

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

**System virtuálnych výučbových prostredí
riadených procesnými grafmi**

Diplomová práca

2024

Bc. Juraj Rudy

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

**System virtuálnych výučbových prostredí
riadených procesnými grafmi**

Diplomová práca

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

Košice 2024

Bc. Juraj Rudy

Abstrakt v SJ

Táto práca sa zameriava na rozšírenie funkcionalít systému pre virtuálne prostredia, ako napríklad ovládania pomocou rôznych typov zariadení, technické úpravy a rozšírenia vo webovom rozhraní. Analyzovali sme predchádzajúce práce, ktoré sa tomuto systému venovali, identifikovali sme nedostatky a možné vylepšenia. Zároveň sme vypracovali vlastný prieskum, aby sme preverili schopnosť ľudí pochopiť tvorbu Petriho sietí po zjednodušenom vysvetlení.

Kľúčové slová v SJ

Petriho siete, virtuálna realita, aframe, angular, Java spring

Abstrakt v AJ

This thesis focuses on extending the system functionalities for virtual environments, such as control using different types of devices, technical modifications and extensions in the web interface. We have analyzed previous works that have addressed this system, identifying shortcomings and possible improvements. At the same time, we developed our own survey to test people's ability to understand the creation of Petri nets after a simplified explanation.

Kľúčové slová v AJ

Petri nets, virtual reality, aframe, angular, Java spring

Bibliografická citácia

RUDY, Juraj. *Systém virtuálnych výučbových prostredí riadených procesnými grafmi*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2024. 59s. Vedúci práce: Ing. Štefan Korečko PhD.

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

**ZADANIE
DIPLOMOVEJ PRÁCE**

Študijný odbor: **Informatika**
Študijný program: **Kyberbezpečnosť**

Názov práce:

Systém virtuálnych výučbových prostredí riadených procesnými grafmi
System of Process-Graph-Driven Virtual Educational Environments

Študent: **Bc. Juraj Rudy**
Školiteľ: **Ing. Štefan Korečko, PhD.**
Školiace pracovisko: **Katedra počítačov a informatiky**
Konzultant práce:
Pracovisko konzultanta:

Pokyny na vypracovanie diplomovej práce:

1. Oboznámiť sa s prototypom systému pre virtuálne výučbové prostredia so scenármi definovanými Petriho sieťami, vyvinutým na školiacom pracovisku,
2. Analyzovať nedostatky a možnosti rozšírenia prototypu.
3. Analyzovať existujúce prístupy k využitiu scenárom riadených virtuálnych prostredí vo výučbe a možnosti praktického použitia Petriho sietí na tento účel.
4. Navrhnuť a implementovať vybrané vylepšenia a rozšírenia prototypu.
5. Návrh a implementáciu koordinovať s ďalšími riešiteľmi prototypu.
6. Výsledok implementácie vyhodnotiť a vypracovať dokumentáciu podľa pokynov vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský
Termín pre odovzdanie práce: 19.04.2024
Dátum zadania diplomovej práce: 31.10.2023



L. Z. Pukhara

prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

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

Košice, 19.5.2024

.....

Vlastnoručný podpis

Podakovanie

Na tomto mieste by som rád poďakoval svojmu vedúcemu práce za jeho čas a odborné vedenie počas riešenia mojej záverečnej práce.

Rovnako by som sa rád poďakoval svojej rodine a priateľom, no predovšetkým svojej mame, za ich podporu a povzbudzovanie počas celého môjho štúdia.

Obsah

Úvod	1
1 Vzdelávanie vo virtuálnom prostredí	3
1.1 Vývoj v oblasti vzdelávania vo virtuálnom prostredí	3
1.1.1 Prvá etapa (1996–2008)	4
1.1.2 Druhá etapa (2009–2016)	4
1.1.3 Tretia etapa (2017–2022)	4
1.2 Výhody vzdelávania vo virtuálnom prostredí	5
1.3 Limitácie vzdelávania vo virtuálnom prostredí	5
1.4 Trendy vo vzdelávaní vo virtuálnom prostredí	5
1.5 Metaverse	6
2 Porozumenie Petriho sieťam neodbornou verejnosťou	7
2.1 Definícia Petriho sietí	7
2.2 Formulácia výskumných otázok	7
2.3 Postup získavania dát	8
2.4 Vyhotovenie vlastného prieskumu	9
2.4.1 Výber respondentov	9
2.4.2 Štruktúra prieskumu	9
2.4.3 Zbieranie údajov	12
2.4.4 Metodika hodnotenia výskumnej úlohy	12
2.4.5 Analýza zozbieraných údajov	13
2.4.6 Vyhodnotenie prieskumu	16
3 Analýza súčasného stavu projektu	17
3.1 Použité technológie	17
3.2 Scenárom riadené výučbové prostredia vo virtuálnej realite	18
3.2.1 Petriho siete	18
3.2.2 Rozšírenie scény	18
3.2.3 Komunikácia so serverom	18

3.2.4	Ovládanie pomocou VR ovládačov	18
3.3	Spracovanie údajov z virtuálnych výučbových prostredí	19
3.3.1	Implementácia autentifikácie	19
3.3.2	Implementácia autorizácie	19
3.3.3	Funkcionality webovej aplikácie	19
3.4	Analýza nedostatkov a návrhy opráv	20
3.4.1	Identifikácia a riešenie implementačných nedostatkov v aktuálneho stavu projektu	20
3.4.2	Analýza ovládania scény pre rôzne druhy zariadení	28
3.4.3	Analýza ovládania pomocou počítača	29
3.4.4	Analýza ovládania pomocou VR headsetu	29
3.4.5	Identifikácia a riešenie všeobecných problémov v projekte	29
4	Nasadenie existujúceho riešenia na TUKE cloud server	31
4.1	Definovanie úlohy	31
4.2	Dostupné možnosti na TUKE cloude	32
4.3	Proces nasadenia na server	32
5	Opravy implementačných nedostatkov v aktuálnej verzii aplikácie	34
5.1	Úpravy v používateľskom rozhraní webu	34
5.2	Doplnenie responzivity častí webu	34
5.3	Vytvorenie prehľadu scenárov a pridanie možnosti odstrániť scenár	36
5.4	Doplnenie obsahu do stránky <i>about</i>	37
5.5	Úpravy v scéne	38
5.6	Ovládanie scény	41
5.6.1	Nové ovládanie pre mobilné zariadenia	41
5.6.2	Úprava ovládania pre VR headsety	43
5.7	SSL certifikát s vlastným podpisovaním	46
5.8	Implementácia pre novšie protokoly HTTP	47
5.9	Implementácia šifrovaného HTTPS protokolu	47
5.10	Zavedenie pravidiel pre vytváranie hesiel	48
6	Vyhodnotenie	50
6.1	Vyhodnotenie prieskumu	50
6.1.1	Vlastné zhodnotenie	51
6.2	Vyhodnotenie vypracovaných vylepšení	51
6.3	Možnosti ďalších vylepšení	53
7	Záver	55

Literatúra	57
Zoznam príloh	60
A Používateľská príručka	61
A.1 Funkcia programu	61
A.2 Inštalácia programu	61
A.2.1 Požiadavky na technické prostriedky	61
A.2.2 Spustenie projektu	62
A.3 Použitie programu	67
A.3.1 Prihlasovacia obrazovka	67
A.3.2 Hlavná obrazovka	67
A.3.3 Dashboard pre učiteľov a adminov	68
A.3.4 Virtuálna scéna	71
A.3.5 Prenášanie exponátov	71
A.3.6 Označovanie exponátov	71
A.3.7 Zobrazovanie informácií k exponátom	72
A.3.8 Zvukové znamenia	72
B Systémová príručka	73
B.1 Funkcia programu	73
B.2 Popis programu	73
B.2.1 Popis riešenia	73
B.2.2 Štruktúra angular projektu	73
B.2.3 Štruktúra java projektu	75
B.3 Popis implementovaných súborov	76
B.3.1 Aframe komponenty	76
B.3.2 Angular komponenty	76
B.3.3 Java komponenty	77
B.4 Využitie technológie	77
B.5 Spustenie programu	78
B.6 Nasadenie na server	81

Zoznam obrázkov

1.1	Graf medziročného nárastu počtu publikácií zameraných na vzdelávanie vo VR	4
3.1	Malé tlačidlá v navigačnom paneli	21
3.2	Stránka <i>/home</i>	22
3.3	Administrátorské menu	23
3.4	Formulár pre registráciu scény	24
3.5	25
3.6	Aktuálna mapa slovenského územia pre dejepisnú úlohu	26
3.7	Pohľad spoza prekročených bariér	26
3.8	Dislokácia hodvábu	27
3.9	Problém s menovkami	28
5.1	Upravené používateľské rozhranie	34
5.2	Informačná stena s upravenou farbou textu	35
5.3	Náhľad stránky about	38
5.4	Informačná stena s upravenou farbou textu	39
5.5	Informačná stena so zarovnaným textom	39
5.6	Opravená viditeľnosť menovky na mieste priradovania	40
5.7	Povolený blink	44
5.8	Nepovolený blink	44
5.9	Manipulácia s objektami v scéne pomocou VR headsetu	46
5.10	Ukážka nesplnenia pravidiel pre heslo	49
A.1	Prihlasovacia obrazovka do pgAdmin	65
A.2	Okno so servermi v pgAdmin	66
A.3	Databáza v pgAdmin	67
A.4	Prihlasovacie okno	68
A.5	Hlavná obrazovka	68
A.6	Výber úlohy	69

A.7	Používateľský dashobard	69
A.8	Prehľad histórie úloh	70
A.9	Interakcie prenášania exponátov	71
A.10	Interakcie označovania exponátov	72
A.11	Interakcie zobrazenia informácie	72
B.1	Otvorenie projektu	78
B.2	Maven závislosti	79
B.3	Spustenie servera	80
B.4	Úvodná obrazovka aplikácie	81
B.5	Vylistovanie obrazov	81
B.6	Tagovanie obrazu	81
B.7	Push do dockerhubu	82

Zoznam tabuliek

2.1	Tabuľka výskumných otázok	8
2.2	Tabuľka dopytov na databázy vedeckých portálov	8
2.3	Zozbierané dáta o netechnickej skupine	13
2.4	Zozbierané dáta o technickej skupine	14
2.5	Tabuľka skratiek pre jednotlivé typy chýb	14
2.6	Výsledky netechnickej skupiny	15
2.7	Výsledky technickej skupiny	15
6.1	Porovnanie výsledkov oboch skupín prieskumu	50

Úvod

V posledných rokoch zažíva svet rozmach technológií virtuálnej reality. Vidíme rôzne odvetvia, či už lekárske[1], vojenské[2], edukačné[3], alebo napríklad aj zábavný priemysel[4], ktoré využívajú nové možnosti ponúkané technológiami virtuálnej reality.

Podľa [5], v období od minulého desaťročia až po súčasnosť sa školy postupne digitalizujú, využívajú sa počítače, tablety, interaktívne tabule a rôzne ďalšie technologické prostriedky. Ich účelom má byť predovšetkým zvýšenie kvality vzdelávania, ale aj študentov motivovať, upútať ich pozornosť, viesť k novým objavom a podobne. Využitie virtuálnej reality vo vzdelávacom procese preto môžeme považovať za akýsi prirodzený vývoj vo vzdelávacích procesoch. Virtuálna realita dokáže poskytnúť študentovi zážitok z pohľadu prvej osoby, najmä v prípadoch, kedy takýto zážitok inak navodiť nevieme, alebo nechceme. Napríklad z bezpečnostných dôvodov dokážeme využiť virtuálny priestor pre rôzne experimenty, ktoré sú v bežnom svete nebezpečné. Využitie VR vo vzdelávaní má množstvo výhod, no netreba však zabúdať aj na určité nevýhody, či možné riziká, napríklad riziko možného zranenia sa študenta.

Keďže tradičné metódy výučby vyžadujú malú, alebo žiadnu interakciu študenta, jeho pozornosť sa postupne vytráca. Aj preto sa v implementačnej časti tejto práce budeme zameriavať na opravy a vylepšenie existujúceho systému. Úlohy by mali udržať študentovu pozornosť, mať pútavý charakter, zlepšiť celkový zážitok študenta a priniesť pozitívny prístup k danej tématike.

Cieľom tejto práce bude nadviazať na prácu Adama Kašelu, ktorá je založená na báze webových technológií a venuje sa alternatívnemu prístupu ku výuke dejepisu pre 7. ročník základných škôl za využitia Petriho siete, a na ktorú už nadviazali Dmytro Damienko a Lukáš Písařík. Budeme sa venovať možnostiam rozšírenia ich prác, návrhu implementácie daných rozšírení.

Formulácia úlohy

Cieľom tejto práce je zamerať sa na opravy a vylepšenie existujúceho systému. Ide o systém zameraný na spojenie virtuálnej reality a vzdelávanie žiakov na školách. Tento systém využíva Petriho siete na návrh scenárov aktivít pre virtuálne prostredia.

V 2. kapitole tejto práce sa pokúsime analyzovať možnosti zapojenia pedagógov do tvorby scenárov vo forme Petriho sietí.

V 3. kapitole sa budeme venovať analýze aktuálneho systému, identifikácii možných nedostatkov a návrhu ich opráv či vylepšení.

Po vykonaní tejto analýzy je potrebné aktuálne riešenie nasadiť na server, keďže aktuálna verzia nie je v implementovaná v žiadnom reálnom systéme, zároveň vymyslieť jednoduchý systém na aktualizovanie aplikácie na servery v prípade novo vznikajúcej implementácii. Túto implementáciu si popíšeme v kapitole číslo 4. V 5. kapitole sa pokúsime navrhované opravy a vylepšenia implementovať, ak to bude možné. V závere budeme vyhodnocovať čo sa v tejto práci podarilo dosiahnuť, prípadne aj čo sa nepodarilo s vysvetlením prečo nebolo možné navrhovanú zmenu implementovať.

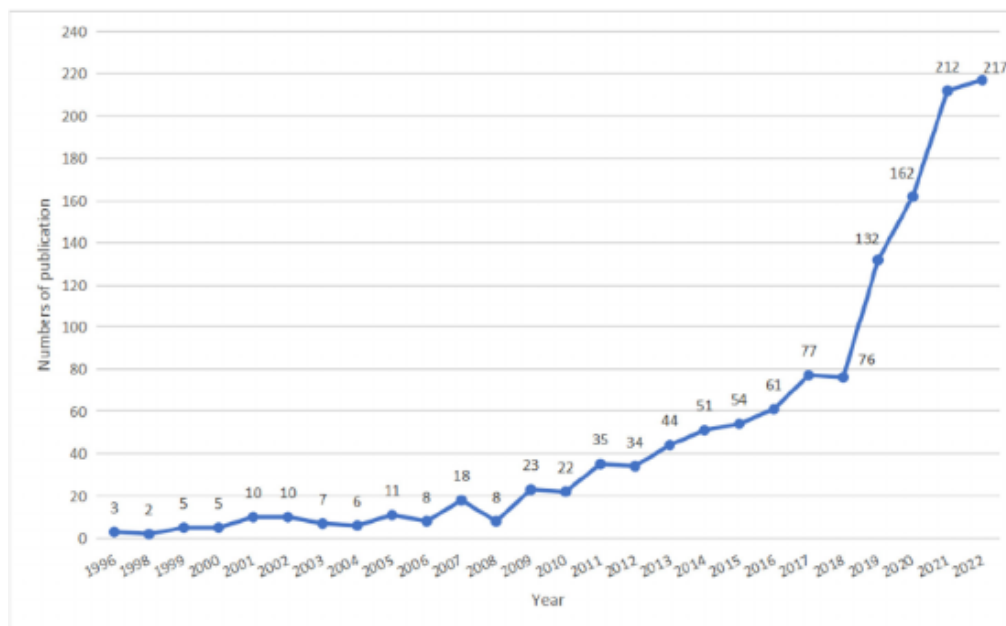
1 Vzdelávanie vo virtuálnom prostredí

Vďaka pokroku vo vzdelávacích technológiách môžu žiaci na internete nájsť obrovské množstvo vzdelávacích zdrojov. Žiaci už nemajú jediný zdroj prístupu k vedomostiam a vyučovacie metódy učiteľov idú v stopách doby. Počas pandémie COVID-19 bolo všetko vyučovanie online. Prvotné online vzdelávanie bolo jednotvárne a nudné a študentom a učiteľom chýbala interakcia. Aplikácia virtuálnej reality do online vzdelávania poskytuje študentom väčšiu motiváciu a radosť a zlepšuje zážitok z učenia. Technológia virtuálnej reality môže študentom poskytnúť bohaté perцепčné podnety a viackanálovú spätnú väzbu, čo umožňuje situačné učenie.

Inovácie v metódach vzdelávania prinášajú lepšie výsledky vzdelávania. S príchodom informačného veku už nie je jednoduché naučiť sa podstatu vedomostí tradičným indoktrinačným a prednáškovým štýlom výučby. Do popredia sa postupne dostáva imersívne, teda pohlcujúce učenie. Napríklad v [6] vedci porovnávali koncepciu učenia sa študentov o relatívnom pohybe pomocou počítačových simulácií v podmienkach imersívneho virtuálneho prostredia (IVE) a neimersívneho desktopového virtuálneho prostredia (DVE) a zistenia ukazujú, že IVE môže uľahčiť pochopenie abstraktných vedeckých javov. V súčasnosti, v dobe informácií, je inteligentné vzdelávanie lídrom zmien a inovácií vo vzdelávaní.

1.1 Vývoj v oblasti vzdelávania vo virtuálnom prostredí

Virtuálne realitné systémy, ako odvetvie IKT (informačno-komunikačné technológie) sa bezpochyby v posledných rokoch tlačí do popredia. Podľa štúdie [7] z roku 2023 počet publikovaných štúdií v oblasti vzdelávania pomocou imersívnej virtuálnej reality z roka na rok narastá. Časový horizont, ktorý bol posudzovaný z hľadiska publikácií začína rokom 1996 až po rok 2022. Toto obdobie následne delíme do troch etáp.



Obr. 1.1: Graf medziročného nárastu počtu publikácií zameraných na vzdelávanie vo VR

1.1.1 Prvá etapa (1996–2008)

Z grafu 1.1 vieme jasne vyčítať, že počet publikácií ročne je bolo tomto období na nízkej úrovni. No už v tejto etape sa objavili prvé pozitívne výsledky, napríklad prínos VR v neuropsychológii, pri poruchách učenia a neurologických poruchách. Ukázalo sa, že VR dokáže zvýšiť rýchlosť hodnotenia kognitívneho a funkčného správania a rehabilitácie.

1.1.2 Druhá etapa (2009–2016)

V tomto období už počet štúdií stabilne narastá každým rokom. V priemere už ide o približne 40 publikovaných prác ročne, predovšetkým z USA. Možeme konštatovať, že v tomto období dochádza k posunu od animácií a obyčajných multimédií ku aplikačným formám VR vzdelávacích systémov.

1.1.3 Tretia etapa (2017–2022)

Za posledných 6 rokov sa literatúra v tejto oblasti enormne rozrástla. Pri až 146 publikáciách v priemere ročne ide o približne 2,6 násobne viac publikácií než v predošlej etape. Do popredia pozornosti sa ku konci tohto obdobia dostáva edukačný metaverse. Metaverse spája fyzickú a virtuálnu realitu a poskytuje kanály na multi-senzorickú interakciu a ponorenie sa do rôznych prostredí. Imersívne

učenie vo virtuálnej realite sa stalo trendom budúceho vzdelávania. S pribúdajúcou literatúrou je výskum imersívneho vzdelávania vo virtuálnej realite čoraz viac diverzifikovaný.

1.2 Výhody vzdelávania vo virtuálnom prostredí

Podľa dostupnej štúdie[6] existuje istá spojitosť medzi virtuálnymi technológiami a zlepšením akademického výkonu študentov, ich motivácie, sociálnych aj koloaboratívnych zručností a psychomotorických a kognitívnych zručností študentov. Aj v prípadoch, kedy simulácia či virtuálne prostredie predstavujú, alebo imitujú skutočný svet a životné situácie, existujú možnosti, ktoré dávajú študentom možnosť skúmať nové oblasti, experimentovať či interpretovať výsledky.

1.3 Limitácie vzdelávania vo virtuálnom prostredí

Do vzdelávacieho prostredia sa postupne dostávajú deti, ktoré vyrastali v obklopení moderných technológií, ako napríklad smartfónov, počítačov, televízorov a podobne. Pre tieto generácie je preto prirodzenejšia interakcia s týmito technológiami oproti starším generáciám. To však ale automaticky nedáva študentom kompetentnosť správne používať príslušné virtuálno-vzdelávacie prostriedky.

Používanie nových inovatívnych technológií nutne nezahŕňa pedagogické inovácie, čo znamená, že stále je tu potreba návrhu vzdelávacích prostredí aj pedagogických prostriedkov a rozhraní tak, aby sa maximalizovali výsledky vzdelávacieho procesu[6].

1.4 Trendy vo vzdelávaní vo virtuálnom prostredí

Z počiatku sa vzdelávacie VR systémy využívali predovšetkým v špecifických sektoroch ako napríklad letecký priemysel, alebo jadrová energetika a to hlavne z dôvodov vysokej ceny a nízkej dostupnosti. K celkovému zlepšeniu podmienok prispelo niekoľko faktorov, a to napríklad výkon a možnosti novších mobilných zariadení, zvýšené investície v oblasti vývoja virtuálnych systémov, alebo prístup k virtuálnemu obsahu vytvorenému používateľmi prostredníctvom sociálnych sietí[6]. To všetko sa podieľalo, aj sa ďalej podieľa, na vyššej cenovej dostupnosti aj všeobecne vyššej dostupnosti k rôznym druhom VR zariadení a systémov. V dôsledku toho bude možné vykonávať imersívne zážitky prostredníctvom interakcie s objektmi, konceptmi alebo procesmi ako bežný pracovný postup učenia

na akejkolvek úrovni vzdelávania, od základnej školy až po vysokoškolské vzdelávanie. Aktuálne sa často objavuje pojem metaverse, a to nie len pri samotných VR technológiách, ale aj vzdelávacích VR prostrediach.

