

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Diplomová práce

System pro distribuci a správu kognitivních her

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Václav KRAFT**

Osobní číslo: **A16N0045P**

Studijní program: **N3902 Inženýrská informatika**

Studijní obor: **Softwarové inženýrství**

Název tématu: **Systém pro distribuci a správu kognitivních her**

Zadávací katedra: **Katedra informatiky a výpočetní techniky**

Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se s problematikou měření mozkové aktivity a kognitivního tréninku u lidí.
2. Prostudujte problematiku měření mozkové aktivity členkami firmy Neurosky Mindwave mobile a seznamte se s hrami implementovanými na KIV pro kognitivní trénink.
3. Navrhněte a implementujte mobilní aplikaci pro kognitivní trénink měřené osoby, u které bude k ovládní využita úroveň meditace, popř. soustředění hráče získaná z Mindwave mobile členky.
4. Navrhněte a implementujte systém, který bude sloužit pro distribuci aplikací pro kognitivní trénink a zároveň bude ukládat naměřená data.
5. Proveďte sérii měření na různých osobách, výsledky vyhodnoťte a diskutujte.

Rozsah grafických prací: **dle potřeby**
Rozsah kvalifikační práce: **doporuč. 50 s. původního textu**
Forma zpracování diplomové práce: **tištěná**
Seznam odborné literatury:
dodá vedoucí diplomové práce.

Vedoucí diplomové práce: **Ing. Petr Brůha**
Nové technologie pro informační společnost

Datum zadání diplomové práce: **1. září 2017**
Termín odevzdání diplomové práce: **17. května 2018**

Radová

Doc. Dr. Ing. Vlasta Radová
děkanka



Brada

Doc. Ing. Přemysl Brada, MSc. Ph.D.
vedoucí katedry

V Plzni dne 14. září 2017

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 17. května 2018

Václav Kraft

Abstract

The aim of this diploma thesis is to design and implement a system for the management and distribution of cognitive games. The motivation for its creation is to simplify administration in therapy for cognitive competence training and to analyze patient results. The Department of Computer Science and Engineering at the University of West Bohemia has already created several cognitive games in the form of mobile applications that can be used for rehabilitation. Cognitive AMP will incorporate these and newly created applications and create an environment for their simple management and analysis of measured medical data.

Abstrakt

Cílem této diplomové práce je navrhnout a implementovat systém pro správu a distribuci kognitivních her. Motivem pro jeho vytvoření je zjednodušení administrativy při terapiích pro trénink kognitivních schopností a možnost analýzy výsledků pacienta. Na Katedře informatiky a výpočetní techniky na Západočeské univerzitě již vzniklo několik kognitivních her ve formě mobilních aplikací, které lze pro rehabilitaci využít. Systém s názvem Cognitive AMP (kognitivní zesilovač) bude tyto a nově vzniklé aplikace obsahovat a vytvoří prostředí pro jejich jednoduchou správu a analýzu naměřených medicínských dat.

Obsah

Úvod	8
Analytická část	10
1 Mozková aktivita	10
1.1 Definice mozkové aktivity a její vznik	10
1.1.1 Neuron	10
1.1.2 Synapse	12
1.1.3 Vznik mozkové aktivity	14
1.2 Měření mozkové aktivity	15
1.2.1 Elektroencefalografie	15
1.2.2 Snímkování	19
1.2.3 Implantované technologie	19
1.3 Manipulace s naměřenými daty - GDPR	20
2 Kognitivní funkce mozku	21
2.1 Rozdělení	21
2.1.1 Základní kognitivní funkce	21
2.1.2 Vyšší kognitivní funkce	24
2.1.3 Metakognitivní schopnosti	25
2.2 Kognitivní porucha	26
2.3 Metody hodnocení kognitivních funkcí	26
2.3.1 Mini-Mental State Examination (MMSE)	26
2.3.2 Montrealský kognitivní test (MoCA-CZ1)	27
2.3.3 Sedmiminutový screeningový test (7MST)	28
2.4 Trénink kognitivních funkcí	28
2.4.1 Pojmy kognitivní trénink, rehabilitace, neurorehabilitace	29
2.4.2 Vedoucí tréninku kognitivních funkcí	29
3 Komerční systémy pro kognitivní trénink	30
4 Hry pro kognitivní trénink na KIV	31
5 MindWave Mobile	32
5.1 Popis zařízení	32
5.2 Specifikace	33
5.3 ThinkGear modul - TGAM1	33

Praktická část - Cognitive Amp	35
6 Vize projektu	35
6.1 Popis problému a účelu systému	35
6.2 Popis zainteresovaných stran	36
6.3 Vize produktu	36
6.4 Seznam funkčních požadavků	36
6.5 Seznam mimo-funkčních požadavků	37
6.6 Rizika projektu	38
6.7 Omezení	39
6.8 Konzultace s odborníky v oboru kognitivního tréninku	39
7 Specifikace požadavků	41
7.1 Případy užití	41
7.2 Aktéři	43
8 Architektura systému	44
8.1 Softwarové a hardwarové vybavení	44
8.1.1 Server	44
8.1.2 Mobilní zařízení	44
8.2 Popis komponent systému	45
8.3 Server	46
8.3.1 Popis	46
8.3.2 Implementační technologie	47
8.3.3 Nasazení	48
8.3.4 Uživatelské rozhraní - webová aplikace	50
8.4 Knihovna - Communicator	50
8.4.1 Popis	50
8.4.2 Implementační technologie	51
8.5 Aplikace - Hub	52
8.5.1 Popis	52
8.5.2 Implementační technologie	53
8.5.3 Uživatelské rozhraní	53
8.6 Aplikace - Save the Princess	54
8.6.1 Popis	54
8.6.2 Implementační technologie	55
8.6.3 Uživatelské rozhraní	56
8.7 Komunikační schéma	56
8.8 Diagram datových toků	57

9 Implementace systému	59
9.1 Databáze	59
9.1.1 ERA model	60
9.1.2 Vložené funkce a triggery	63
9.2 Webová aplikace	64
9.2.1 Struktura projektu	64
9.2.2 Postup obsluhy požadavku	66
9.3 Communicator	67
9.4 Mobilní aplikace - Camp Hub	69
9.5 Mobilní aplikace - Save the Princess	70
9.6 Docker	73
10 Testování	75
10.1 Manuální testování	75
10.1.1 Testovací scénář	75
10.1.2 Průběh testování	77
10.1.3 Sumarizace výsledků a provedení oprav	78
10.2 Automatické testování	81
10.3 Test energetické náročnosti mobilních aplikací	81
10.3.1 Zhodnocení testování	82
11 Diskuze nad dosaženými výsledky	83
12 Doporučení pro rozšíření systému	85
Závěr	87
Literatura	88
Seznam použitých zkratk	90
Přílohy	91
Seznam obrázků	96
Seznam tabulek	97

Úvod

V psychologických teoriích Carla Junga jsou kognitivní funkce definovány jako různé způsoby vnímání a hodnocení. Dále se dělí na "myšlení", "pocit" a "intuice". [9]

Kognitivní funkce člověka, někdy také funkce poznávací, jsou nedílnou součástí lidské psychiky a díky nim je možné poznávat a interagovat s okolním světem. Ergoterapeutická profese zahrnuje vyšetření a případnou rehabilitaci pacienta s problémy, které způsobuje zhoršená funkce kognitivních schopností ve vztahu s běžnými každodenními činnostmi člověka. Nejčastěji se jedná o pacienty po poranění hlavy nebo s degenerativními změnami mozku. Existují metody, označované jako kognitivní hry, kterými je možné jednotlivé kognitivní funkce trénovat.

Katedra informatiky a výpočetní techniky na Západočeské univerzitě v Plzni, konkrétně neuroinformatická laboratoř, se ve spolupráci s Fakultou zdravotnických studií touto oblastí zabývá. Vzniklo zde již několik kognitivních her ve formě mobilních aplikací, které lze pro rehabilitaci využít. Cílem této diplomové práce je vytvořit systém, který bude používání těchto a nově vzniklých aplikací zastřešovat. Systém bude sloužit jak pro správu a distribuci těchto aplikací, ale i jako nástroj pro analýzu výsledků rehabilitace pacienta ergoterapeutem.

Předkládaný dokument je členěn na analytickou a praktickou část. V analytické části je vytvořen teoretický základ, který je nutný k dostatečnému porozumění dané problematice. Nejdříve je definována mozková aktivita jako taková, jakými způsoby je možné ji měřit a jaká omezení plynou pro manipulaci s naměřenými daty. Další kapitola se zabývá kognitivními funkcemi mozku, metodami hodnocení KF a jejich tréninkem. Dále bylo nutné analyzovat již existující komerční systémy pro kognitivní trénink, a aplikace vytvořené na KIV ZČU. Poslední kapitola analytické části obsahuje popis zařízení Mindwave mobile. Mindwave mobile je čelenkou snímající mozkovou aktivitu z frontální části mozku, díky čemuž ji lze využít pro kognitivní trénink.

Praktická část popisuje vytvořený systém *Cognitive AMP* (Kognitivní zesilovač), zkratkou *CAMP*. V první řadě bylo nutné formulovat problém, který bude nový systém řešit, definovat funkční i mimo-funkční požadavky a jiné aspekty nutné pro vytváření kvalitního softwarového produktu. Dále byla vytvořena architektura, která už obsahuje konkrétní technologicky závislá rozhodnutí, která plynou z požadavků zadavatele. V kapitole Imple-

mentace systému jsou vysvětleny zásadní implementační kroky. Následující kapitola se pak věnuje testování. V závěrečné části dokumentu je provedena diskuze nad vytvořeným systémem a formulováno několik doporučení pro plánované rozšiřování a zdokonalování systému *CAMP*.

Analytická část

1 Mozková aktivita

Mozek (latinsky: cerebrum, řecky: encephalon) je orgán, který slouží jako organizační a řídicí centrum nervové soustavy obratlovců. Lidský mozek je složen přibližně z 40 - 100 miliard neuronů. Každý neuron má více než 1000 spojů (synapsí), kterými komunikuje prostřednictvím elektrických signálů s jinými buňkami.[5]

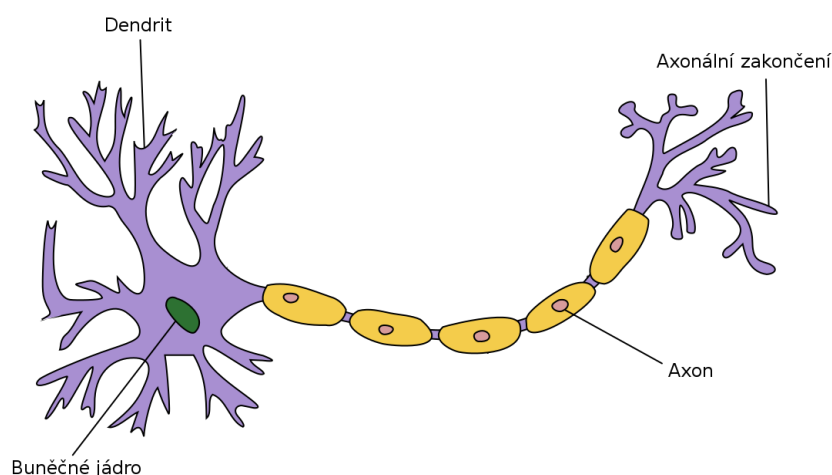
„Nepřeborná mnohotvárnost funkcí, obrovská paměťová kapacita a plasticita nervového systému, je umožněna vzájemným propojením miliard neuronů do neuronových sítí. Informace jsou sbírány z blízkého i vzdáleného okolí a z organismu samotného systémem senzorů, transformovány v nervový impuls a vedeny aferentními vlákny do příslušných center.“[8]

1.1 Definice mozkové aktivity a její vznik

V této podkapitole budou nejprve vysvětleny základními pojmy jako jsou *neuron* a *synapse*. Dále se zaměříme na mozkovou aktivitu jako takovou, jaký způsobem ji lze měřit a jak lze s těmito naměřenými medicínskými daty nakládat. Protože se jedná o velmi citlivá data, je nutné dodržovat jasně definovaná pravidla, postupy a využívat k tomu odpovídající techniku.

1.1.1 Neuron

Neuron je elementární stavební a funkční jednotka nervového systému, buňka uzpůsobená k vytváření vedení vzruchu. Je zpravidla opatřena dlouhými rozvětvenými výběžky, které vytvářejí nervová vlákna. Činnost neuronu bychom mohli připodobnit k funkci relé v komunikační síti. Z dostupných informací je schopen se autonomně rozhodnout o svém budoucím stavu 0 nebo 1. Anatomická i funkční nezávislost jednotlivých neuronů je umožněna jejich složitou stavbou, která je zobrazena na následujícím obrázku.[5]



Obrázek 1.1: Schéma neuronu[2]

Naprostá většina neuronů se v principu skládá ze tří snadno rozlišitelných částí:

- **Tělo** - též perikaryon, obsahuje jádro a přilehlou cytoplasmu. Jeho velikost se pohybuje od $6\ \mu\text{m}$ do $100\ \mu\text{m}$ a dokáže přijímat vzruchy cestou zakončení výběžků jiných neuronů. [8]
- **Dendrity** - krátké, bohatě větvené výběžky, specializované na vedení vzruchu směrem k jádru neuronu. Velmi účinně rozšiřují receptivní oblast neuronu. Existují neurony, ke kterým může vést až 200 tisíc zakončení. [8]
- **Axon** - též neurit sloužící k převodu vzruchu na jiné buňky. Vybíhá obvykle z těla neuronu a končí u efektorové buňky (specializované na konkrétní činnost, např. paměťová buňka) či u dalšího neuronu. Nejdelší axon v lidském těle měří přibližně 100 cm a vede od páteře ke konečkům prstů u nohou. Axony jsou většinou obaleny myelinovou pochvou. Výjimku tvoří neurony CNS, kde je asi 40 % axonů bez obalů (holá vlákna). Axon obalený pochvou se nazývá nervové vlákno. [8]

Tělo neuronu obsahuje kuželovitý iniciální segment zvaný axonální konus, na který je napojen axon. Plazmatická membrána axonu, zvaná axolemma, je podložena tenkou vrstvou denzního (hustého) materiálu a v místě iniciálního segmentu protkána množstvím iontových kanálů schopných generovat nervový vzruch. O tom, zda vzruch vznikne či nevznikne (stav pomyslného relé 1 či 0), rozhoduje algebraický součet excitačních a tlumivých stimulů soustředěných v axonálním konusu z celého recepčního povrchu neuronu.

Toto uspořádání napovídá, že neuron není pouhým vodičem, nýbrž generátorem, který v případě kladného rozhodnutí vzruch obnoví a předá na efektorovou buňku či další neuron v plné intenzitě. [8]

Rozdělení neuronů

Neurony se rozdělují dle profesora Richarda Jelínka následovně[8]:

- **Unipolární neurony** - jedná se o smyslové neurony a nedisponují dendrity, nýbrž pouze axonem, mají jeden výběžek. Mají specializované tyčinky, které nepřijímají signál od druhého neuronu, ale vzruch vytvářejí samy na základě podráždění přicházející z okolí. Může se jednat o stimulaci světla o konkrétní vlnové délce.
- **Bipolární neurony** - mají jeden dendrit a jeden axon na těle neuronu umístěné proti sobě. Příkladem je druhý neuron zrakové dráhy nebo čichové buňky.
- **Pseudounipolární neurony** - typ bipolárního neuronu, kdy dendrit a axon v blízkosti těla splývají. v jediný výběžek. Pseudounipolární neurony jsou typické pro spinální ganglia a ganglia mozkových nervů.
- **Multipolární neurony** - jsou nejběžnějším typem. Mají několik dendritů a jeden axon, takže mají hvězdicový tvar

1.1.2 Synapse

Vzájemné spojení mezi neurony, jehož funkcí je především převedení nervového vzruchu z neuronu na efektorovou buňku či na další neuron, umožňuje specializované buněčné spojení - synapse. Podle stavby a způsobu funkce rozeznáváme synapse elektrické a chemické. A synapse definujeme jako specializované buněčné spojení mezi dvěma neurony uzpůsobené k přenosu vzruchu.[8]

Synapse převádějí elektrický vzruch nejčastěji z axonu na dendrit (axodendrické) nebo perikayon (axosomatické). Možné jsou i přechody z dendritu na dendrit (dendrodendrické), ale i z axonu na axon (axoaxonální).[8]

Druhým typem dělení může být dělení dle funkce synapse:

- inhibiční synapse
- excitační synapse

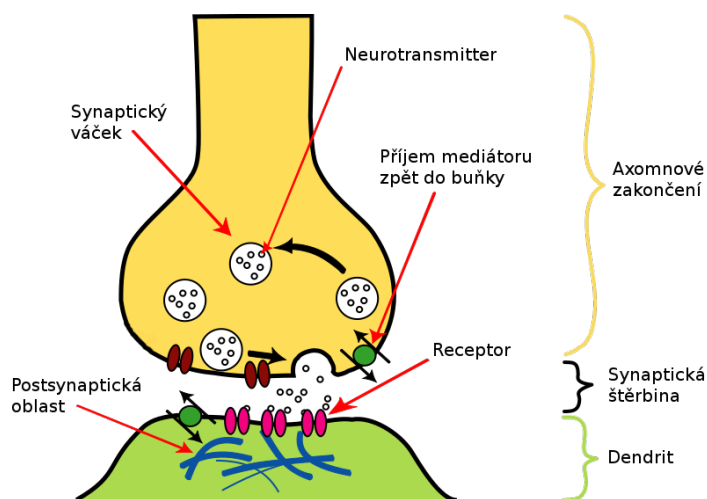
Rozdělení synapsí podle toho, zda-li dochází k vybudování (excitaci) či útlumu (inhibici) neuronů.

Elektrická synapse

Neliší se od nexu, který je typem mezibuněčného spojení. Zajišťuje elektrické spojení dvou neuronů přenosem iontů z jedné strany na druhou. [8]

Chemická synapse

Chemické synapse převádí elektrický vzruch pomocí specifických látek zvaných mediátory či neurotransmitery. Tvar synapse může být různý. Zakončení (terminálie) axonu vytváří kuželovité, košíčkovité či klubičkovité rozšíření. [8]



Obrázek 1.2: Schéma synapse [2](přeloženo)

Synapse se dle Jelínka skládá z [8]:

- cytoplazmatické membrány axonu zvané presynaptická membrána
- úzké štěrbiny (20-30 nm) extracelulárního prostoru (synaptická štěrбина)
- cytoplazmatické membrány dendritu, těla či axonu následné buňky

Synapse převádějí informaci pouze jedním směrem. Přestože jsou nervové dráhy obousměrné, jednosměrné vedení zajistí právě synapse. Zpoždění převodu informace je přibližně 0,5-25 ms. Mediátory jsou z váček uvolňovány

do synaptické štěrbině exocytózou. Exocytóza je proces, kterým buňky uvolňují (nebo vyvrhují) větší molekuly či struktury (obecně látky, které nejsou schopny samostatného prostupu přes plazmatickou membránu) do svého okolí. Na druhé straně se molekuly vypuštěné látky naváží na receptory. Poté se aktivují postsynaptické receptory a vytvoří se postsynaptický potenciál. Když je vytvořeno dostatečné množství těchto potenciálů (překročí-li kritickou mez), vytvoří akční potenciál, který je poslán dalšímu neuronu.[8]

1.1.3 Vznik mozkové aktivity

Při vytvoření akčního potenciálu (vzruchu) neuronů je indukováno elektromagnetické potenciálové pole, které lze registrovat i na povrchu lebky s elektrickým napětím od 5 do 200 μV . Aktivitu všech neuronů dohromady nazýváme mozkovou aktivitou. Lokalizací a analýzou těchto indukovaných potenciálů lze získat informace o procesech v mozku pacienta. Tato metoda se nazývá elektroencefalografie a v kapitole 1.2 *Měření mozkové aktivity* je podrobně popsána.

Lidský mozek je rozlišován na čtyři základní oblasti, z nichž každá má své specifické funkce. V následujícím textu budou uvedeny pro představu jen některé funkce, za které jsou jednotlivé části zodpovědné. Podrobným popisem jednotlivých částí mozku se zde zabývat nebudeme, jednak z důvodu překročení rozsahu této práce a také proto, že tato oblast není zatím dostatečně probádaná. [5]

- **Frontální lalok** - část mozku v oblasti čela, odtud název (latinsky čelo = *fronts*). Dle Blažka je zde umístěno centrum psychické aktivity a kreativity. Dále tato část kooperuje s ostatními laloky při logickém uvažování a řešení strukturovaných úloh. [5]
- **Parientální lalok** - Nachází se v horní části lebky a často bývá též označován jako lalok temenní. Zpracovává nervové impulsy smyslového vnímání, konkrétně například hmat, čich, vnímání teploty a bolesti. [5]
- **Temporální lalok** - často označován jako spánkový lalok se podílí například na paměťových a jazykových funkcích a zpracovává jazykové podmínky. [5]
- **Okcipitální lalok** - českým názvem týlní lalok se nachází v zadní části lebky a obsahuje centra mozku pro rozpoznávání a zpracování vizuálních podmětů. [5]

1.2 Měření mozkové aktivity

Pro diagnostiku stavu mozkové aktivity a její výzkum, je nutné využít některou z metod pro měření mozkové aktivity. V této kapitole jsou uvedeny tři přístupy těchto měření. První je elektroencefalografie, dále metody spadající do kategorie snímování a implantované technologie. Největší prostor bude poskytnutý právě elektroencefalografii, protože bude využita v praktické části této diplomové práce.

1.2.1 Elektroencefalografie

Elektroencefalografie, zkratkou EEG, je metodou snímání činnosti jednotlivých částí mozku, při které je detekována elektrická aktivita, která je způsobená aktivitou neuronů nebo má jiný biologický původ (viz. podkapitola Artefakty). První popis metody publikoval v roce 1929 německý neurolog a psychiatr Johannes Berger. Na pořizovaných záznamech popsal základní křivky alfa a beta (viz níže), které byly později rozšířeny o další tři. Dnes je EEG jedna z nejrozšířenějších diagnostických metod v neurologické praxi.

