

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA DOPRAVNÍ**

Jan Soldát

**NÁVRH INTERAKTIVNÍHO FORMULÁŘE PRO MAPOVÁNÍ
BEZBARIÉROVOSTI**

Bakalářská práce

2016



K614..... Ústav aplikované informatiky v dopravě

ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

Jan Soldát

Kód studijního programu a studijní obor studenta:

B 3710 – ITS – Inteligentní dopravní systémy

Název tématu (česky): **Návrh interaktivního formuláře pro mapování bezbariérovosti**

Název tématu (anglicky): Design of interactive form for wheelchair accessibility mapping

Zásady pro vypracování

Při zpracování bakalářské práce se řiďte osnovou uvedenou v následujících bodech:

- analýza stávajících mobilních platforem a technických prostředků mobilních zařízení sloužících jako prostředek pro vytvoření a používání elektronického interaktivního formuláře
- návrh systémového modelu aplikace elektronického interaktivního formuláře s ohledem na funkčnost, technické prostředky mobilního zařízení, způsobu práce s daty a možného budoucího rozšíření
- nadefinování možných formulářových prvků, ze kterých se může interaktivní formulář skládat
- aplikování navrženého řešení na konkrétním formuláři a jeho otestování

Rozsah grafických prací: stanoví vedoucí práce

Rozsah průvodní zprávy: minimálně 35 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)

Seznam odborné literatury: Ila Neustadt, Jim Arlow. UML 2 a unifikovaný proces vývoje aplikací. Computer Press, 2014. ISBN: 978-80-251-1503-9

Ľuboslav Lacko. Vývoj aplikací pro Android. Computer Press, 2015. ISBN: 978-80-251-4347-6

Vedoucí bakalářské práce: **Ing. Jan Krčál, Ph.D.**
Ing. Mgr. Michal Jeřábek, Ph.D.
Ing. Lucie Krčálová

Datum zadání bakalářské práce: **26. října 2015**
(datum prvního zadání této práce, které musí být nejpozději 10 měsíců před datem prvního předpokládaného odevzdání této práce vyplývajícího ze standardní doby studia)

Datum odevzdání bakalářské práce: **25. srpna 2016**
a) datum prvního předpokládaného odevzdání práce vyplývající ze standardní doby studia a z doporučeného časového plánu studia
b) v případě odkladu odevzdání práce následující datum odevzdání práce vyplývající z doporučeného časového plánu studia



doc. Dr. Ing. Tomáš Brandejský
vedoucí
Ústavu aplikované informatiky v dopravě



prof. Dr. Ing. Miroslav Svítek, dr. h. c.
děkan fakulty

Potvrzuji převzetí zadání bakalářské práce.



Jan Soldát
jméno a podpis studenta

V Praze dne..... 26. října 2015

Poděkování

V první řadě bych tímto rád poděkoval všem vedoucím mé bakalářské práce, panu Ing. Mgr. Michalovi Jeřábkovi, Ph.D., panu Ing. Janu Krčálovi, Ph.D., paní Ing. Lucii Krčálové, za perfektní vedení a přístup, obětování svého času pro konzultace a velmi cenné rady a poznatky.

V neposlední řadě bych chtěl poděkovat celé své rodině a přátelům za podporu, především matce Janě, otci Zdeňkovi, sestře Petře a bratrovi Zdeňkovi.

Prohlášení

Nemám závažný důvod proti užívání tohoto školního díla ve smyslu § 60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

V Sepekově dne 17. 8. 2016

.....
Jan Soldát

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta dopravní

NÁVRH INTERAKTIVNÍHO FORMULÁŘE PRO MAPOVÁNÍ BEZBARIÉROVOSTI

Bakalářská práce

srpen 2016

Úplná bibliografická identifikace

Jméno a příjmení autora: Jan Soldát

Název bakalářské práce: Návrh interaktivního formuláře pro mapování bezbariérovosti

Pracoviště: Ústav aplikované informatiky v dopravě

Vedoucí bakalářské práce: Ing. Mgr. Michal Jeřábek, Ph.D.

Ing. Jan Krčál, Ph.D.

Ing. Lucie Krčálová

Rok obhajoby bakalářské práce: 2016

Počet stran: 59

Příloha: CD-ROM se zdrojovým kódem aplikace

Abstrakt: Bakalářská práce se zabývá problematikou návrhu interaktivního formuláře pro mapování bezbariérovosti. První dvě kapitoly jsou věnovány analýze technických prostředků, operačních systémů a současnému stavu mapování. Následná část je věnována návrhu a tvorbě elektronické verze formuláře v operačním systému Android.

Abstract: This thesis deals with the problems of design an interactive form for mapping accessibility. The first two chapters are focused on analysis of technical tools, operating systems and current mapping situation. The subsequent section is devoted to design and creation an electronic version of form in the Android operating system.

Klíčová slova: Mapování bezbariérovosti, interaktivní formulář, Android, Java, XML

Keywords: Mapping accessibility, interactive form, Android, Java, XML

Obsah

Seznam použitých zkratk	6
Úvod	7
Cíl práce	7
1 Mobilní operační systémy a technické prostředky zařízení	8
1.1 Definice mobilních operačních systémů	8
1.2 Nejčastější operační systémy používané v chytrých telefonech	8
1.2.1 <i>iOS</i>	9
1.2.2 <i>Android</i>	10
1.3 Otevřený a proprietární software	10
1.3.1 <i>Proprietární software</i>	10
1.3.2 <i>Otevřený software</i>	10
1.4 Technické prostředky mobilních zařízení.....	11
1.4.1 <i>Dotykový displej</i>	11
1.4.2 <i>Akcelometr</i>	13
1.4.3 <i>Gyroskop</i>	13
1.4.4 <i>GPS</i>	14
1.4.5 <i>Senzor okolního osvětlení</i>	14
1.4.6 <i>Fotoaparát</i>	15
1.4.7 <i>Mikrofon</i>	16
1.5 Aplikace pomáhající OOSPO	16
1.5.1.1 <i>VozejkMap</i>	17
1.5.1.2 <i>Parking Mobility</i>	17
1.5.1.3 <i>Be Me Eyes- Helping blind see</i>	17
2 Současná situace	18
2.1 Používané papírové formuláře pro mapování bezbariérovosti	18
2.2 Výhody a nevýhody papírového formuláře	19
2.3 Aplikace vytvářející formuláře.....	21
3 Systémový návrh aplikace	22
3.1 Požadavky na aplikaci.....	23
3.1.1 <i>Technické požadavky</i>	23
3.1.2 <i>Funkční požadavky</i>	25
3.2 Struktura aplikace a vývojového prostředí	29
3.3 Prvky navrhované aplikace.....	31
3.3.1 <i>Prvky XML</i>	31
3.3.1.1 <i>Prvky sloužící k rozmístění</i>	31
3.3.1.2 <i>Akční prvky</i>	31
3.3.2 <i>Metody pro práci s XML prvky</i>	35
3.4 Postup vývoje aplikace	37
3.4.1 <i>Ukládání dat</i>	42

3.4.2	<i>Práce s uloženými daty</i>	44
3.4.3	<i>Vzhled navrhované aplikace</i>	45
4	Zkouška aplikace v terénu	47
5	Závěr	49
	Seznam použitých zdrojů	51
	Seznam obrázků	56
	Seznam tabulek	58
	Seznam příloh	59

Seznam použitých zkratek

API	Application programming interface
BLOB	Binary Large Object
dp	Density-independent Pixels
EULA	End-User-License-Agreement
GPS	Global Positioning System
ID	Identificator
JPG/JPEG	Joint Photographic Experts Group
MB	Megabyte
MEMS	Micro-Electro-Mechanical Systems
OHA	Open HandsetAlliance
OOSPO	Osoby s omezenou schopností pohybu a orientace
OS	Operační systém
PNG	Portable Network Graphics
SDK	Software development kit
SQL	Structured Query Language
XML	Extensible Markup Language

Úvod

Mobilní telefony jsou v naší společnosti již mnoho let, avšak v posledních deseti letech jsme svědky enormního nárůstu jejich využívání i výkonového růstu mobilní zařízení, které nepochybně souvisí s rozmachem Internetu. Jako počátek velkého pokroku v této oblasti můžeme uvést představení prvního iPhone¹ od firmy Apple, který zbořil stereotypy o tom, jak by měl vypadat mobilní telefon a vůbec poprvé nám představil smartphone tak, jak jej známe nyní. To jest symbióza funkcí z několika předmětů do jednoho. Mobilní telefon se tedy mírně oprostil od své původní funkce striktního telekomunikačního zařízení k zařízení, jenž je velice multifunkční a dokáže zastoupit, či zcela nahradit několik zařízení najednou.

„An iPod, a phone, and an Internet communicator. An iPod, a phone... are you getting it? These are not three separate devices. This is one device. And we are calling it iPhone. Today, Apple is going to reinvent the phone.” [6][7]

Cíl práce

Cílem práce je zanalyzovat stávající mobilní platformy a technické prostředky mobilních zařízení sloužící jako prostředek pro vytvoření a používání elektronického interaktivního formuláře. Dále zanalyzovat současný stav v oblasti mobilních zařízení s ohledem na zadanou tematiku mapování bezbariérovosti a provést návrh systémového modelu aplikace elektronického interaktivního formuláře z hlediska funkčnosti, technických prostředků mobilních zařízení, způsobu práce s daty a možného budoucího rozšíření. Poté nadefinovat možné prvky formuláře a aplikace navrhovaného řešení na konkrétním formuláři a jeho otestování.

