

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

**Distribúované a modulárne riešenie
neurorehabilitácie vo virtuálnej realite**

Bakalárska práca

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

**Distribúované a modulárne riešenie
neurorehabilitácie vo virtuálnej realite**

Bakalárska práca

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

Košice 2026

Martin Čižmárík

Abstrakt v slovenčine

Táto bakalárska práca sa zaoberá návrhom a implementáciou novej verzie neurorehabilitačnej aplikácie vo virtuálnej realite vytvorenej v prostredí Unity. Cieľom práce je vytvoriť softvérové riešenie, ktoré bude jednoducho škálovateľné, rozšíriteľné a vhodné pre ďalší vývoj v oblasti neurorehabilitácie. Nová verzia aplikácie vychádza z predchádzajúcej implementácie, pričom dôraz je kladený na zlepšenie architektúry a sieťovej komunikácie.

Pri návrhu riešenia boli analyzované a porovnané dve sieťové technológie určené pre Unity aplikácie, konkrétne FishNet a Mirror. Na základe vykonanej analýzy bola vybraná vhodnejšia technológia, ktorá bola následne integrovaná do vytvoreného systému. Implementácia zahŕňala úpravu existujúcej architektúry aplikácie a návrh mechanizmov umožňujúcich synchronizáciu vo virtuálnom prostredí.

Výsledkom práce je prototyp neurorehabilitačnej aplikácie podporujúcej moderné sieťové riešenie s dôrazom na modularitu a budúcu rozšíriteľnosť.

Kľúčové slová v slovenčine

Unity, neurorehabilitácia, virtuálna realita, FishNet, Mirror

Abstrakt v angličtine

This bachelor's thesis focuses on the design and implementation of a new version of a neurorehabilitation application developed in the Unity virtual reality environment. The aim of the thesis is to create a software solution that is easily scalable, extensible, and suitable for further development in the field of neurorehabilitation. The new version is based on a previous implementation, with emphasis placed on improving the application architecture and network communication.

As part of the solution design, two networking technologies for Unity applications, FishNet and Mirror, were analyzed and compared. Based on the performed analysis, the more suitable technology was selected and subsequently integrated into the developed system. The implementation included modifications of the existing application architecture and the design of synchronization mechanisms within the virtual environment.

The result of the thesis is a prototype neurorehabilitation application utilizing a modern networking solution with a focus on modularity and future extensibility.

Klíčové slová v angličtine

Unity, neurorehabilitation, virtual reality, FishNet, Mirror

Bibliografická citácia

ČIŽMÁRIK, Martin. *Distribúované a modulárne riešenie neurorehabilitácie vo virtuálnej realite*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2026. 42 s. Vedúci práce: doc. Ing. Štefan Korečko, PhD.

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

ZADANIE
BAKALÁRSKEJ PRÁCE

Študijný odbor: **Informatika**

Študijný program: **Informatika**

Názov práce:

Distribúované a modulárne riešenie neurorehabilitácie vo virtuálnej realite

Distributed and Modular Solution for Neurorehabilitation in Virtual Reality

Študent:

Martin Čižmárik

Školiteľ:

doc. Ing. Štefan Korečko, PhD.

Školiace pracovisko:

Katedra počítačov a informatiky

Konzultant práce:

Pracovisko konzultanta:

Pokyny na vypracovanie bakalárskej práce:

1. Oboznámiť sa so súčasnou verziou systému L-NeRVEn pre kolaboratívnu neurorehabilitáciu vo virtuálnej realite (VR) a podobnými riešeniami.
2. Analyzovať existujúce riešenia pre zdieľané virtuálne prostredia v hernom rámci Unity.
3. Na základe analýzy a pokynov vedúceho práce navrhnuť a implementovať prototyp novej verzie systému L-NeRVEn s vybraným riešením pre zdieľanie a modulárnou architektúrou.
4. Návrh a implementáciu koordinovať s ďalšími riešiteľmi prác na systéme L-NeRVEn.
5. Implementované riešenie overiť vhodnými testami a na relevantnom VR hardvéri.
6. Vypracovať dokumentáciu podľa pokynov vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: **slovenský**

Termín pre odovzdanie práce: **29.05.2026**

Dátum zadania bakalárskej práce: **31.10.2025**



prof. Ing. Líberios Vokorokos, PhD.

dekan fakulty

Čestné vyhlásenie

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

Podakovanie

Na tomto mieste by som sa rád poďakoval svojmu vedúcemu práce doc. Ing. Štefanovi Korečkovi, PhD. za jeho čas a odborné vedenie počas riešenia mojej záverečnej práce. Rovnako by som sa rád poďakoval svojim rodičom a priateľom za ich podporu a povzbudzovanie počas celého môjho štúdia.

Obsah

1	Úvod	1
2	Stav problematiky	3
2.1	Cievna mozgová príhoda	3
2.2	Klinické prejavy a diagnostika CMP	3
2.3	Virtuálna realita	4
2.4	Neurorehabilitácia vo virtuálnej realite	5
2.4.1	Motorická neurorehabilitácia	5
2.4.2	Senzorická neurorehabilitácia	5
2.4.3	Kognitívna a sociálna neurorehabilitácia	6
2.4.4	Emocionálna neurorehabilitácia	6
2.5	Analýza existujúcich riešení využitia VR v medicíne	8
2.5.1	INREX-VR systém pre neurorehabilitáciu	8
2.5.2	AdaptRehab VR systém pre rehabilitáciu po mozgovej prí- hode	9
2.5.3	Virtuálna realita pre rehabilitáciu po CMP	10
2.5.4	Virtuálny medicínsky systém založený na hybridnej VR- -AR technológii	10
2.6	Záver kapitoly	10
3	Analýza technológií a sieťových riešení	12
3.1	Unity	12
3.2	Multiplayer	12
3.2.1	Kritériá výberu sieťového riešenia	12
3.2.2	Mirror Networking	13
3.2.3	FishNet Networking	14
3.2.4	Porovnanie Mirror a FishNet	16
3.3	Záver analýzy	16

4	Návrh systému	18
4.1	Konceptuálny model systému	18
4.1.1	Hlavné komponenty systému	19
4.1.2	Princíp fungovania systému	20
4.1.3	Význam modelu pre systém	20
5	Prototyp	21
5.1	Požiadavky na prototyp	21
5.1.1	Podpora VR	21
5.1.2	Možnosť kolaborácie (spolupráce viacerých používateľov)	21
5.1.3	Podpora rozdielnej funkcionality pre jednotlivé roly	22
5.1.4	Prepojenie s rozhraním OpenViBE	23
5.1.5	Flexibilita cvičení	23
5.2	Prvý prototyp	24
5.2.1	Testovacie prostredie	24
5.2.2	Vizuálna ukážka	25
6	Implementácia	26
6.1	Stavový stroj	27
6.1.1	Stavový stroj u klientov	28
6.2	Správa viacerých herných miestností	29
6.2.1	Dátový model miestnosti	29
6.2.2	Mapovanie klientov na miestnosti	29
6.2.3	Spracovanie požiadaviek klientov	29
6.2.4	Izolácia scén	30
6.2.5	Správa životného cyklu miestnosti	30
6.2.6	Synchronizačné mechanizmy FishNetu	30
6.2.7	Validácia prítomnosti rolí	32
6.2.8	Separácia serverového a klientského buildu	32
6.2.9	UI navigácia	33
7	Vyhodnotenie	35
8	Záver	37
	Zoznam použitej literatúry	38
	Zoznam príloh	43

Zoznam obrázkov

2.1	Komponenty systému INREX-VR	9
4.1	Petriho sieť	19
5.1	Prvý prototyp implementácie	25
6.1	Architektúra systému prototypu	27
6.2	Referencie priradené cez Unity Inspector	33
6.3	UI navigácia	34

Zoznam tabuliek

3.1	Porovnanie sieťových frameworkov Mirror a FishNet	16
5.1	Konfigurácia testovacieho prostredia	25
7.1	Výsledky testovania prototypu	36

1 Úvod

Cievna mozgová príhoda (CMP) patrí medzi najčastejšie ochorenia nervového systému, ktoré môžu mať závažné následky na motorické, kognitívne aj zmyslové funkcie človeka. Po prekonaní CMP často dochádza k poškodeniu nervových dráh, čo negatívne ovplyvňuje schopnosť pohybu, reči a vykonávania každodenných činností pacienta.

Jednou z možností zlepšenia funkčného stavu pacientov po CMP je neurorehabilitácia. Pri jej aplikácii s využitím virtuálnej reality (VR) boli v posledných rokoch zaznamenané viaceré pozitívne výsledky. Virtuálna realita dokáže podporiť motorické učenie, motivovať pacientov a prispieť k obnove poškodených nervových spojení [1].

Preto je dôležité vytvoriť vhodné virtuálne prostredie, ktoré pacientom poskytne efektívnu a zároveň príjemnú formu rehabilitácie prispôsobenú ich individuálnym potrebám. Práca bola riešená v rámci projektu APVV-21-0105 s názvom „Dôveryhodná interakcia človek–robot a terapeut–pacient vo virtuálnej realite“.

Formulácia úlohy

Hlavným zámerom tejto práce je vytvoriť nový prototyp aplikácie L-NeRVEn, inšpirovaný predchádzajúcou verziou systému, ktorý prostredníctvom technológie virtuálnej reality umožní podporiť a prispôbiť proces neurorehabilitácie pacientov s poruchou hybnosti horných končatín, napríklad po cievnej mozgovej príhode. Bližší popis architektúry predchádzajúcej verzie L-NeRVEn je uvedený v práci [2]. Cieľom je prispieť k efektívnejšej obnove čiastočnej alebo úplnej pohyblivosti postihnutých končatín.

Virtuálna neurorehabilitácia je zameraná na obnovu poškodených nervových funkcií a motorických schopností prostredníctvom stimulácie mozgu pomocou interaktívnych virtuálnych podnetov, ktoré podporujú neuroplasticitu mozgu.

Úlohou pacienta je vykonávať alebo si predstavovať pohyb postihnutej končatiny, pričom jeho priebeh je následne vizualizovaný vo virtuálnom prostredí.

Za prípravu, nastavenie a prispôsobenie terapeutických cvičení realizovaných prostredníctvom VR zodpovedá terapeut. Cvičenia musia byť navrhnuté v súlade s individuálnym zdravotným stavom a funkčnými schopnosťami konkrétneho pacienta, a to s ohľadom na ich náročnosť aj časové trvanie.

Aplikácia má byť navrhnutá s dôrazom na jednoduchosť ovládania zo strany pacienta a na minimalizáciu potreby využívania externých ovládacích prostriedkov. Zároveň je nutné, aby navrhnutá aplikácia mohla v budúcich implementáciách poskytnúť variabilitu úloh a virtuálnych prostredí s cieľom predchádzať strate motivácie pacienta pri opakovanom absolvovaní cvičení.

Kontakt medzi pacientom a terapeutom bude realizovaný prostredníctvom pripojenia do spoločnej terapeutickej miestnosti vo virtuálnej realite, pacient pomocou VR headsetu a terapeut pomocou desktopovej aplikácie.

Dôležitým aspektom práce bude aj zameranie sa na technologické aspekty sieťovej komunikácie, konkrétne na analýzu ich výhod a nevýhod v prostredí zdieľanej virtuálnej reality.

2 Stav problematiky

2.1 Cievna mozgová príhoda

Strata schopnosti ovládať hornú končatinu môže mať viacero príčin, pričom jednou z najčastejších je cievna mozgová príhoda (CMP). Toto ochorenie predstavuje poškodenie mozgového tkaniva spôsobené náhlou poruchou prekrvenia mozgu, ktorá vedie k nedostatku kyslíka a živín a následne k odumieraniu nervových buniek.

Z klinického hľadiska sa CMP delí na ischemickú, spôsobenú uzáverom mozgovej artérie, a hemoragickú, spôsobenú ruptúrou cievy a krvácaním do mozgu. CMP je druhou najčastejšou príčinou úmrtnosti a treťou najčastejšou príčinou kombinovanej úmrtnosti a invalidity, preto zostáva závažným problémom verejného zdravia [3]. Chronické ochorenia, medzi ktoré patrí aj CMP, navyše prispievajú k zníženej kvalite života [4] a k rozvoju depresívnych stavov [5]. Okrem zdravotných dôsledkov má CMP aj významný ekonomický dopad, pričom odhadované globálne náklady spojené s týmto ochorením presahujú 721 miliárd amerických dolárov [3].

Ischemická CMP tvorí približne 76 % všetkých prípadov, zatiaľ čo hemoragická forma postihuje približne jednu štvrtinu pacientov. V roku 2019 žilo celosvetovo približne 101,5 milióna ľudí po prekonaní CMP [6].

2.2 Klinické prejavy a diagnostika CMP

Dôsledky CMP bývajú často závažné a výrazne ovplyvňujú kvalitu života pacienta. Poškodenie mozgového tkaniva vedie k poruchám motorických, senzorických, kognitívnych aj emočných funkcií. Medzi najčastejšie motorické prejavy patrí znížená pohyblivosť jednej polovice tela (hemiparéza) alebo úplná strata pohybu polovice tela (hemiplégia).