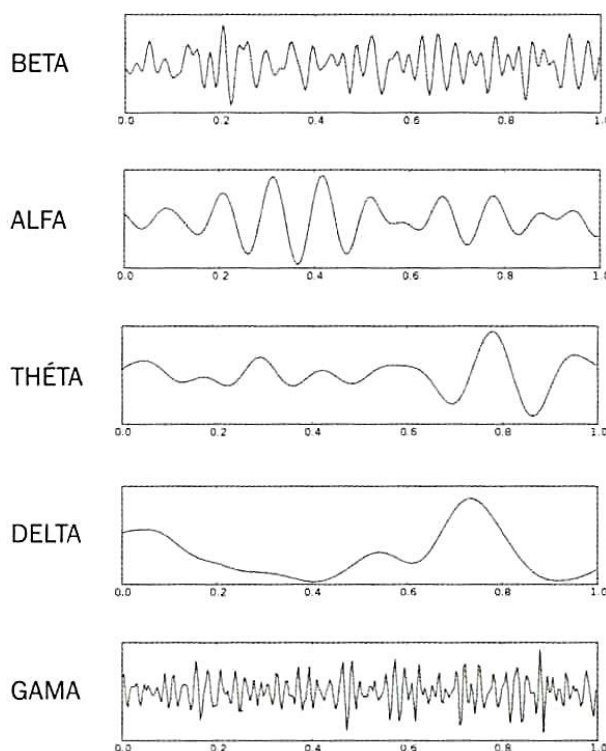
Princip metody je založen na snímání elektrických potenciálů většinou z povrchu hlavy použitím soustavy rozmístěných elektrod, které jsou součástí zařízení s názvem elektroencefalograf. Pořízený záznam se nazývá elektroencefalogram, který je buď lékařem nebo výzkumným pracovníkem dále analyzován. [5]

Typy EEG vln

Činností mozku je vytvářeno elektromagnetické pole s určitou frekvencí. Po dlouholetých experimentech bylo prokázáno, že frekvence tohoto elektromagnetického pole je implikováno s právě probíhajícími mentálními procesy. Odborná veřejnost rozlišuje pět základních frekvenčních rozsahů.

- **Alfa vlny** (8-12 Hz) - s amplitudou do $50 \mu V$. Převládá stav bez jakéhokoliv duševního i tělesného napětí, stav bdělého odpočinku bez pocitu ospalosti. Vyskytuje se přirozeně při senzorickém odpočinku (tzn. zavřené oči), relaxaci či meditaci. Má se za to, že na hranici alfa vln je daný jedinec ve své činnosti nejproduktivnější a nej kreativnější. [5]
- **Beta vlny** (12-30 Hz) - s amplitudou $10 \mu V$, která ale během spánku může být zvýšená. Jedná se o vlny, které se vyskytují v normálním bdělém stavu, počínaje ospalou bdělostí až po soustředěnou činnost. Při zvýšené frekvenci jedinec už zažívá stresové situace, nebo stavy napětí. Dále se dělí na tyto kategorie: [5]

- **SRM** (12-15 Hz) - klidná bdělost.
 - **Beta 1** (15-18 Hz) - stav soustředěnosti, pozornosti, běžná práce.
 - **Beta 2** (18-30 Hz) - jedinec zažívá stres, úzkost, napětí.
- **Theta** (4-7 Hz) - s amplitudou nad $50 \mu V$. Vyskytuje se během nízké formě REM (Rapid Eye Movement) spánku, hluboké relaxaci a meditaci. Jedinec se nachází ve stavu nevědomí . U dětí je výskyt těchto vln zcela běžný. U dospělých může být patologický, např. při poruše pozornosti.[5]
 - **Delta** (0,5-4 Hz) - s amplitudou do $100 \mu V$. Hluboký spánek a nevědomí. U dospělých je v bdělém stavu patologický.
 - **Gama** (nad 30 Hz) - hluboká koncentrace. Podle obecně rozšířené teorie, gama mozkové vlny vstupují do tvorby unifikovaného vnímání vědomí (problém vazeb). Nicméně není zde jasná vědecká shoda nad touto teorií.[5]



Obrázek 1.3: Mozkové vlny
[4]

Elektroencefalograf

Elektroencefalograf je zařízení, které je složeno z pěti částí. Snímací elektrody, předzesilovače, diferenciálního zesilovače, filtru, registračního zařízení.

Snímací elektrody - elektrody se rozdělují na povrchové, podpovrchové a mikroelektrody. Podpovrchové elektrody se zavádějí po kůži nebo se implantují přímo do těla. Povrchové elektrody se dělí na plovoucí, které se používají v kombinaci s vodivým gelem a na suché, které se dále dělí na izolované a neizolované. Mikroelektrody se používají pro měření potenciálu konkrétního neuronu. Na povrchu elektrody dochází ke změně vodivosti. Organismus je definován jako vodič druhé třídy (el. proud je vytvářen pohybem nabitých iontů) a ve vodiči od elektrody je el. proud tvořen pohybem volných elektronů (vodič první třídy). [5]

Při snímání mozkové aktivity mozku elektroda generuje spojitou funkci povrchových elektrických potenciálů v čase. Elektrody se často vyrábějí ze stříbra s vrstvou chloridu stříbritého. Dále pak z nerezové oceli a jiných slitin s obsahem cínu.

Předzesilovač - zařízení zesilující signál získaný elektrodou z μV na mV úroveň. [5]

Diferenciální zesilovač - je propojen s předzesilovačem stíněným kabelem. Má primárně dvě funkce. Tou první je zesílení signálu, obvykle tisíckrát a tou druhou je odstranění rušivých signálů tzv. zesílením difference. Má dva vstupy, jeden vstup pro signál původně ze snímací elektrody a druhý vstup pro signál z tzv. referenční elektrody, která je umístěna obvykle na ušním lalůčku. Oba vstupy se od sebe odečtou a výsledný rozdíl (difference) je poté zesílen. Myšlenka je taková, že na obou vstupech je naindukováno stejně velké rušení. Po jeho odečtení se získá signál, který je okolním rušením ovlivněn minimálně. [5]

Filtr - již zesílený signál je pomocí dolní a horní propustě filtrován tak, aby bylo propuštěno jen takové frekvenční pásmo, které je relevantní. Typicky se jedná o 0,5 - 70 Hz.

Registrační zařízení - EEG signál je nutné z digitalizovat a poté je možné signál zobrazit v grafickém rozhraní, anebo ukládat na paměťové médium.

Artefakty

Za artefakty jsou označovány všechny nežádoucí signály, které byly elektrodami zaznamenány. Tzn. že mají jiný původ než v neurální aktivitě. Rozlišují se dva typy artefaktů:

Biologické artefakty - mají biologický původ v těle jedince. Jedná o záznam srdeční aktivity, pohybů očí a obecně svalové aktivity. Srdeční stahy se promítnou v záznamu v podobě signálu s malou amplitudou, který připomíná tzv. hrot a může být jedním z příznaků epilepsie. Tyto artefakty lze eliminovat tím, že v průběhu EEG se pořizuje i EKG (Elektrokardiogram) a signály se poté od sebe odečtou. Dále artefakty způsobené pohybem očí vznikají uspořádáním samotných očí, které vytváří elektrický dipól. Při pohybu oka jsou zaznamenávány výrazné amplitudy. Elektrody umístěné na frontální části lebky jsou těmito artefakty zatíženy nejvíc. Artefakty lze ze záznamu odstranit záznamem aktivity očí a následným odečtením. Svalové artefakty jsou zapříčiněny svalovými kontrakcemi. Jedná se o velmi časté artefakty biologického původu vůbec a největší podíl na nich mají svaly okolo očí, svaly čelistí a jazyka. Eliminovat tyto artefakty je možné uvolněním těchto svalů v průběhu měření. [5]

Technické artefakty - jedná se o artefakty technického původu se zdrojem v okolí. Dříve se EEG zařízení umísťovala do Faradayovy klece, tak aby byl jejich efekt odstíněn. Dnes se toto řešení nepoužívá hlavně z ekonomických i praktických důvodů a je kladen důraz spíše na správnou konstrukci zařízení a umístění elektrod na těle pacienta. Technické artefakty se dělí na tři základní kategorie [5]:

- Síťový šum – artefakt o kmitočtu elektrické rozvodné sítě a jeho násobky. Původce je samotné EEG zařízení i ostatní el. zařízení zapojená do sítě. Vzniká zejména, když je špatně provedeno uzemnění.
- Šum přístroje – každá elektronická součástka, kterou protéká el. proud, generuje šum (např. vlivem tepla, nerovnoměrného průchodu proudu). V případě přístroje EEG mají největší vliv na generaci šumu jeho vstupní obvody. Protože je šum součástí signálem náhodným, nelze jeho vliv eliminovat.
- Elektrostatické potenciály – příčinou je často špatný kontakt elektrody s měřeným jedincem, nebo změna parametrů elektrod. Tyto artefakty se často vyskytují, pokud se v blízkosti pohybuje elektrostaticky nabitý materiál.

1.2.2 Snímkování

V této části budou krátce popsány tři základní metody používané pro tzv. snímkování mozku člověka. Jedná se o metody, které dokáží vytvořit řez mozku, ale i trojrozměrný model.

Funkční magnetická rezonance - fMRI

Jedná se moderní zobrazovací metodu sloužící k funkčnímu zobrazování mozku, přesněji k mapování mozkové aktivity, která je odezvou na vnější či vnitřní podnět. Díky rostoucímu výkonu výpočetních technologií a zdokonalování statistických metod se rozvíjí metoda fMRI jako nástroj pro vizualizaci struktur mozku zapojených do procesů vnímání, řízení motoriky a myšlení, tedy kognitivních funkcí mozku. Od standardní magnetické rezonance se liší tím, že touto metodou je možné detekovat dynamické změny signálu způsobené lokálním kolísáním poměru oxyhemoglobinu a deoxyhemoglobinu v závislosti na neuronální aktivitě.[18]

Výpočetní tomografie - CT

Metoda CT (Computed Tomography) je radiologickým vyšetřením, které s využitím rentgenového záření umožňuje neinvazivní vyšetření zobrazující vnitřní orgány a tkáně člověka. Lze detekovat aktivizaci mozku pro vytváření nových neurálních spojení stejně jako komplikace spojené s trvalým poškozením mozku.[18]

Pozitronová emisní tomografie - PET

PET zobrazuje tkáně tím, že rozlišuje jejich schopnost absorbovat pacientu podanou radioaktivní látku. Touto látkou může být modifikovaná molekula glukózy. Vyšetření je často používáno ve výzkumu.[18]

1.2.3 Implantované technologie

Neurozařízení je používáno k monitorování nebo regulaci mozkové aktivity. K monitorování jsou použity elektrody, které jsou implantovány přímo pod lebeční kost. Pro stimulaci jsou zaváděny hloubkové mozkové stimulatory, které stimulují konkrétní mozkové oblasti. Toto je disciplína nového vědního oboru s názvem neuromodulace, který využívá neurozařízení a oboru neurochemie. Hlavním tezí, na které je tento obor postaven je, že mozek může být regulován za použití ovlivňování metabolickými či fyziologickými faktory a elektrické stimulace.[18]

1.3 Manipulace s naměřenými daty - GDPR

Naměřená data je nutné ukládat efektivním a bezpečným způsobem, tak aby byla přístupná jen pověřeným osobám a zároveň byla zachována práva subjektu.

Nařízení platné pro celou Evropskou unii z roku 2016 s názvem GDPR bude uvedeno v platnost 25. května 2018. Jedná se o nejvýznamnější nařízení, které chrání osobní údaje, za posledních 20 let. Motivem pro něj je zvýšení ochrany osobních údajů v Evropské unii. Celý dokument obsahuje 99 článků, které v českém jazyce čítají 88 stran. Jedná se tedy o skutečně rozsáhlou právní regulaci. [15]

GDPR definuje tyto role[14]:

- **Správce** - fyzická či právnická osoba, orgán veřejné moci, agentura nebo jiný subjekt, který určuje sám nebo společně s jinými subjekty smysl a způsob zpracování osobních údajů.
- **Zpracovatel** - fyzická či právnická osoba, orgán veřejné moci, agentura nebo jiný subjekt, který zpracovává osobní údaje, kterými disponuje správce.
- **Subjekt** - fyzická osoba, která je původcem osobních údajů. Nemůže se jednat o právnickou osobu.

GDPR je definuje dva nové přístupy. Princip odpovědnosti správce a přístup založeném na riziku. Princip odpovědnosti správce znamená, že správce je povinen dodržovat zásady pro zpracování osobních údajů a zároveň tento soulad doložit. Princip založený na riziku ve volnějším výkladu znamená, že správce již od počátku zpracování osobních údajů musí brát v potaz rozsah, povahu, účel a kontext zpracování a zohlednit pravděpodobná rizika porušení práv a svobod fyzických osob, například z pohledu zabezpečení systému.[14]

V případě shromažďování osobních údajů pro vědecké účely GDPR obsahuje výjimku, že je nutný pouze informovaný souhlas, který je v souladu s uznávanými etickými normami pro vědecký výzkum. Fyzická osoba má právo na přístup ke svým údajům a informaci, za jakým účelem jsou osobní údaje zpracovávány. Dále má subjekt právo na opravu osobních údajů a "právo být zapomenut".[14]

2 Kognitivní funkce mozku

Kognitivní funkce, neboli funkce poznávací jsou mechanismy mozku, které umožňují jedinci poznávání okolního hmotného světa, ale i interakce na úrovni sociální, vnitřní procesy jako je například plánování či sebereflexe nebo uvažování nad svými myšlenkovými pochody jako takovými. Výsledkem tohoto aparátu je tedy myšlení, na jehož základě vzniká cílená činnost.

Jinými slovy toto popisuje krátká, ale výstižná definice:

„Vědomé jednání člověka je řízeno kognitivními procesy.“[17]

Věta sice nevysvětluje o jaké procesy se jedná a co jsou to kognitivní procesy, ale objasňuje, za co jsou zodpovědné. Jednání každého jedince je výstupem soustavy hierarchicky rozložených vzájemně souvisejících procesů, které se v našem mozku odehrávají neustále a jsou podmíněny určitou úrovní vědomí. Kognitivní funkce, dále označovány zkratkou *KF*, jsou tedy základní a nedílnou součástí našeho mozku a zásadním způsobem ovlivňují naše životy.

2.1 Rozdělení

KF se rozdělují do tří kategorií. Jsou to kognice základní, vyšší kognitivní funkce a metakognitivní schopnosti. Mezi základní patří paměť, pozornost, vnímání a orientace. Vyšší kognitivní funkce se skládají z myšlení a exekutivní funkce a mezi metakognitivní schopnosti se řadí přemýšlení a uvažování o vlastních myšlenkových pochodech.[11]

2.1.1 Základní kognitivní funkce

„Základní kognitivní funkce jsou podmíněné neuroanatomickou a fyziologickou integritou mozku, mají vliv na vyšší kognitivní funkce a ovlivňují také metakognitivní procesy.“[11]

Paměť

Jedná se o schopnost, psychickou funkci, která je zodpovědná za přijímání informací, jejich ukládání a pozdější vybavování. Kvalita funkce se liší u každého jedince a je možné ji více či méně trénovat (viz podkapitola 2.4 Trénink kognitivních funkcí). Paměť je ovlivňována jinými *KF*, například myšlením.

V mozku jsou udržovány tzv. paměťové stopy, které jsou v průběhu života vytvářeny. Vytváření těchto stop je přímo závislé na elasticitě mozku, která s rostoucím věkem klesá. Z toho důvodu dítě/mladý člověk je schopen učit se novým informacím v průměru mnohem úspěšněji než člověk vyššího věku. Pokud nejsou tyto paměťové stopy ožívovány (informace nejsou využívány opakovaně), tak stopa postupně slábne, tedy není vyživovaná a postupem času odumírá, až úplně vymizí. Tomuto procesu se říká zapomínání. Jedná se o důležitou složku paměťové funkce, která v první řadě chrání mozek před přehlcením informací a zajišťuje tak kapacitu pro další nové informace.[19]

Zapomínání lze bohužel přímo ovlivnit jen posilováním paměťových stop. Zapomínáme tak i informace, vjemy či zážitky, které bychom chtěli uchovat.

Proces uchovávání si informací lze chronologicky dle Klucké a Volfové rozdělit takto[10]:

- vštípení
- uchování
- vybavení

Jiní autoři, Jiří a Pavel Pavlovský, druhý bod uchování rozdělují na retenci a konzervaci. Retence je schopnost udržet paměťové obsahy. Konzervace je to samé, avšak s tím rozdílem, že jsou informace uchovávány beze změny. [16]

Paměť se dělí dle následujících kategorií:

Senzorická paměť - neboli ultrakrátká či okamžitá paměť je druh paměti, který uchovává vjemy získané ze všech pěti smyslů po dobu několika milisekund. Slouží jako filtr všech informací, které z okolí na člověka působí. Selekcí, kterou je vybudena naše pozornost, je s relevantními informacemi dále nakládáno.[12]

Krátkodobá paměť - též označována jako pracovní je paměť, která přímo navazuje na paměť senzorickou. Pokud informace zaznamenaná pamětí senzorickou vzbudila pozornost jedince, přechází informace do paměti krátkodobé, kde je uchovávána nejdéle v řádu hodin. Její zajímavou vlastností je tvorba asociací, která rapidně zvyšuje množství zapamatovatelných informací. Jedná se ve zkratce o techniku, kdy se jednotlivé informace uspořádají do určitého systému. Ke každé informaci se uměle připojí další význam, tak aby lépe zapadala do celku.[12]

Dlouhodobá - informace v ní setrvávají po delší dobu, nebo jsou zde uloženy nastálo. Dlouhodobou paměť členíme na paměť implicitní, která je zodpovědná za naučení se různých dovedností, zvyků bez vědomého vybavování, a paměť explicitní, kterou si lze vybavit fakta či vzpomínky vědomě. [10]

Dlouhodobá paměť je výsledkem procesu zaznamenávání nových informací. K zapamatování vede přeskupování, organizování, vizualizování a některé opakování poznatků. Cílem všech těchto kroků je rozšíření rozsahu paměti a týká se všech vjemů, které se uchovávají v paměti déle než tři minuty a nepodlehly interferenci. Dlouhodobá paměť obsahuje jak současné, tak dávno proběhlé události. Co se týče současných událostí, je dlouhodobá paměť velmi křehká. [12]

Pozornost

Pozornost je zaměřenost a soustředěnost duševní činnosti na konkrétní děj nebo objekt. Pozornost je předpokladem pro smyslový vjem a zapojení dalších *KF*. Výše je uvedeno, že pozornost je v jistém smyslu propojením mezi pamětí sensorickou a krátkodobou. Na základě vlastností informace získané smyslovými vjemy je vybudována pozornost, která umožňuje další nakládání s touto informací. Upřednostňuje informace, které jsou více či méně relevantní před informacemi nedůležitými, kterých je v okolním prostředí velké množství. V tomto případě se jedná o pozornost selektivní. Další kategorie jsou popsány v dalším odstavci.[7]

Existuje uznávaný model s názvem Sohlberg a Mateer, který se skládá z čtyř kategorií[7]:

- **Střídavá pozornost** - tento druh pozornosti umožňuje plynulý přechod z jedné činnosti na druhou.
- **Vytrvalá pozornost** - kategorie často označována slovy pozornost, koncentrace nebo soustředění je schopností věnovat se danému problému, jevu či objektu dostatečně dlouho na to, aby ho člověk pochopil a zapamatoval si jej.[13]
- **Selektivní pozornost** - lze ji rozdělit ještě na dvě podkategorie, a to na filtraci rušivých podmětů a na výběr těch potřebných. Na jedné straně se potlačují například nepodstatné zrakové vjemy jako jsou vločky padajícího sněhu a na straně druhé jsou upřednostňovány vjemy jiné, například vyzvánění mobilního telefonu.

- **Rozdělená pozornost** - také difúzní či rozptýlená umožňuje provádět dva nebo více různých úkonů najednou.

Orientace

Orientace je schopnost, která poskytuje uvědomění si vztahů v prostoru, času a ve vlastní osobě. Silně souvisí jak s vjemy ze smyslů, tak i s pamětí.[11]

Vnímání

Vnímání (též percepce) zachycuje to, co v daný okamžik působí na smysly. Informuje o vnějším (zima, horko) i vnitřním světě (např. bolest). Vnímání je subjektivním odrazem objektivní reality. Poskytuje nám představu o okolním světě, která také ovlivňuje zkušenosti, očekávání a daný kontext v jakém jsou informace přijímány a zpracovány. [11]

2.1.2 Vyšší kognitivní funkce

Vyšší *KF* Krivošíková definuje následovně:

„Kognitivní funkce jsou výsledkem komplexních a dynamických interakcí mezi strukturami mozku, které tvoří funkční systém. Jsou závislé na neporušených základních kognitivních funkcích.“[11]

Myšlení

Myšlení je proces, díky němuž lze ze známých faktů vyvodit závěry a možné důsledky. K tomu využívá metod řazení, kategorizace, dedukce, řešení problémů a formování pojmů. Řazením lze informace uspořádat do správného pořadí například chronologicky nebo dle priorit. Dedukcí získáváme závěry, které jsou odvozeny logickými postupy ze známých faktů. Kategorizace seskupuje fakta nebo myšlenky podle specifických charakteristik a schopnost formování pojmů analyzuje vztahy mezi objekty.[11]

Myšlení je spojováno s čelním lalokem mozku a slouží k porozumění a orientaci ve vnějším i vnitřním světě, kdy na jejich základě můžeme odvozovat závěry a postupy pro danou situaci.