1.5 Metaverse

Metaverse predstavuje počítačom generovaný, geometricky trojrozmerný (3D) virtuálny svet, ktorý používateľom umožňuje interakciu medzi sebou navzájom, ako aj s digitálnym obsahom a objektmi v reálnom čase, ako keby boli v skutočnom svete, pričom zároveň prekonáva obmedzenia vzdialenosti a časových rozdielov. Na jeho realizáciu sa používajú XR technológie v kombinácii, aby používateľom poskytli virtuálnu prítomnosť v plne digitálnom prostredí. Metaverse môžeme charakterizovať ako realistické digitálne rozšírenie fyzického sveta, ktoré používateľov ponorí do zdieľaných a paralelných virtuálnych svetov prostredníctvom teleprítomnosti [8].

Vzhľadom na svoju povahu sa prijatie a implementácia metaverza vo vzdelávacích prostrediach na všetkých úrovniach vzdelávania stáva čoraz populárnejším[9]. Najmä v týchto imersívnych virtuálnych prostrediach, ktoré využívajú herné prvky, sa študenti môžu učiť pútavejším, zábavnejším a interaktívnejším spôsobom, pričom si súčasne zlepšujú svoju predstavivosť, individuálnu a kolektívnu inteligenciu, ako aj zlepšujú svoju krátkodobú pamäť[10]. Metaverse poskytuje flexibilné, rozmanité, škálovateľné a dynamické vzdelávacie prostredie, ktoré podporuje motiváciu a zapojenie študentov, pretože ponúka pohlcujúcejšie a interaktívnejšie vzdelávacie skúsenosti, podporuje aktívnu komunikáciu a spoluprácu, uľahčuje vykonávanie synchronných aj asynchronných procesov učenia a vyučovania.

Okrem toho, ak sa metaverse používa spôsobom zameraným na študentov, má potenciál zlepšiť ich zručnosti pri riešení problémov, zlepšiť kritické myslenie, zlepšiť ich akademický výkon a študijné výsledky, ako aj zvýšiť ich porozumenie predmetu a v dôsledku toho vytvoriť dynamické vzdelávacie prostredie vyššej kvality[11].

2 Porozumenie Petriho sieťam neodbornou verejnosťou

Petriho siete sú významným nástrojom v oblasti modelovania systémov. V tejto kapitole sa budeme venovať vysvetleniu základných princípov Petriho sietí a tiež sa budeme snažiť zistiť, ako ľudia s a bez technického vzdelania dokážu Petriho sieťam porozumieť.

Motivácia pre túto kapitolu spočíva v tom, že v našej práci chceme pripraviť systém tak, aby si učitelia, ktorí budú s našim systémom pracovať, vedeli jednoduché Petriho siete pripravovať, prípadne upravovať aj sami, ak to bude možné.

2.1 Definícia Petriho sietí

Petriho sieť je bipartitný orientovaný graf s dvoma typmi vrcholov, miesta a prechody. Tie sú navzájom spájané orientovanými hranami, ktoré smú vždy spájať len miesto s prechodom, alebo prechod s miestom. Primárne sa Petriho siete využívajú na modelovanie a simulácie, ale aj na reprezentáciu toku dát, analýzy a rôzne iné účely. Miesta môžu obsahovať tokeny. Tie definujú aktuálny stav modelovanej situácie a reprezentujú dáta. V grafickej podobe majú miesta vždy tvar kružnice/elipsy a prechody majú tvar obdĺžnika[12].

2.2 Formulácia výskumných otázok

Ako už bolo spomenuté, cieľom tejto analýzy je zistiť, ako dokážu Petriho sieťam porozumieť ľudia, ktorý nemajú technické vzdelanie, alebo technicky zamerané zamestnanie. Prípadnou ďalšou motiváciou je ešte zistiť, či sa vyplatí učiť takejto skupine ľudí, teda v našom prípade učiteľom, o základnej problematike Petriho sietí. Položíme si 2 otázky, ktoré sú definované v tabuľke 2.1.

Otázka	Motivácia
O1. Ako dokážu bežný ľudia porozumieť Petriho sieťam	Motiváciou je nájsť výskum, prípadne prieskum o tom, ako náročné je pre bežných ľudí porozumieť Petriho sieťam
O2. Má pre bežných ľudí význam sa učiť Petriho siete	Motiváciou je nájsť prieskum, alebo zhodnotenie, aký význam môžu mať Petriho siete pre človeka

Tabuľka 2.1: Tabuľka výskumných otázok

2.3 Postup získavania dát

Po formulácii výskumných otázok je potrebné vyhľadať zdroje, v ktorých by sme dokázali nájsť odpovede na položené otázky.

Potrebné zdroje sme sa snažili vyhľadať na nasledujúcich databázach a vedeckých portáloch: Web of Science, Scopus, IEEE Xplore, Google scholar, Researchgate. V tabuľke 2.2 sú dopyty, ktoré sme vyhotovili pre vyhľadanie potrebných zdrojov nad jednotlivými databázami.

Databáza	Dopyt
Web of Science	TS= (Petri net AND (comprehension OR understanding OR usability OR worth learning))
Scopus	TITLE-ABS-KEY ('Petri net' AND (comprehension OR understanding OR usability OR 'worth learning'))
Google scholar	"Petri net" AND (comprehension OR understanding OR usability OR "worth learning")
IEEE	TITLE-ABS-KEY ('Petri net' AND (comprehension OR understanding OR usability OR 'worth learning'))
Researchgate	Petri net AND (comprehension OR understanding OR usability OR worth learning)

Tabuľka 2.2: Tabuľka dopytov na databázy vedeckých portálov

S výsledkov jednotlivých dopytov sme nenašli žiadnu zmienku o tom, či sa niekto zaoberal otázkou porozumenia ľudí problematike Petriho sietí. Z tohto dôvodu sme sa rozhodli vyhotoviť si vlastný prieskum.

2.4 Vyhodenie vlastného prieskumu

V tejto sekcii popisujeme spôsob realizácie nášho vlastného prieskumu zameraného na pochopenie Petriho sietí medzi bežnými ľuďmi v porovnaní s ľuďmi, ktorí sa profesne či študijne venujú technickej oblasti, alebo informatike a výpočtovej technike a iným príbuzným odborom, kde by sme mali očakávať lepšie chápanie problematiky. Na prípravu úlohy pre respondentov sme použili nástroj CPN Tools[13].

2.4.1 Výber respondentov

Našu vzorku respondentov sme si vybrali a rozdelili do dvoch skupín. Pre zjednodušenie tieto dve skupiny pomenujeme ako technická a netechnická skupina ľudí.

1. Technická skupina

Respondenti v tejto skupine mali technické znalosti alebo skúsenosti, ktoré by mohli uľahčiť porozumenie Petriho sietí.

2. Netechnická skupina

Táto skupina zahŕňala respondentov bez predchádzajúcich technických znalostí, ktoré by mohli umožniť rýchle porozumenie Petriho sieťam

2.4.2 Štruktúra prieskumu

Prieskum sme rozdelili do dvoch častí a to na prípravnú časť a časť s úlohou. V prípravnej časti sme sa snažili respondentom čo najjednoduchšie vysvetliť základnú problematiku Petriho sietí tak, aby sme im poskytli dostatok vedomostí pre riešenie úlohy, ktorá je v ďalšej časti prieskumu.

Prípravná časť

V tejto časti prieskumu sme respondentom poskytli základné informácie o Petriho sieťach vrátane definície, príkladov a spôsobu ich konštrukcie. Cieľom tejto časti bolo poskytnúť účastníkom primerané základné vedomosti potrebné na pochopenie úloh v prieskume. Aby sme respondentom uľahčili ich úlohu, obmedzili sme sa na také Petriho siete, kde do prechodu a z prechodu môže smerovať vždy najviac jedna hrana. Znenie prípravnej časti vyzerá nasledovne:

Zamestnanie/kariérne zameranie: _____

Ukončený/prebiehajúci stupeň štúdia a odbor: _____

Prieskum - Porozumenie problematiky Petriho Sieťí

Čo to je Petriho sieť?

Ide o typ konštrukčného modelu, ktorý môže byť užitočný pri analýze údajov a toku dát, simuláciách, modelovaní procesov a mnoho ďalších scenároch. Petriho siete, podobne ako vývojové diagramy, vieme využiť ako vizuálnu pomôcku pre opis informácií vo forme grafu.

Z čoho pozostáva Petriho sieť?

Pri modelovaní Petriho sieťí používame 2 základné prvky, ktorými sú *miesta* a *prechody*.

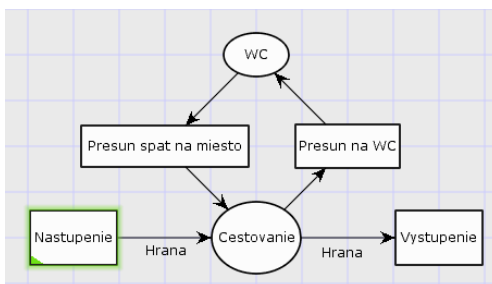
- *Miesta* – reprezentujú miesto, stav, prípadne určitú podmienku (kružnice/elipsy)
- *Prechody* – znázorňuje určitú udalosť (obdĺžniky)

Pravidlá konštruovania PS:

Prechody a miesta navzájom spájame orientovanými hranami (šípkami), nie je však možné prepojiť 2 prechody, alebo 2 miesta. Hrany vyjadrujú tok dát. Keďže prechody vyjadrujú určitú udalosť, v Petriho sieti máme možnosť takéto prechody „odpáliť“, t.j. danú udalosť vykonať. Pre prechody platí, že do prechodu a z prechodu môže smerovať vždy len jedna hrana (dohromady 2 hrany – jedna do a jedna z prechodu).



Jednoduchý príklad Petriho siete

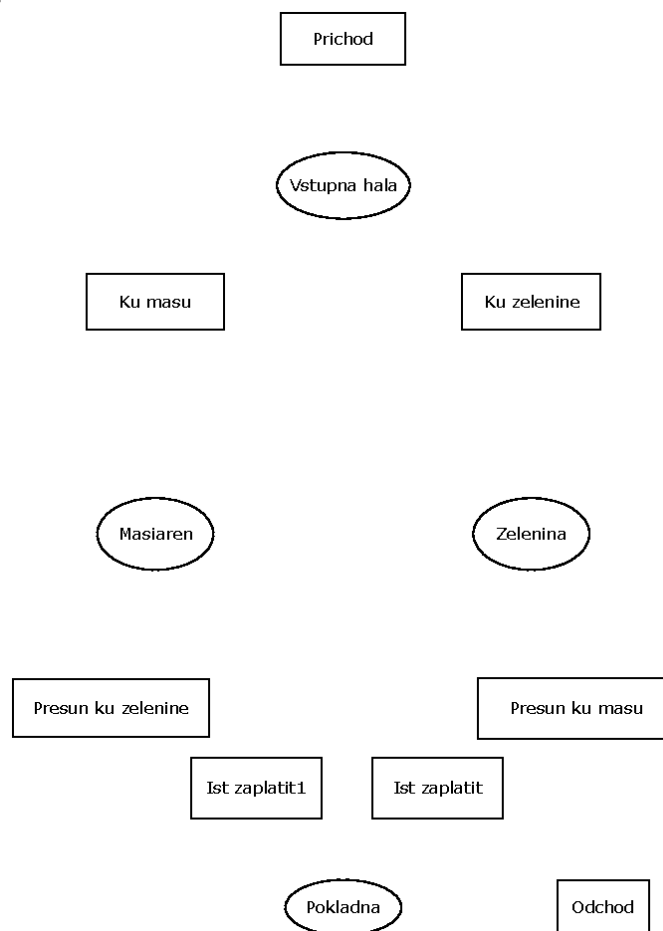


Časť s úlohou

Respondenti v tejto časti riešia úlohu, pri ktorej mali doplniť orientované hrany medzi miestami a prechodmi v zadanej Petriho sieti. Petriho sieť sme sa snažili zostaviť tak, aby popisovala situáciu respondentom známu z reálneho života. Táto časť prieskumu mala overiť úroveň pochopenia Petriho sietí medzi respondentmi oboch skupín.

Úloha:

Na obrázku nižšie sú miesta a prechody bez potrebných orientovaných hrán. Vašou úlohou je dokresliť všetky potrebné hrany do Petriho siete, ktorá má predstavovať proces nakupovania v obchode, v ktorom máme 2 sekcie: Mäsiareň a zelenina. Po príchode do vstupnej haly by mal zákazník mať možnosť vstúpiť do jednej zo sekcií, následne keď zákazník opúšťa sekciu v ktorej sa nachádza, mal by mať možnosť ísť zaplatiť, alebo navštíviť druhú sekciu. **Je potrebné dodržať spomínané pravidlá, ktoré platia pre Petriho siete.**



Nápoveda: Hrany nesmú spájať 2 prechody, alebo 2 miesta. Do prechodu a z prechodu môžu smerovať vždy len jedná hrana.

2.4.3 Zbieranie údajov

Po absolvovaní prípravnej časti a úlohy sme zbierali nasledujúce údaje o respondentoch:

- Vek
- Pohlavie
- Zamestnanie/kariérne zameranie
- Ukončený/prebiehajúci stupeň štúdia a odbor
- Čas prípravy
- Čas riešenia úlohy

Dôležitá vec, ktorú sme sa respondentov v oboch skupinách pýtali je, že či sa už niekedy s Petriho sieťami stretli, alebo ich už v minulosti riešili. To, že sa s tým respondentmi nestretli bola zásadná podmienka pri ich výbere, a preto ak sme narazili na takého potenciálneho respondenta, jeho účasť v prieskume sme okamžite vylúčili.

2.4.4 Metodika hodnotenia výskumnej úlohy

Táto úloha nie je celkom štandardná v porovnaní s prieskumami, kde respondent si často vyberá z možností, alebo vyslovuje svoj názor. Chyby, ktoré je v nej možné vykonať sa rôznia vo svojej povahe a kritickosti, preto nedokážeme použiť bodovací systém, ako napríklad pri štandardnom teste s otázkami a odpoveďami, kde za dobrú odpoveď by sme udelili bod/y, alebo v prípade zlej odpovede by sme neudelili žiaden bod, prípadne bod mohli strhnúť. Preto si jednotlivé chyby rozdelíme do kategórií podľa ich charakteru.

- **Chýbajúce a nadbytočné hrany**

Respondentom v niekoľkých prípadoch chýbali potrebné hrany pre správne vyriešenie úlohy. Naopak, objavilo sa aj riešenie, kde hrany boli prakticky duplicitné. v prípade duplicitných hrán došlo zároveň k porušeniu pravidla, že do a z prechodu smie smerovať vždy len jedna hrana. Toto nie je nevyhnutne chyba, no v zadanej úlohe takéto hrany neboli potrebné, naopak toto pravidlo sme uviedli s cieľom zabezpečiť, aby úloha nebola príliš komplikovaná.

- **Nelogické spojenie prechodov a miest**

V tomto prípade sa jedná o chybu, kde technicky sú potrebné hrany umiestnené správne, t.j. spojenie prechodu a miesta. Ide však o to, že logicky tieto spojenia nedávajú zmysel. Ako príklad môžeme uviesť spojenie prechodu “ku zelenine” a miesta “mäsiareň”.

- **Spojenie dvoch prechodov, alebo dvoch miest**

Ide pravdepodobne o najzávažnejšiu chybu, ktorá porušuje základné princípy konštruovania Petriho sietí.

2.4.5 Analýza zozbieraných údajov

V tejto sekcii si zozbierané dáta analyzujeme s cieľom porovnať výkonnosť a pochopenie Petriho sietí medzi technickou a netechnickou skupinou respondentov.

Riešiteľ	Vek	Pohlavie	Zamestnanie/kariérne zameranie	Ukončený/prebiehajúci stupeň štúdia a odbor
N1	38	Ž	účtovníčka	VŠ (II. st.) - stavebná
N2	56	Ž	referent	SŠ - ekonomická
N3	38	M	predajca	SŠ - učňovka
N4	47	Ž	sociálna pracovníčka	VŠ (III. st.) - soc. práca
N5	35	Ž	vedúca opatrovateľského odd.	VŠ (II. st.) - verejná správa
N6	52	Ž	Perosnalistka	SŠ - ekonomická
N7	38	M	manažér	VŠ (I. st.) - soc. práca
N8	48	M	technický vedúci	VŠ (II. st.) - verejná správa
N9	26	M	Obchodný zástupca	SŠ - športový manažment
N10	24	Ž	Nezamestnaný - študent	VŠ (II. st.) - žurnalistika

Tabuľka 2.3: Zozbierané dáta o netechnickej skupine

V tabuľke 2.3 máme zobrazené dáta desiatich takzvaných netechnických respondentov. Z tejto tabuľky vyplýva, že naši respondenti dosahujú priemerný 40,2 rokov, pomer mužov ku ženám je 4-6 a až 6 respondenti dosiahli určité vysokoškolské vzdelanie.

Riešiteľ	Vek	Pohlavie	Zamestnanie/kariérne zameranie	Ukončený/prebiehajúci stupeň štúdia a odbor
T1	24	M	Nezamestnaný - Študent	VŠ (II. st.) - kyberbezpečnosť
T2	24	M	IT podpora	VŠ (II. st.) - kyberbezpečnosť
T3	25	M	software developer	VŠ (II. st.) - informatika
T4	25	Ž	software developer	VŠ (II. st.) - informatika
T5	23	M	software developer	VŠ (II. st.) - informatika
T6	24	M	Junior Cybersecurity analyst	VŠ (II. st.) - kyberbezpečnosť
T7	27	M	software developer	VŠ (II. st.) - informatika
T8	29	M	software developer	VŠ (II. st.) - informatika
T9	23	M	software developer	VŠ (II. st.) - informatika
T10	50	M	Product owner	VŠ (II. st.) - informatika

Tabuľka 2.4: Zozbierané dáta o technickej skupine

V tabuľke 2.4 môžeme vidieť dáta desiatich takzvaných technických respondentov. Z tejto tabuľky vyplýva, že naši respondenti dosahujú priemerný vek 27,4 rokov, pomer mužov ku ženám je 9-1, čo je pravdepodobne spôsobené hlavne tým, že odvetviam informačno-komunikačným technológiám sa vo vyššej miere venujú práve muži. V tomto prípade všetci respondenti buď dosiahli určité vysokoškolské vzdelanie, alebo ešte stále študujú.

Keďže v nasledujúcich tabuľkách, ktoré sa venujú analýze chýb respondentov oboch skupín, by názvy vyššie spomínaných kategórií boli príliš dlhé, vytvorili sme pre nich skratky, ktoré sú uvedené v tabuľke 2.5.

Celý názov chyby	Skratka
Chýbajúca hrana, nadbytočná hrana	Chyba A
Nelogické spojenie prechodov a miest	Chyba B
Spojenie dvoch prechodov, alebo dvoch miest	Chyba C

Tabuľka 2.5: Tabuľka skratiek pre jednotlivé typy chýb

V tabuľkách nižšie môžeme vidieť, ako sa jednotlivým skupinám darilo pri riešení zadanej úlohy.

Riešiteľ	Čas prípravy	Čas riešenia	Čas celkom	Počet chýb A	Počet chýb B	Počet chýb C	Úspešnosť
N1	1:50	7:49	9:39	4	0	0	Neúspeš/a
N2	1:30	3:31	5:01	7	1	1	Neúspeš/a
N3	3:46	6:50	10:36	4	0	0	Neúspeš/a
N4	1:24	3:56	5:20	0	0	0	Úspeš/a
N5	1:58	8:29	10:27	0	0	0	Úspeš/a
N6	1:27	2:19	3:46	0	0	0	Úspeš/a
N7	2:18	5:27	7:45	5	1	2	Neúspeš/a
N8	0:44	5:42	6:26	4	0	2	Neúspeš/a
N9	2:09	4:27	6:36	0	0	0	Úspeš/a
N10	2:26	7:42	10:08	0	0	0	Úspeš/a

Tabuľka 2.6: Výsledky netechnickej skupiny

Podľa tabuľky 2.6 vidíme, že táto skupina mala pomerne častý problém nájsť všetky potrebné hrany, prípadne sa vyskytol problém, kedy boli dokreslené prebytočné hrany. V súčte sa respondenti dopustili týchto chýb až 24 krát. Dvaja respondenti sa následne dopustili aj chyby, kedy spojenie miesta a prechodu síce bolo technicky v poriadku, ale bolo nelogické z hľadiska požadovaného scenára. Posledný typ chyby sa objavil až 5 krát, kedy boli orientovanými hranami spájané prechody. Zaujímavosťou je, že v ani jednom prípade respondenti nespojili orientovanou hranou 2 miesta.

Respondentom v tejto skupine trvala príprava v priemere 1 minútu 57 sekúnd, riešenie úlohy trvalo v priemere 5 minút 37 sekúnd. Priemerný čas celkovo dosiahol v tejto skupine 7 minút 34 sekúnd.

Riešiteľ	Čas prípravy	Čas riešenia	Čas celkom	Počet chýb A	Počet chýb B	Počet chýb C	Úspešnosť
T1	0:59	3:43	4:42	0	0	0	Úspeš/a
T2	1:40	6:31	8:11	0	0	0	Úspeš/a
T3	2:04	1:40	3:44	0	0	2	Neúspeš/a
T4	2:04	1:55	3:59	0	0	0	Úspeš/a
T5	1:43	3:56	5:39	0	0	0	Úspeš/a
T6	1:41	3:43	5:24	0	0	0	Úspeš/a
T7	2:14	4:58	7:12	1	0	0	Neúspeš/a
T8	1:52	3:25	5:17	0	0	0	Úspeš/a
T9	1:06	1:59	3:05	0	0	0	Úspeš/a
T10	1:39	3:50	5:29	1	0	0	Neúspeš/a

Tabuľka 2.7: Výsledky technickej skupiny

V tabuľke 2.7 vidíme výsledky respondentov našej druhej skupiny, kde hneď z výsledkov vyplýva, že sa im darilo s riešením úlohy omnoho lepšie. Vo dvoch prípadoch došlo ku porušeniu pravidla s chýbajúcou, alebo nadbytočnou hranou a v jednom prípade došlo k spojeniu dvoch prechodov. V tomto jednom prípade si však dovoľíme konštatovať, že logicky riešiteľ úlohu pochopil veľmi správne, no k zlyhaniu došlo až úplne na konci a to tak, že prechod “presun ku zelenine” spojil s prechodom “ku mäsu” a naopak, čím vlastne chcel vyjadriť túto možnosť prechádzať medzi sekciami. No napriek tomu je to chyba ako každá iná.