Senzorické poruchy sa môžu prejaviť ako znížená citlivosť, porucha vnímania polohy tela alebo bolesť centrálného pôvodu. Kognitívne zmeny zahŕňajú

problémy s pamäťou, pozornosťou, orientáciou a exekutívnymi funkciami, teda schopnosťami súvisiacimi s plánovaním a organizáciou činností. Tieto poruchy významne sťažujú návrat pacienta k bežným denným aktivitám [7].

Diagnostika CMP je založená na kombinácii klinického obrazu a zobrazovacích metód. Kľúčovú úlohu zohrávajú počítačová tomografia (CT) a magnetická rezonancia (MRI), ktoré umožňujú rozlíšiť ischemickú a hemoragickú formu CMP a určiť rozsah poškodenia mozgového tkaniva. Doplnkové vyšetrenia, ako napríklad ultrasonografia karotíd, echokardiografia alebo angiografia, pomáhajú identifikovať zdroj embólie alebo ďalšie rizikové faktory [8].

Cieľom rehabilitácie je zmierniť alebo úplne odstrániť následky spôsobené CMP, podporiť neuroplasticitu mozgu a maximalizovať funkčnú nezávislosť pacienta. Včasná a intenzívna rehabilitácia, zahŕňajúca fyzioterapiu, logopédiu, ergoterapiu a psychologickú podporu, predstavuje významný predpoklad úspešnej obnovy stratených funkcií a zlepšenia kvality života pacienta.

2.3 Virtuálna realita

Virtuálna realita (VR) poskytuje používateľom simuláciu reálneho alebo umelo vytvoreného prostredia, s ktorým môžu interagovať pomocou špecializovaných zariadení, najmä VR headsetov a pohybových ovládačov.

Rozlišujeme viacero typov virtuálnej reality, pričom medzi najvýznamnejšie patria:

- neimerzívna virtuálna realita,
- imerzívna virtuálna realita.

Pri neimerzívnej VR používateľ nie je plne ponorený do virtuálneho prostredia, ale sleduje ho prostredníctvom zobrazovacieho zariadenia, napríklad počítačového monitora, tabletu alebo mobilného telefónu [9]. Interakcia so svetom VR je v tomto prípade obmedzená na štandardné vstupné zariadenia, ako sú myš, klávesnica alebo dotyková obrazovka.

Naopak, imerzívna VR umožňuje používateľovi úplné ponorenie do virtuálneho prostredia prostredníctvom zariadení typu head-mounted display (HMD). Pohybové senzory a ovládače zároveň poskytujú prirodzenú interakciu so simulovaným prostredím [1].

Virtuálna realita nachádza uplatnenie vo viacerých oblastiach, napríklad v zábavnom priemysle, vzdelávaní, medicíne, architektúre alebo vojenských simuláciách [10].

2.4 Neurorehabilitácia vo virtuálnej realite

Jednou z oblastí využitia VR je neurorehabilitácia pacientov s neurologickým postihnutím, pričom poskytuje bezpečné, kontrolované a motivujúce prostredie na tréning motorických, senzorických, kognitívnych aj emocionálnych funkcií poškodených neurologickým ochorením alebo úrazom [1].

2.4.1 Motorická neurorehabilitácia

Motorická rehabilitácia patrí medzi najvýznamnejšie oblasti využitia VR. Pacienti môžu prostredníctvom virtuálnych úloh zvyšovať aktívny rozsah pohybu horných a dolných končatín, koordináciu, rovnováhu, chôdzu i jemnú motoriku. Celkovo VR ponúka bezpečné a riadené prostredie, v ktorom je možné realizovať individuálne prispôbené rehabilitačné úlohy zamerané na rozvoj motorického učenia. Možnosť hravej a interaktívnej podoby VR, napr. v podobe videohier, zároveň zvyšuje záujem a motiváciu pacientov, predovšetkým detí, zapájať sa do terapie [11]. Ďalšou výhodou interakcie s virtuálnym prostredím je intenzívne zapájanie viacerých zmyslových systémov, ako sú zrak, rovnováha a propiocepčia v rovnakom čase, čím sa podporuje koordinácia a motorické učenie. Na hlboké ponorenie do VR sa využívajú headsety, ktoré sú ľahko využiteľné aj v domácom prostredí [12]. Bolo preukázané, že VR sa podieľa na zlepšení aktívneho rozsahu pohybu prostredníctvom nárastu objemu sivej hmoty vo viacerých oblastiach mozgu, vrátane hipokampu, bazálnych ganglií, cingulárnej kôry, senzorio-motorických oblastí a zrakovej kôry [13]. Po tréningu prostredníctvom VR sa zaznamenala aj zvýšená mozgová aktivita, a to predovšetkým vo frontálnych oblastiach mozgu [14].

Popri liečebnom význame VR je možné využiť aj jej kombináciu s počítačovou analýzou pohybu, čo umožňuje presnejšie sledovanie a hodnotenie motorických porúch po poškodení mozgu a objektívne monitorovanie rehabilitačného pokroku [15].

2.4.2 Senzorická neurorehabilitácia

VR sa v posledných rokoch stáva dôležitým nástrojom aj pri rehabilitácii senzorických porúch po poškodení mozgu či miechy. Tieto poruchy môžu zahŕňať zníženú citlivosť, problémy s vnímaním dotyku, bolesť, poruchy vnímania polohy tela alebo ťažkosti so spracovaním zmyslových informácií.

Zaujímavým využitím je aplikácia imerzívnej VR pri riešení ťažko liečiteľných

fantómových bolestí po amputácii končatiny v dôsledku úrazov. Jednou z overených metód liečby je zrkadlová terapia, pri ktorej pacient sleduje pohyby zdravej končatiny a mozog ich interpretuje ako pohyby amputovanej končatiny. Bola vytvorená VR verzia tejto metódy, ktorá dokáže výraznejšie aktivovať senzomotorické oblasti mozgu než klasická metóda [16]. Môže byť využitá samostatne alebo v kombinácii s liekmi proti bolesti. Tým sa zvyšuje potenciál VR zlepšovať a rozširovať tradičné postupy liečby bolesti.

2.4.3 Kognitívna a sociálna neurorehabilitácia

VR sa využíva aj pri rehabilitácii narušených kognitívnych funkcií, ako sú pozornosť, pamäť, plánovanie, logické myslenie, riešenie problémov a exekutívne funkcie, ktoré sú pri poškodení mozgu často oslabené. Rovnako dokáže podporiť aj sociálnu integráciu pacientov, napríklad komunikačné schopnosti, sociálne interakcie, spoluprácu a sebavedomie. Hlavným cieľom ich zlepšenia je dosiahnutie lepšej samostatnosti pacientov, ich sociálnej reintegrácie a zlepšenia uplatnenia v každodennom aj pracovnom živote. Hoci v jednej nedávnej štúdii nebola preukázaná významná výhoda VR oproti konvenčnej rehabilitácii vo zvyšovaní kognitívnych a sociálnych zručností [17], VR dokáže vytvárať realistické simulácie každodenných situácií, napríklad nakupovanie vo virtuálnom obchode, používanie bankomatu, príprava jedla vo virtuálnej kuchyni alebo orientácia v priestore, pričom veľkou výhodou VR je bezpečné prostredie, v ktorom môže pacient robiť chyby bez reálnych následkov [18] [19]. Takéto aktivity následne pomáhajú pacientom zlepšovať rôzne praktické schopnosti potrebné pre samostatný život. Štúdie poukazujú na zmeny neurálnej plasticity, zlepšenie pracovnej pamäte, selektívnej pozornosti a kognitívnej flexibility po pravidelnom VR tréningu [20] [21]. U pacientov absolvujúcich tréning kognitívnych schopností vo virtuálnej realite boli napríklad pozorované zlepšenia funkcií súvisiacich s čelným lalokom mozgu, napríklad pri vykonávaní každodenných činností, ako je príprava kávy vo virtuálnom aj reálnom prostredí [1].

2.4.4 Emocionálna neurorehabilitácia

Poškodenie centrálného nervového systému často spôsobuje popri iných poruchách rôzne psychické problémy, ako sú úzkosť, depresia, či nestabilita emócií. Bolo dokázané, že emocionálna rehabilitácia pomáha zvýšiť liečebnú účinnosť používanej farmakoterapie (napr. antidepresív) [22]. VR využíva princíp environmentálneho obohatenia, ktorý podporuje neuroplasticitu a neurogenézu v hi-

pokampe – oblasti mozgu dôležitej pre reguláciu emócií a stresu [23]. Imerzívna virtuálna realita navyše spája zážitkové prostredie s pohybovou aktivitou. Je známe, že fyzické cvičenie priaznivo ovplyvňuje psychické aj sociálne zdravie a podporuje procesy v nervovom systéme, ktoré zlepšujú pozornosť, pamäť, sústredenie a celkovú psychickú pohodu. Tieto priaznivé štrukturálne a neurochemické účinky sú vysvetlené zvýšením hladiny dôležitých neurotransmiterov, najmä katecholamínov [24].

Je však potrebné mať na pamäti, že jednoznačný význam týchto nových metód musí byť ešte overený v kvalitných multicentrických klinických štúdiách so štandardizovanými protokolmi a dlhodobým sledovaním pacientov, ako aj následne v metaanalýzach [25] [26].

Navrhované riešenie bakalárskej práce je zamerané na pacientov s poruchou hybnosti hornej končatiny spôsobenou CMP. Hlavným prínosom by malo byť zlepšenie motorických funkcií tejto dôležitej časti tela, čo sa premietne do lepšej schopnosti pacientov vykonávať každodenné činnosti. To v konečnom dôsledku môže významne zvýšiť ich kvalitu života. Pre úspešný priebeh rehabilitácie je potrebné, aby boli pacient aj terapeut pripojení do spoločného virtuálneho prostredia, v ktorom môže terapeut dohliadať na vykonávanie rehabilitačných cvičení a priebežne ich upravovať podľa potrieb pacienta.

Iná metaanalýza potvrdila, že VR je sľubnou doplnkovou metódou rehabilitácie hornej končatiny po cievnej mozgovej príhode [27]. VR preukázala lepšie výsledky ako tradičná terapia pri zlepšovaní motoriky hornej končatiny, samostatnosti v každodenných činnostiach, kvality života, spasticity aj jemnej motoriky. Plne imerzívne systémy prinášali významnejšie zlepšenia hrubej motoriky, zatiaľ čo neimerzívne formy mali väčší prínos pre jemnú zručnosť. Najlepšie výsledky sa dosahovali pri terapii trvajúcej viac ako šesť týždňov a pri začatí rehabilitácie pomocou VR do šiestich mesiacov po cievnej mozgovej príhode. Význam imerzívnej VR dokumentovala aj ďalšia vedecká práca [28]. Nedávna metaanalýza hodnotiaca 41 štúdií preukázala uskutočniteľnosť a bezpečnosť používania takýchto metód, keďže nežiaduce udalosti boli zriedkavé [26]. Vhodným doplnkom môžu byť aj domáce VR systémy, pričom interaktívne a herné prvky VR zvyšujú motiváciu pacientov pokračovať v rehabilitácii [29].

2.5 Analýza existujúcich riešení využitia VR v medicíne

Nasledujúce riešenia boli vybrané na podrobnejšiu analýzu z hľadiska ich podobnosti so zadaním tejto práce.

2.5.1 INREX-VR systém pre neurorehabilitáciu

Štúdia [30] opisuje systém INREX-VR (INtelligent telerehabilitation EXergame system based on Virtual Reality), ktorý je určený na neurorehabilitáciu pacientov s neurologickými ochoreniami a poruchami pohybu. Cieľom systému je podporiť rehabilitáciu horných a dolných končatín prostredníctvom virtuálnej reality a interaktívnych cvičení.

