

Slovenská technická univerzita v Bratislave  
Fakulta informatiky a informačných technológií  
Ilkovičova 2, 842 16, Bratislava 4

---

# Manažment zdravotného stavu pacienta prostredníctvom monitoringu emócií [eMotion]

Dokumentácia k tímovému projektu časť 2  
(Inžinierske dielo)

Pedagogický vedúci tímu: Ing. Gašpar Peter

Externý vedúci tímu: Ing. Lehocký Fedor, PhD.

Členovia tímu: Bc. Bobotová Zuzana, Bc. Černák Dávid, Bc. Gondová Veronika, Bc. Matlovič  
Tomáš, Bc. Pavlovič Tomáš, Bc. Šmihla Ján

Akademický rok: 2016 / 2017

Verzia číslo: 2

Dátum poslednej zmeny: 11.12.2016

# Obsah

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Úvod</b>  | <b>1</b>  |
| 1.1      | Globálne ciele pre ZS . . . . .                                  | 1         |
| 1.2      | Globálne ciele pre LS . . . . .                                  | 2         |
| <b>2</b> | <b>Celkový pohľad na systém</b>                                  | <b>3</b>  |
| <b>3</b> | <b>Čítanie odborných článkov</b>                                 | <b>6</b>  |
| 3.1      | Analýza detekcie stresu pomocou biometrických senzorov . . . . . | 6         |
| 3.2      | Analýza prístupov k monitorovaniu stresu . . . . .               | 8         |
| 3.3      | Určovanie sentimentu z textu a hovorového slova . . . . .        | 14        |
| 3.4      | Prístupy využívajúce akcelometer . . . . .                       | 15        |
| 3.5      | Nálady vs. emócie vs. stres . . . . .                            | 17        |
| 3.6      | Zhrnutie . . . . .   | 21        |
| <b>4</b> | <b>Moduly systému</b>  | <b>23</b> |
| 4.1      | Mobil - zber dát . . . . .                                       | 23        |
| 4.1.1    | PO3 (Pulse oxygen sensor) . . . . .                              | 23        |
| 4.1.2    | AM3 (krokomer) . . . . .   | 23        |
| 4.1.3    | Bluetooth (zariadenia v okolí) . . . . .                         | 24        |
| 4.1.4    | Komunikačný modul . . . . .                                      | 25        |
| 4.1.5    | Autentifikácia používateľa . . . . .                             | 26        |
| 4.1.6    | Kalendárove udalosti . . . . .                                   | 27        |
| 4.1.7    | WiFi modul . . . . .   | 28        |
| 4.1.8    | GPS modul . . . . .  | 29        |
| 4.1.9    | SMS modul . . . . .  | 30        |
| 4.1.10   | Modul na detekciu hovorov . . . . .                              | 32        |
| 4.1.11   | Zber charakteristík SMS . . . . .                                | 33        |
| 4.1.12   | Mi Band 1S . . . . .   | 34        |
| 4.1.13   | Dotazníky aktivít . . . . .                                      | 35        |
| 4.1.14   | Zber nálad a emócií . . . . .                                    | 37        |
| 4.1.15   | Dashboardy s grafmi . . . . .                                    | 38        |
| 4.1.16   | Big5 dotazník . . . . .  | 39        |
| 4.1.17   | Rozpoznané aktivity . . . . .                                    | 40        |
| 4.2      | Web . . . . .  | 41        |
| 4.2.1    | Autentifikácia používateľa . . . . .                             | 41        |
| 4.2.2    | REST API na príjem dát . . . . .                                 | 42        |
| 4.2.3    | Dashboard s dátami používateľa . . . . .                         | 44        |
| 4.2.4    | Modul sentimentu . . . . .                                       | 45        |

|          |                                       |           |
|----------|---------------------------------------|-----------|
| 4.2.5    | REST API na odosielanie dát . . . . . | 46        |
| <b>A</b> | <b>Testovanie s používateľmi</b>      | <b>49</b> |
| A.1      | Prezentácia na ontožúre . . . . .     | 50        |
| <b>B</b> | <b>Protokol k experimentu</b>         | <b>51</b> |
| <b>C</b> | <b>Inštalčná príručka</b>             | <b>55</b> |
| <b>D</b> | <b>IIT.SRC</b>                        | <b>56</b> |
| <b>E</b> | <b>IIT.SRC poster</b>                 | <b>59</b> |

# 1 Úvod

Emócie, či už pozitívne alebo negatívne, majú rôzny vplyv na náš zdravotný stav. Najvýraznejšie na zdravie však vplýva stres ako počiatočnosť alebo naopak kombinácia rôznych emócií. Obzvlášť ak sa jedná o ľudí trpiacich chorobami ako je cukrovka alebo vysoký krvný tlak. Stres je v dnešnej dobe rozšírený natoľko, že ho ľudia prestávajú vnímať a uvažovať nad ním ako nad problémom. Aby bolo možné stres redukovať, je ho potrebné vedieť správne monitorovať. Cieľom tohto projektu je vytvoriť systém, ktorý umožní zber dát potrebných na identifikáciu stresu prostredníctvom emócií, vyhodnocovať ich a zobrazovať používateľom spätnú väzbu.

Tento dokument obsahuje informácie o finálnom stave projektu po dvoch semestroch. Hlavným cieľom tohto dokumentu je oboznámenie čitateľa s technickou stránkou projektu. Dokument obsahuje podrobný opis globálnych cieľov na zimný a letný semester a celkový pohľad na systém zahŕňajúci opis architektúry systému ako aj jednotlivé moduly systému.

Moduly systému sa delia do dvoch častí - na moduly mobilnej aplikácie a moduly webovej aplikácie. Opis každého jedného modulu opisuje analýzu, návrh, implementáciu a testovanie daného modulu. V analýze sa zameriavame na preštudovanú literatúru a knižnice. V návrhu sú uvedené zvážené možnosti riešenia a v implementácii využité technológie. V testovaní je opísaný postup testovania konkrétneho modulu.

## 1.1 Globálne ciele pre ZS

Aby dokázal náš model na analýzu emócií správne pracovať, potrebujeme dostatočné množstvo dát. Preto bolo našim hlavným cieľom v zimnom semestri vytvoriť časti systému, ktoré umožnili zber dát na mobilných zariadeniach používateľov a následné odosielanie na server. K tomu bolo potrebné vytvoriť mobilnú aplikáciu, ktorá dokáže pracovať s rôznymi zdrojmi dát v systéme Android a taktiež externými senzormi (Oximeter, Krokometer). V mobilnej aplikácii sa nazbierané dáta zobrazujú a zároveň sa odosielajú na server.

Aplikácia na strane servera musí byť schopná spravovať používateľské účty, vytvárať nové účty, prijímať dáta z mobilnej aplikácie a efektívne ich uložiť, analyzovať a odosielať agregované dáta do mobilnej aplikácie. Prihlásený používateľ si môže zobrazit dáta, ktoré odoslal jeho mobilný telefón.

## 1.2 Globálne ciele pre LS

Po zimnom semestri je naša aplikácia v stave, v ktorom je schopná vykonávať všetko potrebné pre zbieranie dát z mobilných zariadení, preto je našim hlavným cieľom pre letný semester príprava na experiment s ľuďmi a jeho plánovanie. Tento experiment je pre budúcnosť projektu veľmi dôležitý, pretože nám poskytne množstvo dát, ktoré použijeme na natrénovanie modelu strojového učenia. Je tiež dôležité sa poradiť s psychológmi o rôznych parametroch tohto experimentu.

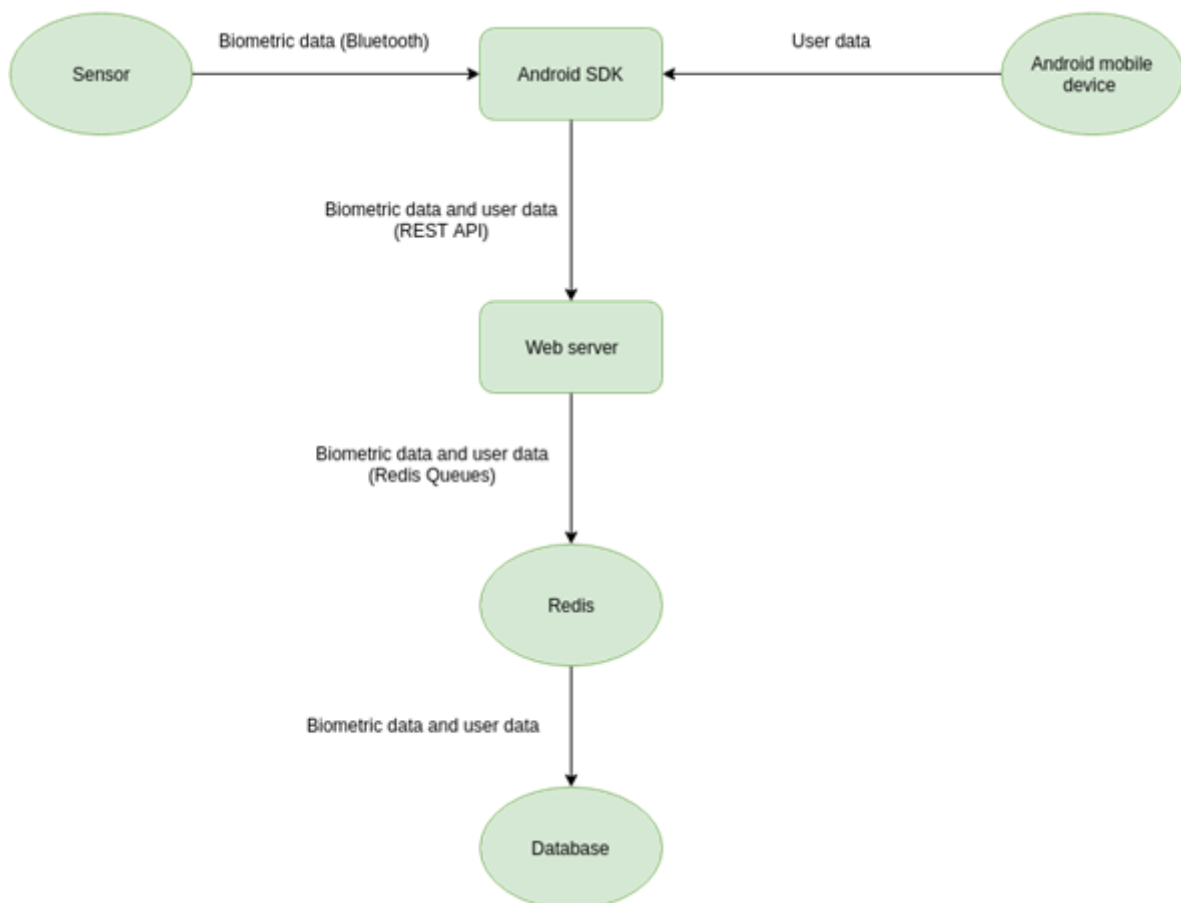
Ďalej bude potrebné zabezpečiť senzory na tento experiment (ak budú potrebné). Ak sa rozhodneme použiť iné senzory ako tie, ktoré máme aktuálne implementované v mobilnej aplikácii, tak bude potrebné do aplikácie implementovať podporu pre tieto senzory. Následne bude potrebné zozbierať spätnú väzbu od účastníkov experimentu a ak to bude možné, aj začať s analýzou zozbieraných dát.

## 2 Celkový pohľad na systém

Náš systém sa skladá z dvoch hlavných častí: mobilnej aplikácie a webovej aplikácie. Mobilná aplikácia je vyvíjaná na platforme Android a webovú aplikáciu tvoríme s použitím Python webového frameworku - Django.

Na obrázku č. 1 môžeme vidieť diagram toku dát. Dáta z biometrických senzorov sú odosielané do mobilnej aplikácie cez Bluetooth. Následne sú odosielané na webový server pomocou nami vytvoreného REST API (ak mobilné zariadenie nemá v danom čase prístup na internet, dáta sú dočasne uložené v zariadení). Spolu s dátami zo senzorov posielame aj dáta o používateľovi a jeho zariadení - kalendár, sms, gps, dostupné wifi pripojenia, viditeľné Bluetooth zariadenia, denník hovorov.

Z dôvodu zachovania čistoty dát sme implementovali na strane servera validáciu dát pomocou

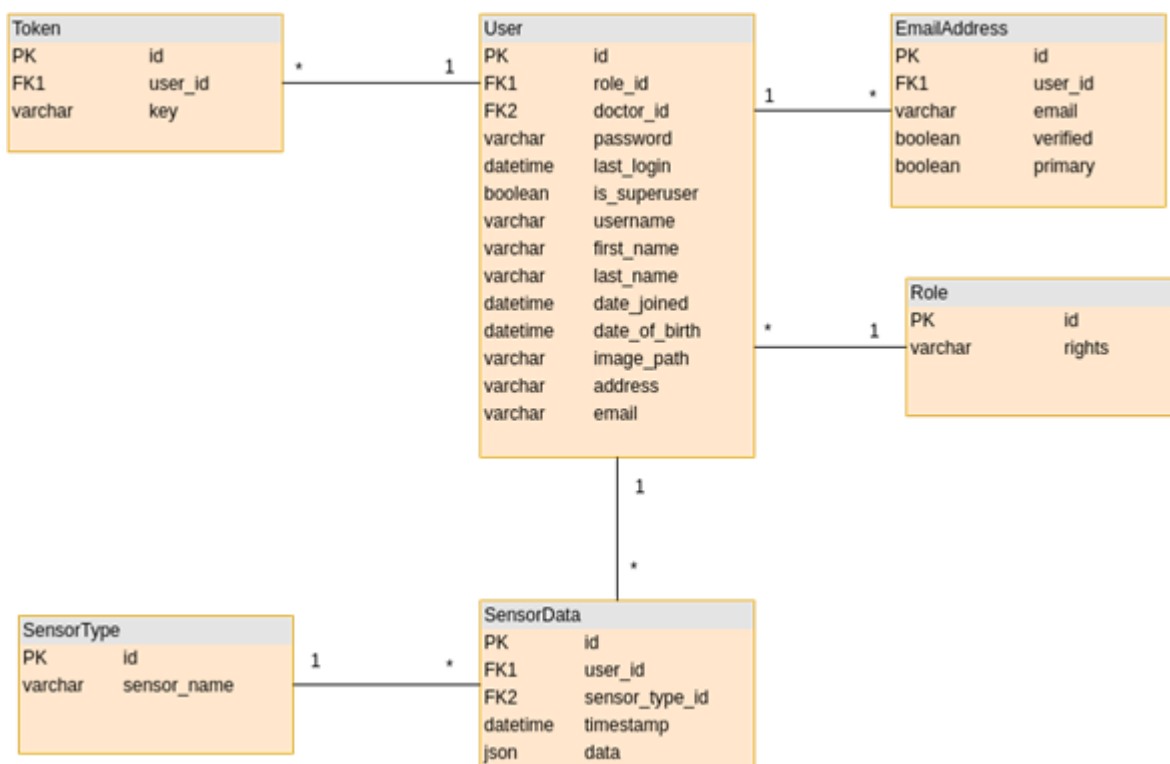


Obr. 1: Diagram toku dát

JSON schém. Dáta z jednotlivých senzorov a dáta z mobilného zariadenia musia mať predpísanú štruktúru, inak nebudú ďalej spracované.

Keďže biometrické dáta su zaznamenávané s relatívne veľkou frekvenciou, rozhodli sme sa použiť tzv. Redis queues. Redis dokáže vďaka využitiu pamäte RAM ukladať dáta s veľkou rýchlosťou. V prvom rade teda dáta uložíme do REDISU. Následne sú dáta postupne vyberané z Redisu a ukladané do relačnej databázy PostgreSQL. Použitím takejto architektúry budeme schopní spracovávať dáta aj keby ich prišlo veľké množstvo naraz.

Dáta sú na serveri spracované. V určitých intervaloch sa z nich počítajú pre každého používateľa metriky ako napríklad elektronická socializácia alebo miera fyzickej aktivity. Výsledky týchto výpočtov sú odoslané do mobilnej aplikácie pomocou REST API, keď príde príslušná požiadavka. Okrem týchto metrických posielame do mobilnej aplikácie aj agregované údaje o emóciách a náladách používateľa.



Obr. 2: Dátový model

Na obrázku č. 2 je zobrazený použitý dátový model. V tabuľke User sa nachádzajú používatelia našej aplikácie. Každý používateľ má pridelenú rolu a môže mať viac emailových adries a tokenov. Biometrické dáta a dáta o používateli z jeho mobilného zariadenia sa nachádzajú v tabuľke SensorData. Táto tabuľka je následne naviazaná na tabuľku SensorType a teda v dátach sa nachádza informácia o tom z akého senzoru sú nazbierané. Typ senzoru môže byť napr. Oximeter, krokomer alebo aj sms alebo gps. Vďaka tomu, že dáta sú vo formáte json, môžeme ich všetky uložiť do jednej tabuľky a znížiť tak komplexitu nášho modelu. V tabuľke Socialization sa nachádzajú výpočty elektronickej socializácie pre každého používateľa aj s údajom o tom v ktorom čase bol tento výpočet uskutočnený. V tabuľke ChosenQuestionnaireActivities sú

uložené ID aktivít, ktoré používateľ zvolil v dotazníku pri prvom spustení aplikácie. Taktiež sa tu nachádzajú aktivity, ktoré používateľ v dotazníku sám dopísal.

Čo sa týka infraštruktúry, webová aplikácia je nasadená na serveri s operačným systémom Linux. Nachádza sa tam aj Redis spolu s relačnou databázou PostgreSQL. Pri testovaní a nasadzovaní novej verzie aplikácie používame kontinuálnu integráciu, a teda všetko je plne automatizované. Využívame na to službu Travis CI. Testovanie vykonáva priamo Travis, akonáhle deteguje zmeny v git repozitári. Oznámenie o výsledku testovania sa nám odosielať do Slacku, kde na to máme vyhradený kanál. Ak išlo o zmenu v hlavnej vetve master tak Travis navyše pošle požiadavku na náš server kde následne prebehne nasadenie novej verzie. V prvom rade sa stiahne aktuálna verzia z git repozitára a následne sa reštartne webový server aby sa prejavili zmeny.

Pri testovaní posielania dát z mobilnej aplikácie na server sme mali problém s Redisom, ktorý často prestal pracovať a bolo ho vždy potrebné manuálne reštartovať. Navrhli a implementovali sme preto program ktorý monitoruje proces Redisu. Ak je tento proces z nejakého dôvodu ukončený, náš program ho reštartne.