Respondentom v tejto skupine trvala príprava v priemere 1 minútu 42 sekúnd, riešenie úlohy trvalo v priemere 3 minúty 34 sekúnd. Priemerný čas celkovo dosiahol v tejto skupine 5 minút 16 sekúnd.

2.4.6 Vyhodnotenie prieskumu

Ako sme mohli vidieť v tabuľkách s výsledkami, technická skupina dosiahla vyššiu úspešnosť pri riešení úlohy ako netechnická skupina, v ktorej uspela presne polovica respondentov. Napriek tomu, že táto skupina dosiahla horšie výsledky, je tu istý potenciál na zapojenie neodborníkov do spolupráce v tejto oblasti. Respondentom, ktorý sa dopustili chýb, sme následne ukázali ich chyby a aj im ich zdôvodnili. Väčšina týchto respondentov porozumela tomu, kde sa dopustili chýb a mnoho z nich bolo presvedčených, že prípadnú ďalšiu podobnú úlohu by už dokázali vyriešiť.

Dôležitým predpokladom na takúto spoluprácu je aj ochota prípadných adeptov sa tejto téme venovať, mať k nej pozitívny prístup. To je taktiež vec, ktorú sme postrehli u respondentov v čase ich riešenia úlohy. Bolo vidieť, že respondenti, ktorí sa nad úlohou poriadne zamysleli, dokázali úlohu vyriešiť, alebo sa k vyriešeniu čo najviac priblížiť. Respondenti s opačným prístupom, kedy môžeme povedať, že ich cieľom bolo čo najskôr úlohu dokončiť, mali zväčša vysoký počet chýb.

3 Analýza súčasného stavu projektu

V rámci tejto kapitoly sa pokúsime zhodnotiť aktuálny stav aplikácie, ktorú posledne rozšírili Lukáš Písarčík[14] a Dmytro Demianenko[15] vo svojich bakalárskych prácach. Obaja naviazali na diplomovú prácu Adama Kašelu[16]. Lukášova práca bola primárne zameraná na implementovanie Petriho siete pre vytvorenie univerzálneho ovládania scény, použiteľnú pre akúkoľvek scénu, ale aj implementácii všeobecnej funkcionality týkajúcej sa interakcie siete s virtuálnym prostredím. Dmytrova práca bola zase zameraná na vytvorenie systému, ktorého cieľom je zhromažďovať údaje o používateľoch počas plnenia úloh a poskytovať podrobné štatistiky o splnených úlohách.

3.1 Použité technológie

Celý projekt je postavený na docker kontajneroch. To nám jednoducho umožňuje izolovať jednotlivé časti systému a tak uľahčiť spravovanie jednotlivých komponentov, ako sú backend, či frontend[17].

Práve pre spomínaný frontend využívame Angular, ako webový rámec, ktorý nám umožňuje vytvárať dynamický a responzívny web[18]. Pre vytváranie virtuálnych prostredí využívame rámec A-Frame, ktorý využíva vlastné HTML tagy na jednoduchú implementáciu virtuálneho prostredia. Ide o nadstavbu knižnice three.js, ktorá slúži na zobrazovanie 3D priestorov a animácií.

V našom projekte využívame aj databázový systém. Na to, aby frontendový komponent dokázal komunikovať s databázou, prípadne aby sme vedeli vykonať istú funkcionality, máme vytvorený backend, ktorý je postavený na technológií Java Spring[19].

3.2 Scenárom riadené výučbové prostredia vo virtuálnej realite

Táto sekcia je zameraná na analýzu implementácie práce Lukáša Pisarčíka. Stanovené ciele tejto práce sa zameriavajú na implementáciu Petriho siete, scénu a komunikáciu prostredia so serverom.

3.2.1 Petriho siete

Implementovanie Petriho sietí je v tomto projekte nápomocné z hľadiska návrhu scenárov a riadenia virtuálnych prostredí. V tejto práci sa podarilo implementovať ovládanie samotnej scény pomocou Petriho sietí, realizáciu obsluhy, ukončovanie úloh, načítavanie Petriho sietí a vykonávanie obsluhy.

3.2.2 Rozšírenie scény

V práci sa podarilo rozšíriť pôvodnú scénu. Rozšírila sa o rôzne technické prvky, ale zväčšila sa aj samotná galéria ako naša scéna. V galérii pribudlo ovládacie centrum, kde sú zobrazené stavy jednotlivých úloh, ale aj ovládacie tlačidlá. Pribudli popisky Petriho sietí pri jednotlivých prvkoch v scéne.

3.2.3 Komunikácia so serverom

V rámci tejto práce bola snaha vylepšiť pôvodne jednoúčelovú aplikáciu na univerzálny systém napríklad tým, aby si používateľ mohol načítať vlastnú scénu, scenár či jazykový súbor, v ktorom sa nachádzajú texty, ktoré chceme v danej scéne zobraziť. Mohli by to byť napríklad otázky, či informačné okná. Využijeme to najmä pri monitorovaní a analyzovaní správania sa používateľov.

3.2.4 Ovládanie pomocou VR ovládačov

Do scény bol pridaný komponent super-hands, ktorý poskytuje prirodzenú interakciu ručného ovládania. Implementované gestá sú Hover, Grab, Stretch a Drag-drop.

3.3 Spracovanie údajov z virtuálnych výučbových prostredí

Táto sekcia sa zameriava na analýzu implementácie práce Dmytra Demianenka. Ciele tejto práce sú zamerané primárne na webovú časť projektu, proces autentifikácie a autorizácie používateľov, spracovanie databáz a pod.

3.3.1 Implementácia autentifikácie

Pre implementáciu autentifikácie sú v tejto práci použité registračný a autentifikačný formulár, kde sa vyžadujú osobné údaje používateľa, ako napríklad meno, e-mail, heslo a podobne. Vyžadované je použiť jedinečný e-mail, ktorý ešte v systéme nie je evidovaný, a heslo, ktoré je potrebné zopakovať. Tým sa zamedzilo nožnej duplicitě používateľských účtov. K autentifikačnej časti sa podarilo vytvoriť aj systém overovania používateľom pomocou overovacieho e-mailu.

3.3.2 Implementácia autorizácie

Po úspešnej autentifikácii a overení sa už registrovaný používateľ dokáže prihlásiť do systému a autorizovať sa. Používateľom sú povolené iba tie možnosti, ktoré im ich rola povoľuje.

3.3.3 Funkcionality webovej aplikácie

Táto časť nadväzuje na tú predošlú hlavne v tom, že webové prostredie je prispôsobené konkrétnym rolám. Ide predovšetkým o privilegované role, v tomto prípade učitelia a správcovia, ktoré majú niekoľko rozšírených možností na správu svojho vyučovacieho procesu.

- **Pridávanie scenárov**

Scenáre sú dôležitou súčasťou tohto projektu, kde s využitím Petriho sietí definujeme scenár pre plnenie úloh. Vo webovom prostredí je vytvorený formulár pre učiteľov a správcov na pridanie scenára, kde je potrebné priložiť aj súbor .zip obsahujúci potrebné súbory.

- **Pridávanie scén**

Scény môžu rovnako pridávať len učitelia a správcovia. Takisto je vytvorený formulár pre pridanie scény, kde je nutné priložiť potrebný súbor definujúci danú scénu.

- **Vytvorenie úloh**

Učiteľ má možnosť pridávať úlohy, k tomu je však potrebné vybrať si scenár a scénu pre danú úlohu.

- **Správa používateľov**

System dáva možnosť učiteľom a správcom aktualizovať a odstraňovať používateľské kontá. Tu majú učitelia možnosť deliť študentov do skupín, prípadne odstraňovať kontá zo systému.

- **Riadenie skupiny**

Každý používateľ, môže byť členom určitej skupiny. Každé skupine môžu byť pridelené rôzne úlohy. Učiteľ má možnosť vytvárať nové skupiny a začleňovať používateľov do príslušných skupín.

- **História relácií v skupinách**

Učitelia a správcovia si dokážu prezerať štatistiky skupín o riešení pridelených úloh. Štatistiky obsahujú počet správne aj nesprávne vykonaných akcií, percento správnych akcií aj čas strávený v relácií riešením úloh.

- **Spustenie úloh**

Toto je už funkcionality predovšetkým pre študentov. Na domovskej stránke je dostupné okno, kde si môžu používatelia vybrať pridelenú úlohu. Ešte pred spustením úlohy majú možnosť si zvoliť z dostupných jazykov.

3.4 Analýza nedostatkov a návrhy opráv

V tejto časti si zanalyzujeme nedostatky, ktoré vznikli implementovaním prác Lukáša a Dmytra, zároveň sa však pozriem na nedostatky projektu, ktorými sa menovaní priamo nezaoberali, no stojí za to sa týmto problémom venovať.

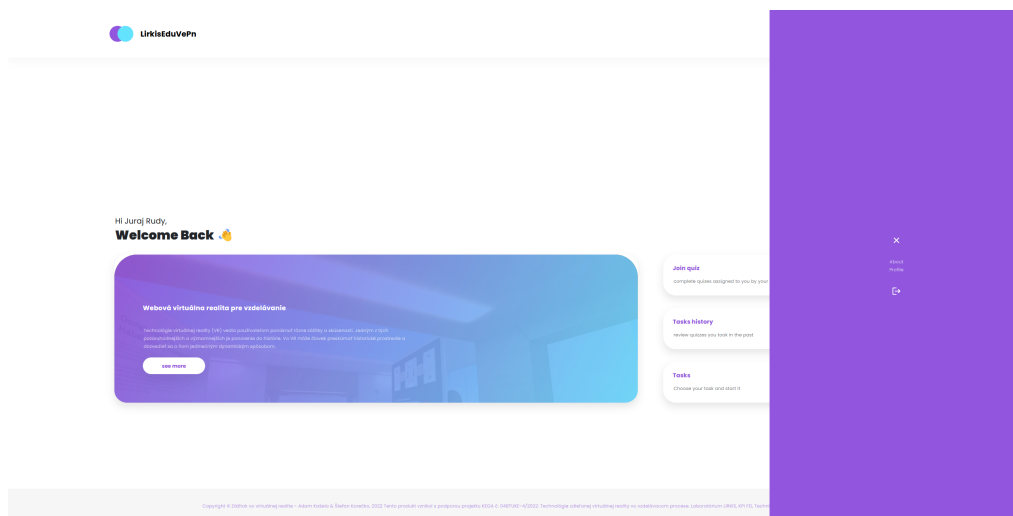
3.4.1 Identifikácia a riešenie implementačných nedostatkov v aktuálneho stavu projektu

V rozšírenej verzii aplikácie, ktorá bola výsledkom práce Lukáša a Dmytra sme našli niekoľko nedostatkov, ktoré by si z nášho pohľadu vyžadovali zmeny prípadne dopracovanie z hľadiska už vypracovanej aplikácie.

- **Malé tlačidlá v navigačnom paneli**

Vo webovej časti práce je množstvo tlačidiel, ktoré sú veľmi malé, a pri tom okolo nich je veľa nevyužitého miesta. Ide primárne o vysúvací navigačný

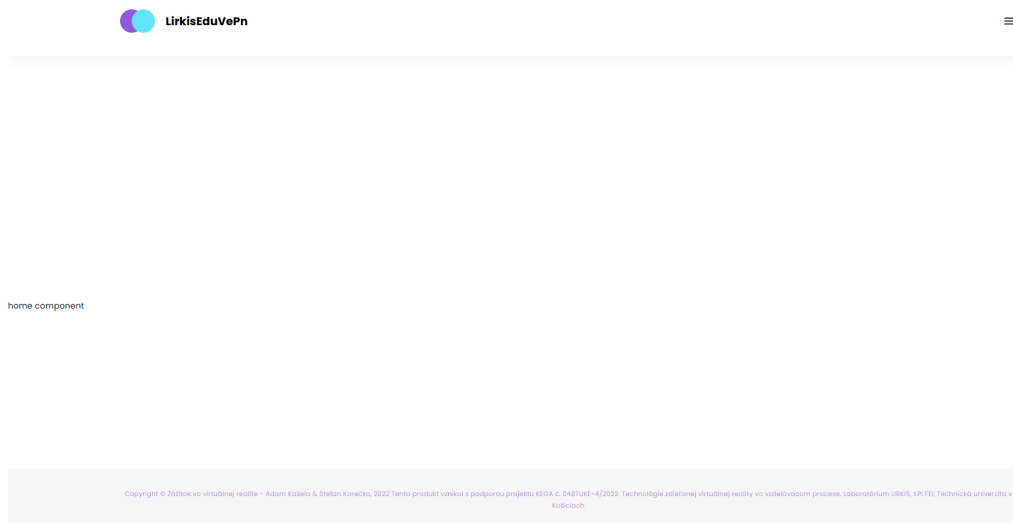
panel, kde veľká časť panela je nevyužitá, no napriek tomu sú tieto tlačidlá príliš malé, je až obtiažne na nich kliknúť.



Obr. 3.1: Malé tlačidlá v navigačnom paneli

- **Nevyužitá strany vo webe**

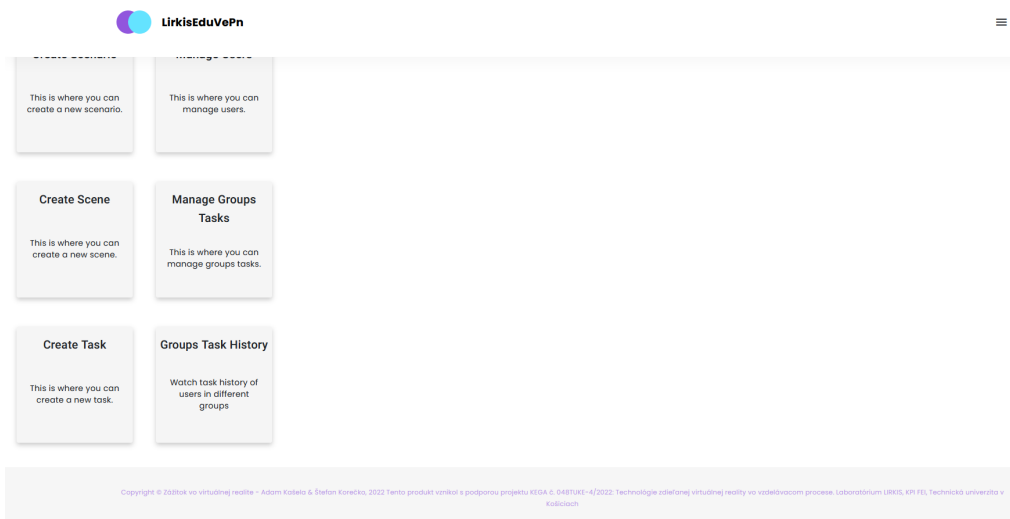
Niektoré webové cesty sú prázdne a nevyužitá. Ide napríklad o cesty */home* alebo */about*.



Obr. 3.2: Stránka */home*

- **Nedostatky v administrátorskom menu**

Administrátorské menu, čo je priestor pre administrátora, alebo učiteľa, kde má možnosť vytvárať vzdelávacie skupiny, manažovať ich, registrovať scenáre a podobne má niekoľko nedostatkov. Jedným z nich je neresponzivita, čo má za následok, že pri zariadeniach s malou zobrazovacou jednotkou sú jednotlivé položky v menu usporiadané tak, že nevyužívajú už všeobecne malý priestor a rovnako aj pri zariadeniach s veľkou zobrazovacou jednotkou, položky nevyužívajú ani polovicu priestoru, tlačidlá sú natlačené na jednej strane a zvyšok priestoru zostáva prázdny. Navyše, ak zariadenie disponuje zobrazovacou jednotkou, ktorá nedosahuje určitú výšku, vrchná lišta prekrýva obsah menu.



Obr. 3.3: Administrátorské menu

Ďalším nedostatkom je nevhodné pomenovanie funkcionalít niektorých tlačidiel, kde napríklad tlačidlo "Create Scene" (v preklade "Vytvoríť Scénu") v skutočnosti neponúka možnosť vytvoriť scénu, ale iba zaregistrovať scénu, ktorá sa už nachádza v súborovom systéme aplikácie. Zároveň pri takomto registrovaní scény je potrebné do formulára vypísať názov adresára, ktorý ako už bolo spomenuté, sa už nachádza v súborovom systéme. Preto by bolo pre používateľa vhodnejšie vytvoriť len akúsi možnosť výberu, ktorý scenár chce zaregistrovať do systému.

Obr. 3.4: Formulár pre registráciu scény

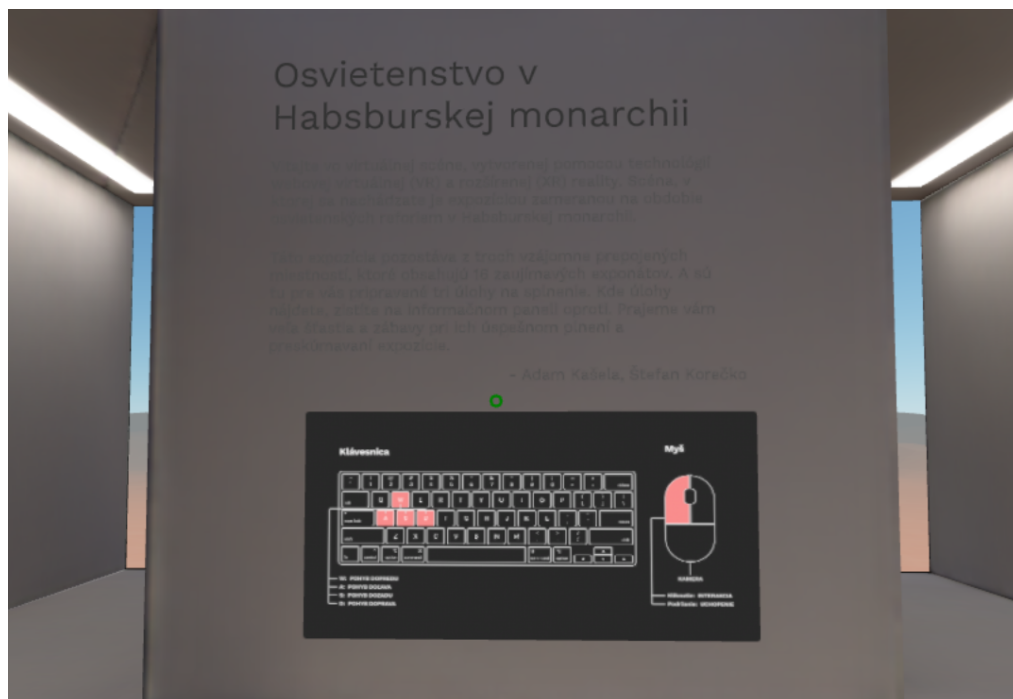
Rovnako aj funkcionálnosť pre nahranie scenára do scény je pomenovaná ako "Create Scenario" (v preklade "Vytvoriť Scenár"), no ide však o možnosť nahrať scenár do systému.

- **Absencia prehľadu a manipulácie so scenármi**

V tomto prípade nám v systéme chýba jednak prehľad scenárov, ktoré sa v systéme nachádzajú, zároveň by bolo vhodné implementovať okrem pridávania scenárov aj možnosť odobratia scenáru zo systému.

- **Ťažko čitateľná informačná stena**

Hneď po spustení scenára sa používateľ objaví v scéne, priamo pred stenou, kde sú prvotné informácie o téme, na ktorú je scenár zameraný. Avšak spomínaná stena a text na nej sú v odtieňoch sivej farby, čo je pre používateľa rozhodne ťažko čitateľné. V tomto prípade by bolo vhodnejšie zvoliť inú farbu textu a nemeniť farbu steny, aby sme zachovali celkový dizajn prostredia.



Obr. 3.5

- **Použitie aktuálnej mapy Slovenska**

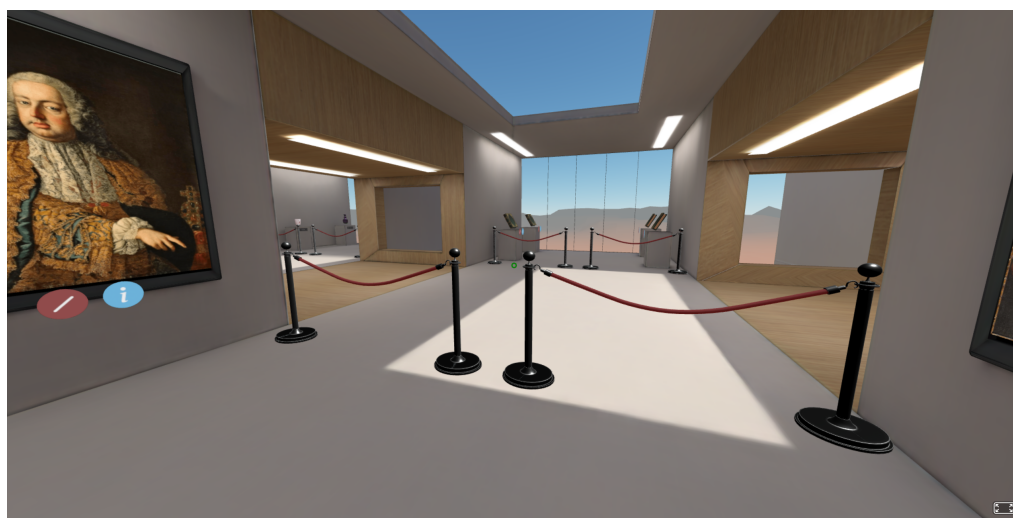
Aktuálna mapa Slovenska, ktorá bola použitá v jednej z úloh, nezodpovedá rozloženiu hraníc v období Habsburskej monarchie, pričom učivo, ktorému je v tejto práci primárne venovaná pozornosť, je práve obdobie Habsburskej monarchie. Riešením by bolo vymeniť mapu za historickú. Po analýze a hľadaní alternatívnej mapy sme prišli na to, že ak by sme použili historickú mapu, veľkú časť z nej by sme museli orezať. Vtedy vznikajú ďalšie problémy, napríklad s kvalitou obrázka mapy, kedy pri veľkom priblížení a orezaní mapy by sa mapa príliš rastrovala. Takto máme zachovanú kvalitu obrázka, preto by sme zmenu neodporúčali. Stav teda nie je ideálny, ale pre danú úlohu vyhovujúci.



