

Mobilní aplikace pro výuku komplexních motorických dovedností

Bc. Martin Kotěna

Diplomová práce
2019



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Martin Kotěna**
Osobní číslo: **A17240**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Počítačové a komunikační systémy**
Forma studia: **prezenční**

Téma práce: **Mobilní aplikace pro výuku komplexních motorických dovedností**
Téma anglicky: **Mobile application for learning complex motor skills**

Zásady pro vypracování:

1. Provedte průzkum aktuálního stavu mobilních aplikací pro výuku pohybových aktivit s komplexními motorickými dovednostmi, např. tanečních, krasobruslařských nebo gymnastických sestav, choreografií "kata" v bojových sportech atd.
2. Analyzujte požadavky na mobilní aplikaci pro výuku pohybových aktivit a vytvořte specifikaci jejích funkcionalit.
3. Implementujte aplikaci dle specifikace.
4. Aplikaci otestujte s reálnými uživateli a zpracujte jejich připomínky.
5. Výslednou aplikaci zveřejněte na některém z veřejných aplikačních portálů.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. LACKO, L'uboslav. Mistrovství Android. Brno: Computer Press, 2017, 647 s. ISBN 978-80-251-4875-4.
2. LACKO, L'uboslav. Vývoj aplikací pro Android. Brno: Computer Press, 2015, 472 s. ISBN 978-80-251-4347-6.
3. SMYTH, Neil. Android Studio 3.0 Development Essentials – Android 8 Edition. Payload Media, 2017, 647 s. ISBN 1977540090.
4. DVOŘÁKOVÁ, Hana, Michaela KUKAČKOVÁ a Martina LIETAVCOVÁ. Rozvíjíme dovednosti hrubé a jemné motoriky dětí. 2. vydání. RAABE, 2017, 647 s. ISBN 978-80-7496-187-8.
5. WEI, Jason. Android Database Programming. Packt Publishing, 2012, 647 s. ISBN 9781849518123.

Vedoucí diplomové práce:

Ing. Tomáš Dulík, Ph.D.

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

30. listopadu 2018

Termín odevzdání diplomové práce:

17. května 2019

Ve Zlíně dne 10. prosince 2018

doc. Mgr. Milan Adámek, Ph.D.
děkan



Ing. Miroslav Matýsek, Ph.D.
ředitel ústavu

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 20.5 2019

Martin Kotěna, v. r.

ABSTRAKT

Předložená diplomová práce se skládá ze dvou částí. Teoretická část se zabývá popisem současného sensorového vybavení chytrých telefonů, a to jak z hardwarového, tak softwarového hlediska. Jsou zde také zmíněny nejznámější aplikace, které tyto senzory využívají v praxi. Praktická část této práce se zabývá vývojem mobilní aplikace využívající sensorového vybavení zařízení s operačním systémem Android pro usnadnění tréninku pohybových aktivit pomocí dostupných videí. Jsou zde rozebrány způsoby získávání a zpracovávání dat ze senzorů, tvorba prezentačních videí a také jsou zde detailně popsány principy funkce komunikačních kanálů mezi uživateli ve vytvořených skupinách, vkládání fotek a jejich sdílení.

Klíčová slova: Android, Firebase, Gadgety, chat, chytré telefony, senzory,

ABSTRACT

The submitted thesis consists of two parts. The theoretical part deals with the description of current sensor equipment in smartphones, both from hardware and software point of view. Several gadgets and applications which use these sensors are also mentioned. The practical part of this thesis deals with the development of a mobile application using sensor equipment of a device running Android in order to facilitate the training of physical activities using available videos. Methods of obtaining and processing sensor data are described in detail, as well as creating presentation videos, principles of communication between people in user-created groups and sharing photos.

Keywords: Android, Firebase, Gadgets, chat, smartphones, sensors,

Za odborné vedení, připomínky a návrhy při vypracování této diplomové práce bych chtěl poděkovat svému vedoucímu diplomové práce Ing. Tomáši Dulíkovi, Ph.D.

Zároveň bych chtěl také poděkovat Ing. Radku Valovi, Ph.D. za podnětné připomínky k praktické části této práce.

„Nemůžete se jednoduše zeptat zákazníků, co chtějí, a pak se to pokusit sestavit. V době, kdy to budete mít hotové, už budou chtít něco jiného.“

- Steve Jobs

Prohlašuji, že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 SENZORY PRO VÝUKU POHYBOVÝCH AKTIVIT	11
1.1 SENZORY POHYBU	12
1.1.1 Akcelerometr	12
1.1.2 Gyroskop	13
1.2 SENZORY POLOHY	14
1.2.1 Magnetometr	14
1.2.2 GPS (Global Positioning System).....	16
1.2.3 Detektor chůze	18
1.3 ENVIRONMENTÁLNÍ SENZORY	18
1.3.1 Senzor tlaku.....	18
1.3.2 Senzor teploty.....	18
1.3.3 Senzor osvitu.....	19
1.3.4 Senzor vlhkosti vzduchu	19
1.3.5 Některé další senzory	20
1.3.5.1 Senzor přiblížení	20
1.3.5.2 Gravitační senzor	21
1.3.5.3 Snímač otisků prstů.....	21
1.3.5.4 Senzor srdečního tepu.....	21
2 ANDROID APLIKACE	23
2.1 VYHODNOCOVÁNÍ DAT	23
2.2 PŘÍKLADY NĚKTERÝCH APLIKACÍ.....	23
2.2.1 Smart Thermometer	23
2.2.2 Senzory toolbox	24
3 APLIKACE A GADGETY V OBLASTI VÝUKY POHYBOVÝCH AKTIVIT	25
3.1 FITNESS NÁRAMKY	25
3.2 CHYTRÉ BOTY A VLOŽKY DO BOT	25
3.3 TANEČNÍ PODLOŽKY	26
3.4 HERNÍ KONZOLE SE SNÍMAČI POHYBU	27
3.5 MOTION CAPTURE SYSTÉMY	28
3.6 NĚKTERÉ VOLNĚ DOSTUPNÉ MOBILNÍ APLIKACE	29
3.6.1 Karate WKF	29
3.6.2 Taekwondo WTF	30
3.6.3 Learn Martial Arts	31
3.6.4 Martial Arts – Training and workouts.....	32
3.6.5 Shrnutí	32
II PRAKTICKÁ ČÁST	33
4 POŽADAVKY NA APLIKACI	34
5 NÁVRH APLIKACE	35
5.1 POŽADAVKY NA FUNKCE APLIKACE	35
5.1.1 Use Case Diagram.....	35

5.2	NEFUNKČNÍ POŽADAVKY	36
5.3	PROGRAMOVACÍ JAZYKY.....	37
5.3.1	Java.....	37
5.3.2	Kotlin.....	37
5.3.3	JavaScript	37
5.4	ZABEZPEČENÍ HESEL – SHA-256 (SECURE HASH ALGORITHM)	38
6	FIREBASE	39
6.1	STORAGE.....	39
6.2	AUTHENTICATION	39
6.3	DATABÁZE	40
6.4	CLOUD MESSAGING	40
6.5	CLOUD FUNCTIONS	41
7	IMPLEMENTACE	42
7.1	PŘIHLÁŠENÍ.....	42
7.2	CHATOVÁNÍ.....	44
7.2.1	RecyclerView	44
7.3	PŘIDÁNÍ FOTOGRAFIE	47
7.4	MÉDIA.....	48
7.5	STAŽENÍ VÝUKOVÝCH VIDEÍ.....	49
7.6	VÝUKOVÁ AKTIVITA	50
8	PREZENTACE VYTVOŘENÉ APLIKACE.....	52
8.1	PŘIHLAŠOVACÍ AKTIVITA	52
8.2	PROFILOVÁ AKTIVITA.....	53
8.3	CHATOVACÍ AKTIVITA.....	54
8.4	MÉDIA.....	55
8.5	NASTAVENÍ	56
8.6	VÝUKOVÁ AKTIVITA	57
9	NÁVOD K OBSLUZE	59
9.1	ON-LINE VERZE	59
9.2	OFF-LINE VERZE.....	59
10	TESTOVÁNÍ APLIKACE	60
	ZÁVĚR	62
	SEZNAM POUŽITÉ LITERATURY.....	63
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	66
	SEZNAM OBRÁZKŮ	67
	SEZNAM KÓDŮ	69
	SEZNAM TABULEK.....	70
	SEZNAM PŘÍLOH.....	71

ÚVOD

Digitální technologie se rapidně stávají zcela přirozenou součástí lidského života, což platí zejména u mladé generace. Mezi hlavní výhody těchto technologií patří konektivita, takže uživatel je v neustálém kontaktu s ostatními uživateli, a téměř neomezený přístup k informacím. Díky těmto přednostem si tyto technologie získaly právoplatné místo i ve volném čase uživatelů.

Ačkoliv mají tyto technologie i své negativní stránky (horší socializace, pasivní trávení volného času), dochází k jejich rapidnímu rozvoji. Jednou ze známých stránek tohoto rozvoje jsou netradiční sportovní aktivity, při kterých jsou hojně využívána chytrá přenosná zařízení. Při těchto aktivitách lze využívat mnohé senzory v zařízeních (fotoaparát, kamera). Další velkou výhodou je také možnost propojení těchto chytrých zařízení, kdy se více uživatelů může podílet na stejné aktivitě.

Předložená diplomová práce pojednává o praktickém využití jednotlivých senzorů v mobilních telefonech a jiných chytrých zařízeních pomocí mobilní aplikace. Díky této aplikaci je možné usnadnit uživatelům naučení nebo procvičení sestav v Taekwondo, karate atd. Aplikace také umožňuje komunikaci mezi uživateli ve vytvořených skupinách, vkládání fotek a jejich sdílení s ostatními.

Teoretická část této práce se zabývá zejména popisem požadavků na danou aplikaci a ukázkou podobných současně dostupných aplikací a chytré elektroniky (např. Fitness náramky). V této části práce jsou také podrobněji popsány jednotlivé senzory nutné ke správné funkci těchto aplikací.

V praktické části je podrobně popsána vytvořená aplikace. Jsou zde popsány jednotlivé použité senzory, způsob získávání dat, jejich vyhodnocování a princip komunikace mezi uživateli ve skupinách.

I. TEORETICKÁ ČÁST

1 SENZORY PRO VÝUKU POHYBOVÝCH AKTIVIT

Mobilní telefony dnes obsahují velké množství sensorů a v mnoha případech mají smyslů víc než lidská bytost. [1]

V širším slova smyslu lze považovat za senzory jakýkoliv zdroj informací, který se předává do řídicí jednotky.

Senzory neboli čidla jsou v dnešní době v základní výbavě chytrých telefonů. Tyto senzory jsou využity nejen při hraní různých her v mobilních telefonech, ale také při automatickém nastavování jasu nebo automatickém uzamykání obrazovky telefonu nebo smartphonu.

Majitelé telefonů nebo smartphonů ani netuší, že se senzory v zařízení vyskytují a k čemu slouží. V dnešní době jsou však senzory neodmyslitelnou součástí každého mobilního zařízení a určitě bude v budoucnu stále přibývat mnoho nových sensorů, které budou usnadňovat ovládání přístrojů. [2,3]

Veškeré senzory jsou specializované komponenty, které slouží k měření konkrétních fyzikálních veličin. Pokud jde o chytré telefony, tak ty mají tyto komponenty zabudované ve svém hardwaru, a to jako samostatné nebo integrované součástky. [2,3]

Senzory se dělí do tří základních kategorií: [2,3]

- **Pohybové** (motion) – měří velikost akceleračních a rotačních sil. Využívají k tomu fyzikální principy a rovnice.
- **Polohové** (position) – poskytují informace o poloze a orientaci zařízení. Využívají geomagnetické údaje a údaje z družic GPS.
- **Environmentální** – měří fyzikální parametry okolního prostředí – nejčastěji měřeným parametrem je intenzita osvětlení, teplota, tlak nebo vlhkost

Ne každé zařízení má však všechny zmíněné typy sensorů. Níže jsou uvedeny nejběžnější senzory v mobilních telefonech:[4]

- **Senzor přiblížení** – měří vzdálenost telefonu od ucha a využívá se k vypnutí dotykové obrazovky během přiložení telefonu k uchu
- **Gyroskop** – měří pootočení zařízení v trojrozměrném prostoru
- **Akcelerometr** – měří pohyb zařízení v trojrozměrném prostoru
- **Geomagnetický senzor** – umožňuje funkci digitálního kompasu tím, že dokáže určit magnetické póly Země

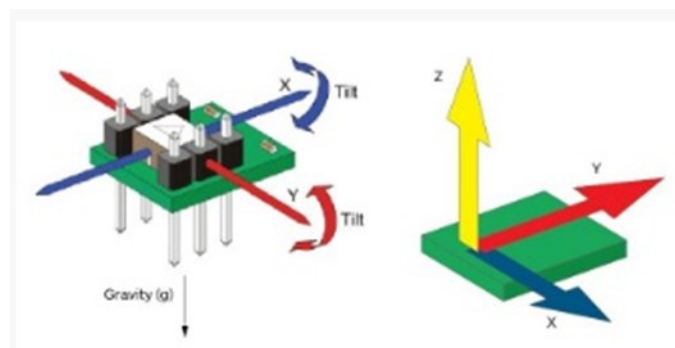
- **Tlakoměr** – měří atmosférický tlak v aktuálním místě a slouží k vytvoření aplikace typu výškoměr
- **Senzor Hallova jevu** – u některých typů slouží ke zjištění, zda je kryt mobilu otevřen nebo zavřen
- **Senzor osvit** – měří intenzitu okolního světla a automaticky přizpůsobuje jas displeje
- **Snímač gest** – jde o infračervené pole, které snímá gesta uživatele nad displejem

Senzory však mají svá fyzická omezení, díky kterým je určení skutečného pohybu a orientace občas složité. Firmware přístroje proto dokáže propojit získávaná data z několika senzorů současně a analyzovat je jako celek. Tímto způsobem je možné získat komplexní informaci a výrazně se zpřesní odhad skutečné, dynamicky se měnící veličiny. Například údaje z akcelerometru jsou odvozeny ze setrvačnosti, která vyplývá z pohybu zařízení. Gyroskop měří rychlost otáčení, ale ne polohu. Proto jsou v operačním systému zařízení implementovány složité geometrické výpočty. Tyto výpočty jsou nutné ke zjištění polohy orientace a pohybu ze surových dat jednotlivých senzorů. [3,5]

1.1 Senzory pohybu

1.1.1 Akcelerometr

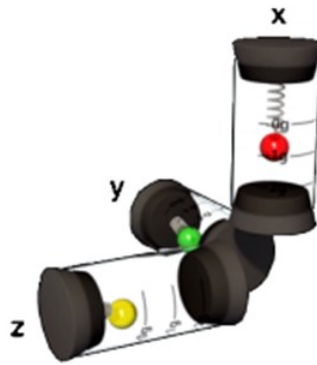
Jde o senzor, který měří setrvačný pohyb v různých osách. Funkce akcelerometru se dá také použít k nahrazení klasického krokoměru. Senzor využívá gravitaci jako referenční osu a vzhledem k této ose zaznamenává zrychlení chytrého zařízení. Tímto způsobem dojde k určení relativní polohy v prostoru vůči Zemi. Návrátové hodnoty jsou v m/s^2 . [2,6]



Obrázek 1: Akcelerometr [6]

Součástka je navržena tak, aby při změně rychlosti změřila určité vibrace, které jsou spojené s tímto pohybem. Ke zjištění těchto vibrací akcelerometr využívá mikroskopické krystalky,

na kterých se při působení těchto vibrací generuje napětí odpovídající určitému zrychlení. Tomuto jevu se říká piezoelektrický jev a pomocí něj lze určit směr gravitace a tím natočení přístroje. Jde o vlastnost, které se využívá při otáčení obrazovky telefonu. [3,6]



Obrázek 2: Osově uspořádání [5]

Vlastnosti akcelerometru našly uplatnění v mnoha aplikacích, ale nejvíce jsou využity při hraní her. Jde však o jednoduché hry, kde je nutno natáčet mobil tak, aby se např. kulička udržela na vytyčené dráze (Skyball). Pokud jde o pokročilejší aplikaci, tak lze otáčet mobilem jako volantem a tím řídit automobil nebo dokonce naklánět motorku na dráze. [3,6]

Senzor se používá v celé řadě průmyslových odvětví - automobilový průmysl (měření zrychlení), stavební průmysl (měření vibrací) nebo environmentální vědy (seismické aktivity, pohyby podzemní vody či lávy).

1.1.2 Gyroskop

Tento senzor bývá velmi často v mobilních zařízeních připojován k akcelerometru. Podobně jako akcelerometr slouží k určení náklonu a natočení mobilního zařízení. Gyroskop měří úhlovou rychlost.