## 3 Čítanie odborných článkov

### 3.1 Analýza detekcie stresu pomocou biometrických senzorov

**W.D. Scherz, J. Ortega, R. Seepold, "Towards emotion pattern extraction with the help of stress detection techniques in order to enable a healthy life", ARCA XXVII Conference on Qualitative Systems and Applications in Diagnosis, Robotics and Ambient Intelligence, ISBN: 978-84-608-5599-6, 2015.**

V tomto článku je opísaný návrh metódy na rozpoznávanie rôznych vzorov stresu za účelom klasifikácie emócií. Využívali pri tom nízkonákladové EKG - elektrokardiograf (z angl. ECG - electrocardiograph). Toto EKG je neinvazívne, zaznamenáva dáta v reálnom čase a má tri elektródy, ktoré zaznamenávajú tep srdca. Z neho vypočítali tzv. RR intervaly, čo sú zjednodušene povedané intervaly medzi jednotlivými údermi srdca. Z týchto intervalov následne určujú variability tepu srdca a z nej následne zistujú hladinu stresu. Potenciál navrhutej metódy overili experimentom. Ukázalo sa, že použitím tejto metódy je možné odhaliť stres v rôznych podmienkach, pri vysokom alebo nízkom tepe srdca.

Dáta z EKG spracúvali pomocou mikročipu Arduino UNO. Priamo na mikročipe bežal aj samotný algoritmus na meranie stresu. Dáta bolo následne možné ukladať na SD kartu alebo posielat' cez bluetooth napr. do smartfónu.

**W.D. Scherz, J. Ortega, N. Martinez Madrid, R. Seepold, "Heart Rate Variability indicating Stress visualized by Correlations Plots", Lecture Notes in Bioinformatics and Biomedical Engineering (LNBI), Volume 9044, Subseries of Lecture Notes in Computer Science, Springer International Publishing, ISSN 0302-9743, 2015.**

V tomto článku sa autori rozhodli vyvinúť nejaký ľahko nositeľný a lacný prístroj, ktorý by vedel spolupracovať s mobilom. Sústredili sa na ECG - aby ukázali, že ako veľmi to má vplyv pri strese. Vytvorili systém tak, aby sa tam dali pridať aj iné senzory.

Systém pozostáva z viacerých častí: ECG senzor a mikroprocesor, ktorý spracúva dáta a zariadenie na zobrazenie dát pre spätnú väzbu. ECG senzor funguje tak, že má tri elektródy, ktoré vysielajú analógový signál mikroprocesoru, ktorý ho transformuje na digitálny a vykoná pre-processing a filtráciu. Následne sa z digitálnych dát vypočíta HR a RR interval (interval medzi peakmi). Stress detection metóda je založená na ECG signále, z HR a RR intervalu vyjadria HRV

a to použijú na určenie stresu. Na evaluáciu využili metódu, pre ktorú je dokázane, že zvyšuje stres. Zistili teda, že stres môže byť zistený na základe meniacej sa hodnoty z ECG.

**P. Datko, J. Martínez Fernández, R. Seepold, "Aggregating individual stress-levels for dynamic clusters", AITA – Workshop on Ambient Intelligence for Telemedicine and Automotive domains, ISBN 978-84-697-0147-8, 2014.**

V tomto článku sa autori zamerali na sledovanie správania vodičov vozidiel. Dospeli k záveru, že stres negatívne ovplyvňuje správanie jednotlivca a tým zvyšuje riziká ku vzniku dopravných nehôd.

Úrovně stresu v tomto kontexte sú založené na externých a interných faktoroch. Do externých faktorov patrí všetko to, čomu vodič čelí pred jazdou, či už aktívne alebo pasívne (nezamestnanosť, finančné problémy). Do vnútorných faktorov patria všetky aktívne úkony počas jazdy, či už časový tlak, sledovanie GPS zariadenia, alebo používania mobilu, ale aj správanie ostatných vodičov (napríklad niekto nedá prednosť v jazde).

V tejto práci vyvinuli systém, ktorý pomocou zhukovania rozdeľuje vodičov do skupín (pohodlná jazda na diaľnici, rušná jazda v meste,...), monitoruje a analyzuje individuálne úrovne stresu vodičov a šíri relevantné informácie v rámci skupiny pre zvýšenie bezpečnosti. Merajú polohu vodičov a aj srdcový tep a úroveň stresu z externého senzora a posielajú na server pomocou technológií Global System for Mobile Communications (GSM), Universal Mobile Telecommunications System (UMTS) or Long Term Evolution (LTE). Na serveri sa to spracuje a vyhodnotí. V prípade zistenia nejakého anomálneho správania sa pošle danej skupine vodičov upozornenie.

**W.D. Scherz, J. Martínez Fernández, R. Seepold, "Monitoring driving behaviour and biometric data to detect stress patterns", AITA – Workshop on Ambient Intelligence for Telemedicine and Automotive domains, ISBN 978-84-697-0147-8, 2014.**

V tomto článku sa autori zamerali na meranie stresu vodičov motorových vozidiel na základe ich správania. Cieľom práce je poskytnúť vodičovi informáciu o jeho biologickom stave a tým zlepšiť kvalitu jazdy. Použili dve merania. Najskôr merajú stres vodiča pomocou dostupných senzorov (emWave) a pri tom paralelne monitorujú aj informácie o vozidle. A potom pre zvolenie relevantných dát, potom používajú e-Health sensory (SP02, Airflow, Body temperature, ECG, Glucometer, GSR, sphygmomanometer, accelerometer, EMG), pripojený na Arduino. Namerané dáta integrujú do dát extrahované z openDS simulátora (rýchlosť, otáčky, stupeň prevodu). Porovnanie oboch meraní pomôže zlepšiť kvalitu vyhodnocovania a tým pádom pravdepodobne zmenšiť šírku parametrov nameraných pomocou senzorov.

**W.D. Scherz, R. Seepold, "Detection of heart rate characteristics as an approach to distinguish between mental stress and physical activity", 12th Russian-German Conference on biomedical engineering (RGC), 2016.**

Autori článku sa snažia monitorovať stres pomocou ECK, EKG. Ale tým že aj pri fyzických aktivitách môže systém vyhodnotiť, že je človek v strese, tak sa snažia v tomto výskume nájsť správne charakteristiky, aby vedeli rozlíšiť či sa jedna o psychickú aktivitu alebo stres.

Robili experiment, v ktorom merali krvný tlak a srdcový tep participantov. Experiment bol zložený z troch fáz, v prvej fáze účastníci relaxovali, v druhej fáze vykonávali fyzické aktivity a v tretej fáze boli v stresovej situácii. Prvá fáza trvala 15 minút, v druhej fáze vykonávali účastníci rôzne cvičenia, na stacionárnom bicykli, každé z nich trvalo 5 minút. V poslednej fáze vykonali na účastníkov Stropov test, TSST a riešiť rôzne matematické úlohy v obmedzenom čase. Účastníci boli študenti vo veku od 18 do 40 rokov. Nikto z nich nebol vrcholový športovec alebo fajčiar.

Metóda rozlišovania fyzickej aktivity od stresu je založená na dvoch princípoch, prvý princíp je detekcia rýchlej zmeny frekvencie srdcového tepu, predpokladá sa, že tieto zmeny závisia od intenzity fyzickej aktivity. Druhá možnosť je založená na Furierovej analýze spektra, kde sa bude analyzovať správanie vysokej a nízkej frekvencie. Na to bude treba ale vykonať viac experimentov s väčším množstvom účastníkov.

## **3.2 Analýza prístupov k monitorovaniu stresu**

**WU, Wanqing, et al. Assessment of Biofeedback Training for Emotion Management Through Wearable Textile Physiological Monitoring System. IEEE Sensors Journal, 2015, 15.12: 7087-7095.**

Práca sa zaoberá meraním stresu prostredníctvom nositeľných senzorov a zaznamenávaním srdcovej frekvencie (HRV), ktoré môžu slúžiť na včasné varovanie pred stresom ako aj vizualizáciu emočného stavu používateľ'a. Výsledky štúdie ukazujú, že HRV je jeden zo základných prostriedkov na meranie stresu a úspešnosť odhadovania stresu pomocou srdcovej frekvencie je vysoká.

Práca sa taktiež zaoberala možnosťou rozlíšenia pozitívneho resp. negatívneho stresu prostredníctvom tzv. hladkých alebo nepravidelných vzorov v srdcovej frekvencii. Experimenty ukazujú, že negatívny stres je spravidla spojený s tzv. nepravidelnými vzormi v srdcovej frek-

vencii. Táto práca sa taktiež zaoberala vplyvom stresu na dýchanie človeka a korelácii medzi dýchaním a srdcovou frekvenciou pacienta.

**PAPAGEORGIU, Achilleas; ZIGOMITROS, Athanasios; PATSAKIS, Constantinos. Personalising and crowdsourcing stress management in urban environments via s-Health. In: Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on. IEEE, 2015. p. 1-4.**

Táto práca sa zaoberá tematikou tzv. inteligentných miest. Jej cieľom je využiť možnosti, ktoré inteligentné mestá ponúkajú na zvýšenie zdravotnej starostlivosti v podobe monitoringu stresu obyvateľov a následného zásahu na hladinu stresu prostredníctvom pokročilých techník IKT. Na určovanie hladiny stresu v práci využívajú senzory, ktoré inteligentné mestá poskytujú, v rástane mobilov, ktoré využívajú samotní obyvatelia. Ich cieľom je zníženie stresu obyvateľov. Zdrojom stresu v meste môžu byť napr. preplnené ulice, zapchatá cesta alebo neprispôsobiví vodiči. Medzi možnosti odbúrania stresu, ktoré boli v práci prezentované, patrí zmena trasy cesty, prehratie obľúbenej hudby alebo nevyhnutná socializácia.

**AL OSMAN, Hussein; DONG, Haiwei; EL SADDIK, Abdulmotaleb. Ubiquitous Biofeedback Serious Game for Stress Management. IEEE Access, 2016, 4: 1274-1286.**

Táto práca je venovaná spôsobu odbúravania stresu prostredníctvom gamifikácie. Používatelia navrhutej aplikácie odbúravajú stres ovládaním hry na základe ich biometrických dát zo senzorov. To znamená, že pokiaľ má človek lepšie výsledky zdravotného stavu zo senzorov, o to lepšie skóre nadobúda v hre. Experimenty ukazujú, že ľudia, ktorí využívali prostriedky navrhutej aplikácie, dokázali lepšie ovládať svoj duševný stav a rýchlejšie sa vyrovnávali so vzniknutým stresom.

**CHEN, Kemeng, et al. Wearable sensor based stress management using integrated respiratory and ECG waveforms. In: 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN). IEEE, 2015. p. 1-6.**

Práca sa zaoberá monitoringom stresu pacientov prostredníctvom nositeľných senzorov a mobilných zariadení a následnej spätnej väzbe v podobe dychových cvičení za účelom zníženia hladiny stresu pacienta. Na odhadovanie hladiny stresu pacienta sú využívané dve merania - variabilita srdcovej frekvencie a cyklus aktivity pacienta (prostredníctvom frekvencie dýchania pacienta). Z výsledkov vyplýva, že dychové cvičenia účinne znižujú úroveň stresu pacientov.

**ROMLI, Awanis; CHA, Arnidcha Peri. An expert system for stress management. In: Internet Technology and Secured Transactions, 2009. ICITST 2009. International Conference**

**for. IEEE, 2009. p. 1-6.**

Táto práca je venovaná expertným systémom, ktorých cieľom je určiť spôsob, ktorým jednotlivec dokáže najlepšie zvládať stres. Táto analýza je vykonaná na základe dvoch testov - záťažovým testom a testom osobnosti. Jej cieľom je vyvinúť efektívne odporúčanie aktivít používateľov, ktoré na základe osobnostných črt a na základe ich schopnosti zvládania stresu, vyhodnotí najvhodnejšiu aktivitu pre zníženie úrovni stresu. Výstupy tejto práce pomáhajú psychológom a expertom z danej oblasti rýchlejšie vyhodnotiť úroveň stresu a následnú liečbu pacientov.

**EID, Mohamad; AL OSMAN, Hussein; EL SADDIK, Abdulmotaleb. A mathematical model for personalized relaxation for stress management. In: Medical Measurements and Applications Proceedings (MeMeA), 2013 IEEE International Symposium on. IEEE, 2013. p. 201-206.**

Práca sa zaoberá vytvorením personalizovaného modelu, ktorý odporúča relaxačné cvičenia pre lepšie zvládanie stresu. Stres je monitorovaný na základe biometrických charakteristík pacientov. Cieľom tejto práce je naučiť používateľa, čo je pre neho najlepšie a ako dokáže najefektívnejšie zvládať stres. Táto práca bola overená prostredníctvom prípadových štúdií, ktoré preukázali úspešnosť matematického modelu. Navrhnutý matematický model je schopný odporúčať vhodné relaxačné cvičenia a je flexibilný na zmenu prostredia.

**CARBONARO, Nicola, et al. Wearable biomonitoring system for stress management: A preliminary study on robust ECG signal processing. In: World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a. IEEE, 2011. p. 1-6.**

Táto práca je zameraná na návrh a vývoj pokročilých simulačných a snímacích technológií pre odhalovanie stresu na základe biosenzorov. Na vyhodnotenie úrovne stresu štúdia využíva EKG signál, na základe ktorého určuje srdcovú frekvenciu a variabilitu srdcovej frekvencie. V práci bol navrhnutý robustný systém, ktorý dokáže odhalovať stres pacientov a je odolný voči znečisteniu EKG signálu, prahovým hodnotám a zároveň poskytuje riešenie pre prostriedok na odhalovanie stresu s nízkou výpočtovou náročnosťou na spracovanie dát v reálnom čase.

**BAUER, Gerald; LUKOWICZ, Paul. Can smartphones detect stress-related changes in the behaviour of individuals?. In: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on. IEEE, 2012. p. 423-426.**

Táto práca sa zaoberá odhalovaním stresu prostredníctvom mobilného zariadenia a prostriedkov,

ktoré mobilné zariadenie ponúka. Cieľom tejto štúdie je odhaliť, či je možné detegovať stres výlučne na základe bežného využívania mobilného telefónu. Štúdia vychádza z predpokladu, že človek sa správa inak pokiaľ je dlhodobo v strese a pokiaľ v strese nie je.

Súčasťou štúdie bol experiment, ktorý bol realizovaný na 7 študentoch počas skúškového obdobia. Počas tohto obdobia študenti využívali mobilné telefóny, z ktorých sa v pravidelných časových intervaloch odosieli dáta na server, ktoré následne slúžili na vyhodnotenie stresu študentov. Experiment potvrdil predpoklad o tom, že človek sa správa inak pokiaľ je vystavený dlhodobému stresu - navštevuje iné miesta, obmedzuje kontakt s ľuďmi a kontaktuje iba vyhradenú skupinu ľudí. V rámci tejto štúdie boli využité nasledovné typy dát:

- *Geografického správania sa používateľa* - GPS a wifi
- *Sociálna interakcia* - Bluetooth
- *Hovory* - čísla, čas, dĺžka
- *SMS* - čísla, čas, dĺžka

**MADHURI, V. J.; MOHAN, Madhumitha R.; KAAVYA, R. Stress Management Using Artificial Intelligence. In: Advances in Computing and Communications (ICACC), 2013 Third International Conference on. IEEE, 2013. p. 54-57.**

Stres ako taký je jedným z hlavných faktorov, ktoré vedú k zdravotným problémom. V tejto práci sa zaoberajú návrhom systému na určovanie stresu z rôznych fyziologických dát - tep srdca, vodivosť pokožky, teplota tela, napätie svalov a krvný tlak. Rôzni ľudia reagujú na stresové udalosti rôznym spôsobom. Na určovanie stresu použili Fuzzy logiku, ktorá by mala zvládnuť aj takéto rozdiely. Týmto spôsobom mali taktiež možnosť určovať aj samotnú hladinu stresu a nie len či človek bol alebo nebol v strese. Po otestovaní navrhutej metódy zistili, že prekonal doposiaľ používané prístupy ako SVM, NN a LDA.

**CERUTTI, Sergio; BIANCHI, Anna M.; REITER, Harald. Analysis of sleep and stress profiles from biomedical signal processing in wearable devices. Proceedings IEEE engineering in medicine and biology society (EMBC), 2006.**

V práci je opísaná metóda na meranie kvality spánku a manažment stresu na základe analýzy variability tepu srdca a dýchania. Analýza je zameraná najmä na dáta z prístrojov, ktoré slúžia na prevenciu proti kardiovaskulárnym chorobám a na celkové zlepšenie zdravia. Veľkou výhodou variability tepu srdca je, že vďaka nej je možné cenné informácie ohľadom stresu a kvality spánku bez potreby klinického zásahu. V práci bolo opísané ako rôzne úrovne stresu vplývajú na zdravie človeka a kvalitu jeho spánku.

**MAYYA, Subramanya, et al. Continuous monitoring of stress on smartphone using heart rate variability. In: Bioinformatics and Bioengineering (BIBE), 2015 IEEE 15th International Conference on. IEEE, 2015. p. 1-5.**

Práca popisuje metódu na meranie hladiny stresu pomocou merania tepu srdca. Metóda pozostáva zo získavania dát, predspracovania dát, vyhodnotenia a nakoniec sa modeluje skóre - hodnota stresu. Pri získavaní dát autori zvolili taký postup, že najskôr počítali hladinu stresu, keď bol človek v pokoji, a potom mu kázali vykonávať nejaké stresujúce aktivity, aby vedeli určiť zmeny. Medzi takéto aktivity patrí Stroop Colour Word Test, Mental Arithmetic Task (MAT), Memory task, Public Speaking task and Backward Counting task. Získané dáta bolo potrebné predspracovať, čo znamenalo odstránenie náhodných - chybových hodnôt. Následne boli dáta spracované, čo zahŕňalo extrakciu príznakov a štatistické analýzy príznakov. Výsledkom bola nameraná výška hladiny stresu, ktorú pekne graficky na mobilnom zariadení vizualizovali.