Obr. 3.6: Aktuálna mapa slovenského územia pre dejepisnú úlohu

- **Priechodné bariéry**

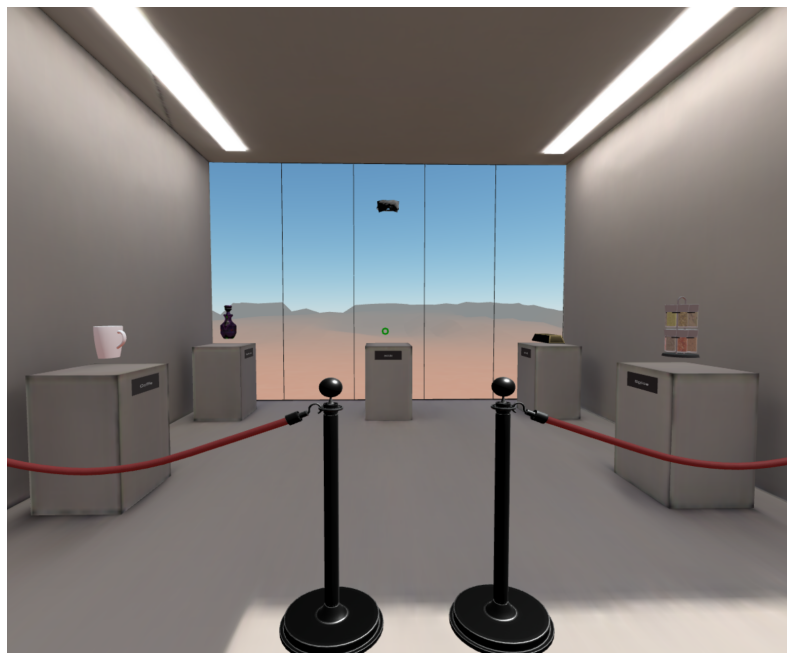
Pridané bariéry sú zaujímavým prvkom, pre jednoduchšie usmernenie študentov, na ktoré úlohy majú zamerať svoju pozornosť. Avšak v tomto prípade sú pridané bariéry priechodné. Neobmedzujú používateľov v pohybe v scéne, čo by malo byť ich účelom.



Obr. 3.7: Pohľad spoza prekročených bariér

- **Lietajúci hodváb**

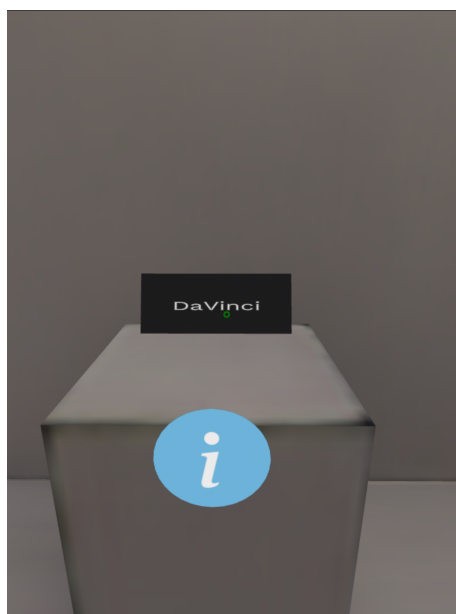
V jednej z úloh je ako interaktívny predmet použitý hodváb. Ten má však svoju počiatočnú pozíciu nastavenú príliš vysoko a tak to pôsobí dojmom, že nie je na podstavci, ale levituje vo vzduchu. Riešením je pravdepodobne len zmeniť počiatočnú pozíciu objektu.



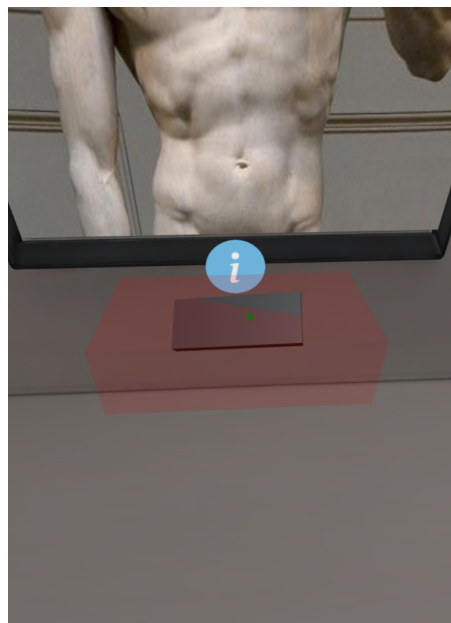
Obr. 3.8: Dislokácia hodvábu

- **Miznúci text na menovkách**

Pri interaktívnej úlohe, kde je potrebné správne priradiť autora diela ku portrétu, nám pri vložení menovky do príslušného boxu mizne text. Pre používateľa to môže byť neprehľadné, ak si chce napríklad skontrolovať, prípadne prehodnotiť svoje rozhodnutia. Riešenie je pre nás v tomto prípade ťažšie odhadnúť, tento prípad si bude vyžadovať hlbšiu analýzu, prečo k tomuto problému dochádza.



(a) Náhľad na menovku s textom



(b) Menovka so zakrytým textom

Obr. 3.9: Problém s menovkami

3.4.2 Analýza ovládania scény pre rôzne druhy zariadení

Používateľom našej aplikácie chceme poskytnúť možnosť ovládať scénu a riešiť úlohy pomocou rôznych druhov elektronických zariadení. V tomto prípade vieme rozoznať 3 základné kategórie:

- Mobilné zariadenia
- Počítače
- VR headsety

Absencia vhodného ovládania pomocou mobilného zariadenia

Jedným z identifikovaných problémov v projekte je absencia vhodného ovládania pre používateľov s mobilnými zariadeniami. Projekt, zameraný na výučbové prostredie vo virtuálnej realite (VR), by mal zohľadniť rôzne spôsoby ovládania, ktoré sú k dispozícii pre mobilné platformy.

Súčasný stav nezabezpečuje optimálne interakcie pre používateľov, ktorí prístupujú k systému prostredníctvom mobilných zariadení. Absencia možnosti využiť senzory ako akcelerometer, alebo iné senzory pre orientáciu zariadenia v

priestore, obmedzuje používateľskú skúsenosť a znižuje efektívnosť výučby vo VR.

Riešením by mohlo byť implementovanie ovládacieho mechanizmu, ktorý by využíval vstupy dostupné na mobilných zariadeniach, čím by vznikla interakcia s výučbovým prostredím. Je potrebné zvážiť možnosti ako dotykové gestá, ovládanie pomocou gyroskopu, alebo inovatívne metódy využitia senzorov pre lepšiu integráciu s mobilnými platformami.

3.4.3 Analýza ovládania pomocou počítača

Pre počítač existuje vhodne implementované ovládanie, kde pohyb je zabezpečený klávesami W, A, S a D a pomocou myši je zabezpečená jednak rotácia pohľadu a zároveň interakcia s objektami v scéne. V tomto prípade nevidíme potrebu túto implementáciu akokoľvek upravovať, či meniť.

3.4.4 Analýza ovládania pomocou VR headsetu

Vzhľadom k tomu, že náš projekt technicky zameriavame na systémy VR, ktoré nás majú vtiahnuť do prostredia, alebo scény a tak zlepšiť celkový pôžitok a v našom prípade aj dodať pridanú hodnotu k vzdelávaciemu procesu, je dôležité mať implementované vhodné ovládanie pre takéto zariadenia.

Aktuálne riešenie nám ponúka pohyb v scéne pomocou thumbstick-ov na ovládačoch. S interakciou sa pri testovaní pomocou zariadenia Oculus(Meta) Quest 2 objavili problémy s manipuláciou objektov. Konkrétne akcie drag-hover-drop nebolo možné s objektami vykonať. Preto bude potrebné navrhnuť opravu tejto implementácie tak, aby manipulácia s objektami bola možná.

Zároveň mnohé aplikácie pre VR ponúkajú možnosť teleportácie, ako alternatívny spôsob pohybu. Tento pohyb je práve vhodný pre veľké scény, kde je potrebné sa presúvať na veľké vzdialenosti. Ideálne by sme chceli zachovať pôvodnú implementáciu pohybu a rozšíriť ju o formu teleportácie.

3.4.5 Identifikácia a riešenie všeobecných problémov v projekte

Počas analýzy systému sme objavili niekoľko problémov, ktoré ovplyvňujú rôzne aspekty fungovania systému, napríklad bezpečnosť, alebo výkonnosť.

- **Nepřítomnosť SSL certifikátu**

Jedným z významných problémov je neexistencia SSL certifikátu pre bezpečné prenášanie údajov. Absencia SSL certifikátu môže mať za následok

nezabezpečenú komunikáciu medzi klientom a serverom, čo predstavuje potenciálne riziko pre utajenie citlivých informácií. Riešením by bolo implementovať SSL certifikát a zaistiť bezpečné pripojenie.

- **Použitie HTTP/1.1**

Nedostatok podpory pre HTTP/2 a HTTP/3 môže mať negatívny vplyv na výkonnosť systému. HTTP/2 prináša výhody ako multiplexovanie, kompresia hlavičiek a prioritné spracovanie, čo môže mať za dôsledok rýchlejšiu komunikáciu medzi klientom a serverom. Odporúča sa prehodnotiť používanie aspoň HTTP/2 s cieľom optimalizovať prenos údajov, čo v našom prípade môže skrátiť čas načítavania scén, kedy sa do webového prehliadaču sťahujú grafické súbory scény. HTTP/3 je stále pomerne nová verzia protokolu a nie všetky webové prehliadače implementujú podporu pre tento protokol. Obzvlášť ide o prehliadače mobilných zariadení, ktoré nepodporujú verziu HTTP/3.

- **Absencia HTTPS - využívanie portu 80**

S implementáciou SSL certifikátu je dôležité zvážiť presun na bezpečný prenos údajov cez HTTPS protokol, ktorý bez SSL certifikátu zväčša webové prehliadače nechcú podporovať rovnako ako protokol HTTP/2 a novší. V dnešnej dobe sa to považuje za štandard a jeho implementácia môže prispieť k bezpečnosti komunikácie medzi klientom a serverom. Odporúča sa implementovať HTTPS pre celkové zabezpečenie systému.

- **Jednoduché heslá**

V našom systéme sme identifikovali z hľadiska hesiel 2 problémy. Prvým problémom je, že v systéme povoľujeme používateľom vytvoriť si heslo, bez akýchkoľvek požiadaviek na jeho dĺžku, používanie kombinácie čísel a písmen a podobne. Druhým problémom je, že v samotnom systéme používame či už jednoduché, alebo aj vopred nastavené heslá, čo pre potenciálneho útočníka vytvára priestor preniknúť do systému.

4 Nasadenie existujúceho riešenia na TUKE cloud server

Táto kapitola sa venuje systému nasadenia aktuálneho riešenia, na ktorom pracovali študenti Dmytro a Lukáš, ale aj budúcich verzií, na cloudový server poskytovaný Ústavom výpočtovej techniky, Technickej univerzity v Košiciach (UVT TUKE).

4.1 Definovanie úlohy

Našou úlohou bolo nasadiť aplikáciu v aktuálnom stave na cloudový server. V tomto prípade ide už o verziu aplikácie, na ktorej pracovali Lukáš a Dmytro vo svojich bakalárskych prácach, ale aj následne ďalšie verzie, ktoré budu nasadené do prevádzky obdobným spôsobom. Celé riešenie je distribuované do nasledujúcich štyroch docker kontajnerov.

- **lirkis-ui**
Služba, ktorá zabezpečuje webové rozhranie, s ktorým interagujú používatelia.
- **lirkis-service**
Jedná sa o backendovú službu, ktorá spracováva komunikáciu medzi webových rozhraním a databázou.
- **lirkis-pgadmin**
Ide o webový administračný nástroj pre databázu. Pomocou tohto nástroja vie administrátor systému nahliadať do databázy, ale aj manipulovať s dátami.
- **lirkis-database**
Tu už beží samotná služba s databázou a jej obsahom.

4.2 Dostupné možnosti na TUKE cloude

Ako študenti máme možnosť využiť cloudové služby poskytované UVT TUKE, kde máme na výber z množstva linuxových a windowsových distribúcií. V tomto prípade nebudeme analyzovať každú možnú distribúciu, ale zvážime v prvom rade, či je pre nás vhodnejšia distribúcia linuxu, alebo windowsu.

- **Linux**

Pre samotný docker je linux natívny systém, čo znamená, že kontajnery sú spustiteľné bez akýchkoľvek prídavných doplnkov, alebo komponentov. Aj vďaka tomu môžu mať kontajnery lepší výkon na linuxovom systéme. Jednou z možných nevýhod je, ak sa niektorý z kontajnerov spolieha na závislosti, ktoré sú špecifické pre windows systém. To môže skomplikovať celý proces nasadenia a vyžadovať ďalšiu konfiguráciu.

- **Windows**

V systémoch windows je ku spusteniu kontajnerov potrebné podsystém windowsu pre linux(WSL). Aj keď táto možnosť funguje dobre, pridáva to ďalšiu vrstvu, čo môže zapríčiniť nižší výkon. V mnohých prípadoch vo windows systémoch nie je virtualizácia štandardne povolená, čo prináša ďalšie komplikácie pre celý proces.

Po zvážení jednotlivých možností sme sa teda rozhodli pre operačný systém Linux. Keďže máme dostupné rôzne distribúcie, v tomto prípade sme sa rozhodli využiť distribúciu Ubuntu, ktorá patrí k najpoužívanejším distribúciám a zároveň sme už s týmto systémom oboznámení z našich predošlých skúseností.

4.3 Proces nasadenia na server

Efektívne a spoľahlivé nasadenie softvérových projektov je kritickým krokom v ich životnom cykle. V našom prípade sme sa rozhodli využiť Dockerhub, ako centralizované úložisko kontajnerov, a nasadiť náš projekt na cloudový server. Táto sekcia podrobne opisuje proces, od nahrania kontajnerov do Dockerhubu po ich úspešné nasadenie a spustenie na cieľovom serveri.

1. Dockerhub

Pri nasadzovaní nášho projektu sme využili Dockerhub ako centrálnu úložisko pre naše kontajnery. Vytvorili sme dva kontajnery, ktoré sme pomenovali lirkis-UI a lirkis-service a nahrali sme ich do Dockerhubu, čím sme zabezpečili jednoduchý a centralizovaný prístup k našim obrazom.

2. Inštalácia kontajnerov

Po úspešnom nahraní našich kontajnerov na Dockerhub sme prešli k nasadeniu našej aplikácie na cloudový server. Postupovali sme tak, že sme na našom cieľovom Linux serveri použili príkaz

```
docker pull
```

na stiahnutie obrazov lirkis-UI a lirkis-service z Dockerhubu. Následne sme využili súbor docker-compose.yml, kde sme definovali konfiguráciu nášho projektu. Súbor docker-compose.yml nám zabezpečí vytvorenie ostatných dvoch kontajnerov.

3. Záverečné spustenie

Po úspešnom stiahnutí obrazov sme vytvorili a spustili všetky kontajnery pomocou príkazu

```
docker-compose up.
```

Tým sme dosiahli kompletne nasadenie nášho projektu na cloudový server. Naša aplikácia bola úspešne spustená, pripravená na poskytovanie služieb na vybranom serveri.

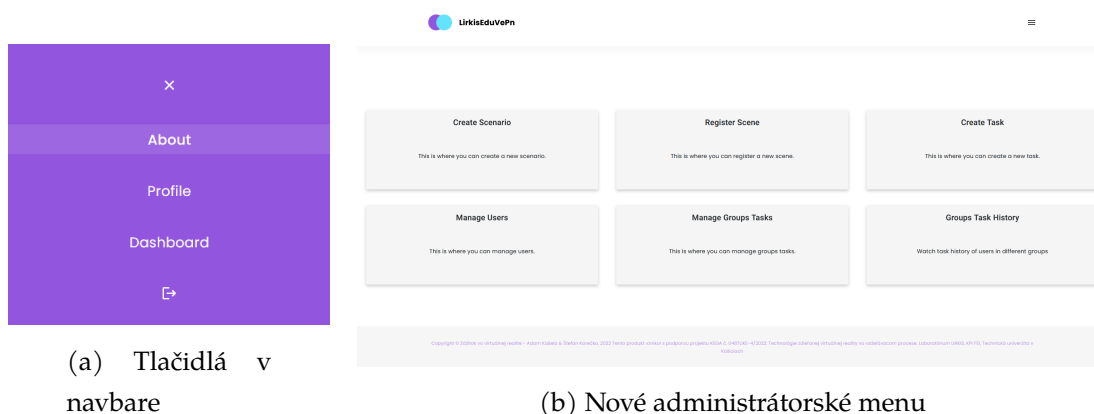
Týmto spôsobom sme dosiahli efektívne a jednoduché nasadenie nášho projektu do produkčného prostredia, pričom sme využili výhody Dockerhubu a jeho možnosti centralizovaného spravovania kontajnerov.

5 Opravy implementačných nedostatkov v aktuálnej verzii aplikácie

Počas analýzy sme objavili niekoľko menších, ale aj väčších nedostatkov v aplikáciách, ktoré je potrebné opraviť. V tejto časti sa budeme venovať opravám jednotlivých nedostatkov.

5.1 Úpravy v používateľskom rozhraní webu

Ako prvé sme v používateľskom rozhraní upravili tlačidlá, ktoré boli veľmi malé. Zväčšili sme text a zároveň celkovú veľkosť priestoru, na ktorý je možné kliknúť, prípadne sme využili prázdne priestory. Za druhé sme zrušili stránku pre *home*. Všetky funkcionality, alebo tlačidlá, ktoré sa odkazovali na túto stránku sme dali presmerovať do cesty *userprofile*.

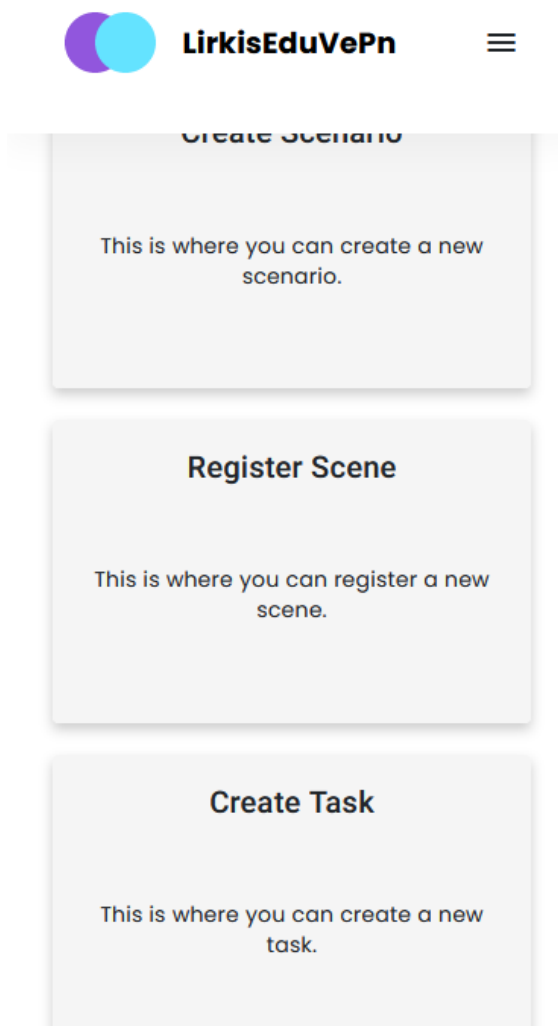


Obr. 5.1: Upravené používateľské rozhranie

5.2 Doplnenie responzivity častí webu

Pre niektoré webové stránky neboli implementované responzívne nastavenia, čo znamená, že stránky sa nezobrazovali optimálne ku veľkosti zobrazovacej jed-

notky. V mnoho prípadoch sa jedná o zobrazovacie jednotky s malým rozlíšením, teda napríklad tablety, alebo mobilné zariadenia. Na obrázku 5.2 je ukážka responzivity upraveného administrátorského menu.



Obr. 5.2: Informačná stena s upravenou farbou textu

5.3 Vytvorenie prehľadu scenárov a pridanie možnosti odstrániť scenár

Pre scenáre sme vytvorili, podobne ako napríklad pre používateľov, stránku s prehľadom scenárov, ktoré sú v systéme. Pridali sme aj možnosť odstránenia scenára. V prípade odstránenia scenára sme implementovali aj odstránenie úloh, v ktorých sa scenár používa. Túto funkcionality bolo potrebné implementovať ako na fronte, tak aj na backende.

