

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

Diplomová práce

Mobilní aplikace pro monitoring pohybu a elektrofyziologických potenciálů

Plzeň, 2015

Bc. Jan Krupička

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

V Plzni dne 1. 5. 2015

Jan Krupička

Abstract

Health is an important topic for mankind. By its improvement or preservation we extend the period of quality life. It is appropriate to base resolution of health problems, research or prevention in the form of regular physical activities on accurate and numerous mensuration of health condition. There are many health indicators and body responses we can measure today. As the availability of personal sensors keeps improving, the mensuration is gradually moving to homes or activity execution places of the users. The popularity of smart devices allows to collect this data while directly analyze it or send it to remote server on the internet. Immediate response motivates to improve one's lifestyle and measured data can be used to establish correct diagnosis or for research.

The aim of this work is to get acquainted with existing applications for monitoring of physical activities and communication with health sensors. Further to get familiar with several provided sensors of different types and find out the possibilities of communication and data retrieval with mobile device. The main goal is on the base of acquired findings to design and implement mobile application for Android platform. It will allow collection of sensor data, its storage, basic visualization and analysis. Further in case of available internet connection it will allow send data to EEG/ERP Portal also developed on the local university.

Abstrakt

Zdraví je pro lidstvo důležité téma. Jeho zlepšováním či udržováním prodlužujeme dobu kvalitního života. Pro řešení zdravotních problémů, výzkum nebo prevenci v podobě pravidelných fyzických aktivit je vhodné vycházet z měření zdravotního stavu. Dnes lze měřit různé zdravotní ukazatele a odezvy našeho těla na vnější podněty. Se stále se zlepšující dostupností osobních senzorů se tato měření pozvolna přesouvají z ordinací do domácností či míst vykonávání aktivit uživatelů. Obliba chytrých zařízení dovoluje tato data sbírat a přímo analyzovat nebo je odeslat k analýze na server v internetu. Okamžitá odezva motivuje uživatele ke zlepšení životního stylu a naměřená data mohou být použita pro správnou diagnózu nebo výzkum.

Cílem této práce je seznámit se s existujícími aplikacemi pro monitorování fyzické aktivity a pro komunikaci se zdravotními senzory. Dále se seznámit s několika poskytnutými senzory různého typu a zjistit možnosti komunikace a získání dat mobilním zařízením. Hlavním cílem je na základě nabytých poznatků navrhnout a implementovat mobilní aplikaci pro systém Android, která umožní sběr dat ze senzorů, jejich uložení, základní vizualizaci a analýzu a v případě dostupného internetového připojení odeslat data na EEG/ERP Portál vyvíjený také na zdejší univerzitě.

Obsah

Úvod	1
1 Osobní zdravotní senzory	3
1.1 Využití	3
1.2 Rozmanitost	4
2 Transport dat	6
2.1 Topologie	6
2.1.1 Jednovrstvá	6
2.1.2 Dvouvrstvá	7
2.1.3 Třívrstvá	8
2.2 Komunikační rozhraní	9
2.2.1 Bluetooth	9
2.2.2 ANT	9
2.2.3 Wi-Fi	10
3 Existující aplikace	11
3.1 Endomondo	11
3.2 Runtastic	13
3.2.1 Pedometer	14
3.2.2 Push-Ups	15
3.2.3 Heart rate	15
3.3 Sports Tracker	15
3.4 Health Tracker	15
3.5 Madbarz	16
3.6 eVito	17
4 EEG/ERP Portál	19
4.1 Architektura	19
4.1.1 Datová vrstva	19
4.1.2 Aplikační vrstva	20
4.1.3 Prezentační vrstva	20

4.1.4	Webové služby	20
5	Zpracovávané senzory	22
5.1	Krokoměr	22
5.1.1	Uživatelský pohled	22
5.1.2	Měření.....	23
5.1.3	Získání dat	23
5.1.4	Poskytovaná data	25
5.2	Tlakoměr	25
5.2.1	Uživatelský pohled	25
5.2.2	Měření.....	26
5.2.3	Získání dat	27
5.2.4	Poskytovaná data	29
5.3	Hrudní pás	29
5.3.1	Uživatelský pohled	29
5.3.2	Měření.....	30
5.3.3	Získání dat	30
5.3.4	Poskytovaná data	30
5.4	Váha	31
5.4.1	Uživatelský pohled	31
5.4.2	Měření.....	31
5.4.3	Získání dat	32
5.4.4	Poskytovaná data	33
5.5	Glukoměr	33
5.5.1	Uživatelský pohled	33
5.5.2	Měření.....	34
5.5.3	Získání dat	34
5.5.4	Poskytovaná data	35
6	Mobilní aplikace.....	36
6.1	Popis činnosti aplikace.....	36
6.1.1	Uživatelské profily	37

6.1.2	Obsluha senzorů	38
6.1.3	Manipulace s uloženými daty	41
6.1.4	Nastavení	42
6.2	Architektura	42
6.2.1	Hlavní balík <code>cz.zcu.kiv.mobile.logger</code>	43
6.2.2	Balík <code>common</code>	44
6.2.3	Balík <code>data</code>	44
6.2.4	Balík <code>devices</code>	45
6.2.5	Balík <code>eegbase</code>	46
6.2.6	Balík <code>layout</code>	47
6.2.7	Balík <code>profiles</code>	47
6.2.8	Balík <code>service</code>	47
6.2.9	Balík <code>settings</code>	48
6.2.10	Balík <code>sync</code>	48
6.2.11	Balík <code>utils</code>	48
6.3	Práce s databází	48
6.3.1	Objekty typu <code>Loader</code>	49
6.3.2	Objekty typu <code>AsyncTask</code>	50
6.4	Komunikace se senzory	50
6.4.1	Hrudní pás	50
6.4.2	Váha	51
6.4.3	Tlakoměr	52
6.4.4	Glukoměr	54
6.5	Komunikace se serverem	54
6.5.1	Ruční upload	55
6.5.2	Automatický upload	55
6.6	Základní analýza dat	56
7	Úprava EEG/ERP Portálu	59
7.1	Ukládání dat	59

7.1.1	Interceptor.....	60
7.1.2	DAO metoda update	60
7.2	REST služby	61
7.2.1	Získání seznamu experimentů	61
7.2.2	Upload měření	62
7.3	Uživatelské rozhraní pro parametry.....	63
7.3.1	Wicket.....	63
7.3.2	Úprava stránky experimentu.....	63
7.3.3	Vytvoření stránky parametru.....	64
8	Zhodnocení výsledků	66
9	Závěr	68
	Reference	70
	Seznam obrázků.....	74
	Seznam tabulek	76
	Příloha A: Obrázky	77

Úvod

Pro jakýkoli organismus je důležitý stav, který označujeme jako zdraví. Ač je zdraví pro každý organismus stejně důležité, z pohledu lidstva se o něj zajímáme tím víc, o čím vyšší formu života se jedná a nejvíce se zajímáme, přirozeně, o své vlastní.

Z povahy zdraví, jež ovlivňuje množství faktorů od vnitřního stavu jedince, přes genetiku až po vliv prostředí, vyplývá, že není snadné všechny aspekty měřit. Dobře, objektivně a relativně jednoduše to jde hlavně u přímé fyzické odezvy těla, jako je např. krevní tlak, obsah cukru v krvi nebo srdeční tep.

V dnešní době chytrých zařízení v čele se smartphony se tato měření pozvolna přesouvají z ordinací do působnosti laických uživatelů. Podle serveru eMarketer [1] v roce 2014 vlastnilo chytrý telefon 1.64 miliard lidí a na konci letošního roku jich mají být již téměř dvě miliardy. Zároveň je na trhu spousta senzorů na měření tělesných veličin, které dokážou naměřená data poskytovat bezdrátovým způsobem přenosu a lze je tak smartphony číst a zpracovávat.

Takto získaná data se dají uživatelem fundamentálně využívat dvěma prolínajícími se způsoby. Asistivní, resp. asistenční využití slouží pro posílení, udržení nebo zlepšení funkční schopnosti jedinců se speciálními potřebami, a tím usnadnění jejich každodenních povinností, zlepšení kvality života, samostatnosti a soběstačnosti (více pro stárnoucí a handicapovanou populaci). Monitoring slouží pro sběr dat, která využívá pro zobrazení aktuálního stavu nebo pro zpětnou analýzu (sport a výzkum).

Mobilní aplikace, která je výstupem této práce se soustředí na druhý způsob, sbírá data ze zdravotních senzorů, zobrazuje je uživateli a provádí jejich základní analýzu. Navíc umožňuje nasbíraná data odesílat do EEG/ERP Portálu [2]. Tato data budou poskytovat dodatečné informace při vyhodnocování experimentů.

V kapitole 1 jsou popsány osobní senzory, osvětlení jejich prospěšnosti a rozmanitosti z hlediska měřených veličin i způsobů měření. Kapitola 2 se věnuje

transportu dat, druhům používaných topologií a komunikačních rozhraní. Kapitola 3 seznamuje s existujícími mobilními aplikacemi, které slouží k monitorování fyzické výkonnosti a pohybové aktivity člověka, případně čtou údaje ze senzorů a zpracovávají je. V kapitole 4 je stručně popsán EEG/ERP Portál, který musel být upraven pro příjem a ukládání dat z mobilní aplikace. Kapitola 5 popisuje osobní senzory, které byly zkoumány za účelem získání jimi naměřených dat. Dále je zde uvedeno ovládání každého senzoru, jak probíhá měření, jakým způsobem probíhá získání dat a jaká data jsou k dispozici. V kapitole 6 je popsána mobilní aplikace, která je výstupem této práce. Je zde popsáno ovládání aplikace, její architektura, práce s databází, implementace komunikace se senzory a komunikace se serverem. V kapitole 7 je popsáno, jak server komunikuje s mobilní aplikací, kam ukládá její data a jakou s nimi dovoluje uživateli interakci.

1 Osobní zdravotní senzory

Neustávající vývoj elektroniky a informačních technologií dává vzniknout stále novým osobním zdravotním sensorům. Díky miniaturizaci elektroniky a pokročilým výrobním postupům se zmenšují jejich rozměry, klesá jejich energetická spotřeba i prodejní cena.

Spotřebu snižuje i vývoj komunikačních technologií (jako jsou Bluetooth [11] či ANT [12]) a to jak hardwarovou, tak i softwarovou implementací. Spolu s vývojem akumulátorů se díky tomu prodlužuje doba činnosti těchto zařízení na jedno nabití či výměnu baterie.

1.1 Využití

Ve dvacátém století s vývojem zdravotnictví a zdravotnických nástrojů nebylo buď prakticky možné provádět vyšetření u pacienta nebo to bylo logisticky a ekonomicky nevýhodné. Proto se péče o zdraví začala soustřeďovat do nemocnic. Tím se spíše zaměřovalo na léčení nemocí než na udržování zdraví [13].

V průběhu dospělého života vyžaduje člověk různou péči o zdraví. V produktivním věku je zapotřebí minimum vyšetření a preventivních prohlídek. Ve věku mezi 55-75 lety stále stačí nezávislá péče o zdraví. Mezi 75-80 lety již může na řadu přijít asistence a denní monitoring aktivity a zdravotního stavu jedince. Ve věku nad 80 je nezdědka zapotřebí nepřetržitá péče s dohledem zkušených profesionálů [13].

V dnešní době se právě díky osobním sensorům přesouvá část vyšetření zpět k lidem domů či kdekoliv se nacházejí [14][15], což je s ohledem na stárnoucí populaci důležité zefektivnění péče o zdraví. V produktivním věku se senzory používají hlavně k monitorování výkonnosti, což také vede k motivaci ke zdravému životnímu stylu a tím k jistému omezení negativních vlivů na zdraví člověka.

V pokročilém věku nastupují asistivní technologie, které pomáhají udržet fungování a samostatnost uživatele bez zásahu další osoby. Dále přichází na řadu asistenční technologie, kdy stále dochází k monitorování jedince v jeho domácím prostředí, ale data mohou být analyzována či odesílána k analýze na jejímž základě je automaticky vyžádána asistence odborníka.

Všechna tato využití přispívají k větší samostatnosti a nezávislosti jedince na pomoci jiných osob a mají pozitivní dopad na kvalitu jeho života.

1.2 Rozmanitost

Jak jsem zmínil v úvodu, zdraví je velmi široká problematika a i samotná fyzická odezva těla poskytuje množství veličin k měření. Přirozeně tak existuje celá řada senzorů, které třeba i pro měření stejné veličiny používají různé přístupy. Některé senzory měří přímo zdravotní ukazatele, jiné se soustředí na výkonnostní ukazatele při pohybu uživatele.

Elektrokardiograf pořizuje časový záznam elektrické aktivity generované činností srdce. Používá se k monitorování srdeční aktivity, přičemž z dat se dokáže určit její poruchy (různé typy arytmie) či hrozící nebo existující poškození srdce (infarkt myokardu) [16][22].

Oxymetr měří nasycenost krve kyslíkem. Nasycenost se neinvazivně měří tak, že se skrz tělo vyšle k fotodiodě světlo o různých vlnových délkách a zjišťuje se jejich úbytek při průchodu tělem a tedy i krví. Úbytek se u různých vlnových délek významně liší v závislosti na nasycení krve kyslíkem [17][23].

Senzory srdečního tepu detekují úder srdce a dávají k dispozici informace o těchto událostech jako čas úderu, interval mezi úderem a zpravidla také vypočítanou hodnotu úderů za minutu. Jeden typ tohoto senzoru měří úderem zkoumáním elektrických potenciálů a bývá upevněn na pružném pásu, který je uchycen okolo hrudníku [24]. Pro svou funkčnost předpokládají vodivý kontakt elektrod integrovaných do pásu s kůží měřeného subjektu. U druhého typu senzoru se pro měření používá prosvícení žíly a fotosenzor [25]. Data z fotosenzoru se analyzují a hledá se změna intenzity světla způsobená pohybem krve [18]. Údaj o srdečním tepu mohou také jednorázově

poskytnout senzory typu pulzní oxymetr, tlakoměr, ECG/EKG či smartphone s diodou a vhodnou aplikací.

Váhou se dnes dá měřit nejen hmotnost těla. Mezi další hodnoty, které moderní váhy dokážou poskytnout patří údaj BMI, procento tělesného tuku, procento hydratace těla, hmotnost svalů, hmotnost kostí nebo počet kalorií pro udržení váhy s využitím metody BIA [19].

Tlakoměr může být klasického charakteru s přípojnou nafukovací manžetou pro horní část paže, nebo kompaktní pro měření na předloktí, prstu či ušním lalůčku.

Domácí či přenosné malé glukometry slouží k orientačnímu měření obsahu glukózy v krvi. U zdravého člověka je koncentrace glukózy na lačno menší než 4-5,7 mmol/l a dvě hodiny po jídle menší než 7,8 mmol/l [20]. U diabetika jsou tyto hodnoty lehce vyšší. Dolní hranicí je 3,3 mmol/l, při nižších hodnotách se jedná o hypoglykemii.

Krokoměr počítá počet ušlých kroků a případně ušlou vzdálenost a počet spálených kalorií. Z původně čistě mechanických čidel jsou dnes elektronická zařízení, která údaje většinou získávají analýzou signálů z jednoho až tří akcelerometrů a detekcí houpavého pohybu kyčlí.

Z geolokačních systémů jako je americký GPS, ruský GLONASS nebo v budoucnu evropský Galileo lze získávat souřadnice pro sestavení trasy pohybu, zjišťování rychlosti či překonaného výškového profilu trasy.

Objevují se také snahy o další zlevnění a zjednodušení obsluhy komplexního monitoringu integrováním několika senzorů do jednoho zařízení. Například společnost Sensaris se snaží se svým kombinovaným senzorem ZaoPod projít certifikací pro zdravotnická zařízení [21]. ZaoPod dokáže měřit nasycenost krve kyslíkem, krevní tlak, tělesnou teplotu, srdeční tep a obsah glukózy v krvi.

Dnes s různými platformami pro rychlé prototypování a mikrokontrolery, v čele s oblíbeným Arduinem a jeho mutacemi, s dostupným příslušenstvím je také možnost si pro výzkum sestavit vlastní senzory. Důvodem může být nižší výsledná cena než při koupi existujícího řešení nebo to, že požadovaný senzor neexistuje či není běžně dostupný.

2 Transport dat

Senzory používají rozdílné způsoby manipulace s naměřenými daty. Často mají vlastní display, na kterém naměřené hodnoty dokážou zobrazit. Také může být přítomna vnitřní paměť pro uložení omezeného počtu měření. Některé senzory neumožňují transport dat.

Nejdůležitější pro tuto práci jsou však senzory, které naměřená data umožňují přečíst externím zařízením nejlépe standardizovaným bezdrátovým způsobem přenosu. Takové senzory buď kontinuálně vysílají naměřené hodnoty a, pokud nemají paměť, lze je tedy získat jen v okamžiku měření. Takto se chovají např. monitory srdečního tepu. Pokud má zařízení i paměť, zpravidla nabízí možnost získat i celou uloženou historii měření.

2.1 Topologie

Různé senzory počítají s rozdílnými topologiemi pro distribuci naměřených dat. Dále uvádím používané způsoby rozložení uzlů, kdy každá další topologie je rozšířením předchozí.

2.1.1 Jednovrstvá

Senzor funguje jako samostatná jednotka. Jeho součástí je display, na němž zobrazuje naměřené hodnoty, dovoluje procházet historii měření (pokud má paměť), případně umožňuje nastavení, kalibraci a správu uložených měření.

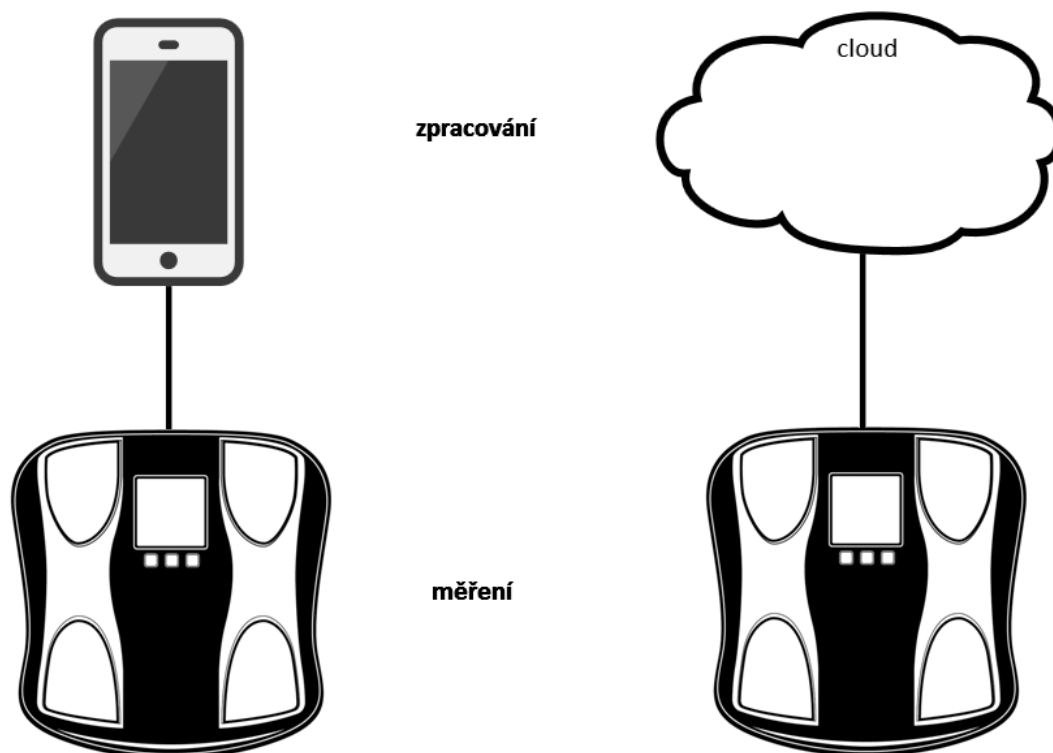
Takto dokáže fungovat většina současných senzorů jako tlakoměry, glukometry, krokoměry, váhy, EKG nebo některé oxymetry.

2.1.2 Dvoustvrá

Senzor je použit k provádění měření a data buď aktivně vysílá nebo je ukládá a poskytuje při vyžádání. Měření se pak zpracuje a zobrazí až v dalším uzlu. Tímto uzlem může být klasický počítač/notebook (zde se využívá i komunikace přes USB), různé chytré/sportovní hodinky, při běžném monitoringu nejčastější smartphone/tablet nebo přímo server v internetu (viz Obr. 1).

Některé senzory, jako např. hrudní pás měřící srdeční tep, nemají vlastní display a tak nemohou být využity samostatně [24].

Přímé odesílání dat na server v internetu je spíše výjimka. Toto umí např. senzory firmy BlipCare, které se dokáží připojit k domácí Wi-Fi síti a automaticky měření odeslat [26]. Daleko častější je následující topologie.