**GARCIA-MANCILLA, Jesus; GONZALEZ, Victor M. Stress Quantification Using a Wearable Device for Daily Feedback to Improve Stress Management. In: International Conference on Smart Health. Springer International Publishing, 2015. p. 204-209.**

Práca sa venuje novo navrhutej metóde na kvantifikáciu stresu u človeka. Autori vychádzajú z toho, že keď je človek v strese, tak na to nervový systém nejakým spôsobom zareaguje. Na zistenie zmien využili nositeľné senzory, ktoré vedia merať tep srdca a teplotu kože. Namerané hodnoty potom použili v nimi navrhnutom vzorci, ktorý dokázal podľa týchto hodnôt určiť hladinu stresu. V článku bola zdôraznená využiteľnosť merania stresu pre medicínske účely, čo znamená sledovanie stresu pacienta, čo by malo dopomôcť k zníženiu stresu a teda zlepšeniu životného štýlu a procesu liečenia.

**UDOCHUKWU, Orizu; HE, Yulan. A Rule-Based Approach to Implicit Emotion Detection in Text. In: International Conference on Applications of Natural Language to Information Systems. Springer International Publishing, 2015. p. 197-203.**

Článok hovorí o rozoznávaní emócií prostredníctvom textu. Autori uvádzajú, že pre určovanie emócií z textu využili známy OCC model. Ten pozostáva zo signifikantných slov pre dané emócie a určuje silu, akú dávajú danej emóci. Ide o rešpektovaný a uznávaný model pre určovanie emócií. V článku sa uvádza, že emócie sa zväčša rozpoznávajú explicitne (napr. veta "Som šťastný, že som zvládol ten test"). Taktiež je uvedené, že implicitné rozpoznávanie emócií je ťažšie (napr. veta "Spravil som ten test."). Autori vo svojom výskume na základe kľúčových slov, použitého času a pozitívnosti zvyšku vety detegovali emócie. Príklad: If Direction = "Self" and Tense = "Future" and Overall Polarity = "Positive" and Event Polarity = "Positive", then

Emotion = "Hope".

**SUN, Yan, et al. Customer emotion detection by emotion expression analysis on adverbs. Information Technology and Management, 2015, 16.4: 303-311.**

Práca hovorí o určovaní emócií pomocou prísloviiek. Autori zhodnotili, že reč je silný nástroj na zisťovanie emócií a posúdili, že málo prác v oblasti rozpoznávania emócií sa zaoberá práve príslovkami. Ľudia pri rozhovoroch často používajú príslovky, ale nie všetky dávajú nejaké podfarbenie pre určenie pocitu. Preto ich treba extrahovať - vybrať tie najvhodnejšie (podobne ani všetky vety nedajú jasne najavo pocity). Počas práce si autori museli vyrobiť vlastný dataset, čo znamenalo rozdeliť príslovky, aby to mohli preskúmať.

**ZENONOS, Alexandros, et al. HealthyOffice: Mood recognition at work using smartphones and wearable sensors. In: 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops). IEEE, 2016. p. 1-6.**

Cieľom práce bolo znížiť stres na pracovisku pomocou senzorov, ktoré merali dáta ako EKG, teplotu kože, akceleráciu. V tejto práci bola vyvinutá aplikácia, ktorej cieľom je zberať informácie o nálađe od užívateľa. Aplikácia vyžadovala od požívateľa spätnú väzbu každé dve hodiny, pričom sa užívateľ vyjadril k 8 náladám pomocou škály od 0 do 100.

- 0 - 19 žiadny náznak emócie
- 20 - 39 jemný náznak emócie
- 40 - 59 stredný náznak emócie
- 60 - 79 väčší náznak emócie
- 80 - 100 extrémny náznak emócie

V práci predstavili svoju víziu, a dospeli k sľubným výsledkom, avšak, ako sami opisujú pracovali s malým počtom ľudí a nazbierali málo dát. Najľahšie sa detegovala emócia hnevu, za ňou nasledovali smútok, šťastie, stres, únava, nuda a pokoj. Pomocou senzorov sa podarilo dosiahnuť úspešnosť približne 70%.

**SANDULESCU, Virginia, et al. Mobile app for stress monitoring using voice features. In: E-Health and Bioengineering Conference (EHB), 2015. IEEE, 2015. p. 1-4.**

Cieľom tejto práce je detegovať stres pomocou hlasu. Za týmto účelom bola vyvinutá aplikácia StressID. Aplikácia sa spúšťa na pozadí určité hodiny v určitý čas, pričom pomocou knižnice TarsosDSP sa snaží nahráť 10 sekundovú audio stopu. Táto nahrávka sa následne



rozdělí po 20 milisekund. Jednotlivé část' sa spracujú pomocou knižnice libSVM. Výsledok nahrávky sa následne odošle na server. V práci dosiahli úspešné výsledky s presnosťou 78% v určovaní pozitívnych alebo negatívnych nahrávok. Taktiež uvádzajú, že hlas je ukladaný iba dočasne - kým sa nespracuje, následne je vymazaný.

**FERDOUS, Raihana; OSMANI, Venet; MAYORA, Oscar. Smartphone app usage as a predictor of perceived stress levels at workplace. In: Proceedings of the 9th International Conference on Pervasive Computing Technologies for Healthcare. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2015. p. 225-228.**

Cieľom práce bola detekcia stresu na pracovisku pomocou detegovania, ako často užívatelia používajú konkrétne aplikácie. V experimente dali 28 pracujúcim ľuďom na 6 týždňov mobilný telefón, ktorý zaznamenával činnosti všetkých aplikácií na mobile - začiatok, koniec, dĺžka. Následne zhromažďovali spätnú väzbu v ktorej účastníci tri krát denne odpovedali na úroveň stresu v konkrétnom časovom úseku. Najčastejšie využívané aplikácie boli mail, kalendár a prehliadač. Najmenej používané boli hry a nadbytočné aplikácie. Výsledky práce boli celkom prekvapivé pričom model sa učil zo 70% zozbieraných dát a zvyšných 30% určil na testovanie. Stres sa v priemere podarilo detegovať v 75%.

### 3.3 Určovanie sentimentu z textu a hovorového slova

Určovanie sentimentu zo slovenského textu je úloha, ktorou sa zaoberá viacero výskumných prác priamo na našej škole. Z tohto dôvodu sme zvažili využitie existujúcich prác, ktoré sú na našej škole dostupné.

Prvou možnosťou bolo kontaktovanie tímu Imagine Cup, ktorí vo svojej práci odhaľujú nevhodné komentáre. Súčasťou ich práce je odhaľovanie sentimentu textu komentárov a však primárne sa zameriavajú na anglický jazyk. Pre slovenský jazyk vytvorili jednoduchý určovač sentimentu na základe sentimentu jednotlivých slov (to znamená, že každé slovo malo hodnotu od -3 po 3 a následne jednotlivé hodnoty sčítali a predelili počtom slov v texte). Tento slovník našli na Githube a pochádza z Fakulty matematiky, fyziky a informatiku UK. Samostatné API k ich určovaču sentimentu však nemajú. Slovník, ktorý sme našli na githube:

<https://github.com/okruhlica/SlovakSentimentLexicon>

Ďalšou možnosťou bolo využitie určovania sentimentu anglického textu s prekladom slovenského textu do angličtiny. Microsoft Azure ponúka riešenie pre vyhodnocovanie sentimentu, a

však iba v angličtine. Na preklad by sme mohli použiť štandardný prekladač a následne vkladať preložený text do API od Microsoft Azure, a však je otázne nakoľko by prekladač zmenil význam textu.

Poslednou možnosťou bolo využitie bakalárskej práce nášho spolužiaka Rast'a Krchňavého, ktorý sa v rámci svojej bakalárskej práce venoval problému sentimentu v slovenskom texte. Jeho práca je aktuálne nasadená na Heroku a určuje sentiment ako binárny problém - čiže kladné alebo záporné. Po komunikácii s Rast'om sme zistili, že je ochotný vytvoriť API šité na mieru. Rast'o nám taktiež poskytol možnosť preučenia jeho modelu, ktorý je aktuálne natrénovaný na facebookových komentároch.

## **Emócie**

Okrem samotného sentimentu slov je možné rozoznávajúce emócií z hovoreného slova. Jednou z možností je Vokaturi (<http://developers.vokaturi.com/getting-started/overview>), ktorí poskytujú zdrojový kód na stiahnutie. Je to voľne dostupné a bezplatné. Táto služba dokáže rozpoznať emóciu z hlasu, a však je dosť slabo zdokumentovaná. Samotná implementácia podporuje Python (ale len pre Mac alebo Windows, čo predstavuje problém).

Ďalšou možnosťou je napr.: <http://www.wcl.ece.upatras.gr/en/ai/resources/demo-emotion-recognition-from-speech>

Emócie sa okrem hovoreného slova dajú rozpoznávať aj priamo z písaného textu. Existuje množstvo API, ktoré na základe vstupného textu a jeho následnej analýzy vrátia JSON, kde je percentuálne uvedené, k akej emócii text patrí. Problémom je opäť jazyk.

Príklad

[https://tone-analyzer-demo.mybluemix.net/?cm\\_mc\\_uid=25274456998314780159884&cm\\_mc\\_sid\\_502000](https://tone-analyzer-demo.mybluemix.net/?cm_mc_uid=25274456998314780159884&cm_mc_sid_502000)

Zaujímavý zdroj

<http://nordicapis.com/20-emotion-recognition-apis-that-will-leave-you-impressed-and-concerned/>

## **3.4 Prístupy využívajúce akcelometer**

**RAVI, Nishkam, et al. Activity recognition from accelerometer data. In: AAI. 2005. p. 1541-1546.**

Táto práca sa zaoberá rozpoznávaním činnosti používateľ a na základe dát z akcelometra. Rozpoz-

návanie činností je vnímané ako problém klasifikácie. Dáta získané z akcelometra sú podrobené extrakcii funkcií a následne spracované algoritmi pre klasifikáciu. Akcelometer na základe experimentov dokáže s pomerne vysokou presnosťou rozoznávať aktivity, ktoré sú spojené s pohybom, a však pokiaľ človek stojí na mieste a vykonáva nejakú aktivitu je to zložitejšie ju rozpoznať. Najväčšiu presnosť pri klasifikácii dosiahla metóda Plurality Voting.

**BREZMES, Tomas; GORRICO, Juan-Luis; COTRINA, Josep. Activity recognition from accelerometer data on a mobile phone. In: International Work-Conference on Artificial Neural Networks. Springer Berlin Heidelberg, 2009. p. 796-799.**

Táto práca sa zaoberá využitím bežného mobilného telefónu s akcelometrom na odhadovanie typu aktivity, ktorú používateľ vykonáva. Cieľom tejto štúdie je odhalenie možností telefónu na spracovanie dát z akcelometra v reálnom čase a ich následné vyhodnotenie. Celé spracovanie a vyhodnotenie nameraných dát bolo realizované v samotnom telefóne a teda dáta neboli preposielané na server. Súčasťou práce bol výber efektívnych metód na klasifikáciu, ktoré by boli schopné pracovať priamo v mobilnom zariadení.

**BAO, Ling; INTILLE, Stephen S. Activity recognition from user-annotated acceleration data. In: International Conference on Pervasive Computing. Springer Berlin Heidelberg, 2004. p. 1-17.**

V tejto práci sa autori zameriavali na rozpoznávanie aktivity na základe dvoch nositeľných akcelometrov, ktoré mali používatelia pripevnené na tele. Experiment bol vykonávaný na 20 účastníkoch. Samotnému experimentu predchádzala kalibrácia, kde používatelia simulovali jednotlivé typy činností. Úspešnosť navrhutej metódy je 84%. Medzi základné vlastnosti, ktoré boli skúmané patria - energia, frekvenčná entropia a korelácia dát z akcelometra.

**HONG, Yu-Jin, et al. Mobile health monitoring system based on activity recognition using accelerometer. Simulation Modelling Practice and Theory, 2010, 18.4: 446-455.**

V tejto práci sa autori zameriavajú na využitie akcelometra a RFID snímača na rozpoznávanie aktivity človeka. Na klasifikáciu 5 ľudských stavov sú v štúdiu využívané dva nositeľné snímače. Na klasifikáciu je využitý algoritmus rozhodovacieho stromu.

**SUN, Lin, et al. Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In: International Conference on Ubiquitous Intelligence and Computing. Springer Berlin Heidelberg, 2010. p. 548-562.**

Táto práca bola venovaná klasifikácii aktivity používateľa na základe dát z akcelometra telefónu. Medzi základné problémy, ktoré sú s mobilným akcelometrom spojené patrí poloha mobilného telefónu. Táto štúdia je zameraná na riešenie tohto problému. V práci bolo definovaných 6 základných pozícií mobilného telefónu a spôsoby klasifikácie na správne odhadnutie polohy telefónu v danom čase (SVM klasifikátor). Ďalším krokom bola samotná klasifikácia aktivít na základe nameraných dát.

#### **Odkazy na datasety:**

<https://archive.ics.uci.edu/ml/datasets/Activity+Recognition+from+Single+Chest-Mounted+Accelerometer#>

<https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones>

<https://archive.ics.uci.edu/ml/datasets/Dataset+for+ADL+Recognition+with+Wrist-worn+Accelerometer>

### **3.5 Nálady vs. emócie vs. stres**

**HAMAKER, E. L., et al. Modeling affect dynamics: State of the art and future challenges. Emotion Review, 2015, 7.4: 316-322.**

Autori v článku vysvetľujú pojem dynamiky, ktorý je charakterizovaný ako pudy, motívy alebo sily, ktoré sú spúšťáčom psychologických procesov. Článok je venovaný prehľadu state-of-the-art metódam, ktoré sa zaoberajú dynamickými afektívnymi procesmi človeka. Tieto metódy vychádzajú z tzv. intensive longitudinal data (ILD). Autori v článku komunikujú vplyv dynamických afektívnych procesov, ktoré ľudské správanie, rozhodovanie, ako aj motiváciu. V článku je prezentovaných 8 základných dichotómií, medzi ktoré patria:

1. single-versus multiple-person data,
2. univariate versus multivariate models,
3. stationary versus nonstationary models,
4. linear versus nonlinear models,
5. discrete time versus continuous time processes,

6. discrete variables versus continuous variables,
7. time domain versus frequency domain a
8. modeling the process versus computing descriptives.

Článok je venovaný skúmaniu jednotlivých dichotómií, ich výhodám a nedostatkom.

**RUSSELL, James A. Four perspectives on the psychology of emotion: An introduction. 2014.**

Autori v článku prezentujú 4 základné pohľady na emócie z hľadiska psychologickéj teórie. Medzi tieto pohľady patrí:

1. basic emotion theory
2. appraisal theory
3. psychological construction theory
4. social construction theory

Autori sa v článku zameriavajú na spísanie rôznych výskumných smerov v rámci výskumu emócií, ktoré sú v rámci psychológie realizované. V článku je zároveň prezentovaná štúdia vplyvu emócií na ľudské správanie. Štúdia potvrdila významný vplyv emócií na ľudské správanie a rozhodovanie.

**ZACHAR, Peter. Comment: Five uses of philosophy in scientific theories of emotion. Emotion Review, 2014, 6.4: 324-326.**

Táto práca je venovaná teórie emócií, ktorá je založená na piatich konceptuálnych kontrastoch. Medzi 5 základných konceptuálnych kontrastov podľa autorov patria:

1. essentialism versus nonessentialism,
2. discriminative versus integrative theories
3. individual versus social focus
4. instrumentalism versus scientific realism
5. Philosophy in the Background Versus in the Foreground

V článku autori opisujú rôzne fakty ovplyvňujúce emócie človeka. Článok je taktiež zameraný na opísanie psychologických postupov, ktoré sa pri skúmaní emócií využívajú.

**HEKTNER, Joel M.; SCHMIDT, Jennifer A.; CSIKSZENTMIHALYI, Mihaly. Experience**

**sampling method: Measuring the quality of everyday life. Sage, 2007.**

V tejto práci sa autori zameriavajú na vyhodnotenie kvality ľudského života a to prostredníctvom meraní zážitkov každodenného života. Autori v práci navrhli a podrobne opísali metódu Experience Sampling Method (ESM). Na kvalitu ľudského života vplyvávajú tri základné faktory: pohlavie jedinca, rodinný život a taktiež pracovná pozícia jedinca. Autori v práci prezentujú návrh meraní, ktoré sú vykonávané raz denne pri experimente v reálnom živote (nejedná sa o laboratórne experimenty).

Metóda ESM je založená na tzv Systematic phenomology, ktorá bola vyvinutá pred 30 rokmi v Chicagu. Základnou podstatou Systematic phenomology je kombinácia skúseností z reálneho života s empirickými nástrojmi (výskumné návrhy, štatistické analýzy). Súčasťou tejto metódy je návrh zberu kontextových a kontentových informácií z denného jedinca jednotlivcov. Metóda zameriava na časové hľadisko v svojí s aktivitami jedinca. Z výskumov vyplýva, že jednoduché vyplnenie malého dotazníka, kde participant odpovie na niekoľko otázok je užitočnejšie než mapa celodenných činností.

Autori v práci navrhli metódu zberu pocitov participanta. Participant označuje na škále 1-7 ako sa cíti (šťastný až nešťastný). Z výskumu vyplýva, že ľudia majú tendenciu zlepšovať svoje pocity šťastia v prípade uspokojenia svojich základných potrieb (napr. po jedle alebo po fyzickej aktivite). Presne opačne to funguje napr. pri upratovaní. Na pocity šťastia okrem iného vplyva aj vek participanta. Vo všeobecnosti pri určovaní miery šťastia platí tzv. absolútna pravda, kedy pocit jedinca je jedinou pravdivou informáciou. Čiže ak participant označí, že je šťastný, tak je šťastný, napriek akýmkoľvek ďalším udalostiam.

Bežný človek počas jedného týždňa zažíva 15 - 50 momentiek. Jedná sa o poklesy nálad alebo naopak o náhle zlepšenia nálad. Autori v práci prezentujú ESM metódu ako najlepší prostriedok na získanie dvoch údajov:

1. Čo ľudia robia počas dňa, kde a ským a
2. Ako ľudia vnímajú jednotlivé chvíľa v ich živote (merané cez rôzne dimenzie).

Autori sa v práci taktiež zaoberajú problémom cross-kulturálnej psychológie a vplyvom kultúry človeka na kvalitu jeho života. Medzi základné využitia metódy ESM patria oblasti epidemiológie alebo klinickej praxi. V klinickej praxi má svoje uplatnenie pri skúmaní vplyvov práce, rodiny na depresie a na kvalitu ľudského života. Taktiež sa využíva pri výskume kvality spánku alebo vplyvov podávaného lieku. ESM vo všeobecnosti ponúka odpoveď na otázku: "Ako sa človek cíti práve teraz?". Metóda neverie do úvahy vývoj pocitov, teda neodpovedá na otázku: "Ako sa človek cítil počas posledného týždňa?".