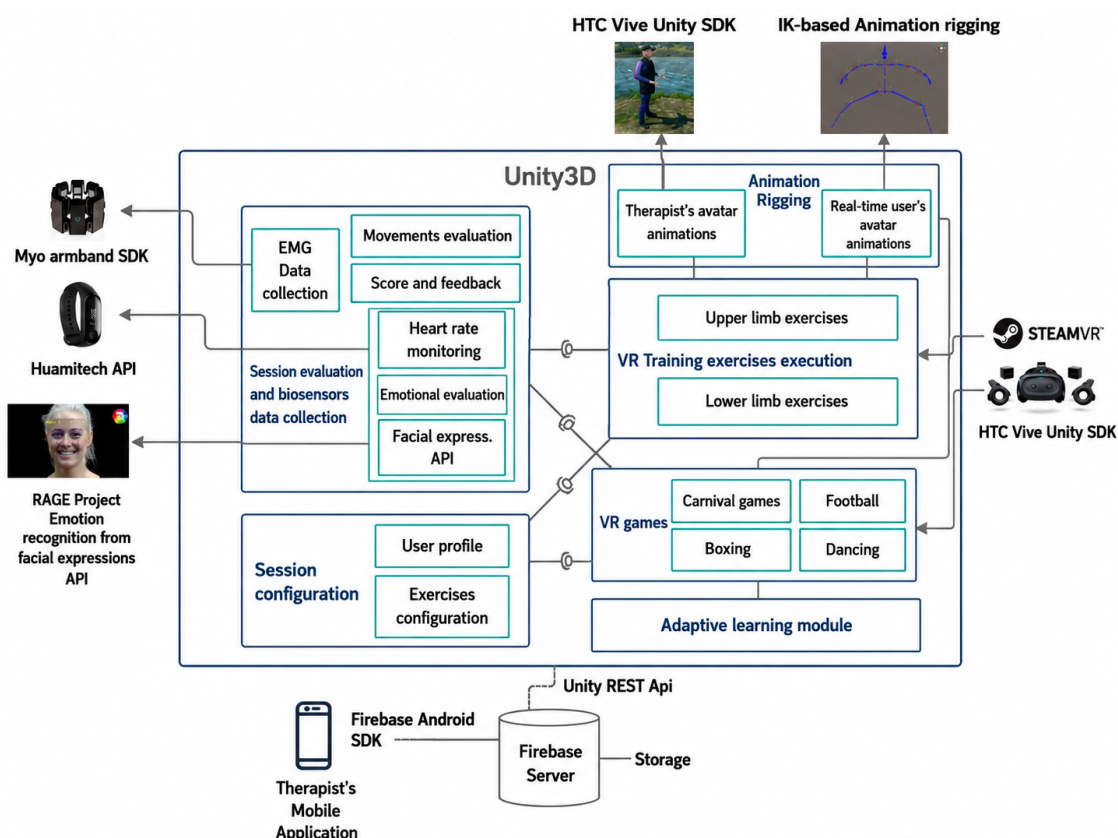
Systém využíva technológiu VR spolu s headsetom HTC Vive. Súčasťou riešenia je aj cloudová infraštruktúra, ktorá zabezpečuje uchovávanie rehabilitačných údajov a umožňuje vzdialený prístup terapeutov k výsledkom pacientov.

V rámci systému boli implementované viaceré rehabilitačné scenáre zamerané na podporu motorických schopností pacienta predovšetkým zamerané na rehabilitáciu horných končatín a rehabilitáciu dolných končatín.

Architektúra systému pozostáva z viacerých častí:

- VR aplikácia pre pacienta,
- databázový systém na uchovávanie údajov,
- mobilná aplikácia pre terapeutov.

Podľa článku bol systém navrhnutý s dôrazom na modularitu, čo umožňuje rozšírenie systému a tak isto vedie k zvýšeniu motiváciu pacientov prostredníctvom interaktívneho virtuálneho prostredia. Systém umožňuje vzdialené monitorovania pacientov, z čoho vyplýva, že pacienti majú možnosť využiť systém aj v domácom prostredí. Systém bol navrhnutý vo vývojovom prostredí Unity. Systém INREX-VR sa skláda z viacerých komponentov systému uvedených na obrázku 2.1.



Obrázok 2.1: Komponenty systému INREX-VR [30]

2.5.2 AdaptRehab VR systém pre rehabilitáciu po mozgovej príhode

Nedávna štúdia [31] opisuje systém AdaptRehab VR určený na rehabilitáciu horných končatín pacientov po CMP. Cieľom riešenia bolo vytvoriť prispôsobiteľnú imerzívnu VR aplikáciu vhodnú najmä pre krajiny s obmedzeným prístupom k rehabilitačnej starostlivosti.

Riešenie obsahuje viacero rehabilitačných hier zameraných na rozvoj motorických schopností a koordinácie pohybov horných končatín. Pre toto riešenie bolo vytvorených šesť rehabilitačných hier napr. Basket Bloom, Strike Zone, TapQuest.

Systém sa upravuje na základe spätnej väzby, údaje o výkone pacienta sa zaznamenávajú a následne sa automaticky upravuje obtiažnosť jednotlivých cvičení. Súčasťou návrhu bolo aj zapojenie fyzioterapeutov a pacientov do procesu vývoja s cieľom zvýšiť použiteľnosť systému v klinickej praxi. Zo štúdie vyplýva, že virtuálna realita podporuje motiváciu pacienta prostredníctvom interaktívnych herných prvkov.

Systém bol implementovaný pomocou herného engine Unity a využíva VR headsety Oculus Quest a podporu hand-trackingu.

2.5.3 Virtuálna realita pre rehabilitáciu po CMP

Existuje štúdia, ktorá poukazuje na využitie VR v rehabilitácii prostredníctvom hier vytvorených v prostredí Unity určených pre pacientov s neurologickými poruchami [32]. Pre úspešnú simuláciu sú potrebné VR okuliare Oculus Quest 2, ktoré umožňujú pacientovi plné ponorenie do virtuálneho prostredia.

V rámci riešenia bola vyvinutá aj mobilná aplikácia „Recover Me“, ktorá umožňuje komunikáciu medzi pacientom a terapeutom, ako aj uchovávanie záznamov jednotlivých pacientov. V štúdiu bolo hodnotených niekoľko hier využiteľných pri rehabilitácii pacientov po CMP: Connect, Drag and Drop, Memory Game.

Zo štúdie vyplýva, že využitie VR rehabilitácie malo pozitívny vplyv na fyzické a kognitívne schopnosti pacientov s neurologickými poruchami. Výsledky regresnej analýzy ukázali zlepšenie približne u 60 % testovaných pacientov. Autori zároveň uvádzajú pozitívnu spätnú väzbu pacientov na využitie herných mechanizmov a VR počas rehabilitácie.

2.5.4 Virtuálny medicínsky systém založený na hybridnej VR-AR technológii

Nedávna vedecká práca [33] sa zaoberá návrhom a vývojom virtuálneho medicínskeho systému integrujúceho technológie VR a rozšírenej reality (AR). Cieľom systému bolo vytvoriť prostredie vhodné pre medicínske simulácie a vzdelávanie.

Systém využíva hybridný prístup kombinujúci imerzívne prostredie VR a interakčné prvky AR. Pri návrhu systému bol kladený dôraz na používateľské rozhranie a vysokú úroveň realizmu s cieľom zlepšiť používateľský zážitok a podporiť medicínske simulácie.

Autori opisujú viaceré časti systému vrátane správy 3D modelov, renderovania virtuálneho prostredia a používateľskej interakcie. Štúdia naznačuje, že hybridné VR-AR riešenia môžu zlepšiť kvalitu medicínskeho vzdelávania a tréningu, zároveň však upozorňuje na technické výzvy súvisiace s výkonom a synchronizáciou systému.

2.6 Záver kapitoly

Na základe vykonanej analýzy existujúcich riešení možno potvrdiť, že virtuálna realita predstavuje perspektívny nástroj v oblasti neurorehabilitácie pacientov

s neurologickými poruchami a pacientov po cievnej mozgovej príhode. Analyzované systémy poukazujú na možnosti využitia VR pri rehabilitácii horných a dolných končatín, rozvoji motorických schopností, koordinácie pohybov a podpore kognitívnych funkcií pacienta.

Významným prvkom analyzovaných riešení je využitie herných mechanizmov a interaktívnych rehabilitačných scenárov, ktoré dokážu zvýšiť motiváciu pacienta počas terapie. Viaceré riešenia zároveň umožňujú personalizáciu rehabilitačných cvičení, zaznamenávanie údajov o priebehu rehabilitácie a vzdialené monitorovanie pacientov prostredníctvom mobilných alebo cloudových systémov.

Analýza taktiež ukázala, že pri vývoji VR rehabilitačných aplikácií sa často využíva herný engine Unity a moderné VR headsety, napríklad Quest alebo HTC Vive. Výhodou týchto technológií je podpora interaktívneho prostredia, sledovania pohybov používateľa a možnosť implementácie rehabilitačných aktivít vo virtuálnej realite.

Získané informácie z analýzy predstavujú vhodný základ pre návrh nového prototypu zameraného na neurorehabilitáciu horných končatín vo virtuálnej realite s využitím herných mechanizmov, interakcie medzi pacientom a terapeutom, modularitou a rozširiteľnosťou v budúcnosti.

3 Analýza technológií a sieťových riešení

3.1 Unity

Unity je multiplatformový herný engine a vývojové prostredie, ktoré umožňuje tvorbu 2D, 3D, VR a AR aplikácií. Svoje postavenie na trhu si získal vďaka prístupnému rozhraniu, rozsiahlej dokumentácii a bohatému obchodu s doplnkami (Asset Store), ktorý výrazne urýchľuje vývojový proces. Unity využíva jazyk C# a poskytuje komplexnú sadu nástrojov na tvorbu grafického rozhrania, fyzikálnych simulácií, animácií a sieťovej komunikácie. Vďaka podpore pre viac ako 25 platforiem vrátane Windows, macOS, iOS, Android, PlayStation, Xbox a rôznych VR zariadení je Unity vhodným riešením pre multiplatformové projekty [34].

3.2 Multiplayer

Multiplayer architektúra je veľmi dôležitá pri kolaboratívnych rehabilitačných aplikáciách. Je potrebné zabezpečiť správnu sieťovú synchronizáciu pre všetkých používateľov systému.

3.2.1 Kritériá výberu sieťového riešenia

Pri výbere sieťového frameworku boli hodnotené najmä nasledujúce kritériá: latencia synchronizácie, podpora VR aplikácií, dlhodobá stabilita projektu, podpora dedikovaného servera a možnosti budúceho rozšírenia aplikácie.

Pri VR aplikáciách je dôležitá najmä vysoká frekvencia synchronizácie objektov, keďže pohyb hlavy, rúk a ovládačov používateľa musí byť prenášaný s minimálnym oneskorením, aby sa zachoval pocit prirodzenej interakcie a stabilný používateľský zážitok.

3.2.2 Mirror Networking

Mirror je sieťové riešenie pre Unity, ktoré je založené na pôvodnom Unity UNET. Po oficiálnom ukončení podpory UNET v roku 2018 a jeho následnom odstránení z novších verzií Unity vznikol Mirror ako open-source náhrada, ktorá si zachováva podobné API a funkcionality, pričom zároveň prináša viaceré vylepšenia v oblasti stability a podpory moderných sieťových scenárov.

Vlastnosti Mirror

Mirror je open-source riešenie dostupné pod licenciou MIT, ktoré umožňuje úplnú kontrolu nad zdrojovým kódom a možnosť voľnej integrácie do projektu. Mirror podporuje architektúru dedikovaného servera, ktorá je dôležitá pre väčšinu multiplayerových aplikácií. Väčšina projektov vytvorených pomocou UNET môže byť prenesená do Mirror s minimálnymi úpravami, keďže Mirror zachováva podobné API a podporuje väčšinu nástrojov a assetov pôvodne vytvorených pre UNET.

Architektúra

Mirror využíva architektúru založenú na udalostiach s podporou RPC (Remote Procedure Calls), ako sú `Command` a `ClientRpc`.

Medzi hlavné súčasti frameworku patria:

- **NetworkManager** – centrálny manažér pre sieťové operácie,
- **NetworkIdentity** – identifikácia sieťových objektov,
- **NetworkBehaviour** – základná trieda pre sieťovú logiku,
- **SyncVars** – automatická synchronizácia premenných medzi serverom a klientmi.

Mirror aj FishNet využívajú oddelenú transportnú vrstvu, čo umožňuje použiť rôzne transportné protokoly podľa požiadaviek aplikácie.

Vyhodnotenie

Mirror poskytuje jednoduchý prechod pre vývojárov používajúcich UNET. Zároveň má rozsiahlu komunitu, čo poskytuje vývojárom veľké množstvo dostupných tutoriálov, ktoré pomáhajú riešiť rôzne implementačné problémy.

Na druhej strane Mirror kladie menší dôraz na pokročilé optimalizačné techniky. Pri náročných synchronizačných scenároch môže byť preto potrebné implementovať vlastné optimalizačné mechanizmy. Mirror poskytuje menej pokročilé mechanizmy synchronizácie najmä v oblasti client-side prediction (predikcia na strane klienta) a tick-based synchronizácie oproti FishNet.

Zdroje a dokumentácia

Mirror má aktívnu komunitu a rozsiahlu dokumentáciu popísanú vo viacerých zdrojoch [35], [36].

3.2.3 FishNet Networking

FishNet je moderná sieťová open-source knižnica pre Unity vytvorená ako alternatíva k starším riešeniam ako UNET alebo Mirror. Na rozdiel od Mirroru ide o samostatne navrhnutú architektúru orientovanú na výkon, nízku latenciu a rozširiteľnosť.

Vlastnosti FishNet

FishNet sa pri návrhu zameriava aj na obmedzenie zbytočných alokácií pamäte a efektívnejšiu synchronizáciu objektov. Pri VR aplikáciách je dôležité udržať stabilnú snímkovú frekvenciu a zároveň presne synchronizovať pohyby používateľa v reálnom čase.

Tento systém poskytuje vstavanú podporu pre klientskú predikciu pohybu, plynulú sieťovú synchronizáciu, možnosť implementovať dedikovaný server a automatizovanú správu sieťových objektov a scén.