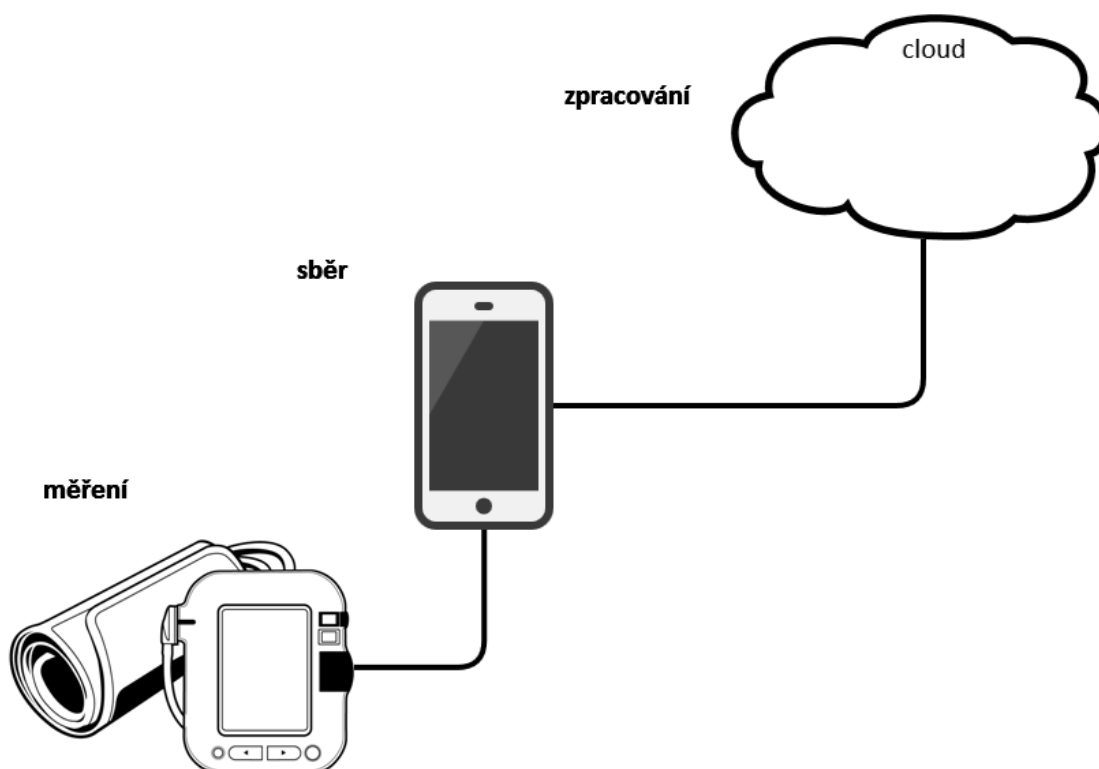
Obr. 1: Dvoustvrá topologie.

2.1.3 Třívrstvá

Toto rozšíření předchozí topologie přidává k lokálnímu zpracování ještě cloudovou vrstvu (viz Obr. 2). Data se ze sběrného zařízení odešlou na server, kde mohou probíhat hlubší analýzy dat nebo v závislosti na výsledcích tento server poskytuje asistenci.

V dnešním světě cloudových služeb je toto velmi časté rozložení a používá ho většina firem zabývajících se eHealth (např. [14]). Je to i případ aplikace vyvíjené v této diplomové práci.

Mezi senzorem a serverem může být více uzlů. Příkladem může být následující způsob použití. Hrudní pás při sportu měří srdeční tep, měření jsou odesílána do sportovních hodinek, které jsou energeticky úspornější než smartphone a dovolují také snadnější zobrazení údajů sportovci. Po ukončení sportovní aktivity jsou data uložena v hodinkách načtena chytrým telefonem, který umožní jejich základní analýzu a ukáže statistiky. Mobilním připojením nebo po připojení k internetu přes Wi-Fi jsou údaje odeslány na server k pokročilé analýze či uskladnění.



Obr. 2: Třívrstvá topologie.

2.2 Komunikační rozhraní

Mezi bezdrátová rozhraní běžně používaná u osobních senzorů patří Bluetooth Smart [11], relativně nový ANT [12] a výjimečně i Wi-Fi.

2.2.1 Bluetooth

Bluetooth je oblíbeným rozhraním pro komunikaci s většinou malými přenosnými zařízeními nebo příslušenstvím jako jsou sluchátka nebo klávesnice a myši. Operuje v bezlicenčním pásmu ISM (industrial, scientific, medical) 2.4 GHz. Již delší dobu je standardní výbavou mobilních telefonů a notebooků. Stolním počítačům bez integrovaného Bluetooth lze toto rozhraní přidat pomocí USB adaptéru.

Tato technologie má za sebou dlouhý vývoj a aktuálně nejnovější verze je 4.2. V kontextu zdravotních senzorů je oblíbená verze 4, alternativně známá jako Bluetooth Low Energy (BLE) či Bluetooth Smart. Tato verze byla vyvinuta právě pro zařízení typu zdravotní senzory a fitness monitory. Soustředí se na minimalizaci energetických nároků, protože senzory mohou být poháněny např. pouze jednou „knoflíkovou“ baterií.

2.2.1.1 Bluetooth profily

Každé zařízení, které komunikuje pomocí Bluetooth musí implementovat základní protokoly LMP, L2CAP a SDP. Service Discovery Protocol slouží ke zjišťování schopností zařízení. Běžně je také k dispozici protokol RFCOMM, což je v podstatě bezdrátová emulace sériového portu.

Nad základními protokoly jsou pak postaveny takzvané profily [27]. Ty jasně specifikují konkrétní určení a způsob komunikace s daným zařízením. Pro tematiku této práce jsou zajímavé hlavně profily jako Health Device Profile a ostatní pro zdravotní a fitness senzory (BLP, CSCP, GLP, HRP, WSP atd.). Běžně se však také pro komunikaci používá základní RFCOMM protokol s proprietárním aplikačním protokolem, což ale oproti jasně daným profilům vylučuje vzájemnou kompatibilitu.

2.2.2 ANT

ANT [7] je další bezdrátové rozhraní operující v 2.4 GHz ISM pásmu. Bylo navrženo pro malé zdravotní a sportovní senzory, což znamená minimální

požadavky na energetické a výpočetní zdroje. Vyznačuje se vysokou efektivitu komunikace i velkého počtu zařízení v jednom místě (např. start maratonu nebo cyklistického závodu).

Potřebná elektronika se začíná integrovat do stále více chytrých telefonů a sportovních hodinek [9]. Alternativně jsou k sehnání externí antény buď speciálně pro Apple produkty nebo univerzální do microUSB a USB portů.

2.2.2.1 ANT+

Podobně jako u Bluetooth existují specifické profily, které se souhrnně označují jako ANT+ [8]. Je to tedy označení množiny specifikací pro povahu dat, která se přenáší sítí ANT. Každá specifikace je svázána s určitým typem zařízení a jeho způsobem použití. Např. pro zařízení typu tělesná váha je definován profil Weight Scale Device Profile, který určuje komunikační protokol mezi váhou a zájemcem o měřené informace. Stejně jako u Bluetooth profilů je velkou výhodou ANT+ vzájemná kompatibilita zařízení a aplikací různých výrobců.

2.2.3 Wi-Fi

Tento druh sítě je dnes obecně velmi oblíbený a je standardní výbavou notebooků a chytrých zařízení jako jsou smartphony či tablety. Jedná se o sadu standardů IEEE 802.11 [28], které řeší komunikaci jen do druhé vrstvy referenčního modelu ISO/OSI. Tato síť byla navržena pro rychlý přenos dat a je více energeticky náročná než předchozí zmiňované. To je hlavní důvod, proč se u zdravotních senzorů využívá jen výjimečně.

Příkladem je firma BlipCare vyrábějící tlakoměr a tělesnou váhu s podporou Wi-Fi verze b/g/n [26]. Tato zařízení se dokážou připojit k domácí síti a autonomně bez potřeby smartphonu či počítače odeslat naměřená data do Internetu na server.

Zajímavostí je, že Bluetooth verze 3+HS při posílání velkých dat sám pouze naváže spojení a pro samotný přenos využije právě Wi-Fi [29], což může být ve výsledku energeticky efektivnější, protože se vysílá sice s vyšší spotřebou, ale po kratší dobu.

3 Existující aplikace

Většina aplikací se soustředí na zaznamenávání údajů při vykonávání sportovní činnosti. Mezi nejoblíbenější patří běh a chůze, ale často jedna aplikace má na výběr z desítek sportů. Jsou koncipované tak, aby si při měření vystačily se senzory integrovanými přímo v telefonu. Základním zdrojem informací je GPS, odkud se zjišťují souřadnice, rychlost a výškový profil. Pohybové senzory se dají využít pro počítání kroků.

Častá je podpora pro senzory srdečního tepu nebo chytrých či sportovních hodinek. Hodinky se však používají většinou jen pro zobrazení statistik. Výjimečně lze z hodinek importovat jimi naměřené hodnoty, pokud tuto funkcionalitu mají, nebo např. u Google Wear lze využít základních hlasových povelů [30].

Pokud aplikaci vyvíjí větší firma, bývá k ní ještě webová služba, do které se synchronizují nasbíraná data. Webový profil pak zpravidla nabízí více funkcí a statistik než mobilní aplikace.

Dalším typem monitorovacích aplikací jsou deníčky, které počítají statistiky z ručně zadávaných hodnot, či specializované aplikace pro jeden konkrétní senzor.

3.1 Endomondo

Oblíbená univerzální aplikace pro monitorování sportovní aktivity [31]. Poskytuje výběr z více než padesáti sportů a aktivit, což se spolu s výběrem zobrazovaných dat provádí v měřicí aktivitě (Obr. 3). Mimo klasického běhu, cyklistiky nebo plavání to mohou být i takové aktivity, jakými jsou různé kolektivní sporty (fotbal, volejbal apod.), tanec nebo šerm. Statistiky po ukončení tréninku jsou vidět na Obr. 4.

Hlavním zdrojem dat je GPS. Pokud necháme GPS vypnuté, sleduje se jen čas aktivity, jinak se ukládá ještě trasa a rychlost. Volitelně je možné připojit kompatibilní

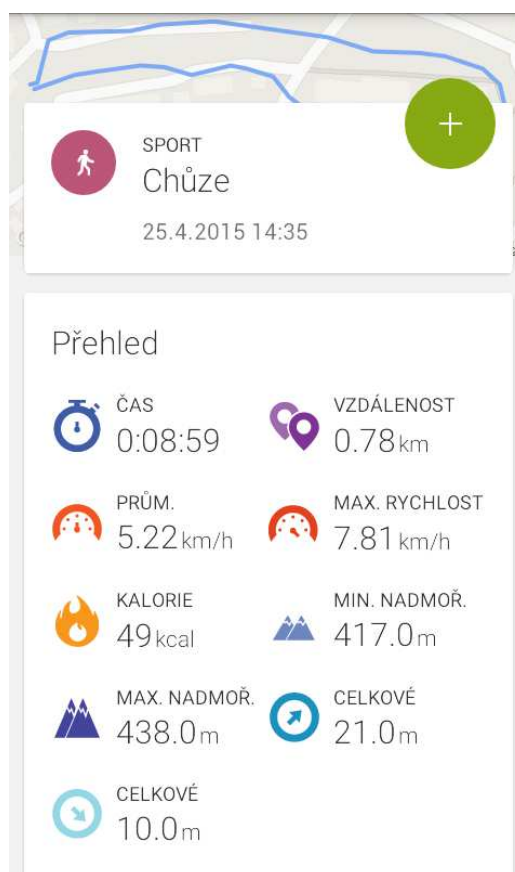
externí senzory. Je to buď monitor tepové frekvence, který může být připojen jak přes Bluetooth, tak i přes ANT+. Jediným dalším typem senzoru je snímač kadence šlapání na kole komunikující pomocí ANT+. Ke každé aktivitě se vypočítá přibližné množství spálených kalorií, kdy výpočet je závislý na typu aktivity.

Aplikace umožňuje informace o tréninku zadat manuálně, přitom se možnosti mezi různými aktivitami nemění. Ať se jedná o běh, golf, tanec nebo posilování, vždy lze zadat jen začátek a dobu aktivity a uraženou vzdálenost. Opět se pro trénink počítají spálené kalorie a podle toho, jestli se jedná o aktivitu s přesunem, se do výpočtu započítá i dosažená vzdálenost.

Dále je možné k aplikaci připojit jedny z podporovaných náramkových hodinek, které pak slouží k zobrazení průběžných statistik. Nebo je zde možnost si tyto statistiky v určitém intervalu či po stisknutí tlačítka ve sluchátkách syntetizovanou řečí poslechnout nebo se jen nechat povzbudit.



Obr. 3: Endomondo - měření.



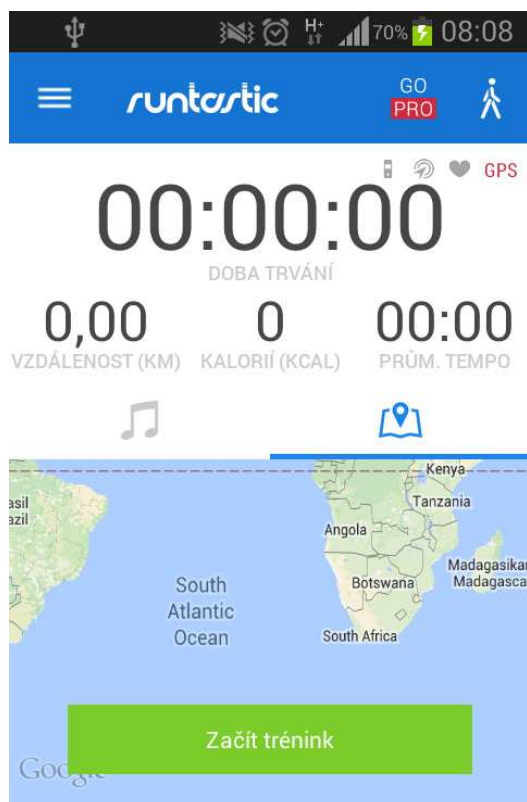
Obr. 4: Endomondo - statistiky.

Přítomné jsou také sociální prvky. Lze se přátelit s ostatními lidmi, sledovat jejich aktivitu nebo s nimi soutěžit a samozřejmě také sdílet své výsledky na sociálních sítích. Mimo mobilní aplikaci je ke službě i webové rozhraní, kde lze hlavně historii měření a statistiky pohodlněji prohlížet.

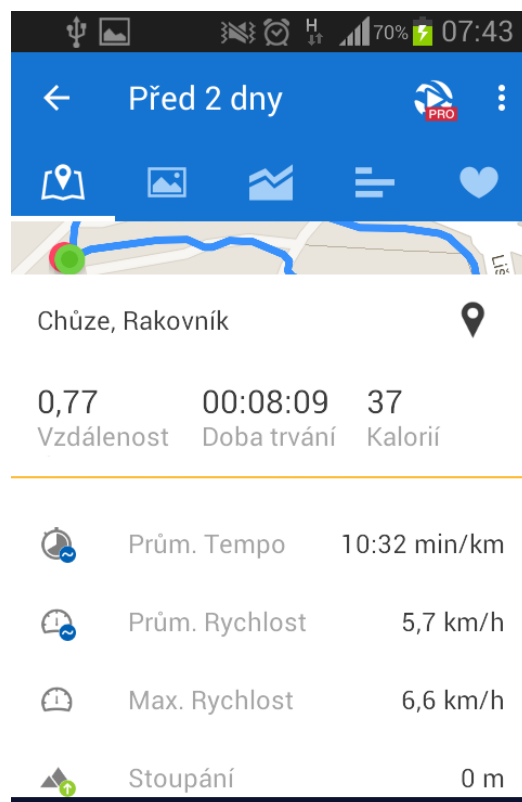
3.2 Runtastic

Toto je další univerzální aplikace pro sledování sportovní a fitness aktivity, velmi podobná té předchozí [32]. Nabízí výběr z více než padesáti aktivit (seznam není identický, např. je zde více vodních sportů) s rozdílným výpočtem spálených kalorií. Měřicí aktivita (Obr. 5) a poskytované statistiky (Obr. 6) nabízí obdobné možnosti.

Hlavním senzorem je opět GPS, zaznamenávají se stejné údaje. Snímač tepové frekvence lze připojit buď přes Bluetooth, nebo proprietární hrudní pás s přijímačem zapojitelným do konektoru Jack. I zde je možnost spárovat některé z chytrých hodinek pro zobrazení průběžných statistik. Pro tento účel lze použít také chytrý náramek Orbit opět vlastní výroby, který navíc slouží jako krokoměr a monitor spánku (délka a cykly).



Obr. 5: Runtastic - měření.



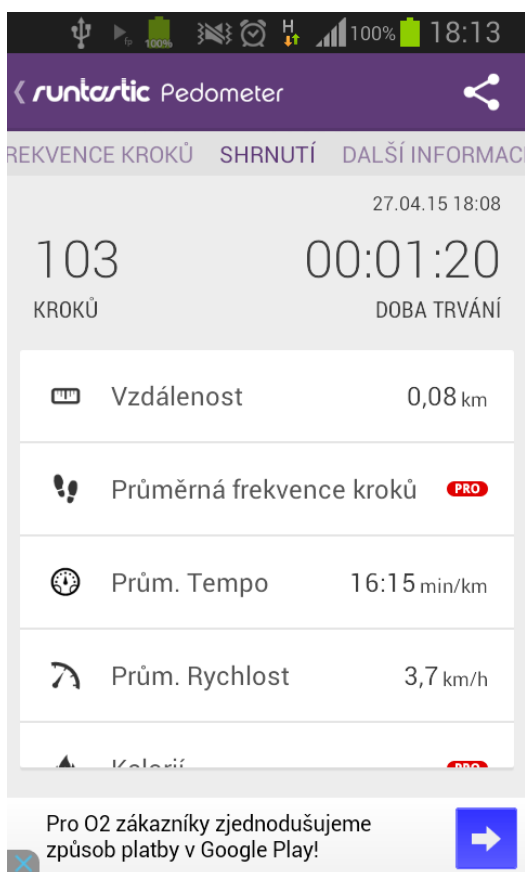
Obr. 6: Runtastic - statistiky.

Kromě této aplikace nabízí společnost Runtastic celou řadu menších specializovaných aplikací. Tyto aplikace se soustředí vždy jen na jednu činnost, k jejíž měření používají jeden integrovaný nebo externí senzor. Pár vybraných stručně popíši níže.

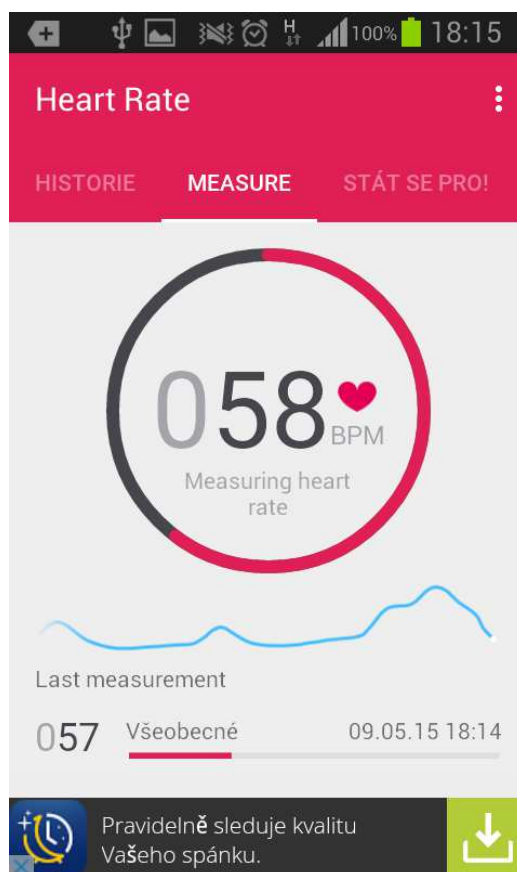
Nad všemi těmito aplikacemi je dostupná webová služba, která všechny naměřené hodnoty umožní agregovat a podávat souhrnné statistiky. Samozřejmostí jsou sociální možnosti obdobné předchozí službě.

3.2.1 Pedometer

Tato aplikace [33] počítá kroky pomocí analýzy dat z integrovaných pohybových senzorů a po zadání délky kroku a hmotnosti také rychlost, ušlou vzdálenost a počet spálených kalorií. Není zde možnost trasovat cestu pomocí GPS ani připojit externí senzor. Podoba statistik je vidět na Obr. 7.



Obr. 7: Runtastic Pedometer - statistiky.



Obr. 8: Runtastic Heart Rate - měření.

3.2.2 Push-Ups

Tentokrát aplikace [34] zaměřená na trénink kliků. Po nastavení uživatelského stávajícího stavu každý den nabídne ztěžující se trénink. Ten sestává z několika sérií po různém počtu opakování. Pokyny se ukazují na displeji a také jsou diktovány pomocí syntezátoru hlasu.

Měření probíhá tak, že se telefon položí na zem a nad ním provádíme kliky. Aplikace pak s využitím proximity senzoru počítá jednotlivá opakování. Alternativně se pro počítání lze při cvičení dotýkat obrazovky špičkou nosu.

3.2.3 Heart rate

Poslední vybraná aplikace [35] od společnosti Runtastic je zajímavá tím, že pro měření srdeční tepové frekvence používá fotoaparát smartphonu a techniku prosvícení popsanou v kapitole 1.2. Průběh měření je na Obr. 8, kde modrý graf zobrazuje detekované změny svitu v závislosti na proudění krve.

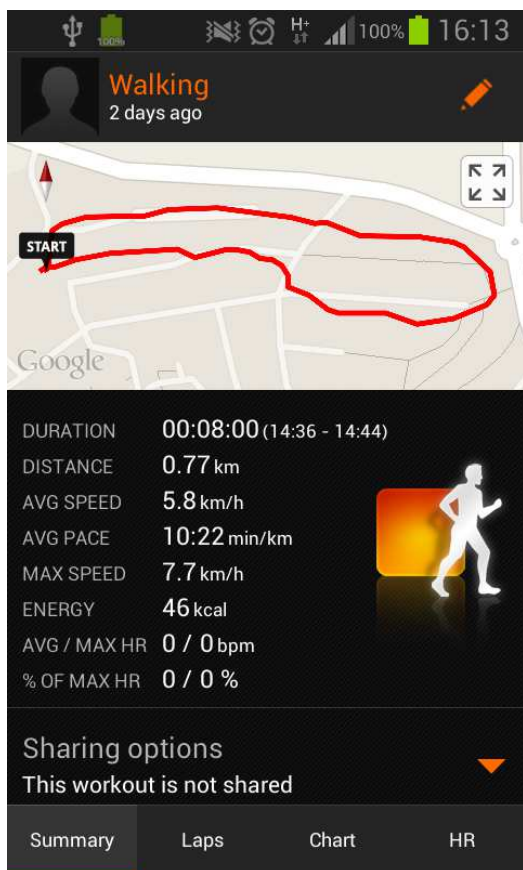
3.3 Sports Tracker

Poslední univerzální aplikace a webová služba se liší od předchozích dvou hlavně designem [36]. Na výběr poskytuje okolo dvaceti sportů. Pro uživatele zajímavé je, že zobrazuje i průběžnou statistiku za poslední kilometr. Ta celková je vidět na Obr. 9.