Exekutivní funkce

Exekutivní funkce obsahují řešení problémů a pravidla sociálního fungování. Dle následujících sedmi tříd exekutivní funkce pomáhají uskutečňovat aktivity[10]:

- **Sebeuvědomění** - průběžné vytváření si náhledu na sebe samého.
- **Stanovení cílů** - schopnost vytyčit si cíle, které jsou reálné a beroucí v potaz naše vlastní omezení či zkušenosti.
- **Iniciace** - jedinec je schopný začít vykonávat aktivitu bez vnějšího podnětu a provádí jednotlivé kroky, které vedou k jejímu úspěšnému dokončení.
- **Inhibice** - potlačení nebo omezení nevhodného chování či myšlenky.
- **Plánování** - vytvoření strukturovaného postupu, které vede k úspěšnému provedení aktivity.
- **Sebehodnocení a sledování** - kritika a průběžné sledování sebe samého. Na základě získaných poznatků, je možné provést korekce svých aktivit či myšlenek.
- **Řešení problémů** - zhodnocení dané situace, předvídat možná rizika a vytváření možných řešení.

2.1.3 Metakognitivní schopnosti

Metakognice je schopnost odhadnout úroveň i možnosti vlastních kompetencí. Vychází ze znalostí a zkušeností s poznávacími funkcemi, vlastními i jiných lidí.[21]

Metakognice je zaměřena jak na orientaci, tj. poznání a porozumění, tak na regulaci a kontrolu poznávacích aktivit. To znamená, že se projevuje i praktickým využitím těchto poznatků. Je to schopnost posoudit problém a jeho obtížnost, vhodnost použití určité strategie a zvolit variantu, kterou lze považovat vzhledem k situaci i vlastním možnostem za adekvátní. Rozvoj metakognice se projevuje vzrůstem přesnosti takového odhadu a flexibility reagování, tj. hledání uspokojivého způsobu řešení.[21]

2.2 Kognitivní porucha

Kognitivní porucha může být způsobena širokou škálou postižení, kterou lze definovat takto:

„Porucha kognitivních funkcí je postižení korových funkcí: paměť, všeobecné zpracování informací, chápání souvislostí, abstraktní a logické myšlení a uvažování, řešení problémů, schopnost učení, rozhodování, plánování a organizování činností, motivace, poznávání, používání předmětů a orientace v prostoru a čase. Mohou být postiženy korové činnosti – faktické poruchy, dysgrafie, dyslexie, dyskalkulie, poruchy praxe (naučených pohybových dovedností, používání předmětů, pohybových stereotypů i konstrukčních schopností) či smyslové agnozie (např. neschopnost rozeznávat obličej – prozopagnozie). Tyto poruchy se mohou vyskytovat izolovaně nebo postihovat současně více funkcí - globální kognitivní porucha. Rozsah i závažnost postižení mohou být různé a bývají doprovázeny změnami/poruchami osobnosti, afektivity a chování.“[20]

2.3 Metody hodnocení kognitivních funkcí

Praxi se pro určení stavu kognitivních funkcí používá celosvětově velké množství více či méně specializovaných testů. Pro účely této práce bude uvedeno několik testů, které využívá AD centrum, které je součástí Psychiatrického centra v Praze a zabývá se kognitivními poruchami na úrovni výzkumu i klinické praxe.

Za hlavní test považují Montrealský kognitivní test. Další alternativou je Sedmiminutový screeningový test a Mini-Mental State Examination.[6]

2.3.1 Mini-Mental State Examination (MMSE)

Mini-Mental State Examination (MMSE) je dosud nejpoužívanější metodou, která se používá k určení stavu celkových kognitivních funkcí a k detekci demence i s určením míry postižení.[6]

Jedná se o jeden z prvních testů a velmi rychle se rozšířil do klinické praxe, výzkumu po celém světě. Dá se použít i jako test referenční, se kterým se zohledňují různé jiné metriky (další testy, vyšetření, stádia onemocnění, léčba). Test se skládá z několika časově nenáročných úkolů. Lze zhodnotit více kognitivních funkcí, výsledky jsou ale pouze orientační.[6]

Pro předpis léčby kognitivních funkcí z prostředků všeobecného zdravotního pojištění se v ČR zohledňuje bodové hodnocení právě tohoto testu. Na

test se vztahují autorská práva a jeho používání je zpoplatněno.[6]

Test se skládá z deseti částí, kterými se hodnotí pacientova orientace v čase a prostoru, krátkodobá paměť, pozornost, psaní, čtení, řeč, početní schopnost a konstrukčně-praktické dovednosti. Testem lze rozeznávat středně těžkou demenci a normální stárnutí. Z důvodu nenáročnosti úkolů je obtížné s tímto testem rozpoznat pouze mírnou kognitivní poruchu. Časová náročnost je mezi 5 a 10 minutami. Nejvyšší skóre v MMSE je 30 bodů. Čím vyšší skóre, tím lepší výsledek. Pokud pacient získá 24 a méně bodů je možné, že trpí nějakou formou demence.[6]

Dnes je tento test považován jako výborný nástroj pro detekci demence, konkrétně u pacientů, kteří již nejsou soběstační. Nehodí se však pro záchyt mírných deficitů nebo lehkých kognitivních poruch. Jak již byl uvedeno, jeho obtížnost je příliš nízká a zvládají ho i jedinci s mírnou demencí.[6]

2.3.2 Montrealský kognitivní test (MoCA-CZ1)

Montrealský kognitivní test (*MoCA-CZ*) se skládá z jedenácti relativně náročných zkoušek k rozpoznání počínajících nebo lehkých kognitivních deficitů. Jedná se tedy o test k detekci mírné kognitivní poruchy. Konkrétně poruchy řečových schopností, zrakově-konstrukční schopností, pozornost a časovou nebo místní orientaci. Dostatečně prověřuje i exekutivních schopností a paměť. Používá se pro zhodnocení kognitivních funkcí v přijatelném čase jednoduchou administrací. Test *MoCA-CZ* je celosvětově rozšířen a existuje v mnoha jazykových verzích. Pro zjednodušení administrace v AD centru upravili formulář, který na rozdíl od originálu, disponuje podrobnými instrukcemi a návodem. Testovat a vyhodnocovat výsledky testovaného subjektu pak může uskutečňovat i méně zasvěcená osoba. Test je vhodné použít u mírné kognitivní poruchy, tzn. kdy jsou potíže provázeny se stále zachovanou soběstačností. Dále u počínající nebo mírné demence, především u Alzheimerovy choroby, u osob s podezřením na kognitivní poruchu, u výsledků z *MMSE* mezi 25 a 30 body a u vzdělaných a relativně mladších seniorů. Test je naopak nevhodný u osob se střední nebo pokročilou demencí. Ti ho pak zvládají s velkými obtížemi. *MoCa* poskytuje rozumný kompromis při kognitivním testování, protože není tak snadný jako *MMSE* a není tak dlouhý jako například Addenbrookský kognitivní test. Tento test AD Centrum považuje jako vhodný k širšímu používání v České Republice.[6]

2.3.3 Sedmiminutový screeningový test (7MST)

Krátký test nazvaný Sedmiminutový screeningový test (7MST) byl vyvinut s cílem vylepšit detekci demence za pomoci nových neuropsychologických poznatků. Test je složen ze čtyř částí hodnotící kognitivní oblasti, konkrétně časové orientace, paměti, zrakově-prostorové schopnosti a slovní zásoby. Výrazně se od ostatních testů liší tím, že výsledky jsou rovnou přepočítávány do pravděpodobnosti výskytu Alzheimerovy nemoci u dané osoby.[6]

Tímto testem se měří vizuální paměť, sémantická paměť, exekutivní funkce, zrakově-prostorové funkce, časová orientace.[6]

Test je vhodné použít u mírné poruchy kognitivních funkcí, u počínající nebo mírné demence, především u Alzheimerovy choroby, u osob s podezřením na kognitivní poruchu, u výsledků z *MMSE* mezi 25 a 30 body a u vzdělaných a relativně mladších seniorů. Důmyslně koncipovaná baterie čtyř zkoušek a jejich vyhodnocení je užitečným nástrojem k odhalení časně Alzheimerovy nemoci. Navzdory tomu si test nezískal větší oblibu ve České Republice ani ve světě. Důvodem může být nutnost použití pomůcky ve formě stolního kroužkového kalendáře a vkládání mezivýsledků do složitého vzorce pro výpočet pravděpodobnosti Alzheimerovou nemoci. Zajímavým aspektem testu je možnost nejen zjišťovat prostorovou dezorientaci, ale také určovat kvantitativní stupeň této dezorientace.[6]

2.4 Trénink kognitivních funkcí

Pokud vynakládáme úsilí k zlepšení dopadů kognitivní poruchy, je důležité si určit cíl, kterého má být dosaženo jak ze strany terapeuta, tak ze strany klienta. Všechny kognitivní funkce je nutno si důkladně vyšetřit rozhovorem, dotazníkem, či standardizovaným testem. Obzvláště je důležitý důraz na vyšetření pozornosti, ve které je základ všech možných problémů. Trénink kognitivních funkcí u lidí po cévní mozkové příhodě by měl být dle hierarchického modelu zaměřen zejména na nácvik pozornosti a vědomí. [13]

TKF se ve světě využívá ve velké míře. V USA se využívá hlavně v logopedii, ve Velké Británii je to hlavně v oboru ergoterapie, ve Skandinávii je převážně využíván ve školství. Psychology obecně je využíván po celém světě.[13]

2.4.1 Pojmy kognitivní trénink, rehabilitace, neurorehabilitace

V odborné literatuře se nachází pojmy: trénink kognitivních funkcí, kognitivní rehabilitace a neurorehabilitace. [10]

Doslovné vysvětlení těchto pojmů je následující[10]:

- **Trénink kognitivních funkcí** - aktivní trénování a cvičení kognitivních funkcí u zdravých osob. Nelze tedy tréninkem zlepšit, či obnovit ztracené funkce, ale pouze aktivizovat a udržovat stávající schopnosti. Trénink slouží pouze jako prevence.
- **Kognitivní rehabilitace** - Opětovné získání ztracených, či poškozených funkcí.
- **Neurorehabilitace** - neurorehabilitace se týká lidí při těžkém poškození mozku. Do neurorehabilitace spadá fyzioterapie, ergoterapie, logopedie, či muzikoterapie a arteterapie. Je prováděna ihned po obnovení životních funkcí. V následné fázi léčby je prováděna kognitivní rehabilitace.

2.4.2 Vedoucí tréninku kognitivních funkcí

Nejčastěji jsou to právě psychologové, kteří vedou kognitivní trénink. Pro nedostatek času, ale začali vést trénink i odborníci z jiných profesí jako např. obor psychologie, sociologie, speciální pedagogika a ergoterapie. Ovšem při tréninku kognitivních funkcí jsou ve velké míře nápomocni lidé v okolí, kterým je personál, pečovatelé, příbuzní, někdy dokonce i zvířata mohou zahrát velkou úlohu v aktivitách.[10]

3 Komerční systémy pro kognitivní trénink

Po analýze byly nalezeny tři systémy, které komplexně testují a trénují kognitivní funkce uživatele. Jedná se o dva systémy ze zahraničí a jeden z České Republiky. Zde je jejich stručný popis:

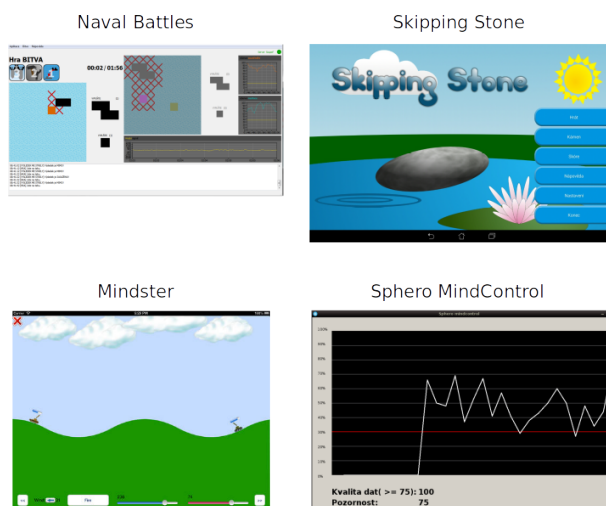
- **HappyNeuron** - systém poskytuje zdravotnickým pracovníkům specificky přizpůsobené kognitivní stimulační nástroje pro použití se svými klienty. Programy jsou implementovány tak, aby bylo možné přizpůsobit nastavení konkrétnímu uživateli v různých zázemí a zdravotních zařízeních. HappyNeuron nabízí 37 různých typů úkolů, které stimulují všechny oblasti poznávání. Jedná se o velmi vyspělý nástroj jak po technické stránce, tak i po odborné.
- **Lumosity** - systém obsahuje více než 50 kognitivních her, které jsou po grafické stránce velmi povedené. Společnost, která tento systém provozuje, spolupracuje s více než 100 výzkumnými pracovníky po celém světě, kteří mají volný přístup k interním nástrojům pro trénink mozku.
- **Mentem** - obsahuje 24 kognitivních her, které zlepšují paměť, pozornost či prostorovou orientaci. Nezisková společnost Mentem – brain training začala vyvíjet systém v roce 2013 a v roce 2014 spustili projekt Mentem.cz.

Výše zmíněné projekty jsou na vysoké úrovni. Všechny poskytují přehledné grafické rozhraní a vysoký počet kognitivních her, které jsou konzultovány a vyvíjeny s odborníky. Velkou nevýhodou je však zpoplatnění plnohodnotného používání systému a fakt, že se jedná o uzavřený produkt. Uzavřený v tom smyslu, že své kognitivní hry vyvíjí interně a není možné integrovat do systému jiné hry. Dále tyto systémy zatím neumožňují zapojit do tréninku další zařízení (například čelenka Neurosky). Sbíraná data tak neobsahují například EEG záznamy.

4 Hry pro kognitivní trénink na KIV

Na Katedře informatiky a výpočetní techniky, konkrétně v neuroinformatické laboratoři bylo vytvořeno několik kognitivních her, které využívají čelenku pro snímání EEG Mindwave mobile. Zde je krátký popis těchto her.

- **Naval Battles** - desková hra pro kognitivní trénink. konkrétně hra s známá jako Lodě. Hra je pro dva hráče, kteří se snaží zničit jako první flotilu toho druhého s využitím úrovně pozornosti a meditace.
- **MindSphero** - aplikace určená pro mikropočítač Raspberry PI, která umožňuje ovládat robotickou kouli Sphero 2.0 s využitím pozornosti a gamepadu.
- **Mindster** - na 2D hrací ploše dva uživatelé ovládají tanky. Každý z uživatelů má nasazenou čelenku Mindwave. Překročí-li pozornost určitou mez, tank může vystřelit.
- **Neuro-skipping stone** - aplikace využívá metriky pozornosti, kterou poskytuje čelenka MindWave mobile. Uživatelovým cílem je udržet kámen pomocí pozornosti nad vodou.



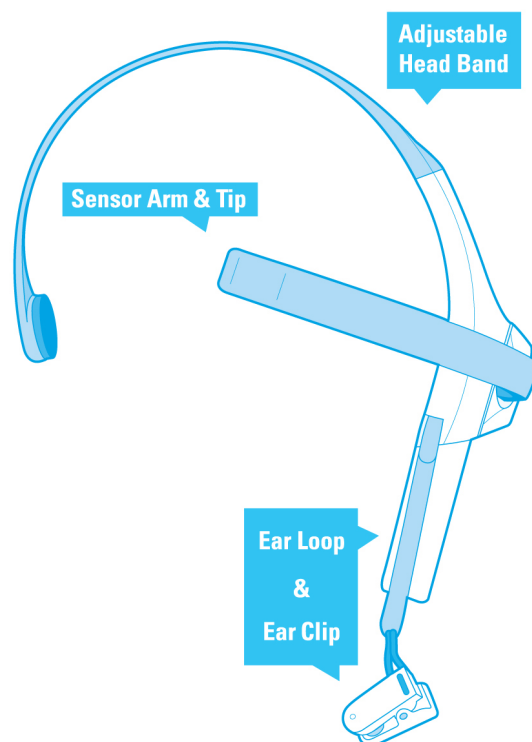
Obrázek 4.1: Hry pro kognitivní trénink

5 MindWave Mobile

MindWave Mobile je zařízení vyvinuté společností NeuroSky, díky kterému je možné snímat mozkovou aktivitu z frontální části mozku uživatele detekováním elektrických potenciálů mozku.

5.1 Popis zařízení

Zařízení je ve tvaru plastové čelenky, která je umísťována na hlavu. Mindwave Mobile disponuje 2 elektrodami. Hlavní elektroda je situována na frontální část hlavy na levou stranu čela. Další elektroda se umísťuje na levý ušní lalůček. Tato elektroda plní funkci referenční elektrody. Elektrické napájení poskytuje AAA baterie, díky které je možné využívat čelenku po dobu 8-10 hodin. MindWave Mobile se připojuje k jinému zařízení technologií Bluetooth. Může se jednat o stolní počítač nebo i mobilní zařízení.[1]



Obrázek 5.1: Schéma MindWave Mobile[1]

5.2 Specifikace

Vlastnosti Mindwave Mobile jsou shrnuté v následující tabulce:

Výrobce	NeuroSky Inc.
Uvedení na trh	2011
Cena[USD]	\$79.99
Váha	90 [g]
Napájení	1xAAA baterie
Výdrž	8-10 [h]
Komunikace	bluetooth 2.0
Maximální výkon	50 [mW]
Radio frekvenční rozsah	2.420-2.471 [GHz]
Radio frek. přenosová rychlost	250 [kbit/s]
Dosah vysílače	10 [m]
Modulační rychlost	57 600 [Baud]
Max. rozsah vstupního EEG	1 [mV] pk-pk
Modul pro zpracování signálu	TGAM1

Tabulka 5.1: Specifikace Mindwave Mobile[1]

5.3 ThinkGear modul - TGAM1

Spolu s elektrodami je zásadní součástí zařízení ThingGear TGAM1 ASIC modul, který získaná (naměřená) data biologického původu zpracuje do cílových datových toků. Překážkou je problém s rozlišováním mezi mozkovou aktivitou a šumem, tedy artefakty jakéhokoliv původu. Rušení je digitálně filtrováno a hrubá data elektrických potenciálů jsou zesílena a zpracována do cílových výstupů. Modul dále disponuje algoritmy emocionálních stavů, tzv. eSense, které jsou uloženy v paměti čipu. Tyto algoritmy byly vyvinuty laboratoří NeuroSky Inc. ve spolupráci s výzkumnými institucemi. Modul TGAM1 má rozměry 2,79 cm x 1,52 cm x 0,25 cm, váží 130 mg a komunikuje rozhraním UART v modulačních rychlostech (1200, 9600 nebo 57600 [Baud]).[3]

ThinkGear čip poskytuje tyto datové proudy[1]:

- Hrubá naměřená data
- Úroveň kvality snímané mozkové aktivity
- Úroveň pozornosti
- Úroveň meditace
- EEG rytmy
- Detekce mrknutí

Praktická část - Cognitive Amp

6 Vize projektu

Kapitola 6 *Vize projektu* obsahuje popis problému a souhrn požadavků zadavatele a dalších zainteresovaných osob, na jehož základě bude vytvořen systém s názvem Cognitive Amp, zkratkou *CAMP*, který je předmětem této diplomové práce. Dále budou definována rizika a omezení projektu, která budou při návrhu architektury a implementaci zohledňována.

6.1 Popis problému a účelu systému

Při kognitivním tréninku jsou využívány mnohé techniky/hry, při kterých se terapeut spolu s klientem snaží dosáhnout stabilizace kvality kognitivních funkcí. S vývojem výpočetních technologií se stále častěji využívá programů, které nahrazují, ale dosud ne plně, tištěné testy a hry, např. klasické pexeso pro podporu paměti klienta. Práce například ergoterapeutů se tedy ulehčuje tím, že odpadá nutnost disponovat různorodými pomůckami a zároveň takto digitalizované hry mohou být interaktivnější a pro klienta zábavnější. Na Katedře informatiky a výpočetní techniky v rámci neuroinformatické laboratoře vzniklo několik mobilních aplikací právě pro tyto účely.

Spolu s vývojem těchto aplikací vznikla potřeba vytvořit platformu, která by vhodným způsobem poskytovala terapeutům tyto a v budoucnu vytvořené aplikace. Uživatel bude mít možnost se na jednom místě seznámit s vytvořenými aplikacemi a jednoduchou akcí si ji bude moci nainstalovat a spustit. Dále systém bude sloužit jako analytický nástroj pro terapeuty. Jeho pomocí bude možné sledovat různé statistické ukazatele vypočítávaných nad výsledky klientů.

6.2 Popis zainteresovaných stran

Během vytváření výše uvedeného systému budou zúčastněny tyto osoby:

- Zadavatel diplomové práce - Ing. Petr Brůha
- Odborníci na kognitivní trénink:
 - Dr.phil. Laco Gaál - neuropsycholog, programátor a autor rehabilitačního programu NEUROP-III
 - ergoterapeuté - Bc. Vladimíra Frouzová a Bc. Richard Frouz
- autor diplomové práce

6.3 Vize produktu

Jednou větou lze systém, dle zadání dip. práce, definovat jako systém pro správu a distribuci kognitivních her. Správou je myšlena plná kontrola nad hrami, dále označovanými jako aplikace, které je možné do systému zahrnout a distribucí poskytnutí těchto aplikací klientům, pokud možno co nejjednodušším způsobem. Vedle tyto dva pojmy je vhodné dosadit ještě analytický rozměr systému, který se zde nabízí. Hraním her totiž uživatel vytváří množství informací, které je možné uložit a vhodně je využít pro analýzu dosažených výsledků terapeutem nebo klientem samotným. Pokud aplikace sbírá nějaká specifická medicínská data, bude je možné v tomto systému bezpečně uchovávat a poté je například ve výzkumu využít.

Základní zjednodušený scénář použití vytvořeného systému bude vypadat následovně:

1. Terapeut v systému vytvoří uživatelskou skupinu a přidá do ní svého klienta.
2. Klient si nainstaluje konkrétní aplikaci a začne ji používat.
3. Terapeut nebo klient vzdáleně kontroluje dosažené výsledky.
4. Vytvořená data je možné exportovat pro další účely.