Ponúka tiež modernejší prístup k synchronizácii objektov a je navrhnutý s dôrazom na efektívnejšiu správu sieťových aktualizácií, čo môže viesť k stabilnejšiemu výkonu pri väčšom počte synchronizovaných objektov alebo pri častom prenose dát. Tieto vlastnosti sú dôležité najmä pri VR aplikáciách, kde sa synchronizujú pozície rúk, ovládačov a interaktívnych objektov s vysokou frekvenciou aktualizácií.

Už spomínaný dedikovaný server je výhodný aj z pohľadu konzistencie rehabilitačných dát, keďže server riadi výsledný stav aplikácie a synchronizované dáta klientov.

Nevýhodou FishNetu je menšia komunita a menšie množstvo dostupných tutoriálov. Vývojár tak musí vo väčšej miere pracovať s technickou dokumentáciou a detailnejšie rozumieť internému fungovaniu systému.

Architektúra

FishNet využíva architektúru založenú na udalostiach s podporou RPC (Remote Procedure Calls), ako sú ServerRpc, ObserversRpc a TargetRpc.

Medzi hlavné súčasti frameworku patria:

- **NetworkManager** – centrálny manažér pre sieťové operácie,
- **NetworkObject** – identifikácia a správa sieťových objektov,
- **NetworkBehaviour** – základná trieda pre implementáciu sieťovej logiky,
- **SyncTypes** – mechanizmy na automatickú synchronizáciu dát medzi serverom a klientmi,
- **PredictionManager** – systém pre client-side prediction,
- **TransportManager** – správa transportnej vrstvy a sieťovej komunikácie.

Vyhodnotenie

FishNet predstavuje moderné sieťové riešenie orientované na výkon, nízku latenciu a pokročilé synchronizačné mechanizmy. V porovnaní s frameworkom Mirror poskytuje vstavanú podporu pre client-side prediction a tick-based synchronizáciu, čo umožňuje presnejšiu a plynulejšiu synchronizáciu objektov pri aplikáciách s vysokými nárokmi na odozvu.

Významnou výhodou FishNetu je dôraz na efektívnu správu sieťových aktualizácií a minimalizáciu alokácií pamäte, čo môže prispieť k stabilnejšiemu výkonu pri väčšom počte synchronizovaných objektov alebo pri častom prenose dát. Tieto vlastnosti sú dôležité najmä pri VR aplikáciách, kde je potrebné synchronizovať pohyby používateľov a interaktívnych objektov s vysokou frekvenciou aktualizácií.

Uvedené sieťové riešenie podporuje viacero transportných vrstiev, čo umožňuje prispôbiť sieťovú komunikáciu požiadavkám aplikácie.

FishNet má však menšiu komunitu a menšie množstvo dostupných tutoriálov a vývojových nástrojov v porovnaní s Mirrorom. Vývojár musí pracovať s technickou dokumentáciou a detailnejšie rozumieť internému fungovaniu frameworku.

Zdroje a dokumentácia

Informácie boli spracované podľa dostupnej dokumentácie [37].

3.2.4 Porovnanie Mirror a FishNet

Hlavné rozdiely medzi frameworkmi Mirror a FishNet sú uvedené v tabuľke 3.1.

Technológia	Mirror	FishNet
Pôvod	Vznikol na základe frameworku UNET	Moderná architektúra navrhnutá od základov
Výkon	Vhodné pre bežné multiplayerové aplikácie	Optimalizované pre nízku latenciu
Synchronizácia objektov	Štandardná synchronizácia sieťových objektov	Pokročilé mechanizmy synchronizácie
VR	Vhodné pre menej náročné VR aplikácie	Vhodné pre VR aplikácie s vysokými nárokmi na presnosť pohybu
Rozšíriteľnosť	Vhodné pre menšie projekty	Lepšie pripravené na komplexnejšie scenáre
Komunita	Veľká komunita	Menšia komunita

Tabuľka 3.1: Porovnanie sieťových frameworkov Mirror a FishNet

3.3 Záver analýzy

Vývojové prostredie Unity bolo už vopred zvolené pri zadaní práce.

Pri výbere sieťového riešenia boli porovnávané najmä frameworky Mirror a FishNet. Mirror predstavuje stabilné a rozšírené riešenie s veľkou komunitou a jednoduchšou implementáciou. Je vhodný najmä pre menšie multiplayerové projekty alebo aplikácie, kde nie sú vysoké požiadavky na synchronizáciu v reálnom čase.

FishNet je navrhnutý s dôrazom na výkon a efektívnejšiu synchronizáciu objektov, čo môže viesť k nižšej záťaži systému pri náročnejších scenároch. Tieto vlastnosti sú významné najmä pri VR aplikáciách, kde je potrebné prenášať veľké množstvo pohybových dát s minimálnou latenciou a vysokou presnosťou.

Pre potreby neurorehabilitačnej aplikácie horných končatín sú kľúčové najmä:

- presná synchronizácia pohybov rúk a ovládačov,
- nízka latencia pri prenose dát,
- stabilná snímková frekvencia vo VR prostredí,
- možnosť budúceho rozšírenia o viac používateľov a viac cvičení.

Na základe uvedených požiadaviek bol pre implementáciu prototypu novej verzie aplikácie zvolený framework FishNet. Napriek menšej komunite poskytuje vhodnejšie technické predpoklady pre budúci vývoj VR rehabilitačnej aplikácie.

4 Návrh systému

Táto kapitola sa zaoberá návrhom systému virtuálnej reality určeného na neurorehabilitáciu. Systém bol navrhnutý s dôrazom na dlhodobý vývoj a intuitívne ovládanie pre nových členov vývojového tímu, pričom na jeho základe bude následne vytvorená finálna stabilná, prehľadná a dobre zdokumentovaná verzia aplikácie. Cieľom je zabezpečiť jednoduchú ďalšiu rozširiteľnosť projektu a zároveň urýchliť a zefektívniť zapojenie nových členov do tímu. Prototyp má zároveň slúžiť na rýchle a efektívne overenie funkčnosti zvolených technológií a na zabezpečenie ich vzájomnej kompatibility v rámci systému. Potenciálne nedostatky by mali byť v prototypovej verzii riešiteľné jednoduchšie a rýchlejšie než v doteraz existujúcej aplikácii. Na účely validácie nového konceptuálneho modelu virtuálneho prostredia bol prototyp implementovaný v prostredí Unity s cieľom overiť jeho funkčnosť a použiteľnosť.

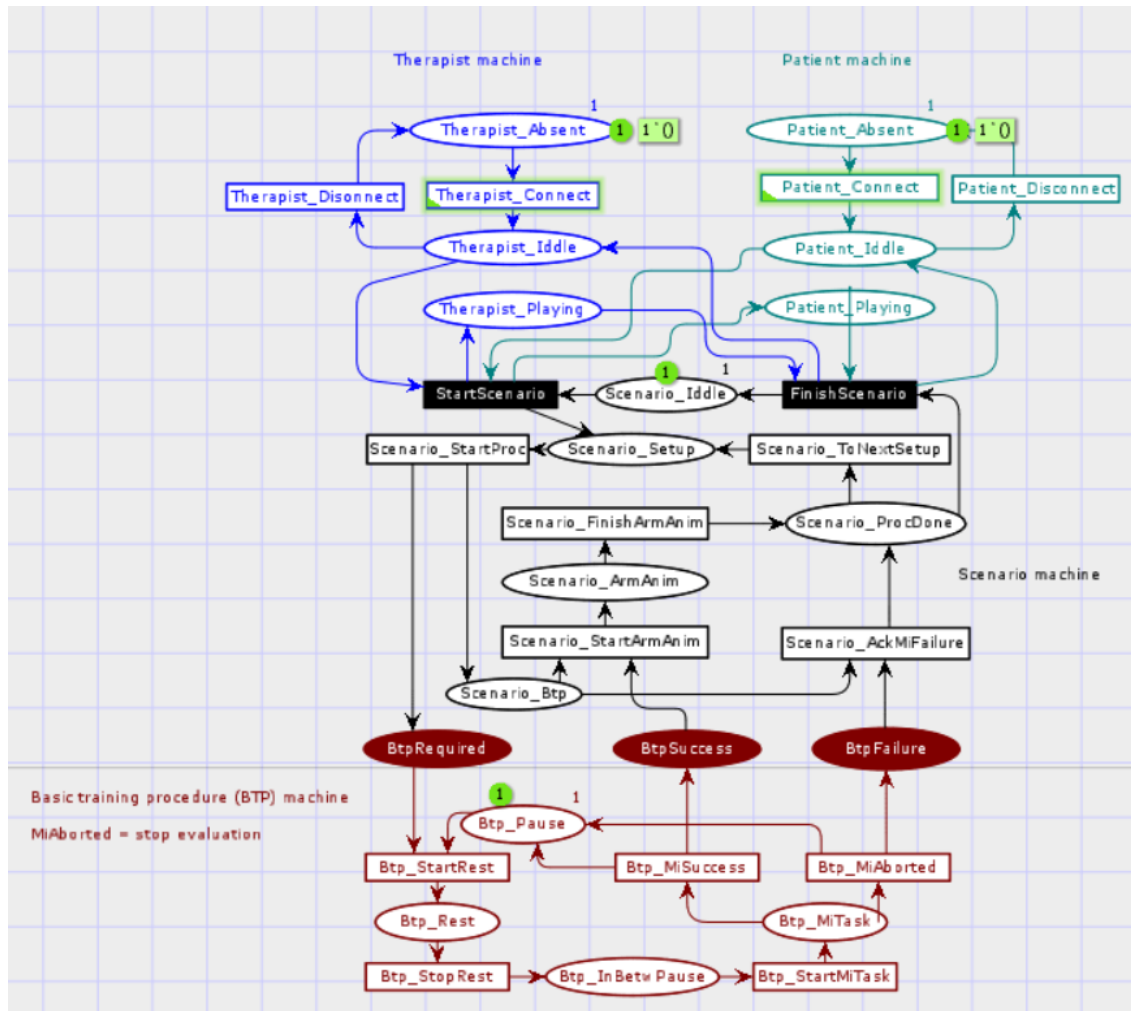
4.1 Konceptuálny model systému

Konceptuálny model systému je navrhnutý ako Petriho sieť navzájom prepojených automatov (stavových strojov). Tieto automaty riadia celý priebeh tréningovej relácie, od pripojenia používateľov až po vykonanie samotných rehabilitačných cvičení a ich vyhodnotenie. Model zaisťuje, že všetky časti systému vzájomne komunikujú a reagujú na akcie terapeuta i pacienta. Tento model bol poskytnutý vedúcim práce 4.1.

Jednotlivé stavové stroje sú v diagrame farebne odlíšené. Obdĺžniky predstavujú jednotlivé stavové prechody medzi stavmi.

V rámci tejto Petriho siete existuje niekoľko špeciálnych typov stavových prechodov. Prvý typ prechodu je vyznačený čiernym obdĺžnikom. Mechanizmus spočíva v tom, že daný stavový prechod sa vykoná iba v prípade, ak sú splnené všetky požadované stavy ostatných stavových strojov. Napríklad prechod *StartScenario* sa vykoná iba vtedy, ak sú aktuálne stavy strojov *Patient_Idle*, *Therapist_Idle* a *Scenario_Idle*.

Ďalší špeciálny typ prechodov sa nachádza pri červených ováloch. Tieto ovály reprezentujú tzv. message token. Prechod vstupujúci do daného komponentu pridáva token, zatiaľ čo výstupný prechod token odoberá. Výstupný prechod sa môže vykonať iba v prípade, že sa v komponente nachádza aspoň jeden token.



Obrázok 4.1: Petriho sieť reprezentujúca model systému (poskytnutá vedúcim práce)

4.1.1 Hlavné komponenty systému

Systém sa skladá z niekoľkých logických celkov, ktoré sú popísané v nasledujúcom texte.

Centrálny riadiaci modul herného scenára

Tento modul je jadrom systému, pričom riadi hlavný cyklus cvičenia a kontroluje kedy začať s rehabilitáciou.

Modul riadiaci pripojenie pacienta a terapeuta (klientov)

Úlohou tohto komponentu je správa aktuálneho stavu klientov v systéme vrátane ich pripájania a odpájania z virtuálnej miestnosti.

Modul Basic Training Procedure (BTP)

Modul BTP riadi samotný priebeh cvičenia mentálnej imaginácie pacienta.

4.1.2 Princíp fungovania systému

1. **Prípravná fáza:** Terapeut a pacient sa pripoja do systému.
2. **Spustenie cvičenia:** Systém overí, či sú splnené všetky podmienky potrebné na spustenie cvičenia.
3. **Požiadavka na zmenu stavu:** Terapeut pošle požiadavku na server o zmenu stavu cvičenia.
4. **Vyhodnotenie a prechod:** Server vyhodnotí požiadavku, pričom valídna požiadavka sa synchronizuje u každého klienta.