¹ MACWORLD SAN FRANCISCO, 9. leden 2007

1 Mobilní operační systémy a technické prostředky zařízení

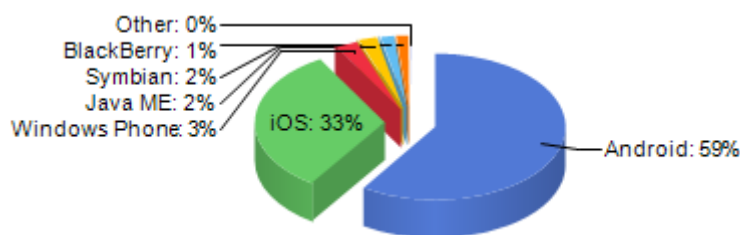
1.1 Definice mobilních operačních systémů

Operační systém spravuje hardware a software daného zařízení a základní úlohy zařízení. Dovoluje nám v jeho prostředí používat další aplikace, které jsou zpravidla podstatně méně náročné. Také zajišťuje neměnné uživatelské prostředí pro aplikace nezávisle na zvoleném hardwaru. Dovoluje tedy uživateli práci s vlastním zařízením. Operační systém je většinou skrytý před uživatelem, který ho tak nemůže přímo ovlivnit. Chytré telefony mají vzhledem ke svým vlastnostem specifické požadavky na operační systém. Tyto vlastnosti například jsou:

- **Velikost a mobilita** – Chytré telefony jsou malé a přenosné, musí tedy vydržet několik hodin bez připojení k napájecí stanici.
- **Načítání z režimu spánku a pohotovost** – Zařízení se nesmí příliš dlouho načítat z režimu spánku a musí být v tomto režimu schopno přijímat hovory a zprávy, mít aktivovaný alarm atd.
- **Omezená paměť** – Operační systém musí být velmi kompaktní, aby zajistil bohatou sadu funkcí při limitované velikosti paměti. [8][12]

1.2 Nejčastější operační systémy používané v chytrých telefonech

Jak ukazuje níže uvedený Obrázek 1 a Tabulka 1, v dnešním rozložení trhu jsou dominantními operačními systémy OS Android od společnosti Google a iOS od společnosti Apple. Na třetím místě je Windows Phone s podstatně menším zastoupením na trhu oproti těmto dvěma. Proto při vytváření naší aplikace má smysl uvažovat pouze o systémech iOS a Android. [9]



Obrázek 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9]

Tabulka 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9]

Operační systém	Zastoupení systému v populaci [%]
Android	59.17
iOS	32.62
Windows Phone	2.72
Java ME	2.26
Symbian	1.66
BlackBerry	1.46
Samsung	0.07
Kindle	0.02
Bada	0.01
LG	0.00
Windows Mobile	0.00

1.2.1 iOS

Dne 9. ledna 2007 představil Steve Jobs světu první iPhone a s ním i nový operační systém. Své jméno, iPhone OS, dostal až o několik měsíců později. Do té doby byl uváděn pouze jako variace systému OS X. Pojmenování iOS nastalo v roce 2010 a jedná se o systém vytvořený společností Apple, který využívají zařízení jako iPhone, iPad, iPod Touch a Apple TV. Inkriminovaný systém je odvozený z Mac OS X a je vymezen jako tzv. proprietární software. Obsahuje čtyři základní vrstvy.[10][11][12]

- **Core OS layer (vrstva jádra systému)** - Zajišťuje nízkoúrovňové funkce jakožto rámce pro bezpečnost a interakci s externím hardwarem.
- **Core services layer (vrstva základních služeb)** – Poskytuje služby požadované vyššími vrstvami.
- **Media layer** - Zaručuje nezbytné grafické, audio a video technologie.
- **Cocoa touch layer** - Rámce zde umístěné jsou často užívány při vytváření aplikací.

Systém iOS obsahuje mnoho základních standardních aplikací, například webový prohlížeč Safari, mediální přehrávač, emailový klient. Nabízí se zde možnost i pro vývojáře, kteří mohou využít vývojovou sadu (SDK) pro vytváření vlastních aplikací. SDK obsahuje rozhraní a nástroje pro vytváření, instalování, spouštění a testování aplikací. Programování probíhá v jazyku Swift nebo Objective-C. Vývojáři mohou své aplikace umisťovat na AppStore, kde zdarma, nebo za poplatek mohou uživatelé tyto aplikace stáhnout. [10][11][12]

1.2.2 Android

Technologie vlastněná společností Google. Za vývojem tohoto otevřeného operačního systému stojí konsorcium zakladatelských členů OHA. OHA je obchodní aliance hardwarových, softwarových a telekomunikačních společností zaměřených na pokrok otevřeného programování pro mobilní telefony. Jedná se tedy o otevřený software, napsaný primárně v jazyku Java a založený na operačním systému Linux. Prvním telefonem, který používal technologii Android, se stal v roce 2008 telefon HTC Dream. Nyní se s výše uvedenou technologií můžeme setkat v chytrých telefonech, tabletech, čtečkách knih a dalších zařízeních. Android má rozsáhlou základnu vývojářů, kteří mohou své aplikace umisťovat do tzv. Obchodu Play (dříve Android Market) a zde je posléze nabízet ostatním uživatelům zdarma, nebo za finanční úhradu. Jedním z hlavních důvodů široké základny vývojářů je zdarma stažitelné vývojové prostředí nazvané Android Studio.[12][13]

1.3 Otevřený a proprietární software

1.3.1 Proprietární software

Termín *proprietary* pochází z latinského slova *proprietas*, což v překladu znamená majetek. Proprietární software je tedy označení pro software, který je vlastněn jedincem nebo společností. Téměř vždy se setkáme s přísnými omezeními na jeho použití. Charakteristickým znakem je skrytí jeho zdrojového kódu před okolním světem. V důsledku toho není odhalena vnitřní struktura. Omezení použití tohoto softwaru jsou obvykle vyjmenována v tzv. dokumentu EULA. EULA je obvykle dlouhá komplexní smlouva. Nejtypičtější příkazy jsou: zákaz tvoření neautorizovaných kopií nebo používání na více počítačích, než je dovoleno.[15][17]

1.3.2 Otevřený software

Pokud je software otevřený, neznamená to pouze volný přístup ke zdrojovému kódu, ale abychom o softwaru mohli říct, že spadá do této kategorie, musí podle Open Source Initiative splnit několik předpokladů.

- 1) **Volná redistribuce** - Licenční podmínky nesmí omezovat jakoukoliv stranu z prodeje nebo rozšíření softwaru dál jako komponentu jiného softwaru nebo společně s jiným softwarem. Licenční podmínky nesmí požadovat za takovýto obchod autorskou nebo jinou daň.
- 2) **Zdrojový kód** - Program musí obsahovat zdrojový kód a musí povolit distribuci jak ve formě zdrojového kódu, tak ve formě spustitelného souboru. Jestliže zdrojový kód není poskytnut, musí být jasně uvedeno jak zdrojový kód, pouze s odůvodnitelnými náklady, získat, a to nejlépe bez nákladů, přes Internet. Zdrojový kód musí být ve stavu, aby programátor mohl upravit program. Záměrně popletený zdrojový kód není povolen. Přejížděvé formy, jako například výstup z preprocesoru nebo překladače, nejsou povoleny.

- 3) **Odvozená díla** - Musí být dovoleny modifikace i další distribuce upraveného programu na základě stejných licenčních podmínek jako originální software.
- 4) **Integrita zdrojového kódu** - Licenční podmínky mohou omezit distribuci upraveného zdrojového kódu, pokud dovoluje šíření opravných souborů se zdrojovým kódem, za účelem úpravy programu během překladu. Musí výslovně povolit šíření softwaru vytvořeného z modifikovaného zdrojového kódu. Může být požádáno, aby upravený software nesl jiné jméno nebo byl označen číslem verze.
- 5) **Zákaz diskriminace osob nebo skupin** - Licenční podmínky nesmí diskriminovat určitou osobu nebo skupinu osob.
- 6) **Zákaz diskriminace podle oboru činnosti** - Licenční podmínky nesmí kohokoliv omezovat v užívání softwaru na základě oboru činnosti.
- 7) **Aplikovatelnost licenčního ujednání** - Práva vztažená k programu se musí týkat všech účastníků, jimž je program redistribuován bez potřeby pro dodání dodatečných licenčních podmínek těmito subjekty.
- 8) **Licenční podmínky nesmí být určeny pouze pro konkrétní softwarový produkt** - Práva spojená s programem nesmí záviset na tom, zda je program součástí určité softwarové distribuce. Licenční podmínky nesmí záviset na tom, je-li program součástí distribuce s jinými programy, nebo ne.
- 9) **Licenční podmínky nesmí zasahovat do právních vztahů k jinému softwaru** - Licenční podmínky nesmí omezovat další software, jenž je distribuován společně s tímto programem spjatým s licenčními podmínkami. Například licenční podmínky nesmí naléhat, aby ostatní programy šířené na stejném hmotném médiu byly také open-source.
- 10) **Technologická neutralita licenčních podmínek** - Žádné ustanovení licenčních podmínek nesmí být založeno na jakékoliv individuální technologii nebo stylu rozhraní.[1][14][15][16]

1.4 Technické prostředky mobilních zařízení

Chytrý telefon se vyznačuje svojí velkou multifunkčností a tím, že dokáže zastoupit více zařízení najednou. K tomu využívá různé technické prostředky.

1.4.1 Dotykový displej

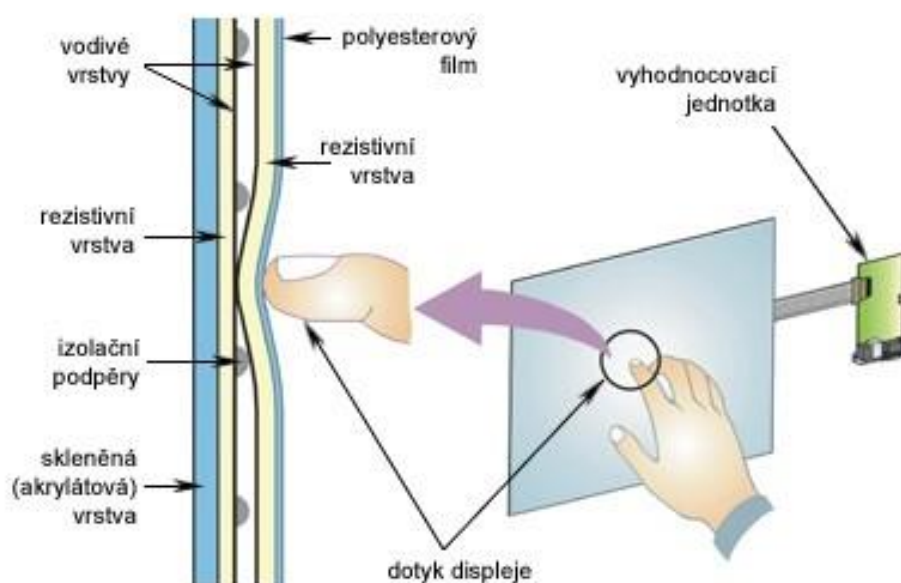
Dotykové displeje se liší podle principu fungování na rezistivní, kapacitní, akustické a infračervené. V chytrých telefonech se používají zejména rezistivní a kapacitní. U naprosté většiny telefonů se používají displeje kapacitní. Displeje rezistivní jsou k nalezení ve starších telefonech nebo levnějších zařízeních. [20]