Autori v práci prezentujú experiment so zberom dát z inteligentných hodínok, kde sledovali 4 základné charakteristiky:

1. aktivitu používateľa,
2. časové pečiatky,
3. lokáciu a
4. spoločnosť účastníka.

Tieto dáta následne kombinovali s tzv. self-reportingom. Pri reportovaní pocitov využívali škály jednotlivých emócií. Taktiež poskytli možnosť zadávania kombinovaných emócií, nakoľko človek môže naraz zažívať niekoľko emócií. Self-reporting bol v práci realizovaný prostredníctvom dvojminútového dotazníka vždy raz denne. Autori v práci taktiež komunikujú dôležitosť poradia položených otázok. ESM teda poskytuje základné postupy na vytvorenie pomerne rozsiahleho profilu používateľa pre psychologické skúmanie.

Autori v práci odporúčajú úvodný zber demografických charakteristík ako sú: pohlavie, etnické pozadie, vek a rasa. Z hľadiska self-reportingu sa zamerali na vyplnenie týchto údajov:

1. čas a dátum
2. primárna aktivita
3. sekundárna aktivita
4. lokácia
5. myšlienky - na čo myslel
6. bol sám
7. s mamou
8. s otcom
9. s priateľmi
10. so spolužiakmi
11. koncentrácia
12. miera užívania si
13. ...

Autori taktiež v práci poskytujú základné rozdelenie aktivít, ktoré využili aj v rámci tejto štúdie:

1. produktívne aktivity - práca v škole, študovanie, platené zamestnanie;
2. základné aktivity - jedenie, osobná starostlivosť, transport, domáca práca, odpočinok, driemanie
3. voľný čas - socializácia, sledovanie televízie, športy, hry, hudbu, umenie, hobby, čítanie, premýšľanie, snívanie - môže sa členiť na vonkajšie (šport, hudba, umenie, koníčky, socializácia) a pasívny voľný čas (TV, čítanie, premýšľanie, snívanie) - problém s premýšľaním: premýšľať sa dá o všetkých uvedených aktivitách, pričom môžu byť aj paralelne vykonávané

## 3.6 Zhrnutie

V rámci výskumu detekcie stresu v rôznych prácach využívali biometrické senzory, hlavne EKG. Neinvazívne EKG senzory zachytávajú tep srdca pomocou elektród. Zo zachyteného signálu je náledne možné vypočítať RR intervaly, čo sú jednoducho povedané, časové intervaly medzi jednotlivými údermi srdca. Z týchto intervalou následne určovali úroveň stresu. Vo viacerých prácach sa im taktiež podarilo odlíšiť stres od fyzickej aktivity. Opísané metódy boli relatívne presné a plánujeme ich použiť.

V ďalších prácach sa zaoberali tým ako odlíšiť negatívny stres od pozitívneho pomocou odhalovania korelácii medzi srdcovou frekvenciou a dýchaním. V iných výskumoch sa zaoberali možnosťami znižovaním stresu. Používali rôzne metódy ako:

- relaxačné cvičenia
- počúvanie hudby
- dychové cvičenia

Niektoré výskumy využívali taktiež dáta z mobilných telefónov a odhalili, že človek sa správa inak keď je pod vplyvom stresu. Túto metódu plánujeme použiť aj v našom projekte.

V rámci nášho projektu budeme taktiež určovať sentiment SMS správ mobilných zariadení. Tento údaj bude jedným z parametrov pre určovanie stresu. Na určovanie sentimentu využívame externé API vyvinuté na fakulte.

Plánovali sme taktiež zaznamenávať hovory z mobilných telefónov a následne určovať emócie z hlasu. Túto metódu zatiaľ neplánujeme využiť z dôvodu veľkej spotreby batérie pri zaznamená-



vaní hlasu. Taktiež by tu bol pravdepodobne problém s narúšaním súkromia.

Pomocou zmeny polohy mobilného telefónu je možné podľa rôznych výskumov určiť aktivitu používateľa. Na tento účel sa využíva akcelerometer, ktorý je zabudovaný vo väčšine mobilných zariadeniach. Existuje viacero prác, ktoré sa tomuto problému venujú. Používajú buď vlastné dáta, alebo existujúce datasety. Na analýzu využívajú buď označovaný dataset alebo vlastné dáta s kalibráciou. Základným problémom môže byť poloha telefónu, ktorá sa dá riešiť klasifikáciou pozície telefónu. Úspešná klasifikácia aktivity používateľa môže dopomôcť k určeniu hladiny stresu.

## 4 Moduly systému

### 4.1 Mobil - zber dát

#### 4.1.1 PO3 (Pulse oxygen sensor)

##### Analýza

Miera okysličenosti krvi a frekvencia krvného tepu hovorí o životnom štýle pacienta, konkrétne akým pohybovým aktivitám sa počas dňa venuje.

##### Návrh

Senzor P03 zbiera dáta pacienta ohľadom okysličenosti krvi (bloodOxygen) a krvného tepu (pulseRate). Je závislý od knižnice dodávateľ a senzoru iHealth. Existuje aplikácia, ktorá dokáže komunikovať s daným senzorom cez Bluetooth, ale dáta posiela na ich server. S kúskom štastia by sa mohla dať vytvoriť implementácia v našej Android aplikácii, ktorá dokáže komunikovať so senzorom cez Bluetooth.

##### Implementácia

Na komunikáciu so senzorom použijeme knižnicu od iHealth<sup>1</sup>. Najskôr senzor detegujeme, pripojíme sa naň a potom budeme z neho ťahať dáta. Dáta budeme posielat' v pravidelných intervaloch (každých 60 sekúnd). Posielame ich vo formáte JSON<sup>2</sup> na adresu <http://147.175.149.141:8000/api/sensor/>.

##### Testovanie

Testovali sme odoslanie dát, či už jednorázovo, alebo v danom intervale (každých 60 sekúnd).

#### 4.1.2 AM3 (krokomer)

##### Analýza

Počet prejdenných krokov a spálených kalórií dokáže čiastočne vypovedať o životnom štýle používateľ a. Presne na toto sa dajú využiť hodinky AM3 od spoločnosti iHealth.

##### Návrh

---

<sup>1</sup><https://github.com/iHealthLabs/Android-SDK>

<sup>2</sup><https://github.com/Crakeh/eMotion-Web-App/wiki/API-Sensors>

Senzor AM3 zbiera dáta pacienta ohľadom počtu krokov za deň a spálených kalórií. Je závislý od knižnice dodávateľ a senzoru iHealth. Už existuje aplikácia, ktorá dokáže komunikovať s daným senzorom cez Bluetooth, ale dáta posielajú na ich server. S kúskom šťastia by sa mohla dať vytvoriť implementácia v našej Android aplikácii, ktorá dokáže komunikovať so senzorom cez Bluetooth.

### **Implementácia**

Pre komunikáciu so senzorom použijeme knižnicu od iHealth<sup>3</sup>. Najskôr senzor detegujeme, potom sa na neho pripojíme a začne komunikácia. Dáta budeme posielajú v pravidelných intervaloch (každú hodinu).

### **Testovanie**

Testovali sme odoslanie dát, či už jednorázovo, alebo v danom intervale (každú hodinu).

## **4.1.3 Bluetooth (zariadenia v okolí)**

### **Analýza**

Zariadenia Bluetooth sa nachádzajú takmer všade - fotoaparát, televízie, tlačiarne, mobilné telefóny a mnoho ďalších. Na základe okolitých Bluetooth zariadení bude teda možné zistiť približnú pozíciu používateľa. Táto pozícia môže mať vplyv na používateľovu hladinu stresu - špecifické Bluetooth zariadenia sa môžu nachádzať na pracovisku kde používateľ pracuje, v škole ktorú navštevuje, u zubára, prípadne rôzne ďalšie stresujúce prostredia. Bluetooth patrí k bežnej výbave moderných mobilných telefónov, takže pravdepodobnosť, že zariadenie, na ktorom bude bežať naša aplikácia nebude vybavené zariadením Bluetooth je prakticky nulová. Pre vyhľadanie okolitých zariadení je potrebné najprv zistiť, či je Bluetooth zapnutý, prípadne ho zapnúť. Po vykonaní skenu je potrebné Bluetooth automaticky vypnúť, pretože znižuje výdrž batérie. Zoznam získaných zariadení je následne potrebné odoslať na server.

### **Návrh**

Pred spustením skenovania je potrebné zistiť, či je zariadenie Bluetooth zapnuté. Ak nie je, treba ho zapnúť a uložiť jeho stav pred spustením skenovania. Ak bol pred skenovaním vypnutý, po ukončení skenovania ho aplikácia tiež vypne. Ak bol zapnutý, tak ho pravdepodobne používateľ práve používa a nechám ho zapnutý. Nájdené zariadenia sa následne odošlú na server pomocou komunikačného modulu, prípadne sa aj zobrazia v aplikácii.

---

<sup>3</sup><https://github.com/iHealthLabs/Android-SDK>

## Implementácia

Pre prácu so zariadením Bluetooth využijeme API, ktorú poskytuje operačný systém Android. Vytvoríme BroadcastReceiver, ktorý bude spracovávať udalosti *ACTION\_FOUND*, *BLUETOOTH\_DISCOVERY\_STARTED* a *BLUETOOTH\_DISCOVERY\_FINISHED*. Po zachytení udalosti *ACTION\_FOUND* overíme, či má zariadenie platný názov, pridáme ho do zoznamu zariadení a odošleme ho na zobrazenie do Aktivity BluetoothActivity, ak je aktívna. Po zachytení udalosti *BLUETOOTH\_DISCOVERY\_FINISHED* tento zoznam pošleme modulu na komunikáciu, ktorý ho následne odošle na server.

## Testovanie

Pre testovanie tohto modulu je potrebné zaistiť, aby bolo v okolí mobilného telefónu minimálne jedno Bluetooth zariadenie. Ak žiadne zariadenie dostupné nieje, test nebude možné uskutočniť. Následne je potrebné spustiť detekciu Bluetooth zariadení v tomto module. Detekcia prebieha automaticky po zapnutí telefónu a potom každých 5 minút, takže je potrebné počkať, alebo telefón reštartovať. Po ukončení detekcie zariadení budú na server odoslané nájdené zariadenia. Výsledok bude viditeľný medzi dátami práve prihláseného používateľa.

### 4.1.4 Komunikačný modul

#### Analýza

Zariadenie bude zbierať veľké množstvá údajov, ktoré bude potrebné následne odoslať na server. Server však môže byť často nedostupný. Či už z dôvodu výpadku internetového pripojenia alebo výpadku serveru samotného. V oboch prípadoch je potrebné dáta uložiť v mobilnom zariadení a na server ich odoslať neskôr. Služba komunikačného modulu musí byť vždy dostupná pre všetky senzory, ktoré odosielajú dáta na server. Musí teda zvládnuť paralelné požiadavky od viacerých tried.

#### Návrh

Komunikačný modul bude bežať na pozadí ako Servis. Dáta na odoslanie sa budú ukladať do zoznamu prostredníctvom volania verejnej metódy. Pre prípad, že túto metódu zavolajú dve vlákna naraz, bude potrebné zoznam chrániť mutexom. Okrem uloženia do zoznamu v pamäti, je dáta potrebné uložiť aj na perzistentné úložisko, aby sa nestratili v prípade vypnutia zariadenia. Po spustení služby bude potrebné zoznam z tohto úložiska tiež naplniť. Servis sa tieto dáta následne pokúsi odoslať na server. Ak nie je dostupný, po čase skúsi znova.

## Implementácia

Servis `DataSenderService` sa automaticky spustí po zapnutí aplikácie. Môže teda nastať prípad, že modul na zber dát zavolá metódu na odoslanie dát príliš skoro. V tomto prípade je potrebné najskôr počkať na inicializáciu `DataSenderService`. Zoznam, do ktorého sa budú ukladať dáta na odoslanie, bude inštancia triedy `ArrayList`.

Po zavolaní metódy na odoslanie dát sa zamkne mutex, ktorý tento zoznam chráni a až potom sa do neho vložia dáta. Dáta sa tiež uložia na persistentné úložisko pomocou `SharedPreferences`. Po vložení sa mutex odomkne. Servis obsahuje cyklus, ktorý čaká na naplnenie zoznamu dátami. Ak zistí, že sú v zozname dáta na odoslanie, vyberie zo zoznamu jeden `JSONArray` a pokúsi sa ho odoslať. Ak sa ho odoslať nepodarí, vráti ho do zoznamu a aj do `SharedPreferences`.

## Testovanie

Testovanie tohto modulu prebieha spoločne s testovaním ostatných modulov, ktoré odosielajú dáta na server (napr. Bluetooth modul).

### 4.1.5 Autentifikácia používateľa

#### Analýza

Dáta ktoré budú zozbierané mobilným zariadením musia byť odoslané na server spolu s informáciou, ktorému používateľovi patria, dáta ktoré nie sú priradené žiadnemu používateľovi ani nemá zmysel odosielať na server. Bude preto potrebné vytvoriť systém, ktorý umožní používateľovi prepojiť svoje mobilné zariadenie so svojim účtom vo webovej aplikácii. Pre pohodlné používanie bude tiež potrebné umožniť pacientovi registrovať sa priamo zo svojho mobilného zariadenia. Tiež bude potrebné používateľa automaticky prihlásiť po reštartovaní aplikácie.

#### Návrh

Aplikácia bude obsahovať možnosti v menu pre registráciu a prihlásenie. Po kliknutí na jednu z týchto možností sa zobrazí registračný resp. prihlasovací formulár, do ktorého používateľ zadá svoje informácie. Po používateľovom potvrdení registrácie resp. prihlásenia, bude odoslaná požiadavka na webový server, ktorý ju spracuje a ako odpoveď odošle informáciu o vykonaní požiadavky (úspech, zlyhanie a ďalšie dodatočné informácie). Po úspešnom prihlásení bude v odpovedi odoslaný token, ktorý sa uloží a bude prítomný v každej požiadavke na odoslanie dát na server. Pomocou tohto tokenu bude vedieť server priradiť dáta ku konkrétnemu používateľovi. Tiež bude potrebné vytvoriť verejnú metódu, pomocou budú môcť ostatné moduly zistiť či je/nie je používateľ práve prihlásený a získať jeho e-mail adresu.

## Implementácia

Požiadavka na prihlásenie resp. registráciu bude odoslaná na aplikačné rozhranie servera ako HTTP POST požiadavka. Ako odpoveď server odošle JSON objekt obsahujúci informácie o vykonaní požiadavky a v prípade úspešného prihlásenia aj spomínaný token. Tento token - reťazec 40 náhodných hexadecimálnych číslíc - bude po úspešnom prihlásení uložený spolu s e-mailom používateľa do SharedPreferences. Automatické prihlásenie bude prebiehať v službe komunikačného modulu, ktorá sa spúšťa automaticky po zapnutí mobilného telefónu. Aby mohli ostatné moduly aplikácie zistiť, či je používateľ prihlásený, bude dostupná verejná metóda `IsLoggedIn()` a `GetLoggedEmail()`.

### 4.1.6 Kalendárove udalosti

#### Analýza

Pre lepšiu analýzu emócií a správania, je vhodné sledovať aj udalosti, ktorých sa pacient zúčastňuje. Bežať z jednej udalosti ku druhej a stíhať to načas, môže negatívne ovplyvniť hladinu stresu. Vďaka dostupným kalendárom v Android zariadení používateľ a sa k nim môžeme dostať.

#### Návrh

Najskôr treba zistiť, či už sú kalendáre s udalosťami pre konkrétneho používateľa uložené na serveri. Potom vytvoríme službu, ktorá bude bežať na pozadí a sledovať všetky akcie používateľa ohľadom kalendára. To zahŕňa pridávanie, odstraňovanie udalostí alebo aktualizáciu jednotlivých udalostí. Každá akcia bude zaznamenaná na serveri.

## Implementácia

Pre získanie dát ohľadom udalostí použijeme Android API rozhranie kalendára `CalendarProvider`<sup>4</sup>. Pri spustení aplikácie jednoduchou požiadavkou skontrolujeme, či už sú kalendáre pre daného používateľa uložené na serveri. Ak nie, tak ich so všetkými udalosťami pošleme na server, všetko v jednej JSON<sup>5</sup> požiadavke. Potom implementujeme službu `CalendarService`, ktorá bude bežať na pozadí a kontrolovať akcie používateľa ohľadom kalendára a udalostí. Každá akcia sa pošle cez službu `DataSenderService` v formáte JSON na server.

## Testovanie

Modul sme otestovali tak, že skontrolujeme či prvom pustený pošle všetky dáta z kalendára.

---

<sup>4</sup><https://developer.android.com/guide/topics/providers/calendar-provider.html>

<sup>5</sup><https://github.com/Crakeh/eMotion-Web-App/wiki/API-Sensors>

Potom sme skúsili pridať, upraviť alebo aj odstrániť niektoré udalosti a skontrolovali, či sa aj tieto zmeny na serveri zaznamenali.

#### 4.1.7 WiFi modul

##### Analýza

V dnešnej dobe takmer každý jeden predaný Android má v sebe WiFi. Cieľom využitia wifi modulu v našej aplikácii je, aby sme vedeli zistiť, kde sa používateľ nachádza (domov, práca, kaviareň, ...) – aké názvy wifi sietí sa nachádzajú v jeho okolí. Android má prácu s wifi zdokumentovanú na veľmi dobrej úrovni a taktiež sa s ním ľahko robí. Prácu s wifi zabezpečuje WifiManager.

##### Návrh

Pre získanie dát z wifi použijeme WifiManager. V Androide na pozadí spustíme nami implementovaný Servis, ktorý bude spúšťaný AlarmManagerom. Dôvodom použitia AlarmManagera je to, aby Servis nebežal na pozadí stále, ale iba vtedy keď je to potrebné. AlarmManager je zodpovedný za to, aby každých 5 minút spustil Servis na detekciu wifi v okolí. Následne, keď zistíme počet a názvy wifi sietí v okolí tento servis vypneme. Týmto sa snažíme šetriť batériu mobilného telefónu. Ak spustíme detekciu wifi v okolí pomocou nášho Servisu a zistíme, že wifi je vypnutá, wifi zapneme. Samotný proces zisťovania wifi v okolí trvá pár sekúnd a po vykonaní skenovania okolia wifi zase vypneme.