6.4 Seznam funkčních požadavků

Jedná se o základní funkční požadavky zadavatele, které byly definovány už v počátku plnění diplomové práce. Detailní vlastnosti požadovaného systému zde uvedeny nejsou.

Správa a distribuce aplikací

Správce systému, či jiný privilegovaný uživatel, bude mít možnost přidávat a odebírat aplikace, které budou do systému začleněny. Tyto aplikace budou moci odesílat získaná data do systému, kde budou uchovávána. Uživatel bude muset vynaložit minimální úsilí, aby mohl tyto aplikace využívat.

Vytváření a správa skupin uživatelů

Terapeut bude mít možnost vytvářet skupiny uživatelů, díky kterým se mu zpřístupní data o výsledcích jejich členů. Uživatel bude muset s tímto členstvím z důvodu bezpečnosti souhlasit. K potenciálně citlivým datům se dostanou pouze privilegované osoby.

Sběr a analýza dat

Aplikace budou generovat data, která bude možné v systému ukládat a analyzovat. Může se jednat i o medicínská data, např. data z EEG, která bude možné exportovat pro další využití.

Vytvoření aplikace pro správu skupin a distribuci aplikací pro kognitivní trénink

Další mobilní aplikace pro Android bude obsahovat funkce pro správu uživatelského členství ve skupinách. Bude se moci odhlásit ze skupiny nebo požádat o členství ve skupině. Dále bude aplikace zobrazovat seznam aplikací pro kognitivní trénink a odkazovat na jejich instalaci nebo spuštění.

Vytvoření aplikace pro kognitivní trénink s využitím čelenky Neurosky mobile

Mobilní aplikace pro Android bude jednoduchou kognitivní hrou, která bude využívat čelenku Mindwave Mobile. Po skončení jedné hry bude aplikace odesílat data (skóre, další metriky, obrázek a EEG data) do vzdáleného úložiště pro budoucí analýzu.

6.5 Seznam mimo-funkčních požadavků

Stejně jako v předchozí podkapitole jsou zde uvedeny jen základní mimo-funkční požadavky. Detailnější vlastnosti systému budou patrné z návrhu architektury, který byl dostatečně konzultován a upravován dle připomínek zadavatele.

Umožnění jednoduché integrace aplikací s vytvořeným systémem

Cílem je, aby systém disponoval co nejvyšším počtem aplikací, které budou moci uživatelé využívat. Některé aplikace pro kognitivní trénink jsou již vytvořené, ale většina teprve vznikne. Na vývoji těchto aplikací se bude podílet rozmanitá skupina vývojářů. Integrace aplikace se systémem *CAMP* proto musí být normalizovaná, aby se předešlo co nejvyššímu počtu chyb. Dále je nutné, aby proces integrace nebyl příliš časově náročný.

Zabezpečení

Protože systém bude uchovávat citlivá data, je nutné klást důraz na náležitá bezpečnostní opatření. Dostatečně zabezpečená musí být vzájemná komunikace jednotlivých částí systému.

6.6 Rizika projektu

Při definování vlastností systému *CAMP* jsme společně se zadavatelem dip. práce Ing. Petrem Brůhou odhadli potenciální rizika, která se mohou vyskytnout v průběhu plnění praktické části, ale i po nasazení systému do produkční verze. Jednotlivá rizika jsou seřazena sestupně dle priority.

- **Špatné zabezpečení systému** - systém bude pracovat s daty osobními, ale i s daty medicínskými. Proto je nutné, aby systém i jeho provozovatel, tedy neuroinformatická laboratoř, potažmo KIV, zohledňoval nařízení GDPR, které je uvedené v analytické části v podkapitole *1.3 Manipulace s naměřenými daty - GDPR*.
- **Nedostatečný počet aplikací v systému** - pro úspěch systému *CAMP* je nutné poskytnout uživatelům dostatečný počet aplikací, tak aby mohli trénovat konkrétní KF, se kterou mají potíže. Možností, jak tomuto riziku předejít, je několik. Nejdříve je možné připojit do systému katedrou již vytvořené aplikace, a poté lze vyvíjet velké množství dalších jednoduchých aplikací (např. personifikované pexeso nebo výpočet jednoduchých příkladů). V tomto ohledu samozřejmě záleží na obtížnosti integrace aplikace se systémem.
- **Nedostatečně přívětivé uživatelské rozhraní** - podstata rizika je taková, že pokud bude mít uživatel z používání systému špatný dojem, neboli špatnou "User Experience", motivace k dalšímu používání systému bude ohrožena.

- **Nedostatek času pro dostatečné testování** - toto riziko má mnoho společného s rizikem předchozím. Je nutné, aby výsledný produkt byl dostatečně otestován, tak aby ho bylo možné provozovat bez větších potíží, stejně jako jednoduše navázat na tuto práci a systém rozšiřovat.
- **Nedostatek času pro implementaci** - protože se jedná o potenciálně rozsáhlý systém, který může disponovat různorodými funkcemi, např. funkcí pro analýzu dat, je nutné klást v průběhu implementace důraz v první řadě na stěžejní části systému. Díky tomu bude pravděpodobněji, že ukončení projektu bude úspěšně a bude jej možné v budoucnosti dále rozšiřovat.

6.7 Omezení

Jako omezení lze definovat požadavek zadavatele, aby vytvořené mobilní aplikace byly pro platformu Android. Důvodem je velké rozšíření tohoto operačního systému a skutečnost, že v rámci neuroinformatické laboratoře již byly pro tuto platformu některé aplikace pro kognitivní trénink vyvinuty.

6.8 Konzultace s odborníky v oboru kognitivního tréninku

Při návrhu vznikajícího systému se uskutečnila konzultace s již zmíněnými odborníky v oboru kognitivního tréninku. Dr. Gaál je neuropsycholog a programátor. Klinickou praxí v oboru neuropsychologické rehabilitace se pohybuje už řadu let a je v této oblasti uznávaným odborníkem. Mimo mnohá jiná pracoviště pracoval zejména na klinice pro neurologickou rehabilitaci Bezirksklinikum Mainkofen a je autorem úspěšného rehabilitačního programu NEUROP-III, který obsahuje celou řadu kognitivních her, které v praxi pro rehabilitaci používá. Konzultace se zúčastnili i ergoterapeuté Bc. Vladimíra Frouzová a Bc. Richard Frouz.

Nejdříve jim byla představena vize navrhovaného systému, s některými architektonickými parametry. Všichni se shodli, že tento systém bude mít s jistotou využití. Na trhu sice existují vyspělé nástroje (viz kapitola 3 *Komerční systémy pro kognitivní trénink*), ale převážně v nich chybí reflexe aspektu kontroly rehabilitace odborníkem, tedy například ergoterapeutem. Systém *CAMP* toto umožňuje. Terapeut bude mít možnost seskupovat své pacienty do skupiny, kde jim bude moci doporučovat kognitivní hry. Výsledky každého svého pacienta pak lze zpětně analyzovat.

Dalším postřehem bylo, že je třeba vytvářet hry pro kognitivní trénink tak, aby bylo možné je pro každého pacienta personifikovat a zároveň klást důraz na emoční a kreativní rovinu.

7 Specifikace požadavků

Na základě předchozí kapitoly a dalších upřesňujících konzultací se zainteresovanými osobami byla vytvořena množina případů užití, na základě kterých poté bude navržena architektura systému.

7.1 Případy užití



Obrázek 7.1: Diagram případů užití

Výše uvedený diagram obsahuje čtyři uživatelské role a celkem sedm unikátních případů užití, které budou v následujícím textu popsány:

- **Správa uživatelů** - jedná se o funkci, kterou bude využívat pouze administrátor (správce) systému, díky které bude moci vytvářet, mazat a blokovat uživatele.
- **Správa uživatelských skupin** - terapeut má možnost vytvářet vlastní skupiny uživatelů (klientů), dále je upravovat a mazat. V této skupině lze i definovat jaké aplikace budou zahrnuty do statistických výpočtů, např. percentil výsledků.
- **Správa aplikací** - správce aplikací vytváří záznam o aplikaci, kterou bude chtít integrovat se systémem. Díky tomuto záznamu ji lze pak distribuovat všem uživatelům.
- **Analýza výsledků** - jedná se o zobrazení výsledků jednotlivých uživatelů, kterým bude mít terapeut přístup, tzn. bude členem jedné z jeho skupin. Výsledky jsou myšleny číselné hodnoty, které budou zaslány aplikací pro kognitivní trénink po ukončení cvičení. Konkrétně se jedná o dosažené skóre a další metriky. Dále se jedná o možnost zobrazení statistických ukazatelů, které jsou vypočítávány nad výsledky z aplikací výše popsanými. Podstatnou částí této funkce je zobrazení percentilu vztahového na všechny uživatele v systému, který lze filtrovat na základě věku a dosaženého vzdělání.
- **Analýza hrubých dat** - spolu s číselnými hodnotami má aplikace možnost zaslat během cvičení snímaná data například medicínského původu. Tato data lze použít se souhlasem původce dat pro další výzkum nebo pro terapii jako takovou.
- **Kognitivní trénink** - uživatel využívá některé z aplikací ze systému pro zlepšení nebo diagnostiku svých kognitivních schopností. Aplikace má možnost odeslat získaná data do systému *CAMP*.
- **Prohlížení vlastních dosažených výsledků** - každý klient má možnost si zobrazit všechna data, která jsou o něm v systému uchovávána. Bude tak mít možnost kontrolovat si své výsledky ze cvičení.

7.2 Aktéři

V diagramu Případu užití (Obrázek 7.1) z předchozí podkapitoly jsou zobrazeni čtyři aktéři. Ty lze v systému interpretovat jako uživatelské role. Jeden uživatel tak může mít více rolí. Například terapeut může zastávat roli klienta, tak aby si mohl plně vyzkoušet jednotlivé aplikace.

Roli správce bude mít pouze pověřená osoba, která se stará o správu celého systému. Roli správce aplikací musí zastávat taktéž pověřená osoba tak, aby byly přidávány jen ověřené a relevantní aplikace. Role terapeuta je libovolná a bude mít přístup jen k datům uživatelů, kteří mu k tomu dají souhlas. Role klienta je tou nejnižší, co se týče privilegií, a je rolí výchozí.

8 Architektura systému

Tato kapitola popisuje architektonické principy řešeného problému, kterými se bude řídit vývoj požadovaného produktu. V první části jsou popsány komponenty, ze kterých bude celý systém složen. Z důvodu rozsáhlosti projektu bude každá komponenta popsána zvlášť spolu s architektonickými rozhodnutími, která byla provedena. V podkapitole 8.7 *Komunikační schéma* bude popsána komunikace mezi jednotlivými komponentami a kapitola 8.8 *Diagram datových toků* zachycuje toky dat napříč systémem.

Budou zde uvedeny pouze zásadní prvky navržené architektury, tak aby bylo zcela jasné, z čeho se systém bude skládat. Detailní vlastnosti systému budou popsány v kapitole Implementace.

Ze zadání a specifikace požadavků vzešel zásadní požadavek na to, aby byl systém *CAMP* centralizovaný. Prvním argumentem je, že data je nutné ukládat bezpečně a řízeně, proto optimálním řešením je uchovávat data tak, aby nad nimi měla plnou kontrolu pouze jedna entita. Systém bude tedy typu "server-client".

8.1 Softwarové a hardwarové vybavení

8.1.1 Server

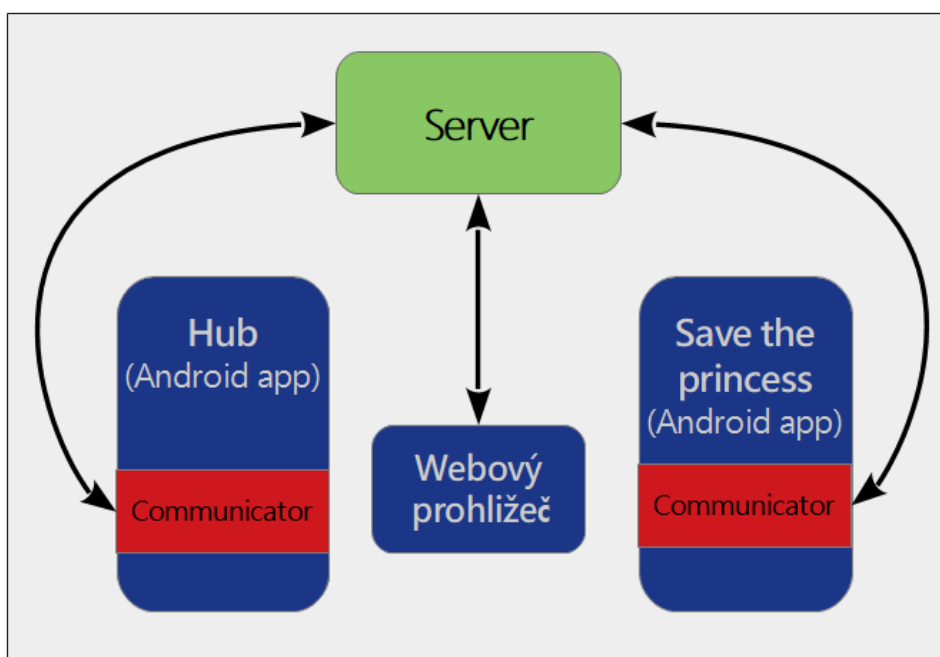
Pro účely této diplomové práce bude využit virtuální server spravovaný na KIV ZČU s IP adresou 147.228.63.49. Server disponuje operačním systémem Debian s verzí 3.16.7, procesorem Intel Xeon CPU E5-4620 s frekvencí 2.60GHz a pamětí RAM 8 GB. Přístup k tomuto serveru je s právy správce (root) systému.

8.1.2 Mobilní zařízení

Po konzultaci se zadavatelem byla jako platforma pro mobilní aplikace vybrán operační systém Android. Dle společnosti Gartner je podíl tohoto mobilního operačního systému bezmála 85 procentní. Z toho důvodu je vhodné v první řadě vyvíjet aplikace právě na tuto platformu. Dalším důvodem je, že některé již vytvořené kognitivní hry, které byly zmíněny v analytické části práce, jsou aplikacemi na Android. Z toho plyne zásadní architektonické rozhodnutí, že všechny mobilní aplikace, které jsou náplní této práce, budou vytvořené pro Android.

8.2 Popis komponent systému

Pro systém *CAMP* byla navržena následující architektura vyjádřená v diagramu níže (Obrázek 8.1). Komponenta *Server* obsahuje webovou aplikaci a databázi. Mobilní aplikace s názvem *Hub*, která bude sloužit primárně pro správu skupin uživatele a distribuci aplikací pro kognitivní trénink a mobilní aplikaci s názvem *Save the Princess*. Ta bude představovat hru pro kog. trénink s využitím čelenky Neurosky pro snímání EEG. Pro komunikaci mezi serverem a všemi mobilními aplikacemi je navržena komponenta *Communicator*

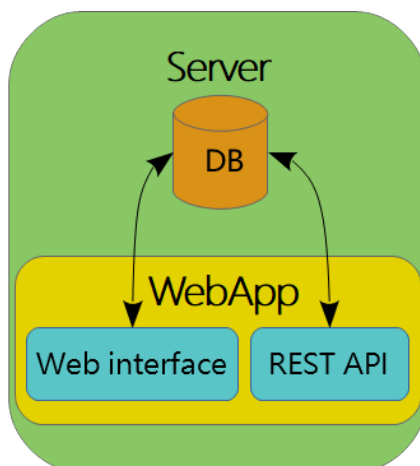


Obrázek 8.1: Komponentový diagram

Komponenta Webový prohlížeč je zde zobrazena jen pro úplnost, tak aby bylo patrné, že systém bude disponovat webovým rozhraním. Ostatní komponenty budou popsány podrobněji v následujících podkapitolách.

8.3 Server

Komponenta *Server* bude obsahovat databázi a webovou aplikaci. Bude disponovat veřejnou IP adresou a doménou, tak aby k ní šlo přistupovat z internetu.



Obrázek 8.2: Komponenta Server

8.3.1 Popis

Webová aplikace

Webová aplikace je navržena dle modelu MVC. Bude tedy rozdělena na části model, view a controller. Controller je dále rozdělen na dva balíčky. Část pro webové rozhraní přijímá http požadavky z prohlížeče. A druhá část je tzv. REST API, které slouží primárně pro komunikaci s mobilními aplikacemi. Model bude implementován objektově relačním zobrazením (ORM). Jedná se o programovací techniku, která zajišťuje automatickou konverzi dat mezi relační databází a objektově orientovaným programovacím jazykem. Webová aplikace bude obsahovat veškerou tzv. "business logic", tedy bude definovat logiku, která určuje vlastnosti systému (manipulace s daty, privilegia uživatelů apod.).

Databáze

Databáze je relačního typu. Relační databázové systémy jsou vhodné pro ukládání i většího množství dat a vyhledávání v nich, ale poskytují nízkou podporu pro manipulaci s nimi. To není ovšem zásadní problém. Při ostrém

provozu se očekává větší množství dat, ale transformace nad nimi budou minimální. Největší výhodou je fakt, že z dostupných databázových přístupů je tím nejnávštěvnějším a napříč technologiemi má velkou podporu a v blízké době se nepočítá se ukončením podpory. Konkrétní databázová služba bude definována v podkapitole *8.3.2 Implementační technologie*.

8.3.2 Implementační technologie

Webová aplikace

Webová aplikace bude implementována staticky typovaným jazykem **Kotlin**. Jedná se o programovací jazyk, který běží nad JVM. Zatímco není s jazykem Java syntakticky kompatibilní, je interoperabilní s knihovnamí a zdrojovými kódy Javy. Jeho hlavní výhodou je intuitivnější syntaxe a jednodušší syntaxe (např. nemá řádek ukončovací znak). Jedná se o moderní jazyk ve stabilní verzi vývoje, který je dobrou alternativou k Javě, se kterou je kompatibilní. Lze tak využít bohatou škálu knihoven a frameworků, které jsou psané v Javě.

Struktura webové aplikace je podřízena frameworku **Spring**. Prvním důvodem pro zvolení tohoto frameworku je to, že se jedná o vyspělý open-source aplikační rámec pro vývoj Java Enterprise aplikací. Je založen na principech Inversion of Control (IoC) a Dependency Injection. Zodpovědnost za vytvoření a provázání závislostí je přesunuta z aplikace na framework, což je pro programátora velkou výhodou. Mimo to disponuje širokou škálou funkcí. Druhým důvodem je skutečnost, že s tímto frameworkem mám již zkušenost.

Pro objektově-relační mapování (ORM) dat je vhodné použít framework **Hibernate**, který je implementací Java Persistence API (JPA). Při správném zacházení usnadňuje otázku provázání aplikace s databází. Je nutné ale dbát na kvalitní návrh modelu databáze. Hibernate generuje SQL dotazy automaticky, pokud není uvedeno jinak, a v některých případech může být nový dotaz příliš složitý a tak způsobí zbytečnou zátěž databáze.

Databáze

Jako databázová služba byla zvolena služba **MySQL**. MySQL je nejpoužívanější ze všech velkých databázových serverů. Jedná se o bohatý open source produkt. Vývoj s MySQL je poměrně jednoduchý a vývojáři mají přístup k velké komunitě. Velká většina webových stránek (a webových aplikací) může jednoduše pracovat s MySQL i přes některá omezení. Tento flexibilní a škálovatelný nástroj je snadno ovladatelný - což se z dlouhodobého hlediska ukazuje jako velmi užitečné.

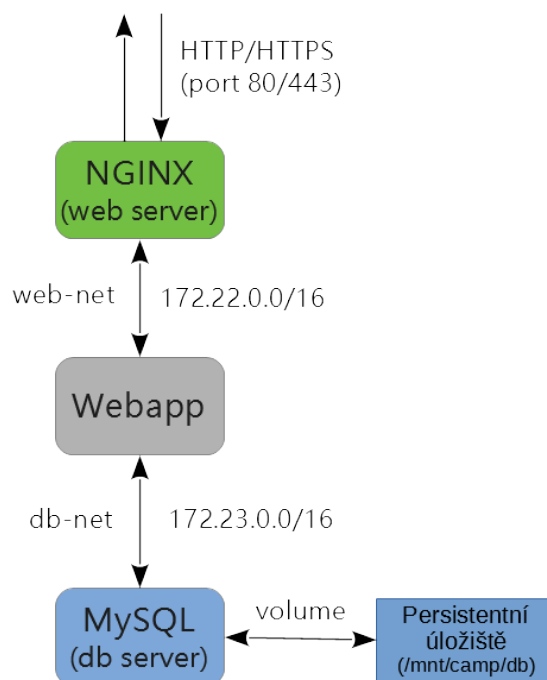
Pro výpočet statistických ukazatelů nad výsledky z cvičení bude vytvořena sada uložených funkcí a procedur společně s tzv. triggerem, které budou reagovat na změny dat. Systém tak bude sice databázově závislý, ale vzhledem k složitosti požadovaných výpočtů je nutné minimalizovat komunikační režii s tím spojenou. Zadavatel například požaduje, aby byl vypočítáván percentil dosažených výsledků a zároveň, aby bylo možné určovat věkové rozpětí uživatelů a jejich úroveň vzdělání, jejichž výsledky budou do výpočtů zahrnuté. Protože bude tento výpočet nutné provádět potenciálně nad velkou množinou dat, bude velkou výhodou, když ho bude provádět přímo databáze.

8.3.3 Nasazení

Webová aplikace společně s databází bude spuštěna a spravována na virtuálním linuxovém serveru s distribucí operačního systému Debian. Zásadním architektonickým rozhodnutím je to, že bude využita "open-source" služba Docker. Tato technologie izoluje jednotlivé aplikace do tzv. kontejnerů. Kontejner obsahuje všechny potřebné soubory, jako jsou spustitelný binární soubor a knihovny. Kontejner je oddělen od hostujícího operačního systému, čímž se zvyšuje bezpečnost a udržitelnost chodu systému. Využívá principy virtualizace operačního systému, ale k samotné virtualizaci nedochází. Tím se rapidně sníží s tím spojená režie. Další velkou výhodou je zjednodušení samotného procesu nasazení.