- **Rezistivní displej** - Je tvořen skleněným panelem a pružnou membránou. Tyto dvě vrstvy jsou odděleny pomocí izolačních podpěr, nejčastěji vzduchovou mezerou. Obě vrstvy jsou

na straně, která sousedí s mezerou, potaženy kovovou vodivou vrstvou. Při dotyku se prohne pružná membrána a elektrovedivé vrstvy se spojí, díky čemuž dojde ke změně proudů a zařízení vyhodnotí polohu dotyku. [18][19]

Tabulka 2: Výhody a nevýhody rezistivního displeje [18][20]

Výhody	Nevýhody
Nízká cena	Průzračnost pouze 80%
Jdou ovládat čímkoli	Odolnost proti mechanickému poškození
Nízká spotřeba energie	Omezené pozorovací úhly

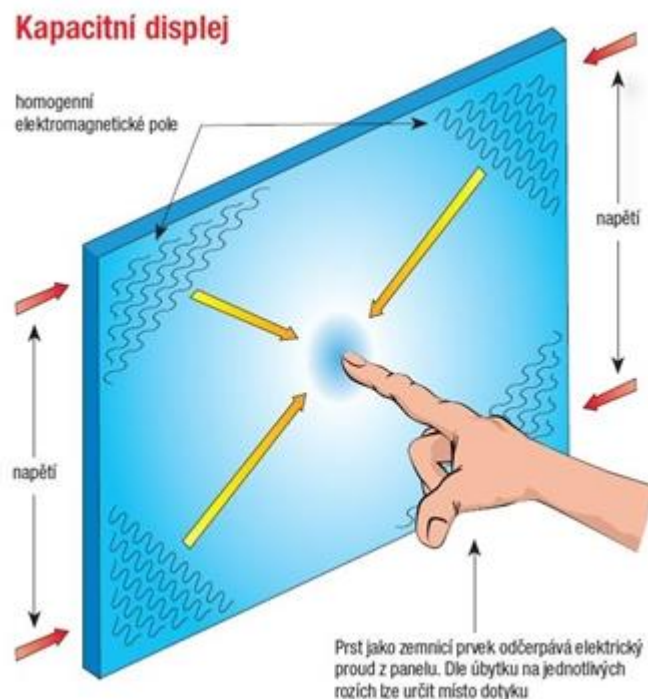


Obrázek 2: Rezistivní displej [21]

- **Kapacitní displej** - Využívá vodivosti objektu, který se dotýká displeje, především tedy prstu. Displej je tvořen sklem, které je potaženo tenkou průzračnou vrstvou vodiče. Při dotyku vodivého předmětu s obrazovkou se naruší elektrostatické pole displeje, které můžeme změřit jako změnu kapacity. [18][19]

Tabulka 3: Výhody a nevýhody kapacitního displeje. [18][20]

Výhody	Nevýhody
Průzračnost zhruba 90%	Nutnost použití elektricky vodivého předmětu
Odolnost proti mechanickému poškození	Dražší výroba



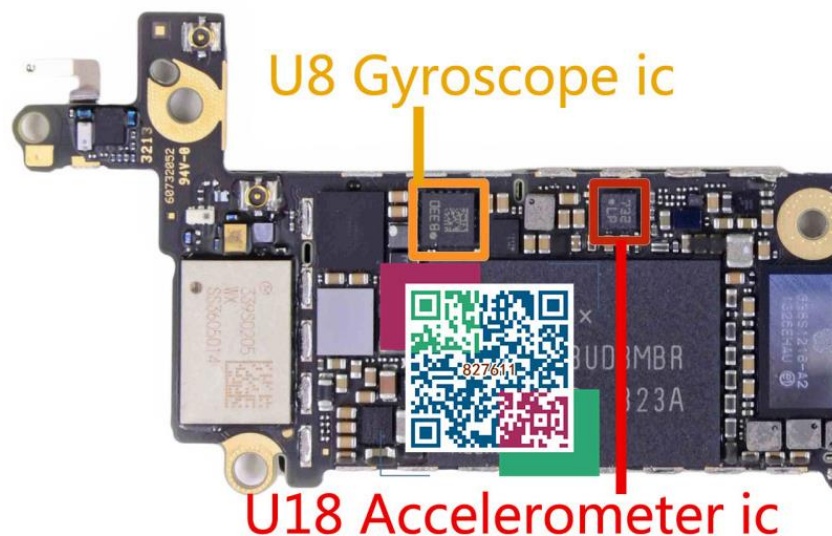
Obrázek 3: Kapacitní displej [22]

1.4.2 Akcelometr

Akcelometr patří do skupiny pohybových senzorů. Měří lineární zrychlení ve dvou osách, dokáže tedy určit směr, kterým se zařízení pohybuje. Využití akcelometru je v různých aplikacích využívajících pohybového ovládání a jedním takovým příkladem může být závodní hra nebo aplikace využívající pohyb telefonu k aktualizaci aplikace. Mobilním telefonům napomáhá ke své funkci piezoelektrický jev. Piezoelektrický jev znamená to, že pokud bude krystal stlačován, budou se měnit polohy náboj nesoucích částic a na protilehlých stranách vzniknou stejně velké inverzní náboje a po připojení elektrod můžeme změřit piezoelektrické napětí. [23][24]

1.4.3 Gyroskop

Funkce gyroskopu je stejná jako v případě akcelometru, měří rotaci či naklonění telefonu. Neměří však zrychlení, nýbrž úhlovou rychlost otáčení a přidává přístroji možnost odhalit pohyb ve třetí ose. Gyroskop je volný rotační setrvačnick. Obsahuje s nepatrným třením v ložiscích se otáčející relativně těžké kolo. Gyroskop nemění bez zásahu vnějších sil směr. Při náklonu měříme změnu kapacitance.[23][25]



Obrázek 4: Ukázka akcelometru a gyroskopu v prostředí chytrého telefonu [26]

1.4.4 GPS

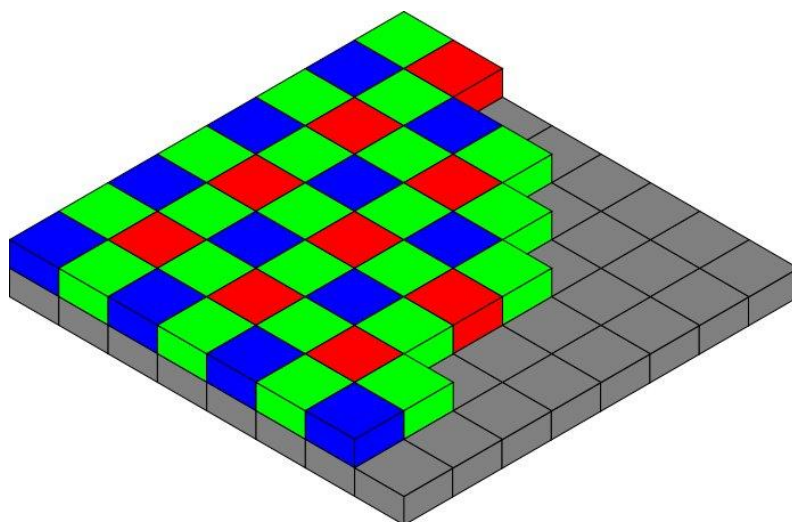
GPS je systém, který dokáže určit polohu zařízení kdekoliv na této planetě. Princip GPS, velmi zjednodušeně, spočívá v tom, že z družice je vyslán signál. Tento signál obsahuje jednak informace o poloze družice, co signál vysílá, jednak i informace o poloze ostatních družic a čas vyslání daného datagramu. Tato informace přijde do našeho zařízení, které je schopno vypočítat vzdálenost zařízení od družice, což společně se znalostí o poloze družice vede k lokalizaci polohy. Jedním z velkých problémů, který se musí řešit, je rozdílná přesnost hodin zařízení a družice. Družice obsahuje velmi přesné atomové hodiny, zatímco naše zařízení ne, proto se musí přistoupit k dalším algoritmům, co tento čas upraví a synchronizují s daty, která přijmeme od družice. [27][28]

1.4.5 Senzor okolního osvětlení

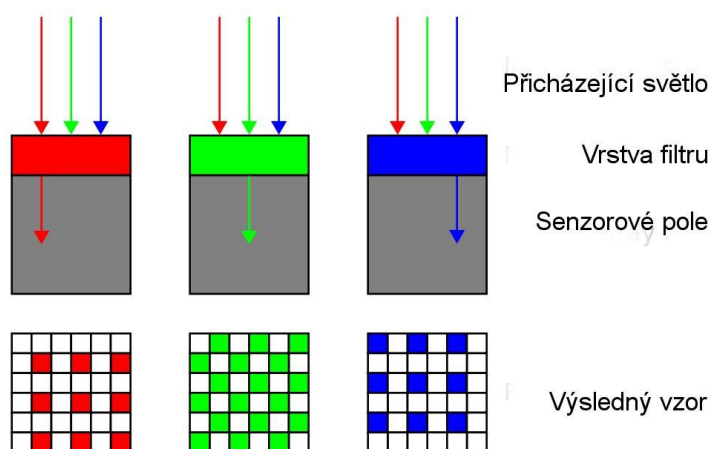
Velkým problémem dnešních chytrých telefonů je krátká výdrž baterie. Pomocí toho senzoru můžeme prodloužit dobu výdrže baterie. Tento senzor snímá úroveň okolního osvětlení, analyzuje jej. Inkriminované hodnocení promítne v úrovni podsvícení displeje. Využívá se fotorezistor, který funguje na principu vnitřního fotoelektrického jevu. [23]

1.4.6 Fotoaparát

Funguje tak, že od objektu odrážející se světlo prochází čočkou a dopadá na světlocitlivou vrstvu. U digitálních fotoaparátů je touto vrstvou obrazový čip, ten následně převádí světlo na elektrický signál. Obrazový čip je plocha pokrytá miniaturními pravidelnými buňkami, jež jsou citlivé na světlo. Tyto buňky označujeme jako pixely² a mohou jich být až miliony, to závisí na rozlišení. Pixely jsou uspořádány v rovnoměrné matici. Princip je takový, že při dopadu světelného paprsku vzniká elektrický náboj. Náboje se dále načítají jako napěťový signál a ten přes A/D převodník konvertuje na digitální údaj o jasu konkrétního bodu. Obraz se však dále ještě dopočítává, protože tato data neobsáhnou všechny obrazové údaje. Aby byl obraz barevný, je před čipem barevný filtr (ten je vytvořen pravidelnou šachovnicí z jednotlivých barev) a každá barva propouští pouze jednu vlnovou délku. Každý pixel tedy dostane informace pouze o jedné barvě a ostatní barevné složky se musí dopočítat.[35]