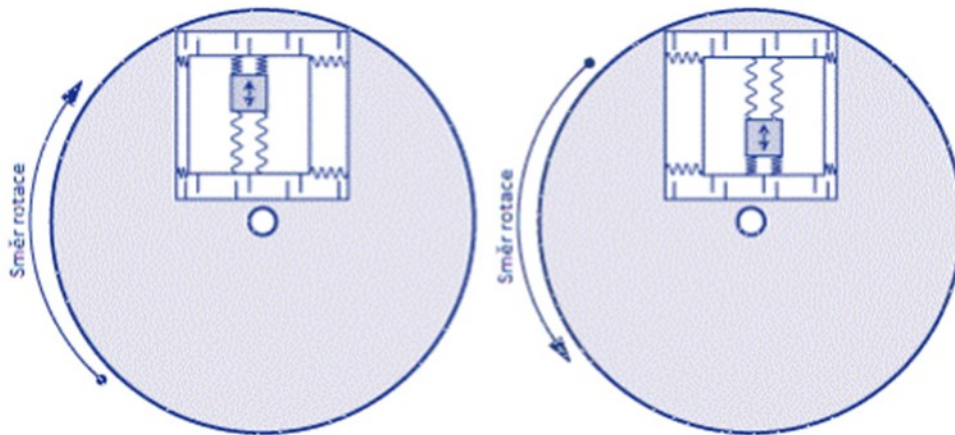
Pokud se zkombinují oba senzory, tak akcelerometr měří zrychlení a gyroskop úhlovou rychlost. Kombinací těchto dvou senzorů tedy dojde k přesnějšímu určení skutečného pohybu mobilního zařízení v prostoru. [3,6]



Obrázek 3: Gyroskop [1]

Gyroskop poskytuje mobilnímu zařízení data o natočení telefonu v prostoru (ve třech osách) vůči výchozímu stavu. Pokud se zařízení nehýbe, tak gyroskop neposkytuje žádná data, ale jakmile se zařízení otáčí, gyroskop poskytuje hodnoty o směru a rychlosti otáčení.

Gyroskop by se dal popsat jako setrvačnick tvořený těžkým kolem. Kolo se otáčí v ložiscích s velmi nepatrným třením. Setrvačnick si po roztočení zachovává polohu osy své rotace v prostoru a dokáže tak určit svou orientaci.[3]



Obrázek 4: Zjednodušené schéma gyroskopu [6]

Rozmach optických gyroskopů (nepoužívají žádné pohyblivé části) nastává v 70. letech, kdy se objevila potřeba po lepších balistických raketách, které by dokázaly letět k cíli a přesně ho zasáhnout bez komunikace s operátorem. Použití gyroskopů pro vojenské účely je však mnohem širší a našli bychom je například i v torpédech.[3]

V civilní oblasti se gyroskopy hojně využívají u letadel, kde je na jejich základě postaven tzv. umělý horizont, ale používají se i například pro navigaci lodí, satelitů a například i na vesmírné stanici ISS, kde čtveřice gyroskopů umožňuje pohyb solárních panelů. [3,7]

1.2 Senzory polohy

1.2.1 Magnetometr

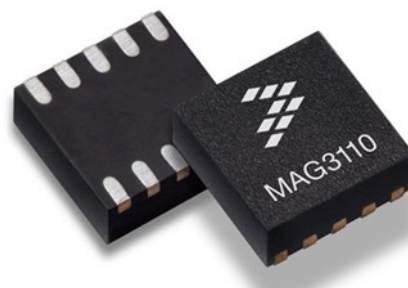
Tento senzor patří mezi významné senzory v mobilním zařízení. Jedná se o magnetický senzor a také se mu říká kompas. Senzor bývá využíván k určení světových stran a je využíván GPS navigací. [8,9]

Snímač detekuje magnetické pole Země a tím dokáže určit světové strany. Při jeho použití dochází k rychlejšímu určení polohy. [8,9]



Obrázek 5: Kalibrace a fungování kompasu [1]

Magnetometr je součástka, která využívá polovodičový senzor na měření magnetického pole. Většinou je tvořen z magnetizovaným kouskem plechu, u kterého víme, jaký má za daného proudu odpor při nulovém vlivu externího magnetického pole. U tohoto kousku plechu dochází vlivem magnetického pole ke změně odporu. Tato změna je vyhodnocována elektrickým obvodem. [10]



Obrázek 6: Magnetometr pro mobilní zařízení [8]

Magnetometry se dělí na skalární a vektorové. Skalární měří pouze intenzitu magnetického pole a vektorové měří i jeho směr. [10]

V současnosti se používají pro měření zemského magnetického pole a různých magnetických anomálií, pro výzkum polární záře, pro archeologii, protože dokáží odhalit i skryté neobvyklé útvary v hloubce desítek metrů, v kosmickém výzkumu, v umělých družicích na výzkum magnetosféry. Používají se i pro vojenské účely – dokáží detekovat ponorky spolehlivěji než sonar. [2,10]

Pro mobilní zařízení jde o součástku velikosti několika milimetrů a jsou založeny na dvou nebo třech snímačích magnetického pole. Svá data poskytují mikroprocesoru a ten z těchto dat vypočítá správnou polohu za pomoci trigonometrie. [10]

1.2.2 GPS (Global Positioning System)

Jde o pasivní systém pro stanovení polohy a času na Zemi. Tento systém je schopen signály poskytovat 24 hodin denně, a to kdekoliv na zemském povrchu i v jeho přilehlém prostoru. Tyto signály zpracovávají přijímače GPS a určují tak polohu v prostoru a také přesný čas. [9,12]

Systém GPS se skládá ze tří základních segmentů:

- Kosmický segment
- Řídící segment
- Uživatelský segment

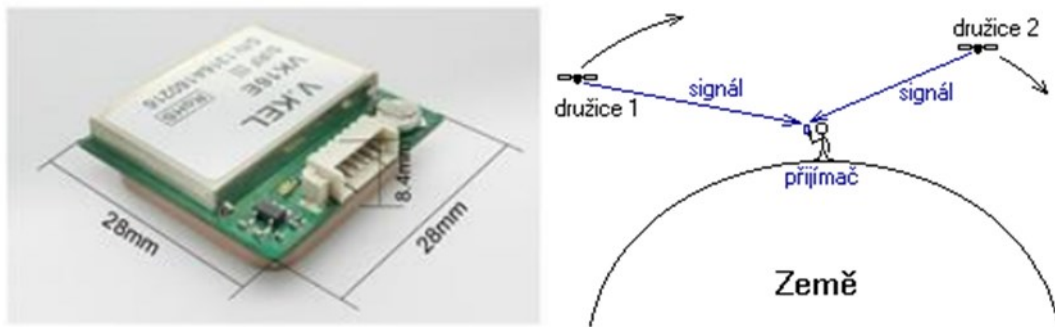
Kosmický segment je tvořen jednotlivými satelity na oběžných drahách Země. Družice obíhají ve výšce cca 20 200 km, doba oběhu je přibližně 12 hodin, a tím je zajištěno, že prakticky všude a v jakýkoliv okamžik jsou nad obzorem minimálně 4 viditelné družice. Aby mohla být určena poloha zařízení v prostoru, je nutné přijmout signály ze 4 družic – a to proto, že jsou tři neznámé souřadnice pro určení polohy v prostoru a čtvrtou neznámou je čas. Každá viditelná družice navíc nám zajistí zlepšení konfigurace a tím i výsledky měření. [9,13]

Řídící segment je tvořen monitorovacími stanicemi po celém světě a hlavní řídicí stanicí (MCS) v Colorado Springs. Monitorovací stanice neustále provádí sběr dat z družic a ty předává do MCS. V hlavní řídicí stanici jsou data zpracována a jsou vypočteny přesné údaje o oběžných drahách a korekce času. Po zpracování jsou tyto data zpětně přenesena pozemními anténami do satelitů. Satelity je potom v rámci navigační zprávy vysílají a data jsou přijímány GPS přijímači. [9,12]

Uživatelský segment je tvořen širokou paletou GPS přístrojů. Tyto přístroje poskytují údaje o poloze, rychlosti a čase uživatelům v nejrůznějších aplikacích. [9,13]

GPS má však i nevýhody. Především jde o nemožnost prostupu signálu podzemím, stěnami budov nebo hustou vegetací, což souvisí s prostupností signálu náročným terénem – GPS vyžaduje přímou viditelnost mezi GPS přijímačem a satelity. [13]

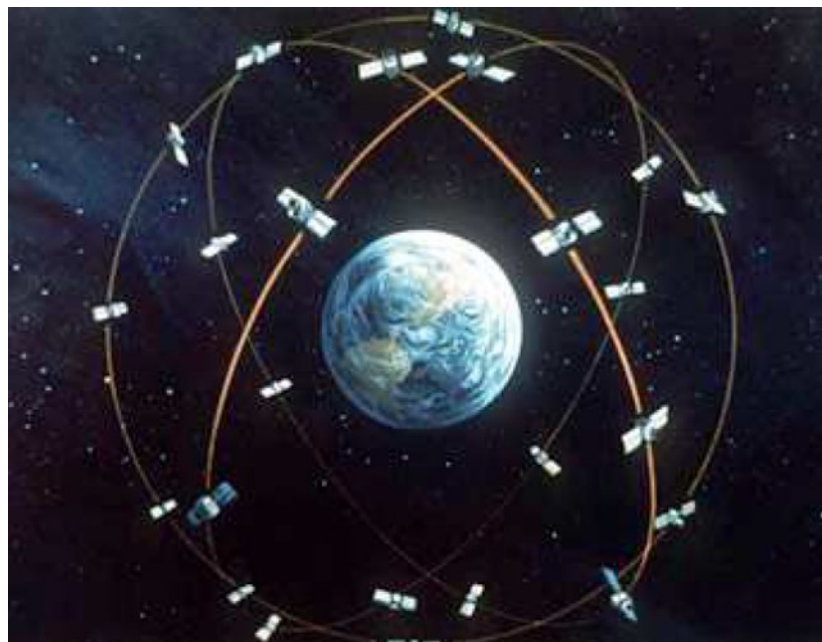
Nejvíce se GPS využívá ve formě pozemní navigace. Dnes jsou již téměř všechny vozy vybaveny touto navigací.



Obrázek 7: GPS čip, funkce GPS [1]

Využití GPS je i ve sledování vozidel v reálném čase (on-line), které slouží pro řízení a kontrolu pohybu např. policejních či záchranářských týmů. Další možností využití je tzv. pasivní sledování vozidel (firemního vozového parku) se záznamem trasy do paměti zařízení a s dostatečnými kontrolními možnostmi, s výstupy pro knihy jízd atd. [9,12]

Pomocí měření s GPS se v aplikacích ukládají zeměpisné souřadnice, které se mohou potom zobrazit na mapě. Souřadnice jsou zaznamenány v trojrozměrném prostoru a obsahují i informace o nadmořské výšce. V případě konkrétního využití např. při běhu, dokáže GPS přijímač skrze aplikaci zobrazit celkový profil tratě. [9,13]



Obrázek 8: Schéma rozmístění družic [12]

1.2.3 Detektor chůze

Jedná se o mechanický senzor na počítání kroků. Krokoměř používá ke své činnosti vestavěný senzor telefonu, takže není třeba GPS – čímž se šetří baterie. Přesnost krokoměru se liší nejen podle druhu přístroje, ale také podle nastavení – např. délka kroku. Nejprecizněji přístroj měří přiměřeně rychlou chůzi na rovném povrchu, a to při upevnění přístroje v optimální pozici. [4]

K přesnému měření vzdálenosti je potřeba kombinace senzorů, protože využití samostatného akcelerometru nám neposkytuje přesná data. Počítání kroků pomocí akcelerometru probíhá tak, že při silnějším nárazu je akcelerometrem indikován krok. [4,9]

1.3 Environmentální senzory

Environmentální podmínky jsou podmínky, které mají zásadní vliv na naši pohodu, pohodlí a produktivitu. Tento druh senzorů poskytuje podrobné a spolehlivé údaje o klíčových environmentálních parametrech, jako je např. vlhkost, teplota, těkavé organické sloučeniny (VOC), částice (PM_{2,5}) a CO₂.

1.3.1 Senzor tlaku

Jedná se o senzor tlaku vzduchu nebo též barometr. Tento senzor měří aktuální atmosférický tlak, který lze využít nejen v meteorologických aplikacích, ale bývá využíván také k přesnějšímu určení polohy, konkrétně nadmořské výšky. [4]



Obrázek 9: Senzor tlaku [1]

1.3.2 Senzor teploty

Jde o senzor určený k měření teploty, který se také nazývá termometr nebo prostě teploměr. Tento senzor měří aktuální teplotu okolí. Senzor se dnes vyskytuje skoro ve všech třídách chytrých zařízení i ve větším množství, protože senzor monitoruje i teplotu interních

hardwarových součástí a baterii. Bohužel však, s ohledem na zahřívání hardwaru během jeho činnosti, nejsou naměřené výsledky zcela spolehlivé. Čidlo při zjištění extrémně vysoké teploty dokáže telefon automaticky vypnout [4,5]

1.3.3 Senzor osvitu

Tento senzor analyzuje úroveň okolního osvětlení a automaticky nastavuje úroveň podsvícení displeje. To pomáhá jednak zlepšit čitelnost, tak i šetřit energii baterie.

Nejčastěji zde bývá použit fotorezistor, který pracuje na principu fotoelektrického jevu. Jednoduše řečeno, čím více světla dopadá na povrch součástky, tím více tato součástka uvolní elektronů a tím více bude vodivá. Podle této naměřené vodivosti řídicí obvod určí dané světelné podmínky a nastaví jas na displeji mobilního zařízení. [1,4]

Většina současných senzorů používá kombinaci několika fotodiod, kde každá je citlivá na jinou část světelného spektra a výsledek je dopočítán. Tím je dáno poměrně přesné měření úrovně světla. [1, 4]



Obrázek 10: Senzor intenzity okolního osvětlení [1]

S těmito senzory se však nesetkáváme jen v mobilních telefonech, ale i v televizorech, notebookech či monitorech.

1.3.4 Senzor vlhkosti vzduchu

Jde o vlhkoměr, který však není ve výbavě smartphonů příliš častý. Dá se říci, že se objevil jen v několika modelech – například v Samsungu Galaxy S4. Senzor slouží k monitorování vlhkosti okolního prostředí a jeho výstupem je potom vyjádření pomocí stupnice, jestli se uživatel nachází v nastaveném „komfortním rozpětí“ hodnot. Měří vlhkost vzduchu, což lze využít v různých meteorologických a sportovních aplikacích. [4]

1.3.5 Některé další senzory

1.3.5.1 Senzor přiblížení

Jedná se o standardní součást každého mobilního zařízení s dotykovým displejem. Nejčastěji je využíván při telefonních hovorech.

Když dojde při telefonování k přiblížení telefonu k uchu, tak tento senzor zaznamená tuto skutečnost a dochází k automatickému uzamčení displeje, aby nedošlo k nechtěné aktivaci některých funkcí nebo dokonce k ukončení hovoru. [1, 4]

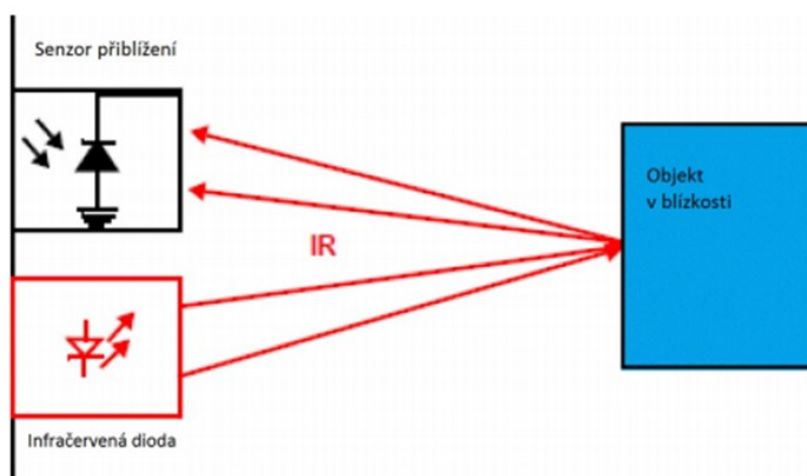
Jedná se o bezkontaktní snímač, který zaznamenává přítomnost objektů v blízkosti bez nutnosti fyzického kontaktu. Skládá se z infradiody a infradetektoru. [4]

Senzor vysílá svazek elektromagnetického záření – nejčastěji infračervené. Pomocí těchto paprsků zjišťuje změny v odraženém signálu. Podle těchto změn senzor vyhodnotí, že je objekt v dostatečné blízkosti a vyhodnotí jej jako přítomnost objektu a provede aktivaci. [1]

S tímto senzorem se setkáváme také u automobilů v podobě parkovacího senzoru, který vyhodnocuje vzdálenost překážky.

V mobilních telefonech jsou používány optické senzory. Tyto senzory nejsou nejlevnější, ale dokážou detekovat přítomnost lidského těla na vzdálenost asi 5 cm. Někdy je senzor doplněn o senzor gest, který indikuje různé pohyby dlaní např. k utlumení hudby nebo k bezdotykovému přijetí hovoru. [1, 4]

Tyto senzory jsou použity i v bezpečnostních zařízeních, kde hlídají, zda jsou např. okna a dveře zavřené.



Obrázek 11: Senzor přiblížení [6]

1.3.5.2 Gravitační senzor

Jako takový měří ve třech osách pouze gravitační sílu. Má podobnou funkci jako akcelerometr, a pokud je zařízení v klidu, tak by měl dát i stejné hodnoty. Rozdíl je v tom, že akcelerometr vrací jako hodnoty součet všech sil, které jsou aplikované na zařízení a gravitační senzor vrací pouze vliv gravitace. [4]

1.3.5.3 Snímač otisků prstů

Tento senzor se v chytrých zařízeních objevuje stále častěji. Existují dva typy. Pokud je chytré zařízení vybaveno prvním typem, tak stačí prst pouze přiložit. U druhého typu je nutné přes senzor přejet prstem. [1].