4.1.3 Význam modelu pre systém

Tento konceptuálny model slúži ako základná architektúra systému. Táto schéma zvyšuje prehľadnosť, keďže každá časť systému má jasne definovanú úlohu. Umožňuje flexibilný vývoj, pretože jednotlivé moduly je možné meniť alebo rozširovať bez ovplyvnenia ostatných častí systému. Systém dokáže reagovať na neočakávané situácie, napríklad odpojenie používateľa alebo prerušenie cvičenia.

Model predstavuje koncepčný základ, na ktorom je možné vybudovať funkčnú a používateľsky prístupnú aplikáciu zameranú na neurorehabilitáciu prostredníctvom mentálnej imaginácie.

5 Prototyp

5.1 Požiadavky na prototyp

Vo fáze prototypovania je nevyhnutné jednoznačne stanoviť ciele, ktoré chceme dosiahnuť. To nám umožní vyhodnotiť, či vybrané technológie sú vhodné pre vývoj kompletnej aplikácie. Pritom je dôležité už v tomto štádiu počítať s potenciálnymi budúcimi rozšíreniami funkcionality. Ako základné požiadavky sme stanovili nasledujúce ciele:

- podpora VR,
- možnosť interakcie viacerých používateľov,
- podpora jednotlivej funkcionality pre rozličné roly (pacient, terapeut),
- možnosť prepojenia s rozhraním OpenViBE,
- flexibilné rozširovanie cvičení.

5.1.1 Podpora VR

Systém je navrhnutý na plnú podporu VR v prostredí Unity s využívaním zariadení Meta Quest. Využíva sa OpenXR alebo Meta XR SDK pre kompatibilitu s headsetmi Meta Quest 2/3/Pro. Interakcie vo VR, ako je uchopovanie objektov, pohyb v priestore a ovládanie používateľského rozhrania sú implementované prostredníctvom XR Interaction Toolkit, ktorý predstavuje súbor nástrojov v prostredí Unity určených na tvorbu interakcií v aplikáciách virtuálnej a rozšírenej reality. Podrobnejšie informácie o tejto časti sú uvedené v práci spoluriešiteľa Tomáša Süča.

5.1.2 Možnosť kolaborácie (spolupráce viacerých používateľov)

Kolaboratívne prostredie je zabezpečené pomocou frameworku FishNet Networking, ktorý poskytuje nástroje na synchronizáciu stavu medzi viacerými klientmi

v reálnom čase. Architektúra systému podporuje multiplayer komunikáciu s dôrazom na stabilitu, nízku latenciu a konzistentný stav virtuálneho prostredia.

Dedikovaný server

Dedikovaný server predstavuje samostatné serverové zariadenie, ktoré beží nezávisle od klientskych inštancií. Tento prístup zabezpečuje stabilné a spoľahlivé sieťové prostredie, pričom server disponuje vlastnými výpočtovými zdrojmi a samostatným sieťovým pripojením.

Medzi hlavné výhody dedikovaného servera patria: konzistentný výkon nezávislý od počtu pripojených používateľov, centralizovaná správa herného stavu, lepšia kontrola synchronizácie objektov, možnosť budúceho rozšírenia systému.

Klienti sa k serveru pripájajú zo samostatných zariadení, ktoré odosielaajú vstupy používateľov a prijímajú aktualizované herné stavy zo servera. Oddelenie serverovej a klientskej logiky umožňuje lepšie škálovanie systému a spoľahlivejšiu synchronizáciu medzi používateľmi.

Synchronizácia objektov a komunikácia

Framework FishNet poskytuje viacero mechanizmov určených na synchronizáciu sieťových objektov a komunikáciu medzi klientmi: synchronizáciu polohy, rotácie a stavu objektov, volania vzdialených procedúr (Remote Procedure Call, RPC), správu sieťových spojení, synchronizáciu avatarov používateľov a spoločnú manipuláciu s objektmi vo virtuálnom prostredí.

Tieto mechanizmy umožňujú súčasnú prítomnosť viacerých používateľov v jednej VR scéne a zabezpečujú konzistentný priebeh spoločných rehabilitačných aktivít.

5.1.3 Podpora rozdielnej funkcionality pre jednotlivé roly

Systém je navrhnutý tak, aby podporoval dve základné používateľské roly:

- pacient,
- terapeut.

Každá rola disponuje odlišným používateľským rozhraním a špecifickými funkcionalitami.

Funkcionalita pre pacienta

Používateľské rozhranie pacienta je navrhnuté s dôrazom na jednoduchosť a intuitívnosť ovládania.

Pacient má k dispozícii najmä: zobrazovanie spätnej väzby, interakciu s objektmi vo virtuálnom prostredí a vizualizáciu aktuálneho stavu cvičenia v prehľadnej forme.

Funkcionalita pre terapeuta

Rozhranie terapeuta poskytuje rozšírené možnosti správy rehabilitačného procesu.

Terapeut má k dispozícii: ovládací panel na požiadavky na zmenu stavu stavového stroja, možnosť budúcej dynamickej úpravy scény a vizualizáciu aktuálneho stavu cvičenia v prehľadnej forme.

Pridelenie rolí

Pridelenie rolí je realizované prostredníctvom servera FishNet. Každá rola má vlastnú inštanciu stavového stroja, čo umožňuje škálovateľnú rozšíriteľnosť v budúcich implementáciách.

5.1.4 Prepojenie s rozhraním OpenViBE

Komunikácia s OpenViBE pre príjem EEG dát je riešená ako samostatný modul bežiaci na autoritatívnom serveri pomocou technológie WebSocket. Tento modul je zodpovedný za pripojenie k výstupnému komunikačnému protokolu Transmission Control Protocol (TCP socketu) servera OpenViBE, načítanie a transformáciu dát do vhodného formátu. Následne tieto spracované dáta WebSocket poskytuje hlavnej hernej logike servera implementovanej pomocou FishNet. Server potom pomocou spoľahlivých RPC volaní distribuuje príslušnú hernú logiku používateľom v kolaboratívnom VR prostredí. Podrobnejšie informácie o tejto časti sú uvedené v práci spoluriešiteľa Tomáša Sücha a takisto v práci Damiana Chmuru.

5.1.5 Flexibilita cvičení

Systém bol navrhnutý a implementovaný s dôrazom na modularitu a rozšíriteľnosť rehabilitačných cvičení v budúcnosti. Architektúra systému umožňuje jednoduché pridávanie nových scenárov, úpravu existujúcich mechaník a konfiguráciu jednotlivých tréningových relácií v budúcnosti.

Medzi hlavné vlastnosti systému patria:

- podpora načítavania rôznych rehabilitačných scén,
- možnosť modifikácie stavových prechodov stavového stroja,
- podpora zmien funkcionality v jednotlivých stavoch stavového stroja.

Sieťová architektúra frameworku FishNet zabezpečuje synchronizáciu všetkých funkcionalít medzi klientmi s minimálnym oneskorením. Zároveň zachováva výkonnosť potrebnú pre VR aplikácie. Štruktúra systému je navrhnutá tak, aby hlavná herná logika bežala na serveri, čím sa zabezpečuje konzistentný stav aplikácie pre všetkých používateľov.

5.2 Prvý prototyp

V rámci prvého prototypu bola implementovaná základná sieťová architektúra postavená na frameworku FishNet pre Unity, ktorá umožňuje spoluprácu dvoch typov používateľov pacienta a terapeuta. Prototyp poskytuje zdieľané 3D prostredie, v ktorom obaja účastníci v reálnom čase vidia svoje avatary aj aktuálny stav interaktívnych objektov. Funkčnosť interakcie s predmetmi je navrhnutá ako symetrická – pacient aj terapeut môžu manipulovať s definovanými objektmi v priestore (napríklad premiestňovanie guľôčky), čím sa vytvára základ pre spoločné terapeutické alebo tréningové aktivity. Synchronizácia stavov objektov prebieha prostredníctvom FishNet s využitím autoritatívneho serverového modelu, ktorý zabezpečuje konzistentný priebeh aplikácie pre všetkých klientov. Tento prototyp tak tvorí funkčný základ pre testovanie mechaník a interakcií v ďalších fázach vývoja.

5.2.1 Testovacie prostredie

Testovanie bolo realizované na dvoch klientských zariadeniach pripojených v lokálnej sieti. Aplikácia bola spustená v desktop režime s cieľom overiť základnú funkcionality synchronizácie objektov a stability spojenia.

Pri testovaní bol využitý framework FishNet spolu s komponentom NetworkTransform na synchronizáciu pohybu objektov medzi klientmi. Detaily testovania uvádza tabuľka 5.1.

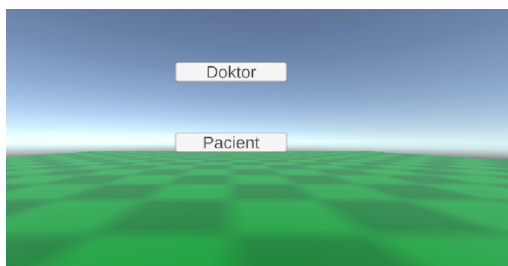
Počas testovania neboli pozorované výrazné problémy so synchronizáciou a zaznamenané oneskorenie bolo minimálne.

Počet klientov	2
Typ pripojenia	LAN
Sieťový framework	FishNet
Synchronizované objekty	Avatar, interaktívne objekty
Použitá synchronizácia	NetworkTransform

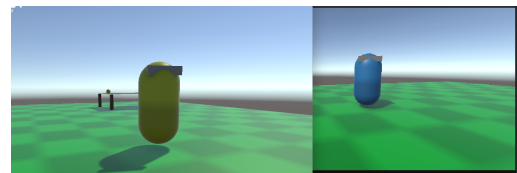
Tabuľka 5.1: Konfigurácia testovacieho prostredia

5.2.2 Vizualná ukážka

Vizuálna ukážka testovaného prototypu zobrazená na obrázku 5.1.



(a) Pohľad 1



(b) Pohľad 2

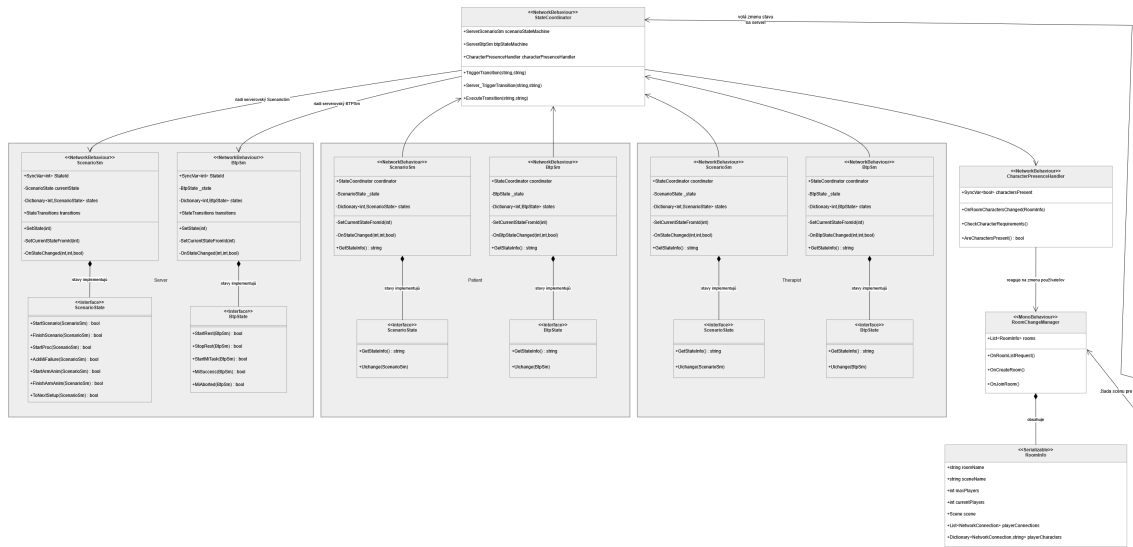
Obrázok 5.1: Dve ukážky prvého prototypu. Ľavý obrázok zobrazuje hlavnú ponuku („menu“), pravý obrázok zobrazuje prihlásenie používateľov.

6 Implementácia

Hlavným cieľom implementácie bolo vytvoriť aplikáciu, ktorá umožňuje klientom (v tomto prípade pacientovi a terapeutovi) vytvoriť vlastnú inštanciu hernej scény. Každá inštancia scény musí zabezpečovať správnu synchronizáciu sieťových objektov, pričom tieto objekty nesmú byť zdieľané ani synchronizované medzi rôznymi inštanciami tej istej scény.

Inými slovami, ak existuje napríklad miestnosť „Rehab1“, ktorá má v jednom časovom okamihu vytvorené dve samostatné inštancie na serveri, zmeny vykonané v jednej inštancii sa nesmú prejaviť v druhej. Každá inštancia musí byť úplne izolovaná, aby sa zabezpečila konzistentnosť a nezávislosť jednotlivých relácií. Táto izolácia je nevyhnutná pri implementovaní stavových strojov, keďže v každej samostatnej relácii môže byť rehabilitačný scénar v inom stave.