Časť kódu, ktorá volá backendovú službu na odstránenie scenára, vyzerá nasledovne:

```
public deleteScenarioById(id: number): Observable<any> {
    return this._client.delete(this.API_URL + "/scenario/" + id, {
        responseType: "json",
        headers: {Authorization: "Bearer " +
                    localStorage.getItem("jwt-token")}
    })
}
```

Tento zdrojový kód volá backendovú službu v súbore

ScenarioController.java,

kde sa vykoná nasledovná funkcia:

```
@DeleteMapping("/{id}")
@PreAuthorize("hasAuthority('ROLE_ADMIN') or
              hasAuthority('ROLE_TEACHER')")
public ResponseEntity<Void> delete(@PathVariable Long id) {
    //deleteScenario returns IDs of all tasks that are attached
    //to this scenario
    List<Long>taskIdList = scenarioService.deleteScenario(id);
    //loop to remove every task that was attached to the scenario
    for (Long taskId : taskIdList) {
        taskService.removeTask(taskId);
    }
    return new ResponseEntity<>(HttpStatus.OK);
}
```


Volaná funkcia

`deleteScenario(id)`,

ktorá je implementovaná v zdrojovom kóde

`ScenarioService.java`,

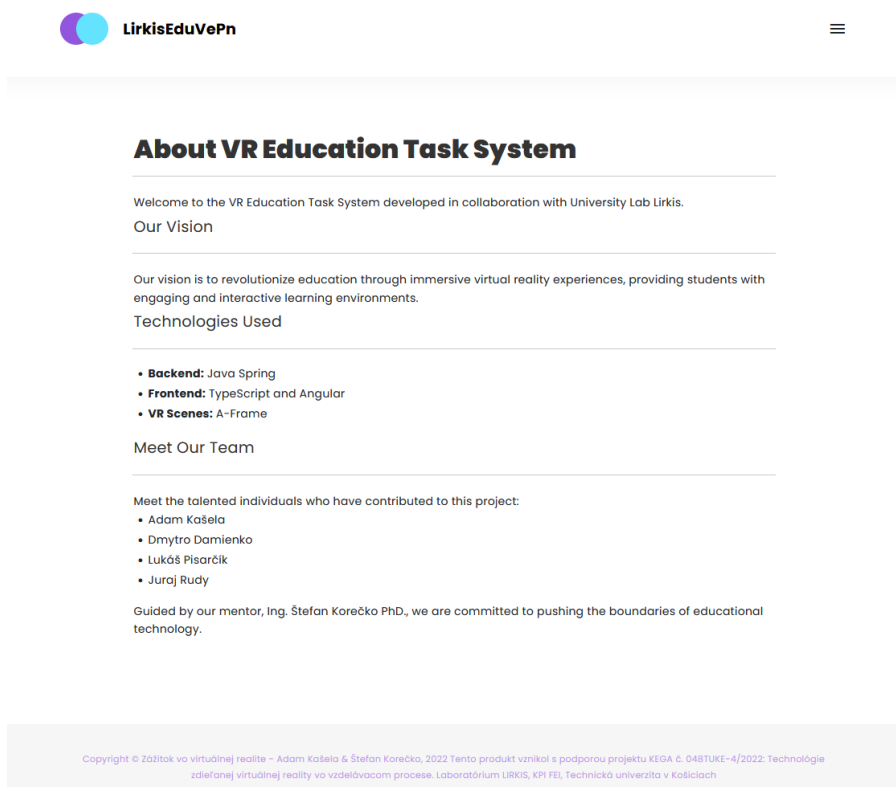
má nasledovnú syntax:

```
public List<Long> deleteScenario(Long id) {
    Scenario scenario = scenarioRepository
        .findById(id)
        .orElseThrow();
    List<Long> tasks = new ArrayList<Long>();
    for (Task task : scenario.getTasks()) {
        tasks.add(task.getId());
    }
    scenarioRepository.deleteById(id);
    return tasks;
}
```

Ostatné funkcie, ktoré sme pri tejto implementácii použili, boli už v systéme vytvorené a preto dáva zmysel ich použiť a nevytvárať nové.

5.4 Doplnenie obsahu do stránky *about*

Do stránky *about* sme doplnili základné informácie o zameraní projektu, kto sú prispievatelia projektu, aké technológie v projekte používame a pod. Stránka vyzera nasledovne:



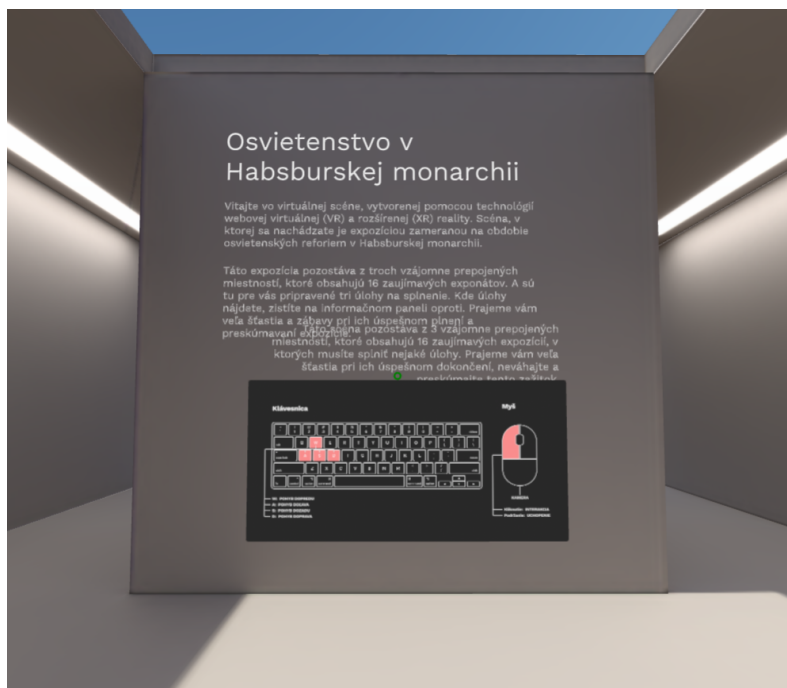
Obr. 5.3: Náhľad stránky about

5.5 Úpravy v scéne

V analytickej časti bolo identifikovaných niekoľko nedostatkov v scéne s tématickou Habsburskej monarchie.

- **Úprava informačnej steny**

Na informačnej stene bola použitá farba textu, ktorá splývala s farbou samotnej steny. Farbu textu sme preto upravili tak, aby bol text čitateľný. Zároveň so zmenou farby textu sme objavili nový problém, a to že text nie je zarovnaný a navzájom sa jednotlivé odseky textu prekrývajú.



Obr. 5.4: Informačná stena s upravenou farbou textu

Preto sme dodatočne zarovnali text na ľavú stranu a odsekom nastavili medzery tak, aby sa navzájom neprekrývali.



Obr. 5.5: Informačná stena so zarovnaným textom

- **Opravené počítačové súradnice hodvábu**

Hodvábu sme upravili súradnicu na osi ypsilon, ktorá vo virtuálnom priestore predstavuje polohu na vertikálnej osi.

- **Oprava miznúceho textu na menovkách**

Pre miznúci text na menovkách, v momente kedy sa menovka presunie na miesto priradenia k určitému obrazu pri jednej z úloh, sme implementovali riešenie, kde sme plný box vymenili iba za konštrukciu. To sa odborné nazýva "wireframe" model. Po implementácii už je text na menovke vidieť jasne.



Obr. 5.6: Opravená viditeľnosť menovky na mieste priradenia

5.6 Ovládanie scény

V tejto sekcii sa zameriame na implementáciu nových prvkov ovládania pohybu a interakcií v scéne. Ako bolo spomenuté pri analýze, pre počítače nevidíme potrebu implementovať akékoľvek zmeny, a preto je táto časť zameraná na mobilné zariadenia a VR headsety.

5.6.1 Nové ovládanie pre mobilné zariadenia

Pre ovládanie pre tento typ zariadení sme hľadali, akým spôsobom by bolo vhodné implementovať pohyb a manipuláciu s objektami v scéne. Aby ovládanie nebolo len o tlačítkach na obrazovke, rozhodli sme sa pre možno netradičnú implementáciu pohybu. Pohyb vpred a vzad je závislý od naklonenia mobilného zariadenia vpred, alebo vzad. Rotácia v scéne je zabezpečená, podobne ako ty býva VR headsetoch, rotáciou mobilu okolo zvislej osi. Týmto sa snažíme používateľovi priblížiť ovládanie v scéne, ovládaniu pomocou VR headsetu.

Pre implementáciu sme vytvorili vlastný komponent

```
mobileViewHandler.component.js.
```

Ten je následne implementovaný v skripte pre danú scénu priamo v entite avatara. Tento komponent cyklicky sleduje naklonenie zariadenia dopredu a dozadu. Následne podľa orientácie v priestore scény tým smerom, ktorým je otočený. Jednotlivý posun v osiach X a Z je dopočítaný pomocou trigonometrických funkcií sínus a kosínus. Následne potrebujeme používateľovi obmedziť tento pohyb v scéne. Pre obmedzenie pohybu využívame “navmesh”, čo predstavuje vyznačenú plochu, po ktorej je povolený pohyb. Preto máme vytvorenú aj funkciu, ktorá pomocou objektu “raycaster” sleduje, či nová pozícia v scéne sa nachádza vo vymedzenom priestore. Kód vyzerá nasledovne:

```
AFRAME.registerComponent('phone-tilt-move', {  
  init: function () {  
  
    this.avatarEntity = document.getElementById('player');  
    this.navmeshEntity = document.getElementById('navmesh-entity');  
    this.navMesh;  
    this.raycaster = new THREE.Raycaster();  
    // Array to store collidable objects  
    this.collidableObjects = [];
```

```
this.navmeshEntity
  .addEventListener('model-loaded', this.setup.bind(this));

// Register device orientation event listener
window.addEventListener('deviceorientation',
  this.handleOrientation.bind(this), true);
},

setup: function () {
  const group = this.navmeshEntity.getObject3D('mesh');
  const mesh = group.children[0];

  this.navMesh = new THREE.Mesh(mesh.geometry, mesh.material);
  this.navMesh.position.copy(new THREE.Vector3(0, 1, 16));
},

handleOrientation: function (event) {
  // Check if the device supports the DeviceOrientationEvent
  if (event.alpha || event.beta || event.gamma) {
    const speed = 0.001;
    const tiltForward = event.beta; // Tilt forward/backward
    const entity = this.avatarEntity.object3D;

    const theta = event.alpha * (Math.PI / 180);
    const moveX = -Math.sin(theta) * tiltForward * speed;
    const moveZ = -Math.cos(theta) * tiltForward * speed;

    // Calculate new position based on tilt
    if(event.beta < 70 ){
      const newPosition = {
        x: entity.position.x + moveX,
        y: entity.position.y,
        z: entity.position.z + moveZ
      };

      const clampedPosition = this.clampToNavmesh(newPosition);
```

```

    // Update entity position
    entity.position.copy(clampedPosition);
  }
}
},
// Perform raycasting to detect collisions with navmesh triangles
clampToNavmesh: function (position) {
  // Cast raycaster downwards
  this.raycaster.set(position, new THREE.Vector3(0, -1, 0));
  this.raycaster.recursive = true;
  const intersects = this.raycaster.intersectObject(this.navMesh);

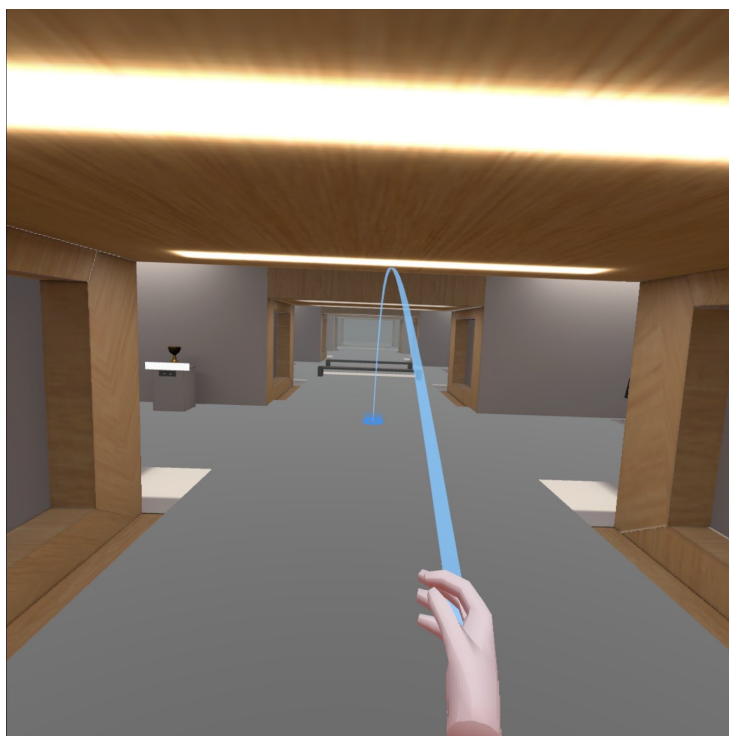
  if (intersects.length > 0) {
    // If collision detected, clamp position to the intersection point
    console.log("intersecting");
    const pos = {
      x: intersects[0].point.x,
      y: position.y,
      z: intersects[0].point.z
    }
    return pos;
  } else {
    // If no collision detected, return original position
    console.log("not intersecting");
    return position;
  }
}
});

```

5.6.2 Úprava ovládania pre VR headsety

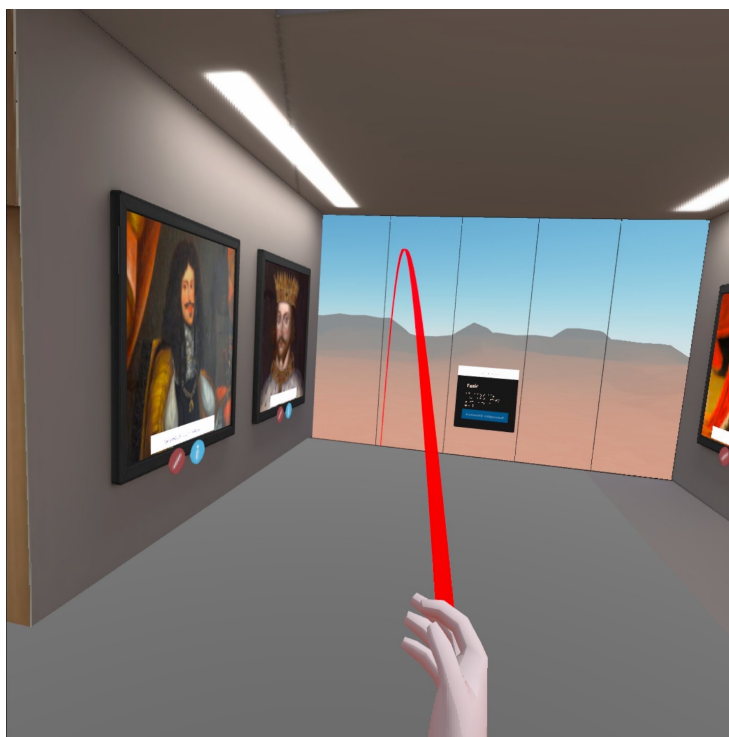
Pre VR headsety sme implementovali 2 základné zmeny. Prvá z nich nám umožňuje efektívnejší presun na veľké vzdialenosti v scéne. V tomto prípade sme využili už existujúci komponent “blink-controls” od Jure Triglav[20], ktorý nám umožňuje jednoducho implementovať krátky teleportáciu. Pre tento komponent sme implementovali príslušné atribúty a celé to implementovali pre entitu praveho ovládača. Nižšie na obrázku 5.7 môžeme vidieť ukážku toho, ako vyzerá

blink, respektíve teleportácia v scéne použitím "blink-controls".



Obr. 5.7: Povolený blink

Na obrázku 5.8 zasa vidíme ukážku toho, ako je blink blokovaný v prípade, že sa používateľ snaží presunúť mimo scénu.

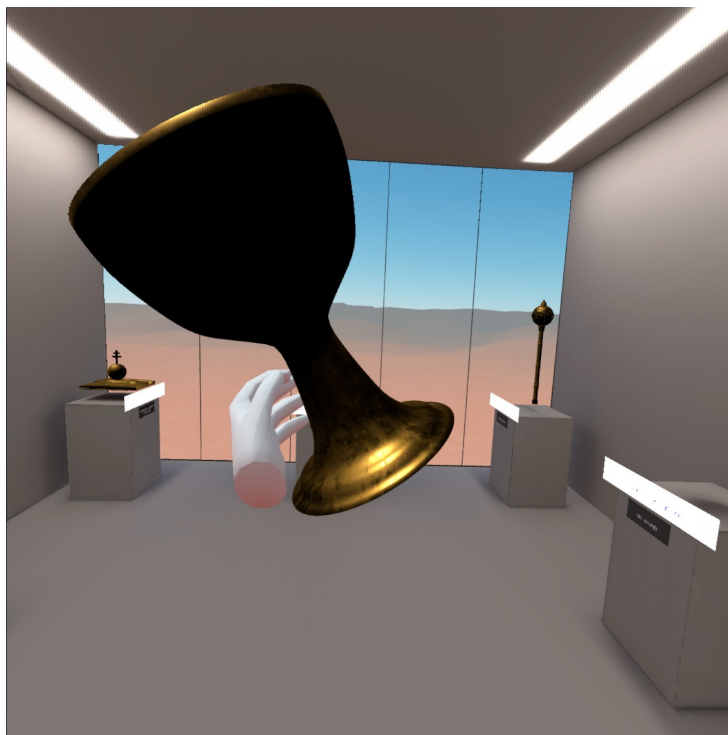


Obr. 5.8: Nepovolený blink

Ďalším s riešených problémov pri používaní VR headsetu je to, že nefunguje manipulácia s objektami, ktoré treba niekam presúvať. Na to máme implementovaný komponent super-hands[21], ktorý nefunguje správne kvôli nie vhodnej implementácii nášho virtuálneho avatara v skripte scény. Upravená časť skriptu, ktorá definuje avatara, je implementovaná nasledovne.

```
<a-entity
  physics-collider
  id="player"
  static-body="shape: sphere; sphereRadius: 0.02"
  position="8.275 1 2.93"
  movement-controls="constrainToNavMesh: true;"
  sound="src: #win; on: win; poolSize: 1;">
  <a-camera id="playerhead"></a-camera>
  <a-entity id="Rhand" hand-controls="hand: right"
    blink-controls="cameraRig: #player;
      teleportOrigin: #playerhead;
      collisionEntities: #navmesh-entity;
      button: trigger;
      rotateOnTeleport: false;">
  </a-entity>
  <a-entity id="Lhand" hand-controls="hand: left" mixin="physics-hands">
  </a-entity>
</a-entity>
```

Výsledok tejto implementácie môžeme vidieť na obrázku 5.9.



Obr. 5.9: Manipulácia s objektami v scéne pomocou VR headsetu

5.7 SSL certifikát s vlastným podpisovaním

Ako bolo v analýze spomenuté, okrem samotnej podstaty SSL certifikátu, je tak tiež potrebný na to, aby sme mohli implementovať zabezpečený protokol HTTPS a takisto pre vyššie verzie HTTP protokolu. Na to, aby sme mohli vlastniť SSL certifikát sa štandardne využívajú certifikačné authority, ktoré tieto certifikáty udeľujú. V našom prípade, kedy je so serverom možná komunikácia iba v rámci VPN siete TUKE sme sa preto rozhodli, využiť možnosť si takýto certifikát sami pridať, aj ho overovať. Vďaka tomu môžeme pristúpiť k implementácii vyššie spomínaných protokolov. Certifikát sme vytvorili nasledujúcim spôsobom v linuxovom termináli:

```
$ openssl genrsa -des3 -out myhostname.key 4096
Enter pass phrase for myhostname.key:
$ openssl req -new -key myhostname.key -out myhostname.csr
Enter pass phrase for myhostname.key:
$ cp myhostname.key myhostname.key.pwd
$ openssl rsa -in myhostname.key.pwd -out myhostname.key
Enter pass phrase for myhostname.key.pwd:
$ openssl x509 -req -in myhostname.csr -signkey myhostname.key
-out myhostname.crt
```

Následné implementovanie vytvoreného certifikátu bolo nastavené v súbore “nginx.conf”, ktorý bude uvedený po nasledujúcich sekciách, keďže nasledujúce sekcie s touto úzko súvisia.

5.8 Implementácia pre novšie protokoly HTTP

Ak sa spomínajú vyššie verzie HTTP protokolu, ide o verzie HTTP/2 a HTTP/3. Vo svojej podstate ide o nastavenia pre server. V našom prípade rovnako, ako v predošlej sekcií, sú tieto nastavenia implementované v súbore “nginx.conf”.

5.9 Implementácia šifrovaného HTTPS protokolu

Pôvodne bola nastavená komunikácia klienta so serverom na porte 80, čo je port pre nezabezpečený protokol HTTP. V nastaveniach konfiguračného súboru pre nginx, sme preto zmenili port, na ktorom server počúva, z hodnoty 80 na hodnotu 443. Je to port využívajúci protokol HTTPS, čo sme chceli docieľať. Zároveň sme upravili v konfiguračnom súbore pre docker, ktorý nám vytvára kontajnery, aby v kontajnery pre front-end presmerovával komunikáciu z portu 4200 na port 443, z pôvodného presmerovania na 80. To znamená, že komunikácia prichádzajúca smerom ku danému kontajneru na port 4200, bude presmerovaná na port 443.

Časť konfiguračného súboru “nginx.conf”, kde sme implementovali ssl certifikát, vyššie protokoly http a zabezpečený protokol https vyzerá nasledovne:

Zdrojový kód 5.1: Konfiguračný súbor Nginx.conf

```
server {
    listen 443 ssl;
    listen [::]:443 ssl;
    http2 on;

    server_name _;
    root /usr/share/nginx/html;
    index index.html;

    ssl_certificate /etc/nginx/ssl/myhostname.crt;
    ssl_certificate_key /etc/nginx/ssl/myhostname.key;

    ssl_session_timeout 5m;
    ssl_session_cache shared:SSL:10m;
```

```
ssl_protocols TLSv1.1 TLSv1.2;
# ssl_ciphers "HIGH:!aNULL:!MD5 or HIGH:!aNULL:!MD5:!3DES";
ssl_ciphers 'TLS_AES_128_GCM_SHA256:
            TLS_AES_256_GCM_SHA384:
            ECDHE-RSA-AES128-GCM-SHA256:
            ECDHE-RSA-AES256-GCM-SHA384';
ssl_prefer_server_ciphers off;

# Enable OCSP stapling
ssl_stapling on;
ssl_stapling_verify on;

# HSTS (ngx_http_headers_module is required)
# (15768000 seconds = 6 months)
add_header Strict-Transport-Security "max-age=15768000" always;

location / {
    try_files $uri $uri/ /index.html;
}
}
```

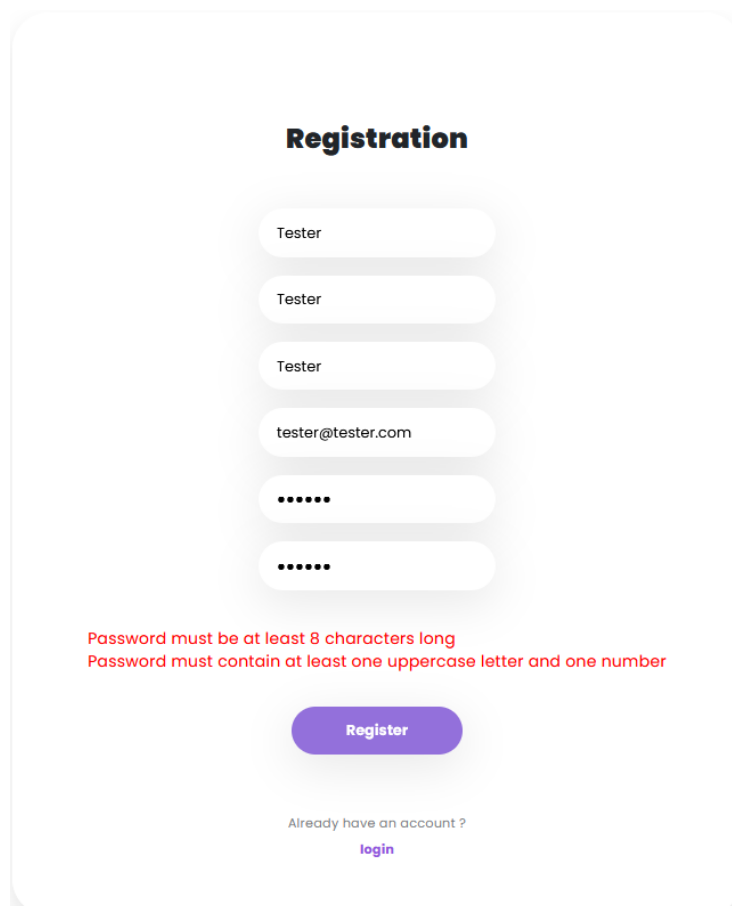
5.10 Zavedenie pravidiel pre vytváranie hesiel

Do systému sme zaviedli niekoľko základných pravidiel pre vytváranie hesiel pri registrácií, no keďže v našom systéme nezhrmažďujeme kritické dáta používateľov, ako napríklad bankové údaje, tieto pravidlá sme prispôbili týmto podmienkam[22].