Otiskem se může:

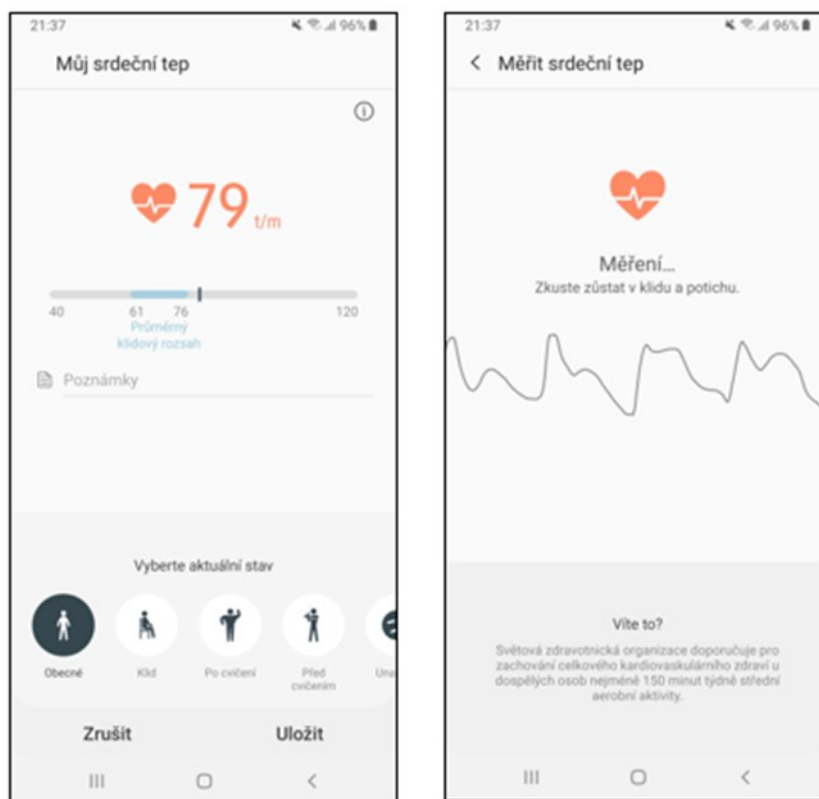
- Primární účel je větší bezpečnost chytrého zařízení
- Odemknout chytré zařízení
- Zaplatit – místo PIN kódu
- Zabezpečit soubory



Obrázek 12: Snímač otisků prstů
[1]

1.3.5.4 Senzor srdečního tepu

Senzor se skládá z červené LED diody a senzoru pulzu. Pro měření je třeba přiložit k senzoru špičku prstu. Dioda prosvítí špičku prstu, senzor pulzu měří následně pohyb červených krvinek a zobrazí srdeční tep. Celkové měření probíhá přes aplikaci „S Health“.



Obrázek 13: Ukázka měření tepu

2 ANDROID APLIKACE

Všechny senzory, které jsou dostupné v Android aplikaci, vracejí naměřené výsledky jako pole hodnot. Jedná se o pole hodnot typu „float“ s přesností 32 bitů. Takové řešení je použito pro účel jednotného způsobu práce se senzory.

2.1 Vyhodnocování dat

V Android aplikaci existují čtyři způsoby, pomocí kterých mohou senzory vytvářet události, tzv. SensorsEvent-y. V rámci těchto událostí dochází k vrácení naměřených hodnot. Režim získávání dat má každý senzor přesně daný a nelze ho nijak měnit.

Existující způsoby: [10]

- **Continuous** (průběžné) Eventy – jsou tvořeny průběžně, vždy po uplynutí konstantního časového úseku. Tento způsob využívá např. akcelerometr, gyroskop, senzor tlaku, gravitační senzor
- **On-change** Eventy (hlásící změnu) – generovány pouze v případě, že dojde ke změně měřicí hodnoty – senzor okolní teploty, senzor dopadajícího světla, proximity senzor, senzor srdečního tepu, krokoměr a senzor relativní vlhkosti
- **One-shot** (jednorázové) – režim využíván pouze senzorem pohybu
- **Speciální režim** – pouze senzorem detekce kroku

Frekvence pro získávání dat je ovlivněna zvoleným režimem.

2.2 Příklady některých aplikací

2.2.1 Smart Thermometer

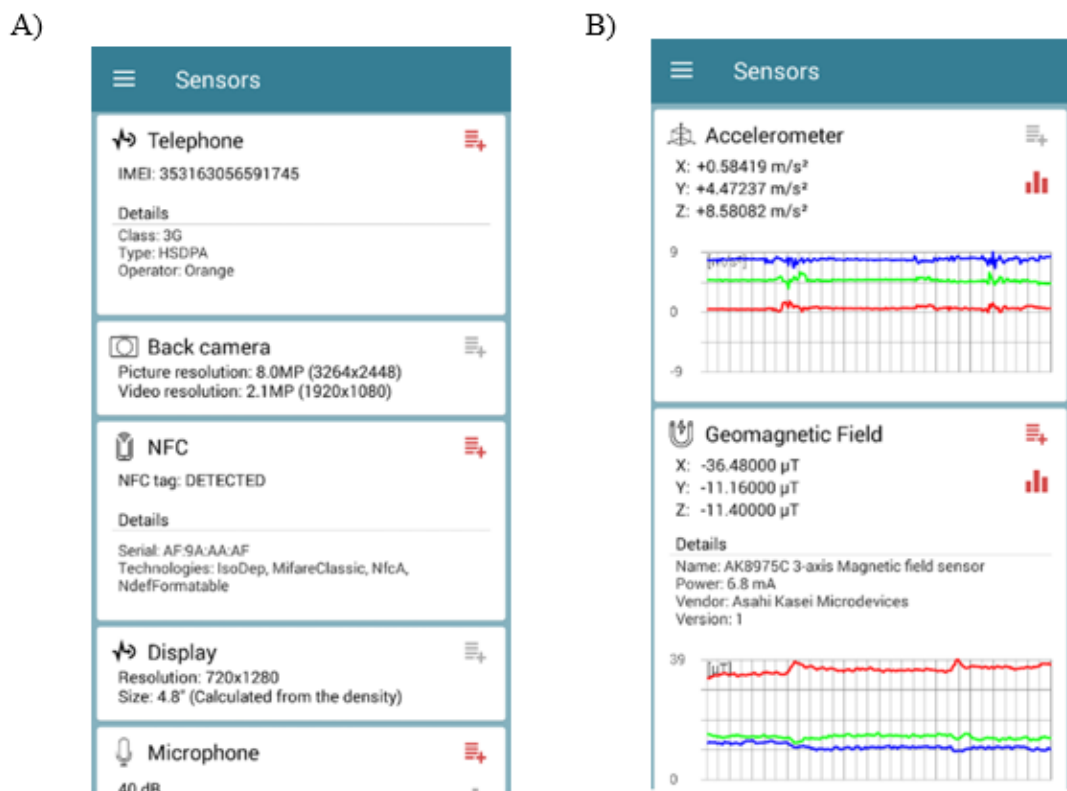
Aplikace, která využívá vnitřní senzor pro měření teploty v mobilním zařízení. Tato aplikace se nachází na Google Play Store.

Aplikace dokáže zobrazit teplotu v okolí mobilního zařízení, ale podmínkou je, že mobilní zařízení musí být vybaveno tímto senzorem, jinak zobrazuje teplotu z internetu, jako každá jiná aplikace na zobrazování počasí. [1]

2.2.2 Senzory toolbox

Absolutní all-in-one diagnostický nástroj. Umožňuje zjistit vše o stavu mobilního zařízení včetně všech dostupných čidel.

Na *Obrázek 14 A* jsou zobrazeny některé dostupné senzory v mobilním zařízení a na *Obrázek 14 B* je vidět ukázka zobrazení dat např. z akcelerometru a geomagnetického senzoru včetně grafu. [11]



Obrázek 14: Dostupné senzory (A) a zobrazení dat (B)

3 APLIKACE A GADGETY V OBLASTI VÝUKY POHYBOVÝCH AKTIVIT

3.1 Fitness náramky

Fitness náramky nabízí svým uživatelům víceméně nepřetržitý monitoring pomocí různých senzorů, jako je například senzor srdečního tepu nebo krokoměr.

Krokoměr je vlastně malé fyzické kyvadélko, které reaguje na houpavou chůzi nebo běh. Software ale začíná počítat až několik vteřin po startu, aby došlo k eliminaci snímání pohybů neznamenajících chůzi. Tento mechanismus po každém došlápnutí přičte krok a zobrazí jej na displeji. [15]

Mezi další funkce zařízení může patřit 3D Akcelerometr, který slouží k přesnému zaznamenávání počtu kroků, dále pak k měření rychlosti a také zaznamenává celkovou překonanou vzdálenost. [15,16]

Vnitřní software náramku pak dokáže ušlé kroky převést na spotřebované kalorie včetně uložení statistik do paměti.

3.2 Chytré boty a vložky do bot

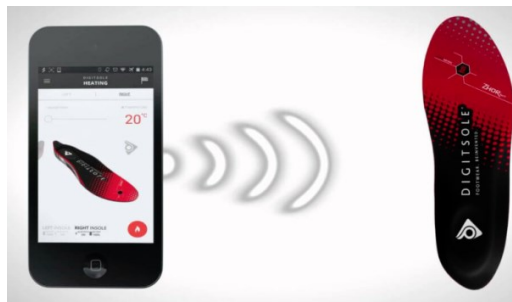
Chytré boty obsahují drobný čip, uložený obvykle pod patou, který v reálném čase zaznamenává množství údajů, jako je například uběhnutá vzdálenost, rychlost běhu/chůze a spálené kalorie. Čip se dá spárovat s dalším zařízením, například s aplikací na telefonu, což umožňuje práci se zaslánými daty či jejich vizualizaci v čase. [17]

Chytré vložky do bot obsahují senzory tlaku, díky kterým je možné analyzovat způsoby došlapu člověka. Tato data pak mohou sloužit ke korekci došlapu (ortopedické vložky). Data je možné získat podobně jako u chytrých bot, tj spárováním s aplikací na mobilu nebo USB přípojkou. [17]

Aplikace dostupné k tomuto účelu bývají poskytovány zdarma k zakoupenému páru vložek.

Mezi obvyklé funkce patří měření počtu kroků, síla a způsob dopadu chodidla, délka kroku, doba kroku a stabilita postoje člověka. Všechna tato data dokáže poskytnutá aplikace analyzovat a vyslovit verdikt o kvalitě běžeckého stylu. Uživatelé se případně objeví varování s doporučením, jak svůj běžecký styl zlepšit.

Každý typ vložek má potom své specifické nadstandartní funkce jako např. integrované vytápění nebo GPS.

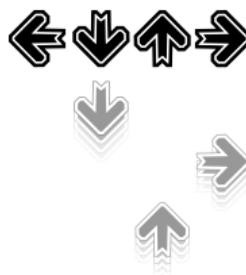


Obrázek 15: Chytré vložky do bot [17]

3.3 Taneční podložky

Jde o senzorovou podložku, na které jsou vyobrazeny čtverce se šípkami. Taneční podložka se často prodává jako příslušenství ke hrám či herním konzolím.

Program či hra na obrazovce přehrává předem danou sekvenci čtverců se šípkami, které musí uživatel na podložce ve správný čas a ve správném pořadí zmáčknout nohou. Rychlost a náročnost tanečního programu přitom udává vybraná skladba.



Obrázek 16: Náhled šipek [18]

Aplikace vyhodnocuje, kdy a jestli došlo ke stisknutí správného čtverce na podložce a podle toho uživatele ohodnotí body.



Obrázek 17: Taneční podložka [18]

3.4 Herní konzole se snímači pohybu

Do povědomí veřejnosti se technologie pohybových senzorů dostala především v souvislosti s herními konzolemi (Xbox 360, 56 Nintendo Wii, Playstation Move), jejichž počítačové hry je možné ovládat pomocí pohybů lidského těla.

Jde o zařízení, která dokáží snímat pohyb lidského těla a za využití počítačové grafiky jej vizualizovat v reálném čase.

Tato zařízení lze rozdělit na dvě skupiny podle toho, jakým způsobem snímají pohyb hráče.

- Fyzický senzor – zařízení využívají senzor plnicí funkci ovladače, který má hráč obvykle v ruce, a hra zaznamenává pohyb tohoto senzoru v prostoru.
 - Příklady - Nintendo Wii, Playstation Move
- Optický senzor – tato zařízení využívají infračervený projektor k mapování pohybu uživatele před obrazovkou. Senzor potom sbírá odražené světlo od těla a software vykresluje změny pohybu hráče. Vzhledem k povaze projektoru je možné hrát za rozmanitých světelných podmínek, včetně úplné tmy.
 - Příklady - MS Kinect, Asus Xtion



Obrázek 18: Ukázka senzorů pro hry [19]

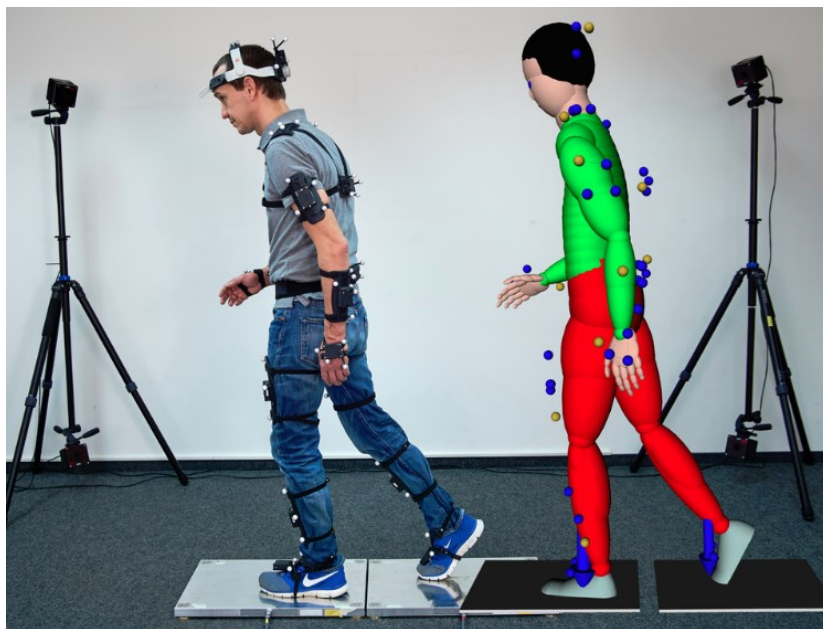
3.5 Motion Capture systémy

Systémy „Motion Capture“ se rozvíjely ruku v ruce s rostoucími požadavky na kvalitu animací jak ve filmu, tak ve hrách. Kvůli rostoucí náročnosti modelovat pohyb natolik přesně, aby v digitálním zobrazení působil věrohodně, bylo potřeba snímat pohyby živých herců či zvířat.

Tyto fyzické animace jsou následně převedeny do digitálních pomocí senzorů, které mají herci na sobě. Systém dokáže přenést do digitální animace nejen pohyby během šermířského vystoupení pro digitální hru, ale také pohyby tváře herce a přenést je na digitální obličej.

Tento systém má několik velkých výhod. Režisérům a vedoucím animátorům umožňuje okamžitě vidět scénu či výraz tváře bez náročného modelování, což výrazně zkracuje čas potřebný k tvorbě finální animace. Dále je také možné scénu pozorovat z různých úhlů, či ji zpomalit a soustředit se na detaily. [20,21,22]

Této výhody se využívá například při tréninku sportovců, kdy se nechá sportovec vykonat opakovaně nějaký úkon a díky tomuto systému lze sledovat rozdíly mezi jednotlivými pokusy. Je možné zaznamenat nejen pohyb sportovce, ale i sportovního náčiní. Díky tomuto sledování lze odhalit slabiny a také vypilovat pohyb. [21,22]



Obrázek 19: Princip MoCap [20]

3.6 Některé volně dostupné mobilní aplikace

Před započítáním vývoje praktické části této diplomové práce jsem provedl průzkum stávajících aplikací, které cílí podobným směrem, tj. na výuku komplexních pohybových aktivit, a to primárně v bojových sportech. Jsou jich desítky, ale většina z nich nestojí za zmínku. V následujících odstavcích proto popisují jen 3 aplikace, které jsou uživatelskou komunitou nejlépe hodnocené.

3.6.1 Karate WKF

Jedná se o volně dostupnou aplikaci na Google Play Store, která je velmi dobře hodnocena uživateli, a to průměrným hodnocením 4,6 bodů z 5ti (od 8401 uživatelů).