Na obrázku 6.1 je prezentovaná schéma architektúry systému testovaného prototypu. Zvyšok kapitoly je zameraný na opis implementačnej časti práce autora. Z dôvodu veľkosti celého obrázka systému nie je v tejto kapitole uvedený celý obrázok systému. Celý obrázok vznikol ako spoločné dielo autora a spoluriešiteľa Tomáša Süča, pričom jeho originálna verzia je súčasťou prílohy C záverečnej práce.



Obrázok 6.1: Architektúra systému prototypu

6.1 Stavový stroj

Súčasťou implementácie je stavový stroj, ktorý bol navrhnutý na základe konceptuálneho modelu Petriho siete opísaného v kapitole 4. Model je rozdelený do dvoch častí: *Scenario* a *Btp*.

```

1 public class ScenarioSm : NetworkBehaviour
2 {
3     private ScenarioState currentState;
4     public readonly SyncVar<int> StateId = new SyncVar<int>();
5 }

```

Implementácia jednotlivých stavov vychádza z návrhového vzoru State Pattern. Každý stav je reprezentovaný samostatnou triedou, ktorá implementuje základnú triedu. Týmto spôsobom je zabezpečená modularita, rozširiteľnosť a prehľadnosť implementácie.

Synchronizácia aktuálneho stavu medzi klientmi v rámci jednej miestnosti je realizovaná pomocou mechanizmu *SyncVar* zo sieťovej knižnice FishNet. Tento mechanizmus zabezpečuje, že všetci pripojení klienti majú konzistentnú hodnotu stavu bez ohľadu na čas ich pripojenia.

```

1 currentState?.StartScenario(this);

```

Vďaka použitiu návrhového vzoru *State Design Pattern* je implementácia stavových prechodov jednoduchá a prehľadná. Prechod medzi stavmi sa vykoná iba v prípade, že aktuálny stav implementuje požadovanú prechodovú metódu;

v opačnom prípade sa žiadna operácia nevykoná.

Na riadenie zmien stavov je použitý centrálny komponent *State Coordinator*, ktorý obsahuje serverovú metódu typu *ServerRpc*. Táto metóda zabezpečuje, že všetky požiadavky na zmenu stavu sú spracované na serveri a následne korektne synchronizované medzi všetkými klientmi.

```

1 [ServerRpc(RequireOwnership = false)]
2 public void ServerHandleStateChange(string machineType, string
   action, NetworkConnection sender = null)

```

V rámci implementácie sa stavové prechody vykonávajú iba ak sú v súlade s modelom Petriho siete v kapitole 4. Prechod *StartScenario* sa vykoná iba ak sú v jednej miestnosti súčasne prítomní terapeut aj pacient a taktiež aktuálny stav stavového stroja *ScenarioSm* je *ScenarioIdle*. V prípade, že by zmena stavu nebola realizovaná prostredníctvom metódy typu *ServerRpc* a namiesto toho by boli priamo volané metódy jednotlivých stavových objektov na klientovi, mohlo by dôjsť k nekonzistentnej synchronizácii medzi klientmi.

Počas testovania bolo pozorované, že takýto prístup fungoval korektne iba v prípade, keď boli klient a server spustené v režime host. V distribuovanom prostredí však dochádzalo k nesprávnemu správaniu aplikácie. Tento jav bol pravdepodobne spôsobený tým, že host režim môže maskovať problémy so synchronizáciou, ktoré sa prejavia až pri oddelenom behu klienta a servera.

6.1.1 Stavový stroj u klientov

V implementácii majú klienti a server odlišnú funkcionality stavových strojov. Tieto stavové stroje sú v rámci implementácie oddelené namespacom. V prípade servera spočíva funkcionality stavového stroja v uskutočňovaní jednotlivých stavových prechodov a následnej zmene stavu prislúchajúcej danému prechodu. Stavový stroj klientov následne reaguje na zmenu stavu serverového stavového stroja a na základe tejto zmeny aktualizuje aj svoj vlastný stav.

Jednotlivé stavy následne umožňujú implementovať špecifickú funkcionality pre pacienta a terapeuta. Takýto prístup zároveň znižuje duplicitu implementovanej logiky, keďže hlavné riadenie stavov prebieha centrálnne na serveri. Oddelenie serverovej a klientskej logiky zároveň zvyšuje modularitu systému, zjednodušuje budúce úpravy stavového stroja a umožňuje jednoduchšie rozširovanie aplikácie o nové stavy, funkcionality alebo používateľské roly bez potreby zásadných zmien existujúcej architektúry.

6.2 Správa viacerých herných miestností

Pre zabezpečenie paralelného behu viacerých nezávislých herných relácií bol implementovaný komponent `RoomManager`, ktorý rozširuje funkcionality sieťovej vrstvy o podporu architektúry založenej na miestnostiach. Každá miestnosť reprezentuje samostatnú inštanciu hernej scény, ktorá je izolovaná od ostatných. Ako inšpirácia pre toto riešenie poslužil návrh uvedený v [38].

6.2.1 Dátový model miestnosti

Základným prvkom je dátová štruktúra `RoomInfo`, ktorá uchováva všetky relevantné informácie o miestnosti, ako sú názov, typ scény, aktuálny počet používateľov a maximálna kapacita. Okrem toho obsahuje aj referenciu na konkrétnu inštanciu scény prostredia Unity a kolekcie aktívnych sieťových spojení.

Pre potreby aplikačnej logiky sú v rámci miestnosti evidované aj zvolené roly jednotlivých klientov, čo umožňuje validovať, či sú splnené podmienky na spustenie scenára (napr. prítomnosť terapeuta a pacienta).

6.2.2 Mapovanie klientov na miestnosti

Na efektívne vyhľadávanie je použitá pomocná mapovacia štruktúra, ktorá priradzuje každé sieťové spojenie (`NetworkConnection`) ku konkrétnej miestnosti. Tento prístup umožňuje konštantnú časovú zložitosť pri zisťovaní, do ktorej miestnosti klient patrí, čo je dôležité najmä pri spracovaní udalostí odpojenia.

6.2.3 Spracovanie požiadaviek klientov

Komunikácia medzi klientom a serverom je realizovaná pomocou broadcast správ poskytovaných knižnicou FishNet. Server registruje obslužné metódy pre správy typu:

- `RoomListRequestMessage` – požiadavka na zoznam dostupných miestností,
- `CreateRoomMessage` – požiadavka na vytvorenie novej miestnosti,
- `JoinRoomMessage` – požiadavka na pripojenie do existujúcej miestnosti.

Pri vytváraní miestnosti je použitý front požiadaviek, ktorý zabezpečuje sekvencné spracovanie, čím sa predchádza problémom pri súbežnom vytváraní via-

cerých scén. Samotné vytvorenie prebieha asynchrónne pomocou korutiny, ktorá načíta scénu v aditívnom režime.

6.2.4 Izolácia scén

Každá miestnosť je reprezentovaná samostatne načítanou scénou pomocou režimu `LoadSceneMode.Additive`. Tento prístup umožňuje existenciu viacerých inštancií tej istej scény súčasne, pričom každá obsahuje vlastné sieťové objekty.

Klienti sú následne priradení ku konkrétnej scéne pomocou metódy:

```
1 InstanceFinder.SceneManager.LoadSceneScenes(conn,
    sceneLoadData);
```

Tým je zabezpečené, že klient vidí iba objekty patriace do svojej miestnosti, čím sa dosahuje požadovaná izolácia.

6.2.5 Správa životného cyklu miestnosti

Systém dynamicky reaguje na pripojenie a odpojenie klientov. Pri odpojení klienta sa aktualizuje počet používateľov v miestnosti a odstránia sa všetky referencie na jeho spojenie. V prípade, že miestnosť ostane prázdna, jej scéna je uvoľnená z pamäte:

```
1 SceneManager.UnloadSceneAsync(scene);
```

Týmto spôsobom sa optimalizuje využitie systémových prostriedkov a zabráňuje sa zbytočnému držaniu neaktívnych scén.

6.2.6 Synchronizačné mechanizmy FishNetu

Synchronizačné mechanizmy vo frameworku FishNet zabezpečujú konzistentný stav hry medzi serverom a klientmi a predstavujú základ multiplayerovej funkcionality aplikácie. FishNet poskytuje viacero nástrojov určených na synchronizáciu objektov, prenos dát a komunikáciu medzi klientom a serverom.

SyncVar

Jedným zo základných mechanizmov sú `SyncVar` premenné, ktoré umožňujú automatickú synchronizáciu dát zo servera na všetkých klientov. Pri zmene hodnoty na serveri sa nová hodnota automaticky propaguje klientom bez potreby manuálneho odosielania správ.

Tento mechanizmus bol využitý napríklad pri synchronizácii aktuálneho stavu stavového stroja alebo pri validácii prítomnosti požadovaných rolí v miestnosti.

NetworkTransform

Na synchronizáciu pohybu objektov bol použitý komponent `NetworkTransform`. Tento komponent zabezpečuje synchronizáciu: pozície objektu, rotácie objektu a mierky objektu.

Synchronizácia prebieha priebežne počas behu aplikácie a môže využívať interpoláciu na dosiahnutie plynulejšieho vizuálneho výsledku. Tento mechanizmus je dôležitý najmä pri VR aplikáciách, kde je potrebné zabezpečiť plynulý pohyb avatarov a interaktívnych objektov.

Remote Procedure Calls (RPC)

FishNet poskytuje podporu pre vzdialené volania procedúr RPC, ktoré umožňujú explicitnú komunikáciu medzi klientom a serverom.

Tieto RPC volania sú: `ServerRpc` (volania smerované z klienta na server) a `ObserversRpc` (volania zo servera na klientov).

Tieto mechanizmy umožňujú synchronizovať hernú logiku, spracovávať požiadavky klientov a zabezpečovať konzistentné vykonávanie stavových zmien medzi všetkými účastníkmi relácie.

Synchronizačné metódy

Pre pokročilejšie scenáre FishNet poskytuje aj synchronizované kolekcie, napríklad `SyncLists`, ktoré umožňujú synchronizáciu zoznamov dát medzi serverom a klientmi.

Okrem toho framework podporuje vlastné serializačné mechanizmy, ktoré umožňujú efektívnejší prenos komplexnejších dátových štruktúr.

Správa udalostí

Pri práci so synchronizačnými mechanizmami je dôležité správne spravovať registráciu a odoberanie udalostí (*event subscription management*). V implementácii bolo potrebné zabezpečiť korektné odhlasovanie udalostí pri deaktivácii objektov.

Na tento účel bola využitá metóda `OnDisable`, v ktorej sa vykonáva odhlásenie zo všetkých registrovaných eventov.

Ak by odhlásenie nebolo vykonané správne, mohlo by dôjsť k: prijímaniu udalostí deaktivovanými objektmi, chybám typu `NullReferenceException`, nekonzistentnému správaniu aplikácie a únikom pamäte.

Správna správa životného cyklu eventov preto výrazne prispieva k stabilite aplikácie a korektnému fungovaniu synchronizačných mechanizmov.

6.2.7 Validácia prítomnosti rolí

Pre správne fungovanie herného scenára je potrebné zabezpečiť, aby sa v každej miestnosti nachádzali požadované roly, konkrétne terapeut a pacient. Na tento účel bol implementovaný komponent `CharacterPresenceHandler`.

Tento komponent využíva synchronizačný mechanizmus `SyncVar` na uchovávanie informácie o tom, či sú v danej miestnosti prítomné všetky požadované roly. Hodnota tejto premennej je riadená výhradne na strane servera, čím sa zabezpečuje konzistentnosť medzi všetkými klientmi.

```
1 public readonly SyncVar<bool> charactersPresent = new SyncVar<bool>
    >();
```

Komponent reaguje na zmeny v zložení miestnosti prostredníctvom udalosti `OnRoomCharactersChanged`. Pri každej zmene sa vykoná kontrola, či sú splnené podmienky prítomnosti požadovaných rolí.

```
1 bool hasTherapist = room.playerCharacters.Values.Contains(
    therapistCharacterName);
2 bool hasPatient = room.playerCharacters.Values.Contains(
    patientCharacterName);
3 charactersPresent.Value = hasTherapist && hasPatient;
```

V prípade, že je kontrola rolí deaktivovaná (napr. pre testovacie účely), komponent umožňuje ignorovať túto podmienku a považuje požiadavky za splnené.

Výsledná hodnota je následne využiteľná ostatnými časťami systému, napríklad stavovým strojom, ktorý môže na jej základe rozhodovať o povolení prechodu do ďalších stavov scenára. Týmto spôsobom je zabezpečené, že herná logika sa vykonáva iba v prípade splnenia základných podmienok.

6.2.8 Separácia serverového a klientského buildu

V prostredí Unity je možné oddeliť serverový a klientský kód pomocou podmienenej kompilácie. Kód umiestnený medzi týmito direktívami sa zahrnie iba do

serverového buildu. Pri klientskom builde sa tento kód nekompiluje. Takýmto spôsobom je možné oddeliť logiku servera od klienta a zároveň znížiť veľkosť klientskej aplikácie.

