýb sa vyskytlo pri *Merkúre* (6) a *Uráne* (5), nasledované Venušou a Neptúnom (po 4). Naopak pri *Zemi* sa nevyskytla žiadna chyba a pri Marse iba dve — ide o planéty, ktoré sú žiakom najznámejšie z bežného života. Vzhľadom na malý počet zaznamenaných chýb na jednotlivú planétu však tieto rozdiely interpretujeme len orientačne.

6.6.6 Vesmírna scéna — skladanie súhvezdia

V druhej časti vesmírnej scény mali respondenti zostaviť súhvezdie z hviezd a spojíc. Respondent si mohol vybrať medzi Veľkým vozom a Oriónom. Skladanie súhvezdia bolo navrhnuté tak, že chyba pri umiestnení hviezdy nie je možná — systém umožní hviezdu vložiť iba do správnej pozície (CorrPlace_).

Najvýraznejším pozorovaním je, že **iba 5 z 10 žiakov** sa vôbec dostalo k umiestňovaniu hviezd. Keďže obe úlohy vesmírnej scény sú navzájom nezávislé a žiak si môže zvoliť, ktorou začne, ostatní žiaci buď úlohu skladania súhvezdia nezvolili, alebo ju nestihli v rámci časovo obmedzenej relácie. Naopak, *všetkých 5 učiteliek umiestnilo hviezdy* a spolu ich umiestnili výrazne viac (69 oproti 39 u žiakov).

Pri voľbe súhvezdia výrazne dominoval Veľký voz — 8 z 10 žiakov si ho zvolilo, a 4 z 5 učiteliek. Iba P06 (žiak) a P14 (učiteľka) si zvolili Orión, a P02 vyskúšal obe (mal viacero relácií). Tento výsledok pravdepodobne reflektuje známosť Veľ-

Tabuľka 6.14: Účasť na úlohe skladania súhvezdia

Respondent	Rola	Hviezd	Úsp. spoj.	Neúsp. spoj.	Voľba
P00	žiak	0	0	1	Veľký voz
P01	žiak	0	0	2	Veľký voz
P02	žiak	20	1	0	V. voz, Orión
P03	žiak	7	0	0	Veľký voz
P04	žiak	0	0	0	—
P05	žiak	0	0	1	Veľký voz
P06	žiak	0	0	0	Orión
P07	žiak	1	0	0	Veľký voz
P08	žiak	5	0	0	Veľký voz
P09	žiak	6	0	0	Veľký voz
P10	učiteľka	9	0	0	Veľký voz
P11	učiteľka	7	0	1	Veľký voz
P12	učiteľka	22	0	0	Veľký voz
P13	učiteľka	16	1	0	Veľký voz
P14	učiteľka	15	1	0	Orión

kého vozu z bežného prostredia (učivo prírodovedy a astronómie pre základné školy).

6.6.7 Využitie nápoved

Vo vesmírnej scéne mali respondenti k dispozícii dve tlačidlá nápovedy určené pre úlohu skladania súhvezdia — `buttonHint1` pre Veľký voz a `buttonHint2` pre Orión. Obe tlačidlá fungujú rovnako, vizuálne zvýraznia tvar príslušného súhvezdia na oblohe. Akcia `ShowHint` sa odpáli pri zobrazení nápovedy. Pre úlohu umiestňovania planét nie sú v aktuálnej verzii prostredia dostupné žiadne nápovedy.

Žiaci spolu vyžiadali **39 nápoved**, učiteľky **16**. Najviac nápoved si vyžiadali žiak P01 (10 nápoved v jednej relácii), čo poukazuje na ťažkosti pri orientácii a hľadaní zobrazenej nápovedy na oblohe. Naopak učiteľky P12 a P13 si vyžiadali iba 2 nápovedy.

6.6.8 Trvanie relácií

Trvanie jednotlivých relácií bolo určené ako rozdiel medzi časom ukončenia a časom začatia relácie zaznamenaným systémom. Tabuľka 6.15 uvádza prehľad podľa scény a skupiny.

Tabuľka 6.15: Trvanie relácií podľa scény a skupiny (v minútach)

Scéna	Skupina	N	Priemer	Medián	Min	Max
Magnet	žiaci	9	15,7	11,9	2,0	28,6
Magnet	učiteľky	5	6,4	5,1	2,6	13,1
Vesmír	žiaci	12	12,3	12,8	2,1	20,5
Vesmír	učiteľky	5	25,0	23,3	15,8	39,6

Pri magnetickej scéne strávili žiaci priemerne výrazne dlhší čas (15,7 min) než učiteľky (6,4 min). To je v súlade s tým, že u žiakov sa magnet stal predmetom experimentovania. Pri vesmírnej scéne sa pomery prevrátili — učiteľky strávili priemerne 25 minút, kým žiaci iba 12,3 minúty. Vysvetľujeme to tým, že učiteľky úlohu skutočne dokončili (vrátane fázy súhvezdia), zatiaľ čo žiaci sa skôr alebo neskôr zasekli alebo bola relácia z časových dôvodov prerušená.

6.6.9 Súhrn nálezov zo zaznamenaných údajov

Analýza zaznamenaných údajov priniesla nasledovné zistenia:

1. **Magnetická scéna funguje pre obe vekové skupiny.** Žiaci aj učiteľky dosiahli porovnateľnú úspešnosť (83 % a 90 %). Koncept magnetizmu je deťom blízky a interakcia s VR im pri tom nerobí problém.
2. **Vesmírna scéna odhaľuje výraznú vedomostnú medzeru pri planétach.** Žiaci dosiahli iba 28 % správnych umiestnení (oproti 90 % u učiteľiek). Úloha v aktuálnej forme presahuje vedomostnú úroveň 4. ročníka ZŠ.
3. **Žiaci aktívne opravujú vlastné chyby.** Pomer 4,2 : 1 medzi opravami a nesprávnymi rozhodnutiami ukazuje, že scenár podporuje iteratívne učenie z chýb.
4. **Polovica žiakov sa nedostala k súhvezdiu.** Z 10 žiakov umiestnilo hviezdy súhvezdia iba 5, kým z učiteľiek všetkých 5. Keďže úlohy sú nezávislé, časť žiakov úlohu súhvezdia nezvolila alebo ju nestihla v rámci časovo obmedzenej relácie.
5. **Pri planétach najčastejšie chybovali na Merkúre a Uráne.** Pri umiestňovaní planét sa najčastejšie chybovalo pri Merkúre (6) a Uráne (5), pri Zemi sa nevyskytla žiadna chyba. Vzhľadom na nízke počty však tieto rozdiely interpretujeme orientačne.

6. **Veľký voz výrazne dominuje voľbu súhvezdia.** Z 15 respondentov si 12 vybralo Veľký voz a iba 3 Orión. Tento pomer reflektuje známosť Veľkého vozu z kurikula prírodovedy.

Tieto zistenia korelujú s výsledkami z dotazníka (kap. 6.5) a poskytujú konkrétne odporúčania pre ďalší vývoj prostredia, ktorým sa venuje diskusia (kap. 6.7).

6.7 Diskusia výsledkov

Limity vyhodnotenia.

Pri interpretácii výsledkov je potrebné brať do úvahy nasledovné metodologické obmedzenia:

- **Veľkosť vzorky** – pätnásť respondentov je metodicky obmedzujúca vzorka pre štatistické zovšeobecnenia. Výsledky preto slúžia primárne na orientačnú spätnú väzbu od cieľovej skupiny, nie na absolútne hodnotenie použiteľnosti systému.
- **Vlastný preklad detskej SUS adaptácie** – slovenský preklad otázok detskej adaptácie SUS bol vytvorený v rámci tejto práce a nebol nezávisle validovaný (napríklad spätným prekladom).
- **Validačný kontext detskej SUS adaptácie** – pôvodná validácia detskej adaptácie SUS podľa Putnam a kol. bola vykonaná v kontexte mobilných herných aplikácií, nie virtuálnej reality. Otázky boli kontextovo prispôsobené pre VR výučbové prostredie, no ich validita v tomto novom kontexte nebola samostatne overená.
- **Asistované vyplnenie** – aj keď tento prístup zabezpečuje porozumenie otázkam, sprostredkúva interpretáciu cez dospelého, čo je dvojsečný metodologický kompromis.
- **Prvý kontakt s VR** – pre žiakov išlo o prvú skúsenosť s technológiou virtuálnej reality, čo môže nadhodnocovať pozitívne hodnotenia z dôvodu novosti zážitku.

6.8 Zhrnutie vyhodnotenia

Testovanie prostredia LirkisEduVePn so žiakmi 4. ročníka a piatimi učiteľkami prinieslo dva paralelné pohľady na použiteľnosť aplikácie. Žiaci absolvovali ve-

kovo upravené verzie dotazníkov so zjednodušenými škálami, učiteľky plné verzie pre dospelých. Priame číselné porovnanie medzi skupinami preto nie je metodicky korektné a v nasledujúcom zhrnutí ich uvádzame samostatne.

Žiaci. Priemerné SUS skóre 75,0 a doplnkové otázky Q11 – Q13 s priemerom 4,73 z 5 ukazujú, že prostredie bolo pre deti atraktívne a väčšina by ho chcela používať opakovane. Magnetická scéna fungovala pre obe vekové skupiny — žiaci dosiahli 83 % správnych umiestnení a zaznamenané údaje preukázali výrazné experimentovanie s magnetom (priemerne 21,3 chytení magnetu na reláciu), čo zodpovedá prirodzenému objavovaniu princípu úlohy. Vesmírna scéna odhalila ostré obmedzenia. Žiaci dosiahli iba 28 % správnych umiestnení planét, traja respondenti (P00, P03, P08) nemali ani jedno správne umiestnenie a iba polovica žiakov vôbec umiestnila hviezdy súhvezdia. Tento rozdiel jasne ukazuje, že planetárna časť scenára presahuje vedomostnú úroveň 4. ročníka — nie z hľadiska ovládania VR, ale z hľadiska predpokladaných znalostí o slnečnej sústave. Nízka vnímaná frustrácia v NASA-TLX (2,50 na škále, kde 3 = vôbec) zároveň ukazuje, že žiakov ani náročnejšia planetárna úloha citelne nefrustrovala. Pozitívnym nálezom bol pomer 4,2 : 1 medzi opravami nesprávnych umiestnení (InCorrPlanetTake) a rozhodovaním správneho stavu (CorrPlanetTake). Scenár teda žiakom umožňoval učiť sa z chýb iteratívnymi pokusmi, čo zodpovedá konštruktivistickému prístupu k výučbe. Symptómy kybernetickej nevoľnosti boli prevažne mierne a vizuálneho charakteru (námaha očí, rozmazané videnie, točenie hlavy — každý u 40 % žiakov). Žiadny žiak nehlásil vážny symptóm a relácie trvali v bezpečnom časovom pásme. V IPQ žiaci hodnotili vysokým skóre priestorovú prítomnosť (pocit „bol som tam“ priemer 4,50 z 5), ale nízkym skóre realističnosť prostredia, čo zodpovedá štylizovanej grafike LirkisEduVePn.