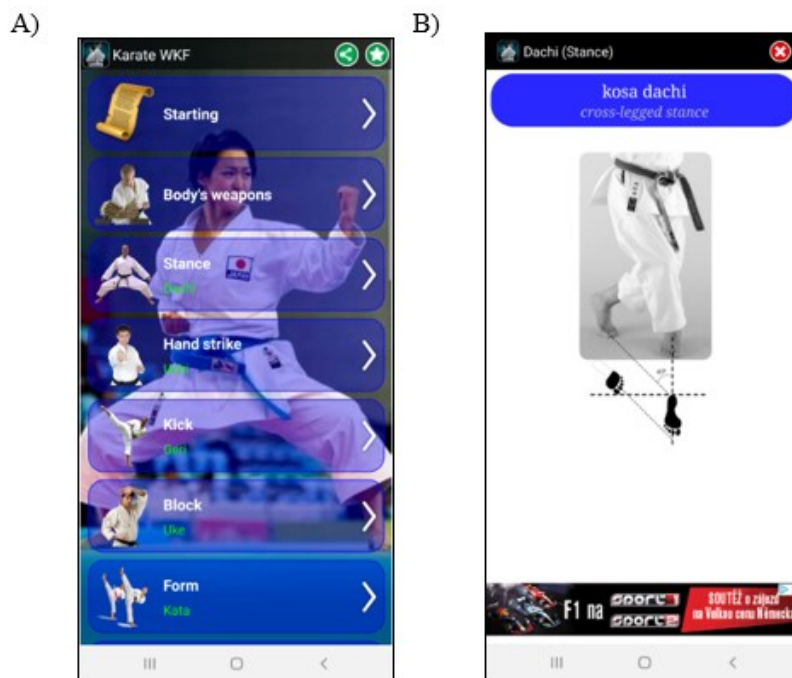
Aplikace se specializuje na bojové umění „Karate“ a je rozdělena na jednotlivé bloky, jako např. (*Obrázek 20A*):

- Postoje, Úder rukou, Kopy, Blokování atd. – tady se pomocí fotografie a nákresu učí správnému postoji a provedení při konkrétním cviku

U některých bloků se nachází také přímé odkazy na Youtube, které spustí aplikaci „Youtube“. Aplikace potom přehraje vybrané video s předvedeným cvikem.

Z mého pohledu vidím přínos této aplikace hlavně pro úplné začátečníky, kteří se zde naučí pomocí nákresů správnému postoji a provedení vybraného cviku (*Obrázek 20B*). Aplikace je jednoduchá na orientaci a přehledná pro výběr jednotlivých bloků.

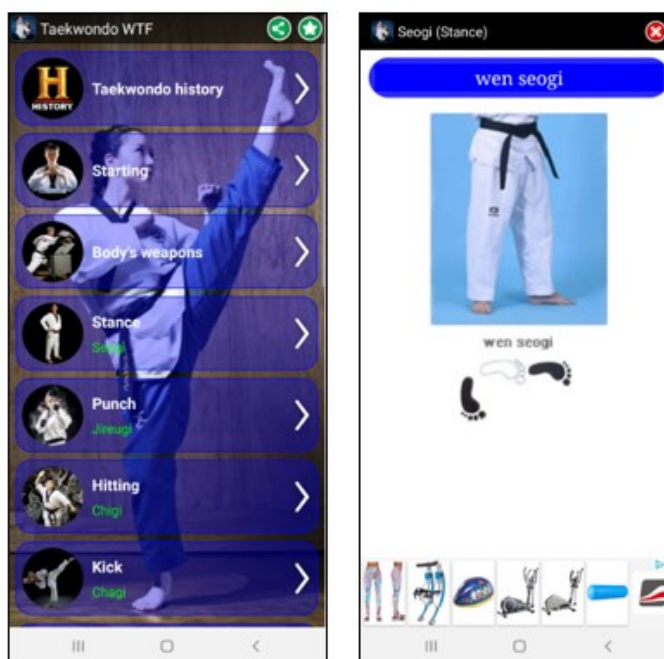
Negativně vnímám odkaz na aplikaci Youtube, která teprve spustí video a uživatel se na něj může podívat, jako by si jej spustil sám na portálu Youtube.



Obrázek 20: Ukázka aplikace „Karate WKF“

3.6.2 Taekwondo WTF

Jedná se o aplikaci od stejného vývojáře jako aplikace Karate WKF a je také velmi dobře hodnocena. Rozložení i princip aplikace je stejný, jen s obsahem pro Taekwondo.

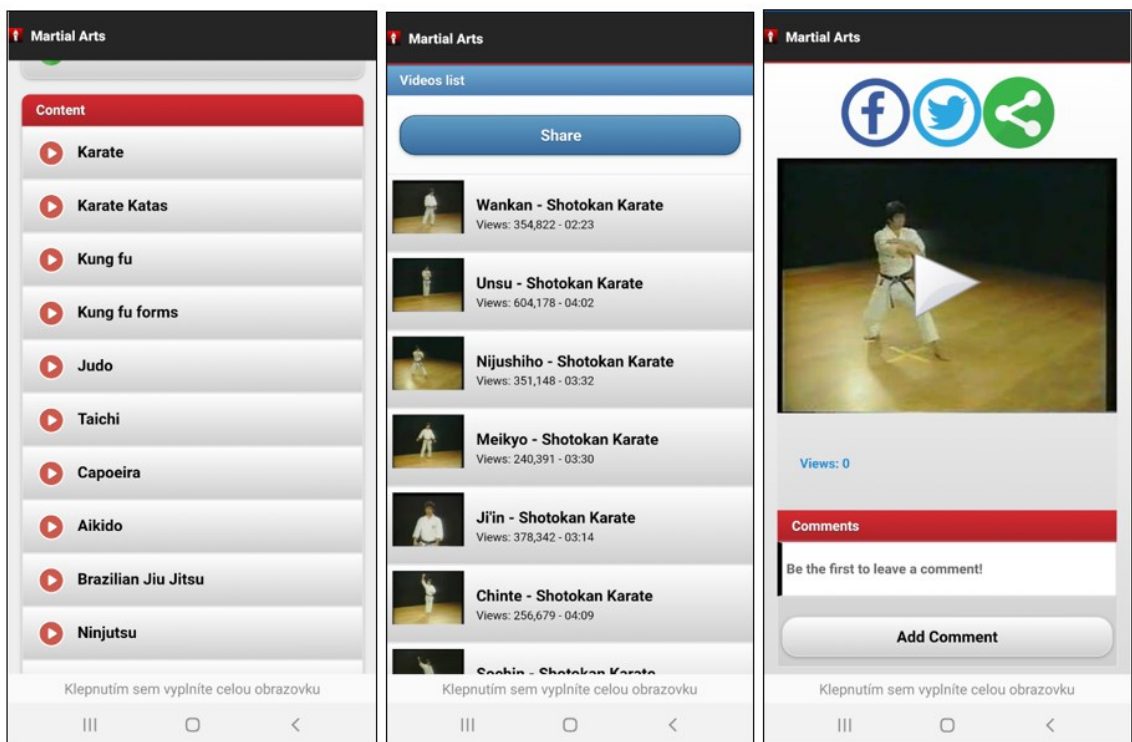


Obrázek 21: Ukázka aplikace „Taekwondo WTF“

3.6.3 Learn Martial Arts

Opět se jedná o aplikaci dostupnou na Google Play Store s hodnocením 4,5 bodů z 5ti od 3473 uživatelů. V této aplikaci je pokryto více bojových umění a ke každému bojovému umění jsou zde pouze shromážděna videa, dostupná na Youtube.

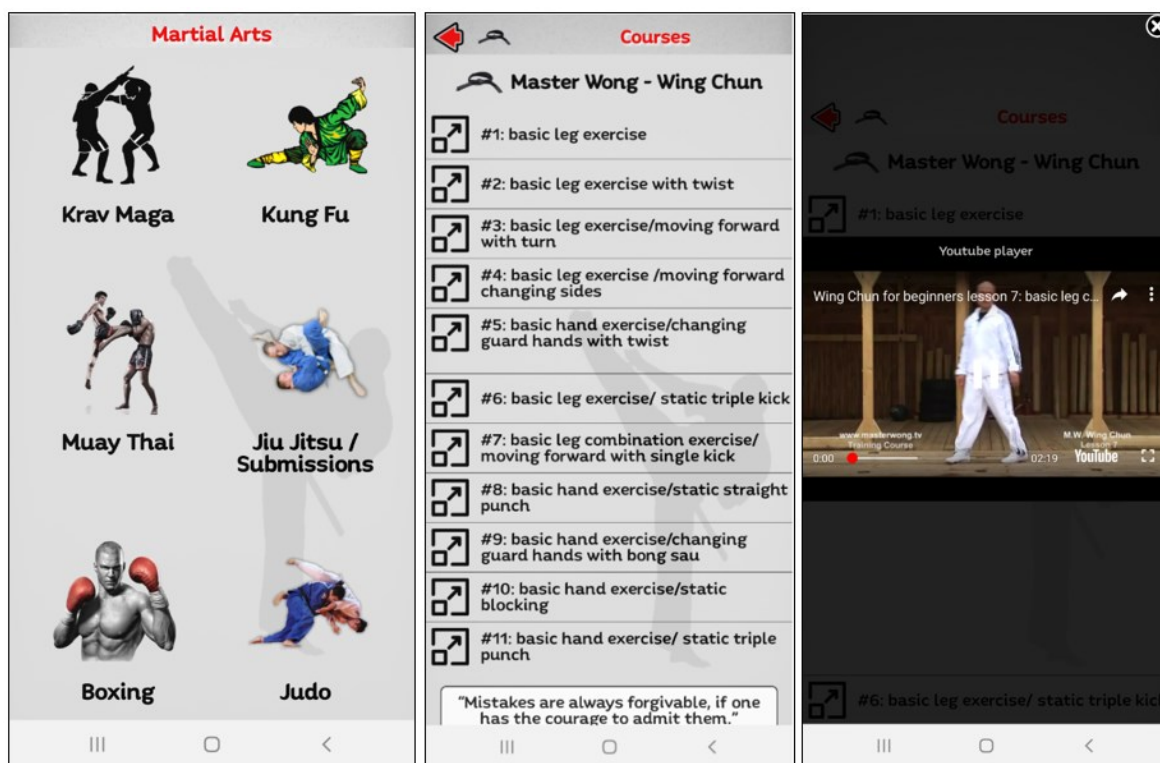
Přínos této aplikace vidím v možnosti nalezení videí k různým bojovým uměním na jednom místě. Jako nedostatek hodnotím to, že pro sledování videí musí být uživatel připojen k internetu.



Obrázek 22: Ukázka aplikace „Learn Martial Arts“

3.6.4 Martial Arts – Training and workouts

Jedná se opět o volně dostupnou aplikaci na Google Play Store s hodnocením 4,6 bodů z 5 od 382 uživatelů. V této aplikaci jsou opět pouze shromážděna videa k bojovým uměním, dostupná na Youtube.



Obrázek 23: Ukázka aplikace „Martial Arts - Training and workouts“

3.6.5 Shrnutí

Dnes dostupné mobilní aplikace vůbec nevyužívají potenciál chytrých telefonů a jejich senzorického vybavení. Jejich funkcí je pouhá „agregace a zobrazení obsahu“. K tomuto účelu ovšem nebylo potřeba tvořit mobilní aplikace, stejnou funkci by zastaly WWW stránky nebo dokumenty EPUB pro čtečky elektronických knih.

Nevýhody stávajících aplikací pomohly při tvorbě specifikace funkcí mé vlastní aplikace. Např. nevýhodu nutnosti připojení k internetu pro přehrávání výukových videí jsem vyřešil možností ukládání videa do interního úložiště telefonu. Moje aplikace také při výuce používá senzory mobilního telefonu – akcelerometr a kompas pro zjištění směru a pohybu uživatele, čímž uživateli pomáhá lépe se zorientovat v prostoru a rychleji se danou pohybovou sestavu naučit.

II. PRAKTICKÁ ČÁST

4 POŽADAVKY NA APLIKACI

Cílem vyvíjené aplikace je podpora uživatele v jeho snaze naučit se a procvičit si určitou pohybovou sestavu nebo cvik.

Vybrané video se sestavou je proto rozkouskováno na jednotlivé kroky. V každé části videa je pak potřeba nadefinovat způsob pohybu uživatele. Aplikace po předvedení dané části zastaví přehrávání videa a čeká na vstup uživatele. Poté jeho pohyb vyhodnotí a sdělí uživateli verdikt, jestli cvik provedl správně či špatně. Pokud uživatel pohyb vykoná správně, aplikace po potvrzení spustí další část videa.

Aplikace využívá přihlašovací systém pro trénink dle videa a komunikaci s dalšími členy ve skupině, komentáře a sdílení fotek.

Hlavním cílem této aplikace je osobní trénink. Uživatel tedy může procvičovat sestavy i v pohodlí domova a tím získat jistotu v nových sestavách, což napomáhá zejména začátečníkům. Dalším přínosem je podpora uživatelům, kteří mají problémy s prostorovou orientací a cviky provádějí špatně.

Aplikace také obsahuje integrovaný chatovací modul, který je vhodný zejména pro komunikaci mezi žákem a učitelem.

Aby aplikace byla funkční, musí umět stahovat uživatelem vybraná videa z internetu, kde jsou videa nepřetržitě dostupná.

5 NÁVRH APLIKACE

Návrh řešení prezentovaného v kapitole 4 je propojení systémových senzorů v smartphonech a tabletech s nativní aplikací pro operační systém Android.

Aplikace bude na základě vstupních hodnot ze systémových senzorů vyhodnocovat, zda uživatel svým pohybem splnil zadané podmínky předpřipravené ke každému videu.

5.1 Požadavky na funkce aplikace

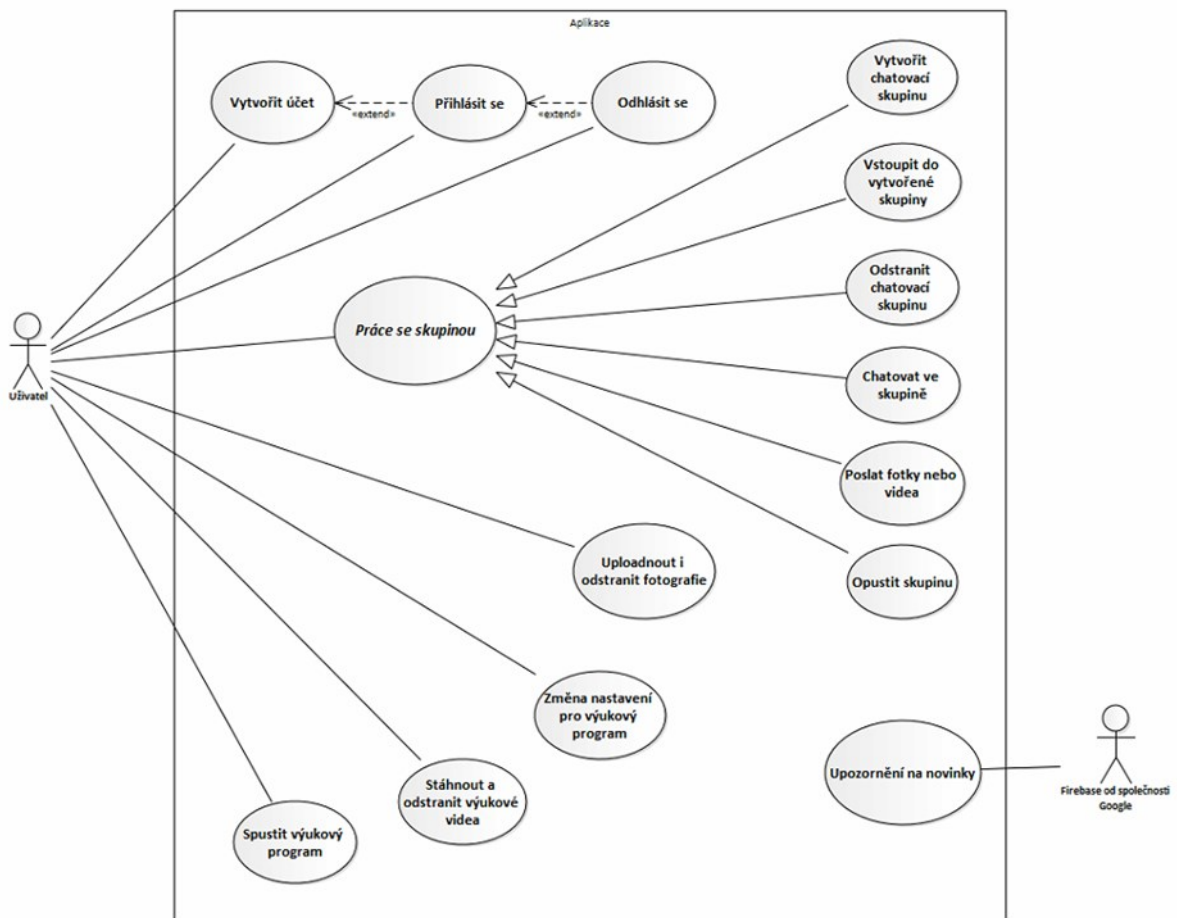
Mezi stěžejní funkce vyvíjené aplikace patří:

- ověření identity uživatele pomocí *Firestore Authentication*.
- Ukládání uživatelských informací (e-mail, jméno, příjmení, login, atd.), sdílených fotografií, soukromý a skupinový chat do *Firestore Databáze* od Google.
- Ukládání profilových obrázků a fotografií uživatelů do *Firestore Storage* od Google
- Zasílání novinek nebo upozornění všem uživatelům této aplikace pomocí *Firestore Cloud Messaging* od Google.
- Jako zdroj pro stáhnutí videa do interního úložiště chytrého zařízení je využit *YouTube*.
- Vyhodnocení probíhá pomocí kompasu.
- Bezpečnost, zajištěná pomocí *Firestore Authentication* pro ověření identity a hashovací funkcí SHA-256 pro skupiny v chatu.

5.1.1 Use Case Diagram

Na *Obrázku 24* je znázorněn *Use Case Diagram*, který byl sestaven na základě funkčních požadavků.