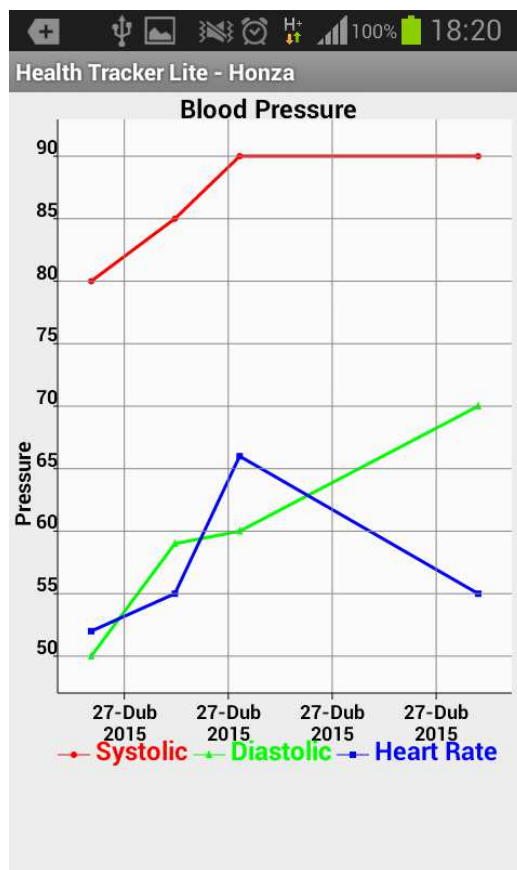
Z integrovaných senzorů dokáže využít GPS a barometr pro alternativní určení nadmořské výšky. Z externích dokáže využít Bluetooth hrudní pás pro měření srdečního tepu nebo ANT+ měřič kadence šlapání na kole.

3.4 Health Tracker

Toto je jednoduchá, graficky ne příliš zdařilá, aplikace typu lékařský deníček [37]. Sama neumí pracovat s žádnými senzory a slouží pouze pro manuální zadávání naměřených hodnot krevního tlaku, glukózy v krvi a tělesné váhy, které statisticky zpracuje a jejich vývoj ukáže v grafech (viz Obr. 10).



Obr. 9: Sports Tracker - statistika.



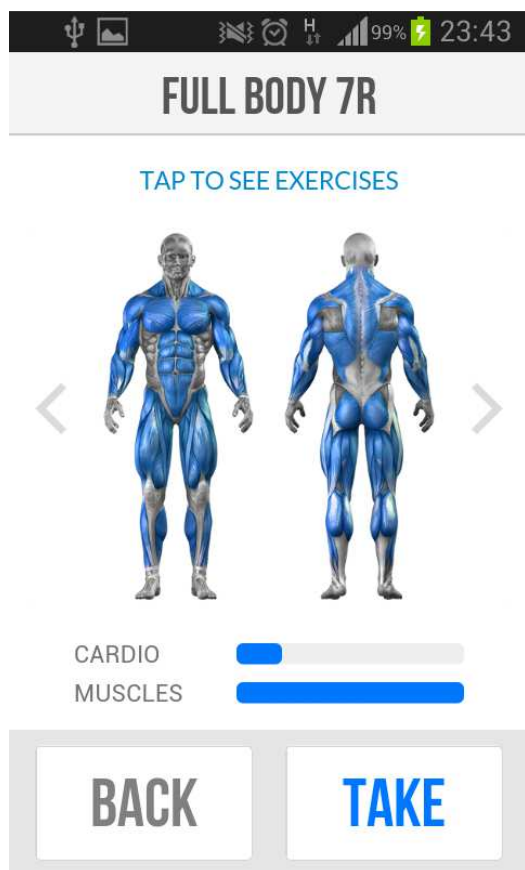
Obr. 10: Health Tracker - statistika.

3.5 Madbarz

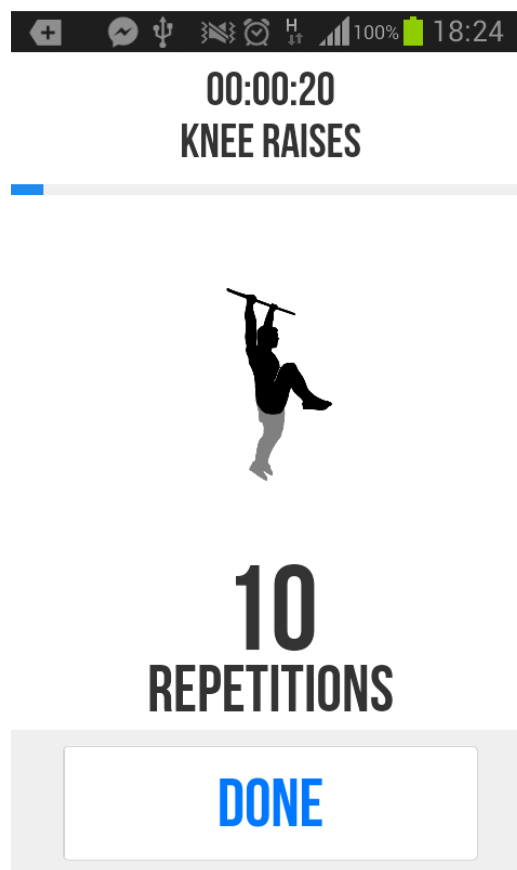
Madbarz je aplikace [38] pro posilování s vlastní vahou těla - kalistenika [39]. Stejně jako předchozí, ani tato aplikace nepoužívá data z žádných senzorů a dokonce je ani neumožňuje přímo zadávat. Výjimkou je pouze údaj o tělesné váze.

V nabídce je na výběr spousta cvičebních rutin. Každá se skládá z několika cviků a je zaměřena na určitou skupinu svalů. Zpravidla se cvičí více sérií dané rutiny. Tato množina je vypsána slovně i znázorněna graficky (viz Obr. 11). Po spuštění rutiny aplikace postupně vyzývá k jednotlivým cvikům. Vždy nejprve zobrazí název cviku s jeho přehledným grafickým znázorněním a mezitím se odpočítává čas do startu (viz Obr. 12). Po odcvičení požadovaného počtu opakování je nutné tuto skutečnost potvrdit. Poté aplikace zařadí krátký odpočinek, než se přejde k dalšímu cviku. Další pauza nastává mezi sériemi a bývá řádově delší než mezi jednotlivými cviky.

Podle odvedených cviků se počítají body. K dispozici jsou také různé statistiky ať už celkové, nebo postup zlepšování znázorněný v grafu. Všechny údaje lze také nahrát do spárovaného účtu na server. Webová služba obsahuje sociální prvky po vzoru Twitteru. Ostatní uživatele lze následovat nebo oni mohou následovat nás. Díky tomu můžeme sledovat jejich aktivitu a případně jim přidělit respekt za jejich výkony.



Obr. 11: Madbarz - výběr rutiny.



Obr. 12: Madbarz - průběh cvičení.

3.6 eVito

Tato komplexní platforma [40] je zdánlivě podobná univerzálním sportovním systémům Endomondo, Runtastic nebo Sports Tracker. Kromě fitness je více zaměřen na monitorování zdraví a sportování bere spíše jako prostředek pro udržení zdraví než jako svůj primární cíl.

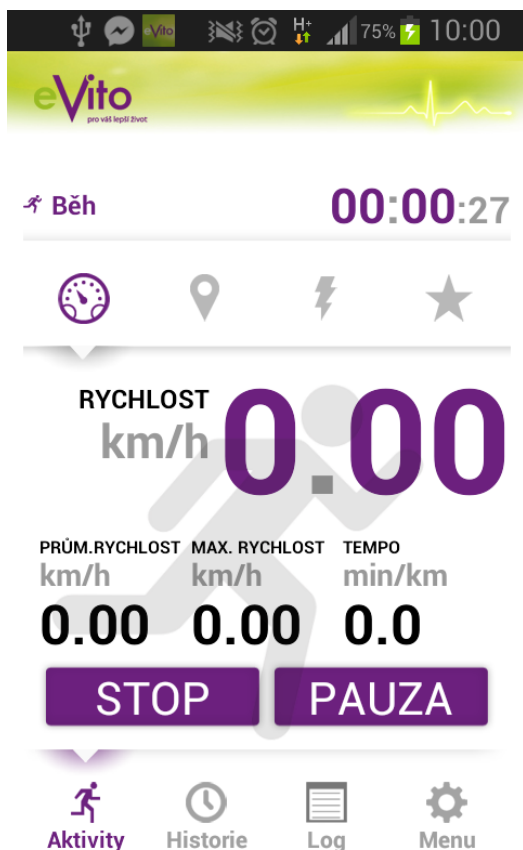
Umožňuje monitorovat sportovní aktivity, ale dovoluje také měřit zdravotní ukazatele. V nabídce má tři verze různě placených účtů (placené účty mají i předchozí systémy, ale tam nejsou výhody tak zajímavé) [41]. Základní verze umí pouze ukládat

měření a zobrazovat statistiku. Verze Active přidává správu jídelníčku, sociální prvky se soutěžemi nebo lepší správu výkonnostních sportů.

Nejdražší verze Medical přináší také asistenční prvky. Jedná se o konzultace s doktory a právníky, možnost poskytovat data vlastnímu lékaři, detailní rozbor naměřených hodnot. Velmi zajímavou funkcí je automatické upozornění rodiny či zvoleného doktora v případě zhoršení zdravotního stavu.

Samozřejmě existuje mobilní aplikace pro chytrá zařízení. Umožňuje měření šesti základních aktivit (viz Obr. 13), jimiž jsou běh, in-line bruslení, turistika, cyklistika, lyže a běžky. Po aktivitě zobrazí základní statistiky (viz Obr. 14). Dále lze číst data z podporovaných senzorů. Těmi jsou ANT+ srdeční pás, tělesná váha či tlakoměr, vlastní ANT krokoměr nebo Bluetooth tlakoměr či glukometry od výrobce FORA.

S těmito senzory lze komunikovat také počítačem. K tomu nejsou poskytovány plnohodnotné aplikace, ale jen systémové služby pro Windows či Mac. Naměřené hodnoty se odesílají na server ke zpracování (viz A. 2).



Obr. 13: eVito - měření.



Obr. 14: eVito - statistika.

4 EEG/ERP Portál

Na Katedře informatiky a výpočetní techniky Západočeské univerzity v Plzni je aktivní výzkumná skupina, která se zabývá výzkumem neuroinformatiky. V rámci této činnosti se zde provádí experimenty zkoumající reakce lidského mozku při různých činnostech nebo v různých situacích.

Tato měření generují velké množství dat, která je potřeba efektivně uchovávat a spravovat, aby je bylo možné později snadno zpracovávat a analyzovat. Právě proto se zde vyvíjí také EEG/ERP Portál, což je webová aplikace umožňující svým uživatelům po registraci vytvářet vlastní experimenty, kam mohou ukládat naměřená data. Přesné důvody pro vznik portálu jsou popsány v [3].

Kromě binárních dat obsahujících časový průběh elektrických signálů z jednotlivých elektrod je důležité uchovávat i metadata. Ta popisují okolnosti experimentu - např. při jakém scénáři měření probíhalo, testovaný subjekt (např. pohlaví, věk) jaké přístroje (hardware a software) byly pro měření použity, stav okolí (např. počasí, čas) apod. Dále se mohou ukládat dodatečná měření dalších odezev těla testovaného subjektu, jež bude poskytovat také aplikace vyvíjená v této práci.

4.1 Architektura

Portál je implementován jako webová aplikace, která používá enterprise technologie programovacího jazyka Java.

4.1.1 Datová vrstva

Pro přístup k datům se používá JPA a jako implementace oblíbený framework Hibernate nad SŘBD PostgreSQL. V loňském roce proběhla diplomová práce, která do projektu implementovat další, tentokrát nerelační, SŘBD pro efektivní možnosti správy a vyhledávání části informací. Jako vhodné řešení byla vybrána

a použita databáze Elasticsearch. Do něj se aktuálně ukládají metadata o provedených experimentech.

Z programového hlediska sem patří POJO třídy a nejnižší DAO vrstva. Detailněji je datová vrstva popsána v již zmíněné diplomové práci [7].

4.1.2 Aplikační vrstva

DAO vrstva se stará vždy jen o jeden typ dat. Nad ní operuje další tzv. Service vrstva. Ta může využívat i více DAO objektů a dokáže již provádět složitější manipulaci s daty a implementovat aplikační logiku. Další úrovní abstrakce je tzv. Facade vrstva. Princip je podobný jako u předchozí. Je určena k provádění rozsáhlých vysokoúrovňových případů užití (use case), k čemuž může využít více Service objektů.

V této vrstvě se používá framework Spring. Jmenovitě se jedná o IoC kontejner pro automatické řešení závislostí, modul Security pro řízení přístupu k datům a také modul Social použitý pro podporu přihlašování přes sociální sítě Facebook a LinkedIn.

4.1.3 Prezentáční vrstva

Vrstva je aktuálně implementována dvěma technologiemi. Pro implementaci se původně využívalo modulu Spring MVC. Od roku 2013 se však vypsanou bakalářskou prací [4] začalo přecházet na modernější komponentový framework Wicket [42]. Aktuálně se ještě stále používají oba frameworky vedle sebe a každý se stará o část uživatelského rozhraní.

4.1.4 Webové služby

Kromě webového rozhraní pro uživatele se dá s portálem komunikovat i přes webové služby. Je zde implementováno tzv. RESTful rozhraní (Representational State Transfer) [43] pro manipulaci s experimenty, uživateli a skupinami, datovými soubory s měřením, rezervaci laboratoře apod.

U tohoto typu rozhraní probíhá komunikace pomocí HTTP protokolu. URL označují zdroj nebo cíl, se kterým chceme manipulovat. Vyměňují se přitom zprávy nejčastěji ve formátech JSON nebo XML.

Mobilní aplikace vyvíjená v této práci využívá právě tohoto rozhraní. Ověřuje přes něj přístupové údaje uživatele a získává seznam experimentů, ke kterým má přihlášený uživatel přístup. K vybranému experimentu pak umožní nahrát hodnoty naměřené použitými senzory.

5 Zpracovávané senzory

K dispozici byla téměř kompletní sada přístrojů nabízených k výše popsanému monitorovacímu systému eVito (viz kapitola 3.6). Do této sady patří krokoměr a váha vlastní výroby, glukometr a tlakoměr značky FORA a sportovní hodinky a snímač tepové frekvence značky Garmin.

5.1 Krokoměr

První na řadu přišel 3D krokoměr SL přímo od společnosti eVito [44]. V době psaní práce měl hodnotu 1500,- Kč, ale probíhala zajímavá akce, kdy po odchození průměrně 10000 kroků denně po dobu jednoho roku vás firma odmění vrácením plné pořizovací ceny. Ačkoli jsem se hodně snažil, nakonec se mi z něj nepodařilo data přečíst.

5.1.1 Uživatelský pohled

Tento krokoměr je velký asi jako mp3 přehrávač. Je dobře dílensky zpracovaný. V černé barvě a lesklém provedení vypadá elegantně, ale tradičně se kvůli této povrchové úpravě rychle pokryje otisky prstů.

Pro ovládání jsou k dispozici dvě tlačítka. Hlavní tlačítko je mechanické a větší a přepínají se jím jednotlivé obrazovky či potvrzují hodnoty. Druhé tlačítko je malé senzorické a slouží ke změně hodnot v nastavení. Na zadní straně je ještě zapuštěné tlačítko reset pro zotavení z případných neznámých stavů (během zkoušení se neprojeví).

Přístroj je napájen pouze z jedné „knoflíkové“ baterie, na kterou dokáže pracovat půl roku až rok. Po výměně se musí vždy znovu nastavit. Do nastavení se dostaneme podržením hlavního tlačítka po dobu třech sekund. Nastavit lze tělesnou váhu v kilech

nebo librách, délku kroku, datum a čas, případně zvuky. Zajímavým je dále nastavení, jak často se má krokoměr pokoušet o synchronizaci dat.

Lze nastavit synchronizaci po minutě, nebo deseti minutách, případně podržením hlavního tlačítka po dobu šesti sekund vyžádat odeslání dat manuálně. Tzv. heartbeat funkce, pokud je zapnuta, hledá každé tři minuty přijímací zařízení, kterému odesílá stav baterie a umožňuje stáhnout si z něj kompletní nastavení, tudíž se nemusí pokaždé (např. po výměně baterie) zadávat ručně.

5.1.2 Měření

K aktivaci zařízení postačí jeho jakýkoli pohyb. Po něm zobrazí na displayi „Hello“ a začne počítat kroky. Krokoměr má vlastní display, na kterém přehledně zobrazuje naměřené hodnoty. Pokud jsou v nastavení přístroje správně uvedeny údaje pro hmotnost nositele a délku jeho kroku, vypočítává se také ušlá vzdálenost a počet spálených kalorií. Na ně lze přepnout hlavním tlačítkem. Počty kroků se zaznamenávají po minutách, takže máme po uploadu ve webovém rozhraní přehled o aktivitě v průběhu dne. Největší rozlišení, které můžeme prohlížet jsou hodiny. V paměti přístroje se data uchovávají po dobu 35 dnů.

Krokoměr eVito používá tři MEMS pro měření pohybu ve třech osách. Tím je zajištěno, že funguje nezávisle na své poloze či natočení. Zpracovává signály ze snímačů a hledá jejich průběh typický pro chůzi. Tím se vyhýbá počítání falešných kroků např. při přesunu autem. O tom jsem se přesvědčil, když jsem se snažil krokoměr ošálit pro získání testovacích dat „ze židle“. Nestalil pouze jednoduchý pohyb nahoru a dolů, ale musela se přidat také lehká rotace.

Pro vyzkoušení v reálu jsem si vzal s sebou krokoměr na světelný Signal festival do Prahy. Tam jsem podle přístroje za 2 hodiny a 40 minut čistého času udělal 11700 kroků, ušel vzdálenost 9,5 kilometru a spálil 680 tisíc kalorií (viz příloha A. 1).

5.1.3 Získání dat

Krokoměr společnosti eVito používá k bezdrátové komunikaci technologii ANT. Přestože je pro krokoměry standardizována komunikace v podobě ANT+ profilu,

u tohoto zařízení není implementována. Místo toho je použit jen čistý ANT a proprietární komunikační protokol.

To představovalo velkou překážku, protože není veřejně dokumentováno nastavení připojení k tomuto krokoměru a navíc i pokud se podaří připojit, zbývá ještě vyzkoumat komunikační protokol. Pro komunikaci je zapotřebí vytvořit jednoduchou síť pro dva uzly spojené ANT kanálem. K tomu je nutné správně nastavit:

- Typ kanálu - simplexní/duplexní, soukromý/sdílený, master/slave
- RF frekvence - offset 0-125 MHz od základní frekvence 2.4 GHz
- Channel ID - 4 byty
 - Transmission type - 8 bitů příznaků
 - Device type - třída zařízení (např. u ANT+ měřič tepu, tlakoměr..)
 - Device number - 16 bitové ID zařízení unikátní ve své třídě
- Channel period - 16 bitů, jak často master posílá zprávy
- Network - identifikátor ANT sítě včetně veřejné/tajné network key

Nejprve však bylo ještě nutné nastavit testovací prostředí se zajištěním hardwarové komunikace. Na stránkách skupiny ANT aliance je seznam aplikací a přístrojů včetně mobilních telefonů [8], které podporují komunikaci přes ANT. Podle tamních informací Samsung Galaxy S2 (testovací přístroj) nemá integrovanou hardwarovou, ale podporuje technologii USB host a měl by tak být schopen použít USB anténku, která byla ke krokoměru přibalena. Po připojení anténky pomocí USB OTG a nainstalování potřebných aplikací s ovladači (ANT USB Service a ANT Radio Service) byla použita mobilní aplikace ANT Checker, která zjistí možnosti využití technologie ANT. Podle výsledků by měl být telefon schopen komunikovat, ale testovací aplikace ANT+ Plugin Sampler hlásila, že nemůže najít adaptér a ani eVito aplikace se nedokázala s krokoměrem spojit.

Pro ANT jsou připravena SDK pro různé platformy, mj. i pro operační systém Android. Po nainstalování následoval další postup pro zprovoznění anténky v Android emulátoru. Kromě opětovné instalace ovladačů v emulátoru je potřeba použít Windows aplikaci ANT Android Emulator Bridge Tool a k ní v emulátoru mobilní aplikaci ANT Emulator Configuration. První jmenovaná je serverová aplikace, která přes TCP/IP přeposílá data mezi anténkou a emulátorem. Pro spuštění přemostění je nutné vybrat USB konektor, ke kterému je připojena anténka (pokud je jich připojeno více),

a TCP port, na kterém bude naslouchat pro připojení emulátoru. V druhé jmenované nastavíme vybrané připojení a komunikace by měla začít fungovat. To potvrdila opět aplikace ANT+ Plugin Sampler.

S využitím ANT SDK pro Android, byla vytvořena testovací aktivita, ve které šlo testovat komunikaci s krokoměrem. Přes vyzkoušení různých přístupů k vytvoření kanálu se ovšem nepodařilo spojení navázat a proto následovalo zkoumání dalších senzorů.

5.1.4 Poskytovaná data

Přesný formát dat není znám, protože se nepodařilo komunikaci navázat. Podle údajů v aplikaci eVito se však s největší pravděpodobností jedná o:

- počet kroků,
- počet spálených kalorií,
- ušlá vzdálenost.

Počet kroků jednak v každé měřené minutě a pak souhrnně za den včetně ostatních údajů. Dodatečnými informacemi jsou ID a název zařízení.

5.2 Tlakoměr

Dalším zkoumaným senzorem je tlakoměr P30 Plus BT [45], který vyrábí švýcarská společnost FORA Care Suisse AG. Hodnota tohoto zařízení je 2000,- Kč.

5.2.1 Uživatelský pohled

Přístroj je vhodný pro měření hodnot v klinickém nebo domácím prostředí. Je to klasický tlakoměr s nafukovací pažní manžetou. K zobrazování informací má na těle velký display. K ovládání zařízení slouží tři tlačítka - jedno pro zapnutí/vypnutí přístroje (v případě nečinnosti se sám vypne po třech minutách), druhé (S) pro ovládání nastavení a třetí (M) pro úpravu hodnot v nastavení nebo ovládání bezdrátové komunikace Bluetooth.