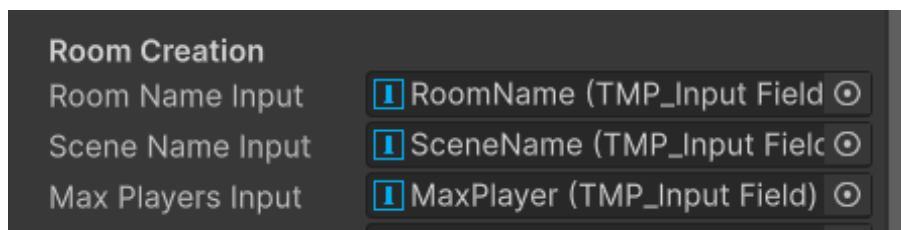
```
1 #if UNITY_SERVER
2 // kód
3 #endif
```

Ak je potrebné oddeliť časť kódu určenú výhradne pre klienta, je možné podmienku negovať.

```
1 #if !UNITY_SERVER
2 // kód
3 #endif
```

Takto oddelený kód zvyšuje optimalizáciu jednotlivých buildov a zároveň zlepšuje prehľadnosť projektu.

Tento prístup však má aj určité obmedzenia. Pri použití referencií na objekty 6.2 priradené cez Unity Inspector je potrebné zabezpečiť, aby deklarácie týchto referencií zostali mimo podmienenej kompilácie. V opačnom prípade môže dôjsť k strate priradených referencií pri prepínaní medzi serverovým a klientskym buildom, čo si následne vyžaduje ich opätovné manuálne priradenie.

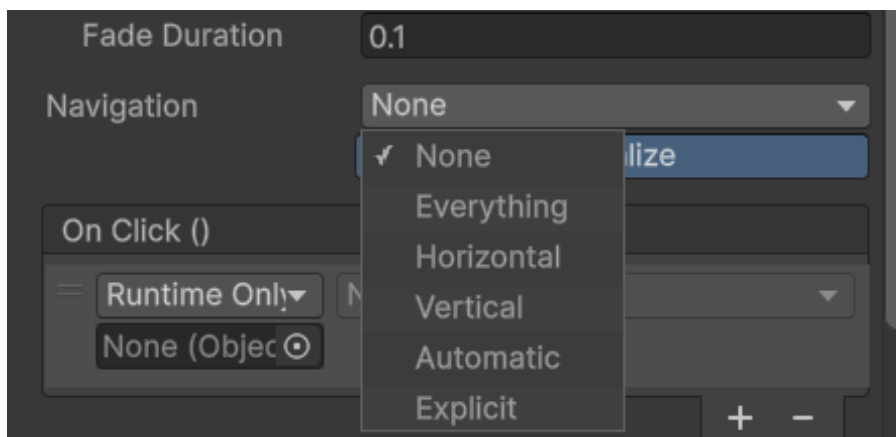


Obrázok 6.2: Referencie priradené cez Unity Inspector

6.2.9 UI navigácia

Počas testovania používateľského rozhrania v Unity bol identifikovaný problém s automatickou navigáciou medzi komponentmi používateľského rozhrania (User Interface – UI). Pri pohybe používateľa pomocou klávesov WASD dochádzalo v niektorých prípadoch k neúmyselnému presunu označenia na iný prvok používateľského rozhrania, napríklad na tlačidlo alebo vstupné pole. Tento problém bol spôsobený automaticky nastavenou navigáciou medzi komponentmi. Riešením bolo upravenie parametra Navigation na hodnotu None ako je na obrázku

6.3, čím sa zabránilo nechcenému prepínaniu medzi UI prvkami a zlepšila sa celková používateľská skúsenosť pri ovládaní aplikácie (v rámci spoločnej implementácie chýba ošetrenie tohto problému z dôvodu časovej limitácie).



Obrázok 6.3: UI navigácia

7 Vyhodnotenie

Cieľom implementácie bolo vytvoriť funkčný prototyp multiplayerovej VR aplikácie určenej pre neurorehabilitáciu horných končatín s podporou synchronizácie viacerých používateľov v reálnom čase. Súčasťou vyhodnotenia bolo overenie správnosti implementácie sieťovej architektúry, synchronizačných mechanizmov, správy herných miestností a stability aplikácie pri behu vo VR aj mimo VR prostredia.

Implementácia bola priebežne testovaná počas celého vývojového procesu. Overované boli najmä:

- správnosť synchronizácie sieťových objektov,
- stabilita komunikácie medzi klientom a serverom,
- korektné fungovanie stavového stroja,
- izolácia jednotlivých herných miestností,
- synchronizácia stavov medzi klientmi,
- správa používateľských rolí,
- stabilita používateľského rozhrania.

Počas testovania boli identifikované viaceré problémy súvisiace najmä so synchronizáciou objektov a riadením stavových prechodov. Významný problém predstavovalo správanie aplikácie v režime host, kde boli niektoré chyby synchronizácie skryté v dôsledku spoločného behu klienta a servera v jednom procese. Pri oddelenom spustení klienta a servera sa následne prejavili nekonzistentné stavové prechody a problémy pri synchronizácii herného stavu. Tento problém bol vyriešený centralizáciou zmien stavov prostredníctvom serverových RPC volaní.

Ďalším identifikovaným problémom bola automatická navigácia medzi prvkami používateľského rozhrania v prostredí Unity. Pri ovládaní aplikácie pomocou klávesnice dochádzalo k nechcenému presúvaniu aktívneho prvku medzi

UI komponentmi. Úpravou parametra `Navigation` na hodnotu `None` sa podarilo odstrániť nežiaduce správanie a zlepšiť používateľský komfort.

Testovanie synchronizácie objektov bolo realizované pomocou dvoch klientov pripojených v lokálnej sieti. Synchronizované boli pohyby avatarov, stav interaktívnych objektov a zmeny stavového stroja. Počas testovania neboli zaznamenané výrazné problémy so synchronizáciou ani nestabilita spojenia. Prenos dát medzi klientmi zostal stabilný aj pri súčasnom pohybe viacerých objektov.

Dôležitou súčasťou vyhodnotenia bolo aj overenie izolácie herných miestností. Testovaním bolo potvrdené, že jednotlivé inštancie scén zostávajú od seba oddelené a zmeny vykonané v jednej miestnosti sa neprejavujú v ostatných reláciách. Tento mechanizmus je nevyhnutný pre budúce rozšírenie systému o podporu viacerých paralelných terapeutických relácií. Výsledky testovania sú uvedené v tabuľke 7.1.

Test	Výsledok
Synchronizácia stavov	úspešná
Izolácia miestností	úspešná
Multiplayer (4 klienti)	úspešný
VR klient	funkčný
Desktop klient	funkčný

Tabuľka 7.1: Výsledky testovania prototypu

Počas testovania nevykazoval framework `FishNet` problémy, ktoré by obmedzovali navrhované využitie systému. Synchronizačné mechanizmy poskytovali dostatočne plynulú komunikáciu medzi klientmi a umožnili efektívnu správu sieťových objektov v reálnom čase.

Výsledný prototyp predstavuje funkčný základ pre ďalší vývoj systému neurorehabilitácie vo virtuálnej realite. Architektúra systému bola navrhnutá s dôrazom na modularitu, rozširiteľnosť a jednoduchšiu údržbu, čo umožňuje budúce rozšírenie o nové rehabilitačné scenáre, typy používateľov alebo ďalšiu budúcu integráciu.

8 Záver

Cieľom záverečnej práce bolo implementovať prototyp novej verzie neurorehabilitačného systému. Navrhnuté riešenie využíva systém stavových strojov na jednoznačnú identifikáciu aktuálneho stavu scenára a riadené prechody medzi jednotlivými stavmi.

Synchronizácia stavových strojov je zabezpečená prostredníctvom mechanizmu SyncVar. Rovnako je pomocou sieťovej knižnice FishNet implementovaná synchronizácia ďalších multiplayerových komponentov. Knižnica FishNet je plne kompatibilná s herným enginom Unity a umožňuje efektívnu správu sieťovej komunikácie medzi klientmi a serverom.

Implementované riešenie podporuje vytváranie samostatných miestností, čím zabezpečuje izoláciu jednotlivých terapeutických relácií. Vďaka tomu môže aplikáciu súčasne využívať viacero pacientov bez vzájomného ovplyvňovania ich rehabilitačných cvičení.

Prototyp zároveň predstavuje stabilný základ pre budúci rozvoj systému. Medzi možné rozšírenia patrí napríklad pridanie roly pozorovateľa, ktorý by mohol sledovať priebeh rehabilitácie bez priameho zásahu do nej. Ďalšie možnosti rozvoja zahŕňajú vytváranie nových rehabilitačných scenárov, rozširovanie virtuálnych prostredí a implementáciu ďalších funkcií podľa potrieb terapeutov a pacientov.

Zoznam použitej literatúry

1. GEORGIEV, Danko D.; GEORGIEVA, Iva; GONG, Zhengya; NANJAPPAN, Vijayakumar; GEORGIEV, Georgi V. Virtual Reality for Neurorehabilitation and Cognitive Enhancement. *Brain Sciences*. 2021, roč. 11, č. 2. ISSN 2076-3425. Dostupné z doi: 10.3390/brainsci11020221.
2. KOREČKO, Štefan; MATHES, Jaroslav; SOBOTA, Branislav; CHMURA, Damián; ROSIPAL, Roman; VANKÓ, Martin. L-NeRVEn Virtual Environment for Neurorehabilitation: Architecture, Functionality, and Evaluation. *Open Computer Science*. 2026, roč. 16, č. 1, s. 20250049. ISSN 2299-1093. Dostupné z doi: 10.1515/comp-2025-0049.
3. ADERINTO, Nicholas; MUIII, Abdulbasit; OLATUNJI, Gbolahan; PATIENCE, Adejumo. Exploring the Transformative Influence of Neuroplasticity on Stroke Rehabilitation: A Narrative Review of Current Evidence. *Annals of Medicine Surgery*. 2023, roč. 85. Dostupné z doi: 10.1097/MS9.00000000000001137.
4. B'ARTLOV'A, Sylva; ŠEDOV'A, Lenka; HAVIERNIKOV'A, Lucie; HUD'AČKOV'A, Andrea; DOL'AK, František; SAD'ILEK, Petr. Quality of Life of Post-stroke Patients. *Zdravstveno varstvo*. 2022, roč. 61, č. 2, s. 101–108. Dostupné z doi: 10.2478/sjph-2022-0014.
5. WIJERATNE, Tissa; SALES, Carmela. Understanding Why Post-Stroke Depression May Be the Norm Rather Than the Exception: The Anatomical and Neuroinflammatory Correlates of Post-Stroke Depression. *Journal of Clinical Medicine*. 2021, roč. 10, č. 8, s. 1674. Dostupné z doi: 10.3390/jcm10081674.
6. FEIGIN, Valery L.; STARK, Brian A.; JOHNSON, Catherine O. et al. Global, regional, and national burden of stroke and its risk factors, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019. *The Lancet Neurology*. 2021, roč. 20, č. 10, s. 795–820. Dostupné z doi: 10.1016/S1474-4422(21)00252-0.

7. LANGHORNE, Peter; BERNHARDT, Julie; KWAKKEL, Gert. Stroke Rehabilitation. *The Lancet*. 2011, roč. 377, č. 9778, s. 1693–1702. ISSN 0140-6736. Dostupné z DOI: 10.1016/S0140-6736(11)60325-5.
8. HANKEY, Graeme J. Stroke. *The Lancet*. 2017, roč. 389, č. 10069, s. 641–654. ISSN 0140-6736. Dostupné z DOI: 10.1016/S0140-6736(16)30962-X.
9. VENTURA, Sara; BRIVIO, Eleonora; RIVA, Giuseppe; BAÑOS, Rosa M. Immersive versus non-immersive experience: Exploring the feasibility of memory assessment through 360 technology. *Frontiers in psychology*. 2019, roč. 10, s. 2509. Dostupné z DOI: 10.3389/fpsyg.2019.02509.
10. NEHILA, Peter. *Neurorehabilitácia v zdieľanej virtuálnej realite*. 2023. Dostupné tiež z: <https://opac.crzp.sk/?fn=detailBiblioForm&sid=880D977B60C8641DD3EE229C4FC3>.
11. LEVAC, Danielle; MILLER, Patricia; MISSIUNA, Cheryl. Usual and Virtual Reality Video Game-Based Physiotherapy for Children and Youth with Acquired Brain Injuries. *Physical & Occupational Therapy in Pediatrics*. 2012, roč. 32, č. 2, s. 180–195. Dostupné z DOI: 10.3109/01942638.2011.616266.
12. SAPOSNIK, Gustavo; TEASELL, Robert; MAMDANI, Muhammad; HALL, Judith; MCILROY, William; CHEUNG, Donna; THORPE, Kevin E.; COHEN, Leonardo G.; BAYLEY, Mark. Effectiveness of Virtual Reality Using Wii Gaming Technology in Stroke Rehabilitation. *Stroke*. 2010, roč. 41, č. 7, s. 1477–1484. Dostupné z DOI: 10.1161/STROKEAHA.110.584979.
13. KELLER, Jiří; ŠTĚTKÁŘOVÁ, Ivana; MACRI, Vince et al. Virtual Reality-Based Treatment for Regaining Upper Extremity Function Induces Cortex Grey Matter Changes in Persons with Acquired Brain Injury. *Journal of NeuroEngineering and Rehabilitation*. 2020, roč. 17, č. 1, s. 127. Dostupné z DOI: 10.1186/s12984-020-00754-7.
14. LEE, Su-Hyun; KIM, Yu-Mi; LEE, Byoung-Hee. Effects of virtual reality-based bilateral upper-extremity training on brain activity in post-stroke patients. *Journal of Physical Therapy Science*. 2015, roč. 27, č. 7, s. 2285–2287. Dostupné z DOI: 10.1589/jpts.27.2285.
15. PIRON, Luigi; CENNI, Francesca; TONIN, Paolo; DAM, Mario. Virtual Reality as an Assessment Tool for Arm Motor Deficits After Brain Lesions. *Studies in Health Technology and Informatics*. 2001, roč. 81, s. 386–392.