Do aplikace vstupují dva aktéři, aktivní a pasivní. Pasivním aktérem je *Firestore* od společnosti Google, který má možnost jen zasílat upozornění. Aktivním aktérem je sám uživatel aplikace schopný využívat všechny možnosti aplikace.



Obrázek 24: Use Case Diagram podle funkčních požadavků

5.2 Nefunkční požadavky

Mezi tyto požadavky patří například dostupnost, výkon nebo bezpečnost. Splnění těchto požadavků je důležité pro správnou funkci softwaru.

Pro vyvíjenou aplikaci jsou to např. tyto požadavky:

- **Síťové požadavky** – Pro plné využití aplikace je nutný přístup k internetu. Aplikace podporuje téměř všechny síťové standardy, jako například WiFi, 3G, LTE..
- **Efektivní využití paměti** – V tomto směru není aplikace nijak náročná. Pro ještě efektivnější využití paměti byl využit *RecyclerView*, který při scrollování chatu, fotografií i videí se jednotlivé instance využívají znovu, jakmile jsou už neviditelné.
- **Optimalizace využití baterie** – Tento požadavek je závislý na mnoha faktorech, které však jsou ovlivnitelné uživateli. Záleží na tom, v kolika skupinách je uživatel přihlášen. Z každé skupiny přicházejí notifikace o změnách. Z toho vyplývá, že čím více skupin, tím více notifikací a tím větší spotřeba baterie.

- **Bezpečnost** – Každý uživatel má svůj login a heslo. Heslo je uloženo v databázi ve formátu *hashe*, který je vytvořen pomocí funkce SHA-256.
- **Lokalizace** – Aplikace byla vyvíjena v českém jazyce
- **Geomagnetický senzor v mobilním zařízení** – senzor snímá intenzitu magnetického pole, je tedy využíván jako digitální kompas

5.3 Programovací jazyky

Pro aplikaci byly vybrány z množství programovacích jazyků jazyk *Java*, *Kotlin* a *JavaScript*.

5.3.1 Java

Jazyk *Java* je oficiální programovací jazyk pro vývoj Androidu. Tento jazyk je jeden z nejvíce podporovaných programovacích jazyků od Google a také nejvíce využívaný jazyk u aplikací na *Google Play Store*. [23]

Samotná *Java* byla vydána společností Sun Microsystems v roce 1995 a je používána pro škálu programovacích aplikací. Kód tohoto jazyky je spuštěn „virtuálním strojem“, který běží na Android zařízeních a tento kód interpretuje. [23]

5.3.2 Kotlin

Tento jazyk vstoupil na scénu jako „jiný“ oficiální jazyk pro vývoj Androidů. V hodně věcech je podobný jazyku *Java*, ale pro každého uživatele se jeví jinak. Někdo ho považuje za jednodušší a někdo zase za složitější. [23]

Stejně jako *Java* běží *Kotlin* na *Java Virtual Machine*, je plně interoperabilní s jazykem *Java* a nezpůsobuje žádné zpomalení. Rozdíl je v tom, že vyžaduje méně opakujících se kódů, což vede k více zjednodušenému a lehce čtenému systému. [23]

5.3.3 JavaScript

JavaScript je programovací jazyk, který se spíše používá pro *www* stránky a je často přímo vkládán do HTML kódu. V mé práci byl tento programovací jazyk použit na straně serveru za použití *Firestore Cloud Functions*.

5.4 Zabezpečení hesel – SHA-256 (Secure Hash Algorithm)

Jedná se o jednu z kryptografických hashovacích funkcí. Funkce jsou uplatňovány v oblastech digitálního podpisu, při ověřování integrity dat atd.

Pomocí funkce SHA-256 se generuje hash hesla. Jde o jednosměrnou funkci – získá se pouze výstup ze vstupu, ale nelze už získat vstup z výstupu. Malá změna vstupu musí vést k velké změně výstupu, tzn. zpráva s malou změnou musí mít hash zcela nepodobný hashí původní zprávy. Funkce poskytuje ze vstupních dat výstup (otisk) fixní délky. V tomto případě délky 256 bitů.

Hashovací algoritmy jsou „bezpečné“, pokud splňují následující podmínky:

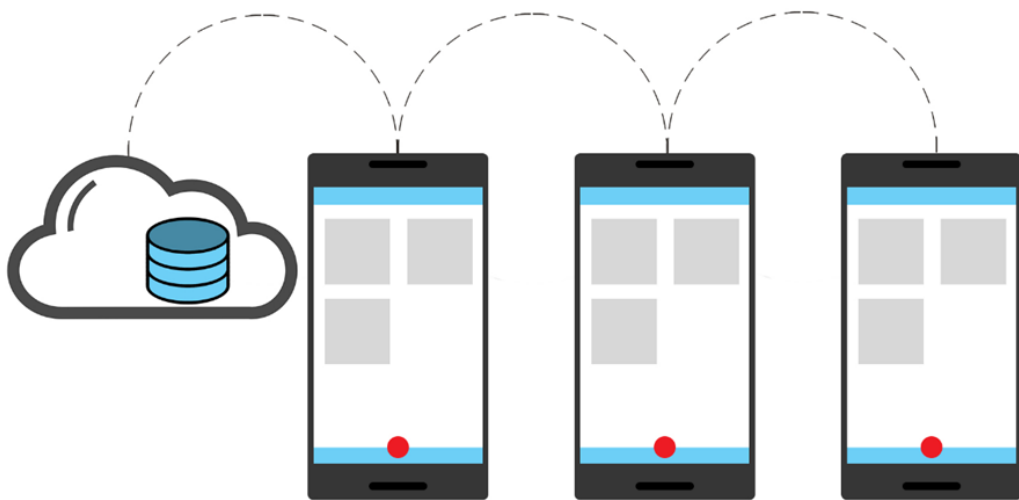
- současnými prostředky je nemožné najít zprávu, která odpovídá svému otisku,
- je nemožné najít dvě rozdílné zprávy, které mají stejný otisk,
- jde o jednosměrnou funkci – je snadné spočítat hash zprávy, ale nemožné nalézt k danému hashí vstupní zprávu,
- jakékoliv množství vstupních dat poskytuje stejně dlouhý hash.

6 FIREBASE

Jedná se o platformu pro vývoj webových a mobilních aplikací, která obsahuje řadu knihoven, jak pro mobilní zařízení (iOS, Android), tak i pro web (JavaScript). Platforma nabízí databáze *real-time* a potřebné nástroje jsou dostupné zdarma.

6.1 Storage

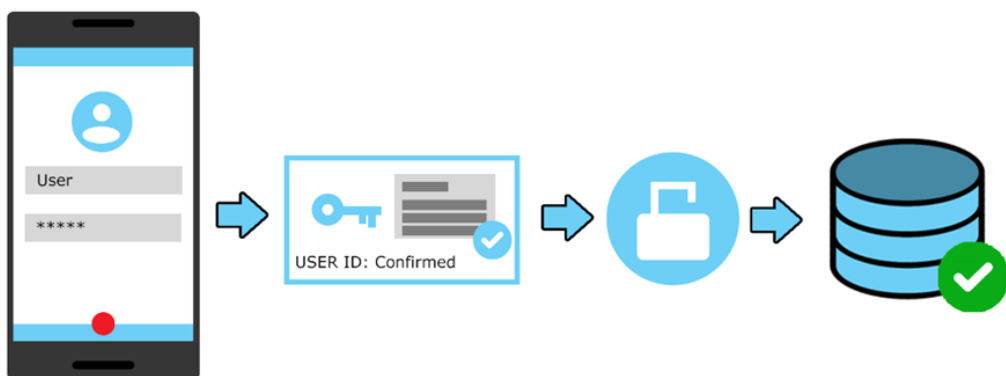
Jedná se o jednoduchou službu pro ukládání fotografií, popřípadě videí. Pokud je síťové připojení špatné, klient je schopen navázat na operaci přímo tam, kde došlo k přerušení.



Obrázek 25: Ukázka Storage

6.2 Authentication

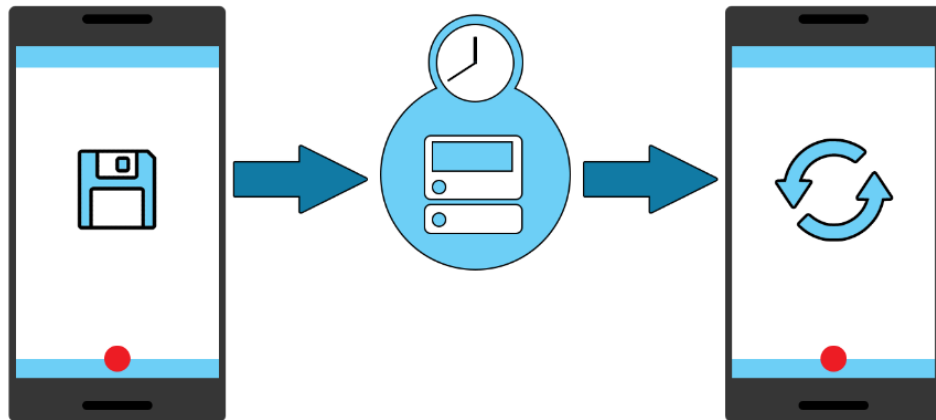
Většina aplikací potřebuje znát identitu uživatele. Ve vyvíjené aplikaci je tato funkce využita pro přihlášení uživatele pomocí emailu a hesla nebo pomocí Google účtu.



Obrázek 26: Ukázka Authentication

6.3 Databáze

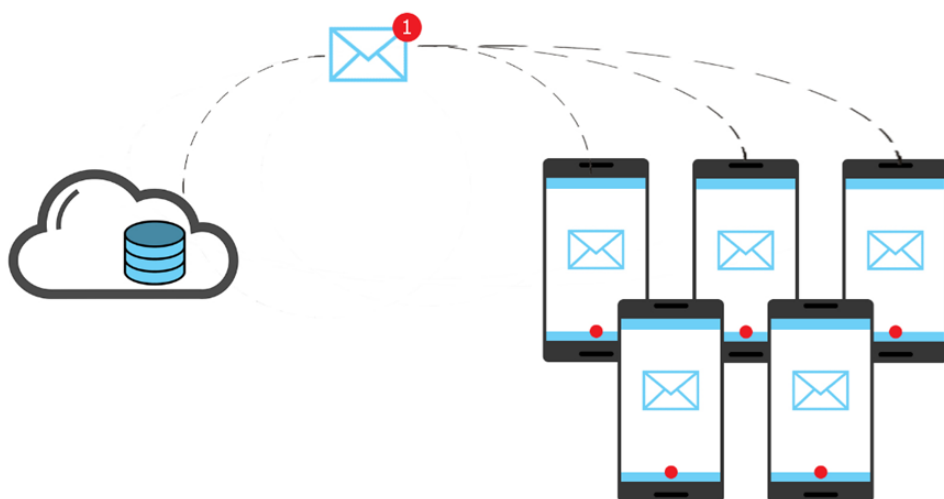
Databáze slouží k ukládání a synchronizaci dat. Data jsou uložena ve formátu JSON a synchronizována u všech klientů v reálném čase. Data zůstávají k dispozici po celou dobu online režimu aplikace.



Obrázek 27: Ukázka Database

6.4 Cloud Messaging

Jedná se o službu, která umožňuje spolehlivě doručovat zprávy bez nutnosti provozovat server. Zpráva může být zaslána s upozorněním, aby došlo k opětovnému zapojení a k udržení uživatelů v systému.

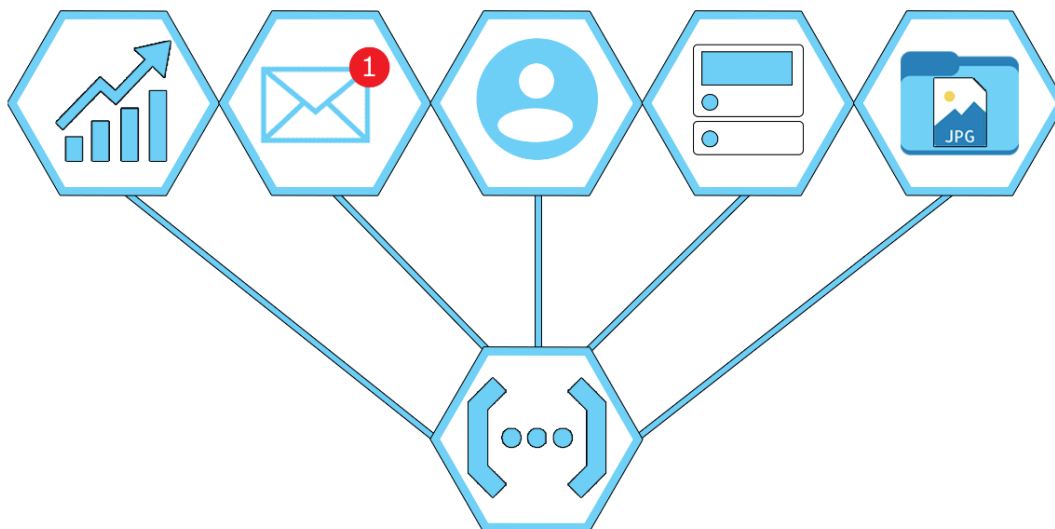


Obrázek 28: Ukázka Cloud Messaging

6.5 Cloud Functions

Tato služba umožňuje automaticky spouštět „backendový“ kód v reakci na události vyvolané Firebase funkcemi a HTTPS požadavky.

Pro každou novou funkci je vývojářem vytvořen kód, který definuje nejen podmínky pro fungování dané funkce, ale i samotnou událost (např. Realtime Database).



Obrázek 29: Ukázka možností Cloud Functions

7 IMPLEMENTACE

Tato aplikace byla vyvíjena pro operační systém Android ve verzích 5.0 Lollipop a vyšších. Dostupnost této aplikace je tímto zaručena pro 86 % uživatelů. Při použití těchto verzí bylo možno využít některých technik a vylepšení, která jsou nedostupná ve starších operačních systémech Androidu. Pro vývoj aplikace bylo použito integrované vývojové prostředí Android Studio.

V následujících podkapitolách jsou popsány vybrané části vývoje aplikace.

7.1 Přihlášení

Pro využití plné funkcionality aplikace je zapotřebí přihlášení uživatele do systému pomocí emailu a hesla. Bez přihlášení lze využít aplikaci v „off-line-mode“ pro trénink a pouze tehdy, když jsou výuková videa již stažena v interním úložišti chytrého zařízení.

Pro každého nového uživatele jsou zde dvě možnosti. První možností je vytvoření účtu pomocí emailu a hesla, druhá možnost je vytvoření účtu pomocí svého stávajícího Google účtu. Je jedno, která možnost je vybrána, neboť informace z obou možností jsou vkládány do „Real-time“ databáze od společnosti Google.



```
ERtkZ601sZctjvdp1r884071Jw52
{
  "countOfPhotos": "0",
  "email": "pp@gmail.com",
  "id": "ERtkZ601sZctjvdp1r884071Jw52",
  "jmeno": "Pepa",
  "level": 1,
  "photoInLocalStorage": "content://media/external/images/media/!",
  "photoInLocalStorageBig": "content://media/external/images/media/!",
  "photosList": "https://firebasestorage.googleapis.com/v0/b/vol",
  "photosListBig": "https://firebasestorage.googleapis.com/v0/b/vol",
  "photosListBigThumb": "https://storage.googleapis.com/volnoaktivita",
  "photosListThumb": "https://storage.googleapis.com/volnoaktivita",
  "prezdivka": "Josef",
  "prijmeni": "Kdoví",
  "skupiny": [],
  "token": "e86LktLKfLg:APA91bGK-V9IbYzg8xVbnZ_rhw-IscrzdUF"
}
```

Obrázek 30: Ukázka jedné hodnoty v uzlu „Users“

Na Obrázku 30 lze vidět uložení informací o jednom uživateli v „Real-time“ databázi. Hlavní uzel je pojmenován „Users“ a dochází v něm k uložení informací každého uživatele.

Každý uživatel má vlastní „child“ uzel, je pojmenován podle unique ID daného přihlášeného uživatele a nachází se pod hlavním uzlem „Users“. V „child“ uzlu se nachází informace o uživateli, jako „countOfPhotos“, „email“, „id“, „jmeno“, „level“, „photoInLocalStorage“, „photoInLocalStorageBig“, „photosList“, „photosListBig“, „photosListBigThumb“, „photosListThumb“, „prezdivka“, „prijmeni“ a „token“, které se později využívají pro běh dané aplikace.

Hodnota u „countOfPhotos“ nám udává aktuální počet fotografií, které uživatel nahrál do systému. Hodnota u „photosList“ je URL odkaz pro stažení profilového obrázku z *Firebase Storage* do „ImageView“ v „AccountActivity“. Token daného zařízení je uložen v hodnotě „token“, to je proto, aby *Firebase Functions* dokázal odeslat příslušné upozornění vybraným uživatelům, a ne všem uživatelům, kteří se v systému nachází.

Přidání nového uživatele do databáze začíná tím, že se deklaruje reference na určitý uzel v databázi, jak je vidno na ukázce „Kód 1“.

```
mDatabaseRef = FirebaseDatabase.getInstance().getReference("Users")
```

Kód 1: Reference na určitý uzel v databázi

Následuje vytvoření účtu ve „*FirebaseAuthentication*“, který potvrdí úspěšnost. Jestli ano, je uživatel vložen do databáze, viz. „Kód 2“.

```
val id = mAuth.currentUser!!.uid
val user = AccountInformation("", id,
    Level, jmeno, prezdivka, prijmeni, email,
    photoList, "", "0", "",
    "", "", "")
mAccountInformation = user
updatePrefsUser()
mDatabaseRef.child(id).setValue(user)
```

Kód 2: Vytvoření účtu

Vytvoří se informace o daném uživateli, kde se z „*FirebaseAuthentication*“ zjistí jeho unikátní ID a toto ID se spolu s informacemi o uživateli se vloží do databáze.