##### Implementácia

Skenovanie wifi spustíme v Servise pomocou WifiManagera, kde najskôr skontrolujeme, či je wifi zapnutá alebo nie. Ak wifi zapnutá nie je, zapneme ju pomocou funkcie *setWifiEnabled(True)*. Následne zaregistrujeme Receiver, ktorý je zodpovedný za prijatie dát z Wifi. V tomto Receivery spracujeme dáta, ktoré prišli vo forme Listu. Skenovanie okolitých wifi následne zavoláme pomocou metódy *startScan()*. Na konci, keď Receiver príjme dáta, ho musíme odregistrovať a skontrolovať, či náš Servis spustil wifi alebo nie. Ak áno, tak musíme wifi vypnúť pomocou vyššie spomenutej funkcie *setWifiEnabled(False)*.

##### Testovanie

Funkčnosť modelu sme testovali v reálnych podmienkach. Po spustení aplikácie sme kontrolovali každých 5 minút, či zoznam obsahuje nové dáta zo senzorov.

## 4.1.8 GPS modul

### Analýza

Na zisťovanie polohy používateľ a je najvhodnejší spôsob používať GPS v mobilnom zariadení. Cieľom zisťovania polohy je určiť či konkrétne miesta vyvolávajú v používateľovi stres. Medzi takéto miesta môžu patriť napríklad preplnené ulice v starom meste, návšteva svokry alebo obchod s nahnevanou predavačkou. GPS patrí k prostriedkom Androidu, ktoré sa nedajú automaticky zapnúť, teda zapnutie GPS musí vykonať samotný používateľ. V Androide sa však nachádza viacero prostriedkov, vďaka ktorým sa dá zisťovať poloha. Jedným z takýchto spôsobov je napríklad internet.

### Návrh

Pre získanie dát o polohy používateľ a vytvoríme Service *MyLocationService*, ktorý sa bude spúšťať na pozadí pomocou *AlarmManager*a. Samotné vyhľadávanie bude implementované pomocou *FusedLocationApi*, ktorého cieľom je nájsť polohu používateľ a. Keď sa nám podarí nájsť používateľovu polohu, uložíme si jednotlivé súradnice – Latitude, Longitude – a čas. Na základe týchto dát budeme vedieť kde a kedy sa používateľ nachádza.

### Implementácia

Pri implementácii zisťovania polohy používateľ a treba na začiatku nastaviť konštanty o frekvenciách detekcií pohybu (ako často v milisekundách obnoviť polohu – zavolá sa metóda *onLocationChanged(Location location)*), koľko krát sa má poloha obnoviť (poloha, ktorá sa určí prvý krát nebýva veľmi presná). Na začatie vyhľadávania polohy sa musíme spojiť s *GoogleApiClient* ako môžeme vidieť na metóde *buildGoogleApiClient()* nižšie.

```
private synchronized void buildGoogleApiClient() {
    Log.i(TAG, "Building GoogleApiClient");
    mGoogleApiClient = new GoogleApiClient.Builder(this)
        .addConnectionCallbacks(this)
        .addOnConnectionFailedListener(this)
        .addApi(LocationServices.API)
        .build();
    createLocationRequest();
}
```

V metóde *createLocationRequest()* nastavíme globálnej premennej *mLocationRequest* typu *Loca-*



tionRequest interval ako často sa má poloha obnovovať (použijeme vyššie spomenutú konštantu) a taktiež s akou presnosťou má polohu vyhľadávať. V našom prípade sa jedná o typ *LocationRequest.PRIORITY\_BALANCED\_POWER\_ACCURACY*. Následne sa skontrolujú povolenia, ktoré by sa mali nachádzať v manifeste - či aplikácia má právo zisťovať polohu užívateľa. Tieto povolenia sú dvoch typov a to *Manifest.permission.ACCESS\_FINE\_LOCATION* a *Manifest.permission.ACCESS\_COARSE\_LOCATION*. Ak tieto dve povolenia sú povolené zavolá sa

```
LocationServices.FusedLocationApi.requestLocationUpdates(  
    mGoogleApiClient, mLocationRequest, this);
```

čím sa začne vykonávať vyhľadávanie polohy. Pri nájdení polohy sa zavolá vyššie spomenutá metóda *onLocationChanged(Location location)*, v ktorej už pracujeme s aktuálnou polohou používateľa.

## Testovanie

Funkčnosť modelu sme testovali v reálnych podmienkach. Po spustení aplikácie sme kontrolovali každých 5 minút, či zoznam obsahuje nové dáta zo senzorov.

## 4.1.9 SMS modul

### Analýza

SMSky sú jedným z prostriedkov, ktoré dokážu v človeku vyvolať stres. Naším cieľom bolo zachytiť číslo odosielateľa a alebo adresáta a dĺžku tejto správy. Na základe týchto informácií sa budeme v budúcnosti snažiť predpovedať vyvolanie stresu aj pomocou SMS. V budúcnosti by sme taktiež chceli vypočítať sentiment správy. SMSky sa dajú v Androide zachytiť, a však len prichádzajúce. SMSky, ktoré používateľ odošle sa zachytiť pomocou nejakého Receiveru v Androide nedá. Práve preto na zachytenie odosielaných SMS bolo potrebné nájsť iný spôsob. Taktiež ak sa snažíme čítať v Androide SMS z Android databázy (miesto, kde sú uložené všetky správy), musíme mať na Android zariadeniach s API väčším alebo rovným ako API level 23 povolenie za behu systému. Pre zariadenia s nižším API level ako 23 stačí toto povolenie mať v Manifeste, kde bude automaticky povolené.

### Návrh

Pre získavanie prichádzajúci SMS v Androide, stačí nastaviť BroadcastReceiver so správnym

IntentFilterom. Pomocou IntentFiltera, dokážeme zachytiť prichádzajúce správy. Po príchode správy sa BroadcastReceiver automaticky zavolá a môžeme pracovať s SMS, ktorá bola doručená.

Odchádzajúce SMS sa však pomocou BroadcastReceiver zachytiť nedajú. Spôsob akým budeme zachytávať odchádzajúce SMS je taký, že pri prvom spustení si spočítame počet odoslaných správ. Následne pomocou AlarmManagera každých niekoľko minút budeme pomocou Service, ktorý spustí AlarmManager, kontrolovať počet odoslaných SMS. Ak sa počet odoslaných SMS zväčší, pozrieme sa na N nových odoslaných správ, ktoré následne spracujeme. Ak sa počet odoslaných zmenšil, tak si v pamäti budeme pamätať nový počet SMSiek.

### Implementácia

Na implementáciu prichádzajúcich SMS musíme v Manifeste nastaviť Receiver, ktorý bude zachytávať všetky udalosti, ktoré budú mať určitý Intent. Receiver nastavíme pomocou Intent filtra:

```
<intent-filter android:priority="999">
    <action android:name="android.provider.Telephony.SMS_RECEIVED"/>
</intent-filter>
```

V našom prípade musíme filtru nastaviť prioritu na 999, aby sa pri príchode SMS vykonal daný Receiver, nakoľko SMSky sú trochu náročnejšie na práva. Následne v android:name môžeme vidieť aký Intent sa snažíme zachytiť. Vďaka tomuto kroku sa nám zavolá náš BroadcastReceiver onReceive(). V tejto metóde už pracujeme s doručanou SMS - zistíme dĺžku správy, telefónne číslo a čas.

Pri implementácii odoslaných SMS treba pri prvom spustení vykonať inicializačné spočítanie odoslaných SMS. Ak používateľ nedá povolenie na prístup do databázy, kde sú uložené SMS, tak sa nastaví ich počet na nula. Následne sa pomocou AlarmManagera volá na pozadí každých N minút Service menom SmsOutgoingService, ktorého cieľom je spočítanie odoslaných SMS. Ak sa počet zmení, vložíme všetky odoslané SMS do ArrayListu. Tieto správy zoradíme podľa času a vyberieme aktuálny počet správ – počet uložených správ. Tieto správy následne znova spracujeme – zistíme dĺžku, komu boli odosielané a čas.

### Testovanie

Funkčnosť modelu sme testovali v reálnych podmienkach. SMS sme si posielali medzi zariadeniami a kontrolovali zoznam odoslaných a prijatých SMS v aplikácii.

## 4.1.10 Modul na detekciu hovorov

### Analýza

Telefónne hovory patria k činnostiam, ktoré nie každý človek zvláda bez stresu. Práve preto je naším cieľom detegovať prichádzajúce, odchádzajúce či zmeškané hovory. Konkrétne dáta, ktoré nás zaujímajú, sú číslo mobilu, trvanie hovoru a čas. Aj na základe týchto dát by sa dalo pracovať so stresom používateľ'a. V Androide sa dá použiť na detekciu prichádzajúceho ale aj odchádzajúceho hovoru BroadcastReceiver. Pri prichádzajúcom hovore dokonca vieme určiť kedy mobilný telefón začal zvonit', kedy bol hovor zdvihnutý a kedy bol hovor ukončený. Pri odchádzajúcom hovore vieme zistiť kedy začalo vytáčanie a kedy sa hovor ukončil, a však nevieme určiť či hovor bol zdvihnutý.

### Návrh

Pri detekcii odchádzajúcich hovorov musíme nastaviť nášmu Receiveru v Manifeste Intent Filter, na ktorý bude Receiver reagovať. Vďaka tomuto Receiveru budeme vedieť zachytiť odchádzajúce hovory. Každý Receiver môže obsahovať viacero filtrov, práve preto nastavíme tomu istému Receiveru aj filter pre zistenie stavu mobilu. Pri prichádzajúcom hovore sa totiž mení práve tento stav a vďaka nemu sa dá detegovať, kedy mobil začal zvonit', kedy bol hovor zdvihnutý a kedy ukončený.

### Implementácia

Implementácia hovorov prebiehala kvôli svojej zložitosti pomocou abstraktného Receiveru, ktorý detegoval odchádzajúce hovory a zmeny stavu mobilu. Pod zmenou stavu môžeme chápať zvonenie, prebiehanie hovoru a ukončenie hovoru. Tento Receiver obsahoval abstraktné metódy, ktoré bolo potrebné prekonať v triede, ktorá z tohto Receiveru dedila. Medzi tieto metódy patrili tieto metódy:

```
protected abstract void onIncomingCallReceived(Context ctx, String number, DateTime start);  
protected abstract void onIncomingCallAnswered(Context ctx, String number, DateTime start);  
protected abstract void onIncomingCallEnded(Context ctx, String number, DateTime start, DateTime end);  
protected abstract void onOutgoingCallStarted(Context ctx, String number, DateTime start);  
protected abstract void onOutgoingCallEnded(Context ctx, String number, DateTime start, DateTime end);  
protected abstract void onMissedCall(Context ctx, String number, DateTime start);
```

Následne ak sa zmenil stav v mobile tak sme takúto metódu zavolali v Receiveri v metóde `onCallStateChanged()` takýmto spôsobom:

```
switch (state) {  
    case TelephonyManager.CALL_STATE_RINGING:  
        isIncoming = true;  
        callStartTime = new DateTime();  
        savedNumber = number;  
        onIncomingCallReceived(context, number, callStartTime);  
        break;  
}
```

Vďaka tomuto sme v samostatnej triede, ktorú sme spomínali vyššie (trieda, ktorá dedí od abstraktného Receiveru a prekonáva vyššie spomenuté metódy) vedeli pracovať so stavmi telefónu a taktiež pracovať so samotnými hovormi - odchádzajúcimi či prichádzajúcimi.

## Testovanie

Funkčnosť modelu sme testovali v reálnych podmienkach. Pomocou dvoch Android zariadení sme si navzájom telefonovali a následne kontrolovali v aplikácii, či sa podarilo tieto hovory zachytiť.

### 4.1.11 Zber charakteristík SMS

#### Analýza

Pre zisťovanie aktuálnej nálady a emócií používateľ a je potrebné získať informácie o tom, ako používateľ komunikuje s ostatnými ľuďmi v jeho okolí. Prvotným cieľom bolo vyhodnocovanie sentimentu SMS správ na vzdialenom servery, avšak s tým by mohol mať samotný používateľ veľké výhrady. Práve preto sme sa sústredili na zberanie informácií z SMS ako je počet otáznikov, výkričníkov, šťastných smajlíkov a smutných smajlíkov priamo na Android zariadení.

#### Implementácia

Implementácia prebiehala tak, že sme si zobrazil sms. Následne sme spočítali všetky otázniky a výkričníky. Proces počítania sme robili tak, že sme nahradili všetky znaky výkričník prázdny stringom a následne sme porovnali dĺžky správ. Ak dĺžka správ sa zachovala znamenalo to, že sa v texte žiadny výkričník nenachádza. Podobne sme postupovali aj pri otáznikoch a smajlíkoch. Na server sa následne len odoslala informácia o počte týchto charakteristík.

## Testovanie

Testovanie prebiehalo na android emulátore, na ktorom sme si simulovali príchody a odosielanie SMS.

### 4.1.12 Mi Band 1S

#### Analýza

Pre čo najpresnejšie určenie emócií používateľ a našej aplikácie, sme sa rozhodli využiť senzor Mi Band 1S na meranie tepu. Pre tento senzor však neexistuje oficiálna API a komunikovať s ním dokáže iba oficiálna aplikácia Mi Fit. Existuje však tiež open-source neoficiálna API, ktorá umožňuje spojenie so senzorom priamo z vlastnej aplikácie, t.j. bez použitia oficiálnej Mi Fit aplikácie. Okrem merania tepu podporuje aj počítanie krokov, ktoré budú taktiež zaznamenávané.

Ďalším problémom je spotreba energie - keďže táto API vyžaduje aby bolo mobilné zariadenie a náramok Mi Band stále v spojení. Po vytvorení prototypu sme však zistili, že aj pri neustálom spojení a meraní pri najvyššej možnej frekvencii (jedno meranie za 10 sekúnd), je spotreba energie akceptovateľná. Zároveň je však potrebné baterku šetriť vždy keď je to možné, teda vypnúť Bluetooth ak používateľ nemá náramok na ruke. Taktiež je potrebné mať podporu pre Bluetooth párovanie náramku a mobilného telefónu priamo z našej aplikácie. Po párovaní by mal mať používateľ prístup k informáciám z náramku, ako je napr. stav batérie, počet krokov a tep.

#### Návrh

Na základe analýzy bude do aplikácie potrebné implementovať neustále meranie tepu a počítanie krokov. Kvôli tomu, bude zas potrebné zakomponovať neoficiálne MiBand SDK do našej aplikácie. Každých približne 10 minút (spolu s ostatnými službami) bude spustená aj služba MiBandService. Ak je spojenie s náramkom aktívne a služba už beží, tak sa táto nová inštancia zastaví. Naopak, ak nie je vytvorené aktívne spojenie s náramkom, tak sa inicializuje API MiBand SDK a vytvorí nové spojenie.

Po úspešnom spojení sa zaregistruje Listener pre prijímanie informácií o tepe a Listener pre prijímanie informácií o krokoch. Tieto údaje sa nebudú okamžite odosielať, pretože budú prijímané veľmi často. Budú sa preto ukladať do zoznamu a odošlú sa raz za desať minút, pri spustení služby. Každé tri hodiny sa vykoná kontrola stavu batérie a v prípade potreby sa zobrazí varovná notifikácia. Ak meranie tepu zlyhá päť krát po sebe, tak bude služba zastavená a Bluetooth vypnutý, pretože používateľ náramok práve nepoužíva.

## **Implementácia**

Boli vytvorené dve nové aktivity - MiBandActivity, ktorá slúži na párovanie s náramkom a MiBandStatus, kde sa zobrazujú údaje o spojení, batérii, tepe a počte krokov. Bola taktiež vytvorená služba MiBandService, ktorá slúži na vykonávanie meraní, odosielanie dát a notifikovanie aktivity MiBandStatus, ak je práve spustená. Bolo potrebné vykonať úpravy v API MiBand SDK, pretože nepodporovala zrušenie Bluetooth spojenia a zrušenie registrácie Listenerov.

Aktivita MiBandActivity je dostupná iba v prípade, keď aplikácia ešte nie je spárovaná s náramkom MiBand 1S. Naopak, aktivita MiBandStatus je dostupná iba v prípade, že je aplikácia korektné spárovaná s náramkom MiBand 1S. Táto aktivita v sebe obsahuje statický Handler, ktorý prijíma správy a na základe ich obsahu aktualizuje dáta viditeľné používateľom. Vždy po otvorení tejto aktivity bude inicializované meranie stavu batérie, aj keď od posledného merania ešte neprešli tri hodiny - aby boli používateľovi vždy dostupné čo najviac aktuálne informácie.

Služba MiBandService odosiela informácie do Handleru v MiBandStatus vždy po prijatí výsledkov merania vo svojich Listeneroch. Ak Handler neexistuje, t.j. je null, tak žiadne dáta odoslané nebudú a hneď začne ďalšie meranie.

## **Testovanie**

Na testovanie tohto modulu sme použili mobilné zariadenie, ktoré má nainštalovanú aplikáciu, avšak zatiaľ nebolo spárované s žiadnym náramkom. Vykonali sme párovanie pomocou aktivity MiBandActivity a následne sme monitorovali namerané údaje v aktivite MiBandStatus. Po odoslaní dát na server sme tiež preverili, či sa tieto dáta zobrazili v našom účte na našej webovej aplikácii.