Před prvním použitím je vhodné přístroj nastavit. Do nastavení se dostaneme podržením tlačítka pro nastavení. Tímto tlačítkem se také přechází mezi možnostmi nastavení, přičemž hodnoty se upravují tlačítkem M. Nejdůležitějším nastavením je datum a čas. Tento údaj se ukládá pro jednotlivá měření, a bez něho nemají měření z dlouhodobého hlediska smysl. Dále lze nastavit už jen měrnou jednotku a zobrazení času na display.

V nastavení je také možnost mazání záznamů uložených v interní paměti. Do této paměti se vejde až 200 záznamů. Poslední možností je zapnutí párování s ostatními Bluetooth zařízeními, která mohou ukládané záznamy získat.

5.2.2 Měření

Platí obecná doporučení být před měřením ve fyzické a psychické pohodě. To znamená alespoň půl hodiny před úkonem nepít, nejíst, nekouřit, necvičit, nekoupat se apod. Také úzkost či stres výsledky zkreslují. Mezi měřeními se doporučuje pauza minimálně pět minut.

Manžeta se nasadí na horní část paže mezi rameno a loket do úrovně srdce, přičemž by se neměla nasazovat obráceně. Směr návleku je popsán na manžetě a pomůckou je její červený okraj, který by měl směřovat dolů. V této poloze také nepřekáží trubička sloužící k nafukování vzduchu.

K dispozici jsou dvě metody měření - automatická oscilometrická metoda a manuální auskulační (Korotkovova) metoda [46]. Po nasazení manžety pro měření tlaku oscilometrickou metodou postačí tlakoměr zapnout a hned se začne pumpovat vzduch. Při této metodě přístroj hledá největší oscilace tepny, čímž najde střední arteriální tlak (MAR). Největší oscilace se vyskytují ve stavu, kdy se tlak v tepně shoduje s tlakem v manžetě. Upouštěním vzduchu se proto přechází od maximálního tlaku, kdy tepnou krev neprochází, po tlak, kdy oscilace bezpečně klesají. Ze získaného MAR se pak vypočítají hodnoty pro systolický a diastolický tlak jednoduchým vynásobením určitými konstantami.

K auskulační metodě je zapotřebí stetoskop, který se přiloží na místo na paži, kde lze nahmatat tep. Dlouhým podržením zapínacího tlačítka se spustí auskulační měření, manžeta zaškrtní tepnu a začne se pozvolna upouštět vzduch. V okamžiku,

kdy se ve stetoskopu ozvou zvuky tepu způsobené obnoveným prouděním krve deformovanou tepnou, zobrazovaná hodnota tlaku odpovídá systolickému tlaku. V okamžiku, kdy tyto zvuky vymizí, aktuální hodnota představuje diastolický tlak.

5.2.3 Získání dat

Jelikož tlakoměr používá k bezdrátové komunikaci technologii Bluetooth, která je standardně podporovaná na mnoha platformách, postup při získávání dat tak byl jednodušší než v případě krokoměru.

Nejprve je nutné tlakoměr spárovat se čtecím zařízením. Pro párování se v nastavení tlakoměru přístroj přepne do módu pro párování, aby ho bylo možné čtecím zařízením detekovat a informace o něm uložit. Dále je potřeba uvést tlakoměr do stavu, kdy naslouchá příchozím požadavkům.

Do tohoto stavu přejde automaticky po každém měření. Existuje ještě druhá cesta, která však překvapivě není uvedena v návodu a zmíněna je jen na spodní straně přístroje. Aby kvůli čtení dat (v mém případě kvůli testování komunikace) nemusel uživatel podstupovat měření, je zde možnost dlouze podržet tlačítko M. Pokud je tlakoměr připraven na přenos dat, kontrolka Bluetooth bliká. Pokud právě probíhá komunikace, tato kontrolka svítí.

Zde je vhodné podotknout, že zatímco probíhala práce na spojení s krokoměrem, probíhala komunikace mezi výzkumnou skupinou a pracovníky společnosti ForaCare. Předmětem komunikace bylo poskytnutí komunikačních protokolů pro čtení dat z tlakoměru a glukometru.

Pro prvotní připojení a výzkum komunikace s tlakoměrem byl použit jednoduchý program v Javě běžící na notebooku s Bluetooth konektivitou. Java API pro komunikaci prostřednictvím Bluetooth je součástí specifikace JSR 82 [47]. Jako implementace byla zvolena knihovna Bluecove [48].

Ačkoli existuje Bluetooth profil pro zdravotní zařízení typu tlakoměr (Blood Pressure Profile - BLP), tento tlakoměr ho neimplementuje. Komunikace probíhá přes RFCOMM kanál ve stylu sériového portu, k čemuž slouží profil SPP (Serial Port Profile).

K získání streamů musíme provést tyto kroky:

- vyhledat dostupná zařízení
- vyhledat příslušnou službu a získat její URL
- otevřít komunikační kanál

Pomocí přístupového bodu k ovládání Bluetooth adaptéru získáme agenta.

```
LocalDevice localDevice = LocalDevice.getLocalDevice();
DiscoveryAgent agent = localDevice.getDiscoveryAgent();
```

Tlakoměr můžeme vyhledat buď z prostředí Windows nebo v kódu spustíme vyhledávání takto:

```
agent.startInquiry(DiscoveryAgent.GIAC, new DiscoveryListener());
```

Nyní vybereme dříve spárovaný či právě nalezený tlakoměr z pole dostupných zařízení:

```
RemoteDevice[] devices = agent.retrieveDevices();
```

Následuje vyhledání služby v zařízení:

```
UUID[] services = new UUID[]{new UUID(0x0003)}; //UUID pro RFCOMM
agent.searchServices(null, services, device, new DiscoveryListener() {
    @Override
    public void servicesDiscovered(int a0, ServiceRecord[] servRecord) {
        connectionURL = servRecord[0].getConnectionURL(0, false);
    }
});
```

Zbývá jen získat streamy pro čtení a zápis:

```
StreamConnection c = (StreamConnection) Connector.open(connectionURL);
OutputStream out = c.openOutputStream();
InputStream in = c.openInputStream();
```

Po navázání spojení a zahájení komunikace ani po odeslání několika pokusných zpráv tlakoměr nereagoval. Nakonec se komunikační protokoly od firmy podařilo získat a mohly se naimplementovat v kódu. Při komunikaci se s přístrojem střídavě vyměňují krátké zprávy o délce osmi bytů. Každá má své dané ID, start a stop byte a kontrolu integrity dat. Zbytek, tedy čtyři byty, slouží pro přenos dat. Detailněji je protokol popsán v kapitole 6.4.3.

5.2.4 Poskytovaná data

Tento tlakoměr poskytuje hodnoty získané při měření automatickou auskulační metodou. Jedná se o:

- naměřenou hodnotu středního arteriálního tlaku,
- vypočítané hodnoty systolického a diastolického tlaku,
- změřenou tepovou frekvenci,
- čas měření.

Přesnost měření jsou 2% u hodnot tlaku a 4% u srdečního tepu.

5.3 Hrudní pás

Zajímavým senzorem byl pro mě hrudní pás od společnosti Garmin [49], který měří tepovou frekvenci srdce, v hodnotě 1600,- Kč.

5.3.1 Uživatelský pohled

Komplet sestává ze samotného senzoru, který se připíná na pružný pás. Pás má v sobě integrované elektrody, které přenáší elektrické signály srdce do senzoru. Ke správné funkci musí být vodivé spojení mezi kůží a třemi body pásu. Dva jsou vpředu symetricky deset centimetrů od středu těla a zbývající bod na levém boku.

Hrudní pás se obléká kolem těla ke spodní hraně prsních svalů. Délka pásu je nastavitelná a měla by být taková, aby pás bezpečně držel na těle a při pohybu nesjížděl dolů, ale zároveň, aby neomezoval pohyb hrudního koše při dýchání.

Senzor se připíná doprostřed hrudního pásu dvěma kovovými cvoky, které umožňují přenos signálu generovaného srdcem. Senzor na sobě nemá žádné ovládací prvky ani display. Je napájen „knoflíkovou“ baterií CR2032, pro jejíž výměnu je třeba odmontovat čtyři šroubky držící zadní stranu těla senzoru.

5.3.2 Měření

Pás se nasadí na hrudník pod prsní svaly a připne se na něj senzor. Ten okamžitě začne analyzovat elektrické potenciály naměřené na elektrodách a detekovat v nich srdeční tep. Naměřená data při tom průběžně vysílá do okolí.

5.3.3 Získání dat

Tento senzor srdečního tepu komunikuje prostřednictvím technologie ANT. Navíc implementuje ANT+ profil pro tento druh zařízení - HR (Heart Rate Monitor). Tím výrazně zjednodušuje implementaci veškeré komunikace. Jen pro zajímavost uvádím, že tento profil byl vůbec prvním definovaným ANT+ profilem.

K implementaci standardních ANT+ profilů v mobilních aplikacích pro Android je na stránkách ANT aliance k dispozici ANT+ SDK [50], které zjednodušuje komunikaci na naprosté minimum. Veškeré komunikační detaily jsou totiž již implementované a SDK k nim poskytuje API. Implementace tohoto API je v mobilní aplikaci ANT+ Plugins, která musí být na cílovém zařízení nainstalována. Každý plugin implementuje jeden ANT+ profil. Pro testování konektivity ANT+ profilů slouží dříve zmíněná aplikace ANT+ Plugin Sampler. Implementace komunikace je ukázána v kapitole 6.4.1.

5.3.4 Poskytovaná data

Využitím ANT+ pluginu pro monitory srdečního tepu lze získat:

- čas posledního úderu srdce,
- čas předchozího úderu srdce,
- kumulativní počet úderů srdce,
- srdeční tep vypočítaný senzorem,
- R-R interval vypočítaný pluginem,
- kumulativní operační čas senzoru,
- ID výrobce a sériové číslo senzoru,
- data specifická pro výrobce.

Přesnost času úderů srdce je 1/1024 sekundy. R-R interval je počet milisekund mezi dvěma posledními údery srdce.

5.4 Váha

Třetím senzorem v pořadí je biometrická váha opět od společnosti eVito [51], aktuálně v hodnotě 3000,- Kč.

5.4.1 Uživatelský pohled

Vzhledem se jedná o klasickou váhu tradičního tvaru. Na pohled vypadá moderně díky skleněné vrchní ploše a čtyřem vodivým kovovým destičkám. K ovládání slouží tři tlačítka umístěná na pravém boku, jimiž jsou šipka nahoru, šipka dolů a tlačítko pro nastavení hodnoty. Pro zobrazení hodnot měření či nastavení slouží velký digitální display.

Váha podporuje až osm osobních profilů pro měření dodatečných metrik (vše kromě tělesné váhy) konkrétních osob. Pro každý profil se nastavuje pohlaví, výška v centimetrech, věk a úroveň aktivity. Všechny tyto údaje se zohledňují při výpočtu dodatečných metrik.

Pro úroveň aktivity slouží škála o čtyřech možnostech:

1. žádná fyzická zátěž,
2. minimální zátěž - procházka, zahradničení,
3. střední zátěž - fyzická aktivita 30 minut 2-4x týdně,
4. vysoká zátěž - fyzická aktivita 30 minut 4-6x týdně.

5.4.2 Měření

Pro správné měření je nutné splnit několik základních podmínek. Váha musí být položena na rovném tvrdém povrchu (z osobní zkušenosti měří o dvě kila méně na koberci než na dlažbě). Aby byla měření porovnatelná, měřená osoba by měla být ve fyzické pohodě, tj. alespoň 15 minut po probuzení, na lačno, po toaletě a bez pohybu na váze. Také je dobré měřit se vždy ve stejný čas. Pokud je zájem měřit také dodatečné metriky, měla by osoba stát bosými chodidly na vodivých ploškách.

Před vážením se nejprve vybere profil konkrétní osoby. Poté se měření zahájí vstupem na váhu. Nejprve se zváží hmotnost osoby a poté se přejde k měření dodatečných metrik, k čemuž se využívá principu bioelektrické impedanční analýzy

(BIA). Po měření se zobrazí všechny výsledky na displayi, přičemž hmotnost se pravidelně střídá s bazální (při zobrazení ikoně pohlaví) a denní (při zobrazení ikoně aktivity) spotřebou energie.

Při měření pomocí BIA [52] se do těla vyšle slabý proud a zkoumá elektrickou impedanci těla pro zjištění jeho složení. Různé složky těla kladou různý elektrický odpor toku proudu vysílaného z jedné nohy do druhé. Nejprve se na základě naměřené impedance odhadne celkový obsah vody v těle, podle něho se odhadne obsah tělesné hmoty bez tuku. Nakonec se s využitím celkové váhy zjistí obsah tuku v těle.

Váha takto může měřit hodnoty jako samostatná jednotka. Pokud však chceme naměřené hodnoty přečíst dalším zařízením postupuje se jinak. Čtecí zařízení si od váhy aktivně vyžádá měření, přičemž při požadavku na pokročilé měření váze odešle také uživatelský profil, pro který se má měření provést. Poté, co se uživatel nechá změřit, se naměřené hodnoty odešlou zpět žádajícímu zařízení.

5.4.3 Získání dat

Tato váha používá ke komunikaci technologii ANT a implementuje ANT+ profil pro tělesné váhy WGT - Weight Scale. Díky tomu byla, stejně jako u hrudního pásu s využitím příslušného pluginu, komunikace velmi jednoduchá.

Nejprve si uživatel vybere váhu, se kterou se hodlá změřit a následně lze zaregistrovat různé posluchače. Zde jsou dva způsoby získání dat. Prvním jsou broadcasty, které váha vysílá po buď základním měření (měření hmotnosti) nebo průběžně s informacemi o zařízení. Druhým jsou odpovědi na vyžádané základní (pouze hmotnost) nebo pokročilé měření (včetně BIA a odhadu energií). Vyžádání měření se provede jednoduše zavoláním příslušné metody nad objektem reprezentujícím váhu. Detaily jsou popsány v kapitole 6.4.2.

5.4.4 Poskytovaná data

Využitím ANT+ pluginu pro monitory srdečního tepu lze získat:

- tělesnou hmotnost (kg),
- procento hydratace těla,
- procento tělesného tuku,
- procento svalové hmoty,
- hmotnost kostí (kg),
- klidová potřeba energie (kcal/den),
- aktivní potřeba energie (kcal/den),

Klidová potřeba energie alias bazální metabolismus je množství energie, které tělo spotřebuje v absolutním klidu (např. celodenní pobyt na lůžku). K jejímu výpočtu se používá Harris-Benedictova rovnice. Aktivní potřeba energie je množství energie spotřebované organismem při provádění činnosti v míře nastavené v uživatelském profilu. Pokud se jídlem a pitím dodá méně energie, rozdíl se získá z spotřebou uložených tuků. Pokud se dodá více, energie se začne ukládat.

5.5 Glukoměr

Posledním senzorem, se kterým jsem pracoval je glukometr FORA Diamond MINI [53] v hodnotě 900,- Kč, který stejně jako dříve popsany tlakoměr vyrábí společnost FORA Care Suisse AG.

5.5.1 Uživatelský pohled

Celý set sestává z měřicího přístroje, odběrového pera a dvou spotřebních materiálů - jehel (lancety) a testovacích proužků. Samotný glukometr je malé podlouhlé zařízení s displayem a hlavním tlačítkem na přední straně a zapuštěným tlačítkem pro nastavení na zadní straně.

Stejně jako u tlakoměru je nutné v nastavení zadat správný čas. Další pro nás zajímavou možností je Bluetooth párování se čtecím zařízením. Mezi ostatní možnosti patří mazání uložených záznamů a nastavení zvukových upomínek.

Odběrové pero je podobné psacímu a slouží pro jednoduché provedení malého vpichu do slabé kůže (doporučeno do bříška prstu), aby se dal snadno získat malý vzorek krve. Pootočením se odejme hrot pera a vloží se odkrytá lanceta. Po vrácení hrotu a natažení pružiny vytažením konce pera je příprava na vpich dokončena. Po přiložení pera k prstu se zmáčkne odjišťovací tlačítko a dojde k průniku lancety do kůže. Rotací hrotu pera lze nastavit šest možných hloubek vpichu.

5.5.2 Měření

Nejprve se připraví odběrové pero ke vpichu. Poté se do glukometru vloží nový testovací proužek. Perem se provede vpich, po čemž by měla začít vytékat kapka krve. První kapka se utře a druhá se přiložením nasaje do kanyly testovacího proužku. Okénko kanyly by mělo být celé zaplněno krví. Potom přístroj začne zkoumat vzorek a nakonec zobrazí naměřené hodnoty.

Glukometr používá pro měření elektrochemickou metodu [54]. Glukóza v nasáté krvi je oxidována katalyticky za přítomnosti enzymů GDH-FAD. Čím více je glukózy v krvi, tím více vznikne peroxidu vodíku. Ten je elektrolýzou rozložen na ionty a jejich přesunem k elektrodám vzniká měřitelný proud, jehož velikost odpovídá hodnotě glykémie.

Přístroj umožňuje čtyři druhy měření. Na lačno, 2 hodiny po jídle, obecný režim a testovací režim. U testovacího režimu se měření provádí na testovacím roztoku, kdy porovnáním získaného a očekávaného výsledku se ověří přesnost měření. Pro získání testovacích dat jsem použil obecný režim.

5.5.3 Získání dat

Komunikační protokol pro glukometr je stejný a liší se pouze odesílanými zprávami. Protokol se podařilo naimplementovat a čtení dat úspěšně realizovat. Detaily komunikace jsou v kapitole 6.4.4.

5.5.4 Poskytovaná data

Každá uložený záznam obsahuje tyto hodnoty:

- datum a čas pořízení
- obsah glukózy v krvi (mg/dl)
- teplota zkoumaného vzorku (°C)
- kód
- typ

Význam kódu se mi nepodařilo zjistit. Typ je typ měření (na lačno apod.).

6 Mobilní aplikace

Cílovou platformou je operační systém Android. Pro vývoj aplikace [55][56] bylo použito vývojové prostředí Eclipse Juno s pluginem ADK. Aplikace byla střídavě vyvíjena na mobilním telefonu Samsung Galaxy S2 a v Android emulátoru. Použití emulátoru bylo vynuceno pro testování komunikace přes technologii ANT, kterou nemá telefon integrovanou a připojení anténky přes USB OTG nefungovalo. Pro testování komunikace přes Bluetooth a další záležitosti byl povětšinou využit smartphone.

Pro verzování zdrojových souborů byl použit nástroj GIT s repozitářem ve službě Github. Pro získávání závislostních balíčků byl použit nástroj Maven.

Aplikace slouží ke sběru dat ze zdravotních senzorů, která lokálně ukládá a to pro více uživatelských účtů. Dále dovoluje uložená data nahrát na EEG/ERP Portál ke zvolenému experimentu.

6.1 Popis činnosti aplikace

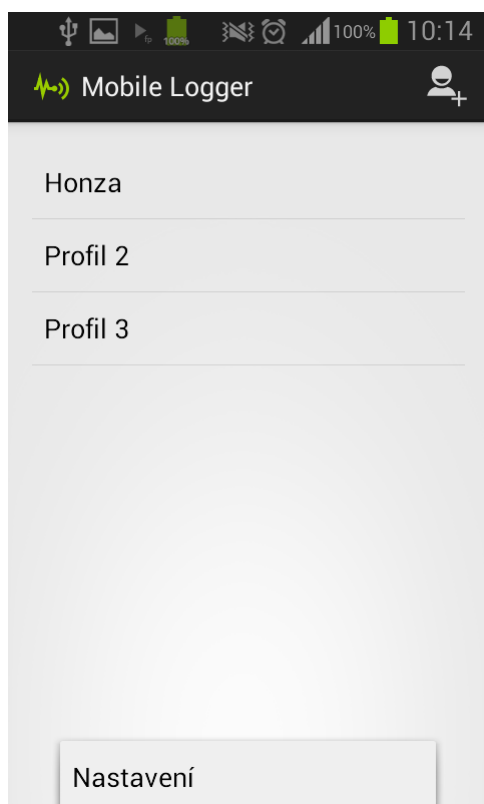
Jako první se vytvoří instance vlastní třídy `Application`. Tato implementace má několik povinností:

- inicializuje databázi,
- inicializuje nastavení načtením výchozích hodnot
- uchovává uživatelský profil,
- poskytuje přístup k databázi, aktuálnímu uživateli
- vypíná/zapíná automatický upload dat
- poskytuje internacionalizované řetězce pro části bez vlastního kontextu (např. adaptéry).

Objekt inicializuje databázi, aby existovala také jen jedna instance a Android se správně postaral o transakce při volání z různých míst najednou. Přečlovává profil aktuálně přihlášeného uživatele, přičemž zapíná a vypíná automatický upload při změnách profilů.

6.1.1 Uživatelské profily

Po inicializaci aplikace přijde na řadu hlavní aktivita, se kterou již uživatel aktivně interaguje. Hlavní aktivita představuje vstupní bod aplikace. Stará se o správu uživatelských účtů. Aplikace podporuje více uživatelských účtů a tato aktivita nabízí jejich seznam. Dovoluje profily upravovat, mazat a vytvářet nové. Také přes menu umožňuje přechod do nastavení aplikace (viz kapitola 6.1.4 a Obr. 15).



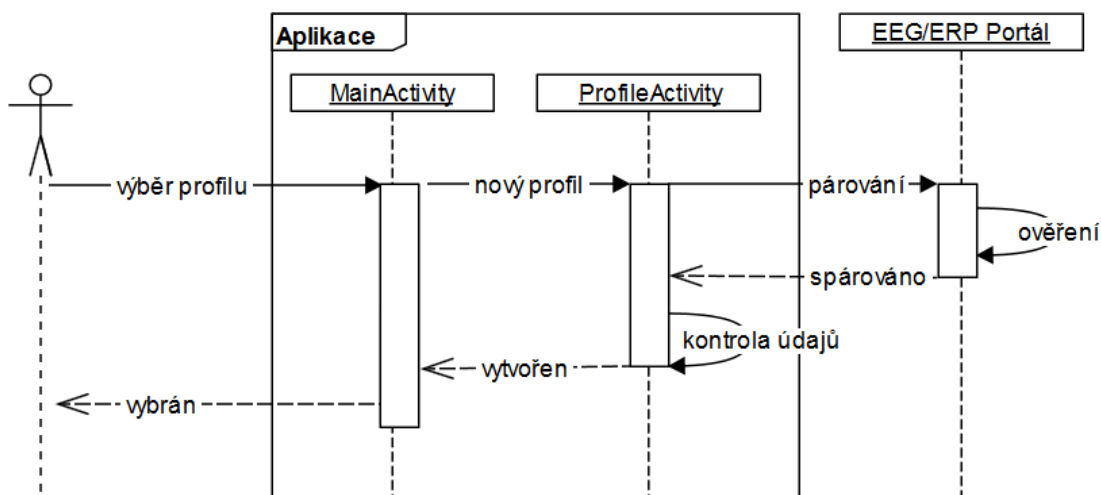
Obr. 15: Hlavní aktivita.