Obrázek 5: Uspořádání barevných filtrů [36]



Obrázek 6: Průřez snímače [37]

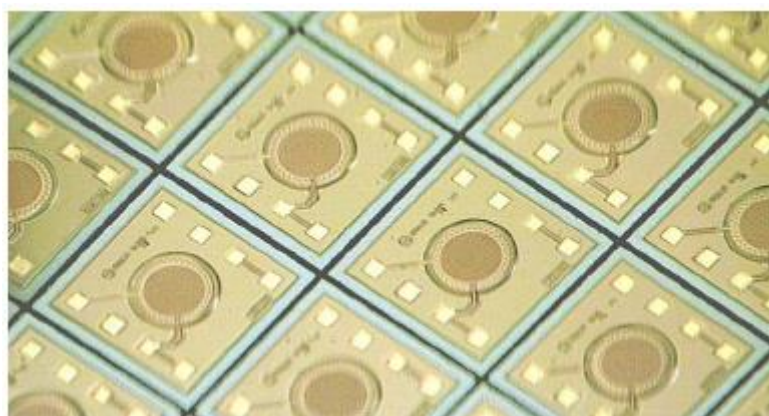
² Picture element = obrazový element



Obrázek 7: Čočka iPhoneu 6s [38]

1.4.7 Mikrofon

V telefonech jsou zabudovány miniaturní mikrofony, jejich velikost je zhruba 3 x 3 x 1 mm. Označují se jako MEMS mikrofony. Jde o velmi malé mikrofony vyrobené z křemíku, které jsou vyráběny jako jedna sjednocená komponenta obsahující tlakově citlivou křemíkovou membránu, sloužící k snímání, tak i integrovaný předzesilovač. Může obsahovat i další pomocné obvody filtrující například šum. Často obsahuje i integrovaný A/D převodník, aby se usnadnilo připojení k digitálním systémům. Tento převodník způsobí to, že na výstupu jsou digitální sériově přenášená data. [34]



Obrázek 8: Fotografie reálné struktury MEMS mikrofonu.

1.5 Aplikace pomáhající OOSPO

Hovoříme o aplikacích většinou nějakým způsobem usnadňující život této skupině lidí. Zde uvádím aplikace, které jsem shledal užitečnými s ohledem na jejich rozmanitost a smysluplnost.

„Osoba s omezenou schopností pohybu a orientace (OOSPO) je definována jako osoba s pohybovým, zrakovým, sluchovým a mentálním postižením, osoba pokročilého věku, těhotná žena, osoba doprovázejícími dítě v kočárku nebo dítě do tří let.“ [42]

1.5.1.1 VozejkMap

Česká aplikace, která je provozována Českou asociací paraplegiků. VozejkMap je databáze bezbariérových míst v ČR. Data jsou vkládána a ověřována komunitou používající tuto aplikaci. Aplikace má okolo jednoho tisíce stažení a je zdarma.[29]

1.5.1.2 Parking Mobility

Aplikace určená k nahlašování neoprávněného stání na místě vymezeném pro lidi s omezenou schopností pohybu. Při nalezení takového prohřešku uživatel pořídí skrz aplikaci 3 fotky a odešle je. Aplikace to nahlásí městu, a to se vypořádá s majitelem automobilu. Dále aplikace slouží k sdílení míst na parkování přístupných pro handicapované spoluobčany. Aplikace pochází ze Spojených států amerických a její počet stažení je okolo deseti tisíc. [31]

1.5.1.3 Be Me Eyes- Helping blind see

Je aplikace založená na živém video přenosu. Nevidomí mohou zažádat o pomoc od zrakově neindisponovaného dobrovolníka, kterému se zavolá a pokud dotyčný přijme žádost, tak mezi oběma stranami vznikne audiovizuální spojení. Během tohoto spojení může vidomý nevidomému říct, co vidí. Jako příklad můžeme uvést informaci o datu expirace nějaké potraviny. Aplikace je pouze na systém iOS a chlubí se více než 300 000 dobrovolníky.[30]

2 Současná situace

2.1 Používané papírové formuláře pro mapování bezbariérovosti

Mapování bezbariérovosti se v současné době provádí pomocí papírových formulářů. Těchto formulářů existuje více různých druhů, pro každou mapovanou oblast jiný formulář (např. parkoviště, přechod pro chodce, komunikace, zastávka MHD, ...). I když by mapování bezbariérovosti měli dělat vyškolení pracovníci či organizace a používat jednotné formuláře v rámci celé ČR, není tomu tak.

Pracovníci provádějící mapování by měli postupovat tak, že si danou šablonu předpřipraví, tj. vyplní předem známé informace jako je název úseku komunikace, jeho kód, kódy sousedících komunikací, GPS souřadnice, mapové schéma atd. Takto předpřipravený formulář si poté vytisknou a vytištěnou kopii si s sebou vezmou do terénu, kde ji náležitě, vzhledem k poznatkům a informacím získaných během samotného mapování, vyplní a na externí zařízení pořídí nutnou fotodokumentaci. Tímto způsobem ručně vyplněný a zpracovaný formulář poté přepíše na počítači do pdf soubor, fotodokumentaci uloží z externího zařízení do počítače a přiloží k formuláři. Tím se zajistí, že formulář je jasně čitelný a nedochází zde k nejasnostem ohledně čitelnosti ručně psaného písma. Takto zpracovaný formulář poté emailem či jinou elektronickou cestou doručí zadavateli práce.

ÚSEK KOMUNIKACE

(uveďte, které 2 stavební objekty nebo úseky komunikací tento úsek komunikace spojuje, ve směru provádění měření)

Kód úseku komunikace (ve tvaru Kx.y, kde x je zvolené číslo celé měřené lokality a y je zvolené číslo tohoto měřené úseku komunikace)

Kódy sousedících objektů / komunikací (kódy sousedících stavebních objektů (např. přechodu pro chodce, zastávky MHD atd.) nebo úseků komunikací, které jsou na počátku a na konci tohoto měřené úseku komunikace)

na počátku úseku komunikace na konci úseku komunikace

Součást měřené trasy / lokality

GPS souřadnice (počátek měřené úseku)
 (konec měřené úseku)

Mapové schéma (2x klikněte na pole a vložte obrázek s mapovým schématem)

Měření provedl(i) dne

Délka úseku komunikace (měřte v ose pěší trasy; v metrech; s přesností na 0,01 m u délky do 3 m; s přesností na 0,1 m u délky od 3 do 10 m a s přesností na 1 m u délky nad 10 m)

Typ komunikace

- pás pro pěší (chodník)
- stezka pro chodce (označeno značkou C07)
- stezka pro chodce a cyklisty (označeno značkou C09 nebo C10)
- nebezpečná komunikace (pěšina, polní cesta...)

Obrázek 9: Ukázka jedné strany papírového formuláře [41]

2.2 Výhody a nevýhody papírového formuláře

Zde již lze odhalit několik míst na optimalizaci tohoto procesu. Pokud odpadne nutnost tisknutí si formuláře předem, ušetříme čas, protože pracovník bude moci vše v aplikaci zapsat až na místě. Odpadne mu také nutnost s sebou vozit další zařízení jako je fotoaparát. Zjednoduší se tedy příprava a zmenší se možnost eventuálního zapomenutí, a to jak předvyplnění formuláře, tak jednoho z nástrojů nutných k mapování (vytištěný formulář, tužka, fotoaparát), protože budeme potřebovat pouze jedno zařízení. Použitím mobilní aplikace odpadne práce s následným přepisováním do elektronické podoby. Vše budeme zapisovat přímo v aplikaci včetně fotodokumentace, kterou také nebude třeba přesouvat z fotoaparátu do počítače

a následně vkládat do pdf, ale jednoduše jedním tlačítkem, v předpřipraveném okně na fotku, přejdeme z aplikace do rozhraní fotoaparátu, daný problém vyfotíme a při spokojenosti s pořízenou fotografií odsouhlasíme vložení do právě tvořeného formuláře. Jelikož se formuláře dají přímo do pdf souboru vyplňovat, můžeme mluvit o jejich interaktivnosti, a dokonce i o mobilitě, v případě, že by si pracovník provádějící mapování si s sebou vzal notebook a tyto formuláře na místě měření do pdf souboru vyplnil. Takové řešení je ale krajně nepraktické, těžko totiž nastane situace, kdy by skutečně pracovník šel mapovat se svým notebookem. Notebook je přeci jen většinou relativně těžké zařízení, vyplňování během takového průzkumu by bylo více než nepohodlné a v praxi se takovéto řešení nepoužívá. Navíc stále neodpadla nutnost nošení zařízení navíc, v podobě fotoaparátu.

Výhodou papírového formuláře je bezesporu jeho jednoduchost. Pokud je vytištěn, nemůže se nám stát, že by nebyl dostatečně čitelný. Při pádu papírového formuláře na zem nehrozí, jako v případě pádu mobilu, nebezpečí rozbití celého přístroje a s tím i spojená ztráta naměřených dat. Není potřeba mobilního zařízení, které by poskytovalo prostředí pro aplikaci, ale pouze papír a tužka. Také proto, že u papírového formuláře, na rozdíl od aplikace, nelze ohlídat, zda byla všechna pole správně vyplněna nebo, zda vůbec. Některá pole mají specifický tvar, například: Kód úseku komunikace, toto pole by se mělo vyplnit ve tvaru $Kx.y$, kde x je zvolené číslo celé měřené lokality a y je zvolené číslo tohoto měřeného úseku komunikace. V aplikaci můžeme uživateli říci, že vyplnil pole špatně nebo nám toto prostředí, umožní mít rovnou pole přednastavené ve správném formátu. Interaktivní řešení pomocí aplikace také může ohlídat, že dané políčko, které je nutno vždy vyplnit, jako například již zmíněný kód úseku komunikace, bylo vyplněno. Předejde se tak tedy mnoha chybám, které mohou nastat a v papírové formě je nelze ohlídat.

Kód úseku komunikace <input type="text"/>	(ve tvaru $Kx.y$, kde x je zvolené číslo celé měřené lokality a y je zvolené číslo tohoto měřeného úseku komunikace)
--	---

Obrázek 10: Ukázka pole: Kód úseku komunikace

Celkově vzato je interaktivní mobilní aplikace výhodnější volbou. Takovéto řešení nám umožní skloubit více zařízení do jednoho, ušetří čas a peníze, je skutečně interaktivní a mobilní. Zpřehlední práci a zajistí omezení chyb z nepozornosti, vzniklých při vyplňování formuláře, bez kontroly vyplnění dat. Umožní nám u sporných bodů formuláře použití nápovědy, což opět zpřesní a zjednoduší práci pracovníkům provádějící průzkum. Tuto premisu můžeme potvrdit tím, že ze strany organizací zabývajících se touto problematikou je o vznik podobné aplikace velký zájem.