Učiteľky. Priemerné SUS skóre 78,0 a väčšinou nízka až stredne nízka záťaž v NASA-TLX (TLX skóre 26,3) ukázali, že učiteľky systém zvládli aj po prvom kontakte. Na rozdiel od žiakov dokončili planetárnu aj úlohu so súhvezdím vo väčšine prípadov — pri planétach dosiahli 90 % správnych umiestnení, čo potvrdzuje, že rozdiel oproti žiakom je daný vedomostnou medzerou, nie problémom s aplikáciou. Výrazný outlier sa objavil u učiteľky P11 (SUS 57,5, TLX 53,3), ktorá v rozhovore uviedla technické ťažkosti s headsetom a kontrolérmi. Tento prípad ukazuje, že úspešné nasadenie systému v škole bude vyžadovať dôsledné nastavenie hardvéru pred reláciou. Symptómy u učiteliek boli mierne, ale výskyt vertiga (4/5 respondentiek) a námahy očí (3/5) bol vyšší než u žiakov. Vysvetľu-

jeme to dlhším priemerným časom stráveným vo vesmírnej scéne (25 min oproti 12 min u žiakov), keďže učiteľky úlohu dokončili namiesto prerušenia. V IPQ učiteľky dosiahli vysoké skóre v dimenziách všeobecnej a priestorovej prítomnosti ($G = 6,00$; $SP = 6,24$ zo 7), čo zodpovedá vzoru typickému pre VR aplikácie zamerané na edukačný obsah skôr než na fotorealizmus.

Celkový pohľad a odporúčania. Prostredie funguje najlepšie pri úlohách založených na priamej manipulácii a experimentovaní, čo potvrdzuje magnetická scéna. Hlavnou prekážkou pre cieľovú skupinu je planetárna úloha, ktorá vyžaduje predznanosti, ktorými žiaci 4. ročníka ešte nedisponujú. Na základe týchto zistení formulujeme štyri konkrétne odporúčania pre ďalší vývoj:

1. **Pridať časovo obmedzenú nápovedu s poradím planét.** Ak žiak po určitom čase neumiestni planétu správne, scenár automaticky zobrazí krátku nápovedu s poradím planét od Slnka. Tým sa úloha posunie od testovania predznaností k učeniu sa nového obsahu, čo lepšie zodpovedá pedagogickému cieľu prostredia.
2. **Zlepšiť navigáciu k nápovede pre súhvezdie.** Tlačidlo na vyvolanie nápovedy je v aktuálnej scéne pre žiakov ťažko nájditeľné a samotná nápoveda sa zobrazuje vo forme súhvezdia na oblohe, ktorú musí žiak v rozľahlom vesmírnom prostredí samostatne lokalizovať. Pre cieľovú skupinu by bolo vhodné tlačidlo nápovedy vizuálne zvýrazniť a po jeho aktivácii žiaka navigovať k zobrazenej nápovede — napríklad smerovou šípku alebo dočasným zvýraznením oblasti oblohy s nápovedou.
3. **Vylepšiť mechaniku uchopovania objektov.** Časť žiakov a jedna učiteľka mali ťažkosti s manipuláciou vo virtuálnom prostredí, predovšetkým preto, že si k sebe nedokázali pritiahnuť vzdialenejšie objekty. Doplnenie funkcie pre pritiahnutie objektu k hráčovi (napr. pomocou *ray-casting* interakcie alebo *attractor* mechaniky) by výrazne uľahčilo prácu so scénou používateľom bez predchádzajúcej VR skúsenosti.

Implementácia týchto úprav by mala posunúť aplikáciu z roviny „zaujímavý experiment“ do roviny použiteľného výučbového nástroja pre 4. ročník ZŠ.

7 Záver

Cieľom tejto bakalárskej práce bolo rozšíriť existujúcu vzdelávaciu platformu LirisEduVePn o nové scenárom riadené výučbové prostredie pre žiakov 4. ročníka základnej školy zamerané na tematický celok *Neživá príroda a prírodné javy*. Práca pokrýva celý cyklus od pedagogickej analýzy a systematického prehľadu literatúry, cez návrh a implementáciu dvoch interaktívnych scén, až po overovacie testovanie v reálnom školskom prostredí.

Splnenie pedagogickej časti zadania. V kapitole 3 sme uskutočnili systematický prehľad literatúry v databázach Web of Science a Scopus, na základe ktorého sme identifikovali kľúčové princípy efektívneho návrhu výučbových prostredí vo VR — vedomé obmedzenie imerzie na úroveň nezaťažujúcu pozornosť žiaka, interaktívny prístup s aktívnym zapojením, multimodálna spätná väzba a voliteľný systém nápodvy vyrovňavajúci rozdiely v predznanostiach. Tieto princípy sme následne premietli do návrhu konkrétnych scenárov v spolupráci so Základnou školou Gemerská Hôrka, ktorá poskytla expertné posúdenie vekovej primeranosti obsahu.

Splnenie technickej časti zadania. Implementovali sme dve samostatné scény — magnetickú a vesmírnu — v softvérovom rámci A-Frame s kombináciou TypeScript komponentov a JavaScript A-Frame komponentov. Postupnosť úloh a podmienky ich vyhodnotenia sú v oboch scénach formálne riadené Petriho sieťami vytvorenými v nástroji CPN Tools (popis sietí v podkap. 4.6). Magnetická scéna obsahuje úvodné hľadanie magnetu pomocou navádzacej guľôčky a následné triedenie šiestich predmetov podľa ich magnetických vlastností. Vesmírna scéna pozostáva z dvoch po sebe nadväzujúcich úloh — umiestňovania ôsmich planét slnečnej sústavy na obežné dráhy a skladania súhvezdia z voliteľnej dvojice (Veľký voz alebo Orión). Všetky scény podporujú tri režimy ovládania: VR headset, počítač a mobilné zariadenie.

Hlavné zistenia z testovania. Overovacie testovanie sa uskutočnilo na Základnej škole Gemerská Hôrka s desiatimi žiakmi 4. ročníka a piatimi učiteľkami. Priemerné SUS skóre žiakov dosiahlo **75,0** a učiteľiek **78,0**, čo v oboch prípadoch zodpovedá hodnoteniu „dobré“ a prevyšuje literárny priemer 68 bodov. Doplnkové otázky merajúce zábavnosť a motiváciu detí dosiahli priemer **4,73 z 5**, pričom sedem z desiatich žiakov označilo všetky tri otázky maximálnou hodnotou. Analýza zaznamenaných údajov odhalila, že magnetická scéna funguje pre obe vekové skupiny porovnateľne (83 % a 90 % správnych umiestnení), no planetárna úloha presahuje vedomostnú úroveň 4. ročníka — žiaci dosiahli iba 28 % správnych umiestnení oproti 90 % u učiteľiek. Pozitívnym nálezom bol pomer 4,2:1 medzi opravami chýb a rozhodnutím správneho stavu, čo dokladá, že scenár podporuje iteratívne učenie z chýb. Symptómy kybernetickej nevoľnosti boli v oboch skupinách prevažne mierne a vizuálneho charakteru, žiadny respondent nehlásil vážny symptóm.

Prínosy práce. Hlavným prínosom práce je rozšírenie platformy LirkisEduVePn o prvé výučbové prostredie cielené priamo na 1. stupeň základnej školy — doposiaľ boli scenáre v projekte zamerané prevažne na starších žiakov a dospelých. Súčasťou prínosu sú aj nové A-Frame komponenty (navádzacia guľôčka proximity-sensor, náhodný výber predmetov random-pool-picker, aktivácia objektov na základe udalostí activate-object-on-event a ďalšie), ktoré sú znovupoužiteľné aj v budúcich scenároch projektu. Z výskumného hľadiska práca prináša aj prvé objektívne dáta zaznamenané počas testovania platformy LirkisEduVePn s touto vekovou skupinou, čo vytvára referenčný bod pre ďalšie štúdie v rámci projektu KEGA 048 TUKE-4/2022.

Limity a budúca práca. Pri interpretácii výsledkov je potrebné zohľadniť obmedzenú veľkosť testovacej vzorky ($n=10$ žiakov, $n=5$ učiteľiek), vlastný preklad detskej adaptácie SUS bez nezávislej validácie spätným prekladom a prvý kontakt žiakov s VR technológiou, ktorý mohol nadhodnotiť pozitívne hodnotenia z dôvodu novosti zážitku. Na základe zistení z testovania formulujeme štyri konkrétne smery pre ďalší vývoj:

1. **Pridať časovo obmedzenú nápovedu s poradím planét** — v aktuálnej verzii nie sú pri planetárnej úlohe dostupné žiadne nápovedy, čo úlohu pre žiakov bez predznalostí robí takmer neriešiteľnou.
2. **Zlepšiť navigáciu k nápovede pre súhvezdie** — tlačidlo na vyvolanie nápovedy je v aktuálnej scéne ťažko nájditeľné a samotná nápoveda na rozľahlej

oblohe sa lokalizuje ťažko.

3. **Vylepšiť mechaniku uchopovania objektov** — doplniť funkciu pre pritiahnutie vzdialeného objektu k hráčovi pomocou *ray-casting* interakcie alebo *attractor* mechaniky.

Implementácia týchto úprav, spolu s rozšírením testovania na väčšiu vzorku žiakov a viacero škôl, by mala posunúť výučbové prostredie z roviny prototypu overujúceho koncepciu do roviny použiteľného výučbového nástroja pre prvý stupeň základnej školy. Práca tak predstavuje nielen ukončený samostatný projekt, ale aj otvorené pole pre ďalšie pokračovanie v rámci nadväzujúcich záverečných prác a výskumných aktivít na Katedre počítačov a informatiky FEI TUKE.