### **4.1.13 Dotazníky aktivít**

#### **Analýza**

Na základe štúdia literatúry sme vybrali základné typy aktivít, ktoré človek v priebehu dňa vykonáva a ktoré majú potenciál ovplyvniť jeho náladu. Tieto aktivity sme rozdelili do 4 skupín na základné aktivity (zahŕňajúce základné ľudské potreby), povinnosti, voľnočasové aktivity a spoločenské aktivity. Podrobný zoznam rozdelenia aktivít je nasledovný:

#### **1. Základné**

- jedenie
- doprava
- osobná hygiena
- domáce práce
- odpočinok
- driemanie

## 2. Povinnosti

- práca v škole
- štúdium
- platené zamestnanie

## 3. Voľný čas a) vnútorné aktivity

- pozeranie TV
- čítanie
- písanie (hudba, básne, ...)
- kreslenie
- počúvanie hudby
- hra na hudobnom nástroji
- spoločenské hry
- počítačové hry

## b) vonkajšie aktivity

- kolektívny šport
- individuálny šport
- tanec
- prechádzky
- turistika
- starostlivosť o zvieratá

#### 4. Spoločenské akcie

- párty
- stretnutie s priateľmi
- návšteva kina
- návšteva divadla
- návšteva kultúrneho podujatia

#### **Návrh**

V rámci návrhu sme zvažili možnosť inicializačného označenia aktivít a dopĺňania vlastných aktivít používateľom. Za týmto účelom sme navrhli inicializačný dotazník, ktorý si používateľ vyplní po registrácii a ktorý obsahuje celý zoznam aktivít. V tomto dotazníku používateľ vyznačí iba tie aktivity, ktoré zvykne vykonávať. Následne po notifikácii vždy večer označí zoznam aktivít, ktoré robil daný deň. V tomto prípade sa jedná o skrátený zoznam na základe inicializačného označovania. Používateľ má zároveň možnosť pridávať vlastné aktivity a to aj v inicializačnom dotazníku, ako aj v priebežných dotazníkoch. Používateľ má vždy možnosť vrátiť sa k inicializačnému dotazníku a doplniť alebo zmeniť označené aktivity. Jednotlivé dáta sú spracované a odoslané na server.

#### **Implementácia**

Zoznam a kategórie týchto aktivít máme v mobilnej aplikácii staticky uložené. Implementovali sme to pomocou klasického vertikálneho RecyclerView, kde každá aktivita je len text s Checkboxom. Kategórie sme implementovali ako obrázky, taktiež cez RecyclerView.

Pre zobrazenie notifikácie sme použili NotificationManager. Pre pravidelné pýtanie sa používateľ a čo robil sme použili AlarmManager.

#### **Testovanie**

Skúšali sme to mať celé dni zapnuté a aktivity čo sme robili si odklikávať.

#### **4.1.14 Zber nálad a emócií**

##### **Analýza**

Pri čítaní odborných článkov sme zistili, aký je vzťah medzi emóciami a náladami. Emócie predstavujú reakciu človeka na vzniknutú udalosť. Sú intenzívnejšie než nálady a menia sa oveľa častejšie. Zo psychologickéj literatúry vyplýva, že najvhodnejšie je ich merať každé tri hodiny, a



to aj vzhľadom na preťaženie a ochotu používateľov. Náklady sú dlhodobejšie (priemerne trvajú niekoľko hodín alebo dní) a sú výsledkom kombinácie niekoľkých emócií. V psychologickej literatúre sú často krát označované ako emocionálny stav.

## **Návrh**

Na základe vykonanej analýzy sme navrhli zber nálad a emócií v podobe notifikácií. Pričom emócie sú zbierané každé tri hodiny (mimo nočných hodín) a náklady vždy raz večer. Používateľ si vyberá medzi tromi náladami a siedmimi emóciami. Vyberané náklady sú: Dobrá, Zlá a Neutrálna. Do budúca zvažujeme možnosť zadávania nálad prostredníctvom škály. Emócie sú zadávané výberom zo 7 emotikonov, ktoré zodpovedajú emóciám: Šťastie, Strach, Znechutenie, Prekvapenie, Smútok, Hnev a Očakávanie. Pri zbere emócií používateľ odpovedá na otázku: “Ako sa cítiš?” a pri zbere nálad na otázku: “Aký si mal deň?”.

## **Implementácia**

Zoznam smajlíkov ktoré reprezentujú náklady a emócie sme implementovali pomocou RecyclerView ktorý používa GridLayout. Pokliknutí na smajlíka sa dáta odošlú na server.

Pre zobrazenie notifikácie sme použili NotificationManager. Pre pravidelné pýtanie sa používateľ a čo robil sme použili AlarmManager.

## **Testovanie**

Skúšali sme to mať celé dni zapnuté a náklady a emócie si odklikávať.

### **4.1.15 Dashboardy s grafmi**

#### **Analýza**

Pri tvorbe grafov, sme sa snažili nájsť knižnicu, ktorá nám ponúkne čo najviac možností pre možnú úpravu a prácu s nimi. Pri hľadaní možných alternatív, sme našli tieto knižnice: MPAndroidChart, Holo Graph Library, aCartEngine, ChartView, aFreeChart, ChartDroid, charts4j, GraphView, AndroidPlot, WilliamChart. Jedným z najdôležitejších faktorov pre výber knižnice bola dostatočná dokumentácia ku grafom. Po viacerých otestovaných alternatívach sme sa rozhodli pracovať s knižnicou MPAndroidChart. Knižnica ponúka dostatočný počet rôznych grafov, veľké možnosti úprav, vlastné prvky v grafoch pričom samotná knižnica má veľkú podporu na GitHubu.

## Návrh a implementácia

V našej aplikácii sme pracovali so štyrmi druhmi grafov. Prvý typ grafu predstavuje PieChart (v preklade kruhový/koláčový graf). Tento typ grafu používame v našej aplikácii na hlavnej obrazovke, k zobrazeniu informácií o náladách a emóciách, ktoré mal používateľ v posledných dňoch. Grafy sa nachádzajú vo ViewPager, ktorý zabezpečuje zobrazovanie starých grafov pomocou gést (posun prstom doľava alebo doprava). Pri vykreslení grafu sa vždy vykoná dopyt na server, na ktorý odošleme údaje o časovom horizonte, z ktorého chceme získať dáta. Server nám vždy vráti percentuálne kategórie, ktoré už len vykreslíme do grafu.

Druhým typom grafu je LineChart (v preklade čiarový). Na tomto type grafu zobrazujeme elektronickú socializáciu používateľ a fyzickú aktivitu používateľ a. Obidva typy dát sa počítajú na servere. Tieto grafy zobrazujeme pre jeden deň, pričom používateľ môže vidieť ako sa vyvíjala jeho aktivita alebo socializácia počas celého dňa.

Predposledným typom grafu bol taktiež LineChart, avšak tento graf zobrazoval dáta v reálnom čase zo senzoru na meranie tepu a okysličenia krvi. Pri tomto grafe bolo dôležité v prvom rade vytvoriť spojenie so zariadením. Pri spustení grafu sa najprv zobrazil používateľovi zoznam Bluetooth zariadení v jeho okolí. Následne po zobrazení zoznamu a preskenovaní všetkých dostupných zariadení sa mobilné zariadenie automaticky pripojilo na náš senzor a začalo komunikovať. Celý proces trvá približne 20 sekúnd. Po uplynutí času používateľ môže sledovať dáta so senzoru v reálnom čase.

Posledným typom grafu bol StackBar, ktorý predstavuje stĺpcový graf, ktorý je ešte v rámci stĺpca delený na viacero kategórií. StackBar sme použili na zobrazenie fyzickej aktivity, pričom tentoraz používateľ môže vidieť aké činnosti používateľ v čase robil. Jeden stĺpec mohol byť rozdelený na 5% kráčanie, 5% preprava vozidlom a 90% bicyklovanie.

### 4.1.16 Big5 dotazník

#### Analýza

Vzhľadom na analýzu dostupnej literatúry, ako aj rozhovoru s tímom psychológov sme zistili, že na vývoj emocionálneho stavu človeka do veľkej miery vplýva jeho osobnosť. Existuje množstvo prístupov, ktoré skúmajú osobnosť človeka. Medzi najznámejšie patria: Eysenck's Personality Questionnaire, Minnesota Multiphasic Personality Inventory a Big Five Inventory. Pre naše účely aj vzhľadom na komunikáciu s psychológmi sme si vybrali Dotazník Veľkej

Pat'ky (z angl. Big Five Inventory). Tento dotazník skúma osobnosť cez 5 charakteristík - neurotizmus, extraverziu, prívetivosť, svedomitosť a otvorenosť voči novej skúsenosti. Neeurotizmus vyjadruje mieru ako sa človek ľahko znepokojí, extraverzia intenzitu medziľudských kontaktov, svedomitosť mieru zodpovednosti, prívetivosť mieru ohľaduplnosti a entuziazmu a otvorenosť voči skúsenosti otvorenosť voči novým ľuďom, skúsenostiam a taktiež mieru estetického cítenia. Pre naše účely sme si vybrali NEO FFI, čo je 60 otázková verzia Dotazníka Veľkej Päťky.

### **Návrh a implementácia**

Pri prvom použití aplikácie používateľ vyplní osobnostný dotazník Big5. Tento dotazník je vyplňa prostredníctvom systému Crowdex a v mobilnej aplikácii je zobrazený cez webové zobrazenie WebView. Pri pripájaní sa do systému Crowdex, bolo potrebné vygenerovať jedinečný link pre každého používateľa. Tento link je generovaný pomocou verejného a privátneho kľúča, ktorý je nakoniec hashovaný pomocou SHA-256. Rovnaký hash sa vykonáva aj na strane servera, pre overenie jedinečnosti linku. Akonáhle používateľ vyplní dotazník, z mobilnej aplikácie sa pošle POST REQUEST na server o tom, že dotazník bol vyplnený spolu s tokenom používateľa, aby bolo jasné kto dotazník vyplnil. Zo serveru na základe toho pošleme požiadavku do systému Crowdex a vyžiadame si výsledky dotazníka. Tie uložíme v našej databáze.

## **4.1.17 Rozpoznané aktivity**

### **Analýza**

Rozpoznávanie aktivít je dôležitou súčasťou našej aplikácie. Dôležitou preto, lebo práve fyzická aktivita, môže výrazne pôsobiť na uvoľňovanie stresu človeka. Ak by sme vedeli detegovať, akú činnosť používateľ androidu robí musíme sa zamerať na činnosť akcelerometra na mobilnom zariadení. Avšak zle naprogramovaný akcelerometer, môže spôsobovať veľké vyvíjanie baterky. Okrem iného, bolo by potrebné nacvičiť model, ktorý bude vedieť identifikovať činnosti používateľov. Keďže táto úloha predstavuje dostatočne veľký problém z hľadiska implementácie, snažili sme sa hľadať už existujúce riešenia. Nakoniec sme našli špeciálne API od Googlu, ktoré robí presne to, čo potrebujeme. API sa volá Awareness API. Dokáže detegovať štyri základné činnosti ako pokoj, beh/chôdza, preprava dopravným prostriedkom a bicyklovanie.

### **Návrh a implementácia**

Rozpoznávanie aktivít používateľa je implementované podobným spôsobom, ako zvyšné získavané dáta na pozadí. Každých päť minút sa spustí IntentService, ktorý je spustený po dobu 20 sekúnd. Počas týchto sekúnd sa používa Awareness API, ktoré na základe akcelerometra zistí uje aké aktivity vykonáva používateľ. Po každej zistenej aktivite, je uložený záznam o nej

do SharedPreferences pamäti, ktorá je neskôr využívaná pri odosielaní dát na server. Awareness api dokáže detegovať aj viacero aktivít naraz. Pri spracovaní údajov, vždy poskytuje informáciu s ako presnosťou používateľ danú činnosť robí.

## Testovanie

Testovanie prebiehalo vykonávaním činností v určitých časových horizontoch a následným kontrolovaním nameraných hodnôt.

## 4.2 Web

### 4.2.1 Autentifikácia používateľa

#### Analýza

Nakoľko náš systém pozostáva z dvoch častí, z webovej aplikácie a mobilnej aplikácie, je potrebné zabezpečiť autentifikáciu na oboch stranách. Používatelia majú možnosť zaregistrovať sa na webovej stránke alebo v mobilnej aplikácii. V systéme sa nachádzajú tri typy používateľov - pacienti, lekári a administrátori. Úlohou administrátorov je registrovanie lekárov. Registrácia lekárov je realizovaná manuálne a je vykonávaná prostredníctvom Django admin panelu. Pacienti sa môžu v aplikácii registrovať svojpomocne.

#### Návrh

Do našej aplikácie sa pacienti môžu zaregistrovať pomocou webového rozhrania alebo pomocou webovej služby. Na vytvorenie webovej registrácie sme použili django-allauth<sup>6</sup>. Táto knižnica ponúka základné funkcie potrebné pri autentifikácii používateľa ako registrácia, prihlásenie, reset hesla, overenie e-mailovej adresy a pod. Taktiež je možné pomocou tejto knižnice vytvoriť vlastný model používateľa. Pre registráciu používateľa prostredníctvom mobilnej aplikácie sme vytvorili webovú službu. Na tento účel sme využili knižnicu django-rest-auth. Táto knižnica vie využiť náš model používateľa a umožňuje jeho registráciu prostredníctvom REST API. Podobne ako django-allauth táto knižnica ponúka všetku potrebnú funkcionálnu k autentifikácii používateľa.

#### Implementácia

Pri registrácii či už prostredníctvom webového rozhrania, alebo mobilnej aplikácie, musí používateľ zadať email a heslo. Následne overíme či používateľ s takým e-mailom už náhodou

---

<sup>6</sup><http://django-allauth.readthedocs.io/en/latest/>

neexistuje a taktiež či používateľ nezadal príliš jednoduché heslo. Ak je všetko v poriadku, odošleme používateľovi e-mail za účelom overenia jeho e-mailovej adresy. V tele e-mailu sa nachádza URL, pomocou ktorej používateľ potvrdí korektnosť jeho e-mailovej adresy a zároveň tým aktivuje svoj účet. Prihlásenie je možné až po úspešnej aktivácii. Po prihlásení má používateľ možnosť zmeniť svoj e-mail alebo heslo. Taktiež ak sa používateľ nevie prihlásiť z toho dôvodu, že zabudol svoje heslo, má možnosť vygenerovať si nové, ktoré mu bude zaslané na e-mailovú adresu použitú pri registrácii.

Knižnica django-allauth ponúka okrem funkcionality aj základné templety pre webové rozhranie. Tieto však nie sú nijak štylizované a teda nevyzerajú prijateľne pre používateľa. Preto bolo potrebné pôvodné teploty prepísať a vytvoriť nové štýly.

Používateľ môže používať aplikáciu aj bez registrácie a prihlásenia. Po prvom spustení mobilnej aplikácie sa na pozadí automaticky na server odošle Android ID, ktoré je pre každý mobilný telefón, a teda aj používateľa unikátne. Na základe tohto Android ID vytvoríme na serveri anonymného používateľa s tokenom na posielanie dát, ktorý odošleme späť do mobilnej aplikácie ako odpoveď. Tento token bude používaný pri odosielaní dát na anonymný účet používateľa.

Ak sa používateľ neskôr zaregistruje, z mobilnej aplikácie odošleme na server Android ID spolu s emailom používateľa. Pomocou týchto dvoch údajov nájdeme používateľov anonymný aj neanonymný účet. Spojíme ich do jedného tým že skopírujeme údaje z neanonymného do anonymného. Neanonymný účet následne odstránime. Robíme to tak preto, lebo na anonymnom účte už môžu byť poslané dáta, ktoré sú naň naviazané. Prenášanie dát na neanonymný účet a jeho následne odstraňovanie by bolo oveľa komplexnejším procesom. Od tohto momentu či už bude používateľ v našej aplikácii prihlásený alebo neprihlásený, bude sa používať stále ten istý účet.

## **Testovanie**

Testovali sme registráciu používateľa pri ktorej je potrebné zadať korektný formát emailovej adresy a taktiež dostatočne silné heslo. Následne sme otestovali odosielenie potvrdzovacieho emailu, aktiváciu účtu, prihlásenie, reset hesla a zmenu prihlasovacích údajov.

## **4.2.2 REST API na príjem dát**

### **Zvolenie vhodnej databázy**

Pri voľbe databázy nakoľko pracujeme s množstvom real-time dát bolo nevyhnutné zvážiť typ databázy, ktorý budeme používať. Demografické dáta o používateľoch ako aj prihlasovacie údaje ukladáme do relačnej databázy, konkrétne sme sa rozhodli pre Postgres. Pri výbere spôsobu ukladania dát zo senzorov a z mobilného zariadenia sme si preštudovali niekoľko zdrojov na StackOverflow ako aj odborných článkov, ktoré sa tejto tematike venujú.

Medzi zdroje podľa ktorých sme sa nakoniec rozhodli patria:

*<http://dba.stackexchange.com/questions/13882/database-redesign-opportunity-what-table-design-to-use-for-this-sensor-data-col>*

**VAN DER VEEN, Jan Sipke; VAN DER WAAIJ, Bram; MEIJER, Robert J. Sensor data storage performance: Sql or nosql, physical or virtual. In: Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on. IEEE, 2012. p. 431-438.**

Z článkov vyplýva, že Postgres je veľmi pomalý na zapisovanie (oproti NoSQL databázam ako sú Mongo alebo Casandra), a však s čítaním je to lepšie. Nakoniec sme sa rozhodli využiť ukladanie dát do relačnej databázy Postgres s tým, že pomalé zapisovanie riešime ukladáním do vyrovnávacej pamäti. Vyrovnávacia pamäť v podobe Redisu preklápa dáta z API, ktoré slúži na príjem dát, do samotného Postgresu. Do budúca v prípade príjmu veľkých dát je možnou alternatívou napr. Mongo s využitím Sparku na dopytovanie.

Čo sa týka samotného modelu zvažovali sme dva typy uloženia. Prvou možnosťou bolo vytvorenie samostatných tabuliek, pre každý jeden typ senzoru. Toto uloženie by bolo prehládnejšie a uľahčilo by dopytovanie na konkrétny typ senzora. Jeho nevýhodou by bola náročnosť dopytovania sa na dáta zo všetkých senzorov za dané časové obdobie a taktiež spôsob pridávania nových senzorov. Nakoľko množina senzorov a dát, ktoré sa z mobilného zariadenia získavajú nemusí byť konečná, rozhodli sme sa uložiť všetky dáta v jednej tabuľke. V tabuľke je špecifikovaný typ senzora, alebo mobilného prostriedku (GPS, Wifi) a dáta zo zariadenia v podobe jsonu. Výhodou takéhoto uloženia je jednoduché pridávanie nových senzorov, nakoľko na strane servera to predstavuje pridanie jedného záznamu do databázy a nie je potrebná žiadna dodatočná implementácia (ak nerátame validáciu vstupu). Nevýhodou tohto uloženia je veľké množstvo záznamov, ktoré máme v jednej tabuľke.