2.3 Aplikace vytvářející formuláře

Relativně velké množství aplikací slouží k víceméně stejnému účelu, tj. konstrukce formuláře a jeho uložení. Tyto formuláře nám ovšem nedovolí si prostředí libovolně, na základě svých požadavků, preferencí a cílů upravovat. Vybral jsem jeden příklad za všechny, protože ostatní aplikace jsou skoro identické a i důvody pro jejich nevyužití se shodují. Jelikož výše zmíněný formulář má některé specifické požadavky, není vhodné využívat těchto aplikací. Těmi požadavky může být například možnost focení přímo z aplikace, možnost nápovědy nebo pouhé grafické zpracování, které by mělo být vytvořeno tak, aby bylo v terénu co možná nejpřehlednější a nejintuitivnější.

Forms – Aplikaci Forms jsem za příklad vybral z důvodu velkého počtu stažení. Stáhlo si ji již přes 100 000 uživatelů. Je to konstruktor formulářů podporující mnoho typů dat jako je text, číslo, email, podpis, lokace atd.... Možné je zařadit i pole, která jsou viditelná pouze za splnění určité podmínky. Aplikace podporuje offline vyplňování i uložení rozdělaného formuláře a doplnění později. Při použití bezplatného účtu formuláře lze podávat z jednoho zařízení a rozposílat pomocí emailu či stáhnout online databáze na jejich stránkách.[32]

Tuto aplikaci, stejně jako i ostatní aplikace, jsem se rozhodl nevyužít, protože nelze přizpůsobit do problematiky mapování bezbariérovosti, uživatel nemá možnost nápovědy u prvků, kde si není jistý jejich správným vyplněním.

3 Systémový návrh aplikace

Při systémovém návrhu aplikace jsem postupoval tak, jak je uvedeno níže. Návrh aplikace byl založen na papírových formulářích, ty jsem tedy detailně zanalyzoval. Dále jsem na základě tohoto formuláře, ale i kontaktu s praxí, zastoupenou pracovníky provádějící mapování nejenom z Fakulty dopravní, ale i z organizací zabývajících se mapováním bezbariérovosti v ČR, stanovil požadavky, co by měla aplikace splňovat.

Vzhledem k množství telefonů s operačním systémem Android, viz Tabulka 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9], jsem vybral jako operační systém právě tento.

Dále si bylo třeba uvědomit logiku takovéto aplikace. Je potřeba, aby uživatel měl možnost zadat data, která se posléze uloží do předem definované struktury a dále je může zobrazit ve stejné struktuře a opravit v případě nějakých nepřesností. Za pomoci různých návodů, tutoriálů a příkladů aplikací jsem získal potřebné poznatky a znalosti nutné pro naprogramování navrhované mobilní aplikace.

Bylo důležité vybrat nejvhodnější řešení použitelná na navrhovanou aplikaci, tu naprogramovat a rozhodnout, jak ukládat naměřená data, nakládat s obrázky a různými prvky aplikace, jak prvky zpět načítat, pomocí kterých prvků vytvořit seznam uložených formulářů, které prvky jasně a nezpochybnitelně identifikují daný formulář, jaké technické vybavení využít a co je kvůli své složitosti naprogramování spíše lepší nevyužívat...

Po těchto otázkách bylo třeba danou aplikaci vytvořit a zkontrolovat, že skutečně funguje tak, jak má. Zda pole, která ukládám, se shodují s poli, které zobrazuji atd. Po vyzkoušení je možné odhalit některé chyby a nedokonalosti degradující dojem z aplikace. Může to být špatné rozložení prvků, chyby v ukládání, chyby v zobrazení, špatné grafické rozložení, nedostačený kontrast, malé písmo. Existuje mnoho chyb, které mohou po vyzkoušení aplikace být objasněny, a to i přesto, že při vývoji aplikace používám k jejímu spuštění mobil, takže na mnoho chyb narazím již během vývoje. Největším problémem, který se při zkoušení již dokončené aplikace vyskytl, byl problém se zpětným zobrazením obrázků. Po úspěšném uložení nešlo formulář s takovými obrázky zobrazit, protože velikost současně přenášených dat byla větší než 1 MB. Pro obrázky, uložené v datovém typu Blob, se nabízela tato řešení:

- Snížit jejich kvalitu během komprese do bytové řady.
- Nedovolit uživateli vložit obrázky větší než je určitá velikost, v tomto případě 1 MB.
- Kombinace těchto dvou metod.

Takovéto opatření by však mělo velký vliv na kvalitu obrázku. Nehledě na přidání práce uživateli. Řešení, které bylo zvoleno, je takové, že do příslušného textového řetězce String se uloží adresa obrázku. Tím se tedy značně sníží velikost přenášených a zpracovávaných dat.

Generální zkouška proběhne přímo v terénu, kde bude aplikace otestována a na tomto základě bude možno aplikaci opravit a dále rozvíjet, rovněž také získat nové požadavky a podmínky na vylepšení a rozšíření.

3.1 Požadavky na aplikaci

Při tvorbě takovéto aplikace na ni vzniká velké množství požadavků ze všech stran, jejichž průnikem získáme to nejlepší možné řešení. Ačkoliv některé požadavky jsou spíše utopického rázu, je třeba je všechny zohlednit a promyslet. Požadavky jsem rozdělil na technické a funkční. Máme požadavky na to, co má zařízení obsahovat, ale také na to, co by mělo umět.

3.1.1 Technické požadavky

- **Výběr systému** – Při návrhu mobilní aplikace je potřeba vyřešit, na který operační systém bude navržena. Je třeba zvážit všechna negativa a pozitiva. Při pohledu na výše zmíněný Obrázek 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9] připadá v úvahu aplikaci tvořit pouze na dva operační systémy, a to iOS nebo Android, přičemž Android je hojněji využívaný systém. To je tedy první a velmi důležitý, důvod proč vybrat operační systém Android. Dalším důvodem je, že systém iOS je dostupný pouze na zařízeních od společnosti Apple, kdežto systém Android se objevuje v telefonech obrovské škály výrobců. Při návrhu je potřeba také myslet na ekonomické hledisko, telefony od společnosti Apple jsou velmi nákladné a značně to tedy omezuje možnost organizací zabývajících se problematikou mapování, jejich nákup. Zvláště pokud za razantně nižší cenu lze koupit telefon s podobnými, ne-li lepšími, technickými parametry s operačním systémem Android. Android má také výborné vývojové prostředí v podobě Android studia a širokou obec vývojářů, z čehož plyne, že na programování v tomto prostředí existuje mnoho tutoriálů, návodů a příruček. V neposlední řadě jsem vlastník telefonu s operačním systémem Android a bylo by tedy pro mě velmi nepraktické a obtížné vytvářet a testovat aplikaci určenou na systém iOS.
- **Verze** – Zařízení je určeno na chytré mobilní telefony. Abych předešel problémům, že aplikace nebude fungovat na starších zařízeních, je potřeba zvolit správné SDK, čímž zajistíme schopnost starších strojů využívat naši aplikaci. Nemusíme však volit verzi 1.0., to by bylo spíše kontraproduktivní, protože tyto počáteční verze neobsahují tolik funkcí, jako verze novější a především už se téměř nevyskytují mezi uživateli. Je tedy krajně nepravděpodobné, že organizace by pořídila telefon, který by byl takto starý. Z tohoto

důvodu jsem vybral API 16: Android 4.1 (Jelly Bean), tato verze by měla pokrýt 95,2% uživatelů.[4]