Zoznam použitej literatúry

1. KAŠELA, Adam. *Výučbové scenáre v rozšírenej realite využívajúce procesné grafy* [online]. Košice, 2022 [cit. 2025-11-02]. Dostupné z: https://hron.feit.tuke.sk/~korecko/zavPr/DP_Kaseľa.pdf. Diplomová práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky. 65 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.
2. PISARČÍK, Lukáš. *Scenárom riadené výučbové prostredia vo virtuálnej realite* [online]. Košice, 2023 [cit. 2025-11-02]. Dostupné z: https://hron.feit.tuke.sk/~korecko/zavPr/BP_Pisarcik.pdf. Bakalárska práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky. 47 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.
3. DEMIANENKO, Dmytro. *Spracovanie údajov z virtuálnych výučbových prostredí* [online]. Košice, 2023 [cit. 2025-11-02]. Dostupné z: https://hron.feit.tuke.sk/~korecko/zavPr/BP_Demianenko.pdf. Bakalárska práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky. 65 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.
4. SAKÁČ, Michal. *Virtuálne výučbové prostredie pre skladanie počítačov* [online]. Košice, 2025 [cit. 2025-11-02]. Dostupné z: https://hron.feit.tuke.sk/~korecko/zavPr/BP_Sakac.pdf. Bakalárska práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky. 58 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.
5. ZSIGÓ, Kristián. *Rozšírenia a ukážková scéna pre systém virtuálnych výučbových prostredí* [online]. Košice, 2025 [cit. 2025-11-02]. Dostupné z: https://hron.feit.tuke.sk/~korecko/zavPr/DP_Zsigo.pdf. Diplomová práca. Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky. 75 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.
6. MAKRANSKY, Guido; MAYER, Richard E. Benefits of Taking a Virtual Field Trip in Immersive Virtual Reality: Evidence for the Immersion Principle in

- Multimedia Learning. *Educational Psychology Review*. 2022, roč. 34, č. 3, s. 1771–1798. ISSN 1040-726X. Dostupné z DOI: 10.1007/s10648-022-09675-4.
7. SCHRADER, Claudia; BASTIAENS, Theo J. The influence of virtual presence: Effects on experienced cognitive load and learning outcomes in educational computer games. *Computers in Human Behavior*. 2012, roč. 28, č. 2, s. 648–658. ISSN 0747-5632. Dostupné z DOI: <https://doi.org/10.1016/j.chb.2011.11.011>.
8. WON, Mihye; UNGU, Dewi Ayu Kencana; MATOVU, Henry; TREAGUST, David F.; TSAI, Chin-Chung; PARK, Jungho; MOCERINO, Mauro; TASKER, Roy. Diverse approaches to learning with immersive Virtual Reality identified from a systematic review. *Computers & Education*. 2023, roč. 195, s. 104701. ISSN 0360-1315. Dostupné z DOI: <https://doi.org/10.1016/j.compedu.2022.104701>.
9. BECK, Dennis; MORGADO, Leonel; O'SHEA, Patrick. Educational Practices and Strategies With Immersive Learning Environments: Mapping of Reviews for Using the Metaverse. *IEEE Transactions on Learning Technologies*. 2024, roč. 17, s. 319–341. ISSN 1939-1382. Dostupné z DOI: 10.1109/TLT.2023.3243946.
10. SHIN, Kwang-Seong; CHO, Chungyeon; RYU, Ji Hyun; JO, Dongsik. Exploring the Perception of the Effect of Three-Dimensional Interaction Feedback Types on Immersive Virtual Reality Education. *Electronics*. 2023, roč. 12, č. 21. ISSN 2079-9292. Dostupné z DOI: 10.3390/electronics12214414.
11. TAÇGIN, Zeynep. The perceived effectiveness regarding Immersive Virtual Reality learning environments changes by the prior knowledge of learners. *Education and Information Technologies*. 2020, roč. 25, č. 4, s. 2791–2809. ISSN 1360-2357. Dostupné z DOI: 10.1007/s10639-019-10088-0.
12. LEE, Silvia Wen-Yu; HSU, Ying-Tai; CHENG, Kun-Hung. Do curious students learn more science in an immersive virtual reality environment? Exploring the impact of advance organizers and epistemic curiosity. *Computers & Education*. 2022, roč. 182. ISSN 0360-1315. Dostupné z DOI: 10.1016/j.compedu.2022.104456.
13. BROOKE, John. SUS: A Quick and Dirty Usability Scale. In: JORDAN, Patrick W.; THOMAS, Bruce; WEERDMEESTER, Bernard A.; MCCLELLAND, Ian L. (ed.). *Usability Evaluation in Industry*. CRC Press, 1996, s. 207–212. ISBN 9780429157011. Dostupné z DOI: 10.1201/9781498710411-35.

14. 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).
15. KENNEDY, Robert S.; LANE, Norman E.; BERBAUM, Kevin S.; LILIENTHAL, Michael G. Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness. *The International Journal of Aviation Psychology*. 1993, roč. 3, č. 3, s. 203–220. Dostupné z doi: [10.1207/s15327108ijap0303_3](https://doi.org/10.1207/s15327108ijap0303_3).
16. SCHUBERT, Thomas; FRIEDMANN, Frank; REGENBRECHT, Holger. The Experience of Presence: Factor Analytic Insights. *Presence: Teleoperators and Virtual Environments*. 2001, roč. 10, č. 3, s. 266–281. Dostupné z doi: [10.1162/105474601300343603](https://doi.org/10.1162/105474601300343603).
17. KOREČKO, Štefan; MATHES, Jaroslav; SOBOTA, Branislav. K slovenskemu prekladu dotazníkov na meranie používateľskej skúsenosti vo virtuálnej realite. In: *Electrical Engineering and Informatics 16*. Košice, Slovensko: Technická univerzita v Košiciach, 2025, s. 582–586. ISBN 978-80-553-4854-4. Dostupné tiež z: https://eei.fei.tuke.sk/wp-content/uploads/2025/07/EEI_16.pdf.
18. PUTNAM, Cynthia; PUTHENMADOM, Melisa; CUERDO, Marjorie Ann; WANG, Wanshu; PAUL, Nathaniel. Adaptation of the System Usability Scale for User Testing with Children. In: *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. Honolulu, HI, USA: Association for Computing Machinery, 2020, s. 1–7. CHI EA '20. ISBN 9781450368193. Dostupné z doi: [10.1145/3334480.3382840](https://doi.org/10.1145/3334480.3382840).
19. BANGOR, Aaron; KORTUM, Philip; MILLER, James. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *Journal of Usability Studies*. 2009, roč. 4, č. 3, s. 114–123. ISSN 1931-3357. Dostupné tiež z: https://uxpajournal.org/wp-content/uploads/sites/7/pdf/JUS_Bangor_May2009.pdf.
20. SAURO, Jeff. *A Practical Guide to the System Usability Scale: Background, Benchmarks and Best Practices*. Denver, CO: Measuring Usability LLC, 2011. ISBN 978-1461062707. Dostupné tiež z: <https://measuringu.com/sus/>.

21. KIM, Hyun K.; PARK, Jaehyun; CHOI, Yeongcheol; CHOE, Mungyeong. Virtual reality sickness questionnaire (VRSQ): Motion sickness measurement index in a virtual reality environment. *Applied Ergonomics*. 2018, roč. 69, s. 66–73. ISSN 0003-6870. Dostupné z DOI: <https://doi.org/10.1016/j.apergo.2017.12.016>.

Zoznam príloh

Príloha A Použité dotazníky

Príloha B Používateľská príručka

Príloha C Systémová príručka

Príloha D CD médium – záverečná práca v elektronickej podobe:

- doc – záverečná práca a prílohy
- tex – zdrojové súbory záverečnej práce
- src – zdrojové kódy vytvoreného riešenia
- lib – automatizovaná dokumentácie aplikácie
- bin – Docker obrazy aplikácie

A Použité dotazníky

V tejto prílohe sú uvedené plné verzie oboch dotazníkov použitých pri overovaní testovaní (kap. 6).

Dotazník pre hodnotenie VR aplikácie

Verzia pre žiakov 4. ročníka základnej školy

Identifikátor: _____

Vek: _____

Trieda: _____

Dátum testovania: _____

Ahoj! Práve si si vyskúšal/a VR aplikáciu. Teraz by sme veľmi radi vedeli, ako sa ti to páčilo a čo si si o tom myslel/a. Dotazník má štyri okruhy. Pri každej otázke vyber tú odpoveď, ktorá najlepšie hovorí o tom, ako si sa cítil/a. Nie sú tu žiadne dobré ani zlé odpovede. Ak niečomu nerozumieš, opýtaj sa pani učiteľky alebo pána učiteľa.

Okruh 1 — Ako sa ti používala VR aplikácia

Označ, do akej miery súhlasíš s nasledujúcimi výrokmí.

1. Keby som mal/a túto VR aplikáciu doma, myslím, že by som v nej rád/rada plnil/a úlohy často.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

2. Veľakrát som bol/a zmätený/á, keď som plnil/a úlohy v tejto VR aplikácii.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

3. Myslel/a som si, že VR aplikácia sa dala ľahko používať.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

4. Potreboval/a by som pomoc dospelého, aby som mohol/mohla pokračovať v plnení úloh.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

5. Vždy som mal/a pocit, že viem, čo mám robiť ďalej, keď som plnil/a úlohy v tejto VR aplikácii.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

6. Niektoré veci, ktoré som musel/a robiť pri plnení úloh, mi nedávali zmysel.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

7. Myslím si, že väčšina mojich kamarátov by sa mohla veľmi rýchlo naučiť používať túto VR aplikáciu.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

8. Niektoré veci, ktoré som musel/a robiť pri plnení úloh, boli akoby divné.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

9. Cítil/a som sa sebavedomo, keď som plnil/a úlohy v tejto VR aplikácii.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

10. Musel/a som sa naučiť veľa vecí, kým som vedel/a úlohy v tejto VR aplikácii dobre plniť.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

11. Plniť úlohy v tejto VR aplikácii ma veľmi bavilo.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

12. Keby sme mali viac času, pokračoval/a by som v plnení úloh.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

13. Plánujem povedať svojim kamarátom o tejto VR aplikácii.

				
☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

Okruh 2 — Aké náročné bolo plnenie úloh

Pri každej otázke označ, ako ti to pripadalo.

1. Ako vedomostne náročné bolo plnenie úloh?

☐

Náročné

☐

Priemerne náročné

☐

Nenáročné

2. Ako manuálne náročné bolo plnenie úloh?

☐

Náročné

☐

Priemerne náročné

☐

Nenáročné

3. Aký bol časový nátlak na plnenie úloh?

☐

Veľmi veľký tlak

☐

Priemerný tlak

☐

Žiadny časový tlak

4. Ako úspešní ste boli pri plnení úloh?

☐

Veľmi úspešní

☐

Priemerne úspešní

☐

Málo úspešní

5. Ako ťažko ste museli pracovať, aby ste dosiahli to, čo sa vám podarilo?

☐

Veľmi ťažko

☐

Priemerne ťažko

☐

Ľahko

6. Ako veľmi ste boli neistí, znechutení, podráždení, vystresovaní a otrávení?

☐

Veľmi

☐

Priemerne

☐

Vôbec

Okruh 3 — Ako si sa cítil/a po VR

Vyznač krížikom, ktoré z nasledujúcich pocitov si pociťoval/a počas alebo po používaní VR. Ak žiadne z nich neplatia, nemusíš nič označiť.

Symptóm	Pociťoval/a som
Únava	☐
Námaha očí	☐
Ťažkosti so zaostrením	☐
Boleť hlavy	☐
Tlak v hlave	☐
Rozmazané videnie	☐
Závraty pri zatvorených očiach	☐
Točenie hlavy	☐

Okruh 4 — Pocit prítomnosti vo virtuálnom svete

Pri každej otázke označ, do akej miery s ňou súhlasíš.

1. V počítačom generovanom svete som mal/a pocit, ako keby som „bol/a tam“.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

2. Mal/a som pocit, že ma virtuálny svet obklopuje.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

3. Mal/a som pocit, že vnímam (vidím) iba obrázky.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

4. Mal/a som pocit, že naozaj konám vo virtuálnom svete a nie, že ho zvonku ovládam.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

5. Cítil/a som sa vo virtuálnom svete prítomný/á.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

6. Do akej miery si vnímal/a reálny svet okolo seba (teplota, zvuky, iní ľudia, ...) počas tvojej aktivity vo virtuálnom svete?