Jedná sa o tieto pravidlá:

- **Použitie veľkého aj malého písmena**
- **Použitie číslíc**
- **Minimálna dĺžka 8 znakov**

V prípade nedodržania týchto pravidiel je proces registrácie blokován. Zároveň sa v registračnom formulári ukážu textové správy s nedodržanými pravidlami. Ukážku môžeme vidieť na obrázku 5.10.



The image shows a registration form titled "Registration". It contains several input fields: a name field with "Tester", a username field with "Tester", an email field with "tester@tester.com", and two password fields, both containing six dots. Below the password fields, there are two red error messages: "Password must be at least 8 characters long" and "Password must contain at least one uppercase letter and one number". At the bottom of the form, there is a purple "Register" button, a link "Already have an account?" with a "login" link below it.

Obr. 5.10: Ukážka nespĺnenia pravidiel pre heslo

6 Vyhodnotenie

V rámci tejto práce sme sa zamerali jednak na vylepšenia niektorých komponentov programu, ale aj na vypracovanie výskumu, ktorý bol zameraný na Petriho siete. Prieskumom sme sa snažili zistiť, ako rozumejú Petriho sieťam ľudia, ktorí nemajú technické vzdelanie, alebo skúsenosti, v porovnaní s ľuďmi z technickej oblasti.

6.1 Vyhodnotenie prieskumu

Pri hľadaní respondentov sme sa zamerali na 2 skupiny ľudí. Jedna skupina, ktorú sme nazvali technická skupina, sa skladala z respondentov, ktorí majú buď vzdelanie, alebo prax v oblasti IKT. V tejto skupine sme ale hľadali výlučne ľudí, ktorý sa s Petriho sieťami ešte nikdy nestretli. Druhá skupina, nazvaná ako netechnická, bola zložená z ľudí, ktorí nemajú skúsenosti, alebo vzdelanie v oblasti IKT.

V oboch skupinách dostali respondenti vypracovať úlohu s Petriho sieťou, ktorú sme sa snažili priblížiť situácii z reálneho sveta. Predtým si všetci respondenti mali príležitosť preštudovať základné pojmy a pravidlá, ako sa takáto Petriho sieť konštruje. V tabuľke 6.1 môžeme vidieť výsledky oboch skupín.

	Technická sk.	Netechnická sk.
Úspešnosť	70%	50%
Počet chýb A	2	24
Počet chýb B	0	2
Počet chýb C	2	5
Priemerný čas prípravy	1:42	1:57
Priemerný čas riešenia	3:34	5:37
Priemerný čas celkom	5:16	7:34

Tabuľka 6.1: Porovnanie výsledkov oboch skupín prieskumu

6.1.1 Vlastné zhodnotenie

Z tabuľky s výsledkami môžeme vidieť niekoľko vyplývajúcich faktov. Technická skupina má nielen vyššiu úspešnosť, no zároveň v priemere respondentom z tejto skupiny trvalo riešenie aj študovanie teórie kratšiu dobu. Najčastejšou chybou, ktorej sa respondenti z netechnickej skupiny dopúšťali, boli chýbajúce a nadbytočné hrany.

Prieskum nám ukázal základný pohľad na to, ako respondenti pochopili problematiku a zadanú úlohu. Častou spätnou väzbou od respondentov, ktorý sa dopustili chýb bolo, že porozumeli svojim chybám a boli presvedčený, že inú úlohu by už zvládli. Bolo by zaujímavé tento prieskum rozšíriť tak, aby respondenti museli vyriešiť viacero úlohu, prípadne aj s stupňujúcou sa obťažnosťou.

6.2 Vyhodnotenie vypracovaných vylepšení

Pri implementácií vylepšení a nových komponentoch sme sa zamerali na rôzne časti systému. V nasledujúcich bodoch si zhrnieme jednotlivé vylepšenia implementované v systéme.

- Doplnenie implementácie vo webovom rozhraní

Vo webovom rozhraní sme implementovali rôzne vylepšenia. Menšími vylepšeniami je napríklad doplnenie responzivity v častiach webu, kde takáto implementácia chýbala. Prázdne stránky vo webe sme buď odstránili, alebo sme v nich implementovali určitý obsah ako napríklad v stránke “/about”. V administrátorskom paneli sme upravili rozloženie samotnej stránky, doplnili popisujúce texty ku tlačidlám, opravili názvoslovie tak, aby odpovedalo tomu, čo dané tlačidlo vykonáva a pridali sme aj panel pre prehľad a úpravu scenárov.

- Oprava nedostatkov v scéne

V scéne s dejepisnými úlohami z obdobia Habsburskej monarchie sme opravili niekoľko identifikovaných nedostatkov a vylepšení. Opravili sme problém s miznúcimi textami na menovkách, opravili sme informačnú stenu, ktorá nebola čitateľná a opravili sme počiatočnú pozíciu objektu hodvábu.

- Implementácia ovládania scén rôznymi zariadeniami

Implementovali sme vlastný komponent pre ovládanie pohybu a interakcií v scéne pomocou mobilných zariadení. Pre VR headsety sme pridali mož-

nosť pohybu cez akciu blink a opravili problém s objektami, s ktorými sa nedalo hýbať.

- Implementácia funkcionalít v backende

Na backendovej časti projektu sme implementovali logiku pre prácu so scenármi. Išlo hlavne o odstránenie scenára so systémom.

6.3 Možnosti ďalších vylepšení

Projekt má výrazný potenciál posunúť vzdelávanie vpred a priniesť tak do procesu vzdelávania inovatívny prístup. Je tu však množstvo príležitostí, ako celú aplikáciu rozšíriť:

- Rozšíriť prieskum
Vyplývajúc z vyhodnotenia prieskumu, bolo by zaujímave rozšíriť prieskum o viacero úloh, prípadne zvyšovať obtiažnosť úloh. Prvá úloha by mohla byť dokonca jednoduchšia než úloha, ktorú sme vymysleli my. Mohli by sme sledovať, ako by si dokázali respondenti poradiť so zvyšujúcou sa náročnosťou, prípadne by sme objavili určitý limit, kde by to už bolo príliš náročné.
- Vytvoriť nové scény a scenáre
Vytvorenie nových scén a scenárov aj pre ďalšie odvetvia vzdelávania, či už pre prírodné vedy, humanitné vedy, alebo akýkoľvek predmet vyučovaný na školách. Zaujímavé by mohlo byť vytvoriť scény a scenáre, ktoré by boli zamerané pre odborné predmety, ktoré sa učia študenti na odborných školách.
- Pridať možnosť spolupráce
Upravenie systému tak, aby sa v jednej scéne mohli vidieť a spolupracovať viacerí študenti zároveň. Prípadne aby sa ku skupine vedel pripojiť aj učiteľ, ktorý by vedel so študentami komunikovať.
- Pridať avatara
Toto vylepšenie by bolo potrebné predovšetkým, ak by sa realizoval predošlý návrh. Študenti by sa tak navzájom v scéne videli, prípadne učiteľ by mohol používať odlišného avatara.
- Refactoring zdrojového kódu
V zdrojových kódoch sa vyskytujú riadky, ktoré sa často krát opakujú, alebo je v nich malý rozdiel, napríklad v určitom parametri. Tieto časti kódu by sa tak mohli zovšeobecniť do funkcií, ktoré by sa už implementovali spolu s príslušnými parametrami. Bolo by zaujímavé implementovať skripty, ktoré by vedeli do scény importovať určité komponenty, ako napríklad komponent avatara. Tým by sa zjednodušila implementácia nových scén.
- Lobby priestor
Zaujímavé by bolo implementovať jednu scénu, ktorá by slúžila ako "čakáreň". Bola by to scéna, do ktorej by sa študentom spustila ako prvá a z tejto

scény by si vedeli ďalej vyberať, ktorú scénu chcú spustiť. Napríklad už vypracovanú scénu o Habsburskej monarchii.

7 Záver

V rámci tejto práce sme sa zamerali na rozšírenie práce, na ktorej pracovali Bc. Lukáš Pisarčík a Bc. Dmytro Damienko. Ich práce nadväzovali na prácu Ing. Adama Kašelu.

Ak ich práce zhrnieme dokopy, tak sme mali fungujúci systém pre virtuálne prostredie s exponátmi a úlohami. Systém fungoval na princípe docker kontajnerov, kde sú rozdelené jednotlivé časti systému ako napríklad frontend, či backend.

V rámci tejto práce sme sa zamerali na niekoľko rôznych aspektov. Ako prvé sme vypracovali prieskum, kde sme sa pokúšali zistiť, ako rozumejú Petriho sieťam bežný ľudia.

Ďalším aspektom bolo vypracovanie analýzy aktuálneho stavu projektu. V analýze sme sa zamerali na nedostatky v projekte, na ktoré sme sa snažili navrhnúť vhodné opravy. Opravy boli rôzneho typu, niektoré sa týkali nedostatkov s scénou, iné zase vo webovom rozhraní.

V práci sme sa okrem opráv snažili implementovať aj nové funkcionality. Zamerali sme sa napríklad na ovládanie scény pomocou rôznych druhov zariadení. Implementačné novinky boli cielené predovšetkým na mobilné zariadenia a VR headsety. Pre mobilné zariadenia sme implementovali vlastný komponent ovládania. Snažili sme sa komponent navrhnúť tak, aby bol čo najviac interaktívny pre používateľov a aby nebol odkázaný na tlačítka na obrazovke. Pre VR headset išlo predovšetkým o opravu implementácie komponentu super-hands, kde nebolo možné prenášať exponáty. Bolo potrebné upraviť implementáciu nášho avatara v scéne. Pridali sme aj možnosť teleportácie, inak povedané blink funkciu. To nám umožňuje sa vo väčších scénach presúvať na väčšie vzdialenosti efektívnejšie. Taktiež sme sa zamerali aj na systémové veci. Upravili sme implementáciu systému tak, aby podporovala zabezpečenú komunikáciu pomocou HTTPS protokolu, ale aj podporu pre vyššie protokoly HTTP.

Prínosom tejto práce je okrem opravených nedostatkov aj rozšírenie o spomínané nové funkcionality, ktoré môžu používateľom zlepšiť zážitok pri využívaní VR systému. Výsledkom práce je fungujúci univerzálny systém pre implemento-

vane scén, riadených Petriho sieťami, zamerané na vzdelávací proces študentov.

Literatúra

1. MAZUREK, Justyna; KIPER, Paweł; CIEŚLIK, Błażej; RUTKOWSKI, Sebastian; MEHLICH, Krzysztof; TUROLLA, Andrea; SZCZEPAŃSKA-GIERACHA, Joanna. *Virtual reality in medicine: A brief overview and future research directions*. Termedia, 2019. Dostupné tiež z: <https://www.termedia.pl/Virtual-reality-in-medicine-a-brief-overview-and-future-research-directions,129,36036,1,1.html>.
2. LIU, Xinxiong; ZHANG, Jing; HOU, Guoxiang; WANG, Zenan. Virtual Reality and Its Application in Military. *IOP Conference Series: Earth and Environmental Science*. 2018, roč. 170, č. 3, s. 032155. Dostupné z doi: 10.1088/1755-1315/170/3/032155.
3. KAVANAGH, Sam; LUXTON-REILLY, Andrew; WUENSCHÉ, Burkhard; PLIMMER, Beryl. A systematic review of Virtual Reality in education. *Themes in Science and Technology Education*. 2017, roč. 10, č. 2, s. 85–119. ISSN 1792-8788. Dostupné tiež z: <https://www.learntechlib.org/p/182115>.
4. *Virtual Reality In Entertainment: The State Of The Industry*. British Academy of Film a Television Arts, 2017.
5. PANTELIDIS, Veronica. Reasons to Use Virtual Reality in Education and Training Courses and a Model to Determine When to Use Virtual Reality. *Themes in Science and Technology Education*. 2009, roč. 2.
6. MARTIN-GUTIERREZ, Jorge; MORA, Carlos; AÑORBE, Beatriz; GONZÁLEZ-MARRERO, Antonio. Virtual Technologies Trends in Education. *Eurasia Journal of Mathematics, Science and Technology Education*. 2017, roč. 13, s. 469–486.
7. WEI, Ziwen; YUAN, Man. Research on the Current Situation and Future Development Trend of Immersive Virtual Reality in the Field of Education. *Sustainability*. 2023, roč. 15, s. 7531. Dostupné z doi: 10.3390/su15097531.

8. LÓPEZ-BELMONTE, Jesús; SÁNCHEZ, Santiago; MORENO GUERRERO, Antonio; LAMPROPOULOS, Georgios. Metaverse in Education: a systematic review. *Revista de Educación a Distancia (RED)*. 2023, roč. 23, s. 1–25. Dostupné z doi: 10.6018/red.511421.
9. SAKULWICHITSINTU, Songlak. Metaverse for Collaborative Online Learning: Short Review. In: *2023 8th International Conference on Information and Network Technologies (ICINT)*. 2023, s. 13–17. Dostupné z doi: 10.1109/ICINT58947.2023.00010.
10. SÁ, Maria José; SERPA, Sandro. Metaverse as a Learning Environment: Some Considerations. *Sustainability*. 2023, roč. 15, č. 3, s. 2186. ISSN 2071-1050. Dostupné z doi: 10.3390/su15032186.
11. ASIKSOY, Gulsum. Empirical Studies on the Metaverse-Based Education: A Systematic Review. *International Journal of Engineering Pedagogy (IJEP)*. 2023, roč. 13, č. 3, pp. 120–133. Dostupné z doi: 10.3991/ijep.v13i3.36227.
12. WANG, Zhonglei; WEN, Tianyu; WU, Weimin. Modeling and simulation of rumor propagation in social networks based on Petri net theory. In: *2015 IEEE 12th International Conference on Networking, Sensing and Control*. 2015, s. 492–497. Dostupné z doi: 10.1109/ICNSC.2015.7116086.
13. RATZER, Anne Vinter; WELLS, Lisa; LASSEN, Henry Michael; LAURSEN, Mads; QVORTRUP, Jacob Frank; STISSING, Martin Stig; WESTERGAARD, Michael; CHRISTENSEN, Søren; JENSEN, Kurt. CPN tools for editing, simulating, and analysing coloured Petri nets. In: *Proceedings of the 24th International Conference on Applications and Theory of Petri Nets*. Eindhoven, The Netherlands: Springer-Verlag, 2003, s. 450–462. ICATPN'03. ISBN 3540403345.
14. PISARČÍK, Lukáš. *Scenárom riadené výučbové prostredia vo virtuálnej realite*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2023. Vedúci práce PhD. ING. ŠTEFAN KOREČKO.
15. DEMIANENKO, Dmytro. *Spracovanie údajov z virtuálnych výučbových prostredí*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2023. Vedúci práce PhD. ING. ŠTEFAN KOREČKO.
16. KAŠELA, Adam. *Výučbové scenáre v rozšírenej realite využívajúce procesné grafy*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2022. Vedúci práce PhD. ING. ŠTEFAN KOREČKO.
17. MERKEL, Dirk. Docker: lightweight linux containers for consistent development and deployment. *Linux journal*. 2014, roč. 2014, č. 239, s. 2.

18. JAIN, Nilesh; BHANSALI, Ashok; MEHTA, Deepak. AngularJS: A modern MVC framework in JavaScript. *Journal of Global Research in Computer Science*. 2014, roč. 5, č. 12, s. 17–23.
19. JOHNSON, Rod; HOELLER, Juergen; DONALD, Keith; SAMPALANU, Colin; HARROP, Rob; RISBERG, Thomas; ARENDSSEN, Alef; DAVISON, Darren; KOPYLENKO, Dmitriy; POLLACK, Mark et al. The spring framework-reference documentation. *interface*. 2004, roč. 21, s. 27.
20. JURE. *aframe-blink-controls: Usage*. n.d. Dostupné tiež z: <https://github.com/jure/aframe-blink-controls?tab=readme-ov-file#usage>. Accessed April 10, 2024.
21. C-FRAME. *aframe-super-hands-component: Compatibility*. n.d. Dostupné tiež z: <https://github.com/c-frame/aframe-super-hands-component#compatibility>. Accessed April 10, 2024.
22. CUI, Xinchun; LI, Comxueqing; QIN, Yiming; DING, Yong. A Password Strength Evaluation Algorithm Based on Sensitive Personal Information. In: *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 2020, s. 1542–1545. Dostupné z doi: 10.1109/TrustCom50675.2020.00211.

Zoznam príloh

Príloha A Používateľská príručka

Príloha B Systémová príručka

Príloha C CD médium – záverečná práca v elektronickej podobe,

Príloha D Výsledky prieskumu

A Používateľská príručka

Používateľská príručka, rovnako ako práca, priamo nadväzuje na práce Dmytra Demianenka[15], Lukáša Pisarčíka[14] a Adama Kašelu [16] a niektoré aspekty používateľskej príručky, ktoré sa našou prácou nijak nemenili sme ponechali v pôvodnom stave.

A.1 Funkcia programu

Táto práca sa zameriava na vývoj systému analýzy údajov v rozšírenej realite (XR), ktorý uľahčuje žiakom plnenie úloh, poskytuje učiteľom dohľad nad úlohami a ponúka administratívne funkcie. Cieľom systému je zhromažďovať údaje o používateľoch počas plnenia úloh a poskytovať podrobné štatistiky o splnených úlohách. Integrácia možnosti analýzy údajov umožňuje získať prehľad v reálnom čase a personalizovanú spätnú väzbu pre používateľov, čo zlepšuje výsledky vzdelávania.

A.2 Inštalácia programu

V tejto kapitole sa rozoberá obsah produktu, po ktorom nasleduje špecifikácia technických a softvérových požiadaviek. Postup inštalácie produktu je opísaný z pohľadu používateľa (administrátora aj bežného používateľa).

A.2.1 Požiadavky na technické prostriedky

Na spustenie projektu musia byť splnené tieto minimálne hardvérové požiadavky pre WebGL, a to:

- 64-bitový operačný systém
- 4 GB pamäte RAM
- Integrovaná grafika s podporou WebGL 2.0 OpenGL ES 3.0

- Dvojjadrový procesor Intel/AMD
- Najnovšia verzia prehliadača Chrome, Firefox alebo Opera

A.2.2 Spustenie projektu

Táto časť opisuje kroky potrebné na otvorenie projektu vo webovom prehliadači pre bežných používateľov alebo na nasadenie projektu na server pre administrátorov.

Otvorenie projektu pre používateľov

Každý používateľ, ktorý má prístup k sieti VPN TUKE alebo k sieti Wi-Fi TUKE, môže tento projekt otvoriť v ľubovoľnom webovom prehliadači na adrese <http://147.232.205.222:4200>, kde sa potom môže zaregistrovať alebo prihlásiť.

Nasadenie aplikácie pre administrátorov

Ak administrátor potrebuje nasadiť projekt na server, môže toto urobiť pomocou nástroja Docker¹.

Po nainštalovaní nástroja Docker stačí na úspešné spustenie projektu len niekoľko krokov. Najprv je potrebné vytvoriť prázdny priečinok na serveri 147.232.205.222 s dvoma súbormi: `docker-compose.yaml` a `servers.json`.

Súbor `docker-compose.yaml` slúži na definovanie služieb, sietí a zväzkov potrebných pre nastavenie nástroja Docker. Umožňuje konfigurovať a spravovať viacero kontajnerov ako jednu aplikáciu. Tu je príklad základného súboru `docker-compose.yaml`:

```
version: '3.8'

services:

  db-postgres:
    image: "postgres:15.1"
    container_name: lirkis-database
    volumes:
```

¹Docker je open-source platforma, ktorá umožňuje kontajnerizáciu aplikácií a poskytuje izolované a prenosné prostredie. Viac informácií nájdete na adrese: <https://www.docker.com/>

```
- lirkis-data:/var/lib/postgresql/data
ports:
  - "5432:5432"
environment:
  - POSTGRES_DB=postgres
  - POSTGRES_USER=postgres
  - POSTGRES_PASSWORD=postgres

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

spring:
  build: yuray/LirkisEduVePn-service
  container_name: lirkis-service
  ports:
    - "8443:8443"
  links:
    - db-postgres

angular:
  build: yuray/LirkisEduVePn-UI
  container_name: lirkis-ui
  ports:
    - "4200:443"
  links:
    - spring

volumes:
  lirkis-data:
```

V tomto súbore máme definované štyri služby: db-postgres, pgadmin, spring a angular.