Obr. 16: Tvorba profilu.

Pro vytváření nebo úpravu uživatelského profilu slouží `ProfileActivity` (viz Obr. 16). U každého profilu se musí uvést několik informací. Prvním je název profilu, který se zobrazí v seznamu profilů. Dalšími položkami jsou hodnoty požadované některými senzory, jako jsou datum narození pro výpočet věku, výška, váha apod.

Zbývají tři nepovinné údaje - email, jméno a příjmení. Email označuje účet, nad kterým má uživatel kontrolu v EEG/ERP Portálu. K párování s tímto účtem se používá `LoginActivity`. Zde jsou dvě pole pro email (přebírá se předvyplněná hodnota z předchozí aktivity) a heslo. Po potvrzení dojde k pokusu o přihlášení do Portálu, při jehož úspěchu se účet spáruje s lokálním profilem. Po spárování se automaticky doplní jméno a příjmení z účtu na serveru.



Obr. 17: UML - práce s profily.

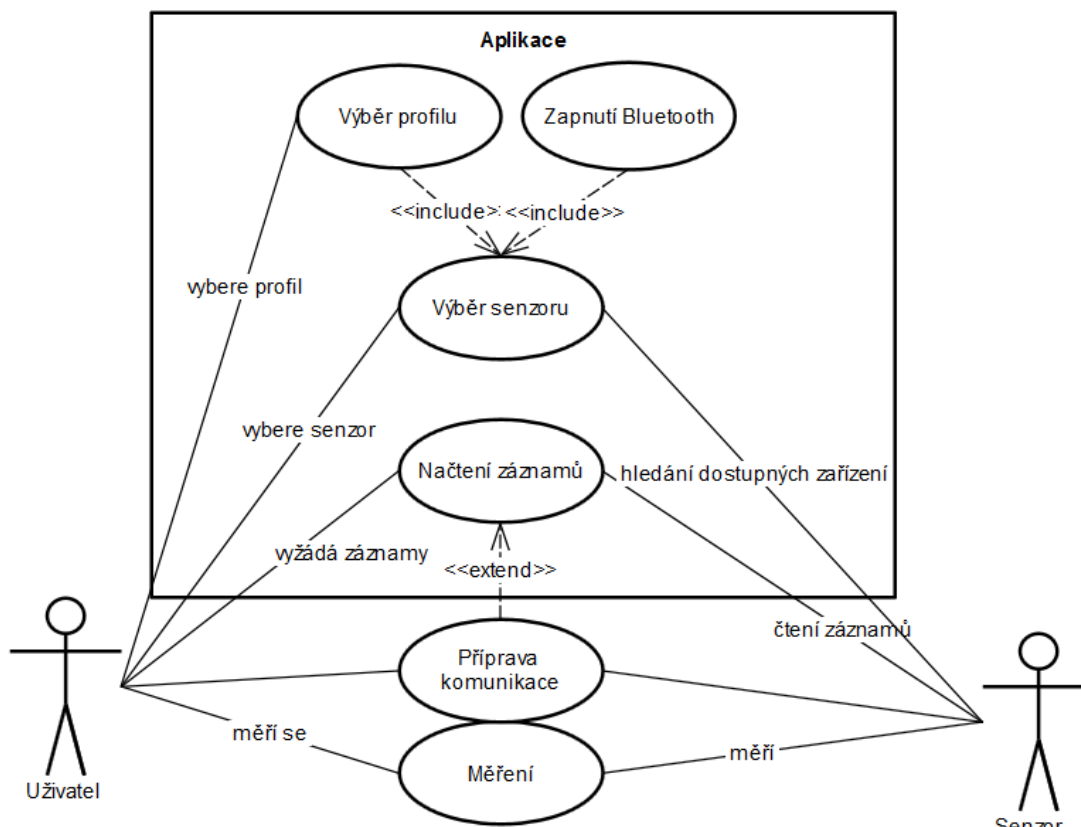
Celý proces je vidět v sekvenčním diagramu na Obr. 17. Po výběru profilu hlavní aktivita nastaví aktuální profil do singletonu `Application`. Následuje spuštění aktivity se seznamem podporovaných (typů) senzorů.

6.1.2 Obsluha senzorů

`DeviceListActivity` zobrazuje seznam podporovaných senzorů. Se senzory lze komunikovat, pokud jsou splněny základní podmínky:

- na straně mobilní aplikace a použitého zařízení
 - podpora technologie ANT (včetně nainstalovaných služeb)
 - zapnutý Bluetooth
 - vybraný senzor ke komunikaci
- na straně senzoru (viz kapitola 5)
 - režim senzoru podporující komunikaci

Jednotlivé typy senzorů vyžadují lehce rozdílnou obsluhu. Obecný postup při měření a získání výsledků je na Obr. 18. Každá obslužná aktivita obsahuje tlačítko umožňující manipulovat s uloženými daty. Toto je popsáno v kapitole 6.1.3.

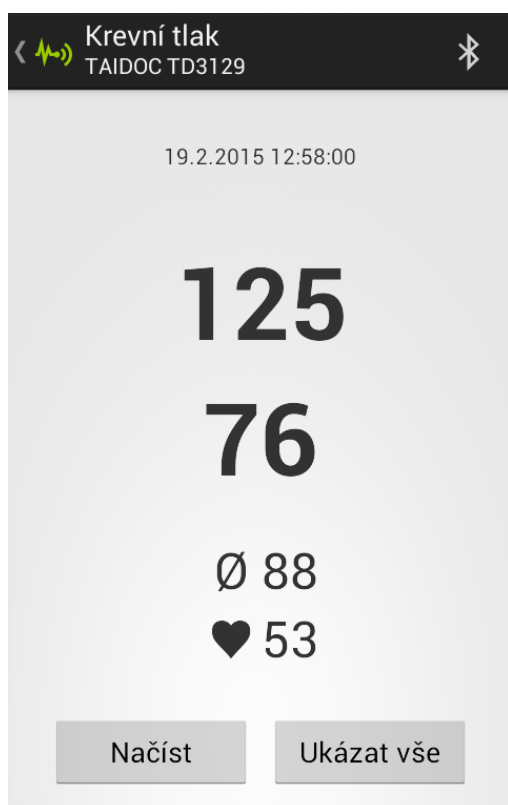


Obr. 18: UML - práce se senzory.

Pro komunikaci s tlakoměrem musí být ve čtecím zařízení zapnut Bluetooth. Pokud není, aplikace k tomu uživatele vyzve. Poté je nutné vybrat konkrétní tlakoměr pro komunikaci. K tomu slouží možnost v menu (action bar), která k tomu využívá známého dialogu systému Android. Nyní lze tlačítkem načíst poslední měření, jehož hodnoty se přehledně zobrazí v této aktivitě (viz Obr. 19 v porovnání s tlakoměrem na Obr. 20). Zároveň se měření uloží do databáze, pokud již nebylo uloženo dříve. Další možností je stáhnout všechny záznamy uložené v tlakoměru. K tomu slouží příslušná možnost v menu. Opět se všechny načtené záznamy uloží do lokální databáze, přičemž se vynechají již v databázi uložené.

Aktivita pro glukoměr, díky totožnému principu komunikace a ukládání dat, je obdobná té pro tlakoměr. Liší se pouze zobrazovanými údaji.

Další je aktivita obsluhující tělesnou váhu. Jak je zmíněno v kapitole 5.3.3, u tohoto senzoru se musí měření aktivně vyžádat. K dispozici tak je tlačítko pro vyžádání základního měření (pouze hmotnost) a tlačítko pro vyžádání pokročilého měření (včetně hodnot BIA). Při spuštění aktivity se automaticky zobrazí další aktivita pro výběr konkrétní ANT+ váhy, jež se má použít. Zbývajícím tlačítkem se opět přechází na správu uložených dat pro tento senzor.



Obr. 19: Zobrazení tlaku v aplikaci.

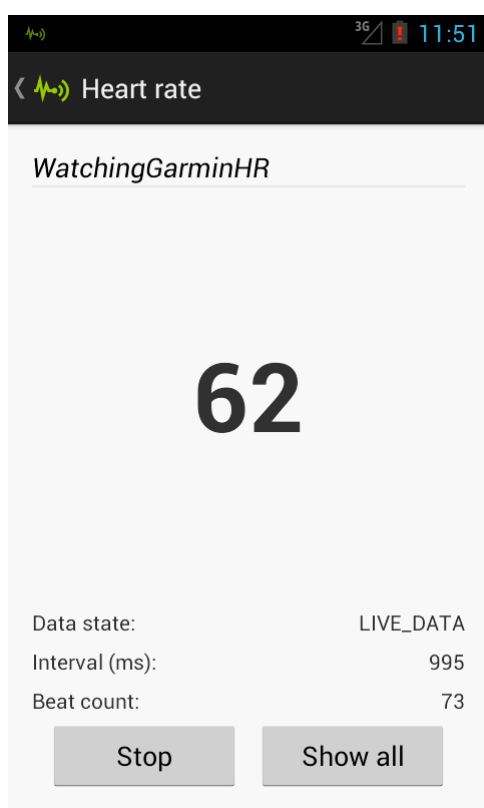


Obr. 20: Zobrazení tlaku tlakoměrem.

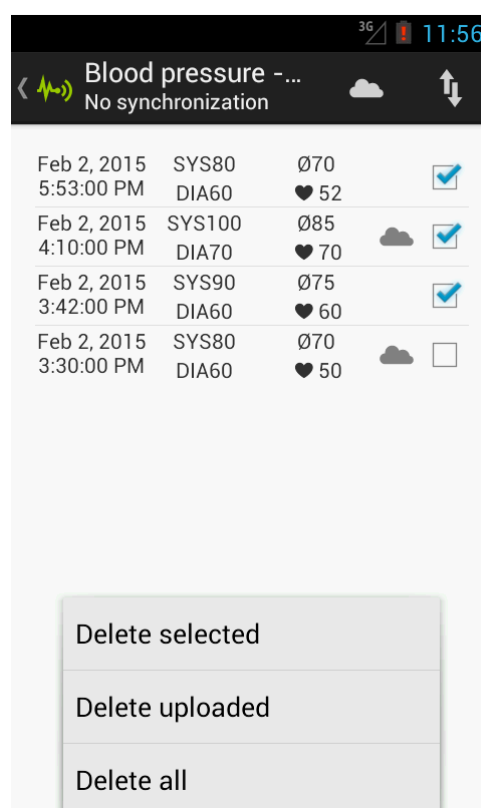
Poslední dostupnou je aktivita pro průběžné zobrazování údajů o srdečním tepu (viz Obr. 21). Díky tomu, že měření tepu probíhá dlouhodobě, bylo nutné zvolit jiný přístup čtení. Tato aktivita pouze zobrazuje naměřené výsledky, zatímco komunikace se odehrává ve službě na pozadí. Tlačítkem pro spuštění aktivity spustí službu, pokud ještě neběží, a zaregistruje u ní posluchače pro příjem měření tepu. Stejným tlačítkem se od služby odpojí, přičemž služba však zůstane běžet. Služba je popsána dále.

Služba pro dlouhodobou komunikaci se senzory na pozadí je tzv. foreground služba. Po celou dobu jejího běhu je k dispozici nezrušitelná notifikace upozorňující uživatele o tom, že je služba aktivní. Výběrem notifikace se dostaneme do aktivity

ovládající službu. Zde lze službu vypnout (pokud je ke službě zrovna připojena jiná aktivita, služba se vypne až po jejím odpojení). Pro každý senzor komunikující dlouhodobě služba inicializuje tzv. komunikátor, který implementuje logiku komunikace a oznamování událostí prostřednictvím registrovaných posluchačů. Aktuálně je implementován jeden komunikátor pro senzory podporující ANT+ profil Heart Rate. Po prvním vyžádání údajů o srdečním tepu komunikátor zobrazí, stejně jako u tělesné váhy, seznam dostupných zařízení a po výběru začne načítat naměřená data, poskytovat je registrovaným posluchačům a zároveň je ukládat do databáze.



Obr. 21: Zobrazení srdečního tepu v aplikaci.



Obr. 22: Manipulace se záznamy krevního tlaku.

6.1.3 Manipulace s uloženými daty

Každá aktivita pro obsluhu senzorů má tlačítko přepínající na aktivitu pro manipulaci dat. Zde lze procházet databázi v podobě seznamu uložených záznamů chronologicky od nejnovějšímu k nejstaršímu. U každého záznamu je zatrhávací políčko, díky kterému lze jedno nebo více měření označit pro další manipulaci (ukázka pro krevní tlak na Obr. 22 výše).

Jednou z možností je mazání. Jeden záznam lze smazat volbou z kontextového menu. Pomocí označování lze vybrat více záznamů a všechny vybrané smazat příslušnou volbou v menu. V menu jsou další dvě možnosti pro mazání záznamů. Samozřejmostí je možnost smazat všechna data z prohlíženého seznamu (nezávisle na označení). Poslední je mazání záznamů, které byly označeny jako odeslané.

Druhou je odesílání dat. Základní možností je označit jednotlivé záznamy ručně a tlačítkem z menu (action bar) údaje odeslat. Pokud není aktuální uživatelský profil spárován s účtem na Portále, je uživatel přesměrován na párování. V opačném případě se zahájí komunikace se serverem, která je popsána v kapitole 6.5.1.

Další možností je nastavení automatického uploadu. K tomu slouží druhé tlačítko v action baru. To dovolí předem vybrat experiment, pro který se budou data na serveru ukládat a následně se budou nově dostupné záznamy automaticky odesílat na server. Toto odesílání se řídí standardními mechanismy synchronizace systému Android a je popsáno v kapitole 6.5.2.

Po jakémkoli úspěšném uploadu (ručním či automatickém) se záznam označí jako odeslaný. To je v seznamu indikováno ikonou mráčku.

6.1.4 Nastavení

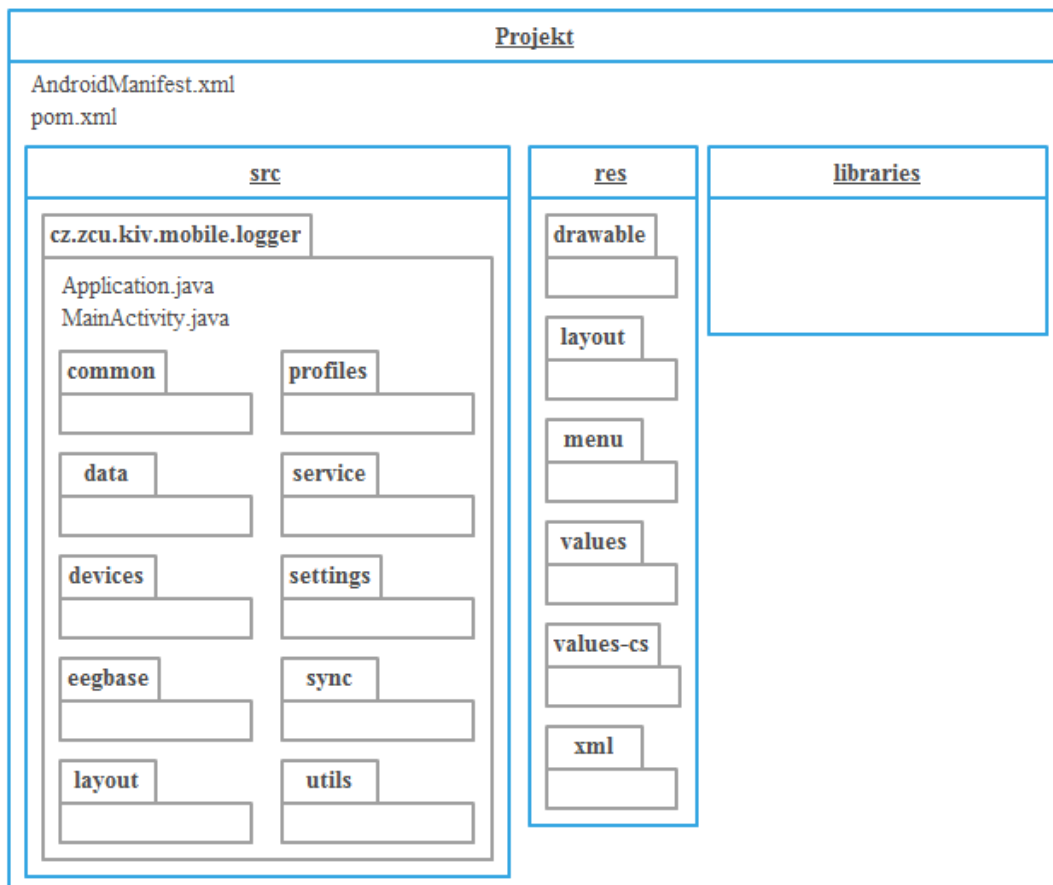
Aplikace umožňuje uživateli nastavit některé používané hodnoty. V sekci Připojení se dá nastavit URL pro RESTové rozhraní cílového EEG/ERP Portálu, na který se budou nahrávat data. V sekci General Parameters lze pro každý typ dat nastavit, pod jakým jménem se na serveru uloží. Dále lze vybrat, zda se mají nově nahraná data na serveru přidat k již existujícím nebo se stará nahradí novými.

6.2 Architektura

Zdrojové soubory jsou rozděleny do balíčků, členění zdrojů je dáno potřebami Androidu. Důležitými soubory v kořenovém adresáři jsou manifest aplikace pro systém Android a objektový model projektu (POM) pro nástroj Maven. Ve složce libraries jsou použité knihovny pro vývoj ANT, která nejsou dostupná v maven repozitářích.

K projektu se přidají jako klasické závislosti do sekce dependencies, ale se scope system a cestou k nim nastavenou v systemPath.

Struktura projektu je zachycena na Obr. 23.



Obr. 23: Struktura projektu.

6.2.1 Hlavní balík `cz.zcu.kiv.mobile.logger`

V hlavním balíku jsou dva základní soubory celé aplikace. `Application` je singleton představující běžící aplikaci. Vlastní implementace tohoto objektu se registruje atributem `name` elementu `application` v manifestu. Systém Android zajistí vytvoření jedné jeho instance při startu aplikace a není potřeba zajišťovat vlastní implementaci návrhového vzoru singleton. Druhým souborem je hlavní aktivita aplikace, která je popsána v kapitole 6.1.

6.2.2 Balík common

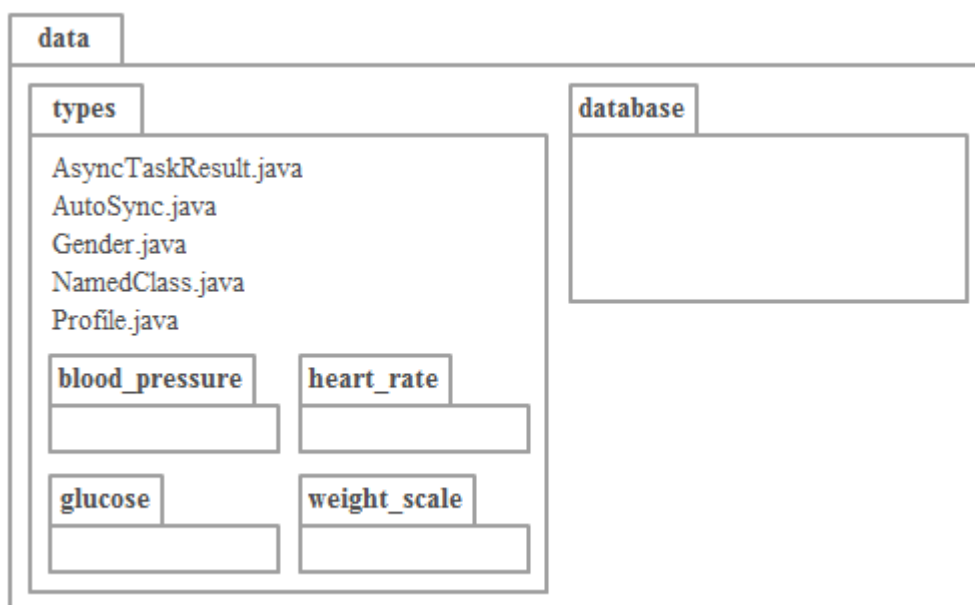
V tomto balíku (Obr. 24) jsou dvě abstraktní aktivity. Obsahují stejný kód pro testování, zda je vybrán uživatelský profil. V případě, že není vybrán, je uživatel přesměrován na výběr profilu. Jedna z aktivit musí být předkem všech ostatních, které závisí na uživatelském profilu. V aplikaci se jedná téměř o všechny, např. aktivity ukládající naměřená data ze senzorů do databáze. Jedna aktivita dědí od obecné aktivity a druhá od `ListActivity` pro případy, kdy je potřeba zobrazit seznam záznamů.



Obr. 24: Obsah balíku common.