☐	☐	☐
Úplne vnímal/a	Priemerne vnímal/a	Vôbec nevnímal/a

7. Ako veľmi reálny ti pripadal virtuálny svet?

☐	☐	☐
Úplne reálny	Neviem sa rozhodnúť	Nepripadal mi vôbec reálny

8. Ako veľmi sa zážitok vo virtuálnom svete zhodoval so zážitkom v reálnom svete?

☐	☐	☐
Vôbec sa nezhodoval	Trocha sa zhodoval	Úplne sa zhodoval

9. Virtuálny svet sa mi zdal skutočnejší ako ten reálny.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Skôr nesúhlasím	Neviem	Skôr súhlasím	Úplne súhlasím

Ďakujeme ti za vyplnenie dotazníka!

Okruh 1 — vlastný slovenský preklad detskej adaptácie SUS podľa Putnam et al. (2020). Adaptation of the System Usability Scale for User Testing with Children. CHI EA '20, doi:10.1145/3334480.3382840.

Okruhy 2 – 4 — vychádzajú zo slovenských prekladov publikovaných v: Korečko, Š., Mathes, J., Sobota, B. (2025). K slovenskému prekladu dotazníkov na meranie používateľskej skúsenosti vo virtuálnej realite. EEI 16, str. 582–586. Otázky boli redukované expertným posúdením pedagogickým zborom.

Pôvodné nástroje: Brooke (1996) · Hart & Staveland (1988) · Kennedy et al. (1993) / Kim et al. (2018) · Schubert et al. (2001).

Dotazník pre hodnotenie VR aplikácie

Štandardizovaná verzia pre dospelých respondentov

Identifikátor
respondenta: _____

Vek: _____

Dátum testovania: _____

Vyplnením tohto dotazníka prispejete k zhodnoteniu používateľskej skúsenosti s VR aplikáciou, ktorú ste práve testovali. Dotazník pozostáva zo štyroch okruhov, ktoré sa zameriavajú na rôzne aspekty zážitku — použiteľnosť systému, záťaž pri plnení úloh, fyzické symptómy a pocit prítomnosti vo virtuálnom svete. Pri každej otázke označte tú odpoveď, ktorá najlepšie vystihuje váš zážitok. Neexistujú správne ani nesprávne odpovede — odpovedajte podľa toho, ako ste to skutočne zažili.

Okruh 1 — Použiteľnosť systému (SUS)

Označte, do akej miery súhlasíte s nasledujúcimi výrokmi.

1. Myslím si, že by som rád(a) používal(a) tento systém často.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Nesúhlasím	Nie som si istý(á)	Súhlasím	Úplne súhlasím

2. Systém mi prišiel zbytočne komplikovaný.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Nesúhlasím	Nie som si istý(á)	Súhlasím	Úplne súhlasím

3. Systém sa mi zdal ľahko použiteľný.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Nesúhlasím	Nie som si istý(á)	Súhlasím	Úplne súhlasím

4. Myslím si, že by som potreboval(a) technickú podporu, aby som vedel(a) používať tento systém.

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Nesúhlasím	Nie som si istý(á)	Súhlasím	Úplne súhlasím

5. Funkcionality systému sú podľa mňa dobre zapracované (integrované).

☐	☐	☐	☐	☐
Vôbec nesúhlasím	Nesúhlasím	Nie som si istý(á)	Súhlasím	Úplne súhlasím

6. Zdalo sa mi, že v tomto systéme je priveľa nekonzistentností (nezrovnalostí).

☐

Vôbec nesúhlasím

☐

Nesúhlasím

☐

Nie som si istý(á)

☐

Súhlasím

☐

Úplne súhlasím

7. Viem si predstaviť, že väčšina ľudí by sa tento systém naučila veľmi rýchlo používať.

☐

Vôbec nesúhlasím

☐

Nesúhlasím

☐

Nie som si istý(á)

☐

Súhlasím

☐

Úplne súhlasím

8. Tento systém mi príde ťažkopádny na použitie.

☐

Vôbec nesúhlasím

☐

Nesúhlasím

☐

Nie som si istý(á)

☐

Súhlasím

☐

Úplne súhlasím

9. Pri používaní systému som sa cítil(a) sebaisto.

☐

Vôbec nesúhlasím

☐

Nesúhlasím

☐

Nie som si istý(á)

☐

Súhlasím

☐

Úplne súhlasím

10. Než som mohol(la) začať pracovať s týmto systémom, musel(a) som sa naučiť veľa vecí.

☐

Vôbec nesúhlasím

☐

Nesúhlasím

☐

Nie som si istý(á)

☐

Súhlasím

☐

Úplne súhlasím

Okruh 2 — Zátěž při plnění úloh (NASA-TLX)

Pri každej dimenzii označte hodnotu od 0 (Veľmi málo) po 100 (Veľmi veľa). Označovať môžete krížikom alebo zvislou čiarou na osi.

Mentálne zaťaženie: Ako mentálne náročné bolo plnenie úloh?



Fyzické zaťaženie: Ako fyzicky náročné bolo plnenie úloh?



Časový nátlak: Ako uponáhľané (pod tlakom) bolo plnenie úloh?



Výkon: Ako úspešní ste boli pri plnení úloh?



Úsilie: Ako ťažko ste museli pracovať, aby ste dosiahli to, čo sa vám podarilo?



Frustrácia: Ako veľmi ste boli neistí, znechutení, podráždení, vystresovaní a otrávení?



Okruh 3 — Symptómy nevoľnosti vo VR (VRSQ)

Pri každom symptóme označte intenzitu, akou ste ho pociťovali počas alebo po používaní VR aplikácie.

Symptóm	Žiadny	Mierny	Stredný	Vážny
Všeobecný diskomfort	☐	☐	☐	☐
Únava	☐	☐	☐	☐
Námaha očí	☐	☐	☐	☐
Ťažkosti so zaostrením	☐	☐	☐	☐
Bolesť hlavy	☐	☐	☐	☐
Tlak v hlave	☐	☐	☐	☐
Rozmazané videnie	☐	☐	☐	☐
Závraty pri zatvorených očiach	☐	☐	☐	☐
Vertigo (točenie hlavy)	☐	☐	☐	☐

Okruh 4 — Pocit prítomnosti vo virtuálnom svete (IPQ)

Pri každej otázke označte odpoveď, ktorá najlepšie vystihuje váš zážitok. Pri niektorých otázkach sa škála líši — vždy je uvedená pri otázke.

1. V počítačom generovanom svete som mal(a) pocit, ako keby som „bol(a) tam“.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

2. Mal(a) som pocit, že ma virtuálny svet obklopuje.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

3. Mal(a) som pocit, že vnímam (vidím) iba obrázky.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

4. Necítil(a) som sa vo virtuálnom svete prítomný(á).

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

5. Mal(a) som pocit, že naozaj konám vo virtuálnom svete a nie, že ho zvonku ovládam.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

6. Cítil(a) som sa vo virtuálnom svete prítomný(á).

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

7. Do akej miery ste si boli vedomí reálneho sveta (teplota, zvuky, iní ľudia, ...) počas vašej aktivity vo virtuálnom svete?

☐	☐	☐	☐	☐	☐	☐
Úplne vedomý(á)			Priemerne vedomý(á)			Nevedomý(á)

8. Neuvedomoval(a) som si okolitý reálny svet.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

9. Stále som venoval(a) pozornosť reálnemu prostrediu.

☐	☐	☐	☐	☐	☐	☐
Vôbec nesúhlasím						Úplne súhlasím

10. Virtuálny svet úplne upútal moju pozornosť.

☐ ☐ ☐ ☐ ☐ ☐ ☐
Vôbec nesúhlasím *Úplne súhlasím*

11. Ako veľmi reálny vám pripadal virtuálny svet?

☐ ☐ ☐ ☐ ☐ ☐ ☐
Úplne reálny *Ani ani* *Vôbec nie reálny*

12. Ako veľmi sa zážitok vo virtuálnom svete zhodoval so zážitkom v reálnom svete?

☐ ☐ ☐ ☐ ☐ ☐ ☐
Vôbec sa nezhodoval *Trocha sa zhodoval* *Úplne sa zhodoval*

13. Nakoľko reálny sa vám zdal virtuálny svet?

☐ ☐ ☐ ☐ ☐ ☐ ☐
Iba natoľko ako vymyslený svet *Nevedel(a) som ho odlíšiť od reálneho sveta*

14. Virtuálny svet sa mi zdal skutočnejší ako ten reálny.

☐ ☐ ☐ ☐ ☐ ☐ ☐
Vôbec nesúhlasím *Úplne súhlasím*

Ďakujeme za vyplnenie dotazníka.

*Slovenský preklad otázok prevzatý z: Korečko, Š., Mathes, J., Sobota, B. (2025). K slovenskému prekladu dotazníkov na meranie používateľskej skúsenosti vo virtuálnej realite. Electrical Engineering and Informatics 16, str. 582–586.
Pôvodné nástroje: Brooke (1996) · Hart & Staveland (1988) · Kim et al. (2018) · Schubert et al. (2001).*

B Používateľská príručka

B.1 Funkcia programu

Predmetom tejto bakalárskej práce je rozšírenie systému LirkisEduVePn o nové virtuálne výučbové prostredie zamerané na výučbu prírodovedy pre žiakov 4. ročníka základných škôl. Prostredie obsahuje dve scény — magnetickú scénu, v ktorej žiaci skúmajú vlastnosti magnetizmu pomocou triedenia predmetov, a vesmírnu scénu, v ktorej skladajú slnečnú sústavu a súhvezdia. Aplikácia umožňuje viacero rolí:

- **Žiak** si po prihlásení vyberie pridelenú úlohu a spustí ju v príslušnom režime (VR headset, počítač alebo mobil).
- **Učiteľ** má prístup k dashboardu, kde vidí výsledky svojich skupín, môže priradovať úlohy a sledovať záznamy o riešení úloh jednotlivých žiakov.
- **Administrátor** má najvyššie oprávnenia — môže registrovať nové scény, vytvárať a importovať scenáre a spravovať používateľov a skupiny.

Prostredie podporuje viacero spôsobov ovládania: imerzívne cez VR headset (Meta Quest 2/3/3S) s ovládačmi, klasicky cez počítač s klávesnicou a myšou a alternatívne cez mobilné zariadenie pomocou dotykovej obrazovky a senzorov polohy. Mobilný režim je súčasťou pôvodného prostredia LirkisEduVePn a podľa dokumentácie projektu je podporovaný; v rámci tejto práce však testovanie prebiehalo vo VR a desktopovom režime.

B.2 Inštalácia programu

B.2.1 Požiadavky na technické prostriedky

Pre spustenie aplikácie sú potrebné nasledovné prostriedky:

- Počítač s nainštalovaným systémom **Docker** (Docker Desktop pre Windows/macOS alebo Docker Engine pre Linux),
- operačný systém Windows 10 a novší, macOS 12 a novší, alebo Linux (testované na Ubuntu 22.04),
- minimálne 8 GB operačnej pamäte (odporúčaných 16 GB),
- minimálne 10 GB voľného miesta na disku,
- pripojenie k internetu pri prvom spustení (kvôli stiahnutiu Docker obrazov).