Služba "db-postgres" používa obraz "postgres", nastavuje premenné prostredia pre predvoleného používateľa, heslo a názov databázy a pripája adresár ./data ako zväzok na uchovávanie údajov databázy. Databáza bude prístupná na porte 5432.

Služba "spring" je prevzatá z obrazu "aftermath1235/lirkiseduvepn-spring", ktorý bol zostavený počas písania systému a vystavuje port 8080 na hostiteľovi, pričom ho mapuje na port 8080 v kontajneri. Táto služba funguje ako server.

Služba "angular" je prevzatá z obrazu "aftermath1235/lirkiseduvepn-angular", ktorý bol vytvorený počas písania systému a vystavuje port 4200 na hostiteľovi a mapuje ho na port 80 v rámci kontajnera. Táto služba funguje ako klient.

Služba "pgadmin" používa obraz "dpage/pgadmin4", nastavuje premenné prostredia pre predvolený e-mail a heslo používateľa admin a vystavuje port 3000 na hostiteľovi, pričom ho mapuje na port 80 v rámci kontajnera. Okrem toho pripája súbor ./servers.json ako zväzok, ktorý tiež musí byť nakonfigurovaný.

Nižšie je uvedený súbor "servers.json", ktorý používa služba "pgadmin" na konfiguráciu pripojenia k serveru PostgreSQL:

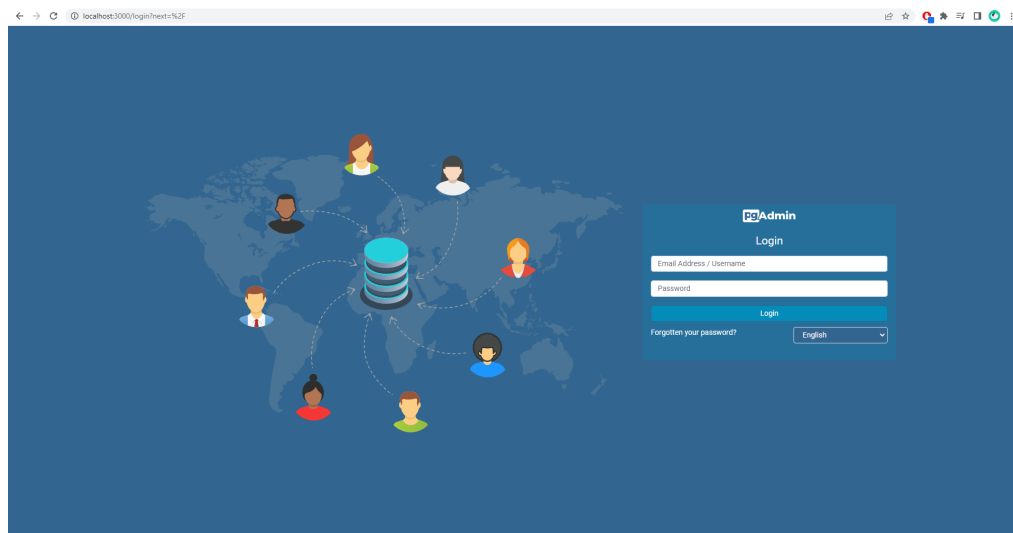
```
{
  "Servers": {
    "1": {
      "Name": "Local",
      "Group": "Servers",
      "Host": "db-postgres",
      "Port": 5432,
      "MaintenanceDB": "postgres",
      "Username": "postgres",
      "SSLMode": "prefer",
      "SSLCert": "<STORAGE_DIR>/.postgresql/postgresql.crt",
      "SSLKey": "<STORAGE_DIR>/.postgresql/postgresql.key",
      "SSLCompression": 0,
      "Timeout": 10,
      "UseSSHTunnel": 0,
    }
  }
}
```

```
"TunnelPort": "22",  
"TunnelAuthentication": 0  
}  
}  
}
```

Po vytvorení dvoch vyššie uvedených súborov stačí na spustenie programu napísať príkaz "docker compose up", ak používate systém Windows, alebo "docker-compose up", ak používate systém Linux.

Po spustení programu môžete prejsť na stránku <https://147.232.205.222:4200>, ktorá bude mať všetky funkcie rovnaké ako v používateľskom režime.

Veľmi dôležitým aspektom je aj pridanie správcu do systému. Keďže pri písaní práce nebolo možné zabezpečiť, aby sa používateľ s rolou administrátora pri vytváraní databázy pomocou skriptu SQL pridával automaticky, je potrebné pridať tohto používateľa ručne. Ak to chcete urobiť, musíte najprv prejsť na stránku <http://147.232.205.222:3000>, ktorá je zodpovedná za grafický panel databázy. Na obrázku A.1 je možné vidieť prihlasovacie okno.



Obr. A.1: Prihlasovacia obrazovka do pgAdmin

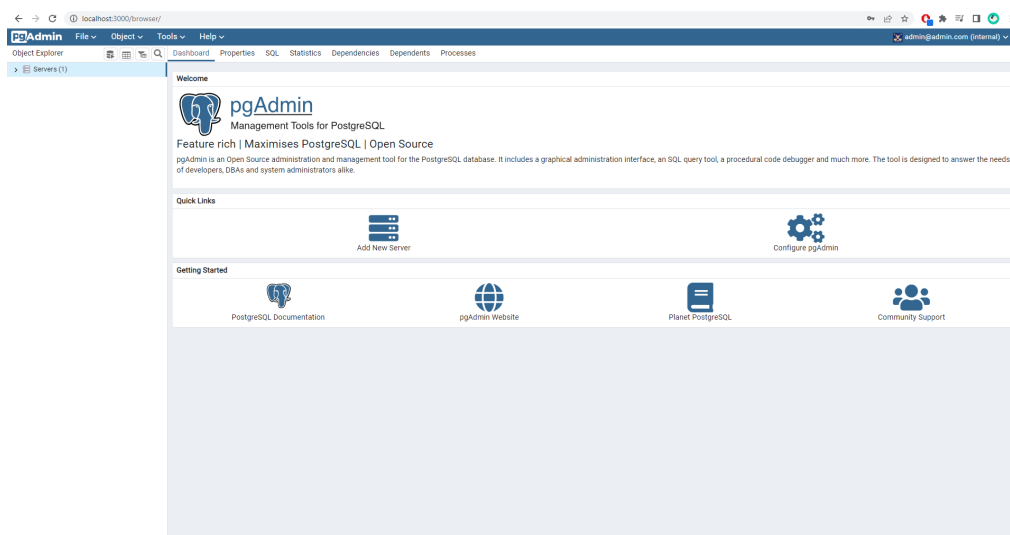
Ak sa chcete prihlásiť, musíte zadať údaje uvedené v súbore "docker-compose.yml" v službe "pgAdmin". V našom prípade je to e-mail "admin@admin.com" a heslo "admin". Po overení sa otvorí ovládací panel. Na obrázku A.2 vľavo hore v rohu vidíte počet registrovaných serverov, v ktorých bude naša databáza. Ak sa pokúsite otvoriť našu databázu, zobrazí sa okno s

požiadavkou na zadanie hesla na prístup do databázy. Heslo je "postgres". Po zadaní hesla sa otvorí naša databáza, ktorú môžete vidieť na obrázku A.3.

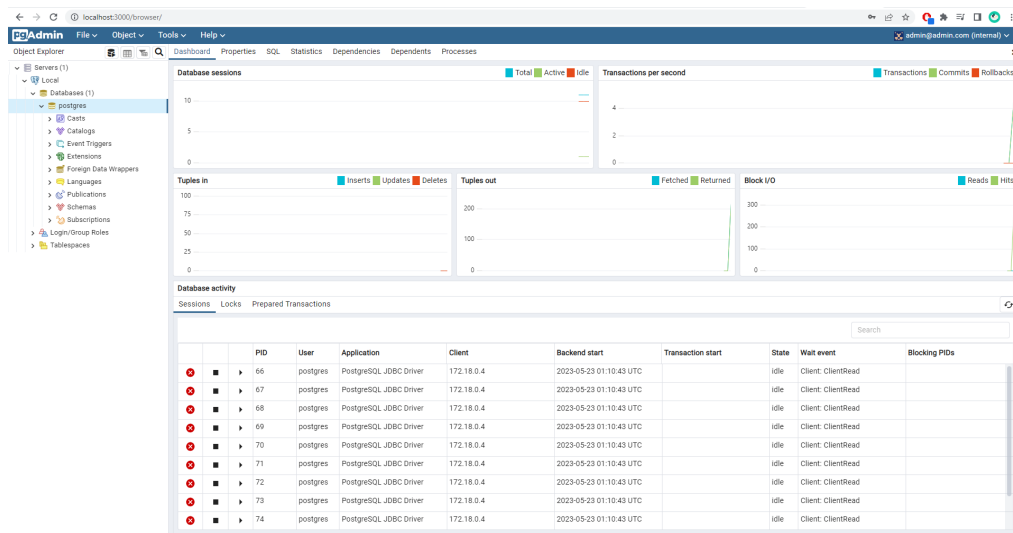
Po otvorení databázy kliknite na tlačidlo "Query Tool" alebo "Alt + Shift + Q", čím otvoríte editor dotazov SQL. Po otvorení editora zadajte nasledujúci dotaz, ktorý pridá nového používateľa s administrátorskými právami:

```
INSERT INTO users(
    email, firstname, is_enabled,
    lastname, username, password, role
)
VALUES (
    'admin@admin', 'admin', true, 'admin', 'admin',
    '$2a$12$3HMoKSw7wbRayzLmoApieJTfvP0oqtC5ZTYmsPyo4PSJoy9TeDmW',
    'ADMIN'
);
```

Po pridaní používateľa s administrátorskými právami sa môžete autentifikovať na serveri pomocou používateľského mena "admin@admin" a hesla "admin", ktoré potom môžete zmeniť. Po autentifikácii bude môcť administrátor využívať všetky funkcie systému.



Obr. A.2: Okno so servermi v pgAdmin



Obr. A.3: Databáza v pgAdmin

A.3 Použitie programu

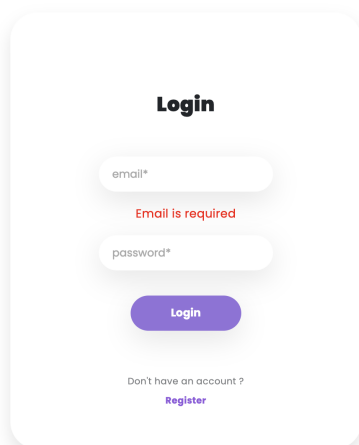
Tento program ponúka užívateľom možnosť vykonávať úlohy v virtuálnom prostredí na základe predom vytvorených scénárov. Okrem toho užívateľ môže vytvárať a upravovať vlastné scénáre a má prístup k histórii svojich predchádzajúcich pokusov. V tejto kapitole nájdete podrobný návod, ako používať aplikáciu.

A.3.1 Prihlasovacia obrazovka

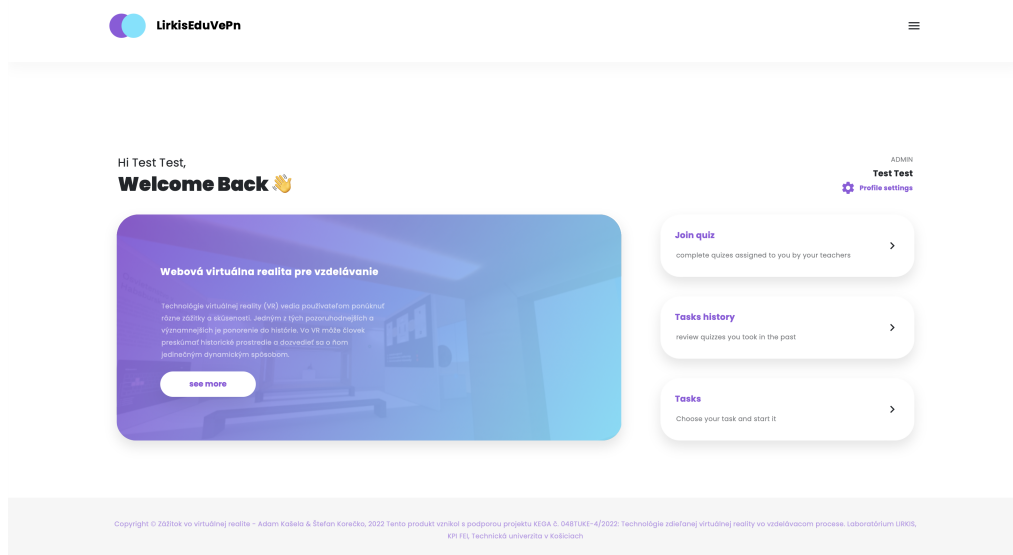
Pri spustení aplikácie sa zobrazí prihlasovacia obrazovka, kde môžete zadať svoje prihlasovacie údaje, ako je používateľské meno a heslo. Ak ste ešte nevytvorili účet, máte možnosť sa zaregistrovať prostredníctvom príslušného odkazu. Po zadaní správnych prihlasovacích údajov a kliknutí na tlačidlo **Prihlásiť sa**, budete presmerovaní do hlavného rozhrania aplikácie.

A.3.2 Hlavná obrazovka

Hlavné menu pre žiakov bolo navrhnuté tak, aby poskytovalo prehľad o všetkých úlohách, ktoré im učiteľ pridelil, a zároveň im umožňovalo zobrazíť si výsledky. Pri výbere niektorej z úloh má užívateľ na výber z niekoľkých jazykov, taktiež má možnosť pokračovať v úlohe ak ju už začali alebo danú úlohu ukončiť, vidno na obrázku A.6. Okrem toho obsahuje aj možnosť prístupu k nastaveniam profilu a zmeny informácií o účte. Hlavnú obrazovku je možné vidieť na obrázku A.5.



Obr. A.4: Prihlasovacie okno

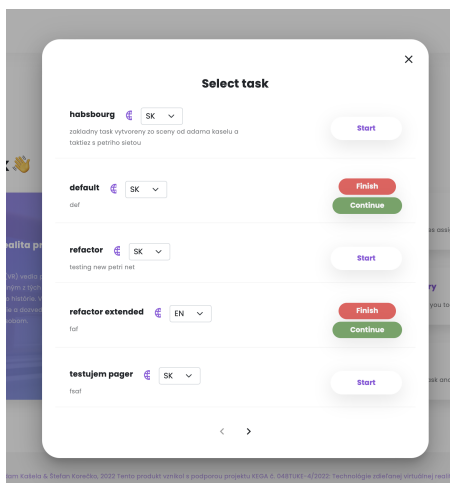


Obr. A.5: Hlavná obrazovka

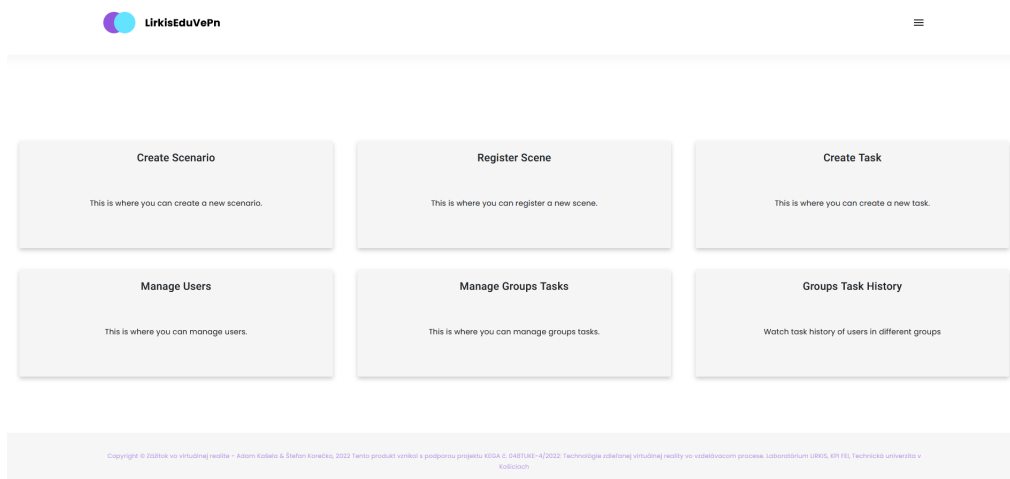
A.3.3 Dashboard pre učiteľov a adminov

Dashboard pre učiteľov a adminov je navrhnutý tak, aby im poskytoval efektívne nástroje na správu a organizáciu vzdelávacieho obsahu a používateľov. Nasledujúce sú niektoré z hlavných funkcií, ktoré tento dashboard ponúka:

- **Vytvoriť scénar** Učítelia a admini majú možnosť vytvoriť nový scénar, kde pridajú scénar v podobe Petriho siete a jazykový súbor s popiskami.
- **Vytvoriť scénu** Pri vytváraní scénarov učítelia a admini môžu vytvárať jednotlivé scény, ktoré sú už predpripravené.



Obr. A.6: Výber úlohy



Obr. A.7: Používateľský dashboard

- **Vytvoriť úlohu** Učítelia a admini majú možnosť vytvoriť úlohy pre študentov, kde zadajú názov, popis, scénar a scénu použitú pre úlohu.
- **Spravovať používateľov** Pre administrátorov je dostupná možnosť spravovať používateľov systému. Môžu vytvárať, upravovať a odstraňovať používateľské účty, priradiť ich do skupín, a spravovať ich oprávnenia a prístup k obsahu.
- **Spravovať skupiny** Učítelia a admini majú možnosť vytvárať a spravovať skupiny študentov. Tieto skupiny môžu slúžiť na organizáciu študentov podľa tried, kurzu alebo iných kritérií. Učítelia môžu priradiť úlohy skupinám a sledovať ich pokrok a výsledky.
- **História úloh skupín** Tento dashboard tiež umožňuje prehľadávať históriu úloh skupín. Učítelia a admini majú prístup k predošlým výkonom a vý-

sledkom študentov v rámci jednotlivých úloh. Môžu si pozrieť štatistiky a pokrok skupín v čase. Ako je možné vidieť na obrázku A.8, po vybratí skupiny má učiteľ k dispozícii všetky vykonané úlohy študentov v rámci skupiny. Pomocou tlačidla *preview* si vie zobrazíť štatistiky pre danú úlohu. Následne pomocou tlačidla *View extended* môže zobrať odpalenia jednotlivých prechodov spolu s informáciami či bol prechod odpálený úspešne, kedy sa odpálil a samozrejme aj názov prechodu.

The image shows two screenshots of the LirkisEduVePn web application. The top screenshot displays a list of sessions under the heading "Sessions in the group". Each session entry includes the title "museum extended", the date "2023-04-26", and the text "Test Test". A "preview" button is located to the right of each entry. Below the list is a "Back" button.

The bottom screenshot shows a detailed view of a session. The user name is "Test Test" and the task name is "museum extended". The start of the task is "27/04/2023 12:23" and the finish is "27/04/2023 12:25". The status is "WAS FINISHED SUCCESSFULLY" with a "Yes" response. The test duration is "2 min 31 sec". Below this information are four statistics: "TOTAL FIRING ATTEMPTS" (14), "CORRECT FIRING ATTEMPTS" (14), "INCORRECT FIRING ATTEMPTS" (0), and "PERCENTAGE OF CORRECT ATTEMPTS" (100%). At the bottom of this view are "View extended" and "close" buttons.

Obr. A.8: Prehľad histórie úloh

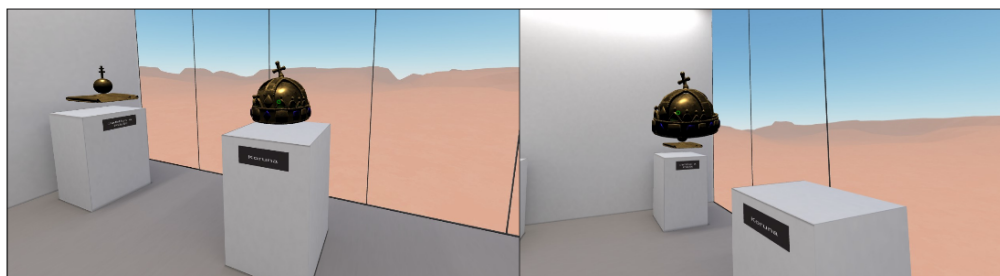
A.3.4 Virtuálna scéna

Virtuálna scéna je prostredie vytvorené v rámci programu, ktoré umožňuje interakciu a simuláciu rôznych situácií. Je to priestor, kde sa odohrávajú úlohy a aktivity pre používateľov. Scéna obsahuje rôzne typy interakcií, ktoré majú používateľom napomôcť alebo umožniť dokončiť stanovené úlohy v scéne. Používateľ môže v scéne uskutočňovať tieto interakcie:

- Prenášanie exponátov
- Označovanie exponátov
- Zobrazovanie informácií k exponátom

A.3.5 Prenášanie exponátov

V jednej z vytvorených miestností používateľ má za úlohu prenášať exponáty na vopred určené miesto. Táto interakcia je zobrazená na obrázku A.9. Používateľ musí umiestniť zelený kurzor na objekt a následne kliknutím a držaním ľavého tlačidla počítačovej myši môže následným pohybom prenášať 3D objekt. V nasledujúcom obrázku je akcia ukázaná pred uchopením a po uchopení objektu zľava doprava.



Obr. A.9: Interakcie prenášania exponátov

A.3.6 Označovanie exponátov

Táto interakcia v scéne je znázornená na obrázku A.10. Slúži na vyberanie správnych alebo nesprávnych exponátov v jednotlivých miestnostiach s úlohami. Ľavý obrázok znázorňuje neoznačený exponát a pravý obrázok znázorňuje označený exponát. Interakcia sa vykonáva klikom počítačovej myši po umiestnení zeleného kurzora na 3D objekt v scéne.



Obr. A.10: Interakcie označovania exponátov

A.3.7 Zobrazovanie informácií k exponátom

Používateľ pri preskúvaní virtuálnej scény môže zobrazovať informácie k jednotlivým exponátom. Tento typ interakcie je znázornený na obrázku A.11. Po umiestení kurzora na modrú ikonu pred exponátom a nasledovaním kliknutím ľavého tlačidla myši vie používateľ, buď zobraziť alebo skryť informácie k exponátu.