Programátor na svém lokálním stroji vytvoří tzn. Docker image. Tento výraz lze interpretovat jako ucelenou adresářovou strukturu, která obsahuje všechny potřebné soubory. Pokud jde například o webový server, obsahuje nainstalovaný server náležitě nakonfigurovaný, zdrojové soubory webové aplikace a prostředí operačního systému. Z vytvořeného obrazu (image) lze spustit kontejner. Programátor si na svém lokálním stroji ověří správnost této konfigurace a vloží obraz do vzdáleného repozitáře na Docker cloud. Správce serveru si stáhne novou verzi obrazu a spustí ho. V tuto chvíli, při dodržení výše zmíněného postupu, je zaručeno, že vše funguje stejně jako na lokálním stroji a nevyskytne se například velmi častý problém se splněním závislostí aplikace. Jedná se o oblíbený a moderní přístup v oblasti správy serverů.

Pro tento systém byla navržena následující množina docker kontejnerů, které budou spolu navzájem komunikovat pomocí tří podsítí. Podsítě jsou vytvořeny službou docker.



Obrázek 8.3: Schéma komunikace docker kontejnerů

První kontejner s názvem *NGINX* (v něm spuštěn webový server) zachytává příchozí HTTP/HTTPS požadavky a předává je kontejneru *Webapp*. Kontajner *NGINX* je na tomto serveru již používán i pro jiné účely. *Webapp* obsahuje webovou aplikaci, která je popsána výše, ve formě war souboru. War soubor bude spuštěn na portu 8080 službou Tomcat. Webová aplikace komunikuje s databázovým kontejnerem pro tuto určenou podsítí. Protože obsah kontejnerů není perzistentní, tzn. data po vypnutí/restartu kontejneru nejsou zachována, musí databáze využívat tzv. volume. Jedná se o mechanismus, který kontejneru zpřístupní paměťové úložiště, které je mimo tento kontejner. V tomto případě se bude jednat o složku s cestou `"/mnt/camp/db"` na hostujícím serveru. Tímto způsobem je dosažena persistence dat.

Výhodou tohoto přístupu je tedy snadný vývoj a nasazení nových verzí, bezpečnost a zjednodušení případné migrace na jiný server. Za nevýhodu tohoto přístupu může být považována nutnost znalosti další technologie, která je ovšem vykompenzována výší zmíněných přínosů.

8.3.4 Uživatelské rozhraní - webová aplikace

Grafické rozhraní webové aplikace bude využívat CSS knihovny Bootstrap a bude plně responzivní.



Obrázek 8.4: CAMP - návrh GUI

Obrázek vykresluje návrh GUI. V tomto případě se jedná o stránku, které zobrazuje výsledky uživatele. Obsahuje vypočtenou hodnotu percentilu, dále graf ukazující křivku skóre v čase a tabulku s všemi provedenými cvičeními (sezení).

8.4 Knihovna - Communicator

8.4.1 Popis

Komponentu *Communicator* budou využívat všechny mobilní aplikace pro komunikaci s webovou aplikací.



Obrázek 8.5: Komponenta Communicator

Webová aplikace vystavuje skrz HTTP protokol tzv. REST API. Jedná se o rozhraní, které se používá pro jednotný a jednoduchý přístup ke zdrojům.

Může se jednat o data nebo stavy aplikace. Rest API je orientováno datově, každý zdroj má svůj identifikátor (URI) a definuje čtyři metody pro přístup k datům (Retrieve, Create, Update, Delete).

V tomto případě budou data v komunikaci ve formátu JSON (JavaScript Object Notation). Jedná se o formát, který je odvozený z JavaScriptu, jeho používáním vzniká nízká provozní režie a jeho struktura není nijak složitá.

Komponenta *Communicator* bude představovat knihovnu, kterou bude možné jednoduše použít ve všech mobilních aplikacích. Programátor aplikace pro kognitivní trénink tak téměř nebude muset řešit komunikaci se serverem.

Pro představu, pokud bude tento programátor chtít odeslat data do systému *CAMP*, do svého projektu vloží tuto knihovnu a provede volání následujících metod:

1. **login** - s parametrem uživatelské jméno a heslo uživatele
2. **sendData** - jako parametr vloží datový objekt s následujícími parametry (objekt je definován podrobněji v kapitole *9 Implementace systém*):
 - skóre
 - první dodatečná metrika
 - druhá dodatečná metrika

Communicator implementuje tyto a další metody, tak aby integrace se systémem *CAMP* mohla proběhnout tím nejsnazším způsobem.

Protože *Communicator* využívá jednotné REST API, programátor mobilních aplikací bude moci v případě nutnosti (např. rozdílný programovací jazyk) využít právě toto rozhraní. Tento způsob je ovšem zdatelně složitější, ale možný je.

Stejně jako všechny ostatní mobilní aplikace bude tato komponenta napsaná v programovacím jazyce Kotlin. Volba tohoto programovacího jazyka je ozejmena v podkapitole *8.4.2 Implementační technologie*.

8.4.2 Implementační technologie

Komponenta *Communicator* bude psaná v jazyce Kotlin, takže bude jednoduše integrovatelná s projekty stejného jazyka a i s projekty psané jazyce Java. Zdrojové soubory budou zkompileovány do souboru jar, který bude v dalších projektech využíván jako knihovna.

Pro komunikaci se serverem skrz REST API, bylo rozhodnuto využít knihovny Retrofit2 s licenci Apache License. Knihovna vytváří HTTP klienta a řeší posílání konkrétních HTTP požadavků a následné zpracování

odpovědi. V praxi to funguje tak, pokud chceme odeslat data na server, vytvoříme funkci například "sendData", vložíme objekt představující data a anotací nadefinujeme jaká URL se má pro tento dotaz použít. Po zavolání této funkce je poslán HTTP požadavek a knihovna následně zpracuje odpověď, kterou může být JSON objekt chybového hlášení. Knihovna mimo jiné nabízí i synchronní či asynchronní zpracovávání.

Použitím této knihovny odpadá nutnost řešit samotnou HTTP komunikaci mezi serverem a *Communicatorem*.

8.5 Aplikace - Hub

8.5.1 Popis

Komponenta s názvem *Hub* představuje mobilní aplikaci pro Android. Důvod volby této platformy je uveden v podkapitole *18.5.2 implementační technologie*. Aplikace bude využívat komponentu (knihovnu) *Communicator* pro komunikaci se serverem výše zmíněným způsobem. Aplikace bude do jisté míry zastávat webové rozhraní v tom smyslu, že bude nabízet možnost spravovat členství v uživatelských skupinách a zobrazit dosažené výsledky. Hlavní funkcí však bude distribuce aplikací pro kog. trénink realizovaná následujícím způsobem.



Obrázek 8.6: Komponenta Hub

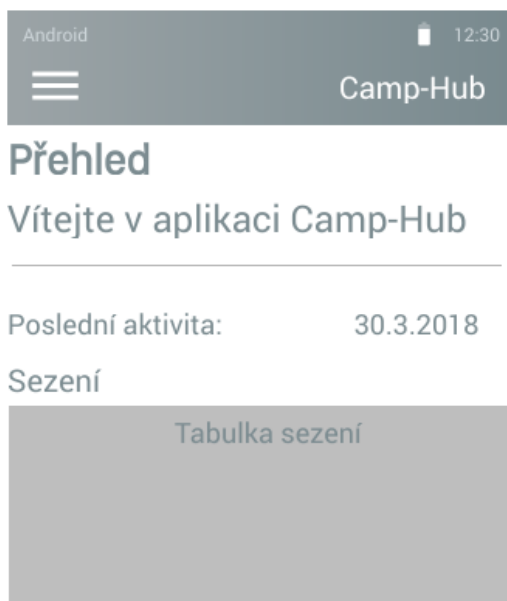
Aplikace zobrazí uživateli všechny dostupné aplikace, které jsou v systému *CAMP* registrovány, spolu s jejich stručným popisem. Po výběru aplikace uživatel klikne na tlačítko, které způsobí buď spuštění aplikace nebo, pokud není nainstalovaná na jeho zařízení, odkáže uživatele na profil aplikace v Android Play Store, kde bude mít možnosti ji nainstalovat. Aplikace bude taktéž vyvíjena v programovacím jazyce Kotlin.

8.5.2 Implementační technologie

Aplikace pro Android mohou být vytvářeny primárně pomocí jazyků Java, C++ a Kotlin. Vzhledem k tomu, že webová aplikace a *Communicator* jsou psané v jazyce Kotlin, je vhodné pro využití této možnosti vyvíjet mobilní aplikace ve stejném jazyce. Zdrojové kódy celého systému tak budou ucelené a přehlednější. Další výhodou je možnost jednoduchého sdílení souborů napříč projektem. Toto se týká například DTO objektu mezi webovou aplikací a *Communicatorem*, potažmo mobilní aplikací, které používá tuto knihovnu. Android studio jazyk Kotlin podporuje. SDK pro Android zkompiluje kód spolu s datovými a zdrojovými soubory do souboru APK.

8.5.3 Uživatelské rozhraní

Bylo navrženo následující jednoduché uživatelské rozhraní. Obrázek zobrazuje přehled, který se objeví po spuštění aplikace.



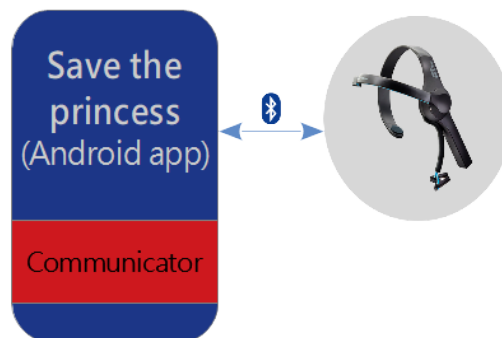
Obrázek 8.7: Camp-Hub - GUI

Menu bude obsahovat položky Přehled, Moje skupiny, Přidat skupiny, Zprávy a Aplikace.

8.6 Aplikace - Save the Princess

8.6.1 Popis

Komponenta představuje příklad kognitivní hry, která využívá čelenku Mindwave mobile pro snímání EEG. Bude se jednat o první aplikaci zařazenou do tohoto systému. Všechny funkce systému *CAMP* tak budou kompletní a otestované už při vydání první verze, tzn. při dokončení této diplomové práce.



Obrázek 8.8: Komponenta Save the Princess

Princip hry bude založen na otestování pozornosti uživatele. Pozornost uživatele, tedy kognitivní funkce, je odvozena z nasnímaného EEG záznamu s využitím čelenky Mindwave mobile, která je popsána v analytické části. Čelenka je propojena s Android zařízením bezdrátovou technologií Bluetooth, přičemž obsluha je nenáročná. Tyto dvě zařízení bude nutné pouze spárovat a poté spustit aplikaci. Po přihlášení do systému *CAMP* pomocí uživatelského jména a hesla bude možné spustit cvičení.

Po spuštění cvičení se zobrazí 2D hrací plátno s pohledem shora, na kterém bude do jisté míry náhodně umístěna postavička princezny a určité množství beden (boxů), které je nutné v časovém limitu 60 vteřin zničit pomocí dělových projektilů. Dělo je umístěno ve spodní části plátna. Kliknutím na pozici konkrétní bedny je vystřelen projektil letící k směrem k této pozici. Kolizí je umenšována hodnota přiřazená k dané bedně, která představuje její odolnost. Hodnota o kterou je odolnost zmenšena, je určena úrovní pozornosti uživatele v době, kdy byl projektil vystřelen. Cvičení je úspěšně

dokončeno ve chvíli, kdy jsou odstraněny všechny bedny. Hodnocení cvičení, neboli skóre bude závislé na rychlosti odstranění všech beden a průměrné hodnoty pozornosti.

Po úspěšném dokončení cvičení jsou zobrazeny uživateli výsledky a na server odeslána data. Data se skládají ze:

- **skóre**
- **počet odstraněných beden**
- **zbývající čas**
- **textový řetězec** - obsahující všechny hodnoty pozornosti, které byly během hry nasnímány.
- **obrázek** - zachycuje rozložení hracího pole.

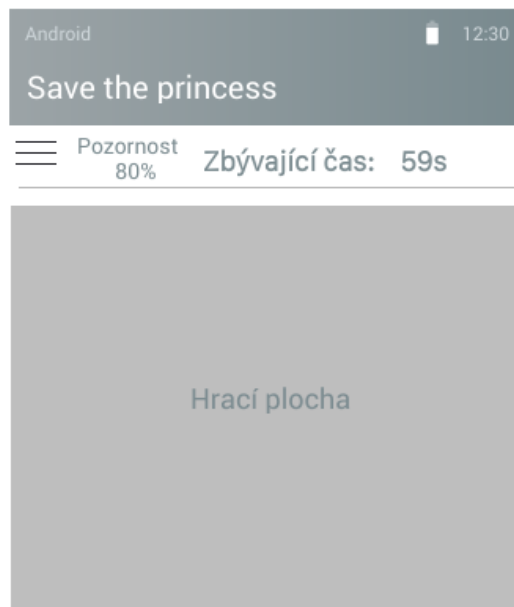
Obrázek odesílaný na server má v této hře pro terapeuta minimální vypovídající hodnotu. Při analýze však vzešel požadavek na tuto možnost odesílat obrázek spolu s ostatními daty. Větší využití má v situaci, kdy uživatel prokazuje vlastní invenci, či kreativitu při plnění stanoveného úkolu. Aplikace také může vyhotovovat například nějaký graf, který může být pro terapeuta relevantní.

8.6.2 Implementační technologie

Tato aplikace bude vytvořena stejně jako aplikace *Hub* pomocí jazyka Kotlin z důvodů, které byly již uvedeny. V zásadě se jedná možnost použití jednoho programovacího jazyka napříč systémem, nepočítáme-li například prog. jazyky použité pro View v rámci MVC modelu ve webové aplikaci a SQL pro databázi.

8.6.3 Uživatelské rozhraní

Následující návrh GUI obsahuje obrazovku po spuštění hry.

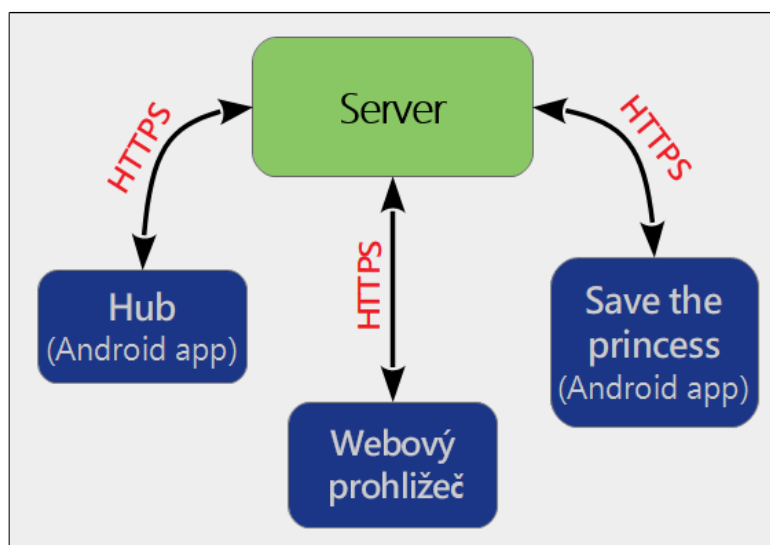


Obrázek 8.9: Save the Princess - GUI

V horní části má uživatel možnost přerušit hru zobrazením menu. Vedle se nachází hodnota aktuální pozornosti a zbývající čas. Ve zbytku obrazovky je vykreslováno hrací pole.

8.7 Komunikační schéma

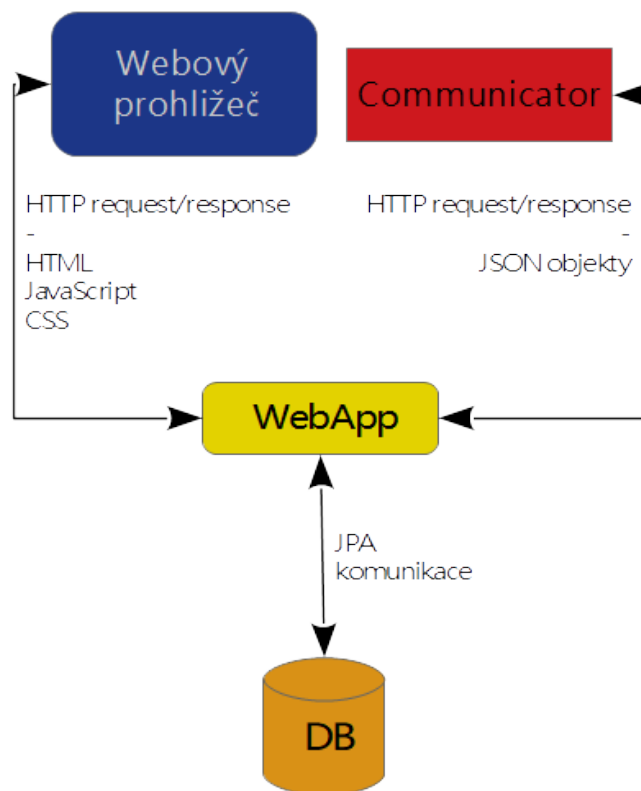
Jedná se o jednoduché schéma, které zachycuje jakým způsobem a protokolem budou jednotlivé komponenty komunikovat se serverovou částí. Ve všech případech se jedná o zabezpečený HTTP protokol (Hypertext Transfer Protocol), tedy HTTPS (Hypertext Transfer Protocol Secure). Tento protokol využívá SSL nebo TLS protokol, který asymetrickou šifrou ověřuje identitu webového serveru. Po ověření probíhá komunikace symetrickou šifrou. HTTPS je dnes běžným standardem a je stále považován jako bezpečný pro internetovou komunikaci. Je ovšem nutné získat HTTPS certifikát od jedné z certifikačních autorit a certifikát je nutné po čase obnovovat. Na komerčním trhu jsou však autority, které certifikáty vystavují bezplatně.



Obrázek 8.10: Komunikační schéma

8.8 Diagram datových toků

Datové toky v systému jsou zachyceny následujícím jednoduchým DFD diagramem. Jak už bylo zmíněno, webový prohlížeč uživatele i komponenta *Communicator* komunikují s webovou aplikací pomocí HTTPS komunikace.



Obrázek 8.11: Diagram datových toků

Komunikace mezi webovým prohlížečem a serverem bude realizována prostřednictvím výměny dokumentů, které obsahují značkovací jazyk HTML, skriptovací jazyk JavaScript a stylovací (stylesheet) jazyk CSS.

Komponenta *Communicator*, která bude obsažena v každé mobilní aplikaci, si s webovou aplikací vyměňuje JSON dokumenty, které mohou představovat DTO (Data Transfer Object) objekty nebo objekty chybového hlášení.

Webová aplikace je propojena pomocí rozhraní JPA (Java Persistence API). Implementace rozhraní JPA umožňuje vkládat instance tříd do relační databáze, jde tedy o ORM (objektově relační mapování). Lze je logicky i upravovat, mazat a zase získávat.

9 Implementace systému

Dle navržené architektury je možné celý projekt rozdělit na 6 separátních částí, a to:

- **Databáze** - úložiště dat, vložené funkce a triggerery
- **Webová aplikace** - MVC model, obsluha webového rozhraní a REST API
- **Docker** - nasazení webové aplikace a databáze na serveru
- **Mobilní aplikace - Camp Hub** - umožní uživateli částečně spravovat svůj účet a poskytuje (distribuuje) kognitivní hry v systému
- **Mobilní aplikace - Save the Princess** - kognitivní hra využívající Mindwave mobile
- **Knihovna - Communicator** - knihovna pro komunikace mobilní aplikace a webové aplikace (REST API)

V této kapitole budou samostatně popsány všechny tyto části. Každá podkapitola obsahuje další doplňující architektonická rozhodnutí, která je vhodné zmiňovat společně s popisem implementace¹ jako takové.

9.1 Databáze

Dle architektury byl zvolen relační open-source databázový systém MySQL. Nejdříve bylo nutné vytvořit tabulky dle navrženého ERA modelu, který bude popsán níže. Vytváření a úprava tabulek je však obsluhována výhradně webovou aplikací pomocí výše zmíněného frameworku Hibernate. Ten umožňuje objektově relační mapování. Vytvořením příslušných tříd (entit) vznikla v databázi množina tabulek dle navrženého ERA modelu.