Pre VR režim je potrebný VR headset s podporou WebXR. Aplikácia bola testovaná na zariadeniach **Meta Quest 2**, **Meta Quest 3** a **Meta Quest 3S**. VR headset musí byť pripojený do tej istej siete ako server, na ktorom beží aplikácia. Pre desktopový režim je dostatočný akýkoľvek moderný webový prehliadač s podporou WebGL (Google Chrome, Mozilla Firefox, Microsoft Edge). Pre mobilný režim je potrebný smartfón so senzormi polohy (akcelerometer, gyroskop).

B.2.2 Režimy spustenia

Aplikáciu je možné spustiť tromi spôsobmi podľa toho, kde má bežať server a aké zariadenia sa k nej budú pripájať:

1. **Lokálne na jednom zariadení (localhost).** Server aj klient bežia rovnakom počítači a aplikácia je dostupná na adrese `https://localhost:4200`. Tento režim je vhodný na vývoj a desktopové testovanie; VR headset ani mobil sa k adrese `localhost` nepripoja, keďže tá odkazuje vždy na lokálne zariadenie.
2. **Lokálny server s prístupom z lokálnej siete (VR a mobil).** Server beží na jednom počítači a ostatné zariadenia (VR headset, mobil, iný počítač) v tej istej sieti sa k nemu pripájajú cez jeho IP adresu. Tento režim bol použitý pri používateľskom testovaní v Základnej škole Gemerská Hôrka, kde sa headset Meta Quest 3S pripájal na lokálny server cez školskú sieť.
3. **Produkčné nasadenie na serveri.** Aplikácia je trvalo nasadená na serveri pomocou vopred zostavených Docker obrazov a reverzného proxy servera *nginx*. Tento režim popisuje podkapitola B.2.7.

B.2.3 Spustenie projektu lokálne

Projekt sa spúšťa pomocou Docker Compose, ktorý orchestruje štyri kontajnery: databázu *PostgreSQL*, administračné rozhranie *pgAdmin*, backend (Spring Boot) a frontend (Angular). Konfigurácia týchto kontajnerov je definovaná v súbore `docker-compose.yml` uvedenom vo výpise B.1.

Výpis B.1: Konfigurácia súboru `docker-compose.yml`

```
name: "lirkiseduvepn"
services:

  db-postgres:
    image: postgres:15.1
    container_name: lirkis-database
    ports:
      - "5432:5432"
    volumes:
      - lirkis-data:/var/lib/postgresql/data
    environment:
      - POSTGRES_DB=postgres
      - POSTGRES_USER=postgres
      - POSTGRES_PASSWORD=postgres

  pgadmin:
    image: dpage/pgadmin4:7.1
    container_name: lirkis-pgadmin
    ports:
      - "3000:80"
    volumes:
      - lirkis-pgadmin-data:/var/lib/pgadmin
      - ./servers.json:/pgadmin4/servers.json
    environment:
      - PGADMIN_DEFAULT_EMAIL=admin@admin.com
      - PGADMIN_DEFAULT_PASSWORD=admin
      - PGADMIN_REPLACE_SERVERS_ON_STARTUP=True
    depends_on:
      - db-postgres

  spring:
    build: ./LirkisEduVePn-service
    container_name: lirkis-service
    ports:
      - "8443:8443"
    environment:
```

```

- LIRKIS_BACKEND_URL=https://localhost:8443
- LIRKIS_FRONTEND_URL=https://localhost:4200
depends_on:
- db-postgres

angular:
  build: ./LirkisEduVePn-UI
  container_name: lirkis-ui
  ports:
    - "4200:443"
  environment:
    - LIRKIS_BACKEND_URL=https://localhost:8443
  depends_on:
    - spring

volumes:
  lirkis-data:
  lirkis-pgadmin-data:

```

Súbor `servers.json` pripojený do kontajnera *pgAdmin* obsahuje vopred pripravenú definíciu pripojenia k databáze. Vďaka premennej konfiguračného súboru `PGADMIN_REPLACE_SERVERS_ON_STARTUP` sa toto pripojenie automaticky načíta pri každom spustení kontajnera, takže používateľ nemusí databázový server v *pgAdmin*e pridávať ručne.

Pre lokálne spustenie na jednom zariadení (režim 1) nie je potrebné konfiguráciu meniť. Pre prístup z iných zariadení v sieti (režim 2 – VR headset, mobil) je potrebné v súbore `docker-compose.yml` zmeniť hodnoty premenných `LIRKIS_BACKEND_URL` a `LIRKIS_FRONTEND_URL` z `localhost` na IP adresu počítača v lokálnej sieti, a to **v oboch službách** (`spring` aj `angular`). Napríklad ak má počítač adresu `192.168.1.2`, premenné budú nastavené nasledovne:

Výpis B.2: Konfigurácia URL pre prístup cez lokálnu sieť

```

environment:
- LIRKIS_BACKEND_URL=https://192.168.1.2:8443
- LIRKIS_FRONTEND_URL=https://192.168.1.2:4200

```

Po úprave konfigurácie sa projekt spustí príkazom v koreňovom adresári:

Výpis B.3: Spustenie projektu s prestavbou obrazov

```
docker compose up --build
```

Príznak `--build` zabezpečí prestavenie Docker obrazov, čo je potrebné iba pri prvom spustení alebo po úprave zdrojového kódu. Pri ďalších spusteniach stačí príkaz:

Výpis B.4: Spustenie bez prestavby obrazov

```
docker compose up
```

Projekt sa zastaví kombináciou klávesov `Ctrl+C` v termináli alebo príkazom:

Výpis B.5: Zastavenie projektu

```
docker compose down
```

Po úspešnom spustení sú dostupné nasledovné rozhrania:

- <https://localhost:4200> — hlavná aplikácia (frontend),
- <https://localhost:8443> — backend (REST API),
- <http://localhost:3000> — pgAdmin (správa databázy),
- <localhost:5432> — PostgreSQL databáza (priamy prístup cez SQL klienta).

Poznámka k zabezpečenému pripojeniu. Technológia WebXR, ktorá zabezpečuje imerzívny VR režim, vyžaduje zabezpečené pripojenie cez protokol HTTPS; výnimkou je iba prístup cez `localhost`, ktorý prehliadače tolerujú aj bez platného certifikátu. Pri pripojení z iného zariadenia cez IP adresu (režim 2) preto aplikácia beží cez HTTPS s lokálne vygenerovaným (*self-signed*) certifikátom. Pri prvom pripojení z headsetu alebo iného zariadenia môže prehliadač zobraziť bezpečnostné varovanie o neoverenom certifikáte, ktoré je v tomto prípade potrebné akceptovať a pokračovať na stránku.

B.2.4 Vytvorenie administrátorského účtu

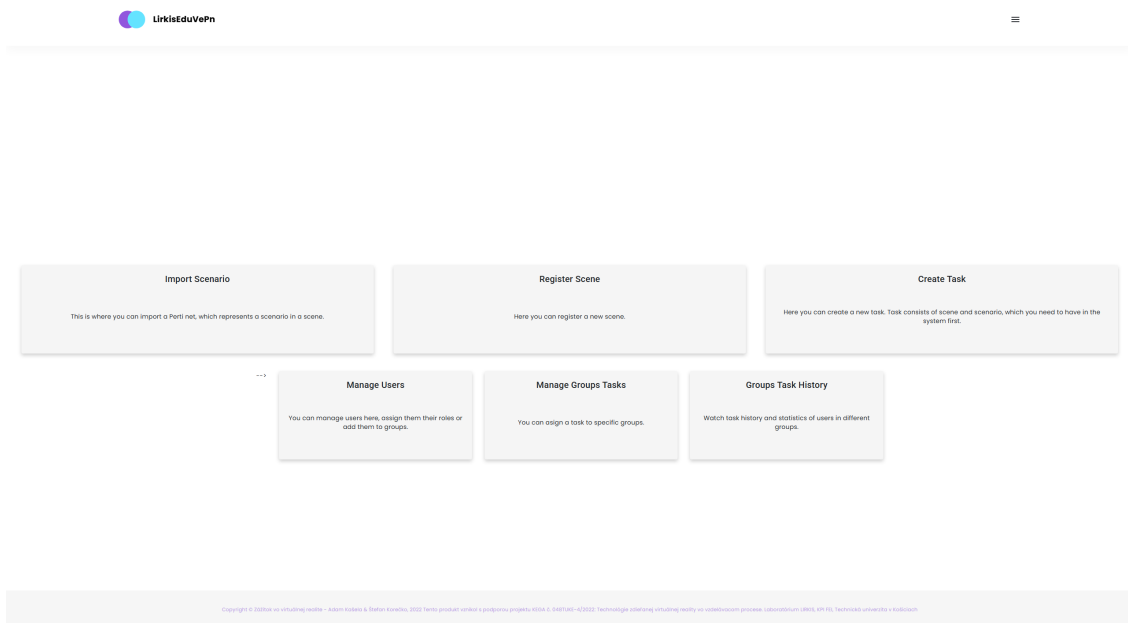
Po spustení projektu je potrebné vytvoriť používateľský účet a povýšiť ho na administrátora. V hlavnej aplikácii na adrese <https://localhost:4200> je potrebné kliknúť na možnosť *Register*, vyplniť registračný formulár a potvrdiť účet cez aktivačný link zaslaný na e-mailovú adresu. Po aktivácii je možné sa prihlásiť, no novovytvorený účet má štandardne rolu *student*. Pre povýšenie účtu na administrátora je potrebné vykonať SQL dotaz v databáze. To je možné cez pgAdmin na adrese <http://localhost:3000> (prihlasovacie údaje sú `admin@admin.com` a heslo `admin`) alebo cez ľubovoľného SQL klienta s parametrami uvedenými v predchádzajúcej časti. Databázové pripojenie je v pgAdmine vďaka súboru `servers.json` už predpripravené, takže ho netreba pridávať ručne.

V pgAdmin je potrebné rozkliknúť databázu postgres, pravým kliknutím vyvolať *Query Tool* a spustiť dotaz uvedený vo výpise B.6:

Výpis B.6: SQL dotaz na povýšenie účtu na administrátora

```
UPDATE users SET role = 'ADMIN' WHERE email = 'moj@email.sk';
```

Po vykonaní dotazu je potrebné sa v hlavnej aplikácii odhlásiť a znovu prihlásiť. V pravom hornom rohu sa zobrazí ikona menu, v ktorom je viditeľná položka *Dashboard* — to je potvrdenie, že účet má administrátorské oprávnenia. Obrázok B.1 znázorňuje *Dashboard* menu.



Obrázok B.1: Dashboard administrátora s možnosťami importu scenára, registrácie scény a vytvorenia úlohy

B.2.5 Import scenára a vytvorenie úlohy

Pre použitie nového virtuálneho prostredia pre prírodovedu je potrebné importovať príslušné scenáre a registrovať scény.