16. DIERS, Martin; KAMPING, Sandra; KIRSCH, Philipp et al. Illusion-Related Brain Activations: A New Virtual Reality Mirror Box System for Use During Functional Magnetic Resonance Imaging. *Brain Research*. 2015, roč. 1594, s. 173–182. Dostupné z doi: 10.1016/j.brainres.2014.11.001.
17. DABROWSKÁ, M.; HONZÍKOVÁ, L.; PASTUCHA, D. et al. Virtual Reality as a Potential Therapy in a Rehabilitation Sanatorium for Patients After Ischemic Stroke: Impact on Quality of Life and Social Participation—A Randomized Trial. *Frontiers in Rehabilitation Sciences*. 2025, roč. 6, s. 1539175. Dostupné z doi: 10.3389/fresc.2025.1539175.
18. BESNARD, Jeremy; RICHARD, Paul; BANVILLE, Frederic; NOLIN, Pierre; AUBIN, Ghislaine; GALL, Didier Le; RICHARD, Isabelle; ALLAIN, Philippe. Virtual reality and neuropsychological assessment: The reliability of a virtual kitchen to assess daily-life activities in victims of traumatic brain injury. *Applied Neuropsychology: Adult*. 2016, roč. 23, č. 3, s. 223–235. Dostupné z doi: 10.1080/23279095.2015.1048514. PMID: 26569498.
19. FONG, Kenneth N. K.; CHOW, Kathy Y. Y.; CHAN, Bianca C. H. et al. Usability of a Virtual Reality Environment Simulating an Automated Teller Machine for Assessing and Training Persons with Acquired Brain Injury. *Journal of NeuroEngineering and Rehabilitation*. 2010, roč. 7, č. 1, s. 19. Dostupné z doi: 10.1186/1743-0003-7-19.
20. GREALY, Madeleine A.; JOHNSON, David A.; RUSHTON, Simon K. Improving Cognitive Function After Brain Injury: The Use of Exercise and Virtual Reality. *Archives of Physical Medicine and Rehabilitation*. 1999, roč. 80, č. 6, s. 661–667. Dostupné z doi: 10.1016/S0003-9993(99)90169-7.
21. DE LUCA, Rosaria; MAGGIO, Maria Grazia; MARESCA, Giuseppa; LATELLA, Desiree; CANNAVÒ, Antonino; SCIARRONE, Francesca; LO VOI, Emanuele; ACCORINTI, Maria; BRAMANTI, Placido; CALABRÒ, Rocco Salvatore. Improving Cognitive Function after Traumatic Brain Injury: A Clinical Trial on the Potential Use of the Semi-Immersive Virtual Reality. *Behavioural Neurology*. 2019, roč. 2019, č. 1, s. 9268179. Dostupné z doi: <https://doi.org/10.1155/2019/9268179>.
22. NEUMANN, Dawn; MALEC, James F.; HAMMOND, Flora M. Reductions in Alexithymia and Emotion Dysregulation After Training Emotional Self-Awareness Following Traumatic Brain Injury: A Phase I Trial. *The Journal of Head Trauma Rehabilitation*. 2017, roč. 32, č. 5, s. 286–295. Dostupné z doi: 10.1097/HTR.000000000000277.

23. BERDUGO-VEGA, Gabriel; ARIAS-GIL, Gonzalo; LÓPEZ-FERNÁNDEZ, Adrian et al. Increasing neurogenesis refines hippocampal activity rejuvenating navigational learning strategies and contextual memory throughout life. *Nature Communications*. 2020, roč. 11, č. 1, s. 135. Dostupné z doi: 10.1038/s41467-019-14026-z.
24. BASSO, Julia C.; SUZUKI, Wendy A. The Effects of Acute Exercise on Mood, Cognition, Neurophysiology, and Neurochemical Pathways: A Review. *Brain Plasticity*. 2017, roč. 2, č. 2, s. 127–152. ISSN 2213-6304. Dostupné z doi: 10.3233/BPL-160040.
25. CAÑA-PINO, Alejandro; HOLGADO-LÓPEZ, Paula. Wearable-Sensor and Virtual Reality-Based Interventions for Gait and Balance Rehabilitation in Stroke Survivors: A Systematic Review. *Signals*. 2025, roč. 6, č. 3, s. 48. ISSN 2624-6120. Dostupné z doi: 10.3390/signals6030048.
26. POP-KUN, R.; TRUȚĂ, A.; ȘTEFĂNESCU, E.; MUREȘANU, D.; STRILCIUC, Ș.; CLICHICI, S. Virtual Reality-Assisted Rehabilitation for Upper-Limb Function in Stroke Survivors: A Systematic Review. *Brain Sciences*. 2026, roč. 16, č. 4, s. 417. ISSN 2076-3425. Dostupné z doi: 10.3390/brainsci16040417.
27. SOLEIMANI, Mohsen; GHAZISAEEDI, Marjan; HEYDARI, Soroush. The efficacy of virtual reality for upper limb rehabilitation in stroke patients: a systematic review and meta-analysis. *BMC Medical Informatics and Decision Making*. 2024, roč. 24, č. 1, s. 135. ISSN 1472-6947. Dostupné z doi: 10.1186/s12911-024-02534-y.
28. DONATI, Davide; PINOTTI, Elena; MANTOVANI, Marta; CASAROTTI, Silvia; FINI, Alessandra; TEDESCHI, Riccardo; CASELLI, Stefano. The Role of Immersive Virtual Reality in Upper Limb Rehabilitation for Subacute Stroke: A Review. *Journal of Clinical Medicine*. 2025, roč. 14, č. 6, s. 1903. ISSN 2077-0383. Dostupné z doi: 10.3390/jcm14061903.
29. FLUET, Gerard G.; QIU, Qiang; PATEL, Jigar; CRONCE, Amanda; MERIANS, Aviva S.; ADAMOVICH, Sergei V. Autonomous Use of the Home Virtual Rehabilitation System: A Feasibility and Pilot Study. *Games for Health Journal*. 2019, roč. 8, č. 6, s. 432–438. Dostupné z doi: 10.1089/g4h.2019.0012.
30. STANICA, Iulia-Cristina; MOLDOVEANU, Florica; PORTELLI, Giovanni-Paul; DASCALU, Maria-Iuliana; MOLDOVEANU, Alin; RISTEA, Mariana Georgiana. Flexible Virtual Reality System for Neurorehabilitation and Quality of Life Improvement. *Sensors*. 2020, roč. 20, č. 21. ISSN 1424-8220. Dostupné z doi: 10.3390/s20216045.

31. KENEA, Chala Diriba; ABESSA, Teklu Gemechu; LAMBA, Dheeraj; BONNECHÈRE, Bruno. AdaptRehab VR: Development of an Immersive Virtual Reality System for Upper Limb Stroke Rehabilitation Designed for Low-and Middle-Income Countries Using a Participatory Co-Creation Approach. *Bioengineering*. 2025, roč. 12, č. 6. ISSN 2306-5354. Dostupné z DOI: 10.3390/bioengineering12060581.
32. SALEH, Neven; SALAHELDIN, Ahmed M.; BADAWI, Maged; EL-BIALY, Ahmed. Rehabilitative game-based system for enhancing physical and cognitive abilities of neurological disorders. *Cognitive Neurodynamics*. 2025, roč. 19, č. 1, s. 48. Dostupné z DOI: 10.1007/s11571-025-10229-x.
33. CONG, Xu; LI, Tingting. Design and Development of Virtual Medical System Interface Based on VR-AR Hybrid Technology. *Computational and Mathematical Methods in Medicine*. 2020, roč. 2020, Article ID 7108147. Dostupné z DOI: 10.1155/2020/7108147.
34. UNITY TECHNOLOGIES. *Unity User Manual* [online]. 2025. [cit. 2025-11-11]. Dostupné z: <https://docs.unity3d.com/Manual/index.html>.
35. MIRROR NETWORKING COMMUNITY. *Mirror Networking Documentation* [online]. 2025. [cit. 2025-11-11]. Dostupné z: <https://mirror-networking.com/>.
36. VIS2K; MIRROR NETWORKING CONTRIBUTORS. *Mirror Networking* [online]. 2025. [cit. 2025-11-11]. Dostupné z: <https://github.com/MirrorNetworking/Mirror>.
37. NETWORKING, Fish. *Fish Networking Documentation* [online]. 2025. [cit. 2025-11-24]. Dostupné z: <https://fish-networking.gitbook.io/docs>.
38. SEAN-ZEO. *Multi-Room-Manager-For-FishNet-Unity* [online]. 2025. [cit. 2026-03-03]. Dostupné z: <https://github.com/Sean-Zeo/Multi-Room-Manager-For-FishNet-Unity/>.

Zoznam príloh

Príloha A Používateľská príručka

Príloha B Systémová príručka

Príloha C CD médium – záverečná práca v elektronickej podobe obsahuje zdrojový kód, systémovú dokumentáciu, spustiteľné súbory a ďalšie prílohy

A Používateľská príručka

Používateľskú príručku pre tento prototyp je možné nájsť v práci spoluriešiteľa projektu Tomáša Süča, ktorý popisuje spôsob inštalácie, spustenia a používania systému L-NeRVEn.

B Systémová príručka

Táto systémová príručka bola vytvorená pre spoločný prototyp novej verzie systému L-NeRVEn, na ktorom sa podieľali autori Martin Čižmárik a Tomáš Süč.

1 Vytváranie spustiteľných súborov

Proces vytvárania spustiteľných súborov v prostredí Unity sa vykonáva prostredníctvom menu File → Build Settings. V tomto okne je možné zvoliť cieľovú platformu, spravovať scény zahrnuté do buildu a upraviť ďalšie nastavenia potrebné pre vytvorenie aplikácie.

Pred spustením buildu je potrebné skontrolovať, či sú v zozname zahrnuté všetky scény používané projektom. Scény, ktoré nie sú označené, nebudú súčasťou výslednej aplikácie, čo môže spôsobiť chyby pri spustení alebo nefunkčné prechody medzi scénami. Je potrebné sa uistiť, že úvodná scéna má index 0, aby sa predišlo neočakávaným chybám.

1.1 Vytváranie serverovej aplikácie pre Windows

V prostredí Unity je najskôr potrebné otvoriť okno Build Settings cez menu File → Build Settings. V ľavej časti okna sa následne vyberie možnosť Dedicated Server. Ako cieľová platforma sa nastaví Windows, keďže serverová časť aplikácie bola primárne testovaná práve na tomto operačnom systéme. Následne sa klikne na tlačidlo Switch Platform.

Pred vytvorením buildu je potrebné overiť, že sa v Lobby scéne na objekte Network nachádza v skripte NetworkConnectionHandler parameter

`serverBuildStartMode` nastavený na hodnotu `AutoStartServer`, aby mohol framework FishNet automaticky spustiť aplikáciu v režime servera.

1.2 Vytváranie aplikácie pre desktopové zariadenia

Pri vytváraní desktopovej verzie aplikácie sa v okne Build Settings vyberie platforma Windows a následne sa klikne na tlačidlo Switch Platform.

1.3 Vytváranie aplikácie pre VR

Pri vytváraní VR verzie aplikácie pre zariadenia Meta Quest je potrebné v okne Build Settings zvoliť platformu Android a následne kliknúť na tlačidlo Switch Platform. Pred vytvorením buildu je potrebné mať v projekte nainštalovaný Android Build Support vrátane Android SDK.

V nastaveniach projektu je zároveň potrebné skontrolovať, či sú povolené XR pluginy OpenXR a Meta XR. Tieto nastavenia sa nachádzajú v menu Edit → Project Settings → XR Plug-in Management.

Po úspešnom prepnutí platformy je možné vytvoriť APK build pomocou tlačidla Build alebo Build And Run.

2 Dokumentácia kódu

Zdrojový kód systémových komponentov opísaných v tejto práci a znázornených v architektúre systému obsahuje dokumentačné komentáre vo formáte C#, z ktorých bola pomocou nástroja Doxygen vygenerovaná technická dokumentácia implementovaných tried a metód. Táto dokumentácia sa nachádza v CD prílohe práce.