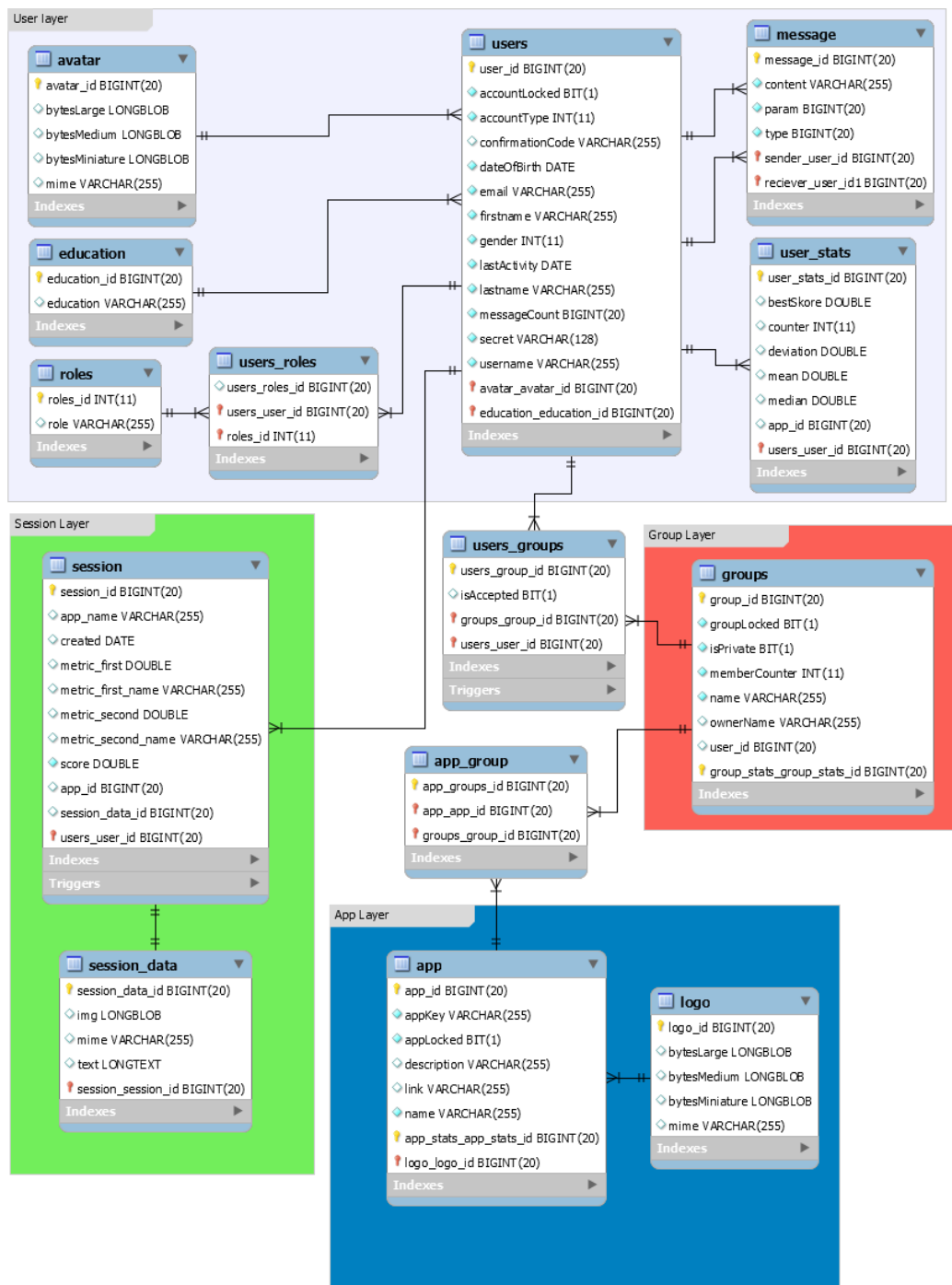
Další část je věnována triggerům (spouštěčům) a vloženým funkcím v databázi, které jsou využívány pro některé vedlejší funkce (např. inkrementace/dekrementace ukazatele počtu členů ve skupině) a pro výpočet statistických ukazatelů, konkrétně percentil.

Databázový server bude v rámci docker kontejneru spuštěn na portu 3306 a bude přístupný skrz privátní síť pouze webové aplikaci.

¹Digramy třídy jsou z důvodu velkého rozsahu k nahlédnutí na příloženém CD.

9.1.1 ERA model

Před implementací byl vytvořen ERA model databáze, které byl při implementaci rozšířen o několik atributů.



Obrázek 9.1: Navržený ERA model

Diagram výše je již konečnou verzí. Je rozdělen na 4 vrstvy. *User Layer* obsahuje všechny tabulky spojené s entitou uživatel. Stejně je to i ostatních vrstev. *Group Layer* obsahuje tabulky pro entitu skupina, *Session Layer* pro entitu sezení (jedno cvičení) a *App Layer* pro mobilní aplikace v systému. Tento ERA model tedy odpovídá skutečnému schématu v databázi. Nyní budou popsány jednotlivé tabulky v rámci uvedených vrstev.

User Layer

- **Users** - tabulka, která obsahuje záznamy o uživatelích. Kromě osobních dat (jméno, datum narození) obsahuje i kontakt a heslo, které je ukládáno z důvodu bezpečnosti ve formě řetězce vytvořeným hash funkcí bcrypt. Dále uživatelské jméno, záznam o tom, zda je uživatel blokován, a počet nevyřízených zpráv (níže popsáno podrobněji).
- **Avatar** - záznam v této tabulce obsahuje tři bytová pole pro jeden obrázek. Uživatel vkládá tento obrázek jako profilový obrázek, čímž se vytvoří záznam v této tabulce. Při ukládání se obrázek konvertuje do tří rozlišení (jedno bytové pole pro jedno rozlišení), tak aby byl obrázek přenášen jen v takové kvalitě, v jaké je to nutné.
- **Education** - tabulka obsahuje všechny úrovně dosaženého vzdělání. Každému uživateli je přiřazena právě jedna úroveň.
- **Roles** - každý uživatel může disponovat n rolemi. Vzniká tak vazba n:m, která je implementována pomocí spojovací tabulky `user_roles`. V systému jsou definovány tyto role: `admin`, `app_manager`, `therapist` a `client`.
- **Message** - tato tabulka je s tabulkou `Users` spojena pomocí dvou vazeb n:1. A to z toho důvodu, že každá zpráva má jednoho odesílatele a příjemce.
- **User_stats** - pro snížení výpočetní zátěže a síťové komunikace analýzy dosažených výsledků (data jednoho sezení), má každý uživatel pro každou aplikaci vytvořený záznam v tabulce `User_stats`, kde jsou pomocí triggerů (spouštěčů) průběžně aktualizovány statistické ukazatele. Jedná se aritmetický průměr výsledků uživatele z jedné aplikace, odchylka výsledků, medián, čítač všech sezení a nejlepší skóre. Tyto hodnoty budou dále využívány.

Group Layer

- **Groups** - tabulka obsahuje záznamy o všech uživatelských skupinách. Skupinu vytváří a upravuje terapeut (role therapist), popř. administrator (role admin). Terapeut určí, jaké aplikace budou do skupiny začleněny. Tyto aplikace budou jednak zahrnuty do statistik skupiny, ale zároveň člen skupiny uvidí, jaké aplikace má využívat. Dále terapeut spravuje množinu členů. Pokud chce nějakého uživatele přidat do skupiny, tomuto uživateli je zaslána zpráva s požadavkem. Odstraňovat uživatele ze skupiny lze libovolně. Pokud je skupina označena terapeutem jako soukromá a pokud se uživatel chce stát členem, jeho členství schvaluje terapeut. Členstvím ve skupině se terapeutovi zpřístupní všechna data uživatele.

Session Layer

- **Session** - záznam v tabulce představuje jedno sezení uživatele, tedy jedno proběhlé cvičení v aplikaci pro kognitivní trénink. Každá aplikace pomocí komponenty *Communicator*, popř. REST API, posílá DTO objekt sezení, ze kterého je vytvořen plnohodnotný záznam. Obsahuje reálné číslo představující skóre (ohodnocení sezení), další dvě libovolné číselné metriky a jejich pojmenování. Každé sezení je spjato s konkrétním uživatelem a aplikací pro kognitivní trénink.
- **Session_data** - k záznamu o sezení může být volitelně přiřazen záznam z tabulky `session_data` s kardinalitou 1:1. Ten uchovává obrázek v binární podobě a textový řetězec. Aplikace může k danému sezení tedy odesílat jeden obrázek obsahující například snímek obrazovky či graf, a text, který představuje například hrubá naměřená EEG data. Sémantika těchto dat je libovolná.

App Layer

- **App** - záznam o aplikaci obsahuje název aplikace, popis aplikace, název balíčku. Dále pak generovaný klíč, kterým je zajištěno, že data odesílaná z aplikací jsou věrohodná. Správce aplikací má možnost aplikaci zablokovat, čemuž slouží jeden bit v záznamu.
- **Logo** - k aplikaci může být přiřazeno logo, které je uchováváno stejným způsobem, jako avatar (profilový obrázek) uživatele.

9.1.2 Vložené funkce a triggery

Vložená funkce (stored function) je podprogram uložený v databázi. Funkce má název, seznam parametrů a příkazy SQL. A trigger (spoušť) je sada akcí, které se spouštějí automaticky, když se nad zadanou tabulkou provede daná SQL operace (INSERT, UPDATE nebo DELETE).

Triggery

V databázi byly vytřeny tyto triggery:

- **users_groups_AFTER_INSERT** - trigger, který při přihlášení uživatele do skupiny inkrementuje čítač uživatelů v záznamu skupiny
- **users_groups_AFTER_DELETE** - odstraněním uživatele ze skupiny dekrementuje čítač uživatelů v záznamu skupiny
- **session_AFTER_INSERT** - při vložení nového sezení se tímto triggerem přepočítají statistické ukazatele v tabulce `user_stats`. Jedná se o aritmetické průměr, standardní odchylku, a maximum nad všemi sezeními uživatele z jedné aplikace.

Pro představu je zde uvedena signatura a stěžejní část těla triggeru pro přepočítávání statistik uživatele (`session_AFTER_INSERT`).

```
CREATE TRIGGER 'cognitiveamp' . 'session_AFTER_INSERT'
AFTER INSERT ON 'session' FOR EACH ROW
BEGIN
    ...
END
```

Trigger je spuštěn automaticky po vložení každého nového záznamu do tabulky `session`. Dotazem níže jsou vypočítány nad všemi sezeními nové hodnoty pomocí funkcí databáze (AVG, STD, MAX).

```
SELECT AVG(session.skore), STD(session.skore),
       MAX(session.skore) INTO avg_, std_, max_
FROM session
WHERE session.app_id = NEW.app_id
AND session.user_id = NEW.user_id;
```

Nové hodnoty jsou následně propsány do příslušného záznamu v `user_stats`. Jedná se o jednoduchý SQL INSERT příkaz, proto zde není uveden.

Vložené funkce

Pro účely analýzy výsledků sezení byly vytvořeny dvě funkce. První funkce slouží pro výpočet percentilu jednoho uživatele vůči všem uživatelům v databázi. A druhá funkce slouží pro výpočet percentilu skupiny.

Funkce pro výpočet percentilu uživatele je deklarována následujícím způsobem.

```
CREATE FUNCTION 'get_user_percentil '  
(user_id INTEGER, app_id INTEGER)  
RETURNS double  
BEGIN  
    ...  
END
```

Nejdříve je zjištěná průměrná hodnota výsledků uživatele z tabulky *user_stats*. Tato hodnota je porovnána s průměrem všech ostatních uživatelů. Z počtu uživatelů, kteří mají menší průměr, lze vypočítat percentil.

```
SELECT user_stats.mean INTO avg_ FROM user_stats  
WHERE user_stats.user_id = user_id  
AND user_stats.app_id = app_id;  
  
SELECT SUM(IF(user_stats.mean < avg_, 1, 0)),  
COUNT(user_stats.user_stats_id) INTO pct_, ttl_  
FROM user_stats  
WHERE user_stats.app_id = app_id;  
RETURN pct_ / (ttl_ - 1);
```

9.2 Webová aplikace

Webová aplikace využívá aplikační framework Spring a knihovnu Hibernate pro ORM (Object-Relational Mapping). Obsluhuje HTTP požadavky z webového prohlížeče a poskytuje REST API pro komunikaci s mobilními aplikacemi.

9.2.1 Struktura projektu

Projekt je složen z balíčků obsahující zdrojové soubory.

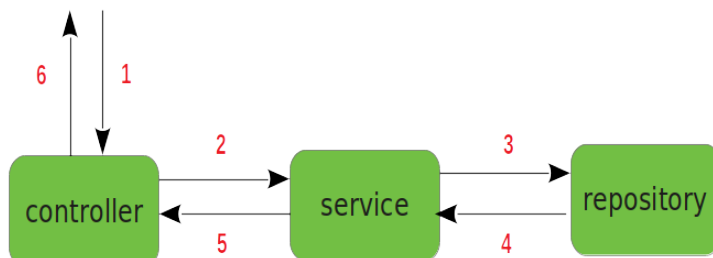
- **configuration** - obsahuje třídy pro konfiguraci Spring aplikace. Definiuje se například umístění JSP (JavaServer Pages), klientské cookie a

připojení k databázi. Důležitým souborem je `SecurityConfiguration.kt`, který obsahuje pravidla definující jednotlivé uživatelské role. Například uživatel bez role `THERAPIST` nemůže vstoupit na stránku se správou skupin.

- **domain** - obsahuje třídy představující perzistentní doménové objekty (tzv. entity). Entita představuje v databázi tabulku a každá instance entity odpovídá řádku v této tabulce. Dále tento balíček obsahuje složku `dto` s DTO třídami pro výše zmíněné entity. Složka je sdílená napříč projekty systému `CAMP`.
- **repository** - pro každou entitu je vytvořen tzv. Spring JPA Repository. Použitím JPA repositáře se významně sníží množství kódu pro implementaci vrstev přístupu k datům. V JPA Repository lze jednoduchým způsobem definovat funkce, kterými budou data (entity) manipulována.
- **service** - takzvané služby, obsahují veškerou "business logiku". Například `UserService` obsahuje funkci, která vytvoří nového uživatele. Pouze služba manipuluje skrz repositář přímo s daty. Použití funkcí služeb je podmíněno uživatelskými rolemi.
- **controller** - třídy typu *controller* obsahují funkce, které zachytí http požadavek a spustí funkci dané služby a vrátí odpověď (například html dokument).
- **restController** - tyto soubory jsou implementovány podobným způsobem jako controller, až na to, že vrací odpověď výhradně ve formě JSON objektů. Slouží pro vytvoření REST API a je zpřístupněno url prefixem *rest/*.
- **exception** - třídy představující výjimky. Obsahují název výjimky, zprávu a i kód http odpovědi. Takže například služba vyhodí výjimku kvůli nedostatečnému oprávnění a *controller* tuto výjimku zachytí a vrátí http odpověď s kódem 403.
- **utils** - tento balíček obsahuje některé pomocné třídy. Jedná se například o třídu pro odesílání emailových zpráv ze serveru a třídu pro konverzi obrázků.

9.2.2 Postup obsluhy požadavku

V této části bude popsáno, jakým způsobem je zpracován HTTP požadavek a jak jednotlivé součásti webové aplikace spolu kooperují. K tomu je zde uveden následující diagram.



Obrázek 9.2: Obsluha požadavku

Controller zachytí HTTP požadavek klienta a zkontroluje, zda obsahuje parametry které má. Pokud se jedná například o požadavek, který má zobrazit stránku s informacemi o uživateli, parametrem bude ID požadovaného uživatele. *Controller* disponuje instancí služby (*service*), která je inicializovaná aplikačním frameworkem Spring dle paradigmatu Inversion of Control. V tomto modelovém příkladu *controller* využije službu *UserService* a zavolá funkci se signaturou:

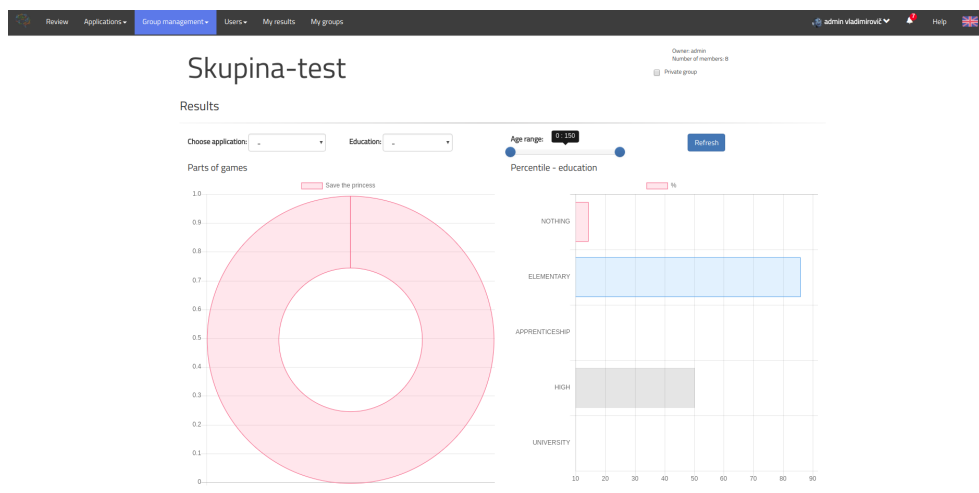
```
fun getUser(userId : Long) : UserDto
```

Parametr typu *Long* představuje uživatelské ID a návratovou hodnotou je DTO objekt uživatele. V této funkci je nejdříve ověřeno, zda má uživatel oprávnění pro tuto funkci. Pokud ano, využije instanci repositáře (*repository*), v tom případě se jedná o *UserRepository* a použije funkci se signaturou:

```
fun getUserById(userId : Long) : User
```

Touto funkcí se vytvoří SQL dotaz, který je poté zaslán databázi. Přijatá data jsou automaticky namapována na objekt *User*. Služba instanci uživatele převede do DTO objektu, který je následně vrácen controlleru. Controller data vloží jazykem JSP do html šablony, které je následně odeslána jako HTTP response (odpověď).

Obrázek níže ukazuje detail skupiny, který si má možnost zobrazit terapeut. Na této ukázce jsou vidět grafy vztažené na výsledky sezení členů skupiny.



Obrázek 9.3: Detail skupiny

Během implementace této komponenty vzniklo 179 zdrojových souborů, které celkem obsahují 9898 řádek kódu.

9.3 Communicator

Communicator představuje knihovnu, kterou je možné využít pro komunikaci se serverem. Byla použita knihovna Retrofit2 pro implementaci HTTP klienta.

Hlavní třídou knihovny je třída *Communicator*, která uživateli poskytuje funkce (např. login nebo getUser) a proměnné (např. loggedUser s datovým typem UserDto). Funkce pro přihlášení vypadá takto:

```

override fun login(username: String?,
    password: String?): UsersDto {

    var call: Call<UsersDto> = services?.basicLogin()
    ?: throw NetworkException("Unable to login")
    loggedUser = execute(call).body()
    if (loggedUser != null) isLogged = true
    return loggedUser
}

```

Nejdříve je vytvořena instance volání (*Call*) použitím třídy *Services*, kde jsou definovaná jednotlivá HTTP volání (viz níže). Dále se toto volání provede a zpracuje se tělo odpovědi. Pokud je přihlášení úspěšné, je uživateli poskytnut DTO objekt přihlášeného uživatele. Pokud přihlášení selže (např. špatné heslo), je při zpracování odpovědi chyba rozpoznána (server posílá

chybová hlášení ve formě JSON) a důsledkem toho je vyhozena výjimka. Výjimky musí zachytit uživatel, tedy programátor, a náležitým způsobem na ně zareagovat.

HTTP požadavek pro přihlášení je takto definován v souboru *Services*:

```
@POST("rest/login")
fun basicLogin(): Call<UsersDto>
```

Požadavek bude typu *POST* a URL doplněna o "rest/login". Celá URL bude pak "https://camp.kiv.zcu.cz/rest/login". V těle odpovědi se očekává DTO instance úspěšně přihlášeného uživatele. Retrofit2 na základě takto definovaného rozhraní provede příslušné volání pomocí HTTP klienta. Stejným způsobem jsou implementovány všechny ostatní funkce, které lze rozdělit do těchto skupiny:

- přihlášení/odhlášení
- odesílání dat
- získávání dto objektů - uživatel, skupina, aplikace, sezení, zprávy
- změny stavů - odpovědi na zprávy, členství ve skupinách

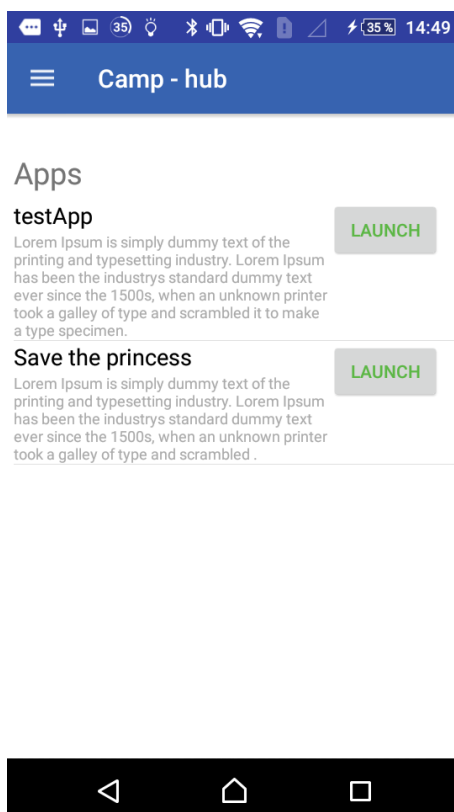
Společně s příručkou pro integraci je knihovna distribuována s příkladovým projektem, který obsahuje implementaci použití této knihovny. Knihovnu tak lze využít i s minimální znalostí jednotlivých částí systému (například REST API). Přihlášení do systému z pohledu uživatele knihovny vypadá následovně:

```
var communicator = Communicator()
var user : UserDto
try {
    user = communicator.login(username, password)
} catch (e : RuntimeException){
    ...
}
```

Během implementace této komponenty vzniklo 18 zdrojových souborů, které celkem obsahují 424 řádek kódu.

9.4 Mobilní aplikace - Camp Hub

Mobilní aplikace pro android *Camp Hub* byla vytvořena primárně pro klienty (pacienty). V aplikaci je možné si po přihlášení prohlížet své výsledky a spravovat členství ve skupinách (přidávat a odebírat). Dále má uživatel možnost zobrazit všechny dostupné aplikace pro kognitivní trénink a přečíst si k nim popis. Pokud má danou aplikaci již nainstalovanou, zobrazí se mu tlačítko pro spuštění. A pokud ne, tlačítko pro instalaci ho přeměruje do Google Play Store na profil dané aplikace, kde je možné aplikace stáhnout a nainstalovat. Toto je znázorněno na následujícím obrázku:



Obrázek 9.4: Zobrazení aplikací

Aplikace využívá knihovny *Communicator* pro komunikaci se serverem. Obsahuje jednu aktivitu, ve které je zobrazují tyto fragmenty:

- **LoginFragment** - fragment, který obsahuje grafické rozhraní pro přihlášení. Po přihlášení je pomocí knihovny *Communicator* zobrazen fragment *PreviewFragment*.
- **PreviewFragment** - zobrazuje informaci o poslední aktivitě uživatele a list všech sezení uživatele.