Import scenárov. V dashboarde administrátora je potrebné navigovať na url <https://localhost:4200/dashboard/scenarios> a kliknúť na tlačidlo *Import Scenario*. Súbory oboch zo scenárov sa nachádzajú v projektovom adresári `scenarios/`. Pre toto prostredie sú pripravené scenáre pre obe scény — magnetickú scénu (`nature_test`) a vesmírnu scénu (`space_solar_system`). Štruktúru priečinka scenára znázorňuje výpis B.7 na príklade vesmírnej scény:

Výpis B.7: Štruktúra priečinka scenára vesmírnej scény

```
space_solar_system/
```

```
|-- lang/
|   |-- sceneDescription.en.json
|   `-- sceneDescription.sk.json
|-- space_solar_system.cpn
```

Súbor `.cpn` obsahuje Petriho sieť s definíciou úloh a interakcií v scéne. Súbory `.json` v priečinku `lang` obsahujú jazykové preklady textov zobrazovaných v scéne (slovenský a anglický preklad).

Registrácia scény. Na adrese `https://localhost:4200/dashboard` je potrebné kliknúť na tlačidlo *Register Scene*. V dialógovom okne sa vyplní názov scény (napr. *Magnetizmus* alebo *Vesmír*) a nahrá sa náhľadový obrázok.

Vytvorenie úlohy. Po registrácii scény je potrebné kliknúť na tlačidlo *Create Task*, vyplniť formulár a v rozbaľovacích zoznamoch vybrať importovaný scenár a registrovanú scénu.

B.2.6 Vytvorenie skupiny a priradenie žiakov

Po vytvorení úlohy je potrebné založiť skupinu žiakov, ktorí budú mať k tejto úlohe prístup. Na adrese `https://localhost:4200/dashboard/groups` sa pomocou tlačidla *Create Group* vytvorí nová skupina, do ktorej je možné priradiť jednu alebo viacero úloh. Žiaci sa do konkrétnej skupiny priradujú na adrese `https://localhost:4200/dashboard/users`. Pri každom používateľovi je ikona ceruzky, ktorou sa otvorí editačný dialóg. V dialógu sa vyberú skupiny, do ktorých má byť žiak zaradený, a zmeny sa potvrdia tlačidlom *Save*.

B.2.7 Produkčné nasadenie na serveri

Okrem lokálneho spustenia je aplikáciu možné nasadiť aj trvalo na samostatnom serveri. Toto nasadenie nevyužíva zostavovanie obrazov zo zdrojového kódu, ale vopred zostavené Docker obrazy nahraté do registra *Docker Hub*, ktoré sa na serveri iba stiahnu a spustia. Pred serverom je zároveň nasadený reverzný proxy server *nginx*, ktorý zabezpečuje smerovanie požiadaviek a TLS šifrovanie.

V čase písania tejto práce je aplikácia takto nasadená na serveri Technickej univerzity v Košiciach s IP adresou `147.232.205.222` a je dostupná na adrese `https://147.232.205.222` z prostredia školskej siete alebo prostredníctvom školskej VPN.

Postup nasadenia pozostáva z troch krokov. Najprv sa lokálne zostavené Docker obrazy označia a nahrajú do registra Docker Hub:

Výpis B.8: Označenie a nahranie Docker obrazu

```
docker tag lirkiseduvepn-spring pouzivatel/lirkiseduvepn-spring:
  deploy
docker push pouzivatel/lirkiseduvepn-spring:deploy
```

Následne sa na serveri v súbore `docker-compose.yml` odkáže na nahraté obrazy (namiesto direktívy `build`) a kontajnery sa spustia v režime na pozadí:

Výpis B.9: Spustenie na serveri v režime na pozadí

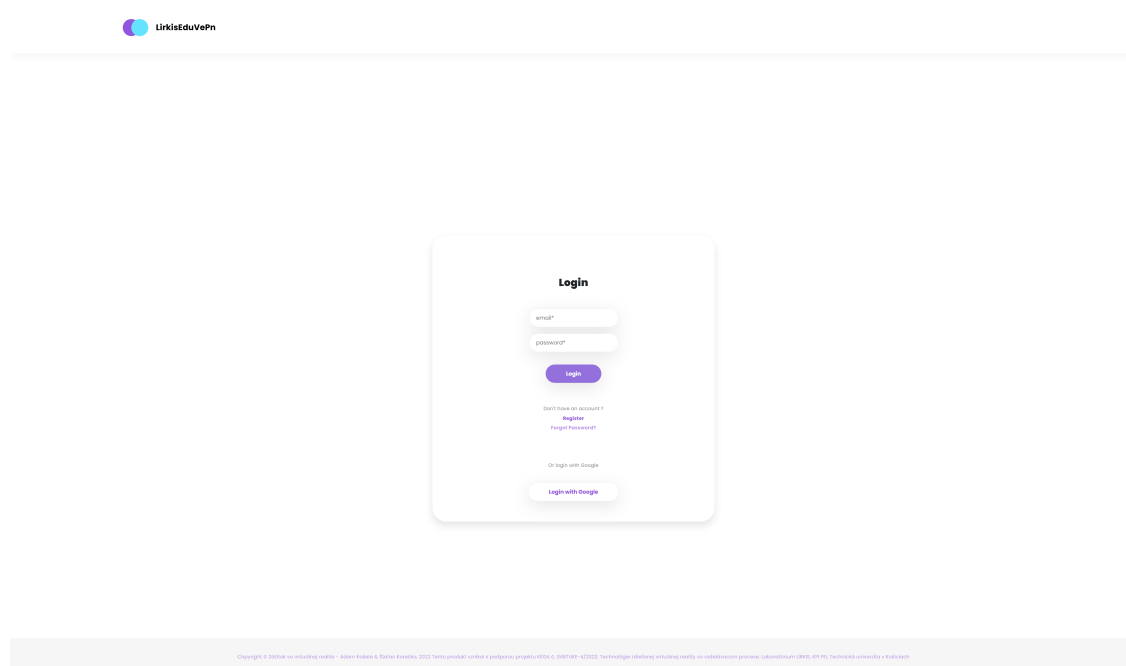
```
docker compose up -d
```

Nakoniec sa nakonfiguruje `nginx` ako reverzný proxy server. Konfiguračný súbor (`nginx.conf`) sa skopíruje do adresára konfigurácií `nginx`, kde sa nastaví názov servera a cesty k SSL certifikátu a kľúču. Podrobný postup konfigurácie `nginx` je súčasťou dokumentácie projektu v jeho repozitári.

B.3 Použitie programu

B.3.1 Prihlasovacia obrazovka

Po otvorení aplikácie sa zobrazí prihlasovacia obrazovka (viď obrázok B.2), na ktorej žiak vyplní svoje prihlasovacie údaje (e-mailovú adresu a heslo) a klikne na tlačidlo *Login*. Pri prvom prihlásení sa žiak najprv zaregistruje cez tlačidlo *Register*, vyplní registračný formulár a potvrdí svoj účet cez aktivačný link.



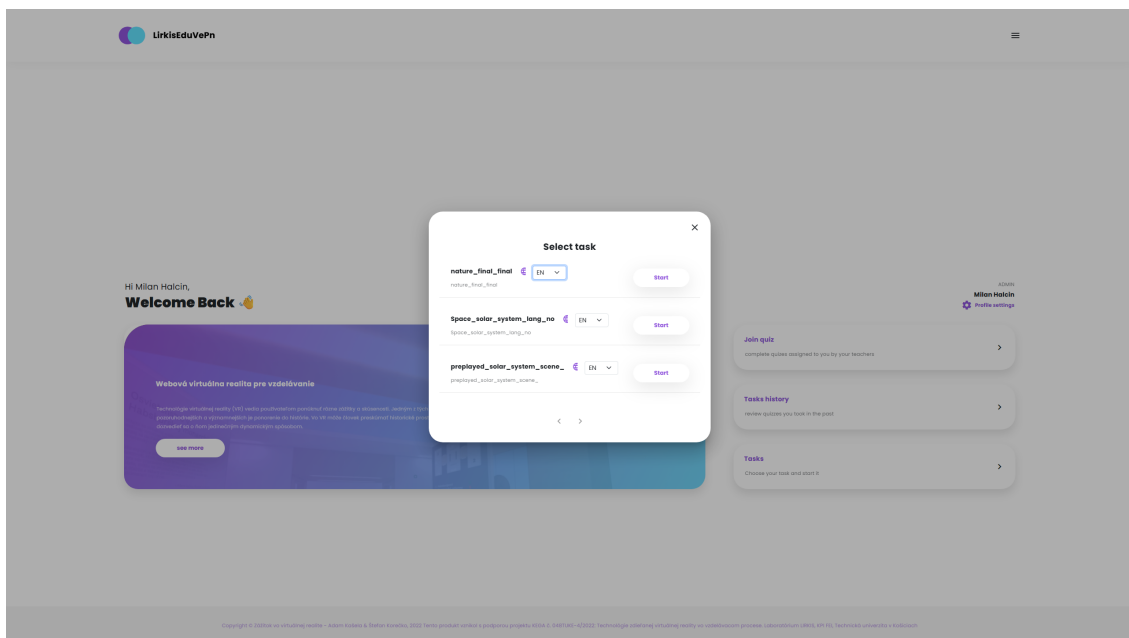
Obrázok B.2: Prihlasovacia obrazovka aplikácie

B.3.2 Hlavná obrazovka

Po úspešnom prihlásení sa žiak ocitne na hlavnej obrazovke aplikácie. V hornom pravom rohu je ikona, ktorá otvára bočné menu s navigáciou. Pre žiaka je tu prístupná položka *Profile*, ktorá vedie k osobnému profilu s priradenými úlohami.

B.3.3 Profil žiaka a spustenie úlohy

Profil žiaka je dostupný na adrese `https://localhost:4200/user/profile`. V sekcii *Tasks* sú zobrazené úlohy priradené žiakovi v rámci jeho skupín (viď obrázok B.3). Pri každej úlohe je rozbaľovacia šípka, ktorou sa otvorí výber jazyka a tlačidlo *Start*. Po kliknutí na *Start* sa otvorí samotná virtuálna scéna.



Obrázok B.3: Profil žiaka s priradenými úlohami a tlačidlom *Start*

B.3.4 Ovládanie virtuálnej scény

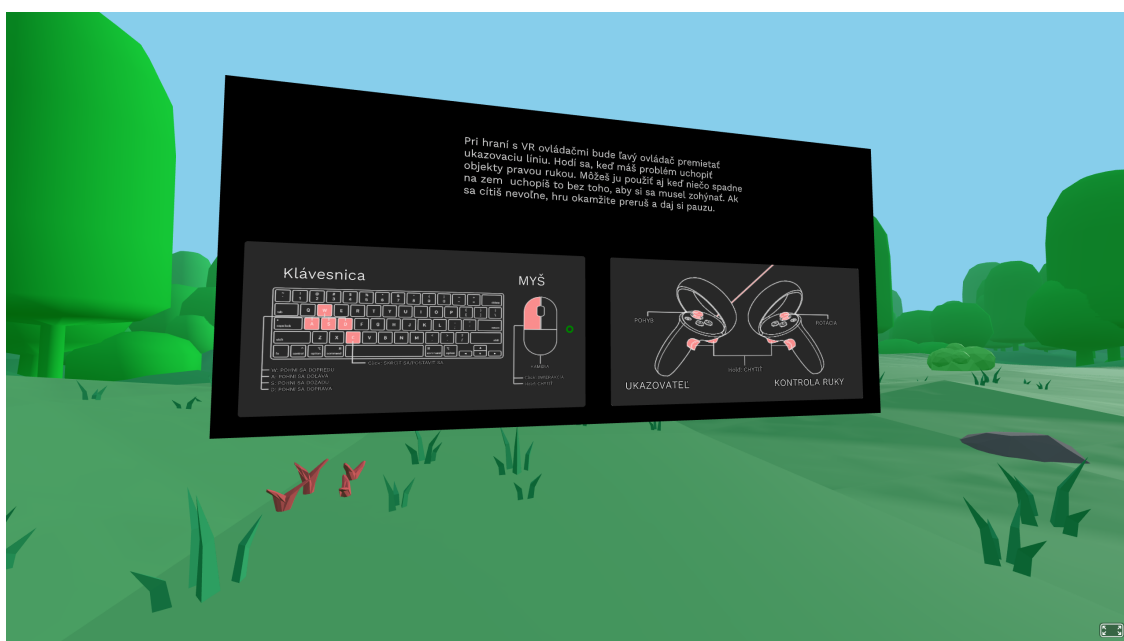
V oboch scénach je návod na ovládanie permanentne zobrazený v prostredí, čo znázorňuje obrázok B.4.