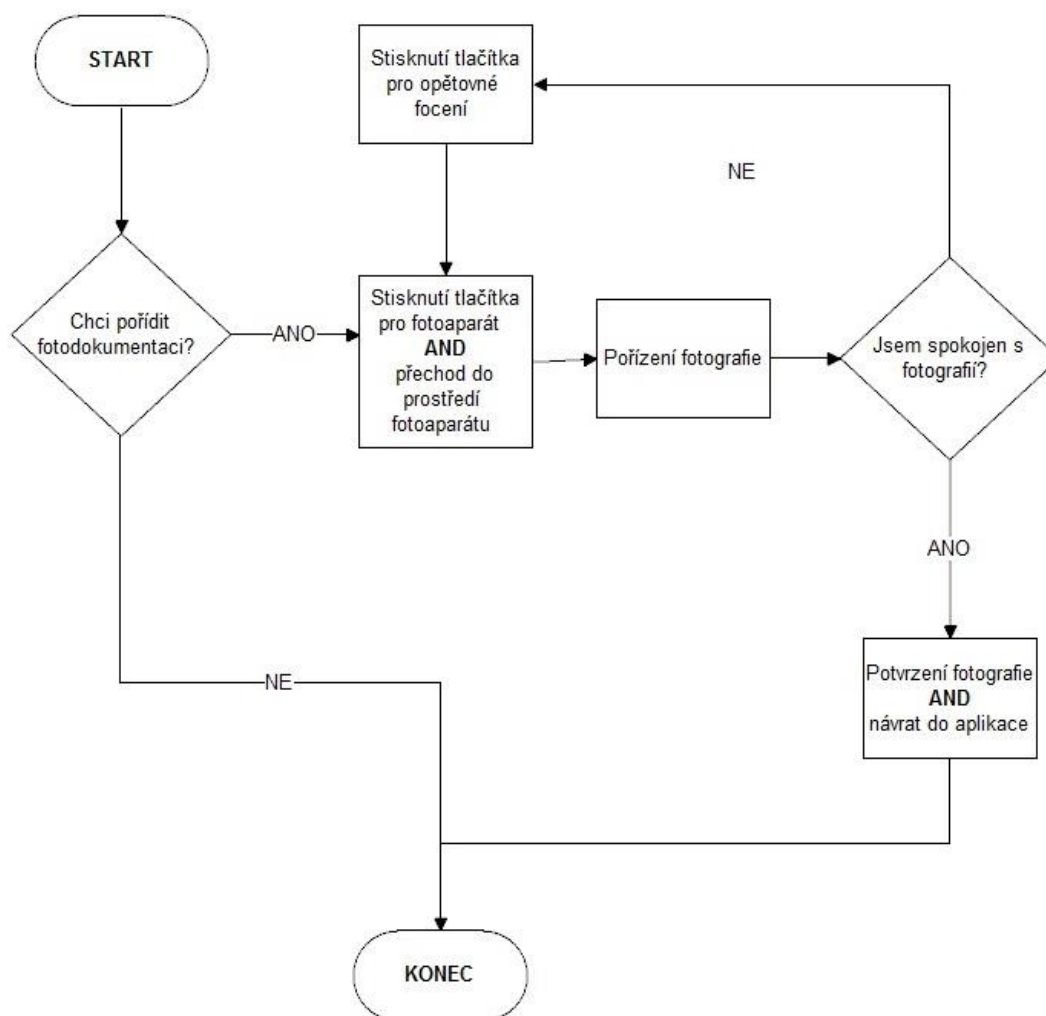
- **Fotoaparát** – Návrh takovéto aplikace bez požadavku na zabudovaný fotoaparát v chytrém zařízení by byl téměř bezpředmětný. Tento požadavek přinejmenším, s trochou nadsázky, odlehčí pracovníkovi batoh. Fotoaparát by měl být v rámci možností kvalitní, tj. po vyfocení fotodokumentace nesmí být problém s nekvalitou fotografie. Ostrost fotografií, souvisí i s pracovníkovou schopností fotit, proto je velmi výhodné, že fotka lze v momentě, kdy se pracovníkovi nebude zdát dostatečně kvalitní, znovu nafotit.
- **Mikrofon** – Důležitým požadavkem je i mikrofon. Pokud v aplikaci bude možnost hlasové poznámky, je třeba, aby telefon měl mikrofon. Není zapotřebí, aby mikrofon byl té nejvyšší kvality, nejedná se zde o studiové nahrávání hudby. Jde především o to, aby v něm nahraný hlas byl zřetelný a aby z důvodu šumu nezaknuly některé důležité informace řečené pracovníkem. Stejně jako u fotoaparátu jsou v dnešní době mikrofony u neporouchaných zařízení dostačující kvality.
- **Dotykový displej** – Celá aplikace stojí na dotykovém ovládní. Je tedy naprosto zřejmé a nutné, aby byl použit chytrý telefon s dotykovým displejem. V současné době ani systém Android jiný telefon nepodporuje. Aplikace je tvořena tak, aby byla ovládána pomocí dotyku, proto je zde požadavek na buďto rezistivní, nebo kapacitní displej. Přičemž výhoda rezistivního displeje spočívá v tom, že není nutné se dotýkat přímo prstem, ale lze využít i nevodivý materiál nebo v zimě použít klasické rukavice. Ovšem rezistivní displej má horší průzračnost displeje a na přímém slunci tedy bude hůře čitelný. Displej kapacitní je na slunci čitelný o mnoho lépe. Tento displej však požaduje, díky principu na kterém funguje, dotyk vodivého předmětu, proto v zimě nelze využít klasické rukavice. Tento problém se dá vyřešit pomocí speciálních rukavic určených k používání kapacitního displeje. Kapacitní displej je tedy lepší volba. V dnešní době ovládá drtivým poměrem současné displeje v chytrých telefonech.
- **Další technické vybavení** – Od telefonu požadujeme, aby obsahoval i další technické zařízení jako je gyroskop, akcelometr nebo GPS.



Obrázek 11: Blokové schéma technických prostředků

3.1.2 Funkční požadavky

- **Interaktivnost** – Jak už z názvu vyplývá, chytrý telefon je multifunkční zařízení, které dokáže obsáhnout několik zařízení najednou. Proto velkým požadavkem je co možná nejvíce využít potenciál tohoto zařízení. Tento potenciál se například skrývá ve fotoaparátu - proč bychom měli používat externí fotoaparát, když dnešní mobilní telefony mají vestavěné fotoaparáty na srovnatelné úrovni, jako je běžný digitální fotoaparát. Odpadne tím i následná práce s přesouváním fotodokumentace z fotoaparátu a vkládáním do formuláře. V případě aplikace se uloží cesta k souboru do databáze, a tak bude fotodokumentace spjata s daným formulářem. Navíc focení v aplikaci probíhá pomocí jednoho dotyku, kterým přejdeme do režimu fotoaparátu. Druhým dotykem pořídíme fotografii a třetím ji bud potvrdíme, nebo vymažeme a nafotíme znovu. Pokud fotografii odsouhlasíme a poté si při pohledu na její náhled uvědomíme, že byla nafocena špatně, případně bylo omylem zmáčknuto tlačítko potvrzení, je tu možnost ji nafotit znovu.

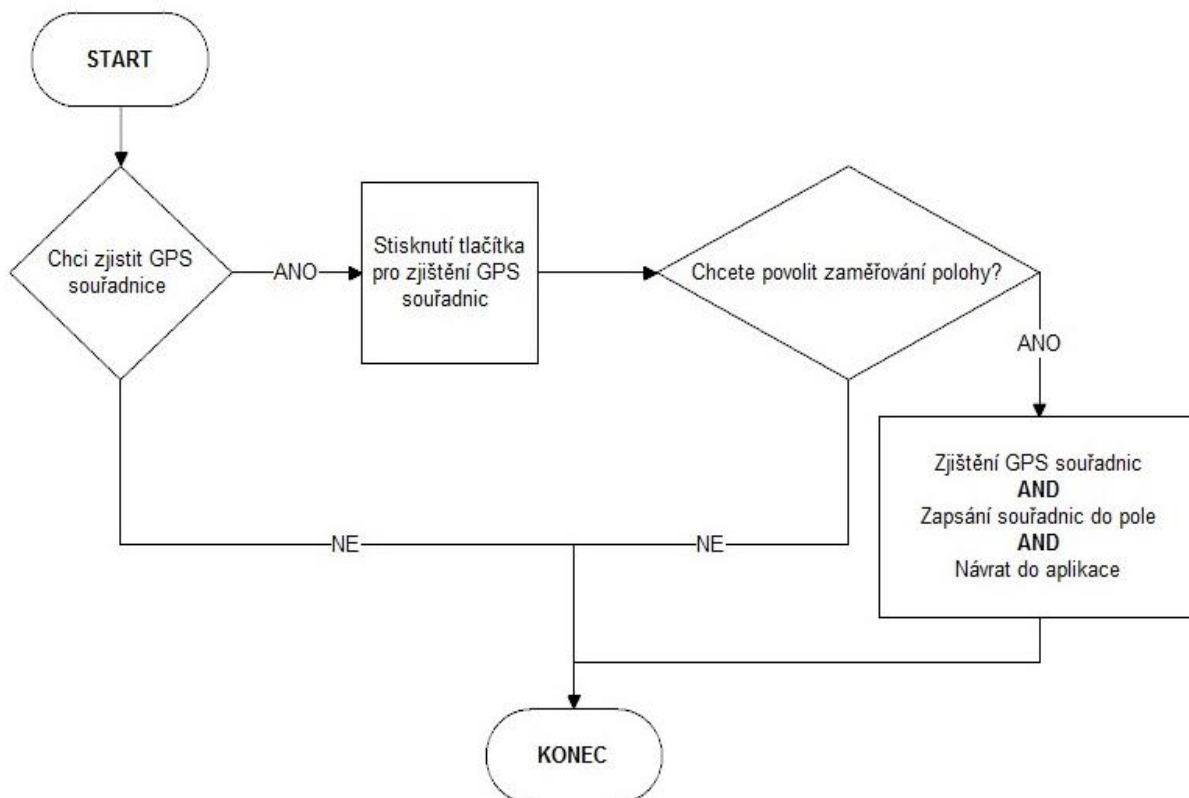


Obrázek 12: Vývojový diagram: Pořízení fotografie



Obrázek 13: Ukázka pořízení fotografie v prostředí aplikace

Velkým lákadlem je také využití GPS navigace, to zapříčiní zjednodušení práce pro pracovníky. V takovém případě není třeba hledat GPS souřadnice předem z mapových podkladů, ale pracovník pouze přijde na místo začátku měření, stiskne tlačítko (případně ještě zařízení povolí vyhledávání polohy) a zařízení samo určí zeměpisné souřadnice, vypíše je do pole a umožní jejich uložení.



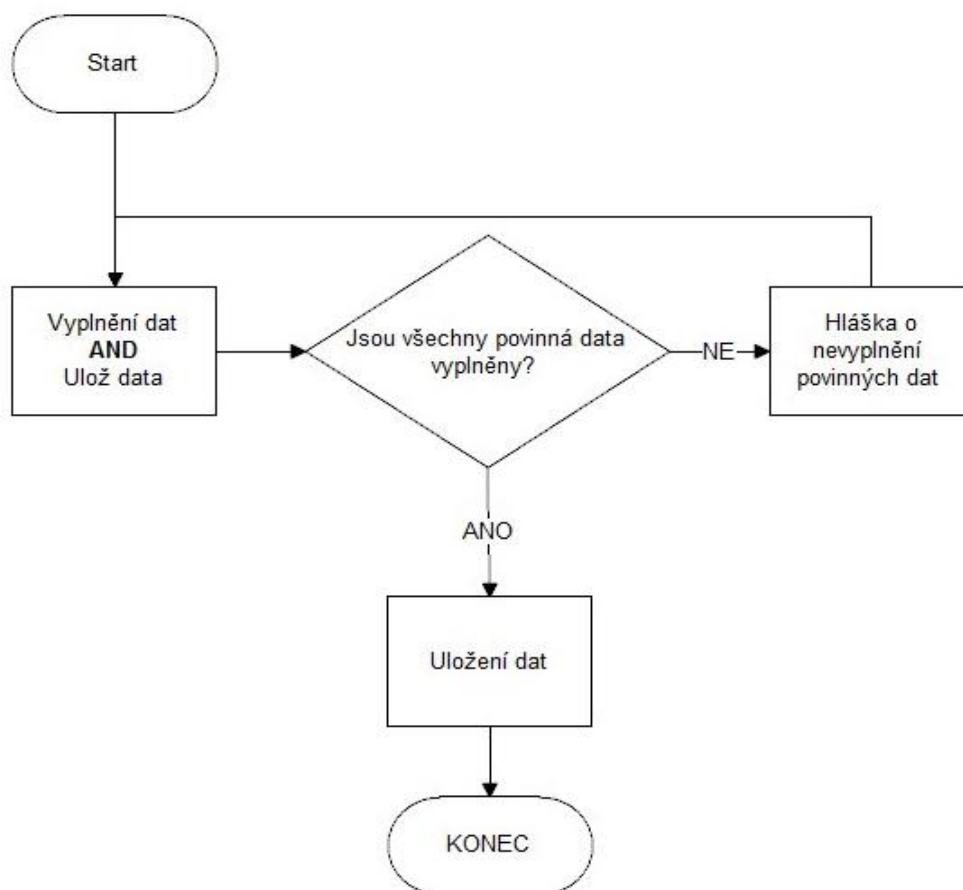
Obrázek 14: Vývojový diagram: Postup zjišťování GPS souřadnic