- **MessagesFragment** - obsahuje list všech nevyřízených příchozích zpráv. V současné době zpráva může obsahovat pouze žádost o členství ve skupině, nebo pozvánku. Kliknutím na tlačítko "Přijmout", nebo "Zamítnout" se odešle odpověď.
- **GroupsFragment** - zobrazuje v *ListView* záznam o všech skupinách ve kterých je uživatel členem. Uživatel má možnost z dané skupiny odejít.
- **AvailableGroupsFragment** - opět v *ListView* jsou zobrazeny všechny skupiny ve kterých uživatel členem není a informace o tom, zda jsou soukromé. Pokud ano, musí členství schválit vlastník skupiny (terapeut).
- **AppsFragment** - v *ListView* jsou zobrazeny všech aplikace pro kognitivní trénink. Ke každé aplikaci je uveden krátký popis a tlačítko, které buď aplikaci spustí, anebo odkáže na Google Play Store.

Aplikace je podporována od verze sdk 15. Je napsaná v jazyce Kotlin a kompilována android sdk s verzí 25 a podporuje dvě jazykové mutace (čeština, angličtina). Během implementace této komponenty vzniklo 60 zdrojových souborů, které celkem obsahují 2575 řádek kódu.

9.5 Mobilní aplikace - Save the Princess

Aplikace představuje jednoduchou kognitivní hru, která využívá čelenku Mindwave pro snímání EEG. Po konzultaci s vedoucím práce bylo rozhodnuto, že aplikace bude využívat úroveň pozornosti, kterou čelenka poskytuje. Jak už bylo zmíněno v architektuře systému, uživatel má za úkol držet svoji pozornost na co nejvyšší úrovni. Po hrací ploše jsou náhodně rozmístěny bedny, které má za úkol v časovém limitu zničit pomocí děla. Účinnost vypálených projektilů je určena úrovní pozornosti. Herní pole je zobrazeno na další stránce.



Obrázek 9.5: Herní pole

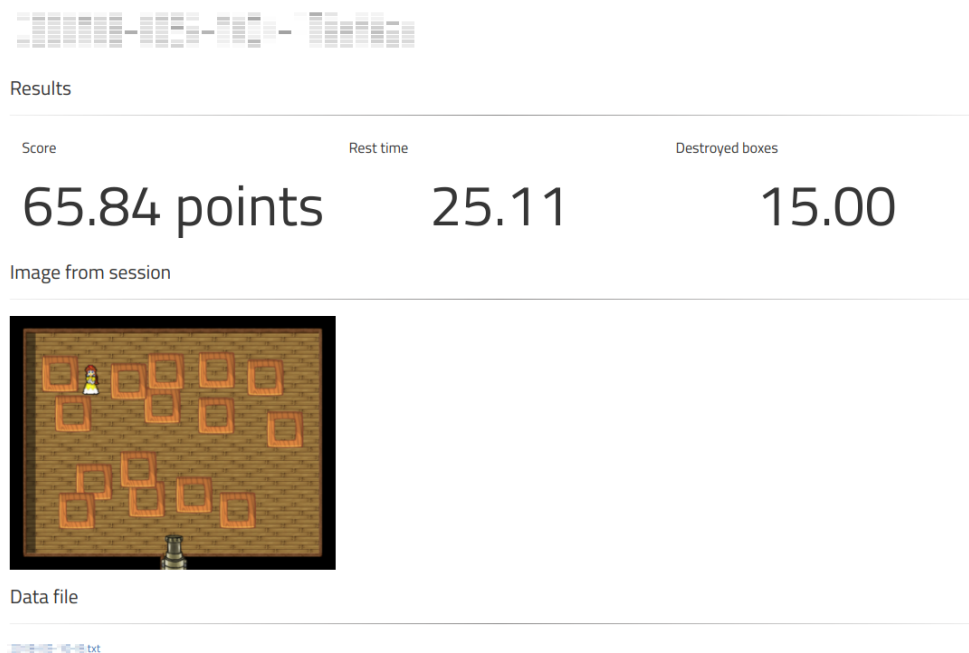
Aplikace obsahuje spouštit jednu aktivitu ve které se zobrazují 4 fragmenty:

- **MenuFragment** - fragment, který se zobrazí po spuštění aplikace, obsahuje tlačítko pro spuštění hry, tlačítko pro přihlášení, a pokud je uživatel již přihlášen, tak i jméno uživatele. Hru lze spustit i bez přihlášení.
- **LoginFragment** - obsahuje dva prvky `TextView` pro přihlašovací údaje a tlačítko pro přihlášení. Pokud je uživatel úspěšně přihlášen, je přeměrován zpátky na *MenuFragment*.
- **GameFragment** - v horní části zobrazuje aktuální úroveň pozornosti a zbývající čas ve vteřinách. Níže je umístěn prvek s názvem *GameView*, které dědí vlastnosti *SurfaceView*. Jedná se o plátno pro vykreslování, ve kterém bude hra probíhat.

- **MindwaveDisconnectedFragment** - pokud není připojeno zařízení Mindwave nebo pokud se odpojí, je zobrazen tento fragment, ve kterém je o odpojení uživatel informován.

Gameview při inicializaci vytvoří *Canvas*, do kterého jsou vykresleny textury pozadí, krabic, děla a princezny. Poté je spuštěn tzv *GameLoopThread*, ve kterém je ve vlastním vlákne spuštěna nekonečná smyčka, která vykresluje zaktualizovaný obraz (posun projektilu, výbuch apod.). Animace probíhá v taktu 35 FPS (Frame per second).

Po skončení hry se zobrazí dialog s výsledky a tlačítka (návrat do menu a restart). Zároveň se na pozadí s využitím knihovny *Communicator* odesílají DTO objekty *Session* a *SessionData* na server. *SessionDto* obsahuje skóre které závisí na zbylém čase a průměrné koncentraci. Dále počet zničených krabic a zbylý čas. *SessionDataDto* obsahuje bytové pole snímku hrací plochy, který je pořízen před začátkem hry a textový řetězec, který obsahuje všechny úrovně pozornosti, které byly nasnímány během sezení. Po odeslání je uživatel informován *Toust* hláškou. Data jsou ve webovém prohlížeči zobrazena takto:



Obrázek 9.6: Data ve webové aplikaci (anglická verze, anonymizované)

Přístup k těmto datům má jen terapeut. Datový soubor je ve formátu *.TXT*.

Během implementace této komponenty vzniklo 50 zdrojových souborů, které celkem obsahují 2993 řádek kódu.

9.6 Docker

Pro nasazení na server byly vytvořeny dva docker kontejnery a dvě podsítě. Jedna podsít pro komunikaci webového serveru (NGINX) a webové aplikace (s názvem *camp-webapp*). Druhá je použita pro komunikaci kontejneru *camp-mysql* (databázový server) a *camp-webapp*.

Pro kompilaci tzv. docker image je nutné vytvořit soubor s názvem *Dockerfile*. Dockerfile je textový dokument, který obsahuje příkazy, které se při kompilaci mají vykonat. Zde je příklad pro kompilaci *camp-mysql*.

```
FROM mysql:5.6

## The maintainer name and email
MAINTAINER camp <cognitive.amp@gmail.com>

ENV MYSQL_ROOT_PASSWORD=secret_password
ENV MYSQL_DATABASE=secret_database
ENV MYSQL_USER=secret_user
ENV MYSQL_PASSWORD=secret_password

RUN apt-get update && apt-get install -y \
  nano
# Copy the database schema to the /data directory
ADD db_cognitive_amp.sql run_db /tmp/
```

Obraz *camp-mysql* využívá obraz *mysql* s verzí 5.6, který je stažen z oficiálního repozitáře společnosti Oracle. Tento obraz je dále konfigurován příkazy ENV, RUN a ADD. ENV nastavuje proměnné prostředí. V tomto případě se jedná o heslo administrátora, název nového schématu a jméno a heslo nového uživatele, kterého bude využívat pro přístup k databázi webová aplikace. RUN vykonává příkazy v nově vytvořeném obrazu. V tomto dokumentu se pomocí RUN instaluje pouze textový prohlížeč "nano". ADD instrukce kopíruje soubory z hostujícího zařízení do obrazu. Zde se jedná o SQL skript, který obsahuje vytvoření triggerů a funkcí databáze, které byly popsány výše.

Příkazem "*docker build -t camp-mysql .*" se zkompiluje nový obraz, který je poté možné poslat na Docker cloud. Na serveru je poté nový obraz stažen. Spuštěn je následujícím příkazem:

```
docker run -d --rm --name camp-mysql -v \  
/mnt/camp/data:/var/lib/mysql cognitiveamp/camp-mysql
```

Z obrazu je spuštěn kontejner s názvem opět *camp-mysql*. K němu je připojen *volume*, které poskytuje perzistentní úložiště pro databázi na hostujícím stroji. Spuštěný kontejner se připojí do podsítě tímto příkazem:

```
docker network connect --ip 172.22.0.2 \  
camp-dbnet camp-mysql
```

Úložiště, tedy složka `"/mnt/camp/data"`, je pravidelně zálohováno. Nyní se záloha databáze nachází na stejném serveru a to není z bezpečnostních důvodů optimální. V budoucnu se však počítá s tím, že zálohy budou odesílány na jiné zařízení.

Konfigurace kontejneru *camp-webapp* probíhá obdobným způsobem, proto ji zde není nutné uvádět.

10 Testování

S vývojem softwaru je nutně spojen proces testování s cílem ověřit, zda produkt splňuje deklarované vlastnosti a chová se očekávaným způsobem. Dále je potřeba diskutovat dosažené výsledky z pohledu validace, tzn. zda výsledný produkt je smysluplný a pro dané účely použitelný. Systém *CAMP* byl testován v průběhu implementace tzv. unit (jednotkovými) testy, vytvořenými pro každou komponentu. Po ukončení vývoje proběhlo manuální testování dle testovacího scénáře. Dále bylo zkoumáno, jakou energetickou spotřebu mají vytvořené mobilní aplikace.

10.1 Manuální testování

Cílem manuálního testování bylo ověřit v praxi, zda systém *CAMP* poskytuje základní funkcionalitu bez vad i v případě, že jej testuje uživatel bez hlubších znalostí o konstrukci systému a informačních technologií jako takových. Další motivací bylo zjistit, jaký mají uživatelé názor na grafické rozhraní a případně získat nějaká doporučení na jeho změnu.

Nejdříve bylo nutné sestavit testovací scénář, dle kterého se bude testovací subjekt řídit. Skládá se z testů, do kterých jsou zahrnuty všechny komponenty systému. Poté se přešlo k samotnému testování na množině testovacích subjektů. Následně proběhlo vyhodnocení výsledků a provedení oprav/změn produktu.

10.1.1 Testovací scénář

Protokol z testování obsahuje stručný popis systému *CAMP*, informace o kontaktní osobě, poučení testera a testovací scénář. Ten se skládá z celkem 11 úkolů, které jsou prováděny na jednom desktopovém zařízení (označení PC) a dvou mobilních zařízeních (označení MOB). Úkoly jsou následující:

PC-1

V prohlížeči přejděte na stránku camp.kivzcu.cz.

PC-2

Zaregistrujte se do systému (do formuláře uveďte libovolný existující email).

PC-3

Ověřte svůj účet pomocí emailové schránky.

PC-4

Přihlaste se pomocí Vámi zadaných přihlašovacích údajů

PC-5

Změňte své heslo.

PC-6

Přejděte do sekce Moje skupiny a přidejte se do skupiny s názvem „Skupina-test“.

MOB-A1

Na mobilním zařízení spusťte aplikaci camp-hub a přihlaste se pomocí Vámi zvolených přihlašovacích údajů.

MOB-A2

Přejděte do sekce Skupiny a zkontrolujte, že jste členem skupiny s názvem „Skupina-test“.

MOB-A3

Přejděte do sekce Přidat skupinu a přidejte se do skupiny s názvem „Skupina-test-private“

MOB-B1

Nasaďte si čelenku Mindwave. Poté spusťte aplikaci Save the Princess a přihlaste se.

MOB-B2

Spusťte hru a úspěšně ji dokončete. Pokud skončí neúspěchem zkuste to znovu.

MOB-A4

Otevřete aplikaci Camp-hub a přejděte do sekce Přehled. Ujistěte se, že se v seznamu ve spodní části obrazovky zobrazí data o Vámi odehrané hře.

PC-6

Ve webové aplikaci na PC přejděte do sekce Moje výsledky, vyhledejte záznam o odehrané.

PC-7

Klikněte na tento záznam a zkontrolujte, zda se na stránce zobrazí obrázek ze hry a odkaz na textový soubor.

Ke každému úkolu tester napíše, zda byl úkol splněn bez problému a pokud ne, popíše důvod neúspěchu. Na konci dokumentu má tester prostor se vyjádřit v jakémkoliv ohledu. Může se jednat například o podmínky na zlepšení.

10.1.2 Průběh testování

Množina testovacích subjektů se skládala převážně ze studentů FZS (Fakulta zdravotnických studií) a FAV (Fakulta aplikovaných věd) na Západočeské univerzitě v Plzni. Jejich počet byl 20, což lze pokládat za dostačující v rámci možností plnění této diplomové práce. Seznámení testera s jeho úkoly a vyplnění jednoho testovacího protokolu trvalo přibližně 15 - 20 minut. Protože bylo nutné, aby testování probíhalo za přítomnosti obsluhy, například z důvodu zápisu nastalé chyby či sledování chování testera, proces manuálního testování se stal časově náročným. Dalším omezením byla nutnost použití specializovaného hardwaru (čelenka Mindwave).

Testování probíhalo celkem na 4 mobilních zařízeních a dvou osobních počítačích.

Popis osobních počítačů

Tabulka obsahuje technické parametry využívaných počítačů.

Označení	PC-1	PC-2
Operační systém	Windows 10	Windows 7 Professional
Procesor	Intel Core i7, 2.40 GHz	Intel Core i7, 2.40 GHz
RAM	8 GB	16 GB
Monitor	15"	22"

Tabulka 10.1: Specifikace osobních počítačů využitých při testování

Popis mobilních zařízení

Tabulka obsahuje technické parametry využívaných mobilních zařízení.

Označení	MOB-1	MOB-2	MOB-3	MOB-4
Model	Sony D6603	ASUS K01E	ASUS K00L	Sony M2
Verze OS	6.01	4.4.2	4.2.2	5.1
Procesor	Qualcomm 801	Qualcomm S4	Qualcomm Quad-Core	Qualcomm 400
RAM	3072 MB	1024 MB	1024 MB	1024 MB
Displej	5,2"	9,4"	8"	5"

Tabulka 10.2: Specifikace mobilních zařízení využitých při testování

10.1.3 Sumarizace výsledků a provedení oprav

Po dokončení testování bylo nutné vyhodnotit výsledky a navrhnout řešení nalezených problémů či nedostatků. V tabulce na další straně je vyčísleno vyhodnocení všech nalezených závad.

	Označeno jako chyba
Chyba	9
Není chyba	3

Tabulka 10.3: Vyhodnocení chyb

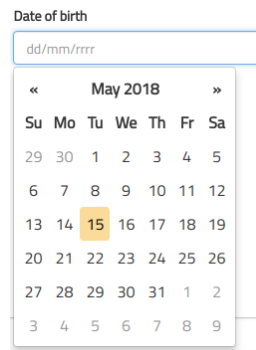
Tabulka zobrazuje, kolik jevů bylo opravdu chybných (TP - true positive) a počet jevů označených jako chyba, zatímco se o chybu nejednalo (FP - false positive). První skupina, tj. TP, jsou chyby, kterými je nutné se zabývat. Postupně zde budou uvedeny všechny chyby bez duplikátů ve formě:

označení úkolu/označení chyby či připomínky - stručný popis

PC-2/P1 - automatické doplňování lomítka v datu narození

Některým uživatelům se jevilo jako problém to, že je při registraci uživatele nutné do textového pole pro datum narození napsat datum ve formátu dd/mm/yyyy. Od jednoho testera zazněla připomínka, že by se mělo lomítko doplňovat automaticky.

Řešení: Textové pole pro datum narození bylo nahrazeno komponentou, která zobrazuje kalendář a textovou formu vygeneruje sama.



Obrázek 10.1: Komponenta pro výběr data

PC-5/E1 - uživatelské rozhraní nebyla nalezena možnost změny hesla

K tomuto úkolu bylo záměrně zadání formulováno vágně, a to "Změňte své heslo". Cílem bylo otestovat, že je uživatelské rozhraní v tomto směru navrženo správně. Ukázalo se, že stránku, kde je možné změnit své heslo, měli problém najít 3 testéři.

Řešení: Vzhledem k tomu, že problém měl nízký počet testerů, grafické rozhraní není nutné upravovat.

MOB-A1/E2 - nelze se přihlásit s uživ. jménem obsahující souhlásku s diakritikou

Po registraci, kde tester zadal jako své uživatelské jméno, se již nešlo přihlásit.

Řešení: Regulární výraz pro kontrolu uživatelského jména byl chybný. Nyní opraveno a čeká na další testování.

MOB-B2/E3 - krabice umístěna za princeznou

Pro dokončení hry je nutné zničit všechny krabice. Jedna byla však umístěna vůči dělu za princeznou. Generování náhodných pozic s touto možností počítá, avšak konstanta, která určuje minimální vzdálenost není dostatečně velká.

Řešení: Zvýšila se minimální vzdálenost krabice od princezny.

PC-2/E4 - HTTP status 500 po zadání datumu narození ve formátu dd.mm.rrrr

Uživatel zadal datum narození ve špatném formátu (správný formát je dd/mm/rrrr). Pomocí možnosti, kterou poskytuje HTML5 je před odesláním každý textový vstup zkontrolován regulárním výrazem. Na serveru je kontrola opět zopakována. Přesto však chyba nastala.

Řešení: Regulární výraz pro kontrolu datumu byl chybný na straně serveru i v rámci vstupní HTML komponenty. Opraveno novou komponentou pro zadávání data narození.

PC-4/E5 - nelze se přihlásit pomocí hesla s diakritikou

Čtyři uživatelé se nemohli přihlásit, pokud si zvolili heslo s českou diakritikou.

Řešení: Stejně řešení jako u chyby E2.

Téměř všechny chyby, které byly vyhodnoceny jako nechtěná vlastnost systému a byly opraveny. Chyba s označením E1, se týkala orientace v GUI. Protože tento problém hlásilo pouze malé množství respondentů, prozatím změny ve webové aplikaci provedeny nebudou. Nicméně při budoucím vylepšování systému, bude toto zohledněno. Manuální testování lze považovat za úspěšné, protože bylo nalezeno malém množství chyb a jejich oprava nebyla náročná.

10.2 Automatické testování

Z důvodu rozsáhlosti implementace (téměř 16 000 řádek kódu) se nepodařilo systém pokrýt plně jednotkovými testy. Nicméně byla vytvořena sada testů pro každou komponentu, která ověřovala správnost kritických funkcí.

U mobilních aplikací se validují zejména funkce pro komunikaci se serverem.

Webová aplikace pak obsahuje jednotkové testy, které kontrolují integritu zásadních dat v databázi (administrátorský účet, definice rolí a definice úrovní vzdělávání). Dále je z větší části testována funkce objektově-relačního mapování, kdy je získáván z databáze objekt každé entity. Webová aplikace byla průběžně testována i pomocí frameworku Selenium, kdy bylo vytvořeno pět testovacích scénářů (přihlášení, registrace, změna hesla, vytvoření skupiny, přidat skupinu).

10.3 Test energetické náročnosti mobilních aplikací

Nástrojem Android Debug Bridge (ADB) byla sledována energetická náročnost mobilních aplikací Save the Princess a Camp-Bub. Jedná se o velmi důležitou charakteristiku mobilních aplikací. Cílem je samozřejmě spotřebu minimalizovat. Vysoká spotřeba může také indikovat zbytečně vysoké využívání zdrojů, což se dá považovat za chybu. Pro každou aplikaci byla měřena spotřeba v 5 minutových intervalech desetkrát. Hodnoty byly poté zprůměrovány. Měření bylo provedeno na mobilní telefonu s označením MOB-4, který disponuje baterií s kapacitou 2400 mAh.

Save the Princess

Odhadovaná spotřeba | 18,1 mAh

Camp-Hub

Odhadovaná spotřeba | 11,6 mAh

Naměřená s průměrovaná data byla poté porovnána s ostatními nainstalovanými aplikacemi a hodnoty byly srovnatelné. Pouze u aplikace Save the Princess je spotřeba vyšší. Vzhledem k tomu, že k vykreslování je používáno SurfaceView, které vykresluje pomocí GPU je tento jev pochopitelný.

10.3.1 Zhodnocení testování

Testování systému byl úspěšné. Verifikací systému byly nalezeny chyby, které byly následně odstraněny. Zjištění, zda systém se dá využít k účelu, ke kterému byl navržen, bylo ověřeno předvedením implementovaného systému ergoterapeutce Bc. Vladimíře Frouzové. Ta pokládala vytvořené funkcionality a využitelné. Podmínkou však je, aby systém nabízel dostatečné množství aplikací. Dále byl systém validován při manuálních testech. Mezi testovanými osobami byli i studenti FZS, kteří se v dané problematice z odborného hlediska také orientují.

11 Diskuze nad dosaženými výsledky

Systém *CAMP* byl úspěšně implementovaný dle navržené architektury. Nyní je webová aplikace spuštěna pod doménou *camp.kiv.zcu.cz*, kde je připravena po budoucí použití. Aplikace pro Android jsou přidány do Google Play Store, kde jsou zdarma dostupné pro instalaci. Všechny třídy jsou okomentovány v angličtině a byly také vytvořeny programátorské manuály. Dále byl zkompletoována složka určená pro vývojáře kognitivních her, která obsahuje instrukce pro integraci se systémem pomocí knihovny *Communicator* nebo REST API. V průběhu vytváření softwaru se nevyskytl žádný závažný problém spojený s implementací nebo návrhem architektury. Nicméně bylo nutné přistoupit k některým kompromisům, a to hlavně z důvodu velkého rozsahu práce.