Po úspěšném přihlášení a vložení informací do databáze se automaticky otevře druhá aktivita aplikace, která má na starosti zobrazení patřičných informací o daném uživateli.

7.2 Chatování

Pokud se uživatel přihlásí, je mu umožněno využívat plnou funkcionalitu aplikace. Jednou z těchto funkcionalit je právě chat, který funguje ve skupinách, tj. dva a více uživatelů. Pomocí tohoto chatu mezi uživateli ve skupině lze nejen psát, ale i sdílet fotografie a videa.

7.2.1 RecyclerView

„RecyclerView“ je widget vhodný pro použití při zobrazování chatovacích oken stejného typu. Typy jsou dva – jeden pro odesílatele a druhý pro příjemce zpráv, které jsou barevně odlišeny. Na *Obrázku 34* v kapitole 8.3 je znázorněn odesílatel zeleně a od ostatních uživatelů chatu v této skupině přijímá šedě.

Když je jeden *view* odebrán, tedy posunut mimo viditelnost na displeji o určitou vzdálenost, potom je přidán zpět do zásoby prázdných. Když je vytvořen nový *view*, tak je přidán do „RecyclerView“ a tím je odebrán ze zásoby prázdných. Díky tomuto je šetrný na využití paměti chytrých zařízení na rozdíl od „ListView“. Tento rozdíl je znázorněn na *Obrázku 31 A, B*.



Obrázek 31: *ListView* (A) a *RecyclerView* (B)

Na ukázce „Kód 3“ ve funkci „*onCreateViewHolder()*“ jsou vytvářeny nové prázdné *view* a přidávány do zásoby.

```
@Override
public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int
viewType) {
    switch(viewType)
    {
        case 0: return new
ChatAdapter.ImageViewHolder(LayoutInflater.from(mContext).inflate(R.layout.
recyclerview_row_chat, parent, false));
        case 2: return new
ChatAdapter.ImageViewHolder1(LayoutInflater.from(mContext).inflate(R.layout
.recyclerview_row_me_chat, parent, false));
    }
    return null;
}
```

Kód 3: Vytvoření nové View

Následuje funkce „*onBindViewHolder()*“, která získá *view* a naplní ho danými daty.

Proměnná „*uploadCurrent*“ typu „*ChatUploads*“ se mění na základě pozice zobrazovaného *view*, jak je vidno na ukázce „Kód 4“.

```
@Override
public void onBindViewHolder(@NonNull RecyclerView.ViewHolder viewHolder,
final int i) {
    ChatUploads uploadCurrent;
```

Kód 4: Deklarace proměnné „*uploadCurrent*“ pro následné upravování

Příkaz *switch* dělí program do více větví, v našem případě do dvou. Rozhoduje se podle typu *view*. Pokud je typ *view* roven nule spustí se první větev, jak je vidno na ukázce „Kód 5,6,7“.

```
switch(viewHolder.getItemViewType())
{
    case 0:

        ImageViewHolder viewHold = (ImageViewHolder) viewHolder;
        uploadCurrent = mUploads.get(i);
        viewHold.textViewName.setText(uploadCurrent.getMName());
        viewHold.textViewDescription.setText(uploadCurrent.getMText());
        viewHold.time.setText(uploadCurrent.getMTime());

        if(uploadCurrent.getMText().equals(""))
        {
            viewHold.textViewDescription.setVisibility(View.GONE);
        }else {
            viewHold.textViewDescription.setVisibility(View.VISIBLE);
        }
        Glide.with(mContext)
            .load(uploadCurrent.getMImageUrl())
            .into(viewHold.imageViewImageProfile);
```

Kód 5: Začátek první větve

```

        if(uploadCurrent.getmImageUrlSended().equals("") &&
uploadCurrent.getmVideoUrl().equals(""))
        {
            viewHold.imageViewImageSended.setVisibility(View.GONE);
        }else {

            if(uploadCurrent.getmImageUrlSended().equals(""))
            {

viewHold.imageViewImageSended.setVisibility(View.VISIBLE);
                Glide.with(mContext)

.load("https://firebasestorage.googleapis.com/v0/b/volnoaktivita.appspot.co
m/o/Thumbnails%2Fthumb_play-button-512.png?alt=media&token=104fe204-de70-
4596-9c64-3174eabf7d5e")

                .apply(new RequestOptions()
                    .centerCrop()
                    .placeholder(R.drawable.loader_test)
                    .error(R.drawable.loader_test)

.diskCacheStrategy(DiskCacheStrategy.ALL)
                    .priority(Priority.HIGH)
                    .dontAnimate()
                    .override(mWidth, mHeight)
                    .dontTransform()
                )
                .into(viewHold.imageViewImageSended);
            }
        }
    }
}

```

Kód 6: Pokračování první větve

```

        }else{

viewHold.imageViewImageSended.setVisibility(View.VISIBLE);
            Animation a = AnimationUtils.loadAnimation(mContext,
R.anim.progress_anim);
            if(uploadCurrent.getmImageThumbUrl().equals(""))
            {
                a.setDuration(2000);
                viewHold.imageViewImageSended.startAnimation(a);
            }else {
                viewHold.imageViewImageSended.clearAnimation();
                viewHold.imageViewImageSended.animate().cancel();
            }
            Glide.with(mContext)
                .load(uploadCurrent.getmImageThumbUrl())
                .apply(new RequestOptions()
                    .centerCrop()
                    .placeholder(R.drawable.loader_test)
                    .error(R.drawable.loader_test)

.diskCacheStrategy(DiskCacheStrategy.ALL)
                    .priority(Priority.HIGH)
                    .dontAnimate()
                    .override(mWidth, mHeight)
                    .dontTransform()
                )
                .into(viewHold.imageViewImageSended);
        }
    }
}
break;

```

Kód 7: Konec první větve

Pokud je typ *view* roven dva spustí se druhá větve.

Všechna data v určité skupině jsou obsažena v „*ArrayList*“ s názvem „*mUploads*“

7.3 Přidání fotografie

Po vybrání fotografie z interního úložiště se vytvoří notifikace, která má za úkol informovat uživatele o průběhu nahrání této fotografie do „*Firestore*“. Po úspěšném nahrání se do databáze přidají informace o této fotografii (viz Kód 8,9) a spustí se samovolně funkce v jazyku JavaScript, která se nachází na „*Firestore*“, které reagují na jakoukoliv změnu ve „*Firestore*“. Tato funkce v JavaScriptu si vezme originální nahranou fotografii a provede její zmenšení v rozlišení tak, aby její následné použití využívalo co nejméně dat.

```
if (resultCode == Activity.RESULT_OK)
{
    val largeIconUploadImage =
        BitmapFactory.decodeFile (data!!.data!!.toString())

    val integerForNotification = Random.nextInt()
    val notification =
        NotificationCompat.Builder (this,
            "channel2")
            .setSmallIcon (R.drawable.ic_notification)
            .setContentTitle (getString (R.string.chatActivityNotifikaceTitle))
            .setContentText (getString (R.string.chatActivityNotifikaceText))
            .setLargeIcon (largeIconUploadImage)
            .setPriority (NotificationCompat.PRIORITY_HIGH)
            .setCategory (NotificationCompat.CATEGORY_PROGRESS)
            .setColor (Color.BLUE)
            .setOngoing (true)
            .setOnlyAlertOnce (true)
            .setProgress (100, 0, false)
    notificationManager.notify (integerForNotification,
        notification.build())
}
```

Kód 8: Vytvoření a zobrazení notifikace

```

val l = System.currentTimeMillis().toString() + "." +
    getFileExtension(data!!.data)
val filereference = mStorageRef!!
    .child("/$newStringIDGroup/$l")
filereference.putFile(data!!.data!!)
    .addOnSuccessListener(OnSuccessListener<UploadTask.TaskSnapshot>
{ taskSnapshot ->
    val handler = Handler()
    val r = Runnable {
        kotlin.run {
            notification.setContentText(
                getString(R.string.chatActivityNotifikaceSended))
                .setOngoing(false)
                .setProgress(0,0,false)

notificationManager.notify(integerForNotification,notification.build())
        }
    }
    handler.postDelayed(r,500)
    val task = Objects.requireNonNull<StorageReference>(
Objects.requireNonNull<StorageMetadata>(taskSnapshot.metadata).reference
    ).downloadUrl
    task.addOnSuccessListener{
        val currentTime = Calendar.getInstance().time
        val upload =
ChatUploads(currentTime.toString().substring(3,16),newStringJmeno,"",newStr
ingFotka, newStringID, it.toString(),newStringNazevSkupiny,mUploads.size +
1,"","")
            mDatabaseRef.child((mUploads.size +
1).toString()).setValue(upload)
        }

    }) .addOnProgressListener {
        val progress = (100.0 * it.bytesTransferred /
it.totalByteCount)
        notification.setProgress(100,progress.toInt(),false)
        notification.setOngoing(false)

notificationManager.notify(integerForNotification,notification.build())
    }
    mDatabaseRef.addValueEventListener(messageListener)
    mDatabaseRefPeople.addValueEventListener(messageListener1)
}

```

Kód 9: Přidání a uložení fotografie

Po úspěšném zmenšení dané fotografie se zobrazí pro všechny uživatele chatu v dané skupině. Jednotliví uživatelé chatu mají možnost kliknout na obrázek zmenšené fotografie a zobrazit si danou nahranou fotografii ve větším okně a v plném rozlišení.

7.4 Média

Jedná se o aktivitu určenou převážně pro přihlášené uživatele této aplikace. V této aktivitě je možnost nahrání jakéhokoliv videa či fotografie. Slouží to jako zálohování v Cloud úložišti. Přístup k uloženým mediím má pouze uživatel, který je uložil.

Je zde využít „*RecyclerView*“, který má podobnou funkcionalitu, která je popsána v kapitole 7.2.1. Rozdíl je v tom, že je použit jiný *view*. Dochází zde ke střídání *view* mezi malým a velkým zobrazením. Tyto rozdílné pohledy jsou zobrazeny na *Obrázku 35* v kapitole 8.4.

7.5 Stažení výukových videí

Informace o jednotlivých videích jsou obsažena v externím XML souboru „*data*“. Každé video obsahuje:

- Název – jméno příslušného videa
- URL adresu – internetový odkaz na YouTube video
- Timecodes – určují čas spuštění a zastavení videa
- Strings – názvy cviků
- Nadpis – označení videa
- Senzory – určují směr a hodnotu natočení

Každé video může být staženo do interního uložení pomocí URL odkazu. O toto stažení se stará systémová služba „*DownloadManager*“. Jedná se o službu, která zpracovává dlouhotrvající stahování http na pozadí (viz Kód 10).

```
DownloadManager downloadmanager =
    (DownloadManager) mContext.getSystemService(
        Context.DOWNLOAD_SERVICE);
Uri uri = Uri.parse(mTempDownloadVideos.get(i));

DownloadManager.Request request =
    new DownloadManager.Request(uri);
request.setTitle(mUploads.get(i).getmTitle());
request.setDescription(mUploads.get(i).getmDescription());
request.setNotificationVisibility(DownloadManager.
    Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED);
request.setVisibleInDownloadsUi(true);
String p = Environment.getExternalStorageDirectory().
    getAbsolutePath() + File.separator +
    Environment.DIRECTORY_MOVIES + File.separator +

mContext.getResources().getString(R.string.nazevSlozkyVideaYoutube);
File konec = new File(p);
if(!konec.exists())
{
    konec.mkdirs();
}
request.setDestinationUri(Uri.parse(String.valueOf(
    Uri.fromFile(new File(p + File.separator +
        mUploads.get(i).getmTitle() + ".mp4")))));

downloadmanager.enqueue(request);
Toast.makeText(mContext, mContext.getResources().getString(R.string.
    stahovaniSlozkyVideaYoutube),
    Toast.LENGTH_SHORT).show();
viewHolder.downloadButton.setImageResource(R.drawable.
    ic_check_download_videos);
viewHolder.deleteButton.setEnabled(true);
```

Kód 10: Systémová služba „DownloadManager“

7.6 Výuková aktivita

Pro spuštění této aktivity je podmínkou mít dané výukové video stažené v interním úložišti mobilního zařízení.

Při spuštění aktivity dojde k načtení parametrů zvolených uživatelem při nastavování viz. Kapitola 8.4, proběhne inicializace a aktivace určitých senzorů. V dané aplikaci jde o akcelerometr a magnetometr.

Po stisknutí tlačítka „START“ se spustí výukový program, který využívá tři vlákna:

User Interface (uživatelské rozhraní)

- má na starosti uživatelské rozhraní
- jako jediné toto rozhraní může aktualizovat

- Je-li na tomto vlákne provedeno mnoho složitých kódů s dlouhými procesy, má aplikace za následek slabý výkon – zpomalit, zmrazit nebo dokonce zcela zhroutit

allTheTime (po celou dobu výuky)

Při spuštění má zadané dva parametry, se kterými pracuje. Prvním parametrem jsou veškerá data z XML souboru a druhý parametr udává index videa. Hlavním úkolem tohoto vlákna je ovládání videa v reálném čase. Při pozastavení videa toto vlákno předá určitá data vláknu „*someTime*“ a také jej spustí. Po spuštění vlákna „*someTime*“ se vlákno „*allTheTime*“ pozastaví a čeká na impuls z vlákna „*someTime*“ ke svému znovu oživení. Po dokončení celé výukové sestavy se vlákno „*allTheTime*“ vypne.

someTime (vlákno občas)

Po získání dat od vlákna „*allTheTime*“ dochází pomocí těchto dat k výpočtu správného natočení, které je využito pro natočení červené šipky a pro vyhodnocení správného natočení zařízení v prostoru. Po splnění úkonu odešle impuls vláknu „*allTheTime*“ a samo se vypne.

Pokud dojde ke stisknutí tlačítka „*TEST*“, tak nedochází k využití výukového programu jako takového, ale nastává možnost seznámit se s daným videem bez vyhodnocování.

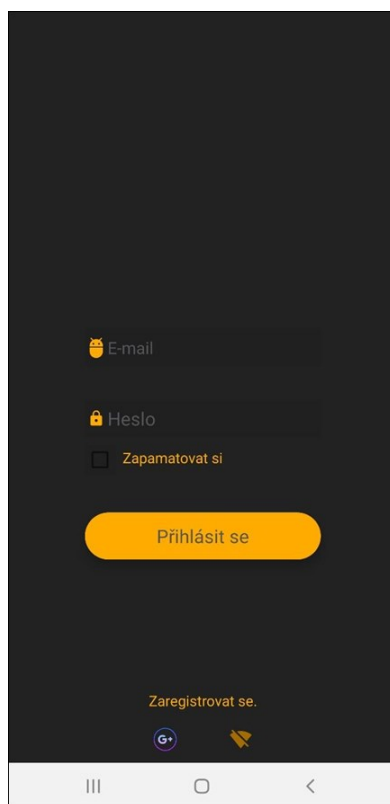
8 PREZENTACE VYTVOŘENÉ APLIKACE

Vytvořená aplikace se skládá z následujících aktivit:

- Přihlašovací aktivita – slouží k přihlášení uživatele nebo ke spuštění off-line verze
- Profilová aktivita – slouží pro zobrazení informací o uživateli
- Chatovací aktivita – slouží pro komunikaci ve vytvořených skupinách
- Média – slouží pouze pro uživatele (fotografie, videa)
- Nastavení – slouží pro výběr možnosti pokračování při výuce
- Výuková aktivita – slouží pro výuku dle zvoleného videa

8.1 Přihlašovací aktivita

Po spuštění aplikace se uživateli zobrazí přihlašovací obrazovka s možností přihlášení a registrace (Obrázek 32). Přihlášení se dá provést dvěma způsoby. První je přihlášení pomocí emailu a hesla, druhý způsob je pomocí Google účtu. Pokud nechce uživatel využít ani jednu z uvedených možností a do aplikace se nepřihlásí, má možnost trénovat u předem stažených videí. Tento trénink se automaticky vypne po úspěšném přihlášení.