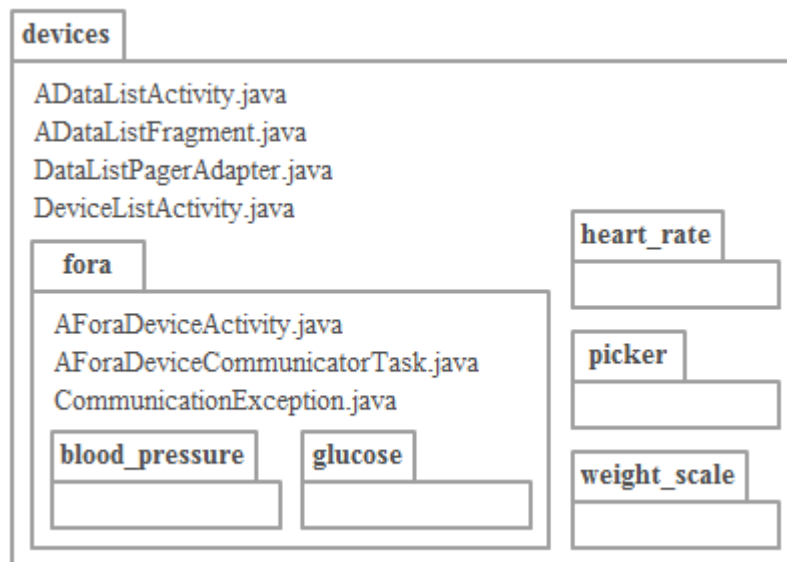
6.2.3 Balík data

V tomto balíku (Obr. 25) jsou dva typy zdrojových kódů. Jedná se hlavně o třídy pro manipulaci s databází v podbalíku `database`. Práce s databází je blíže popsána v kapitole 6.3. V podbalíku `types` je většina používaných POJO tříd a výčetových typů.



Obr. 25: Obsah balíku data.

6.2.4 Balík devices



Obr. 26: Obsah balíku devices.

Všechny aktivity, které slouží pro obsluhu senzorů, a jejich další podpůrné třídy jsou v tomto balíku (Obr. 26). V kořeni je již dříve zmíněná aktivita pro výběr typu senzoru a také připravené abstraktní třídy používané při manipulaci s daty.

Díky stejnému principu komunikace u FORA senzorů velká část kódu, která se stará o obsluhu a komunikaci a která je společný, umístěna do abstraktních předků. Detaily jsou doplněny v potomcích pro konkrétní senzor.

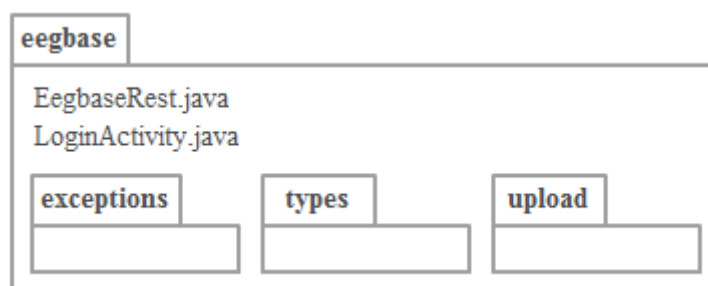
Pro každý senzor je zde několik typů tříd. Hlavní je aktivita pro obsluhu senzoru a zobrazení aktuálního měření. Další je aktivita pro manipulaci s uloženými daty (viz kapitola 6.1.3). Jelikož manipulace s daty je všude stejná a zahrnuje načtení a zobrazení seznamu záznamů, jejich výběr s možností mazání a uploadu, množství kódu bylo opět možno dědit, přičemž tento je umístěný ve zmíněných abstraktních třídách v kořeni balíku.

Senzory mohou poskytovat jeden typ dat (např. FORA senzory) nebo více typů dat (např. ANT+ váha). Pro zjednodušení implementace jejich manipulace jsem bral první případ jako speciální případ druhého. Více typů dat jsem implementoval přecházením (horizontálně posunem prstu) mezi obrazovkami pomocí PagerAdapteru,

kdy na každé stránce je jeden typ dat. Každá stránka potřebuje `Loader` pro načítání z databáze, `Adapter` pro mapování načtených dat do `View` pro záznamy a `Fragment`, který obsluhuje logiku. Tento přístup byl tedy zachován i pro jeden typ dat, aby se vyhnulo zbytečným duplicitám. V případě jednoho typu dat je také skryt panel s nadpisem popisujícím o jaký typ dat se jedná.

Posledními v tomto balíku jsou tzv. pickery. Jsou to aktivity pro výběr zařízení, která jsou v blízkosti dostupná. Implementován je picker pro Bluetooth a pro dva ANT+ profily - váha a srdeční tep.

6.2.5 Balík eegbase



Obr. 27: Obsah balíku eegbase.

Zde se nachází všechny třídy implementující komunikaci s EEG/ERP Portálem (Obr. 27). Komunikace s Portálem zahrnuje přihlašování, získávání dostupných experimentů a upload lokálních dat pro vybraný experiment. V kořenu je `LoginActivity` implementující přihlašování do Portálu a statická třída `EegbaseRest`, starající se o přímou komunikaci prostřednictvím RESTového rozhraní.

V podbalíku `types` jsou umístěny všechny POJO třídy, které se používají pro komunikaci se serverem. Podbalík `upload` zahrnuje funkcionalitu pro nahrávání lokálních dat na Portál. Sem patří výběr z dostupných experimentů, samotný upload a potom objekty typu `UploadHelper`, které z části přispívají k optimalizaci průběhu procesu nahrávání (viz kapitola 6.5.1).

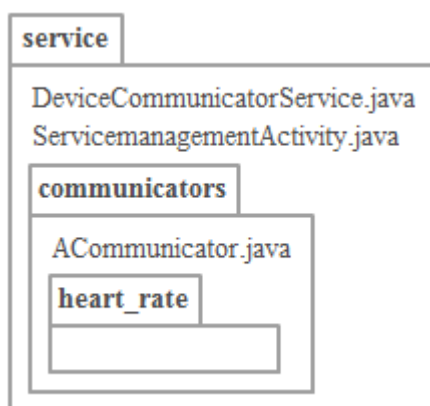
6.2.6 Balík layout

Balík obsahuje vylepšenou implementaci `LinearLayout`, která implementuje rozhraní `Checkable`. Tento layout je použit pro jednotlivé záznamy lokálních dat pro zobrazení v příslušných aktivitách, resp. fragmentech (viz kapitola 6.1.3). Je zde vynucena přítomnost vnořeného `CheckedTextView` označeného identifikátorem `android.R.id.checkbox`, který se použije pro ovládání stavu označení. To umožňuje, aby se daly jednotlivé záznamy vybírat.

6.2.7 Balík profiles

Obsahem balíku je aktivita pro tvorbu či úpravu profilů (viz kapitola 6.1.1) a další podpůrné třídy. `ProfileLoader` slouží pro asynchronní načítání profilů uložených v databázi a `DatePickerFragment` pro komfortní výběr data.

6.2.8 Balík service



Obr. 28: Obsah balíku `service`.

V důležitém balíku `service` (Obr. 28) je implementace služby sloužící pro kontinuální čtení dat ze senzorů, které měří hodnoty průběžně. Konkrétně je zde aktivita pro obsluhu služby, samotná služba a její moduly, tzv. komunikátory. Aktuálně je zde pouze jeden komunikátor pro měřiče srdečního tepu (viz kapitola 6.4.1).

6.2.9 Balík settings

V tomto balíku je pouze jednoduchá aktivita pro zobrazení a manipulaci s nastavením aplikace (viz kapitolu 6.1.4).

6.2.10 Balík sync

Balík `sync` sdružuje všechny implementace požadované operačním systémem Android pro automatickou obsluhu synchronizace. Toho je využíváno pro implementaci automatického upladu dat. Detaily implementace jsou popsány v kapitole 6.5.2.

6.2.11 Balík utils

Zde je několik tříd zjednodušující různé opakované úkony. `AndroidUtils` obsahuje metody pro zjednodušené zobrazování upozornění uživateli. `CloseUtils` provádí uvolňování zdrojů u objektů implementujících i nekompatibilních s rozhraním `Closeable`. `DateUtils` poskytuje několik metod pro práci s časem. Poslední `DialogUtils` obsahuje společný kód pro vytvoření dialogu pro potvrzení mazání.

6.3 Práce s databází

Strukturu databáze zachycuje ERA model v příloze A. 1. Přístupovým bodem k práci s databází v této aplikaci je třída `Database`, která se stará o správu lokální relační databáze. Správa zahrnuje vytvoření, upgrade, otevření databáze a poskytování přístupu k jednotlivým tabulkám. Pro zvýšení přehlednosti a zlepšení rozšiřitelnosti je práce s každou tabulkou umístěna do samostatné třídy.

Trochu méně přehledné je jejich umístění na úroveň třídy `Database`, což má jednoduchý důvod, kterým jsou modifikátory viditelnosti. Tabulka se chová jako samostatný modul a tak se stará také o své vytvoření a upgrade. Při potřebě těchto úkonů je pak třída `Database` jednoduše deleguje všem registrovaným tabulkám. K těmto metodám by se pak již nemělo zvenku přistupovat a tudíž by k nim ani neměl být poskytován přístup. Aby nebylo nutné pro každou třídu vytvořit fasádu,

kteřá by jen zpřístupňovala ostatní metody, využil jsem vlastnosti tzv. friendly přístupu, který tyto metody zpřístupní jen ze stejného balíku.

V aplikaci by měla existovat pouze jedna instance třídy `Database` a veškerá manipulace s databází by měla začínat zde, aby nedocházelo k problémům s více současnými přístupy. O inicializaci databáze se stará vlastní implementace `Application` (viz kapitola 6.1).

6.3.1 Objekty typu `Loader`

U aplikací interagujících s uživatelem se klade velký důraz na její responsibilitu. Práce s databází je relativně dlouhotrvající činnost a tak by měla probíhat asynchronně na pozadí, aby mohl uživatel dál aplikaci ovládat a ta se mu nezablokovala.

Pro načítání dat Android poskytuje loader framework. Pokud je potřeba např. načíst seznam profilů pro zobrazení v aktivitě, zaregistruje se loader dědící např. od `AsyncTaskLoader`:

```
getLoaderManager().initLoader(LOADER_PROFILES, bundle, callbacks);
```

Automaticky se pak v loaderu volají metody pro načtení dat či uvolnění zdrojů a v registrovaném posluchači (callbacks) metody pro zpracování výsledku načítání.

Výsledek načítání je parametricky definovaný jeden objekt, přičemž výjimky, vzniklé v průběhu načítání se automaticky nepropagují zpět do aktivity. Aby bylo možné uživatele upozornit na konkrétní chybové stavy, používám pro výsledky načítání objekt `AsyncTaskResult`. Ten v případě chyby obsahuje příslušnou výjimku, v případě úspěchu načtená data.

Při načítání seznamů se jako data vrací databázový kurzor, tudíž se efektivně donačítají potřebná data v průběhu používání.

Loadery mají ještě další výhodu, jíž je automatické znovunačtení dat, pokud došlo k jejich změně. K tomu je potřeba monitorovat změny použitého datového zdroje a při jejich detekování zavolat v loaderu metodu `onContentChanged`.

6.3.2 Objekty typu AsyncTask

Dalším adeptem pro asynchronní zpracování je ukládání záznamů do databáze. K tomu slouží vlastní implementace třídy `AsyncTask`, používané stylem návrhového vzoru `command` [58]. Jako návratovou hodnotu opět používám `AsyncTaskResult`, která slouží k propagaci výsledku či chyby zpět do volající aktivity.

6.4 Komunikace se senzory

V této kapitole popíši detaily komunikace se senzory a v případě senzorů FORA také komunikační protokol a jeho možnosti. Pro komunikaci technologií ANT+ je potřeba, aby v cílovém zařízení byly nainstalovány aplikace ANT Radio Service, ANT+ Plugins Service, popř. ANT USB Service.

6.4.1 Hrudní pás

Jak již bylo zmíněno, hrudní pás podporuje ANT+ profil, ke kterým SDK poskytuje jednoduché API. Nejprve je nutné získat objekt reprezentující konkrétní pás. Po vyžádání objektu se zobrazí aktivita s výběrem dostupných snímačů:

```
PccReleaseHandle<AntPlusHeartRatePcc> heartRateReleaseHandle =
    AntPlusHeartRatePcc.requestAccess(
        userActivity,
        contextToBindTo,
        new IPluginAccessResultReceiver<AntPlusHeartRatePcc>() {
            @Override
            public void onResultReceived(
                AntPlusHeartRatePcc result,
                RequestAccessResult resultCode,
                DeviceState initialDeviceState) {

                switch(resultCode) {
                    case SUCCESS:
                        heartRateDevice = result;
                        subscribeToHrEvents();
                        return;

                    case DEPENDENCY_NOT_INSTALLED:
                        //přesměrovat na stažení chybějící závislosti
                        return;

                    case: ...
                }
            }
        }, deviceStateChangeReceiver
    );
```

Nad získaným objektem se dají zaregistrovat posluchači pro přijetí příchozích dat. Senzor posílá několik typů dat:

```
protected void subscribeToHrEvents() {
    heartRateDevice.subscribeHeartRateDataEvent(
        new IHeartRateDataReceiver() {...});
    heartRateDevice.subscribePage4AddtDataEvent(
        new IPage4AddtDataReceiver() {...});
    heartRateDevice.subscribeCumulativeOperatingTimeEvent(
        new ICumulativeOperatingTimeReceiver() {...});
    heartRateDevice.subscribeManufacturerAndSerialEvent(
        new IManufacturerAndSerialReceiver() {...});
    heartRateDevice.subscribeVersionAndModelEvent(
        new IVersionAndModelReceiver() {...});
    heartRateDevice.subscribeCalculatedRrIntervalEvent(
        new ICalculatedRrIntervalReceiver() {...});
}
```

6.4.2 Váha

Opět díky ANT+ podpoře je komunikace velmi jednoduchá. Získá se objekt reprezentující konkrétní váhu:

```
PccReleaseHandle<AntPlusWeightScalePcc> releaseHandle =
    AntPlusWeightScalePcc.requestAccess(
        userActivity,
        contextToBindTo,
        new IPluginAccessResultReceiver<AntPlusWeightScalePcc>() {
            @Override
            public void onResultReceived(
                AntPlusWeightScalePcc result,
                RequestAccessResult resultCode,
                DeviceState initialDeviceState) {

                switch (resultCode) {
                    case SUCCESS:
                        weightScaleDevice = result;
                        subscribeToWsEvents();
                        break;

                    case: ...
                }
            }
        }
    );
```

Poté lze zaregistrovat posluchače. Zde jsou dva způsoby získání dat. Prvním jsou broadcasty, které váha vysílá buď při základním měření (měření hmotnosti) nebo průběžně s informacemi o zařízení:

```
protected void subscribeToWsEvents() {
    weightScaleDevice.subscribeBatteryStatusEvent(
        new IBatteryStatusReceiver() {...});
    weightScaleDevice.subscribeManufacturerIdentificationEvent(
        new IManufacturerIdentificationReceiver() {...});
    weightScaleDevice.subscribeManufacturerSpecificDataEvent(
        new IManufacturerSpecificDataReceiver() {...});
    weightScaleDevice.subscribeProductInformationEvent(
        new IProductInformationReceiver() {...});
    weightScaleDevice.subscribeBodyWeightBroadcastEvent(
        new IBodyWeightBroadcastReceiver() {...}
    );
}
```

Druhým jsou odpovědi na vyžádané základní (pouze hmotnost) nebo pokročilé měření (včetně BIA a odhadu energií):

```
boolean success = weightScaleDevice.requestBasicMeasurement(
    new IBasicMeasurementFinishedReceiver() {...});
boolean success = weightScaleDevice.requestAdvancedMeasurement(
    new IAdvancedMeasurementFinishedReceiver() {...},
    userProfileANT);
```

6.4.3 Tlakoměr

Prvním krokem je získání objektu reprezentujícího tlakoměr. Pro toto se využívá prostředků systému tak, že se naslouchá broadcastu pro událost výběru zařízení:

```
registerReceiver(
    myReceiver,
    new IntentFilter(
        "android.bluetooth.devicepicker.action.DEVICE_SELECTED" ));
```

a odešle se příslušný záměr:

```
startActivityForResult(
    new Intent("android.bluetooth.devicepicker.action.LAUNCH")
        .putExtra("android.bluetooth.devicepicker.extra.NEED_AUTH", false)
        .putExtra("android.bluetooth.devicepicker.extra.FILTER_TYPE", 0)
        //0 = všechna zařízení
        .setFlags(Intent.FLAG_ACTIVITY_EXCLUDE_FROM_RECENTS),
    REQUEST_PICK_DEVICE);
```

Zahájení komunikace s takto získaným zařízením je v operačním systému Android velice snadné:

```
UUID spp = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
BluetoothDevice device; //získáno broadcastem

BluetoothSocket socket =
    device.createInsecureRfcommSocketToServiceRecord(spp);
socket.connect();
InputStream in = socket.getInputStream();
OutputStream out = socket.getOutputStream();
```

U zařízení FORA se při komunikaci vyměňují zprávy o velikosti 8 bytů s následujícím významem:

1. start byte
2. typ zprávy
3. data 1
4. data 2
5. data 3
6. data 4
7. end byte
8. checksum

Start a end byte jsou neměnné a mají hodnotu 81, resp. -93. Datové byty nesou jednotlivé významové bity, ze kterých se extrahují hodnoty měření. Checksum slouží pro kontrolu integrity. Jeho hodnotou je jednoduše součet všech předchozích bytů, přičemž se pro výpočet používá jeden byte a dochází k přetečení (resp. lze počítat s integery a použít LSB).

Komunikaci začíná klient odesláním zprávy. Tlakoměr pak vrátí odpověď se stejným typem zprávy. Zde jsou zprávy používané tlakoměrem:

- START = 43
- A = 37
- B = 38
- END = 80

Po odeslání START vrátí tlakoměr počet uložených záznamů, ten je neznamínkově uložen v data 1:

```
int recordCount = 0xFF & message[2];
```

Počet lze využít pro cyklus s dotazováním na konkrétní záznamy. Na záznam se ptá jeho pořadovým číslem, přičemž jsou uloženy chronologicky a prvním je nejnovější. Na jeden záznam připadají dvě zprávy (A, B). Byte data 4 zde musí být nastaven na hodnotu 1. U obou se musí nastavit, jaký záznam v pořadí chceme získat. K tomu slouží byty data 1 a 2:

```
msg1[2] = ((byte) record);  
msg1[3] = ((byte)(record >> 8));
```

Kromě prvního záznamu musí být u zprávy A nastaven data 4 byte na hodnotu 0. Pro zprávu A se nám vrátí informace o čase pořízení záznamu. Čas získáme takto:

```
int day = 0xFF & 0x1F & message[2];
int month = (0xFF & 0x07 & message[2] >> 5)
            + ((0xFF & 0x01 & message[3]) << 3);
int year = 2000 + (0xFF & 0x3F & message[3] >> 1);
int minute = 0x3F & message[4];
int hour = 0x1F & message[5];
```

Pro zprávu B tlakoměr vrátí naměřené údaje:

```
setSystolicPressure(0xFF & message[2]);
setDiastolicPressure(0xFF & message[4]);
setMeanPressure(0xFF & message[3]);
setHeartRate(0xFF & message[5]);
```

Poslední zpráva END po odeslání vypne tlakoměr. Může se odeslat kdykoli, např. po přečtení prvního, tj. nejnovějšího, záznamu. Také se nemusí posílat vůbec, potom po uzavření socketu tlakoměr čeká do timeoutu na další spojení.

6.4.4 Glukoměr

Pro komunikaci s glukoměrem platí vše stejně jako u tlakoměru a liší se pouze v odpovědi na zprávu B. Ta obsahuje informace relevantní pro měření glukózy tímto glukoměrem:

```
setGlucose(0xFF & message[2] & 255 + (message[3] << 8));
setTemperature(0xFF & message[4]);
setCode(0xFF & 0x3F & message[5]);
setType(0xFF & 0x3 & message[5] >> 6);
```

6.5 Komunikace se serverem

S EEG/ERP Portálem aplikace komunikuje prostřednictvím RESTful rozhraní. Část tohoto rozhraní byla navržena v důsledku požadavků na tuto práci a je popsána v kapitole 7.2.

Pro implementaci RESTové komunikace jsem po vzoru [57] využil knihovnu `spring-android-rest-template`, pro mapování JSON knihovnu `jackson-databind` a pro mapování XML knihovnu `simple-xml`.

6.5.1 Ruční upload

Po stisku tlačítka pro upload (ikona obláčku) dojde k vyvolání aktivita `UploadGenericParametersActivity`. Jedná se o obecnou aktivitu, která nahrává jakékoli předložené generické parametry. Je tedy nutné vybrané záznamy na generické parametry převést, což lze udělat na dvou místech. Převod lze provést hned a aktivitu volat s již sestavenými POJO objekty připravenými pro mapování do XML. Druhou možností je použít implementaci rozhraní `IExperimentParametersUploadHelper`, které se předají pouze ID záznamů. Díky tomu se přes životní cykly aktivit přenáší jen pole hodnot typu `long` a samotné generické parametry se sestaví až těsně před odesláním.

Pro upload parametrů je potřeba znát experiment, ke kterému se mají na serveru záznamy uložit. K tomu je určena aktivita `SelectExperimentActivity`. Tato aktivita umožňuje stáhnout seznam experimentů dostupných pro aktuálního uživatele. Tento seznam se až do případného znovunačtení (tlačítko refresh) uchovává v odloučeném fragmentu (retained) [56].

Po výběru experimentu se z aktivity pro upload odešlou záznamy na Portál. Po úspěšném odevzdání se záznamy v databázi označí jako odeslané. To však platí pouze při použití implementace `IExperimentParametersUploadHelper`.

6.5.2 Automatický upload

Aplikace umožňuje také automatický upload záznamů. V aktivitě pro manipulaci dat se po vybrání tlačítka automatického uploadu (dvě šipky) spustí výběr experimentu. Po výběru experimentu se do databáze uloží ID experimentu a typ dat. Tato informace je později využita v objektu `SyncAdapter`. Automatický upload je implementován synchronizačním frameworkem Androidu.