Počítač. V desktopovom režime sa pohyb v scéne ovláda klávesami W, A, S, D a pohľad sa otáča pohybom myši. Interakcia s objektmi (uchopenie, polozenie, stlačenie tlačidla) sa vykonáva ľavým tlačidlom myši. V scéne je v ľavom hornom rohu zobrazený informačný panel s prehľadom klávesových skratiek.

VR headset. V imerzívnom režime cez VR headset (Meta Quest) sa žiak pohybuje fyzickým pohybom v priestore alebo pomocou teleportácie. Teleportácia

sa aktivuje držaním joysticku na ovládači a uvoľnením v cieľovej pozícii. Objekty sa uchopujú stlačením spúšte (*trigger*) na ovládači, kým je ovládač v dotyku s objektom. Uvoľnenie spúšte objekt pustí.

Mobilný telefón. V mobilnom režime sa pohyb po scéne vykonáva naklonením telefónu vpred alebo vzad, rotácia pohľadu sa vykonáva otáčaním telefónu okolo zvislej osi. Interakcia s objektmi sa vykonáva dotykom obrazovky.



Obrázok B.4: Informačný panel s prehľadom ovládacích prvkov v scéne

B.3.5 Magnetická scéna

V magnetickej scéne (viď obrázok B.5) sa žiak ocitne v lesnom prostredí pri stole. Vedľa stola sa nachádzajú dva magnetické koše — jeden pre magnetické predmety, druhý pre nemagnetické. Žiakovi sa po vstupe do scény zobrazí navádzacia guľôčka (komponent proximity-sensor), ktorá vedie žiaka k magnetu položenému v scéne. Po uchopení magnetu sa v prostredí objavia predmety určené na triedenie — ich konkrétna skupina je v každom spustení náhodne vybraná z väčšej zásoby predmetov, pričom počet vyberaných predmetov je riadený Petriho sieťou scenára (štandardne šesť). Žiak skúša priložiť magnet k jednotlivým predmetom a podľa reakcie magnetu zistí, či je predmet magnetický. Následne predmet umiestni do správneho koša. Úloha je splnená, keď žiak roztriedi predmety do správnych košov.



Obrázok B.5: Magnetická scéna – triedenie predmetov pomocou magnetu

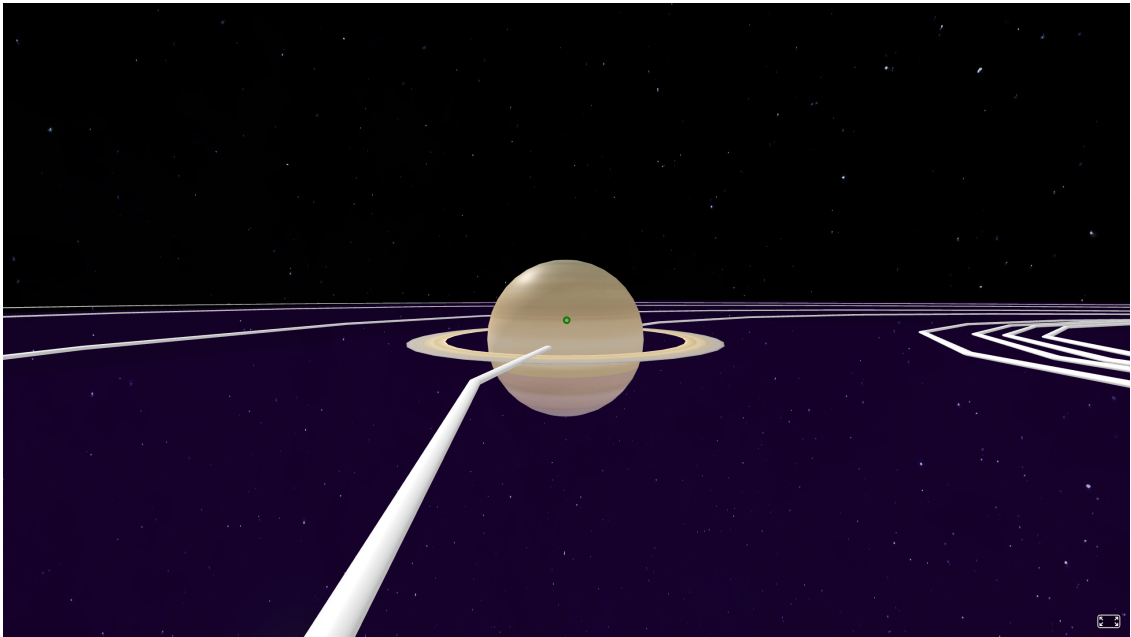
B.3.6 Vesmírna scéna

Vesmírna scéna obsahuje dve úlohy — umiestňovanie planét a skladanie súhvezdia. Obe úlohy sú navzájom nezávislé a žiak si môže zvoliť, ktorou z nich začne; splnenie jednej úlohy nie je podmienkou pre prístup k druhej.

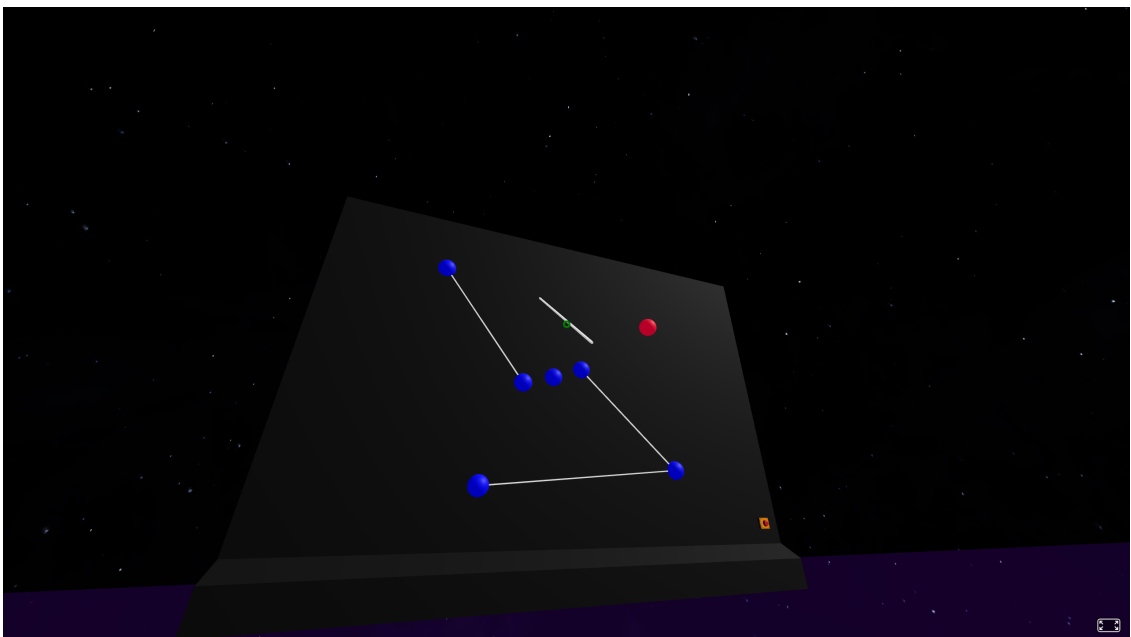
Úloha umiestňovania planét. V scéne je v strede umiestnené Slnko, okolo ktorého sú obežné dráhy pre osem planét. Planéty sú rozložené v priestore okolo žiaka. Úlohou je uchopiť každú planétu a umiestniť ju na správnu obežnú dráhu (čo zachytáva obrázok B.6) podľa jej skutočného poradia v slnečnej sústave (Merkúr, Venuša, Zem, Mars, Jupiter, Saturn, Urán, Neptún). Pre túto úlohu nie sú v aktuálnej verzii prostredia dostupné žiadne nápovedy — správne poradie planét musí žiak vedieť alebo si ho vydedukovať pokusom a omylom.

Úloha skladania súhvezdia. Žiak si vyberie jedno z dvoch súhvezdí, ktoré chce zostaviť — Veľký voz alebo Orión. V scéne sú rozložené hviezdy a spojnice, ktoré žiak postupne umiestňuje na obraz súhvezdia na oblohe (viď obrázok B.7). Systém umožní vložiť hviezdu iba do správnej pozície, čím vedie žiaka k úspešnému dokončeniu súhvezdia.

Pre úlohu skladania súhvezdia sú v scéne dostupné dve nápovedy — tlačidlo `buttonHint1` zobrazí nápovedu pre Veľký voz, tlačidlo `buttonHint2` pre Orión. Obe nápovedy majú rovnaký formát — vizuálne zvýraznia príslušné súhvezdie na oblohe, aby žiak videl cieľový tvar, ktorý má zostaviť.



Obrázok B.6: Vesmírna scéna – umiestňovanie planét na obežné dráhy



Obrázok B.7: Vesmírna scéna – skladanie súhvezdia podľa nápovedy

B.3.7 Ukončenie a opakovanie úlohy

Po úspešnom splnení úlohy sa zobrazí oznámenie o ukončení a žiak môže zo scény vystúpiť kliknutím na tlačidlo *Exit* v menu scény. Žiakove výsledky sa automaticky uložia do databázy a sú dostupné v dashboarde učiteľa a administrátora. V prípade potreby si môže žiak úlohu zopakovať opätovným spustením z profilu.

C Systémová príručka

C.1 Funkcia programu

Virtuálne výučbové prostredie pre prírodovedu je súčasťou systému LirkisEduVePn vyvíjaného na Katedre počítačov a informatiky FEI TUKE. Práca rozširuje existujúci systém o nové scény a scenáre zamerané na žiakov 4. ročníka ZŠ. Systém je postavený ako webová aplikácia, ktorej hlavnou výhodou je prístupnosť z viacerých zariadení bez nutnosti inštalácie natívneho softvéru.