## **Analýza**

Keďže náš systém pozostáva z dvoch nezávislých aplikácií (mobilná a webová), je potrebné

medzi nimi zabezpečiť určitý druh komunikácie z dôvodu prenosu dát. Toto je potrebné najmä pre prenos biometrických dát z mobilného zariadenia na webový server, kde budú následne uložené do databázy. Kvôli čistote a konzistentnosti dát je potrebné vytvoriť určitý druh validácie. Komunikácia bude využívaná aj pri registrácii alebo prihlásení používateľa do mobilnej aplikácie.

## Návrh

Mobilná aplikácia bude komunikovať s mobilným serverom prostredníctvom REST API. Na tvorbu REST API použijeme Django REST framework <sup>7</sup>, ktorý je v spolupráci s Django často používaný. Navyše tento framework ponúka širokú škálu funkcionalít ako napríklad rôzne spôsoby implementácie samotnej API a taktiež viaceré možnosti autentifikácie. Na validáciu dát využijeme JSON schémy.

## Implementácia

Na implementáciu API sme zvolili tzv. “Class-based views”, kde každá URL je reprezentovaná jednou triedou na webovom servere. V rámci triedy sa ďalej nachádzajú metódy, ktoré predstavujú jednotlivé typy HTTP requestov. Priamo v danej triede je možné zvoliť či na prístup k danej URL je potrebná autentifikácia alebo nie. V našom prípade je potrebná autentifikácia vždy, keďže ide o dáta samotných používateľov. Zvolili sme metódu autentifikácie pomocou tokenov. Po zaregistrovaní je každému používateľovi vygenerovaný token, pomocou ktorého sa môže autentifikovať. Aktuálne token nemá žiadnu expiráciu, čo do budúcnosti zvažujeme zmeniť. Hlavnou súčasťou API je posielanie biometrických dát a dát o používateľovi, ktoré sú hneď ako dorazia na webový server validované. Ak validácia prebehne úspešne dáta sú uložené do Redisu a následne postupne ukladané do relačnej databázy.

## Testovanie

Testovali sme prijímanie dát z jednotlivých senzorov, validáciu dát, ukladanie dát do Redisu a do relačnej databázy PostgreSQL.

### 4.2.3 Dashboard s dátami používateľa

#### Analýza

Hlavným účelom nášho systému je detekcia negatívnych emócií používateľa s cieľom znížiť negatívny vplyv týchto emócií na zdravie človeka. Nevyhnutnou súčasťou takejto aplikácie je vizualizácia dát, na základe ktorých sú detekované emócie. V našej aplikácii máme dvoch typov používateľov - pacientov a lekárov. Tieto dáta sú potrebné pre obidva typy používateľov. Pacient

---

<sup>7</sup><http://django-allauth.readthedocs.io/en/latest/>

by mal mať prístup k svojim dátam, aby si mohol prezrieť svoje historické dáta a videl ako sa zlepšuje, alebo naopak zhoršuje. Taktiež lekár by mal mať prístup k sumarizovaným dátam o jednotlivých pacientoch, ktoré mu môžu pomôcť k lepšiemu smerovaniu liečby.

### **Návrh**

Webová aplikácia bude poskytovať dva pohľady na dáta - pohľad pre lekára a pohľad pre pacienta. Pacientovi sa budú zobrazovať biometrické dáta v podobe prehľadných grafov. Zároveň bude mať pacient zobrazený graf o vývoji jeho negatívnych emócií, ktoré sú vypočítané na základe dát z mobilného zariadenia. Tieto dáta (wifi, sms, hovory a pod.) však pacientovi priamo zobrazované nebudú. Lekár má k dispozícii vývoj negatívnych emócií jednotlivých pacientov sumarizované v maximálne dvoch grafoch na každého pacienta, tak aby týmito dátami nebol zahltený.

### **Implementácia**

Z tohto modulu je aktuálne naimplementované kumulatívne zobrazovanie dát o jednotlivých používateľoch v tabuľke. To znamená, že každý používateľ si vie prezrieť všetky svoje záznamy z mobilného zariadenia alebo biometrických senzorov vo svojom účte v podobe tabuľky. Implementácia grafov zatiaľ nie je realizovaná.

### **Testovanie**

Testovali sme správne zobrazovanie dát jednotlivých používateľov.

## **4.2.4 Modul sentimentu**

### **Analýza**

Keďže cieľom našej práce je rozpoznanie emócií a na emócie má veľký vplyv obsah komunikácie, rozhodli sme sa vyhodnocovať sentiment SMS správ z mobilného zariadenia. Na základe sentimentu dokážeme zistiť, či je obsah správ pozitívny alebo negatívny, čo nám pomôže lepšie pochopiť kontext nálady používateľa. V rámci analýzy sme zvážili niekoľko možností. Existuje množstvo nástrojov pre vyhodnocovanie anglického textu, a však nakoľko je naša aplikácia určená primárne pre slovenský jazyk, tieto nástroje nemôžeme využiť. Z tohto dôvodu sme naviazali spoluprácu s jedným z našich spolužiakov, ktorý v rámci bakalárskej práce vytvoril vyhodnocovač sentimentu pre slovenský jazyk.

### **Návrh**

Po dohode s Rastom Krchňavým sme spoločne navrhli API na vyhodnocovanie sentimentu,



ktoré vyhovuje obom stranám. Náš aktuálny návrh zahŕňa využívanie Rast'ovho API, ktoré je natrénované na facebookovských príspevkoch a dáva pomerne presné výsledky a naimplementovanie vlastného vyhodnocovania sentimentu na základe dostupného slovníka.

## **Implementácia**

V súčasnosti modul zahŕňa pravidelné volanie API po príchode údajov z mobilnej aplikácie. Keďže text SMS-iek je citlivá informácia, tento text je zašifrovaný šifrovaním AES. Po prijatí je event s SMS-kou uložený do Redisu a následne je predspracovaný. Predspracovanie zahŕňa odšifrovanie správy, zavolanie API na určenie sentimentu, doplnenie dát o sentimente do tela eventu a následné uloženie do Postgresu. Nakoľko API nemusí byť vždy dostupné, SMS-ky ktoré neboli spracované, sú priebežne monitorované a spracovávané. Tento update je realizovaný každú hodinu v podobe cron jobu, ktorý nájde všetky SMS-ky, ktoré nemajú určený sentiment, následne tieto SMS-ky rozdelí do skupín, odšifruje a odošle na spracovanie. Posledným krokom je uloženie dodatočnej informácie do databázy. Uloženie je realizované funkciou *bulk\_update()*, ktorá umožňuje aktualizovať N objektov v rámci jednej query, čo je oveľa efektívnejšie než jednoduché *update()*.

## **Testovanie**

Modul bol testovaný simulovaním príjmu dát z mobilného zariadenia. Pri testovaní sme sa zamerali predovšetkým na funkčnosť a efektívnosť. Cron job bol testovaný prostredníctvom generovaných dát a manuálnym spúšťaním skriptu.

### **4.2.5 REST API na odosielanie dát**

Na mobilný telefón odosielame dáta o elektronickej socializácii, fyzickej aktivite, emóciách a náladách používateľ a. Presnejšie povedané z mobilu je odoslaná požiadavka na náš server. V tejto požiadavke sú dva dátumy - počiatočný a koncový. Podľa nich vieme, z ktorého obdobia máme odoslať do mobilnej aplikácie dáta ako odpoveď.

Mieru elektronickej socializácie a fyzickej aktivity prezentujeme jedným číslom. Teda za dané obdobie bude odoslané pole takýchto údajov. Emócie a nálady za požadované obdobie agregujeme. Vypočítame pre každú emóciu a náladu ako často sa vyskytujú v danom období resp. ako často ich používateľ pociťoval. Ako odpoveď do mobilnej aplikácie pošleme percentuálne zastúpenie všetkých nálad a emócií.

## **Modul elektronickej socializácie**

Jednou z metrík ktoré počítame z nazbieraných dát je elektronickej socializácia. Na jej výpočet používame nasledujúce údaje:

- Počet okolitých Bluetooth zariadení
- Počet okolitých WIFI zariadení
- Frekvencia a dĺžka telefonátov
- Frekvencia a dĺžka SMS správ
- Počet udalostí v kalendári

Najprv sa pozrieme na historické dáta používateľ'a. Vypočítame priemerné hodnoty dĺžky a frekvencie telefonátov, počtu okolitých zariadení a podobne. Každý používateľ môže mať tieto hodnoty iné, a práve preto ich počítame. Odhalíme tak strednú hodnotu na základe ktorej budeme porovnávať nasledujúce výpočty. Ak uvidíme že používateľ telefonoval častejšie alebo mal viacero udalostí v kalendári ako obvyčajne, odrazí sa to aj na väčšej elektronickej socializácia. Tento proces funguje samozrejme aj naopak.

Dáta o vypočítanej elektronickej socializácii sme sa rozhodli zobrazovať po týždňoch pomocou čiarového diagramu.

## **Modul fyzickej aktivity**

### **Analýza**

Pri analýze možností rozpoznávania fyzickej aktivity sme zvažili dve možnosti. Spočiatku sme plánovali rozpoznávanie fyzickej aktivity na základe dát z akcelometra. K tomuto sme si našťudovali množstvo literatúry, ale nakoniec sa ukázalo, že neustály zber dát z akcelometra predstavuje veľkú záťaž na batériu telefónu a museli sme zvažovať iné možnosti. Ako schodná cesta sa ukázalo využitie Awarnes API na rozpoznávanie aktivít. Toto API je súčasťou našej mobilnej aplikácie a vieme z neho získať pravdepodobnosti základných aktivít, ktoré človek môže vykonávať (chôdza alebo beh, vo vozidle, na bicykli, nečinnosť alebo nerozpoznaná aktivita).

### **Návrh**

Pri našom návrhu sme brali do úvahy 4 základné aktivity a to: chôdza, na bicykli, nečinnosť a vo vozidle, pričom za fyzickú aktivitu pokladáme aktivity: na bicykli a chôdza. Výpočet fyzickej aktivity teda predstavuje pomer pravdepodobnosti týchto aktivít k ostatným aktivitám (nečinný stav a vo vozidle). Tieto dáta sú normalizované na interval 0-100, na základe čoho dokážeme

určiť mieru fyzickej aktivity v percentách.

Na základe toho, že aj dáta z API, ktoré používame na rozpoznávanie aktivít sú zaujímavé, rozhodli sme sa vizualizovať aj tieto pomocou skladaného stĺpcového grafu, ktorý zobrazuje percentuálnu pravdepodobnosť vykonávania danej aktivity v danom čase. Fyzyckú aktivitu vypočítanú z týchto dát zobrazuje pomocou čiarového diagramu za jeden deň.

### **Implementácia**

Navrhnutý výpočet je realizovaný vždy po prijímaní dát z mobilného zariadenia. Po prijímaní dát z Awarnes API je vypočítaná miera fyzickej aktivity a toto číslo je uložené k záznamu o jednotlivých aktivitách. V prípade zlyhania výpočtu je naimplementovaný cron job, ktorý vždy raz v noci skontroluje všetky aktivity a v prípade chýbajúceho atribútu fyzickej aktivity znovu spustí výpočet.

### **Testovanie**

Testovanie bolo realizované spolu s testovaním odosielania a prijímania dát z Awarnes API. Testovanie cron jobu bolo realizované manuálnym vytvorením chybného datasetu a následným spustením cron jobu.

# A Testovanie s používateľmi

Mobilnú aplikáciu sme ukázali 5 potenciálnym používateľom, aby sme získali prvú spätnú väzbu. Bola to skupina zložená z mladých ľudí, troch študentov FIIT (muž: 1, žena: 2) a dvoch absolventov FIIT (muž: 2, žena: 0). Účastníci dostali aplikáciu do ruky a mohli si ju vyskúšať, pritom rozprávali, čo sa im páči, ale aj čomu nerozumejú alebo čo sa im nezdá. Mnohé pripomienky sa aj opakovali. Všetky sme si ich prešli, zhodnotili a vhodne zapracovali do aplikácie. Pripomienky a naše reakcie na ne sú prehľadne zobrazené v nasledujúcej tabuľke.

| PRIPOMIENKA  | Ako sme ju zapracovali do aplikácie?   |
|--|--|
| na hornej lište aplikácie sú skratky, ktorým používatelia nerozumejú         | skratky sme zmenili na celoslovné názvy a odkazy sme z hornej lišty presunuli do menu  |
| rozdiel medzi DONE a ACTIVITY  | jasne sme oddelili dotazník aktivít, ktorý sa teraz spúšťa len raz na začiatku a nachádza sa pri ďalšom takomto dotazníku s dotazníkom, ktorý sa pýta na každodenné aktivity používateľa automaticky na základe výsledkov prvého dotazníka |
| aplikácia by sa mi mala dať otáčať aj ako portrait aj ako landscape          | takéto otáčanie sme pre väčšinu okien už umožnili  |
| z niektorých grafov sa nedá intuitívne vrátiť späť                           | tým, že sme pridali možnosť landscape aj portrait pre všetky obrazovky, tak aj toto je intuitívnejšie a pridali sme aj šípku na návrat   |
| pri zadávaní nálad a emócií dlho trvá kým sa niečo stane, chýba spätná väzba | dielo vytvorili sme nový návrh, ktorý bude v dohľadnej dobe implementovaný - používateľ okamžite po kliknutí dostane spätnú väzbu, že údaje boli uložené   |
| posúvanie medzi obrazovkami pri posune obrazovkou prstom                     | toto sme sa rozhodli neimplementovať nakoľko posun prstom sme využili na prechod medzi rôznymi dátumami v grafoch  |
| posúvanie medzi grafmi pri posune obrazovkou prstom                          | toto sme sa rozhodli neimplementovať nakoľko posun prstom sme využili na prechod medzi rôznymi dátumami v grafoch  |
| formát dátumu nie je dobrý, bolo by vhodné aby tam bol celý dátum a aj deň   | zmenili sme formát dátumu a pridali sme aj deň   |
| aký je rozdiel medzi náladou a emóciou                                       | vytvorili sme úvodný návod, ktorý vysvetlí jednotlivé grafy a rozdiely medzi nimi  |
| čo je socializácia   | vytvorili sme úvodný návod, ktorý vysvetlí jednotlivé grafy a rozdiely medzi nimi  |

|   |  |
|---|--|
| čo znamenajú percentá pri grafoch                   | vytvorili sme úvodný návod, ktorý vysvetlí jednotlivé grafy a rozdiely medzi nimi  |
| vedieť pridať nové aktivity v dotazníku aktivít     | okrem výberu aktivít, ktoré sme tam dali my chceme v dohľadnej dobe pridať možnosť, kde si používateľ bude môcť zadať vlastné aktivity                                 |
| chcem mať real time dáta                            | v podstate všetky dáta sa v grafoch prejavia okamžite okrem socializácie, ktorá má komplikovanejší výpočet - ten sme však z jednodňového intervalu zmenili na hodinový |
| niekde je slovenský a niekde anglický text          | celú aplikáciu sme dali do slovenčiny  |
| pri kliknutí na šípku pri grafe to nie vždy reaguje | zväčšili sme šípku a pridali sme akciu, že na prechod obrazovkou prstom sa prepne medzi dátumami (čo je aj funkcia šípky)  |
| rovnako nazvané grafy                               | grafy sme odlíšili rôznymi názvami a vysvetlili v návode na úvod   |
| nemoderná registrácia a prihlásenie                 | spravili sme nový návrh dizajnu aplikácie, ktorý by mal byť v dohľadnej dobe implementovaný  |
| v menu zaniká logo                                  | spravili sme nový návrh dizajnu aplikácie, ktorý by mal byť v dohľadnej dobe implementovaný  |

Webovú časť sme netestovali s používateľmi, keďže tá slúži primárne na prácu s dátami a zobrazuje ich len v surovom stave. Dobudúcna sa nebránime ani rozšíreniu použiteľnosti na webe, ak zistíme, že to má potenciál.

## A.1 Prezentácia na ontožúre

Naša aplikácia, ako aj návrh experimentu bol prezentovaný na jarnom Ontožúre, kde sme získali rôzne pripomienky na vyjasnenie pojmov, celkového konceptu aplikácie, návrhu experimentu, ako aj ďalšieho smerovania projektu.

|  |  |
|--|--|
| PRIPOMIENKA  | Ako sme ju zapracovali do aplikácie?   |
| Zobrazovať socializáciu v rôznych častiach dňa                     | Socializácia bola prerobená na strane servera, ale aj na strane aplikácie tak, aby zobrazovala informácie každých pár hodín. |
| Chýba jednoduchšia varianta zadávania emócií a nálad do aplikácie. | Do aplikácie bol implementovaný widget.  |

# B Protokol k experimentu

**Meno a priezvisko** tím eMotion

**Supervisor** Ing. Peter Gašpar, Ing. Fedor Lehocki PhD.

**Názov projektu** Vytvorenie datasetu pre rozpoznávanie nálad

## **Stručný opis projektu**

Cieľ experimentu je vytvorenie datasetu, ktorý bude využitý pri návrhu metódy klasifikácie nálad. Naším cieľom je vytvorenie datasetu, ktorý bude obsahovať dostatok informácií na rozpoznávanie nálad používateľa. Rozpoznávanie nálad ako aj výber najvhodnejšej metódy bude predmetom ďalšieho výskumu. Zber dát bude realizovaný prostredníctvom mobilnej aplikácie eMotion a inteligentného náramku Mi Band 1S. Vzhľadom na počet dostupných náramkov, experiment bude realizovaný s 21 participantmi. Dĺžka trvania experimentu je 1 mesiac.

Výstupom experimentu by mal byť dataset obsahujúci nasledovné dáta: pre každého jedného používateľa - vyplnený BigFive dotazník a iniciálny dotazník aktivít a základná demografia (vek, pohlavie, zamestnanie, rodinný stav, váha, výška); nálada pre každý deň používania; aktivity pre každý deň používania; emócie (minimálne 3 krát za deň); kalendárové udalosti (prvotný import + zmeny); Wifi pripojenia v okolí; Bluetooth pripojenia; hovory (história hovorov, všetky odchádzajúce a prichádzajúce hovory - dĺžka trvania, typ hovoru, adresát); SMS - prijaté a odoslané (dĺžka správy, adresát, základná analýza emotikonov a interpunkcie); rozpoznané aktivity cez Awareness API; tep z Mi Band 1S a GPS. Jednotlivé charakteristiky budú zbierané v pravidelných časových intervaloch a ich zber bude priebežne sledovaný. Tieto dáta budú následne použité na klasifikáciu nálad, čo zahŕňa výber najvhodnejšej klasifikačnej metódy, ktorá s takýmto datasetom dokáže pracovať. V prípade úspešnej klasifikácie nálad je taktiež možnosť skúmania prepojenia medzi stresom a náladami. Stres v tomto kontexte môže byť určený na základe tepu z Mi Band 1S.