Požadavkem na interaktivnost je také využití mikrofону, jehož použitím je možné značně zkrátit formulář, protože poznámky, které v papírové formě musíme psát, je možné nahrát jako zvukový záznam a uložit cestu k tomuto souboru do databáze. Pokud budeme mít možnost zvukovou poznámku nahrát, je i vhodné zvukovou poznámku přehrát a zkontrolovat tak tedy, zda bylo vše důležité správně řečeno. Pracovník tedy může být obsáhlejší než při vyplňování textového pole a neomezuje ho velikost daného pole. Úskalí zvukové poznámky je, že zvukový záznam je hůře zpracovatelný/ využitelný co se datových informací týče. Pro práci s ní by ji zřejmě musel pracovník přepsat do textové podoby. Požadavky na interaktivnost také zahrnují specifické formulářové prvky. Těmito prvky můžeme rozumět například tlačítka, různé druhy polí, Radio Buttony, Checkboxy, Spinnery atd. Obecně řečeno tlačítka, která po stisknutí provedou nějakou činnost. Tato tlačítka mohou sloužit například k uložení dokumentu, přechodu na jinou obrazovku, použití fotoaparátu, začátek a konec nahrávání hlasu atd.

Požadavek zabývající se interaktivitou je i možnost u staničení různě dlouhého úseku. Velikost tohoto úseku nastaví uživatel s ohledem na délku úseku, kterou měří. Nemusí totiž vždy využít předem nastavených 100 metrů. Je velmi pravděpodobné, že někdy bude potřebovat třeba jen 70 metrů. V takovém případě by zadal, že bude měřit do 70 metrů a pole se zbylými 30 metry se vůbec nezobrazí.

Organizace mají požadavek, aby byl takovýto formulář 100% interaktivní, což znamená, že zaměstnanec má možnost vytvořit si zcela unikátní formulář. Využíval by jistého zásobníku, kde by byly uloženy všechny typy formulářových prvků a on si poskládal z těchto prvků vlastní formulář. Měl by možnost se na úvodní stránce aplikace přihlásit a poté buď nahlédnout do databáze již vytvořených formulářů nebo začít s prázdným formulářem a vyplnit si ho dle svých představ a požadavků. Tento formulář by si předpřipravil ve webové aplikaci, uložil do databáze a poté si jej v zařízení stáhnul a vyplnil. Nejlépe aby mohl používat jak systém Android, tak iOS a vše se ukládalo do jedné databáze, takže by data z obou systémů byla propojena.

- **Ověřování vstupních údajů** – V aplikaci se objevuje několik typů polí. Požadavek zní tak, aby u polí, kde předem víme, jakého budou formátu, nešlo vyplnit nic jiného. To v praxi znamená, že pokud je uživatel vyzván, aby zadal telefonní číslo, musí zadat pouze devět číslic. Další požadavek je na verifikaci zadávání povinných dat. To znamená, že daná aplikace zkontroluje pole, která musí být za každé okolnosti vyplněna. Pokud pole nebude vyplněno, aplikace nedovolí uživateli formulář uložit, dokud nebude sjednána náprava a označí nevyplněná pole. Měla by zde být provedena i kontrola, že například u pole, kam se vkládá délka úseku, je číslo, které navíc musí být kladné.



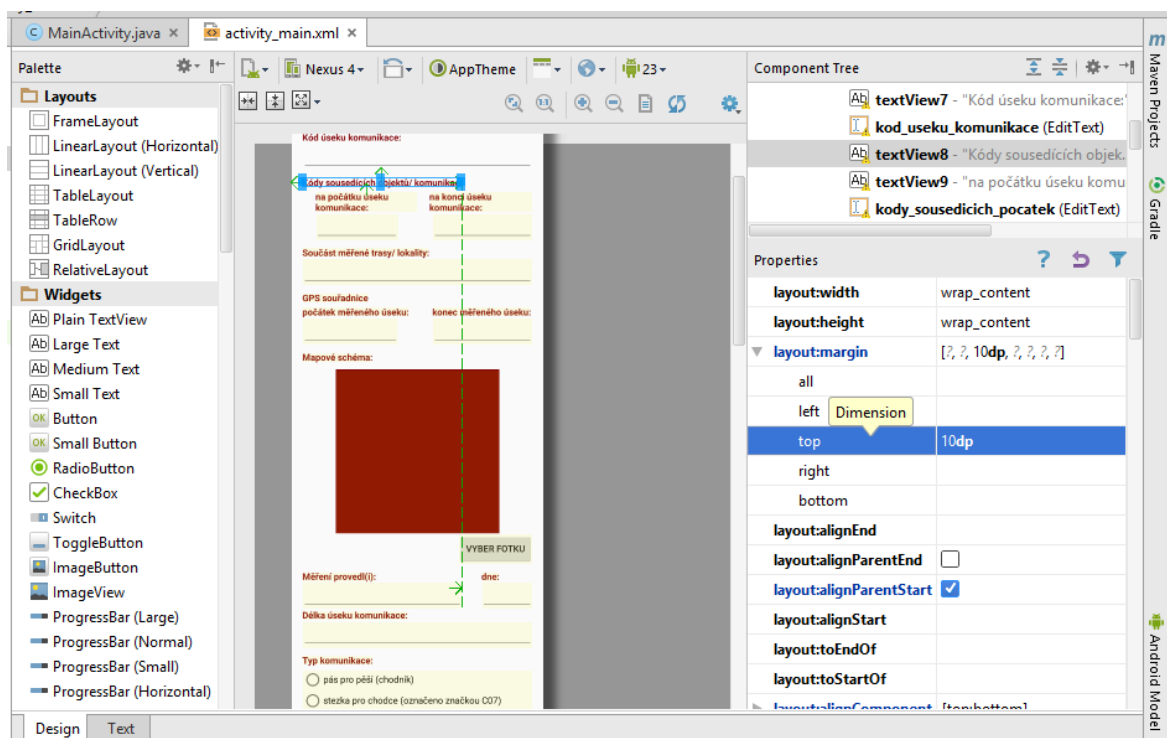
Obrázek 15: Vývojový diagram: Verifikace dat

- **Jednoduchost** – Naše aplikace musí být jednoduchá a srozumitelná pro uživatele, kteří nesmějí příliš dlouho přemýšlet, jak aplikace funguje. Zároveň by však měla být komplexní a obsáhnout všechny základní požadované vlastnosti. Ovládání by mělo být jednoduché a intuitivní. Musíme předpokládat, že pracovník provádějící průzkum je řadový uživatel chytrých zařízení, který sice podstoupil nějaké školení, ale neměl by ztrácet čas při seznamování se se složitým ovládáním naší aplikace. Všechny prvky by měly být jasně determinované a intuitivní. Aplikace musí být také přehledná a logicky uspořádaná. Nesmí se stát, že by došlo k špatnému vyplnění formuláře kvůli návrhovým nedokonalostem. Na první pohled musí být jasné, jaké pole právě vyplňujeme. Aplikace nesmí být příliš složitá na vytvoření a musí být provozu schopná. Z požadavků o interaktivitě tak díky tomu například vypadnou požadavky týkající se 100% interaktivnosti, multisystémové aplikace atd. Všechna rozhodnutí budou v návrhu zdůvodněna.
- **Uložení dat** – Je třeba se zamyslet nad tím, jak budeme data ukládat. Co bude pro pracovníky nejpříjemnější a nejjednodušší. Ukládat data do zařízení či na Internet? Z hlediska nestabilního připojení k síti Wifi a nestálé možnosti využití datových služeb se jeví nejspolehlivějším a nejjednodušším způsobem uložení dat do zařízení i jejich následné exportace v pohodlí domova.

3.2 Struktura aplikace a vývojového prostředí

Aplikace obsahuje dva základní typy souborů, s nimiž se pracuje, a které zároveň obsahují zdrojový kód aplikace. Jsou to soubory typu XML a JAVA.

Soubory typu XML slouží k návrhu všech prvků. V těchto souborech se skládají prvky dohromady, přidělují se jim vlastnosti, jako jsou: barva, výška, šířka, viditelnost atd... Upravuje se vzhled aplikace a její celkové rozložení. Tvoří základ v podobě obsazení všech prvků, s nimiž se následně operuje. Právě do těchto souborů se vkládají prvky, jako jsou, Edit Texty, Radio Butony, Butony atd. XML soubor je také to, co uživatel používající aplikaci vidí. V prostředí, které navrhne do tohoto souboru, tráví celou dobu používání aplikace. Proto je potřebné, aby struktura tohoto souboru byla přehledná a intuitivní, a to nejen pro programátora aplikace, ale především pro potenciální uživatele. Android studio umožňuje pracovat s takovýmto souborem ve dvou různých typech prostředí. Jedná se o prostředí designové a textové, přičemž mezi oběma typy je možnost plynule přecházet a aktivně tak využívat výhody obou prostředí.



Obrázek 16: Ukázka designového prostředí

V designovém prostředí odpadá nutnost znalosti XML jazyka. Obrazovka je rozdělena do tří hlavních bloků. Levý blok obsahuje pestrou škálu prvků, které může vývojář do své aplikace umístit. Jsou rozděleny do sedmi skupin podle funkce a společných vlastností. Například se zde nachází skupina s názvem Text Fields. Ta obsahuje velké množství prvků Edit Text, které se liší ve svých vlastnostech, které vyplývají již z názvu například speciální Edit Text pro číslo, prostý text, email, datum, telefon ... Vložením například čísla zajistíme, že už od začátku má

takovéto pole vlastnosti nedovolující napsat jiné znaky než jsou čísla a tečka, což nám ulehčí práci s následným vybíráním vhodných vlastností prvku. Samozřejmě lze vložit i prosté pole a následně ho pomocí vlastností upravit tak, aby bylo číselného formátu. Pravý blok obsahuje hierarchický seznam prvků používaných v aplikaci a okno s vlastnostmi daného prvku, které můžeme libovolně měnit a upravovat. Obrázek 16: Ukázka designového prostředí zobrazuje vybrání prvku Text View a v okně vlastnosti je vidět atribut margin top, který je nastaven na 10dp. To nám určuje horní odsazení vnějšího okraje pole.