C.2 Popis programu

C.2.1 Popis riešenia

Systém LirkisEduVePn pozostáva z troch hlavných častí — frontendu, backendu a databázy. Tieto časti komunikujú medzi sebou cez HTTP protokol a sú nasadené ako samostatné Docker kontajnery. Toto rozdelenie umožňuje nezávislý vývoj a nasadenie jednotlivých komponentov.

Frontend je implementovaný v softvérovom rámci Angular s využitím rámca *A-Frame* pre zobrazenie 3D scén vo virtuálnej realite v prehliadači. Pre fyzikálnu simuláciu (kolízie, gravitácia) sa využíva knižnica *Ammo.js*. Frontend komunikuje s backendom cez REST API a odosiela požiadavky v JSON formáte.

Backend je implementovaný v jazyku Java s využitím rámca Spring Boot. Zabezpečuje autentifikáciu používateľov, správu scén a scenároov, vyhodnocovanie úloh a ukladanie záznamov o riešení úloh.

Databáza PostgreSQL ukladá údaje o používateľoch, scénach, scenároch, skupinách, úlohách a záznamoch o priebehu riešenia úloh. Backend k nej pristupuje cez JDBC a SQL dotazy.

C.2.2 Štruktúra prostredia pre prírodovedu

Virtuálne prostredie pre prírodovedu pozostáva z dvoch nezávislých scén — magnetickej scény a vesmírnej scény — z ktorých každá má samostatný Petriho sieťový scenár a vlastné jazykové súbory. Scenár magnetickej scény je definovaný v súbore `nature_test.cpn` a vesmírnej scény v `space_solar_system.cpn`.

C.3 Popis implementovaných súborov

Implementované A-Frame komponenty sa nachádzajú v adresári `components` v rámci frontendovej časti projektu (`LirkisEduVePn-UI`). Scenáre vo formáte Petriho sietí a ich jazykové mutácie sú umiestnené v adresári `scenarios`, samotné Angular komponenty scén v adresári `scenes`. Podrobnú súborovú štruktúru komponentov a aktív tohto prostredia uvádza kapitola 5 (obrázok so štruktúrou projektu).

Jednotlivé časti štruktúry majú nasledovné určenie:

- **components** — adresár obsahujúci A-Frame komponenty. Okrem komponentov vytvorených v rámci tejto práce sa tu nachádzajú aj komponenty z predchádzajúcich prác a z paralelnej diplomovej práce.
- **modules** — pomocné moduly pre prácu s Petriho sieťami (`petriNet.mjs`, `petriNetLoader.mjs`) a pre zaznamenávanie priebehu riešenia jednotlivých úloh (`userActivityLogger.js`).
- **scenarios** — adresár so scenármi vo forme Petriho sietí (`.cpn` súbory) a ich jazykovými mutáciami v priečinku `lang`.
- **scenes** — Angular komponenty samotných scén s príslušnými šablónami a štýlmi.

C.4 Popis implementovaných A-Frame komponentov

V rámci tejto práce bola implementovaná skupina A-Frame komponentov, ktoré zabezpečujú interaktivitu a logiku oboch scén. Tabuľka C.1 uvádza prehľad týchto komponentov spolu s názvom zdrojového súboru a názvom, pod ktorým je komponent registrovaný v A-Frame. Podrobný popis princípu fungovania, parametrov a implementačných detailov každého komponentu sa nachádza v kapitole 5.

Tabuľka C.1: Prehľad implementovaných A-Frame komponentov

Registrovaný názov	Zdrojový súbor	Funkcia
magnet-box	magnet.component.js	Simulácia magnetického poľa pôsobiaceho na okolité predmety v magnetickej scéne.
random-pool-picker	randomPoolPicker.component.js	Náhodný výber podmnožiny predmetov z väčšej zásoby, počet riadený Petriho sieťou.
orbit-on-collision	planetOrbit.component.js	Obeh telesa po kruhovej dráhe pri kolízii s povoleným objektom (planéty vo vesmírnej scéne).
proximity-sensor	objectProximitySensor.component.js	Detekcia blízkosti objektov a priestorová spätná väzba (navádzacia guľôčka, farebné zvýraznenie).
activate-object-on-event	onEventActivateObject.component.js	Riadené zobrazovanie objektov v scéne v reakcii na definovanú udalosť.

Implementácia ďalej nadväzuje na komponenty pochádzajúce z paralelnej diplomovej práce, ktoré sú esenciálne pre fungovanie tohto prostredia — `ammo-init` (súbor `ammoInit.component.js`) pre riadenú inicializáciu fyzikálnych telies, `grabbable-by-parenting` (súbor `grabbableByParenting.component.js`) pre uchopovanie objektov a kľúčový podporný komponent `attachment-box` (súbor `attachmentBox.component.js`) pre deterministické pripájanie objektov na vopred určené pozície. Pôvod a úloha týchto komponentov sú podrobnejšie rozobrané v kapitole 5.

C.5 Využité technológie

Pre implementáciu virtuálneho prostredia boli využité nasledovné technológie a knižnice:

- **A-Frame** (verzia 1.7.1) — webový softvérový rámec pre tvorbu 3D scén a virtuálnej reality, postavený na knižnici *Three.js* a využívajúci rozhranie WebXR.
- **Three.js** (verzia 0.173.0) — JavaScript knižnica pre 3D grafiku v prehliadači cez WebGL.
- **Ammo.js** — JavaScript verzia fyzikálnej knižnice Bullet Physics, použitá pre kolízie a fyzikálnu simuláciu objektov.
- **Angular** (verzia 19.1.7) — softvérový rámec pre frontend aplikácie.
- **Spring Boot** — softvérový rámec pre backend v jazyku Java.
- **PostgreSQL** (verzia 15.1) — relačná databáza pre ukladanie údajov.
- **Docker** — platforma pre kontajnerizáciu aplikácií, ktorá zjednodušuje nasadenie a vývoj.
- **nginx** — webový server používaný ako reverzný proxy pri produkčnom nasadení.

Pre tvorbu a úpravu Petriho sieťových scenárov bol využitý nástroj **CPN Tools**¹², ktorý umožňuje editovať Petriho siete v ich štandardnom formáte `.cpn`.

C.6 Spustenie programu

Pre lokálne spustenie programu počas vývoja je potrebné mať nainštalovaný Docker a v koreňovom adresári projektu spustiť príkaz:

Výpis C.1: Spustenie projektu pre lokálny vývoj

```
docker compose up --build
```

Príkaz prečíta súbor `docker-compose.yml`, zostaví Docker obrazy pre frontend (Angular) a backend (Spring Boot) a spustí všetky štyri kontajnery (databáza, pgAdmin, backend, frontend). Po prvom úspešnom zostavení obrazov je

¹²Dostupné na: <https://cpntools.org>

možné v ďalších spusteniach vynechať príznak `--build`, čo výrazne urýchli štartovanie:

Výpis C.2: Spustenie bez prestavby obrazov

```
docker compose up
```

Pre spustenie na pozadí (bez blokovania terminálu) je možné použiť príznak `-d`:

Výpis C.3: Spustenie na pozadí

```
docker compose up -d
```

Aplikácia je následne po spustení dostupná vo webovom prehliadači na adrese `https://localhost:4200`. Podrobný popis jednotlivých režimov spustenia (lokálne, prístup z lokálnej siete pre VR a mobil, produkčné nasadenie) je uvedený v používateľskej príručke v prílohe B.

C.7 Nasadenie na server

Aktuálna verzia aplikácie je nasadená na serveri TUKE a dostupná iba zo školskej siete alebo cez TUKE VPN na adrese `https://147.232.205.222`.

C.7.1 Aktualizácia aplikácie na serveri

Pre aktualizáciu aplikácie na produkčnom serveri je potrebné publikovať nové Docker obrazy do registra *Docker Hub* a následne na serveri stiahnuť aktualizované obrazy. Najprv je potrebné prihlásiť sa do Docker Hubu z lokálneho stroja:

Výpis C.4: Prihlásenie do Docker Hub

```
docker login
```

Po lokálnom zostavení obrazov je potrebné im priradiť značky (tagy) odkazujúce na používateľské konto Docker Hubu:

Výpis C.5: Označenie obrazov tagmi

```
docker tag lirkiseduvepn-spring kicksuch/lirkiseduvepn-spring:  
deploy_v3.4
```

```
docker tag lirkiseduvepn-angular kicksuch/lirkiseduvepn-angular:  
deploy_v3.4
```

Potom je možné obrazy nahrať na Docker Hub:

Výpis C.6: Nahratie obrazov na Docker Hub

```
docker push kicksuch/lirkiseduvepn-spring:deploy_v3.4
docker push kicksuch/lirkiseduvepn-angular:deploy_v3.4
```

Na serveri sa potom v adresári s konfiguráciou `docker-compose.yml` (odlišnej od vývojovej verzie — používa publikované obrazy namiesto lokálneho zostavenia) spustí príkaz:

Výpis C.7: Aktualizácia aplikácie na serveri

```
docker-compose up -d
```

Docker automaticky stiahne aktualizované obrazy z Docker Hubu a reštartuje kontajnery.

C.7.2 Konfigurácia servera

Na produkčnom serveri sa používa odlišný `docker-compose.yml`, ktorý namiesto lokálneho zostavenia využíva publikované Docker obrazy z Docker Hubu. Ukážka konfigurácie pre služby `spring` a `angular` je vo výpise C.8:

Výpis C.8: Konfigurácia `docker-compose.yml` na produkčnom serveri

`spring:`

```
  image: kicksuch/lirkiseduvepn-spring:deploy_v3.4
```

`ports:`

```
  - "8443:8443"
```

`environment:`

```
  - LIRKIS_BACKEND_URL=https://147.232.205.222
```

```
  - LIRKIS_FRONTEND_URL=https://147.232.205.222
```

`angular:`

```
  image: kicksuch/lirkiseduvepn-angular:deploy_v3.4
```

`ports:`

```
  - "4200:443"
```

`environment:`

```
  - LIRKIS_BACKEND_URL=https://147.232.205.222
```

C.7.3 Konfigurácia nginx

Pre presmerovanie HTTPS prevádzky na správne porty kontajnerov sa na serveri používa reverzný proxy server `nginx`. Konfiguračný súbor je uložený v adresári

/etc/nginx/sites-available/lirkiseduven a aktivovaný symbolickým odkazom v adresári /etc/nginx/sites-enabled/.

Pri prvotnom nastavení servera je potrebné:

1. Nainštalovať Docker a nginx na serveri.
2. Skopírovať obsah súboru `nginx.conf` z tohto projektu do priečinka `/etc/nginx/sites-available/lirkiseduven` a upraviť tieto hodnoty podľa parametrov servera:
 - `server_name`,
 - `ssl_certificate`,
 - `ssl_certificate_key`.
3. Deaktivovať predvolenú konfiguráciu a aktivovať konfiguráciu projektu:

Výpis C.9: Aktivácia konfigurácie nginx

```
rm /etc/nginx/sites-enabled/default
ln -s /etc/nginx/sites-available/lirkiseduven /etc/nginx/sites-enabled/lirkiseduven
service nginx restart
```

Po reštarte nginx je aplikácia dostupná na adrese servera s plnou HTTPS podporou.