Obrázek 32: Přihlášení uživatele

Přihlašovací obrazovka byla sestavena z těchto prvků:

- **EditText** – jednoduchý widget obsahující textové pole, do kterého lze vepisovat text. Složí pro zadání emailu a hesla uživatele. První má vstupní typ „emailové adresy“ a druhý „heslo“, po vepsání hesla se tento text postupně mění na tečky.
- **Drawable** – obecná abstrakce pro něco, co lze kreslit. Nemá možnost přijímat události nebo jinak komunikovat s uživatelem.
- **TextView** – jednoduchý widget obsahující pouze textové pole s neměnným textem. Využit pro pojmenování CheckBoxu, pojmenování tlačítka a pro registraci.
- **ImageView** – widget pro zobrazení obrázku. Pro účel v této aktivitě bylo použito logo společnosti Google. Při kliknutí na toto logo se zapne Intent pro přihlášení pomocí Google účtu.
- **Button** – tlačítko pro přihlášení do systému po vyplnění emailu a hesla
- **CheckBox** – slouží ke změně stavu mezi pravdou a nepravdou

Při prvním spuštění této aplikace jsme dotázáni na oprávnění ke čtení a zápisu dat v chytrém zařízení. Při potvrzení tohoto požadavku se při dalším opětovném spuštění již aplikace nedotazuje. Dalším krokem je výběr přihlášení nebo vytvoření účtu. Po úspěšném přihlášení se přihlašovací obrazovka zavře a spustí se profilová aktivita (*Obrázek 33A*)

8.2 Profilová aktivita

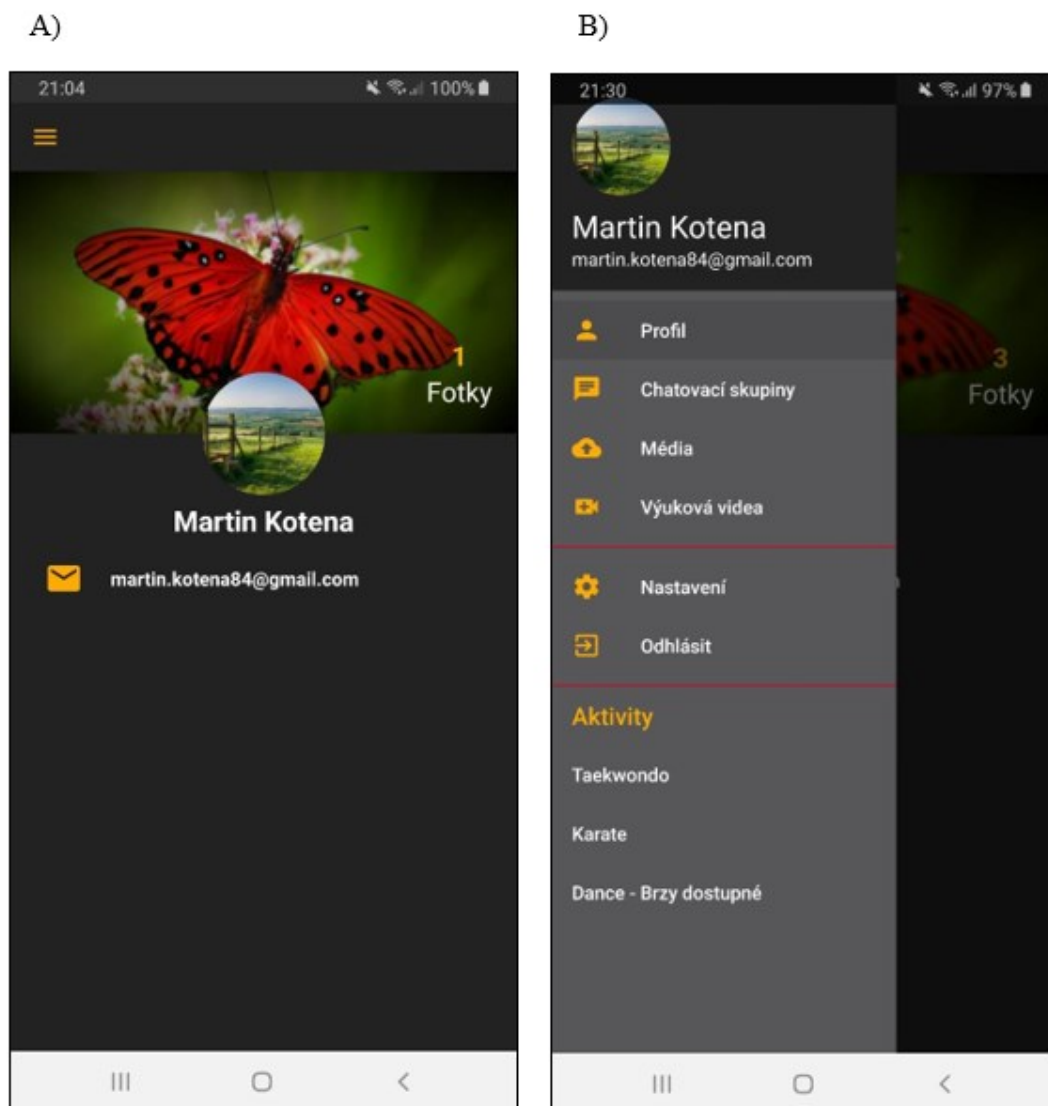
Tato aktivita se skládá z prvků TextView, ImageView a NavigationView. První dva jsou popsány v přihlašovací aktivitě a NavigationView reprezentuje standardní navigační nabídku pro aplikaci.

V této aktivitě si uživatel může vybrat profilové fotografie z vlastního zařízení. Po vybrání příslušné fotografie se tato nahraje do cloudového úložiště do Firebase Storage od společnosti Google. Po úspěšném nahrání se pomocí Firebase Cloud Functions aktivuje JavaScript funkce, která samostatně přemění tuto fotografii do menšího rozlišení tzv. thumbnail.

Tento thumbnail je používán pro opětovné načítání fotografií z důvodu snížení přenosu dat. Pro zobrazení vybrané fotografie se využívá Glide. Jde o rychlé a efektivní open source pro

načítání, dekodování a zobrazování obrázků a animovaných souborů GIF. Obrázky mohou být jak v interním úložišti zařízení, tak i URL odkaz pro obrázek na internetu.

Přes NavigationView, který se otevře pomocí přetažení prstu po displeji z levého okraje obrazovky k pravému okraji nebo pomocí tlačítka, se otevře menu (Obrázek 33B), kde si uživatel vybere z nabídky možností.



Obrázek 33: Profilová aktivita (A) Menu Profilové aktivity (B)

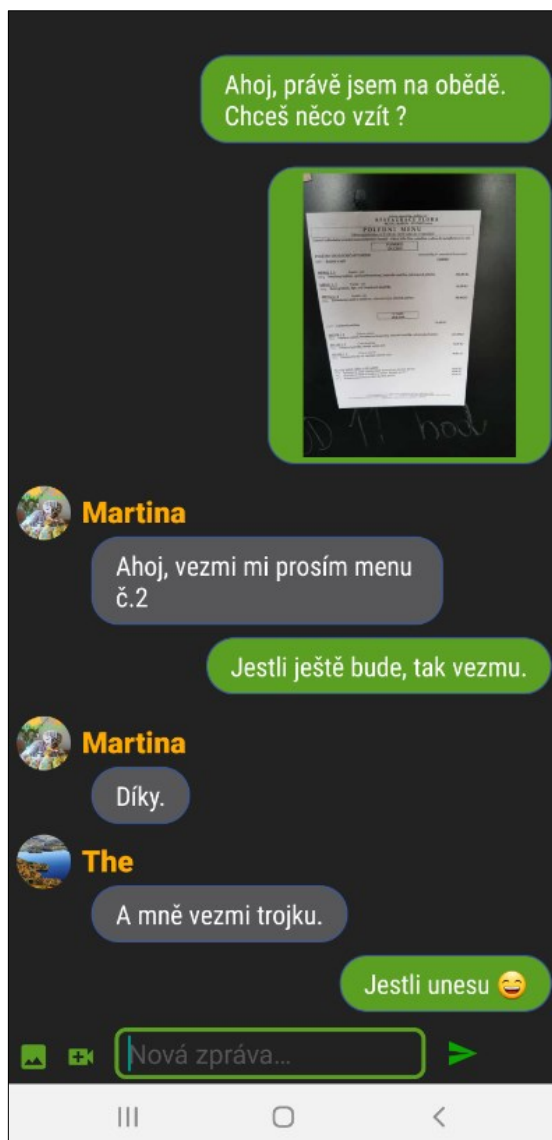
8.3 Chatovací aktivita

Tato aktivita se skládá z prvků *EditText*, *ImageView*, *Button* a *RecyclerView*.

EditText zde slouží pro psaní textových zpráv. *ImageView* slouží pro zobrazení jednotlivých ikoněk. Po kliknutí na patřičnou ikonku se otevře *Intent* pro vybrání fotografie či videa

z interního úložiště chytrého zařízení a automatického odeslání do chatu. *Button* slouží pro odeslání zprávy do chatu.

RecyclerView – jedná se o nejdůležitější prvek celé aktivity. Rozlišuje mezi dvěma pohledy. První pohled je doručená zpráva uživateli. Skládá se z *CircleImageView* a *TextView*, které nám identifikují odesílatele a z textové zprávy nebo fotografie či videa. Druhý pohled je odeslaná zpráva uživatelem, která se skládá pouze z textové zprávy nebo fotografie či videa.

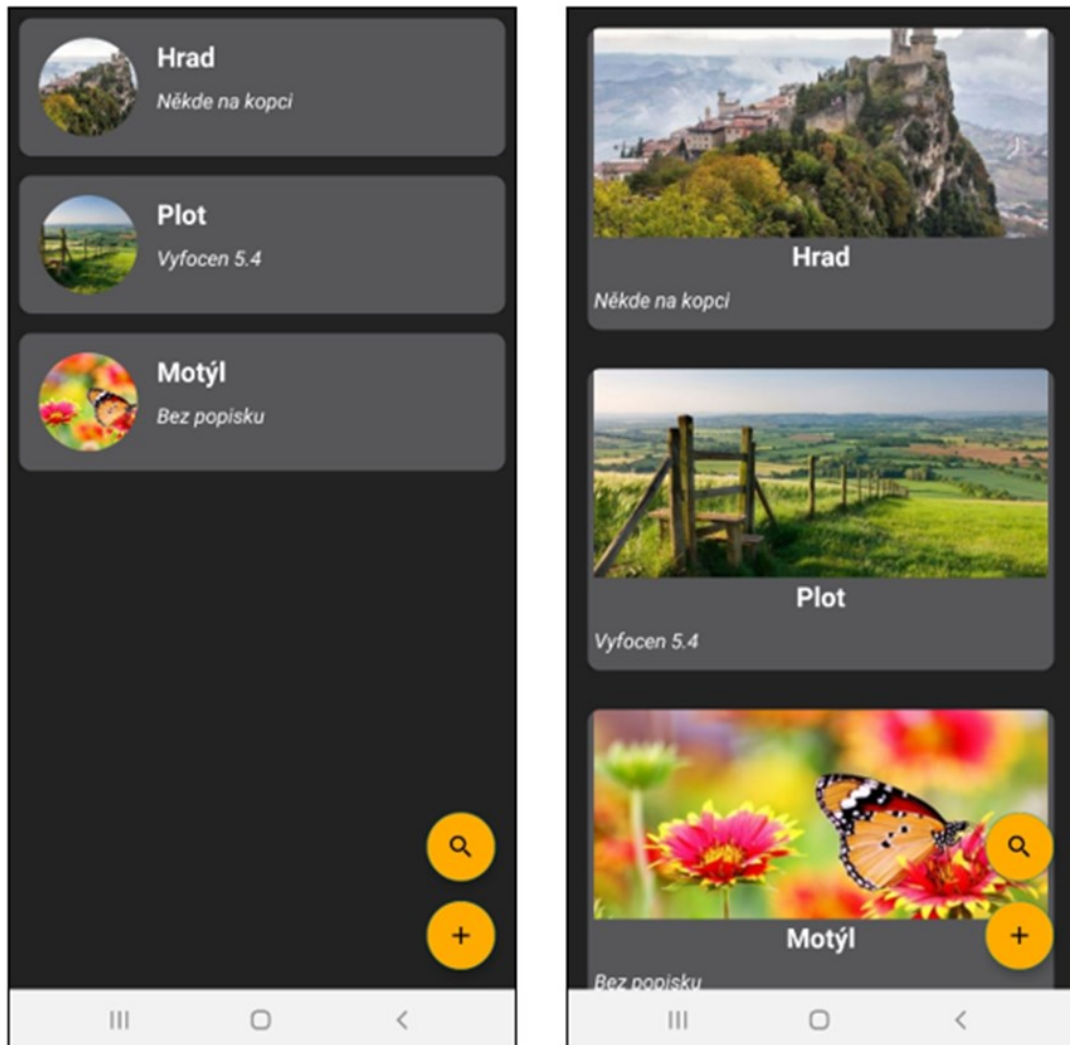


Obrázek 34: Jednotlivé pohledy *RecyclerView*

8.4 Média

Tato aktivita se skládá z *RecyclerView*, 2x *FloatingActionButton* a *ImageView*. *RecyclerView* má opět dva vzhledy – jeden se skládá *CircleImageView* a 2x *TextView* (jeden slouží

k názvu a druhý k popisku) a druhý vzhled z *ImageView* a *2x TextView*. K přepínání mezi jednotlivými vzhledy nám slouží *FloatingActionButton* s ikonkou „lupa“. *FloatingActionButton* s ikonkou „plus“ slouží pro přidání média do *RecyclerView*. *ImageView* slouží pro zpětnou vazbu uživateli, že se něco děje na pozadí. Po načtení tento *ImageView* zmizí.



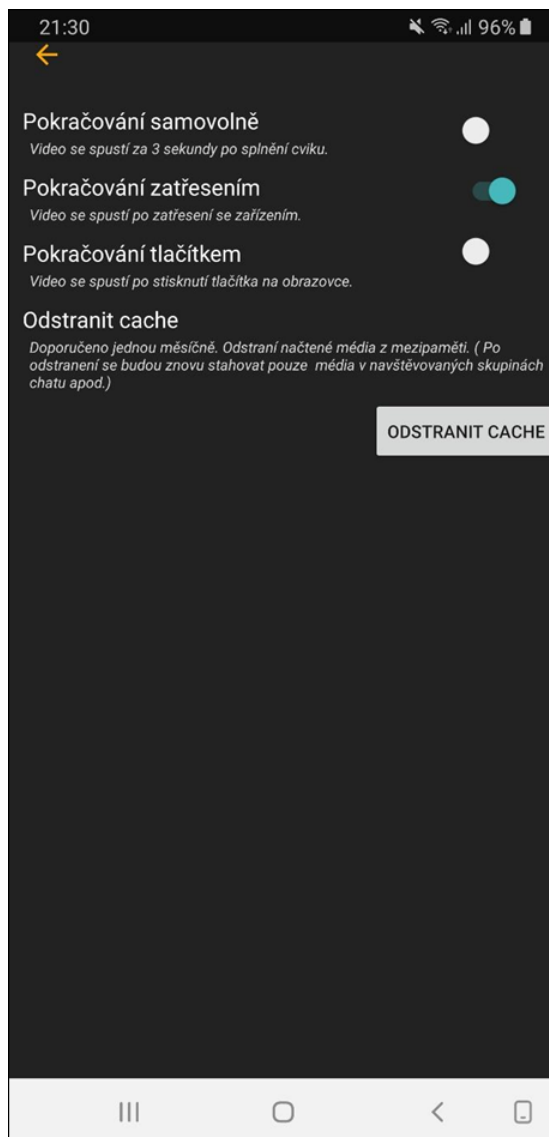
Obrázek 35: Jednotlivé vzhledy

8.5 Nastavení

Po otevření dané aktivity se zobrazí možnosti výběru pro uživatele. Každý uživatel si zvolí jen jednu možnost, a to takovou, která mu nejvíce bude vyhovovat pro pokračování výuky po splnění úkolu. Možnosti jsou tři:

- První je pokračování samovolně, která je nastavena automaticky.
- Druhá je pokračovat po zatřesení

- Třetí je pomocí tlačítka



Obrázek 36: Možnosti výběru

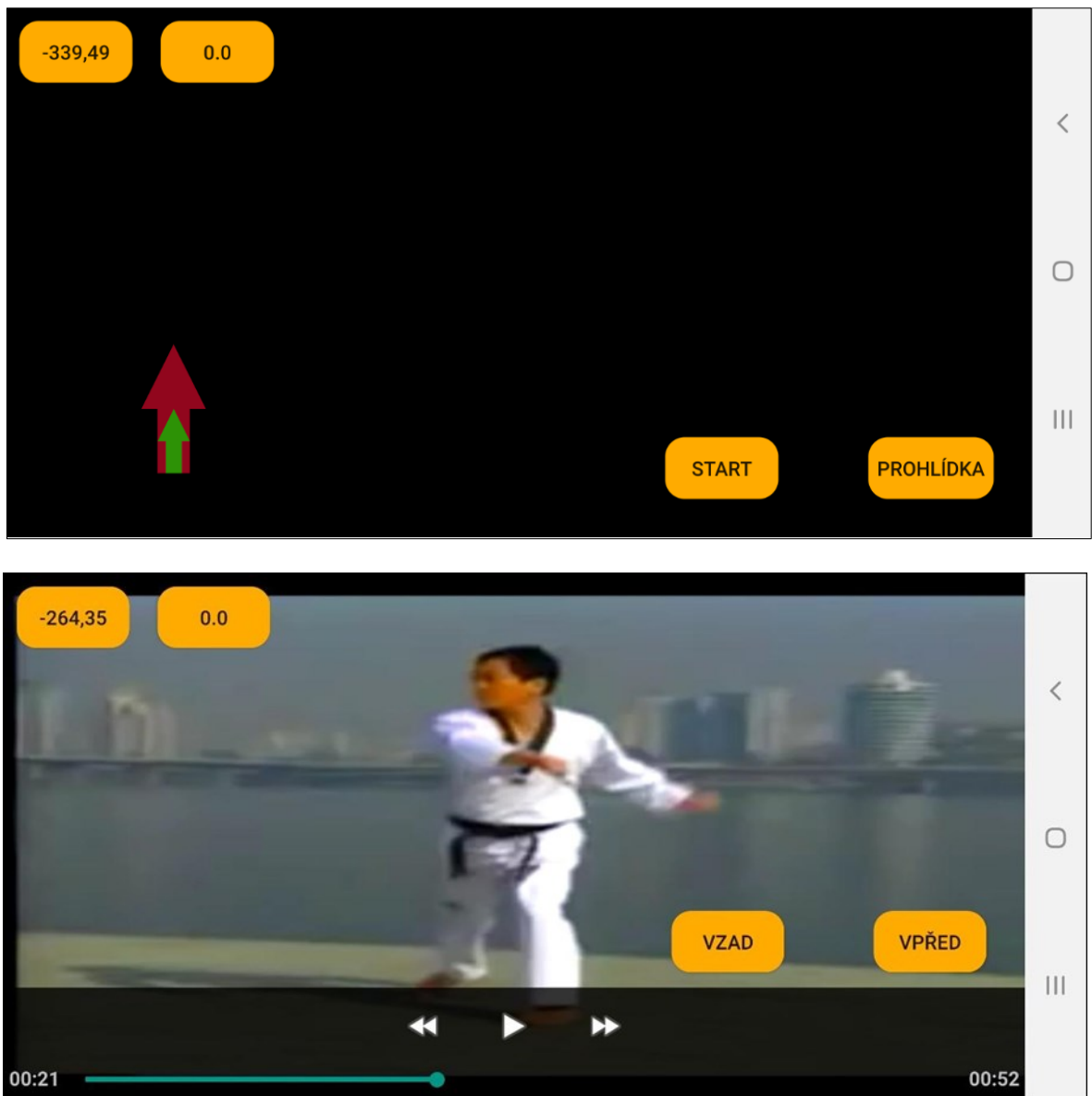
Dále se zde také nachází možnost „Odstranit Cache“, která slouží pro vymazání dočasných souborů v „mezipaměti“.