Přestože proběhlé testování proběhlo vcelku uspokojivě, je nutné zmínit, že veškeré funkce systému nejsou pokryty jednotkovými testy. Existují pouze pro kritické funkce, tak aby byla zajištěna stabilita zejména webové aplikace. Pro budoucí rozšiřování a zdokonalování systému může být toto nevýhodou. Přidáváním nových funkcí mohou ty stávající degradovat.

Za další slabinu lze považovat grafické rozhraní aplikace *Save the Princess*, které z estetického hlediska nepovažuji z úplně povedené. Primární cílem byl v průběhu plnění vytvořit funkční kód a jelikož se celý systém skládá z velkého počtu řádek kódu (webová aplikace, knihovna, dvě mobilní aplikace), byla implementace velmi časově náročná a musel jsem dát přednost právě tomuto cíli.

Nicméně vytvořený systém je použitelný v rámci laboratoře a je připraven k integraci s dalšími aplikacemi pro kognitivní trénink. V rámci reálné kognitivní terapie však využít zatím nejde. Jedním důvodem je nedostačný počet aplikací a dalším je právní aspekt celého procesu. Je nutné, aby systém prošel auditem z hlediska právního. Jsou zde totiž nejasnosti ohledně uchovávání osobních údajů, přestože jsou data zabezpečena před vnějším zneužitím. Další otázkou je, zda je nutné upravovat nějakým způsobem právní vztah mezi provozovatelem systému a terapeutem. Terapeut je totiž vázán zachováváním lékařského tajemství.

Integrace jako taková byla otestována už při vývoji aplikace *Save the Princess*. Po ukončení vývoje vytvořil student Bc. Michal Horký novou kognitivní hru pro Android s názvem Neuro-skipping Stone, která je do systému také přidána. Nyní jsou v systému dvě funkční kognitivní hry.

12 Doporučení pro rozšíření systému

Systém *CAMP* bude v průběhu dalších let spravován a rozšiřován o další funkce. Nyní zde budou uvedeny některé funkce či nedostatky, které je možné v budoucnu implementovat.

- **Doplnění dalších statistických ukazatelů a grafů** - ve webové aplikaci je možné si zobrazit výsledky uživatele nebo skupiny. Nyní je zde zobrazován percentil, podíl hraných her a percentil pro každou skupiny dosaženého vzdělání zvlášť. Tuto sadu lze snadno doplnit o další údaje. K tomu je nutné vytvořit příslušnou funkci v databázi, tak aby byla komunikační režie co nejmenší, a poté získaná data zobrazit ve webové aplikaci.
- **Vyřešení právní otázky** - proces získávání a uchovávání dat je auditovat z právního hlediska. Poté bude možné systém použít v ergoterapeutické praxi.
- **Notifikace plánovaného cvičení (sezení)** - terapeut by v rámci skupiny určil, jak často mají dané cvičení provést. Aplikace *Camp-Hub* oznamovala pomocí notifikací, že je nutné cvičení provést.
- **Notifikace pozvánky/žádosti vstupu do skupiny** - pokud by uživatel poslal žádost ke vstupu do skupiny nebo by terapeut poslal pozvánku do skupiny, systém by mohl adresátovi podat o tom informaci. Buď formou emailu nebo formou notifikace v systému Android.
- **Vytvoření aplikací pro testování KF** - v kapitole 2.3 *Metody hodnocení kognitivních funkcí* jsou zmíněny nejpoužívanější testy, které je možné implementovat do podoby mobilních aplikací. Je však nutné, vyřešit otázku autorských práv těchto metod.
- **Vytvoření kompletní sady jednotkových testů** - v předchozí kapitole bylo zmíněno, že nejsou vytvořeny jednotkové testy pro všechny funkce systému. Při rozšiřování by tak mohl nastat problém s již implementovanými funkcemi.

- **Vytváření záloh databáze na dalším serveru** - nyní jsou pravidelně prováděny zálohy databáze, které se ukládají na stejný server. Zálohy je vhodné odesílat na jiný server. Je však zároveň nutné zohledňovat bezpečnostní aspekt tohoto procesu.

Závěr

Cílem této diplomové práce bylo navrhnout a vytvořit systém pro správu a distribuci aplikací pro kognitivní trénink a mobilní aplikaci pro kognitivní trénink, která bude využívat snímač EEG, členku Neurosky Mobile.

V analytické části je uveden vytvořený teoretický aparát, který zahrnoval popis mozkové aktivity a její snímání, kognitivních funkcí a kognitivních her vytvořených na Katedře informatiky a výpočetní techniky.

Poté dle požadavků zadavatele byla vytvořena architektura systému Cognitive AMP (CAMP), která byla následně úspěšně implementována. Systém *CAMP* je složen z komponent: webová aplikace (*web-app*), knihovny pro komunikaci s webovou aplikací (*Communicator*), mobilní aplikací pro správu uživatelských skupin a distribuci kognitivních her (*Camp-Hub*). Dále byla implementována mobilní aplikace s názvem *Save the Princess*, která slouží pro kognitivní trénink. Pro ovládání využívá členku Mindwave Mobile, kdy je na základě úrovně pozornosti odvozen výsledek cvičení. Tato mobilní aplikace po skončení cvičení odesílá o něm záznam společně s daty, která je poté možné ve webové aplikaci analyzovat.

Terapeut, který vede rehabilitaci má kontrolu nad postupem svých pacientů při i po vykonávání jednotlivých cvičení. Jednotlivá cvičení je možné prohlížet a analyzovat pomocí vytvořeného výpočtu statistických ukazatelů a grafů.

Nové aplikace pro kognitivní trénink je možné pomocí knihovny *Communicator* jednoduše integrovat se systémem *CAMP*. Integrace způsobí v nové aplikaci nárůst příkazů jen řádu jednotek. Použití této knihovny je snadné, což bylo cílem.

Webová aplikace je na serveru spuštěna pomocí služby Docker, což je výhodné z hlediska jednoduché správy serveru. V případě nutnosti je možné přesunout webovou aplikaci nebo databázovou službu během několika málo úkonů.

Systém se skládá z pěti částí (komponent), pro které bylo vytvořeno celkem 310 zdrojových souborů s téměř 16 000 řádky kódu. Z toho plyne, že se jednalo o velmi časově náročnou práci.

Veškeré body zadání diplomové práce byly splněny.

Literatura

- [1] *MindWave mobile* [online]. Silicon Valley, Kalifornie: Neurosky, Inc., 2015. [cit. 2016/03/09]. Dostupné z: store.neurosky.com/products/mindwave-1.
- [2] *What are the average dimensions of a synapse* [online]. Quora, 2012. [cit. 2018/03/10]. Dostupné z: <https://www.quora.com/What-are-the-average-dimensions-of-a-synapse>.
- [3] *TGAM1 Chip* [online]. Pittsburgh, Pennsylvanie: Xeno Innovations, Inc., 2011. [cit. 2016/03/09]. Dostupné z: http://xenoinc.org/xi.wiki/index.php?title=TGAM1_Chip.
- [4] *Druhy mozkových vln* [online]. Mindtrix, 2015. [cit. 2018/03/11]. Dostupné z: <http://mindtrix.cz/hladina-alfa/>.
- [5] BLAŽEK, V. *Zaklady neurofyzologie a neuroanatomie cloveka*. Aleš Čeněk, 2006. ISBN 80-86898-63-6.
- [6] *KRÁTKÉ KOGNITIVNÍ TESTY* [online]. AD Centrum, 2017. [cit. 2017/28/02]. Dostupné z: http://www.nudz.cz/adcentrum/kratke_kognitivni_testy.html.
- [7] HARL – HARTLOVÁ, P. – HELENA. *Psychologický slovník*. Portál, 2000. ISBN 80-7178-303-X.
- [8] JELÍNEK, R. – AL. [online]. 3. lékařská fakulta UK., 2006. Dostupné z: <http://histologie.lf3.cuni.cz/histologie/materialy/doc/skripta.pdf>.
- [9] JUNG, C. *Neuroscience*. Princeton University Press, 1971. ISBN 0-691-01813-8.
- [10] KLUCKA – VOLFOVÁ, J. – PAVLA. *Kognitivní trénink v praxi*. Grada Publishing, a.s., 2009. ISBN 978-80-247-2608-3.
- [11] KRIVOŠÍKOVÁ, M. *Úvod do ergoterapie*. Grada Publishing, a.s., 2011. ISBN 978-80-247-2699-1.
- [12] LAIR, S. *Trénink paměti: principy, metody a cvičení pro využití a rozvoj paměti*. Portál, 2011. ISBN 978-80-7367-902-6.

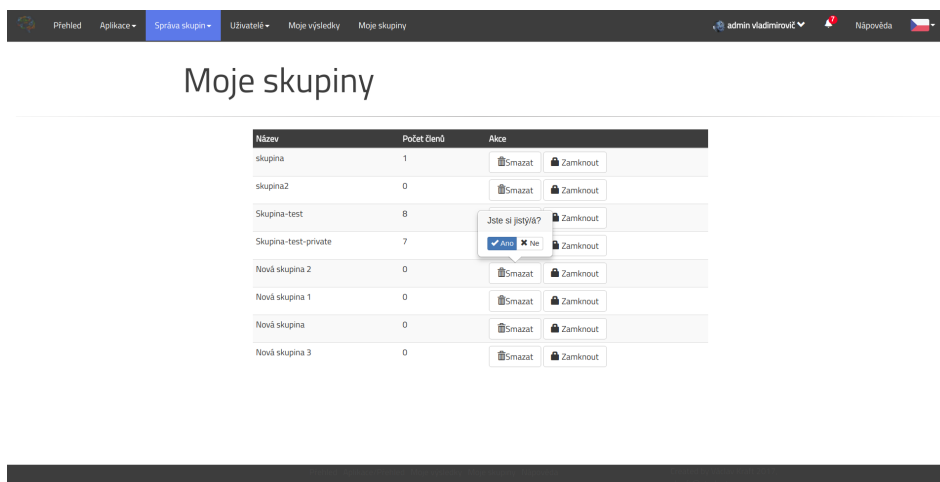
- [13] MALIA – BRANNAGAN, K. – ANNE. *Jak provádět trénink kognitivních funkcí: praktická příručka pro každého*. Cerebrum, 2010. ISBN 978-80-904357-3-5.
- [14] NAVRÁTIL, J. *GDPR PRO PRAXI*. Vydavatelství a nakladatelství Aleš Čeněk s.r.o., 2018. ISBN 978-80-7380-689-7.
- [15] NULÍČEK, M. *GDPR / Obecné nařízení o ochraně osobních údajů*. Wolters Kluwer ČR a.s., 217. ISBN 979-807552-765-3.
- [16] PAVLOVSKÝ, J. – PAVEL. *Psychiatrie: minimum pro praxi*. Triton, 2001. ISBN 80-7254-156-0.
- [17] PREISS – KŘIVOHLAVÝ, M. – JARO. *Trénování paměti a poznávacích schopností*. Grada Publishing, a.s., 2009. ISBN 978-80-247-2738-7.
- [18] PURVES, D. *Psychological Types, Collected Works, Volume 6*. Princeton University Press, 2007. ISBN 978-0-87893-697-7.
- [19] SUCHÁ, J. *Trénujte si paměť*. Portál, 21. ISBN 978-80-7367-791-6.
- [20] TOPINKOVÁ et al. [online]. 2002. [cit. 2017/27/02]. Dostupné z: <http://www.cls.cz/dokumenty2/os/t271.rtf>.
- [21] *Metakognice* [online]. AKTUGRAM, 2016. [cit. 2017/25/2]. Dostupné z: <https://turbo.cdv.tul.cz/mod/book/view.php?id=5965&chapterid=6237>.

Seznam použitých zkratek

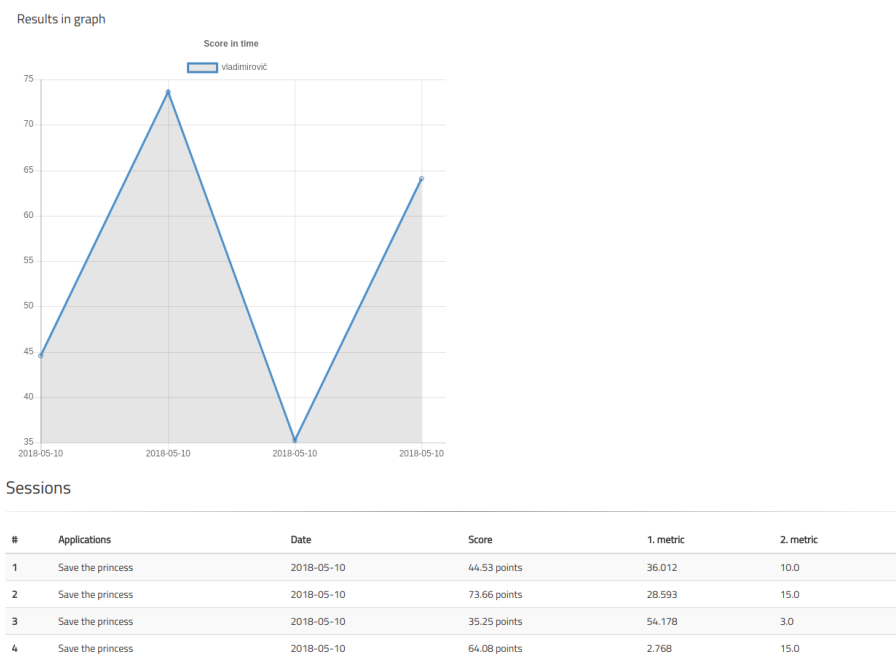
CAMP	Cognitive Amp
CNS	Centrální nervová soustava
CT	Computed Tomography
DFD	Diagramy datových toků
EEG	Elektroencefalografie
fMRI	Funkční magnetická rezonance
JVM	Java Virtual Machine
KF	Kognitivní funkce
KIV	Katedra informatiky a výpočetní techniky
MMSE	Mini-Mental State Examination
MoCA-CZ	Montrealský kognitivní test – 2012
MVC	Model-View-Controller model
MST	Sedmiminutový screeningový test
ORM	Objektově-relační mapování
PET	Pozitronová emisní tomografie
REM	Rapid eye movement
TKF	Trénink kognitivních funkcí
ZČU	Západočeská univerzita v Plzni

Přílohy

Webová aplikace - obrázky

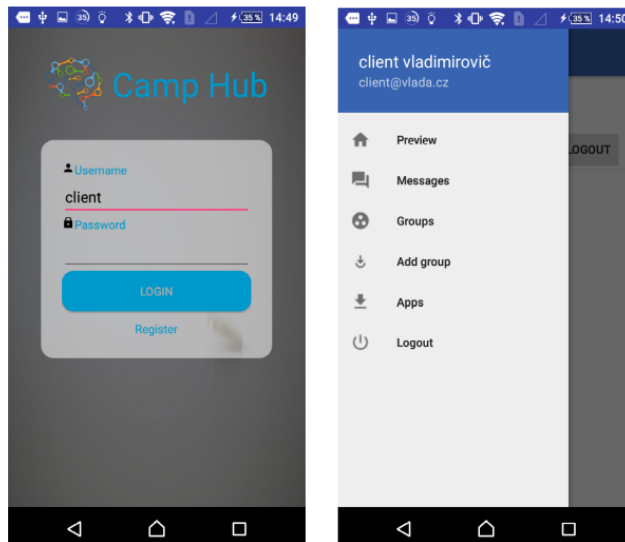


Obrázek 12.1: Správa skupin



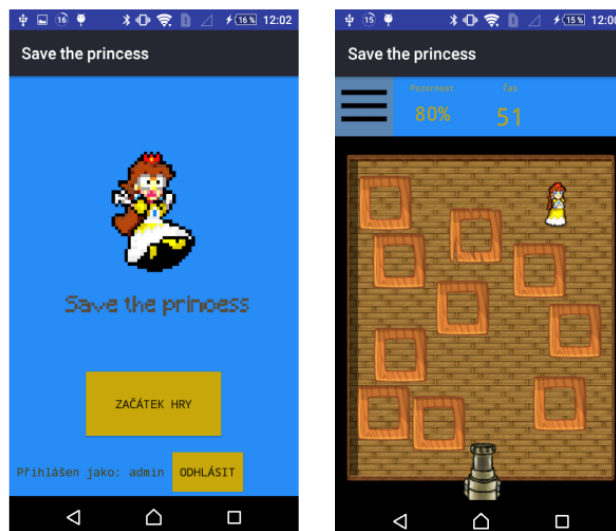
Obrázek 12.2: Výsledky uživatele

CampHub - obrázky



Obrázek 12.3: Camp-Hub - GUI

Save the Princess - obrázky



Obrázek 12.4: Save the Princess - GUI

Příručka pro nasazení systému na serveru

Toto je příručka, která obsahuje návod k zprovoznění vytvořené webové aplikace a databáze na serveru. Na Docker Cloud pod účtem *cognitiveamp* jsou vloženy dva kontejnery. Jeden pro databázi (camp-mysql) a druhý obsahuje webovou aplikaci (camp-webamp). Pro nasazení následujte tyto instrukce. Přihlašte se na server s uživatelem, který má *root* oprávnění a postupně zadejte následující příkazy:

camp-mysql

1. `mkdir -p /mnt/camp/data`
2. `docker network create --ip-range 172.22.0.0/16 --subnet 172.22.0.0/16 camp-dbnet`
3. `docker run -d --rm --name camp-mysql -v /mnt/camp/data:/var/lib/mysql cognitiveamp/camp-mysql --character-set-server=utf8 --collation-server=utf8_general_ci`
4. `docker network connect --ip 172.22.0.2 camp-dbnet camp-mysql`
5. `cd tmp`
6. `mysql -u root -p`
7. `use cognitiveamp;`
8. `source db_cognitive_amp.sql;`

camp-webamp

1. `docker run -it -d --rm -p 9090:8080 --network=camp-dbnet --ip 172.22.0.3 --name camp-webapp cognitiveamp/camp-webapp`
2. `docker exec -it camp-mysql bash`
3. `rm -r webapp/ROOT`

Nyní je webová aplikace spuštěna a naslouchá na portu 9090.

Příručka pro integraci mobilní aplikace

Pro integraci mobilní aplikace lze zvolit ze dvou možností. Pokud je Vaše kognitivní hra psaná v Javě nebo Kotlinu, je výhodné využít knihovnu *Communicator*. Pokud ale používáte jiný jazyk, máte možnost přístupu k REST API na serveru. V tomto dokumentu je popsáno pouze použití knihovny. Ve složce přiložené k tomuto souboru je knihovna ve formátu *.jar* a příkladový projekt.

Nejdříve do svého projektu importujte knihovnu s kořenovým adresářem *kiv.vkraft.communicator*. Vytvořte instanci třídy *Communicator* a poté může využívat tyto funkce, které jsou v syntaxi jazyka Kotlin. Pro jazyk Java jsou obdobné.

- **login(username : String, password : String) : UsersDt** - funkce pro přihlášení do systému pomocí uživatelského jména a hesla. Úspěšné přihlášení je podmínkou pro přístup do systému. Funkce vrátí DTO objekt přihlášeného uživatele.
- **getUserById(userId: Long): UsersDto** - vrátí DTO objekt na základě zadaného uživatelského ID.
- **setApiKey(appKey : String)** - nastavení aplikačního klíče, který získáte od správce systému. Je nutnou podmínkou pro odesílání dat o proběhlých cvičeních uživatele (*SessionDto* a *SessionDataDto*).
- **sendSession(sessionDto: SessionDto): Long** - pošle sezení, které obsahuje číselné hodnoty: skóre, první volitelná metrika. druhá volitelná metrika. Funkce vrátí ID nově přidaného sezení.
- **sendSessionData(sessionId : Long, sessionDataDto: SessionDataDto)** - odesílá doplňující data k sezení. DTO objekt *SessionData* obsahuje obrázek ve formě bytového pole a textový řetězec (například pro medicínská data).
- **getSessions():List<SessionDto>** - vrátí všechna sezení přihlášeného uživatele.
- **getGroups(): List<GroupsDto>** - vrátí DTO objekty skupin, ke kterým je uživatel přihlášen.
- **getApps(): List<AppDto>** - vrátí všechny aplikace v systému.
- **getAvailableGroups(): Map<Long, String>** - vrátí všechny skupiny do kterých se může uživatel přihlásit.

- **logoutGroup(groupId : Long) : Boolean** - odhlášení uživatele.
- **addGroup(groupId : Long) : Boolean** - přihlášení uživatele ke skupině.
- **getMessages(): List<MessageDto>** - vrátí všechny nevyřízené správy (požadavky).
- **messageAnswer(messageId : Long, answer : Boolean): Boolean** - funkcí lze odpovědět na nevyřízenou zprávu. Lze ji buď potvrdit nebo zamítnout.

Seznam obrázků

1.1	Schéma neuronu[2]	11
1.2	Schéma synapse	13
1.3	Mozkové vlny	16
4.1	Hry pro kognitivní trénink	31
5.1	Schéma MindWave Mobile[1]	32
7.1	Diagram případů užití	41
8.1	Komponentový diagram	45
8.2	Komponenta Server	46
8.3	Schéma komunikace docker kontejnerů	49
8.4	CAMP - návrh GUI	50
8.5	Komponenta Communicator	50
8.6	Komponenta Hub	52
8.7	Camp-Hub - GUI	53
8.8	Komponenta Save the Princess	54
8.9	Save the Princess - GUI	56
8.10	Komunikační schéma	57
8.11	Diagram datových toků	58
9.1	Navržený ERA model	60
9.2	Obsluha požadavku	66
9.3	Detail skupiny	67
9.4	Zobrazení aplikací	69
9.5	Herní pole	71
9.6	Data ve webové aplikaci (anglická verze, anonymizované)	72
10.1	Komponenta pro výběr data	79
12.1	Správa skupin	91
12.2	Výsledky uživatele	91
12.3	Camp-Hub - GUI	92
12.4	Save the Princess - GUI	92

Seznam tabulek

5.1	Specifikace Mindwave Mobile[1]	33
10.1	Specifikace osobních počítačů využitých při testování	78
10.2	Specifikace mobilních zařízení využitých při testování	78
10.3	Vyhodnocení chyb	78