**Cieľ experimentu** Primárnym cieľom experimentu je vytvorenie datasetu pre ďalšiu analýzu. Sekundárnym cieľom je otestovanie použiteľnosti mobilnej aplikácie.

**Parametre experimentu** V experimente všetci participantia pracujú s jednou verziou aplikácie a všetci majú nastavené rovnaké podmienky pre zber dát.

**Účastníci** Súčasťou experimentu bude 21 participantov. Primárne by sa malo jednať o študentov vysokých alebo stredných škôl. Ideálne je rovnomerné zastúpenie pohlaví, čo sa nám ale s najväčšou pravdepodobnosťou nepodarí. Keďže našim ďalším cieľom je skúmanie datasetu ideálne je ďalšie rozdelenie (alebo výber konkrétnej podskupiny) z hľadiska rodinného stavu, zamestnania, ako aj sexuálnej aktivity (tu však vidíme trochu problém, nakoľko ľudia o takýchto veciach neradi hovoria).

**Sledované metriky** Našou základnou metrikou sú samotné dáta. Ich zber bude kontrolovaný priamo na webovom serveri. Metrikou použiteľnosti je počet resp. frekvencia zadávania spätnej väzby (počet zadaných emócií v priebehu dňa, počet zadaných nálad za týždeň, počet vyplnení dotazníka aktivít za týždeň).

**Scenár experimentu** Dĺžka trvania experimentu je jeden mesiac. Experiment začína stretnutím s participantmi (v prípade študentov našej fakulty sa môže jednať o kolektívne stretnutie v iných prípadoch sa zrejme bude jednať o viacero individuálnych stretnutí), na ktorom budú mať odprezentovanú základnú myšlienku projektu, celkový priebeh experimentu a taktiež im bude vysvetlený zoznam zbieraných atribútov (čo presne zbierame, ako to zbierame a prečo to zbierame). Súčasťou tohto stretnutia bude oboznámenie s inštrukciami experimentu (Inštrukcie participantí dostanú na papieri + budú prístupné na našej stránke). Odkaz na stránku, dokument s inštrukciami, ako aj zoznam kontaktov bude participantom odoslaný na mail.

Po oboznámení participantov s priebehom experimentu a ich súhlase, že sa chcú takéhoto experimentu zúčastniť im bude nainštalovaná aplikácia (s inštaláciou im pomôžeme priamo my na stretnutí) a odovzdaný senzor (tu by sme mali pomôcť s párovaním a taktiež by bolo vhodné otestovať funkčnosť + podpísanie hmotnej zodpovednosti). Po nainštalovaní by malo prebehnúť úvodné otestovanie funkčnosti. (vyžiadanie dát z každého atribútu, ktorý testujeme a kontrola dát na serveri - ich prijatie a formát). Po otestovaní iniciálneho zberu je participantovi ukázaná naša aplikácia.

Ďalším krokom je samotné vyplnenie troch dotazníkov: dotazník na demografické dáta, vyplnenie osobnostného dotazníka a dotazníka na aktivity (prebieha mimo stretnutia). Počas experimentu participantí odklikávajú nálady - vždy raz večer, aktivity - raz večer a emócie nálady ideálne každé tri hodiny alebo častejšie. Všetky ich aktivity v mobilnej aplikácii sú logované. Zároveň sa na webovom serveri vyhodnocuje úspešnosť zberu jednotlivých parametrov (vždy sa viaže na jedného participanta). Dva krát do týždňa sú jednotliví používatelia kontaktovaní či z ich

strany nenastali nejaké problémy. V prípade zníženej aktivity v zadávaní spätnej väzby, alebo problémov so zberom dát sú participanti kontaktovaní aj priebežne. V prípade akýchkoľvek otázok majú participanti možnosť zaslať email na adresu tímového projektu - tieto otázky, pripomienky a problémy musia byť promptne riešené. Po ukončení experimentu (možno aj každý týždeň) participanti vyhodnotia ich spokojnosť s mobilnou aplikáciou. Na záver testovania vyplnia SUS dotazník a bude im odovzdaná odmena.

### **Osnova inštrukcií pre participantov**

- vysvetlenie cieľ a projektu
- podrobné opis zbieraných dát
- Čo od nich potrebujeme (aj s obrázkami a postupom ako na to):
  - po registrácii
  - priebežne (emócie viacej naraz, skala pri naladach?)
  - 2 krát za týždeň
  - na záver
- Čo za to získajú
- Kontakt

### **Čo treba do testovanie dorobiť**

- zmena smajlíkov pri nálade na škálu
- možnosť označenia viacerých emócií
- oddialenie notifikácie vždy tri hodiny od poslednej aktivity
- možnosť nastavenia rôznych časových intervalov notifikácií
- nastavenie času, kedy participant nechce byť rušený
- pridanie dotazníka na demografiu do mobilnej aplikácie
- pridanie sexuálnej aktivity do zoznamu aktivít nakoľko má veľký vplyv na emócie jedinca
- doplnenie iniciálneho ottestovania
- dorobenie jednoduchšieho ottestovania na webe.



- pripraviť pekné a nápomocné inštrukcie už s novou funkcionalitou

# C Inštalčná príučka

## **Inštalácia aplikácie do mobilného telefónu**

Inštalácia aplikácie do mobilného telefónu je veľmi jednoduchá záležitosť. V prvom bode je potrebné skontrolovať, či vaše mobilné zariadenie má vyššie API ako 15, teda 16 a vyššie. API verzia 16 predstavuje Android verziu Jelly Bean. Naša aplikácia momentálne podporuje viac ako 96% mobilných zariadení na trhu. Následne je potrebné povoliť v nastaveniach inštaláciu aplikácií z iných zdrojov. Po tomto kroku stačí nakopírovať .apk súbor do mobilného zariadenia a dať ho inštalovať. Ak máte Android verziu 5 a vyššie, pri prvom spustení si bude aplikácia vyžadovať povolenia k istým prístupovým bodom ako je napr. Wifi, Bluetooth, kalendár, záznamy hovorov a ďalšie. Pre presnejšie informácie by ste mali povoliť prístup k týmto senzorum.

## **Inštalácia aplikácie pomocou Android Studio**

Ako prvé je potrebné stiahnuť z oficiálnej stránky Androidu – [developer.android.com](http://developer.android.com) – Android Studio. Po stiahnutí si ho nainštalujeme. Po nainštalovaní je potrebné v hornej lište nájsť SDK manažéra, cez ktorého môžeme stiahnuť potrebné balíčky. Správnu verziu balíčkov, ktoré máme sťahovať môžeme nájsť v súbore `build.gradle(Module:app)` na riadku, ktorý obsahuje informáciu `buildToolsVersion`. Po povolení vývojárskych možností môžeme buildovať aplikáciu na mobil.

# **D IIT.SRC**

# eMotion: Emotion Management and Mood Recognition Framework

Zuzana BOBOTOVÁ<sup>1</sup>, Dávid ČERNÁK<sup>2</sup>, Veronika GONDOVÁ<sup>1</sup>  
Tomáš MATLOVIČ<sup>1</sup>, Tomáš PAVLOVIČ<sup>1</sup>, Ján ŠMIHLA<sup>1\*</sup>

*Slovak University of Technology in Bratislava  
Faculty of Informatics and Information Technologies  
Ilkovičova 2, 842 16 Bratislava, Slovakia  
emotion.fiit@gmail.com*

Emotions influence people's everyday decisions, choices, and relationships. During the day we have to deal with many emotions, which can be triggered by various situations or objects. Mood is rather a longer state of the person, which can last for a days or even weeks. Although the emotions are generally short-lasting events, they also affect the overall mood, which have a bigger impact on the quality of our life and health.

A good example of emotion influence is a study from Lerner et. al in [2]. Authors conducted an experiment with 200 participants. The participants were in the role of merchants and their task was to trade with economists. Negative emotions such as disgust or sadness were triggered to participants while they were buying and selling various products. The results of the experiment showed a dramatic impact of the emotions on the economic behavior. If people want to make smart decisions, knowing yourself and management of emotions is almost necessary for them.

In our work we present an application for emotion management and mood recognition. We monitor user emotions in real time using the context data from the mobile phone and the short questionnaires. Collected data are used to compute different characteristics about user, such as level of socialization or physical activity. They are presented using the charts. Furthermore, we propose a method for mood recognition using the machine learning techniques. We can detect user's negative mood and help him to deal with it by suggesting him some of his favourite activities.

There are several approaches of detecting emotions automatically. We can divide them by the in-

put they use: EEG [3], ECG, electrodermal activity, speech and voice intonation, facial expressions, body language and text [5]. Based on how they work and what try to achieve, we can divide them into three categories:

1. Applications collecting data about users that try to find the cause of their negative emotions (e.g Mood Tracker).
2. Applications collecting data about the users that try to get rid of negative emotions by recommending them to do some physical activities, listening music etc. (Self-Help Anxiety Management, Pacifica, Breathe 2 Relax, Happify).
3. Applications that do not collect any data but already expect users to have the negative emotions. They focus on dealing with negative emotions. Pay It Forward is application which recommends good deeds the user could do to become happier.

There are more options how to present results of the collected data to the user and also many ways how to relieve the negative mood. Therefore, we created a questionnaire, which was filled by 492 participants - 289 women and 203 men. The most of them were from the two age categories: people between 20 and 26 years (340 participants) and people between 15 and 19 years (110 participants). We asked people whether they experience negative emotions often, whether and how they would like to solve it. Figure 1 shows a result of the first two questions - people feel stress in their life and they would like to solve it. In the Figure 2 is a result of the key question which strengthen our conviction that charts are the way we should give

---

\* Master study programme in field: <sup>1</sup> Information Systems / <sup>2</sup> Software Engineering

Supervisors: Peter Gašpar, Institute of Informatics, Information Systems and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Dr. Fedor Lehocki, Institute of Robotics and Cybernetics, Faculty of Electrical Engineering and Information Technology STU in Bratislava

a feedback to the user. Our product spans across all three earlier mentioned categories and in addition we introduce completely innovative approach in the field of the emotion management.

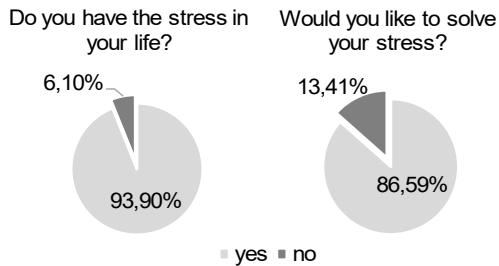


Figure 1. Answers on the questions about stress in people's life

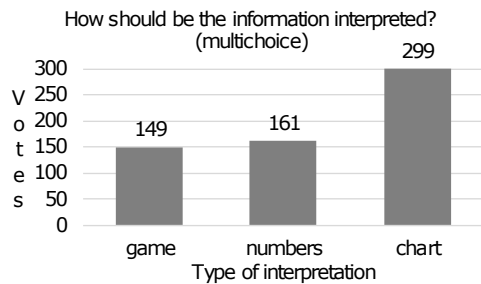


Figure 2. Answers on the question about interpretation of the information

The main idea behind our application is inspired by the research team from the University of Cambridge. They created an application which was intended to regularly collect data about user's behavior [4]. Data included information about the user activity (through GPS and accelerometer), the socialization of user (through a number of contacts and organizing of activities), and the sentiment of communication. We also added biometric data using certificated medical sensors (such as an oximeter and an activity tracker). However, our approach is not limited to those sensors only. We proposed an architecture that enables a simple connection to any new sensor into the mobile application and an automatic processing of this data, as well. Inspired by the fact that emotions and mood are influenced by the personality traits [1] too, we started a cooperation with the team of psychologist.

Data collected by our application are used to determine four different features:

- Level of socialization (frequency of texts/calls, texts sentiment, calendar events, WiFi, BT)
- Mood (questionnaires, model)
- Physical activity (GPS, accelerometer)

- Heart activity (HRV)

User is able to explore this information using our mobile or web application.

Architecture of our framework is composed of two main components. The first part is a web application implemented in Python (Django) that is primary designed for data processing. Real time data are effectively stored using Redis. Web application also provides chart analysis for the user. The second part of the system is a mobile application for Android platform. It is used to collect mobile usage data and also allows pairing and collecting raw data from biometric sensors (AM3, PO3). In addition user can analyse calculated statistics using charts and also get recommended activity. Communication between mobile and web applications is handled by the API.

We have evaluated our approach by conducting two controlled experiments. First one was with approximately 40 participants. During the experiment they were using our mobile application and the data were labelled using the short questionnaires. Second one was only with 3 people where the biometric sensors were also used to label the context data from the mobile phone.

The main goal of our work is to provide people a tool for the management of their emotions. User's mood is detected using the proven scientific researches. Our main contribution is that the application provides personalized feedback to the user. We are also planning to use our application for medical treat by doctors or psychologists. Moreover, there are also other domains where the emotion recognition plays an important role, such as: e-learning, entertainment, marketing or law.

## References

- [1] Buss, A.H., Plomin, R.: *Temperament (PLE: Emotion): Early developing personality traits*. Volume 3. Psychology Press, 2014.
- [2] Lerner, J.S., Keltner, D.: Beyond valence: Toward a model of emotion-specific influences on judgement and choice. *Cognition & Emotion*, 2000, vol. 14, no. 4, pp. 473–493.
- [3] Petrantonakis, P.C., Hadjileontiadis, L.J.: Emotion Recognition From EEG Using Higher Order Crossings. *IEEE Transactions on Information Technology in Biomedicine*, 2010, vol. 14, no. 2, pp. 186–197.
- [4] Rachuri, K.K., Musolesi, M., Mascolo, C., Rentfrow, P.J., Longworth, C., Aucinas, A.: EmotionSense: a mobile phones based adaptive platform for experimental social psychology research. In: *Proceedings of the 12th ACM international conference on Ubiquitous computing*, ACM, 2010, pp. 281–290.
- [5] Shivhare, S.N., Saritha, S.K.: Emotion Detection From Text Documents. *International Journal of Data Mining & Knowledge Management Process*, 2014, vol. 4, no. 6, p. 51.

# **E IIT.SRC poster**

## STRESS → MENTAL BREAKDOWN



memory problems  
loss of concentration



unhappiness  
mood changes



depression  
the idea of death

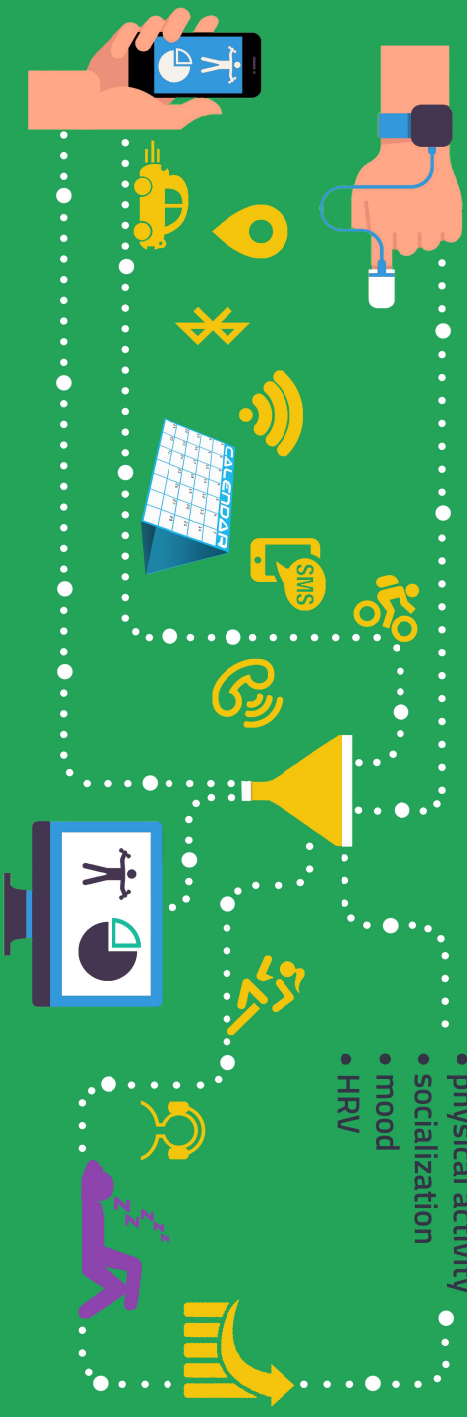
## EMOTIONS

response to the situation  
14 - 55 times per day

## MOODS

combination of emotions  
= emotional state

## HOW IT WORKS?



- computation of:
- physical activity
  - socialization
  - mood
  - HRV

## APPLICATION

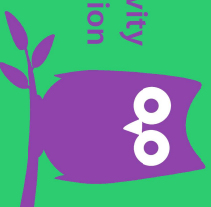
The application interface includes several screens:

- Questionnaire:** A screen with a colorful brain icon and 'my activities' list.
- Questionnaire (Detailed):** A list of activities with checkboxes: Zábava, Práce, Společnost, Sportovní aktivity, Studium, Průmysl, a section for 'Jiné aktivity'.
- emotions:** A donut chart showing mood distribution for '25.4.2017' in 'Ustoupek'.
- emotions (Detailed):** A donut chart with a legend for 'Nadšen', 'Smutek', 'Stres', and 'Uklid'. It also shows 'my activities'.
- Aby si měl den? NESKOR:** A screen with three smiley face icons representing different mood levels.
- Aby si měl den? NESKOR (Detailed):** A bar chart showing mood levels over time for '25.4.2017'.
- Kdo si dělá? NESKOR:** A screen with various emoji icons representing different people or roles.
- Kdo si dělá? NESKOR (Detailed):** A line graph showing mood levels over time for 'Fyzická aktivita 25.4'.
- emotions:** A screen with a 'Socializace' section and a line graph for '25.4'.
- emotions (Detailed):** A line graph showing mood levels over time for 'Fyzická aktivita 25.4' with a data point for '6.4.2017'.

## USABILITY

- Phase 1
- obligatory feedback
  - vizualization
- Phase 2
- voluntary feedback
  - vizualization
  - mood recognition

activity  
recommendation



## Did you know?

"In European Union countries  
27% of the adult population had  
experienced at least one of a series  
of mental disorders in the past  
Year = 83 million people."  
(WHO)