8.6 Výuková aktivita

Tato aktivita se skládá z *VideoView*, 2x *TextView*, 2x *ImageView* a 6x *Button*.

VideoView umožňuje spuštění videa z interního úložiště chytrého zařízení. První *TextView* obsahuje slovo „Výborně“, které se uživateli zobrazí pouze při správném natočení a druhý obsahuje název daného cviku. *ImageView* slouží pro zobrazení šipky. Červená šipka signalizuje směr správného natočení při výuce a zelená šipka signalizuje aktuální natočení

uživatele. Tlačítko „START“ umožňuje spustit výuku a tlačítko „PROHLÍDKA“ umožňuje pustit, pozastavit, posunout v čase dané video podle uživatele a seznámit se tak s výukovým videem. Tlačítko „VPŘED“, které slouží k posunu videa vpřed a druhé „VZAD“, které slouží k posunu videa zpět na určitý moment v čase videa. V levém horním rohu se nachází dvě tlačítka, která však nemají funkci klasického tlačítka, ale zobrazují aktuální hodnotu kompasu a na jakou hodnotu se má uživatel otočit.



Obrázek 37: Vzhled Výukové aktivity

9 NÁVOD K OBSLUZE

9.1 On-line verze

- Přihlášení
- Otevření boční nabídky
- Výběr – výuková videa
- Výběr videa z nabídky a stisknutí ikonky pro stáhnutí
- Po úspěšném stažení v bočním menu pod nadpisem „*Activity*“ se vybere příslušný název sportu ke staženému videu. Poté se otevře dialog menu, kde se vybere název a stiskne se „*OK*“. Pokud se objeví „*Toast*“ s textem „*chybí video*“, tak došlo ke stažení jiného videa.

9.2 Off-line verze

- Kliknutí na ikonku „přeškrtnuté WiFi“
- Vybrat z nabídky video – lze vybrat pouze označené zelenou „*fajfkou*“
- Po vybrání videa stisknout tlačítko „*OK*“
- Spuštění výukové aktivity

10 TESTOVÁNÍ APLIKACE

Pro testování aplikace bylo využíváno emulátoru integrovaném ve vývojářském prostředí Android Studio. Tímto způsobem byla testována aplikace kromě výukového programu, protože testování této části aplikace je výhodnější na reálném zařízení.

Vyzkoušeny byly různé verze API, různé velikosti a rozlišení obrazovky chytrých zařízení. Testování aplikace probíhalo na reálných zařízeních viz. Tabulka 1.

Tabulka 1: Reálná zařízení pro testování

Název zařízení	Verze Androida	Verze API
Samsung Galaxy S9+	9.0	28
Samsung Galaxy S7	8.0	26
Huawei P9	7.0	24
Huawei P8 Lite	6.0	23

Další využitá možnost pro testování aplikace bez výukového programu byla služba *Google Firebase Test Lab*. Tato služba slouží k automatickému testování na reálných i virtuálních zařízeních. Test je automatický a simulující uživatele aplikace. Na *Obrázku 38* se nachází výsledek tohoto testování reálných chytrých zařízení s různými verzemi Androidu v režimu obrazovky portrét.

Test execution	Duration	Locale	Orientation	Issues
✓ Samsung Galaxy S9+ (US), API Level 26	2 min 14 sec	Czech (Czechia)	Portrait	–
✓ Pixel, API Level 26	5 min 7 sec	Czech (Czechia)	Portrait	–
✓ Nexus 6, API Level 23	5 min 14 sec	Czech (Czechia)	Portrait	–

Obrázek 38: *Firebase on-line Robo test*

Výukový program této aplikace byl testován s reálnými uživateli (viz. *Obrázek 39 A, B, C, D*) a jejich připomínky byly do aplikace zapracovány. Jednalo se o následující požadavky:

- Přidání rotující šipky, která ulehčuje orientaci v prostoru,
- posun videa „VPŘED“ a „VZAD“ při prohlídce,
- zobrazení pouze navštěvovaných skupin,
- odebrání opětovného zadávání hesla pro vstup do již dříve navštívené skupiny,

- přidání více možností pro pokračování ve výuce,
- přidání možnosti cvičení v off-line verzi.



Obrázek 39: Testování aplikace s reálnými uživateli

ZÁVĚR

Cílem předložené diplomové práce byl vývoj aplikace využívající senzorové vybavení moderních telefonů pro rozvoj netradičních pohybových aktivit, která provádí vyhodnocení pohybů uživatele v reálném čase a umožňuje konzultaci s ostatními uživateli aplikace. Tento cíl byl splněn.

Implementace řešení byla konzultována s mnoha uživateli, kteří tuto aplikaci využívají. Nejlépe byla hodnocena možnost chatu a zasílání fotek, což urychluje komunikaci a řešení možných problémů.

Aplikace je umístěna na Google Store pod názvem „Martial Forms“.

Aplikace byla vyvinuta s ohledem na budoucí možnosti rozšíření. Náměty na budoucí vylepšení aplikace mohou vzejít také od uživatelů. Mezi takové rozšíření patří například přidání dalších výukových videí, další možnosti přihlášení (Facebook, telefonní číslo, Instagram atd.), přidání více uživatelských nastavení (změna profilu, vybrat místo uložení výukových videí atd.), přenos hlasových zpráv v chatu, přidání rozpoznání kroku při výuce, rozpoznání hlasu při výuce, vylepšení funkcionality při výuce pomocí RxJava 2.0, přidání možnosti uživateli ihned reagovat na zprávu v chatu atd.

SEZNAM POUŽITÉ LITERATURY

- [1] CHROUST, Martin a Filip KŮŽEL. *Smartphony mají 19 smyslů. Znáte je všechny ?* [online]. 2015, 1 [cit. 2019-05-08]. Dostupné z: <https://www.mobilmania.cz/clanky/smartphony-maji-19-smyslu-znate-je-vsechny/sc-3-a-1329584/default.aspx>
- [2] LACKO, Ľuboslav. *Mistrovství - Android*. 1. Brno: Computer Press, 2017. ISBN 978-80-251-4875-4.
- [3] LACKO, Ľuboslav. *Vývoj aplikací pro Android*. 1. Brno: Computer Press, 2015. ISBN 978-80-251-4347-6.
- [4] KILIÁN, Karel. *Fungují vašemu telefonu všechny senzory? Aplikace Sensors to umí zjistit*[online]. 2017, 2017 [cit. 2019-05-08]. Dostupné z: <https://www.svetandrida.cz/senzory-v-telefonu/>
- [5] ŠKOPEK, Pavel. *Techbox: váš telefon je prošpikovaný senzory* [online]. 2013, 2013 [cit. 2019-05-08]. Dostupné z: <https://mobilenet.cz/clanky/techbox-vas-telefon-je-prospikovany-senzory-12496>
- [6] PRIYADARSHINI, Manisha. *Which Sensors Do I Have In My Smartphone? How Do They Work?*[online]. 2018, 2018 [cit. 2019-05-08]. Dostupné z: <https://fossbytes.com/which-smartphone-sensors-how-work/>
- [7] *Magnetometr – technologie pro mobilní telefony (vědecké okénko)* [online]. 2012, 2012 [cit. 2019-05-08]. Dostupné z: <https://mobilizujeme.cz/clanky/magnetometr-nova-technologie-pro-mobilni-telefony>
- [8] SIDEROPULOS, Igor. *Co je GPS* [online]. [cit. 2019-05-08]. Dostupné z: http://www.gpsnavigace.cz/prispevky/co_je_gps.htm
- [9] *Senzory mobilních zařízení a jejich využití při programování her pro platformu Android* [online]. Opava, 2016 [cit. 2019-05-08]. Dostupné z: http://www.it-spy.cz/wp-content/uploads/2016/11/IT_SPY_2016_paper_56.pdf. Diplomová práce. Filozoficko-přírodovědecká fakulta v Opavě.
- [10] *GOOGLE INC. Source architecture. Sensor Types* [online]. [cit. 2019-05-08]. Dostupné z: <https://source.android.com/devices/sensors/sensor-types.html>

- [11] PROCHÁZKA, David. *GPS prodělá upgrade za 8 miliard dolarů* [online]. 2010, 2010 [cit. 2019-05-08]. Dostupné z: <https://magazin.stahuj.cz/gps-prodela-upgrade-za-8-miliard-dolaru/>
- [12] DOLEJŠ, Jan. *Jak funguje zaměření polohy pomocí GPS?* [online]. 2015, 2015 [cit. 2019-05-08]. Dostupné z: <https://www.svetandroida.cz/gps-princip/>
- [13] HUBR, Ivo. *MĚŘENÍ VZDÁLENOSTI POMOCÍ SENZORŮ MOBILNÍHO TELEFONU* [online]. Brno, 2013 [cit. 2019-05-08]. Dostupné z: <https://core.ac.uk/download/pdf/30299165.pdf>. Diplomová práce. VUT Brno.
- [14] *Arecenze.cz* [online]. Procontent [cit. 2019-05-08]. Dostupné z: <https://www.arecenze.cz/fitness-naramky/>
- [15] KEJDUŠ, Radomír. Chytré motivátory pohybu: test pěti fitness náramků (wearables). *Cnews.cz* [online]. 2014, 2014 [cit. 2019-05-08]. Dostupné z: <https://www.cnews.cz/chytre-motivatory-pohybu-test-peti-fitness-naramku-wearables/>
- [16] KNĚŽÍNEK, Vít. Chytré vložky do bot DigitSole vás naučí, jak správně běhat. *Rungo.idnes.cz* [online]. 2017, 2017 [cit. 2019-05-08]. Dostupné z: https://rungo.idnes.cz/chytre-vlozky-do-bot-gps-sporttester-digitsole-run-profiler-ntp-/behani.aspx?c=A170220_223605_behani_ize
- [17] In: *Fundance.cz* [online]. [cit. 2019-05-08]. Dostupné z: <http://www.fundance.cz/?informace-o-tanecnich-podlozkach,2>
- [18] In: *Diit.cz* [online]. 2018 [cit. 2019-05-08]. Dostupné z: <https://diit.cz/clanek/microsoft-ziskal-patent-na-bezdotykovove-ovladani-jako-ze-sci-fi>
- [19] In: *Ar-tracking.com* [online]. [cit. 2019-05-08]. Dostupné z: <https://ar-tracking.com/applications/motion-capture/>
- [20] STRACHOŇOVÁ, Helena. *Technologie motion capture a její využití v umění a populární kultuře* [online]. Brno, 2015 [cit. 2019-05-08]. Dostupné z: https://is.muni.cz/th/z7quq/Technologie_motion_capture_a_jeji_vyuziti_v_umeni_a_popularni_kulture.pdf. Bakalářská práce. MASARYKOVA UNIVERZITA FILOZOFICKÁ FAKULTA.
- [21] PŘÍHODA, Martin. *Implementace Motion Capture v softwaru Unity* [online]. Plzeň, 2015 [cit. 2019-05-08]. Dostupné z:

<https://dspace5.zcu.cz/bitstream/11025/20336/1/BP%20-%20Prihoda%20-%202015.pdf>. Bakalářská práce. Západočeská Univerzita.

- [22] SINICKI, Adam. *I want to develop Android Apps — What languages should I learn?* [online]. 2017, 2017 [cit. 2019-05-08]. Dostupné z: <https://www.androidauthority.com/develop-android-apps-languages-learn-391008/>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

GIF	Graphics Interchange Format
GPS	Global Positioning System
HTML	HyperText Markup Language
IMT	International Mobile Telecommunications
iOS	iPhone Operating System
ISS	International Space Station
LTE	Long Term Evolution
MCS	Master Control Station
MoCap	Motion Capture
SHA	Secure Hash Algorithm
URL	Uniform Resource Locator
WiFi	Bezdrátová komunikace v počítačových sítích
3G	„třetí generace mobilních telekomunikačních technologií

SEZNAM OBRÁZKŮ

<i>Obrázek 1: Akcelerometr [6]</i>	12
<i>Obrázek 2: Osové uspořádání [5]</i>	13
<i>Obrázek 3: Gyroskop [1]</i>	13
<i>Obrázek 4: Zjednodušené schéma gyroskopu [6]</i>	14
<i>Obrázek 5: Kalibrace a fungování kompasu [1]</i>	15
<i>Obrázek 6: Magnetometr pro mobilní zařízení [8]</i>	15
<i>Obrázek 7: GPS čip, funkce GPS [1]</i>	17
<i>Obrázek 8: Schéma rozmístění družic [12]</i>	17
<i>Obrázek 9: Senzor tlaku [1]</i>	18
<i>Obrázek 10: Senzor intenzity okolního osvětlení [1]</i>	19
<i>Obrázek 11: Senzor přiblížení [6]</i>	20
<i>Obrázek 12: Snímač otisků prstů [1]</i>	21
<i>Obrázek 13: Ukázka měření tepu</i>	22
<i>Obrázek 14: Dostupné senzory (A) a zobrazení dat (B)</i>	24
<i>Obrázek 15: Chytré vložky do bot [17]</i>	26
<i>Obrázek 16: Náhled šipek [18]</i>	26
<i>Obrázek 17: Taneční podložka [18]</i>	26
<i>Obrázek 18: Ukázka senzorů pro hry [19]</i>	27
<i>Obrázek 19: Princip MoCap [20]</i>	28
<i>Obrázek 20: Ukázka aplikace „Karate WKF“</i>	30
<i>Obrázek 21: Ukázka aplikace „Taekwondo WTF“</i>	30
<i>Obrázek 22: Ukázka aplikace „Learn Martial Arts“</i>	31
<i>Obrázek 23: Ukázka aplikace „Martial Arts - Training and workouts“</i>	32
<i>Obrázek 24: Use Case Diagram podle funkčních požadavků</i>	36
<i>Obrázek 25: Ukázka Storage</i>	39
<i>Obrázek 26: Ukázka Authentication</i>	39
<i>Obrázek 27: Ukázka Database</i>	40
<i>Obrázek 28: Ukázka Cloud Messaging</i>	40
<i>Obrázek 29: Ukázka možností Cloud Functions</i>	41
<i>Obrázek 30: Ukázka jedné hodnoty v uzlu „Users“</i>	42
<i>Obrázek 31: ListView (A) a RecyclerView (B)</i>	44
<i>Obrázek 32: Přihlášení uživatele</i>	52

<i>Obrázek 33: Profilová aktivita (A) Menu Profilové aktivity (B)</i>	54
<i>Obrázek 34: Jednotlivé pohledy RecyclerView</i>	55
<i>Obrázek 35: Jednotlivé vzhledy</i>	56
<i>Obrázek 36: Možnosti výběru</i>	57
<i>Obrázek 37: Vzhled Výukové aktivity</i>	58
<i>Obrázek 38: Firebase on-line Robo test</i>	60
<i>Obrázek 39: Testování aplikace s reálnými uživateli</i>	61

SEZNAM KÓDŮ

Kód 1: Reference na určitý uzel v databázi	43
Kód 2: Vytvoření účtu	43
Kód 3: Vytvoření nové View	45
Kód 4: Deklarace proměnné „ <i>uploadCurrent</i> “ pro následné upravování.....	45
Kód 5: Začátek první větve.....	45
Kód 6: Pokračování první větve	46
Kód 7: Konec první větve	46
Kód 8: Vytvoření a zobrazení notifikace	47
Kód 9: Přidání a uložení fotografie.....	48
Kód 10: Systémová služba „ <i>DownloadManager</i> “	50

SEZNAM TABULEK

Tabulka 1: Reálná zařízení pro testování.....	60
---	----

SEZNAM PŘÍLOH

P1 CD-ROM s textem této práce a instalačním souborem vyvinuté aplikace