Obr. A.11: Interakcie zobrazenia informácie

A.3.8 Zvukové znamenia

Pri uskutočňovaní jednotlivých interakcií je používateľ sprevádzaný zvukovými znamienami. Jednotlivé zvuky a ich význam:

- **Úspešná odpoveď** - Zvukový signál, ktorý sa prehraje pri vykonaní potvrdenia na paneli úlohy v prípade keď odpoveď je správna.
- **Neúspešná odpoveď** - Zvukový signál, ktorý sa prehraje pri vykonaní potvrdenia na paneli úlohy v prípade keď odpoveď je nesprávna.
- **Bežná akcia** - Zvuky použité pri bežnej interakcii s expozíciami.
- **Úspešné vykonanie všetkých úloh** - Po dokončení všetkých úloh zaznie tento zvuk.

B Systémová príručka

B.1 Funkcia programu

Táto práca sa zameriava na vývoj systému analýzy údajov v rozšírenej realite (XR), ktorý uľahčuje žiakom plnenie úloh, poskytuje učiteľom dohľad nad úlohami a ponúka administratívne funkcie. Cieľom systému je zhromažďovať údaje o používateľoch počas plnenia úloh a poskytovať podrobné štatistiky o splnených úlohách. Integrácia možností analýzy údajov umožňuje získať prehľad v reálnom čase a personalizovanú spätnú väzbu pre používateľov, čo zlepšuje výsledky vzdelávania.

B.2 Popis programu

V tejto časti opíšeme základnú architektúru riešenia, taktiež opíšeme všetky implementované modely, moduly a komponenty, opíšeme dôležité algoritmy a premenené.

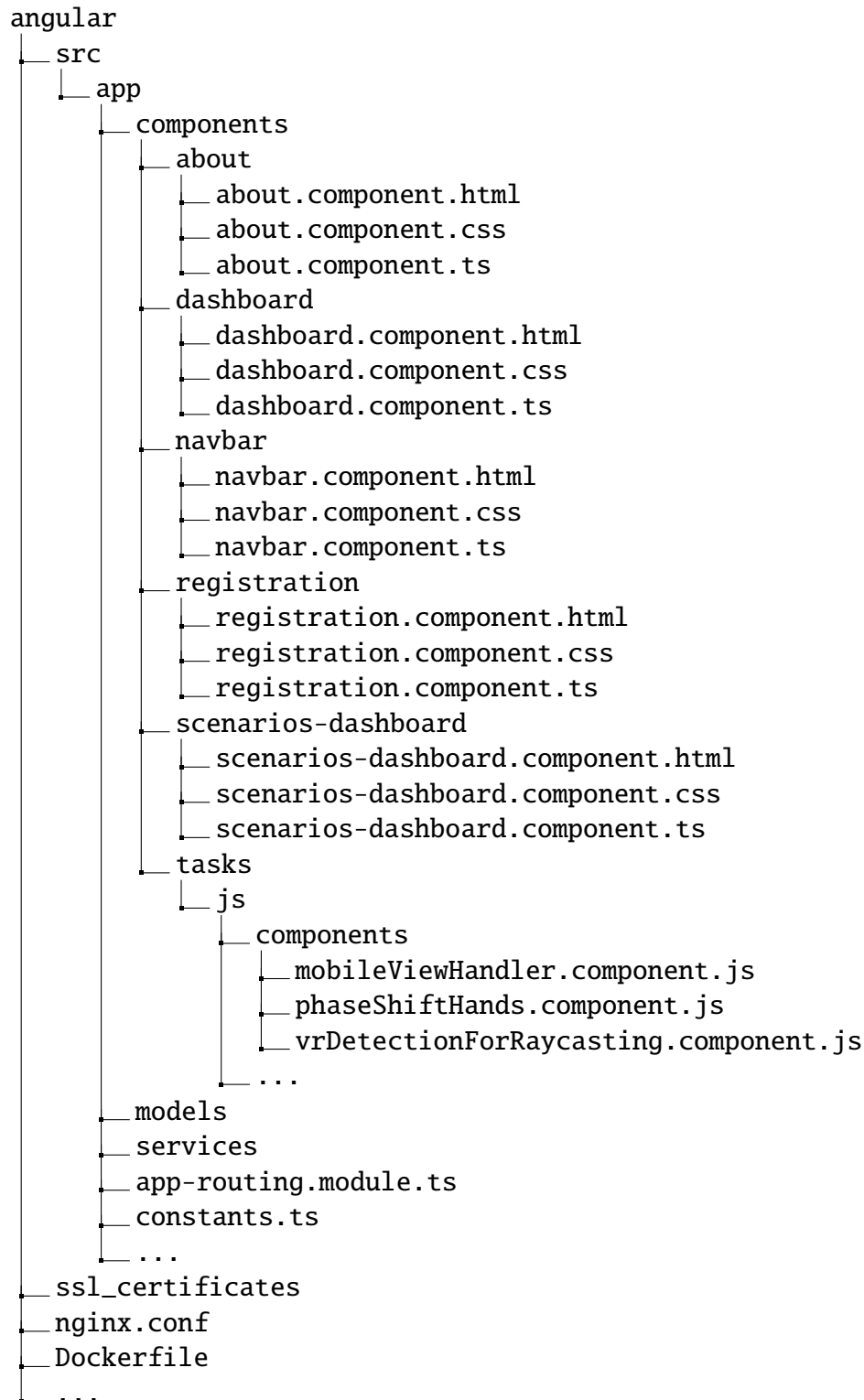
B.2.1 Popis riešenia

Riešenie je založené na ECS (Entity-Component-System) architektúre. ECS umožňuje oddelenie dátových entít (entity) od ich správania (component) a systémov, ktoré tieto entity spracúvajú. ECS architektúru sme obohatili o webový rámec Angular, ktorý poskytuje robustné nástroje na tvorbu užívateľského rozhrania. Angular nám umožňuje efektívne spravovať komponenty a moduly, čo je veľmi dôležité pri rozsiahlejších projektoch.

B.2.2 Štruktúra angular projektu

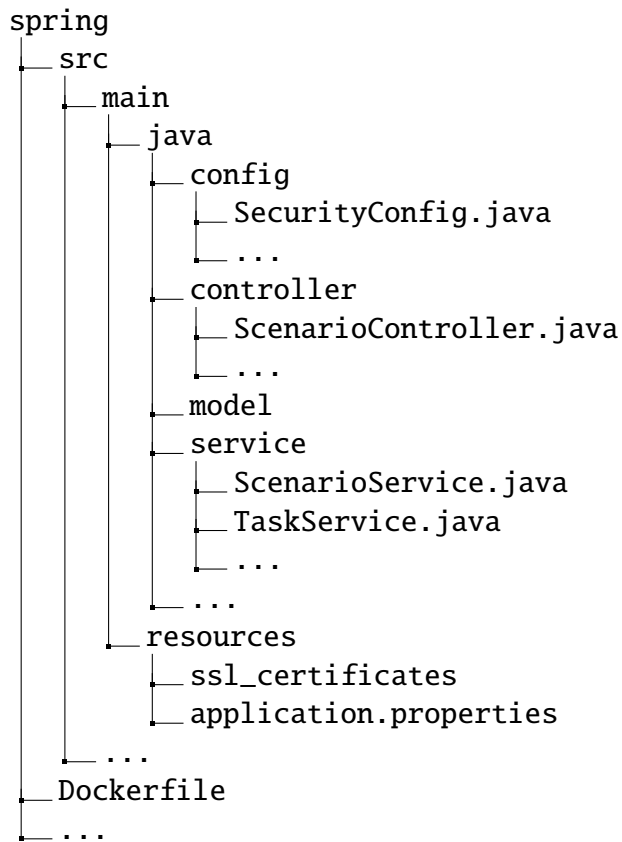
Na obrázku nižšie je zobrazená základná štruktúra angularovej časti projektu, kde sú zobrazené iba najpodstatnejšie súbory, ktoré sme v práci implementovali,

alebo upravovali. Medzi angulárové komponenty patria aj samotné scény, pričom každá scéna obsahuje skripty, komponenty, ktoré sú špecifické iba pre danú scénu. Komponenty, ktoré sú použiteľné pre viacero scén, sú definované v priečinku js. V našom prípade sú v tomto priečinku definované ovládacie komponenty pre rôzne typy zariadení.



B.2.3 Štruktúra java projektu

Na obrázku nižšie máme zobrazenú štruktúru časti projektu s javou, kde sú zobrazené iba najpodstatnejšie súbory.



B.3 Popis implementovaných súborov

B.3.1 Aframe komponenty

- **mobileViewHandler.component.js** - tento komponent rieši problematiku ovládania scény pomocou mobilného zariadenia
- **phaseShiftHands.component.js** - tento pomocný komponent pre super-hands slúži na to, aby sme virtuálnymi rukami nenarážali do objektov v scéne
- **vrDetectionForRaycasting.component.js** - slúži ako detekciu prepnutia do VR režimu. Následne po presune do VR režimu sa zo scény odstráni ray-caster, ktorý slúži pre ovládanie počítačom

B.3.2 Angular komponenty

Webové komponenty, ktoré sú rozložené do súborov html, css a ts, budeme v zozname nižšie uvádzať ako jeden komponent.

- **about.component** - súbory definujúce stránku about
- **dashboard.component** - súbory definujúce administrátorský dashboard
- **navbar.component** - súbory definujúce komponent navbar
- **registration.component** - súbory definujúce stránku pre registráciu
- **scenarios-dashboard.component** - súbory definujúce stránku pre manažovanie scenárov
- **app-routing.module.ts** - komponent, ktorý definuje navigačnú logiku, cesty pre rôzne časti aplikácie a konfiguráciu routovania
- **constants.ts** - súbor definujúci všeobecné globálne premenné.
- **ssl_certificates** - v tomto repozitári sú uložené potrebné súbory pre ssl certifikáciu.
- **nginx.conf** - súbor pre základnú konfiguráciu frondovej časti servera.

B.3.3 Java komponenty

- **SecurityConfig.java** - súbor definujúci základné bezpečnostné konfigurácie. V tomto súbore sme upravili konfiguráciu CORS pre prijímanie serverovej IP adresy
- **ScenarioController.java** - súbor definujúci REST endpoint pre vykonanie služieb so scenármi. Tu sme implementovali REST endpoint na odstránenie scenára
- **ScenarioService.java** - súbor obsahujúci metódy na manažovanie scenárov. Tu sme implementovali logiku pre odstránenie scenára
- **TaskService.java** - súbor obsahujúci metódy na manažovanie úloh. Úloha sa skladá zo scény a scenára. V tomto súbore sme implementovali logiku odstránenia úlohy, ktorú je potrebné odstrániť, ak sme odstránili jej príslušný scenár
- **ssl_certificates** - v tomto repozitári sú uložené potrebné súbory pre ssl certifikáciu.
- **application.properties** - v tomto súbore je definovaná základná konfigurácia backendového systému.

B.4 Využité technológie

Technológie, ktoré sme použili na vytvorenie aplikácie. V zozname je uvedená aj minimálna verzia technológie, ktorá je potrebná na úspešné spustenie programu:

- *Aframe (1.4.1)*
- *Angular (15.0.4)*
- *Bootstrap (5.2.3)*
- *Java (17)*
- *PostgreSQL (15.3)*
- *Docker*

B.5 Spustenie programu

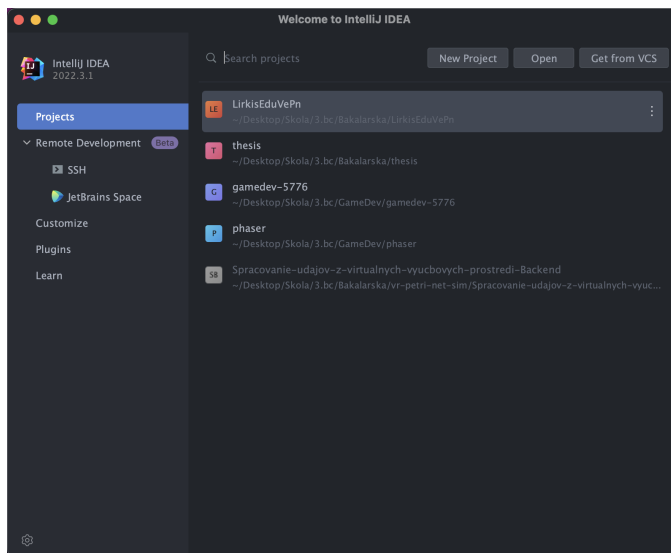
Na lokálny vývoj a spustenie aplikácie lokálne je potrebné postupovať podľa nasledujúceho postupu:

1. Stiahnutie zdrojových kódov

Medzi prílohami práce sa nachádzajú aj zdrojové kódy aplikácie, je potrebné si ich stiahnuť a následne odbaliť komprimovaný balíček

2. Otvorenie v IDE

Po stiahnutí potrebných súborov si projekt otvorte v ľubovoľnom IDE. Odporúčame IntelliJ IDEA od spoločnosti JetBrains, ktorý aj v návode budeme používať. Po otvorení IntelliJ sa vám zobrazí úvodné okno ako je možné vidieť na obrázku B.1, následne je potrebné pomocou tlačidla *Open* vyhľadať rozbalený priečinok so zdrojovými kódmi aplikácie.



Obr. B.1: Otvorenie projektu

3. Príprava projektu na spustenie

Pred spustením aplikácie je najprv potrebné stiahnuť všetky potrebné závislosti. Je potrebné mať nainštalovanú minimálne Javu 17, PostgreSQL 15.3 a NodeJS (verziu 8 alebo novšiu) a NPM.

Nastavenie databázy

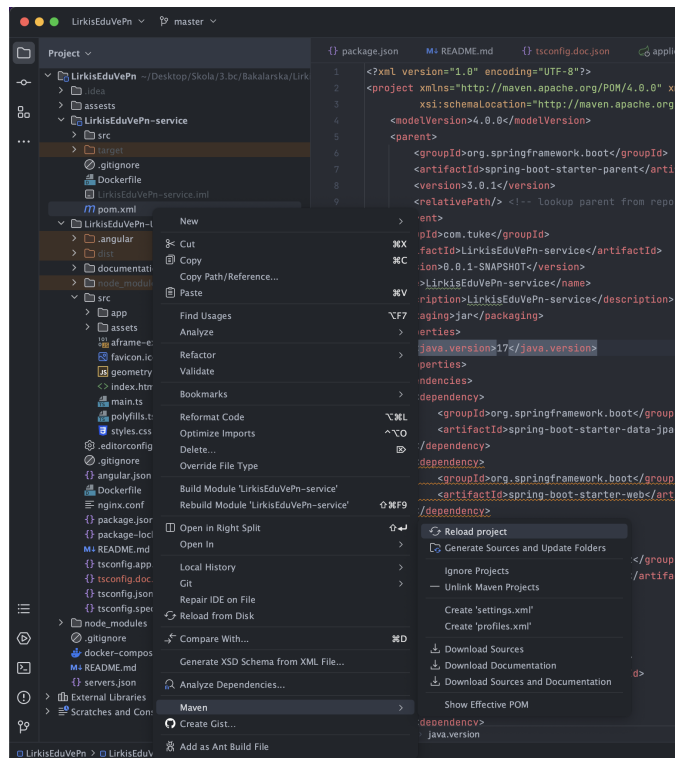
Ďalej je potrebné mať správne nakonfigurovanú databázu. Po nainštalovaní PostgreSQL vytvorte lokálnu prázdnu databázu v termináli (psql) alebo desktopovej aplikácii (pgAdmin), ktorú poskytuje PostgreSQL počas inštalácie. Následne je potrebné nastaviť vytvorenú databázu pre SpringBoot a to

v súbore ktorý nájdete v `LirkisEduVePn-service/src/main/resources/application.properties`. V súbore `application.properties` je potrebné upraviť `url`, `username` a `password` podľa nastavení PostgreSQL. Príklad súboru je vidno na nasledujúcom kóde.

```
spring.datasource.url=jdbc:postgresql://localhost:5432/postgres
spring.datasource.username=postgres
spring.datasource.password=postgres
spring.jpa.hibernate.ddl-auto=update
jwt.secret=655368566D5971337436773979244226452948404D635166546A
jwt.expiration=120
spring.servlet.multipart.max-file-size=10MB
spring.servlet.multipart.max-request-size=10MB
```

Inštalácia závislosti

Na spustenie serverovej časti je potrebné mať stiahnuté všetky potrebné závislosti, v našom prípade používame na inštaláciu *Maven*, vďaka ktorému je inštalácia veľmi jednoduchá. Stačí niekde v priečinku `LirkisEduVePn-service` stlačiť pravým tlačidlom a následne vybrať možnosť *Maven* a **Reload project**. Výber by mal vyzeráť podobne ako na obrázku B.2.



Obr. B.2: Maven závislosti

Rovnako aj na správne fungovanie klientskej časti je potrebné nainštalovať

potrebné knižnice, ktoré sú definované v súbore *package.json*. Je potrebné sa premiestniť do priečinka *LirkisEduVePn-UI* a do terminálu zadať nasledujúci príkaz:

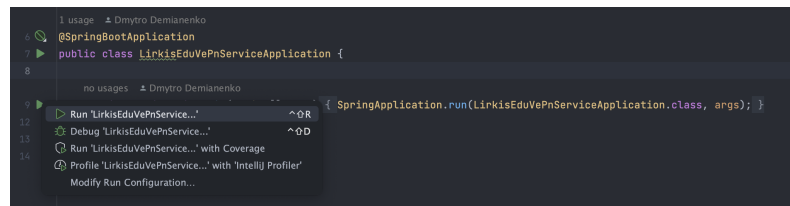
```
npm install
```

4. Spustenie projektu

Na fungovanie aplikácie je potrebné zvlášť spustiť klientsku a serverovú časť. Klientsku časť spustíme v termináli v priečinku *LirkisEduVePn-UI*, pomocou príkazu:

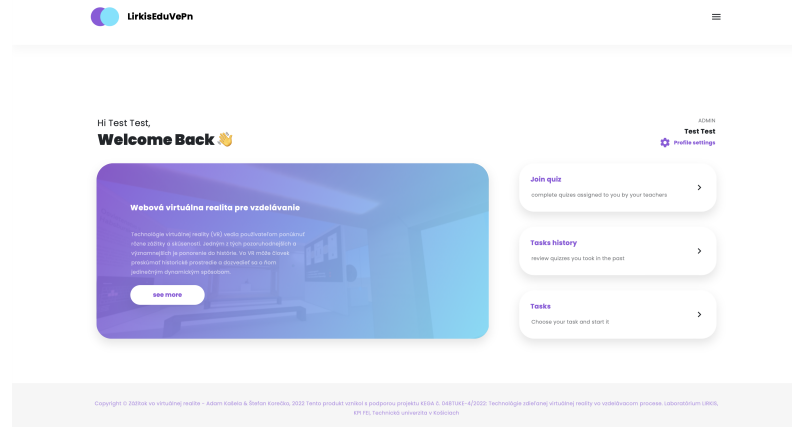
```
ng serve
```

Na spustenie serverovej časti je potrebné otvoriť súbor *LirkisEduVePnServiceApplication.java*, ktorý sa nachádza na ceste *LirkisEduVePn-service/src/main/java/com/tuke/lirkiseduvepnservice/LirkisEduVePnServiceApplication.java*. Následne je potrebné spustiť server, pomocou zelenej šípky ako je vidno na obrázku B.3.



Obr. B.3: Spustenie servera

Následne je aplikácia dostupná na adrese <https://localhost:4200>, ktorú treba zadať do prehliadača. Následne je potrebné sa do systému prihlásiť alebo zaregistrovať ak nemáte vytvorený účet. Po úspešnom prihlásení sa zobrazí úvodná obrazovka ako na obrázku B.4.



Obr. B.4: Úvodná obrazovka aplikácie

B.6 Nasadenie na server

Pre nasadenie novej verzie na server je potrebný nasledovný postup:

1. Nahrať potrebné obrazy kontajnerov do dockerhubu

Pre nahranie obrazov (container image) do dockerhubu odporúčame používať tagy. Tie slúžia na rozoznávanie rôznych verzií nahratých obrazov. Najskôr si vylistujeme zoznam obrazov. Potom si nad konkrétnym obrazom vytvoríme tagovanú kópiu. Následne pushneme obraz do dockerhubu. Ako príklad použijeme tag v tvare "deploy_v2.2". Tieto kroky sú znázornené na obrázkoch nižšie.

```
LirkisEduVePn> docker image ls
REPOSITORY          TAG       IMAGE ID       CREATED        SIZE
lirkiseduvepn-angular  latest   5fb7be35f990  4 hours ago   219MB
lirkiseduvepn-spring  latest   06f09e8bb493  7 days ago    991MB
yuray/lirkiseduvepn-angular  deploy   8d78cbe760f7  7 weeks ago   213MB
yuray/lirkiseduvepn-spring  deploy   3f0eb4cb3488  7 weeks ago   991MB
dpage/pgadmin4        7.1      03adfa2b5a55  11 months ago 491MB
postgres              15.1     ccd94e8b5fd9  14 months ago 379MB
```

Obr. B.5: Vylistovanie obrazov

```
LirkisEduVePn> docker tag 5fb7be35f990 yuray/angular:deploy_v2.2
```

Obr. B.6: Tagovanie obrazu

```
LirkisEduVePn> docker tag 5fb7be35f990 yuray/angular:deploy_v2.2
```

Obr. B.7: Push do dockerhubu

Tento postup je potrebné vykonať pre spring aj pre angular obrazy.

2. Stiahnutie obrazov na server

Po prihlásení sa na server je potrebné obrazy z dockerhubu stiahnuť. Ako príklad uvidíme stiahnutie obrazu s rovnakým tagom, ako sme použili v predošlom kroku.

```
docker pull yuray/angular:deploy_v2.2
docker pull yuray/spring:deploy_v2.2
```

Keď máme stiahnuté obrazy, potrebujeme si buď stiahnuť, alebo vytvoriť súbor

```
docker-compose.yml .
```

3. Spustenie systému

Pre spustenie systému použijeme tento príkaz:

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

V tomto prípade nám prepínač "-- build" nainštaluje nové obrazy a prepínač "-d" nám uvoľní terminál po spustení kontajnerov, v opačnom prípade bude terminál blokovaný do doby, kým sa kontajnery zase nevypnú.