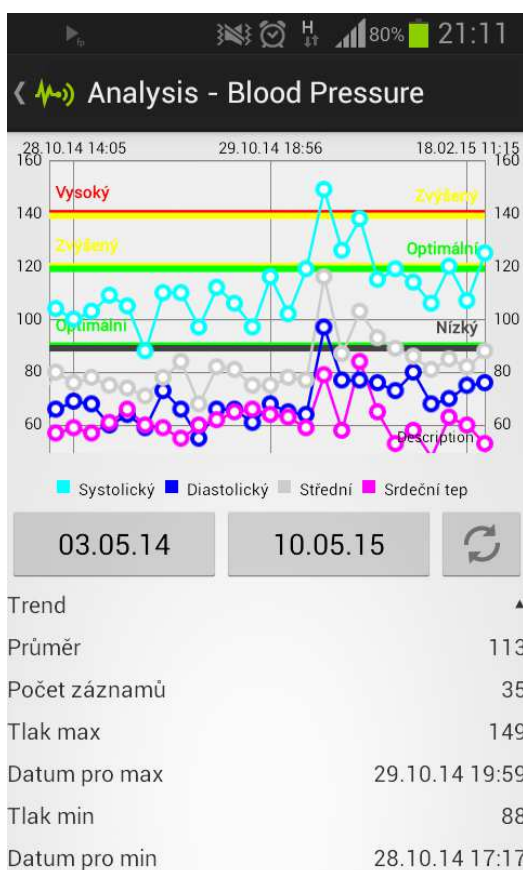
Při implementaci synchronizace jsem se řídil oficiální dokumentací [9], podle níž je ke zprovoznění potřeba několik JAVA a XML souborů a také několik oprávnění v manifestu. Jelikož pro práci s databází není použit `ContentProvider` [55], většina implementovaných tříd jsou jen prázdné spojky (stub). Hlavní implementace se nachází ve třídě `SyncAdapter`. Zde se načtou informace o tom, ke kterému experimentu se mají nahrávat jaké záznamy a data se odešlou.

SyncAdapter je volán automaticky systémem Android. K tomu musí být v nastavení povolen globální přepínač synchronizace.

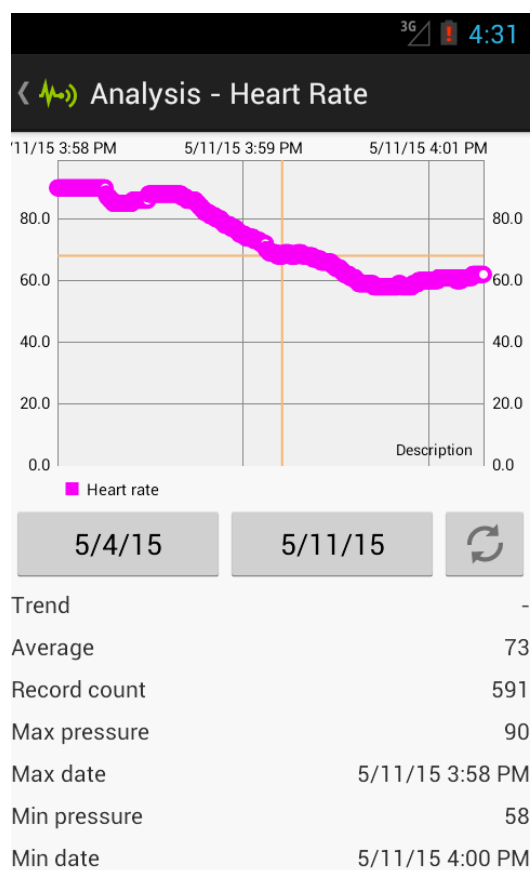
6.6 Základní analýza dat

Každá aktivita pro obsluhu senzoru má, stejně jako přechod k manipulaci dat, možnost spustit aktivitu pro základní analýzu dat. Tyto aktivity umožňují vybrat časové rozpětí pro analýzu. K tomu slouží tři tlačítka, dvě pro výběr hraničních dnů a třetí pro spuštění analýzy. Po spuštění aktivity je automaticky přednastaven rozsah za posledních sedm dnů. V každé aktivitě je vizualizace analyzovaných dat ve formě grafu.

Pro vykreslení grafů byla použita knihovna MPAndroidChart [60]. Ta umožňuje snadno vytvářet různé druhy grafů včetně čárového, sloupcového (vertikální i horizontální), koláčového nebo dokonce kombinovaného sloučením více typů grafu.



Obr. 29: Základní analýza krevního tlaku.



Obr. 30: Základní analýza srdečního tepu.

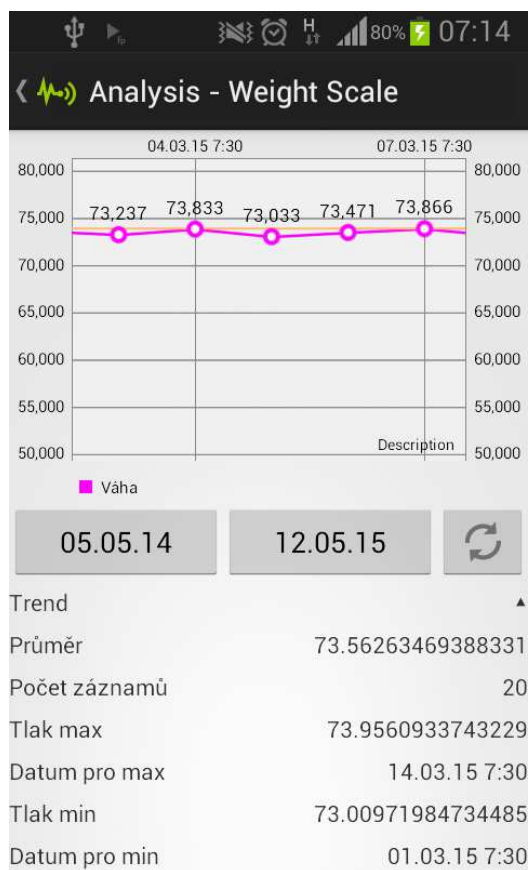
Pro statistiky v této aplikaci bylo využito čárových grafů. Pro ty knihovna poskytuje různá užitečná nastavení, jako jsou limitní čáry nebo použití více setů dat. Pro prohlížení vizualizovaných dat se hodí možnost škálování grafu v obou osách a posun v případně takto zvětšeném grafu.

Při analýze je kromě grafu zobrazena také základní statistika pro první veličinu v grafu. Zjišťuje se trend (klesající, stoupající, stagnující), průměrná hodnota, počet použitých záznamů a nejvyšší či nejnižší hodnoty včetně času, kdy nastaly

Na Obr. 29 je vidět statistika pro měření krevního tlaku. Zde bylo využito limitních čar pro označení různých úrovní systolického tlaku. V grafu je současně zobrazen i střední a diastolický tlak a frekvence srdečního tepu, pro kterou jsou osy v úderech za minutu.



Obr. 31: Základní analýza glykémie.



Obr. 32: Základní analýza váhy.

Statistika záznamů průběžného měření srdečního tepu je vidět na Obr. 30. Zde je i pro krátký časový rozsah relativně hodně záznamů a tudíž zvolení dlouhého intervalu může mít výkonnostní dopad na výpočet analýzy. Statistika pro měření obsahu

cukru v krvi jsou na Obr. 31. Limitními čarami je zde znázorněn pás optimálních hodnot pro zdravého člověka. Na Obr. 32 je analýza pro tělesnou hmotnost. Před pořízením obrázku zde bylo využito zvětšení grafu pro lepší zobrazení hodnot

Při práci s knihovnou bylo naraženo na limity použitého vývojového prostředí. ADT plugin pro Eclipse neumí použít knihovny pro Android, zpravidla poskytované jako AAR balík. MPAndroidChart je poskytován také jako maven závislost, ale ani tak nedošlo k funkčnímu nasazení aplikace. Nakonec musela být využita možnost importu projektu knihovny a její přidání v nastavení projektu aplikace.

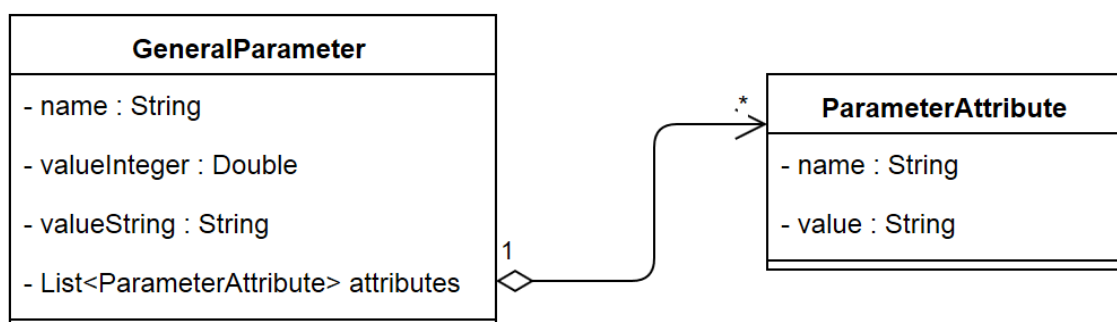
7 Úprava EEG/ERP Portálu

V mobilní aplikaci je naprogramována možnost uploadu posbíraných dat na server. Portál však zatím tuto možnost nepodporoval a bylo nutné ji doimplementovat.

7.1 Ukládání dat

Údaje naměřené senzory přímo souvisí s konkrétním experimentem, proto je dobré, aby byly dostupné z POJO představujícího experiment. V diplomové práci [6] byla do projektu přidána nerelační databáze, do které se ukládají všechna metadata experimentu a týká se to tedy také naměřených dat.

POJO experimentu se nazývá `Experiment` a nachází se v balíku `cz.zcu.kiv.eegdatabase.data.pojo`. Jedním z atributů této třídy je objekt typu `ExperimentElastic`, který obsahuje tu část údajů, jež se mapují do nerelační databáze. V tomto objektu je kolekce parametrů typu `GenericParameter`, které mají vhodnou strukturu pro uložení naměřených dat (viz Obr. 33).



Obr. 33: Struktura objektu `GeneralParameter`.

Hodnotu `name` lze použít pro popis typu dat, např. pro srdeční tep "Heart Rate Measurements". Hodnoty `valueString` nebo `valueInteger` lze využít pro jednorázové typy měření, která mají jen jednu hodnotu. Pro měření s více hodnotami

je vhodná kolekce `attributes` typu `ParameterAttribute`. V tomto případě lze hodnoty `valueString` a `valueInteger` využít také k popisu této kolekce.

`ParameterAttribute` obsahuje pouze dva řetězce. Hodnota `name` lze použít pro jednoznačný identifikátor měření, `value` lze použít pro naměřenou hodnotu. Nevýhodou je, že je nutná konverze dat na řetězec. Naopak výhodou je možnost ukládání měření obsahujících více hodnot, která je možné reprezentovat např. formátem JSON.

7.1.1 Interceptor

V průběhu testování jsem narazil na problém, že se neukládala vložená měření. O zajištění fyzického uložení dat do nerelační databáze se stará framework Hibernate. Ten poskytuje objekt `Interceptor`, jehož metody jsou volány v reakci na různé události při práci s databází, jako je načítání, ukládání atd.

Některá metadata o experimentu jsou stále ukládána do relační databáze a redundantně synchronizována do nerelační databáze. Právě `Interceptor` byl použit v diplomové práci [6] pro synchronizaci. Je zde dočasná metoda `syncExperimentParams`, která se používá při událostech `onSave` a `onFlushDirty`. Její implementace při synchronizaci namapuje metadata z objektu `Experiment` do kolekce v objektu `ExperimentElastic`. Pokaždé však vytvoří novou kolekci a původní parametry zahodí. Toto chování jsem změnil tak, že se stávající hodnoty zachovávají a nové se v kolekci přidávají či aktualizují.

7.1.2 DAO metoda update

Dalším problémem bylo, že se `Interceptor` z předchozí kapitoly nevyvolal v případě, že se pouze přidaly nové prvky do kolekce a nezměnila se také hodnota některých metadat. To značilo, že se měření neukládají, a opravdu po načtení nové hodnoty chyběly.

Tento problém byl nahlášen jako chyba na datové vrstvě a konzultován s jedním z vývojářů Portálu. Řešením byla úprava metody `update` v implementaci rozhraní `cz.zcu.kiv.eegdatabase.data.dao.GenericDAO`, kterou provedl dotyčný vývojář.

7.2 REST služby

Portál již poskytuje několik přístupových bodů pro manipulaci s uloženými daty pomocí RESTful rozhraní. Jejich implementace se nachází v balíku `cz.zcu.kiv.eegdatabase.webservices.rest`. Práci s experimenty zajišťuje kontrolér s názvem `ExperimentServiceController` v podbalíku `experiment`.

Na možnosti REST služeb se klienti odkazují URI adresami. K základní adrese rozhraní se u kontroléru přidá další část pomocí anotace `RequestMapping`.

```
@RequestMapping("/experiments")
public class ExperimentServiceController
```

Poslední část adresy označující přístupový bod (tzv. endpoint) je stejnou anotací určena u metod, které implementují jeho logiku.

7.2.1 Získání seznamu experimentů

Podle kapitoly 6.5 je potřeba nejprve poskytnout seznam experimentů. Taková metoda již v kontroléru existuje:

```
@RequestMapping(value = "/mine")
public ExperimentDataList getMyExperiments() {
    return new ExperimentDataList(service.getMyExperiments());
}
```

Nevýhoda použití této metody je, že vrací veškeré údaje dostupné o experimentu. To má dva negativní důsledky. Jednak se při posílání více experimentů prodlužuje celková doba přenosu, což snižuje uživatelskou zkušenost v aplikaci. Dále se plýtvá prostředky mobilního připojení, které bývá pro uživatele limitováno objemem nebo cenou.

To je z pohledu mobilní aplikace zbytečné, protože ta potřebuje pouze identifikátor experimentu a uživateli stačí pro rozhodnutí o výběru jen malá množina údajů. Pro omezení obou nevýhod byla přidána efektivnější metoda. Vybrány byly tyto hodnoty:

- ID experimentu,
- název experimentu,
- název výzkumné skupiny,
- čas experimentu,
- jméno a příjmení testovaného subjektu.

Metoda s definovanou částí URI a získáváním vybraných údajů vypadá takto:

```
@RequestMapping(value = "/mineInfo")
public ExperimentInfoList getMyExperimentInfoList() {
    List<ExperimentData> experiments = service.getMyExperiments();
    List<ExperimentInfo> infoList =
        new ArrayList<ExperimentInfo>(experiments.size());

    for(ExperimentData experiment : experiments) {
        infoList.add(new ExperimentInfo(
            experiment.getExperimentId(),
            experiment.getScenario().getScenarioName(),
            experiment.getResearchGroup().getGroupName(),
            experiment.getStartTime(),
            experiment.getEndTime(),
            experiment.getSubject().getName(),
            experiment.getSubject().getSurname()));
    }
    return new ExperimentInfoList(infoList);
}
```

7.2.2 Upload měření

Přístupový bod pro upload dat ještě neexistoval a musel být vytvořen. Byla zvolena obecná implementace nezávislá na druhu dat. Toho bylo docíleno strukturou přijímaných dat analogickou k `GenericParameter`. Tím lze endpoint použít i pro jiné parametry než je měření zdravotních metrik a mapování konkrétních dat je ponecháno na volajícím klientovi.

Metoda přijímá seznam parametrů, který se má uložit pro daný experiment. Identifikátor experimentu je předán proměnnou v URI endpointu. Dále byl do dat přidán příznak, zda se mají hodnoty parametrů nahradit, nebo přidat k existujícím.

```
@RequestMapping(
    value="/addGeneralParameters/{experimentId}",
    method = RequestMethod.POST)
public AddExperimentDataResult addGeneralParameters(
    @PathVariable int experimentId,
    @RequestBody ExperimentParametersData data) {
    service.addGeneralParameters(experimentId, data);
    return new AddExperimentDataResult(true);
}
```

Metoda deleguje volání do servisní vrstvy, kde je implementována logika uložení dat. Nejprve se získá experiment podle identifikátoru. Poté se pro každý parametr zkontroluje, zda je již v experimentu přítomen a případně se vytvoří nový. Pokud je nastaven příznak `append` na `false`, smažou se všechny dosavadní atributy parametru. Nakonec se vloží nové hodnoty a prostřednictvím DAO vrstvy se změny uloží.

7.3 Uživatelské rozhraní pro parametry

Dosud se do parametrů experimentů ukládala jen metadata z relační části a neexistovalo pro ně uživatelské rozhraní. Nyní s ukládáním měření ze senzoru vznikla potřeba mít tato data možnost zobrazit a spravovat.

7.3.1 Wicket

Uživatelské prostředí je implementováno komponentovým frameworkem Wicket [3][59] a zdrojové soubory pro něj se nacházejí v balíku `cz.zcu.kiv.eegdatabase.wui.ui`. Každá webová stránka je implementována dvojicí stejně pojmenovaných souborů. HTML soubor slouží pro popis prezentace dat uživateli. V JAVA souboru se pak sestaví model stránky a implementuje se logika jejího chování podobně, jako je tomu u standardní grafické knihovny Swing. Texty (popisky, hlavičky tabulek atd.) se získávají internacionalizačním mechanismem a překlady jsou ve složce `resources`. Zatím jediný překlad je v souboru `messages.properties`.

7.3.2 Úprava stránky experimentu

Stránka pro experiment již existovala a stačilo pouze doplnit výpis seznamu parametrů. Jmenuje se `ExperimentsDetailPage` a nachází se v podbalíku `experiments`. Pro každý parametr se zobrazí jeho jméno a hodnota. Jelikož má parametr možnost výběru mezi textovou a číselnou hodnotou, je vybrána ta, která je nastavená. Dále je u každého parametru odkaz na smazání a na stránku s jeho detailem. Po výběru mazání je uživateli javascriptem zobrazen potvrzovací dialog.

Takto vypadá část HTML souboru s přidáním parametry:

```
<h2>
  <wicket:message key="heading.genericParameters" />
</h2>
<table class="dataTable">
  <thead>
    <tr>
      <th><wicket:message key="heading.parameterName" /></th>
      <th><wicket:message key="heading.parameterValue" /></th>
      <th><!-- column with link for detail --></th>
      <th><!-- column with link for delete --></th>
    </tr>
  </thead>
  <tbody>
    <tr wicket:id="genericParameters">
      <td wicket:id="genericParameterName" />
      <td wicket:id="genericParameterValue" />
      <td wicket:id="detail" />
      <td><a href="#" wicket:id="deleteLink">
        <wicket:message key="link.delete" /></a></td>
    </tr>
  </tbody>
</table>
```

V JAVA souboru se pak k jednotlivým elementům přidají modely s mapováním vlastních hodnot. Je zde využito metody `isVisible` pro skrytí odkazu pro mazání, pokud uživatel nemá dostatečná práva. Výsledek je vidět v příloze A. 3.

7.3.3 Vytvoření stránky parametru

Stránka pro zobrazení detailu parametru musela být vytvořena nová a je umístěna v podbalíku `data` s názvem `GenericParameterDetailPage`. Zde se opět zobrazí jméno parametru, tentokrát obě hodnoty (textová i číselná) a počet atributů v parametru.

Nakonec je na stránce tabulka s výpisem všech atributů. Pro každý atribut je zobrazen jeho název a hodnota. Aby při absenci atributů nebyla na stránce zbytečně hlavička tabulky, je využito elementu `enclosure`:


```
<wicket:enclosure child="attributes">
  <h2>
    <wicket:message key="heading.genericParameterAttributes" />
  </h2>
  <table class="dataTable">
    <thead>
      <tr>
        <th><wicket:message key="heading.attributeName" /></th>
        <th><wicket:message key="heading.attributeValue" /></th>
      </tr>
    </thead>
    <tbody>
      <tr wicket:id="attributes">
        <td wicket:id="attributeName" />
        <td wicket:id="attributeValue" />
      </tr>
    </tbody>
  </table>
</wicket:enclosure>
```

Element `enclosure` se odkazuje na vnořený element `attributes`. V příslušném JAVA souboru je u jeho modelu implementována metoda `isVisible`, která v závislosti na počtu atributů celou tabulku skryje či nechá viditelnou. Výsledná stránka je vidět v příloze A. 4.

8 Zhodnocení výsledků

V průběhu práce se podařilo implementovat aplikaci pro monitorování zdravotních ukazatelů těla. Nese název Elfyz Data Mobile Logger - zkráceně Mobile Logger. Pro zhodnocení se hodí porovnat aplikaci (viz Tab. 1) s ostatními z kapitoly 3.

Většina aplikací se soustředí na monitorování aktivity při vykonávání sportovních činností. K tomu využívají převážně integrované senzory v čele s GPS. Výstupní aplikace této práce naproti tomu klade důraz na komunikaci s externími senzory. V počtu podporovaných druhů externích senzorů si vede lépe pouze aplikace eVito. Při znalosti komunikačních protokolů je však aplikace snadno rozšiřitelná o podporu dalších senzorů, komunikujících zejména prostřednictvím ANT+.

Z povahy porovnaných aplikací se tyto soustředí na měření průběžných hodnot z GPS, srdečního monitoru, krokoměru nebo výjimečně kadence při jízdě na kole. Mobile Logger z tohoto typu měření podporuje monitor srdečního tepu. Umožňuje ale také získat jednorázová měření z tlakoměru, glukoměru či tělesné váhy. Zde opět drží krok jen aplikace eVito.

Všechny aplikace bez výjimky obsahují možnost provést nad daty základní analýzu a zobrazit jejich statistiku. Existující aplikace poskytují analýzu souhrnnou nebo pro konkrétní absolvovaný trénink. Mobile Logger umožňuje vybrat libovolný časový úsek, nad nímž bude analýza provedena.

Až na Health Tracker jsou aplikace součástí systému s webovým serverem, na který se data odesílají a zpracovávají. Webovým serverem pro Mobile Logger je EEG/ERP Portál. Ten aktuálně umožňuje měření pouze ukládat a zobrazit, zatímco ostatní webové aplikace nabízejí také analýzu dat.

	využití integrovaných senzorů	využití externích senzorů	průběžná měření	jednorázová měření	statistiky	webová služba	vzhled
Mobile Logger		4	•	•	•	•	-
Endomondo	2	3	•		•	•	+
Runtastic	2	2	•		•	•	+
Pedometer	1		•		•	•	-
Push-ups	1		•		•	•	+
Heart rate	1			•	•	•	+
Sports Tracker	1	2	•		•	•	+
Health Tracker					•		-
Madbarz					•	•	+
eVito	1	5	•	•	•	•	+

Tab. 1: Porovnání aplikací.

Největší slabinou aplikace je její vzhled, který využívá pouze základních grafických prvků poskytovaných systémem Android a několika ikon. Z uživatelského hlediska je vzhled důležitý a může pomoci při orientaci v aplikaci. Ostatní aplikace jsou s výjimkou Health Tracker profesionálně zpracovány s líbivou a funkční grafikou.

Největším přínosem je aplikace pro výzkumnou skupinu okolo EEG/ERP Portálu. Dosud se zde pracovalo pouze s měřením evokovaných potenciálů, nyní je možné získat a zkoumat také měření z dalších senzorů. Mobile Logger tedy nabízí další směr rozvoje Portálu a možností výzkumu.

9 Závěr

Cílem této práce bylo navrhnout a implementovat mobilní aplikaci pro platformu Android pro sběr dat ze zdravotních senzorů, tato data dovolit lokálně uložit a v případě dostupnosti internetu je odeslat do EEG/ERP Portálu.

Nejprve proběhlo seznámení s více než deseti volně dostupnými mobilními aplikacemi pro monitorování výkonnosti člověka. Tyto aplikace se většinou soustředí na monitorování dat pořízených senzory integrovanými v použitém mobilním zařízení. Z externích senzorů je podporován zpravidla jen monitor srdečního tepu. Výjimku tvoří aplikace eVito, která dokáže komunikovat s pěti druhy senzorů.

Dalším krokem bylo se fyzicky seznámit s poskytnutými senzory a najít způsob, jak jimi naměřená data získat. Mezi tyto senzory patří krokoměr, tlakoměr, hrudní pás pro měření srdečního tepu, tělesná váha a glukometr. Jedná se o zařízení podporované platformou eVito a její aplikací. Senzory komunikují prostřednictvím různých bezdrátových technologií, jimiž jsou ANT, ANT+ a Bluetooth. Úspěšně se podařilo implementovat komunikaci se čtyřmi z pěti senzorů. Výjimku tvoří krokoměr od společnosti eVito, jenž ke komunikaci využívá ANT a proprietární protokol.

Následně byla, na základě poznatků nabytých v předchozích krocích, navržena mobilní aplikace pro komunikaci se zkoumanými senzory. V případě průběžného měření srdečního tepu bylo měření implementováno ve službě na pozadí, aby mohlo být zaznamenáváno v průběhu činnosti. U ostatních senzorů jsou data vyžádána jednorázově, např. po měření. Aplikace data lokálně ukládá do SQLite databáze a umožňuje jejich základní manipulaci a analýzu.

Posledním úkolem bylo implementovat možnost v případě dostupnosti internetu odeslání lokálních dat a jejich uložení v EEG/ERP Portálu k vybranému experimentu. K tomu byl v mobilní aplikaci využit synchronizační framework systému Android, který automaticky hlídá dostupnost připojení a průběžně volá synchronizační logiku aplikace. Ke splnění tohoto cíle musela být upravena také datová a prezentační vrstva

portálu. Data jsou ukládána do nerelační databáze a bylo implementováno jejich prohlížení a mazání.

Pro další vývoj aplikace se nabízí několik témat. Je možné doimplementovat podporu pro další senzory, rozšířit možnosti analýzy dat a vylepšit grafické rozhraní aplikace. Dále je možné provést refaktoring projektu, kde doporučuji také přechod z automatizačního nástroje Maven na Gradle, který se více hodí pro flexibilní sestavování aplikací pro Android.

Reference

- [1] 2 Billion Consumers Worldwide to Get Smart(phones) by 2016 [online]. [cit. 11.12.2014] <<http://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694>>
- [2] JEŽEK, Petr. Database of EEG/ERP experiments. Plzeň, 2010. The State of the Art and the Concept of Ph.D. Thesis. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky.
- [3] RINKES, Jakub. Prezenční vrstva EEG/ERP Portálu. Plzeň, 2013. Bakalářská práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky.
- [4] ŠTĚBETÁK, Jan. Computational tools in EEG/ERP portal. Plzeň, Diploma thesis. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky.
- [5] EEGbase homepage [online]. [cit. 1.5.2015] <<https://eegdatabase.kiv.zcu.cz>>
- [6] BYDŽOVSKÝ, Martin. Relační a nerelační modelování pro portál elektrofyziologických experimentů. Plzeň, 2014. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky.
- [7] The Wireless Sensor Network Solution - THIS IS ANT [online]. [cit. 1.5.2015] <<http://www.thisisant.com/>>
- [8] ANT / ANT+ Defined [online]. [cit. 1.5.2015] <<http://www.thisisant.com/developer/ant-plus/ant-antplus-defined/>>
- [9] ANT directory [online]. [cit. 1.5.2015] <<http://www.thisisant.com/directory/>>
- [10] Transferring Data Using Sync Adapters [online]. [cit. 1.5.2015] <<https://developer.android.com/training/sync-adapters/index.html>>
- [11] The Low Energy Technology Behind Bluetooth Smart [online]. [cit. 1.5.2015] <<http://www.bluetooth.com/Pages/low-energy-tech-info.aspx>>
- [12] ANT+ / BT Smart Comparison [online]. [cit. 1.5.2015] <<http://www.thisisant.com/developer/ant-plus/ant-ble-comparison/>>

- [13] WANG, Yunfeng, Lei LI, Bo WANG a Lei WANG. A Body Sensor Network Platform for In-home Health Monitoring Application. In: Proceedings of the 4th International Conference on Ubiquitous Information Technologies & Applications [online]. 2009 [cit. 2015-05-07]. DOI: 10.1109/icut.2009.5405731.
- [14] eVito systém aktivního zdraví [online]. [cit. 1.5.2015]
<<http://www.evito.cz/PoznejEVito/SystemEVito.aspx>>
- [15] How self-monitoring is transforming health [online]. [cit. 1.5.2015]
<<http://www.medicalnewstoday.com/articles/264784.php>>
- [16] Měření EKG [online]. [cit. 1.5.2015]
<http://www.zelenalaborator.cz/files/Biologie/GF_metodika4_BIOLOGIE_EKG_FINAL.pdf>
- [17] Principles of Pulse Oximetry Technology [online]. [cit. 1.5.2015]
<<http://www.oximetry.org/pulseox/principles.htm>>
- [18] Designing a Heart Rate Monitor for Wearable Devices [online]. [cit. 1.5.2015]
<<http://www.mdtmag.com/articles/2014/09/designing-heart-rate-monitor-wearable-devices>>
- [19] KYLE, U. Bioelectrical impedance analysis?part I: review of principles and methods. Clinical Nutrition [online]. 2004, vol. 23, issue 5, s. 1226-1243 [cit. 2015-05-07]. DOI: 10.1016/j.clnu.2004.06.004.
- [20] Blood Sugar Level Ranges [online]. [cit. 1.1.2015]
<http://www.diabetes.co.uk/diabetes_care/blood-sugar-level-ranges.html>
- [21] COWIN helps Sensaris certify the ZAO biomedical sensor [online]. [cit. 1.5.2015]
<<http://www.cowin4u.eu/news/cowin-helps-sensaris-certify-the-zao-biomedical-sensor>>
- [22] FLASHLIGHT Sensor BT 12 [online]. [cit. 1.1.2015]
<<http://www.medset.com/ekg-verstaerker-bluetooth1.html?&L=5>>
- [23] Nonin 3230 Bluetooth SMART [online]. [cit. 1.1.2015]
<http://www.nonin.com/OEMsolutions/Nonin_3230_Bluetooth_SMART>
- [24] HRM Premium [online]. [cit. 1.5.2015]
<<http://www.garmin.cz/produkty/outdoor/prislusenstvi-outdoor/bezdratova-cidla/premiovvy-snimac-tepove-frekvence-hrm-premium.html>>
- [25] Mio ALPHA 2 Heart Rate Watch [online]. [cit. 1.1.2015]
<<https://www.mioglobal.com/Mio-ALPHA-2-Heart-Rate-Sport-Watch/Product.aspx?ProductID=28&DeptID=1>>

- [26] Blip BP [online]. [cit. 1.1.2015] <<http://www.blipcare.com/blip-bp.html>>
- [27] Adopted Bluetooth Profiles, Services, Protocols and Transports [online]. [cit. 1.1.2015] <<https://www.bluetooth.org/en-us/specification/adopted-specifications>>
- [28] IEEE 802.11™: Wireless LANs [online]. [cit. 1.5.2015] <<http://standards.ieee.org/about/get/802/802.11.html>>
- [29] Bluetooth High Speed Technology Overview [online]. [cit. 1.5.2015] <http://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=227418>
- [30] Adding Voice Capabilities [online]. [cit. 1.5.2015] <<https://developer.android.com/training/wearables/apps/voice.html>>
- [31] Endomondo [online]. [cit. 1.5.2015] <<https://www.endomondo.com/>>
- [32] Runtastic [online]. [cit. 1.5.2015] <<https://www.runtastic.com/>>
- [33] Pedometer [online]. [cit. 1.5.2015] <<https://www.runtastic.com/en/apps/pedometer>>
- [34] Push-Ups [online]. [cit. 1.5.2015] <<https://www.runtastic.com/en/apps/pushups>>
- [35] Heart Rate [online]. [cit. 1.5.2015] <<https://www.runtastic.com/en/apps/heartate>>
- [36] Sports Tracker [online]. [cit. 1.5.2015] <<http://www.sports-tracker.com/>>
- [37] Health Tracker [online]. [cit. 1.5.2015] <https://play.google.com/store/apps/details?id=com.benoved.phr_lite>
- [38] Madbarz [online]. [cit. 1.5.2015] <<http://madbarz.com/>>
- [39] What are Calisthenics Workouts [online]. [cit. 1.5.2015] <<http://calisthenicsdaily.com/what-are-calisthenics/>>
- [40] eVito [online]. [cit. 1.5.2015] <<http://www.evito.cz/>>
- [41] Typy eVito účtů [online]. [cit. 1.5.2015] <<http://www.evito.cz/PoznejEVito/SrovnaniUctu.aspx>>
- [42] Apache Wicket [online]. [cit. 1.5.2015] <<https://wicket.apache.org/>>
- [43] RESTful Web services: The basics [online]. [cit. 1.5.2015] <<https://www.ibm.com/developerworks/webservices/library/ws-restful/>>
- [44] eVito 3D krokoměr SL [online]. [cit. 1.5.2015] <<https://www.evito.cz/eshop/product/2>>
- [45] ForaCare p30 Plus BT [online]. [cit. 1.5.2015] <<https://www.evito.cz/eshop/product/62>>
- [46] Method for the measurement of the blood pressure [online]. [cit. 1.5.2015] <<http://www.blood-pressure-hypertension.com/how-to-measure/measure-blood-pressure-8.shtml>>

- [47] JSR 82: Java™ APIs for Bluetooth [online]. [cit. 1.5.2015]
<<https://jcp.org/en/jsr/detail?id=82>>
- [48] BlueCove [online]. [cit. 1.5.2015] <<http://bluecove.org/>>
- [49] Garmin hrudní pás premium [online]. [cit. 1.5.2015]
<<https://www.evito.cz/eshop/product/7>>
- [50] Android API [online]. [cit. 1.5.2015]
<<http://www.thisisant.com/developer/resources/downloads/>>
- [51] eVito váha SL [online]. [cit. 1.5.2015] <<https://www.evito.cz/eshop/product/3>>
- [52] Bioelectrical Impedance Analysis (BIA) [online]. [cit. 1.5.2015]
<<http://www.brianmac.co.uk/fatbia.htm>>
- [53] FORA Diamond MINI [online]. [cit. 1.5.2015]
<<https://www.evito.cz/eshop/product/31>>
- [54] YOO, Eun-Hyung a Soo-Youn LEE. Glucose Biosensors: An Overview of Use in Clinical Practice. Sensors [online]. 2010, vol. 10, issue 5, s. 4558-4576 [cit. 2015-05-08]. DOI: 10.3390/s100504558.
- [55] LACKO, Ľuboslav. Vývoj aplikací pro Android. 1. vyd. Brno: Computer Press, 2015, 472 s. ISBN 978-80-251-4347-6.
- [56] ALLEN, Grant. Android 4: průvodce programováním mobilních aplikací. 1. vyd. Brno: Computer Press, 2013, 656 s. ISBN 978-80-251-3782-6.
- [57] Miko, Petr. Mobilní systém pro správu EEG/ERP experimentů. Pilsen, 2013. Diplomová práce (Ing.). Západočeská univerzita v Plzni, Fakulta aplikovaných věd.
- [58] Command Pattern [online]. [cit. 1.5.2015] <<http://www.oodeesign.com/command-pattern.html>>
- [59] Apache Wicket User Guide [online]. [cit. 1.5.2015]
<<https://wicket.apache.org/start/userguide.html>>
- [60] MPAndroidChart [online]. [cit. 1.5.2015]
<<https://github.com/PhilJay/MPAndroidChart>>

Seznam obrázků

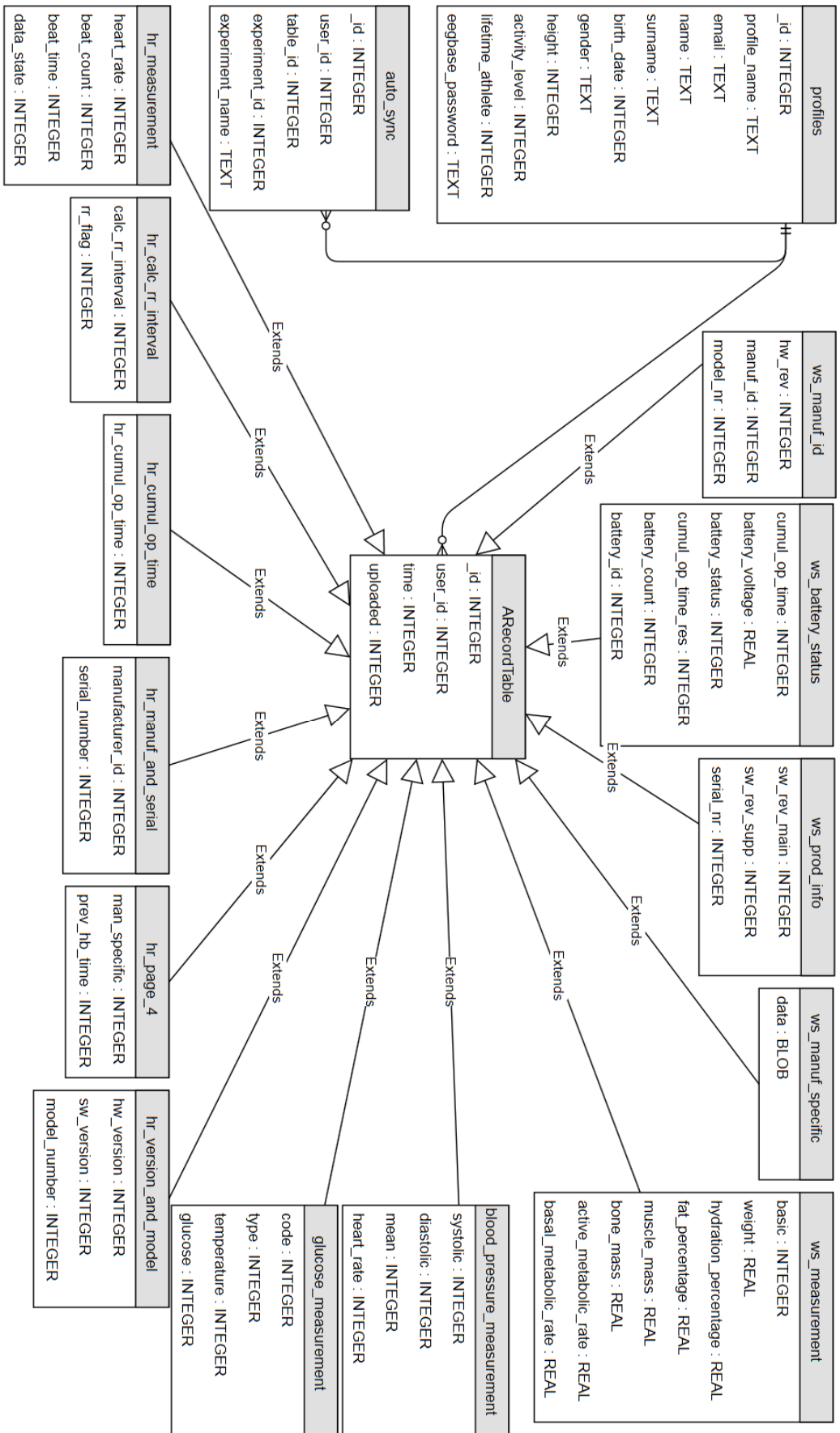
Obr. 1: Dvouvrstvá topologie.	7
Obr. 2: Třívrstvá topologie.	8
Obr. 3: Endomondo - měření.	12
Obr. 4: Endomondo - statistiky.	12
Obr. 5: Runtastic - měření.	13
Obr. 6: Runtastic - statistiky.	13
Obr. 7: Runtastic Pedometer - statistiky.	14
Obr. 8: Runtastic Heart Rate - měření.	14
Obr. 9: Sports Tracker - statistika.	16
Obr. 10: Health Tracker - statistika.	16
Obr. 11: Madbarz - výběr rutiny.	17
Obr. 12: Madbarz - průběh cvičení.	17
Obr. 13: eVito - měření.	18
Obr. 14: eVito - statistika.	18
Obr. 15: Hlavní aktivita.	37
Obr. 16: Tvorba profilu.	37
Obr. 17: UML - práce s profily.	38
Obr. 18: UML - práce se senzory.	39
Obr. 19: Zobrazení tlaku v aplikaci.	40
Obr. 20: Zobrazení tlaku tlakoměrem.	40
Obr. 21: Zobrazení srdečního tepu v aplikaci.	41
Obr. 22: Manipulace se záznamy krevního tlaku.	41
Obr. 23: Struktura projektu.	43
Obr. 24: Obsah balíku common.	44
Obr. 25: Obsah balíku data.	44
Obr. 26: Obsah balíku devices.	45
Obr. 27: Obsah balíku eegbase.	46

Obr. 28: Obsah balíku <code>service</code>	47
Obr. 29: Základní analýza krevního tlaku.	56
Obr. 30: Základní analýza srdečního tepu.	56
Obr. 31: Základní analýza glykémie.....	57
Obr. 32: Základní analýza váhy.....	57
Obr. 33: Struktura objektu <code>GeneralParameter</code>	59

Seznam tabulek

Tab. 1: Porovnání aplikací.....	67
---------------------------------	----

Příloha A: Obrázky



A. 1: ERA model databáze.



A. 2: Webové statistiky eVito.

Disease

Title	Description

Pharmaceutical

Title	Description

Project type

Title	Description
Test project	Type for test projects

Optional parameters

Parameter name	Value

Data files

File name	Description
idea.sh	Detail Delete


Generic parameters

Parameter name	Parameter value	Detail	Delete
software	ELFYZ_TST_SW	Detail	Delete
hardware	ELFYZ_TST_HW	Detail	Delete
weather	All weather	Detail	Delete
projectType	Test project	Detail	Delete
name1	value1	Detail	Delete
name2	value2	Detail	Delete
digitization	NotKnown	Detail	Delete
temperature	0	Detail	Delete
Blood pressure measurements	0	Detail	Delete

[Optional parameters](#)
[Add data file](#)
[Edit experiment](#)
[Add to Cart](#)

EEGbase - database for data gained in electrophysiology research.
 Copyright © [The University of West Bohemia](#) 2008-2015

A. 3: Výpis seznamu Generic Parameter na webu EEG/ERP Portálu.



EEGbase

Logged user: krupa@students.zcu.cz | [My account](#) | [My Cart: 0](#) | [Log out](#)

[Home](#) [Articles](#) [Search](#) [Experiments](#) [Scenarios](#) [Groups](#) [Lists](#) [History](#)

All experiments
My experiments
Me as subject
Search experiment
Add experiments

Generic parameter

Parameter name	Blood pressure measurements
String value	
Integer value	0
Attribute count	72

Parameter attributes

Name	Value
1388582580000	{"mean":81,"dia":70,"time":1388582580000,"heartRate":83,"sys":106}
1412945700000	{"mean":97,"dia":83,"time":1412945700000,"heartRate":69,"sys":127}
1412945760000	{"mean":94,"dia":87,"time":1412945760000,"heartRate":65,"sys":121}
1412945940000	{"mean":90,"dia":68,"time":1412945940000,"heartRate":66,"sys":119}
1414240740000	{"mean":84,"dia":70,"time":1414240740000,"heartRate":50,"sys":108}
1414240980000	{"mean":114,"dia":69,"time":1414240980000,"heartRate":52,"sys":117}
1414443780000	{"mean":86,"dia":70,"time":1414443780000,"heartRate":48,"sys":110}
1414445100000	{"mean":104,"dia":95,"time":1414445100000,"heartRate":57,"sys":124}
1414492800000	{"mean":80,"dia":65,"time":1414492800000,"heartRate":58,"sys":111}
1414493220000	{"mean":104,"dia":100,"time":1414493220000,"heartRate":70,"sys":133}
1414494900000	{"mean":69,"dia":59,"time":1414494900000,"heartRate":59,"sys":101}
1414495440000	{"mean":80,"dia":66,"time":1414495440000,"heartRate":57,"sys":104}
1414501500000	{"mean":76,"dia":69,"time":1414501500000,"heartRate":59,"sys":100}
1414501740000	{"mean":78,"dia":68,"time":1414501740000,"heartRate":57,"sys":103}
1414509600000	{"mean":75,"dia":60,"time":1414509600000,"heartRate":61,"sys":109}
1414510260000	{"mean":74,"dia":64,"time":1414510260000,"heartRate":66,"sys":105}
1414513020000	{"mean":71,"dia":59,"time":1414513020000,"heartRate":60,"sys":88}
1414513260000	{"mean":78,"dia":73,"time":1414513260000,"heartRate":59,"sys":110}
1414514160000	{"mean":84,"dia":66,"time":1414514160000,"heartRate":55,"sys":110}
1414516500000	{"mean":68,"dia":55,"time":1414516500000,"heartRate":60,"sys":97}
1414604040000	{"mean":82,"dia":66,"time":1414604040000,"heartRate":62,"sys":112}

A. 4: Detail parametru na webu EEG/ERP Portálu s měřením nahraným mobilní aplikací.