Níže můžeme pozorovat stejný atribut zapsaný v textovém prostředí. Okno vlastnosti se dynamicky mění podle vybraného prvku. Rozdílné atributy lze pozorovat u Radio Buttonu v porovnání s Edit Textem. Prostřední blok je věnován samotnému umístování jednotlivých prvků a celkovému vzhledu aplikace. V tomto bloku je možné jednotlivé prvky skládat a posouvat podle vlastního uvážení, pokud nevyužijeme okno vlastností nebo textový editor.

Textové prostředí funguje na principu textového editoru, kam vkládáme požadovaný kód. Pokud znovu uvedu ten samý příklad textu, je možné si povšimnout, mimo jiné, stejného atributu margin. Textový editor barevně rozlišuje určité úrovně a tím velmi zpřehledňuje práci. Navíc je prediktivní, takže když zadáme první písmena, už nám nabízí nejvhodnější možnosti ke zvolení, nemusíme tedy vše psát.



```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:id="@+id/kod_useku_komunikace"
android:layout_below="@+id/textView7"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_alignParentRight="true"
android:layout_alignParentEnd="true" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Kódy sousedících objektů / komunikací"
    android:id="@+id/textView8"
    android:layout_column="0"
    android:textColor="#870b0f"
    android:textStyle="bold"
    android:layout_below="@+id/kod_useku_komunikace"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_marginTop="10dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="na počátku úseku komunikace:"
    android:id="@+id/textView9"
    android:layout_column="0"
    android:textColor="#870b0f"
    android:textStyle="bold"
    android:layout_below="@+id/textView8"
    android:layout_marginTop="5dp" />
```

Obrázek 17: Ukázka textového editoru

Java soubory slouží k vytvoření daných funkcí, kde jednotlivé prvky přicházejí k životu. Zde už se pracuje pouze s textovým prostředím, které je také barevně rozlišené, prediktivní atd.

3.3 Prvky navrhované aplikace

Aplikace obsahuje velké množství rozdílných prvků. Jak prvky, které jsou na první pohled viditelné a uživatel s nimi pracuje, tak prvky, které uživatel využívá, a jsou v Java souboru, ale registruje až výsledky jejich působení.

Důležitý prvek aplikace jménem Activity nelze zařadit do níže uvedených skupin. Takto se v prostředí androidu nazývá obrazovka, se kterou uživatel může pracovat a skrz ni komunikovat s aplikací. V aplikaci je několik aktivit, které jsou vzájemně propojeny. Jedná se o XML soubor s přiřazeným Java souborem. V XML se navrhuje níže uvedené prvky a v Java souboru se navrhuje metody a funkce pro jejich práci.

3.3.1 Prvky XML

V práci zachovávám konvenci v psaní názvů z prostředí Android Studio, proto je například Scroll View psáno v obou slovech s velkými písmeny.

3.3.1.1 Prvky sloužící k rozmístění

Scroll View – Aplikace je rozsáhlá, co se týče kvantity prvků a všechny nelze na jedné stránce obsáhnout. Nabízí se možnost takovou aplikaci rozdělit na více stránek a pomocí daných tlačítek mezi nimi přecházet nebo použít právě Scroll View. Díky tomu je možno využít, v tomto případě, vertikální posuvník. Vše je tedy přehledně zobrazeno na jedné stránce a uživatel může intuitivně posouvat obrazovku pomocí určeného gesta.

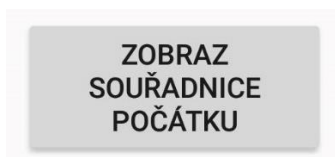
Table Layout – V aplikaci je použit tento prvek, protože bylo potřeba vytvořit dvě tabulky 4 x 50 a 2 x 20. Díky tomu, jsou všechny tyto prvky rovnoměrně rozmístěny, jsou přehledné, stejné a na pohled uhlazené.

Table Row – Jinými slovy řádek. Vložíme jej do Table Layoutu. Do tohoto řádku vkládáme texty a pole. Dále, krom prvních dvou řádků, tj. hlavička tabulky a první řádek, je u všech řádků nastavena viditelnost na „gone“, což znamená, že tyto řádky se tváří jako by tam nebyly a dále je zviditelňujeme až v souvisejícím Java souboru, v důsledku stisknutí tlačítka.

3.3.1.2 Akční prvky

Button – Klasické tlačítko. Po stisknutí tohoto tlačítka se provede náležitá akce s kódem v Java souboru. V XML souboru se, kromě jiného, zabýváme tím, kde má být umístěno a jak má tlačítko vypadat, dále pak textem, který má tlačítko obsahovat, id tlačítka pro snadnou

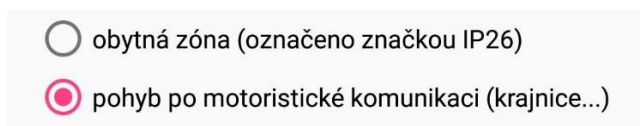
orientaci a okamžité přiřazení a pojmenováním funkce OnClick, kterou následně voláme v souvisejícím Java souboru.



Obrázek 18: Ukázka prvků- Button

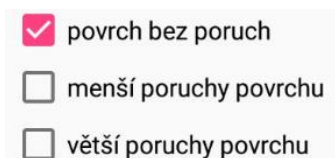
Radio Button Group – Skupina do, které sdružujeme Radio Buttony. Tím, že jednotlivé Radio Buttony umístíme do skupin, zajistíme, že lze v dané skupině vybrat pouze jedna možnost. Nestane se, že by uživatel mohl zaškrtnout více než jednu možnost.

Radio Button – Tlačítko předem definovaného výběru. Uživatel pouze volí z navržených možností. Odpadá tak chybovost daná vyplňováním uživatele.



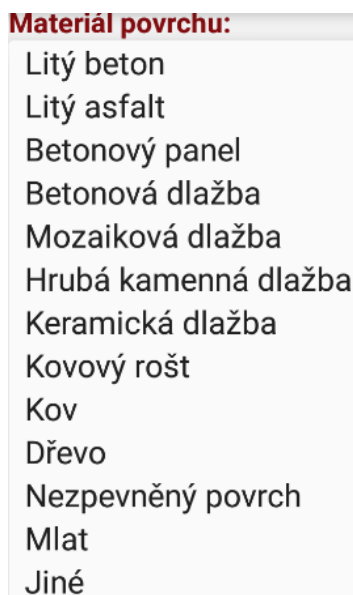
Obrázek 19: Ukázka prvků - Radio Button

Checkbox – Uživatel vybírá, pomocí zaškrtnutí nejvhodnější odpověď. V případě této aplikace je ještě zřízena sada pravidel tak, aby se četnost zobrazovala, pouze pokud to dává logický smysl a nešlo vybrat více než jednu odpověď.



Obrázek 20: Ukázka prvků - Checkbox

Spinner – Neboli rozvírací seznam, u kterého můžeme vybrat pouze jednu položku. V případě této aplikace obsahuje třináct položek. Dvanáct z nich popisuje nějaký druh materiálu. Třináctý s popiskem „Jiný – uveďte“, zobrazuje Edit Text a dává uživateli možnost napsat vlastní druh materiálu. Tento prvek je vhodný pro předem dané odpovědi. Díky tomu, že je rozevírací a všechny jeho odpovědi nejsou tedy hned v aplikaci vypsány, ušetří velmi mnoho místa a aplikaci zpřehledňuje.



Obrázek 21: Ukázka prvků – Spinner

Edit Text – Je nejvíce zastoupený a nejvíce stěžejní prvek, s kterým uživatel pracuje. Samozřejmě mimo klasického textu, který je ovšem pouze informativního a popisného charakteru. Evidentně každý Edit Text má unikátní id, které slouží k přesnému určení. Rozdíly jsou dále v typu tohoto pole. U některých polí je žádané, aby uživatel mohl zadat pouze určité znaky, například u pole staničení je žádané, aby mohl operovat pouze s čísly a písmenem m (jednotka metru). Tento problém je vyřešen tak, že jsem zvolil typ pole jako číslo, což zajistí, že se po stisknutí tohoto pole zobrazí číselná klávesnice. Povolené znaky jsem pak mimo čísel obohatil i o písmeno m. Pro větší přehlednost jsem zvolil roztažení těchto polí přes celou obrazovku. Avšak u některých případů bylo vhodné, opět z důvodu přehlednosti, nechat dvě pole vedle sebe. Může obsahovat vlastnost hint., a to znamená, že pole můžeme vyplnit šedým textem nápovědy, který zmizí při stisknutí pole. I tento text se neuloží do databáze a takové pole je považováno za prázdné.

A rectangular text input field with a light gray background and a thin border. The text inside the field is "Přípustné znaky 0123456789," in a gray font, which serves as a hint for the user.

Obrázek 22: Ukázka prvků - Edit Text s atributem hint

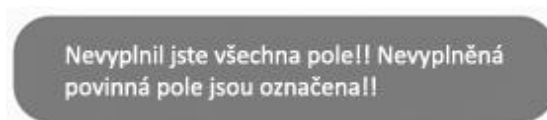
Image View – Tento prvek, o velikosti 250 x 250 dp, slouží jako náhled uživatelem vyfoceného nebo vybraného obrázku z galerie. Také slouží na hlavní stránce, jako hlavní menu v aplikaci.

Mapové schéma:



Obrázek 23: Ukázka prvku - Image View

Toast – Krátká vyskakovací zpráva. Slouží k informování uživatele, že se něco děje, například, že proběhlo uložení do databáze.

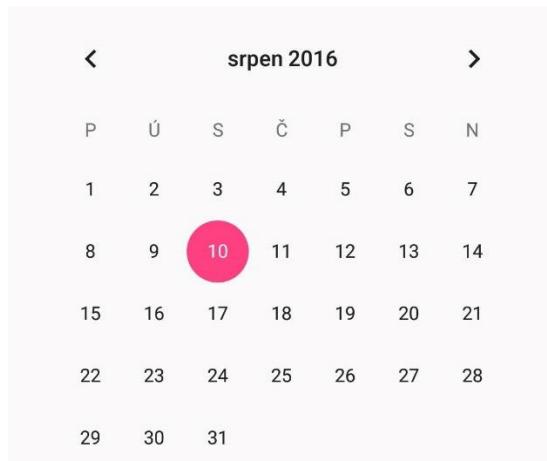


Obrázek 24: Ukázka prvku - Toast

Image Button – Stejné vlastnosti jako klasický Button, s tím rozdílem, že zde můžeme obrázek zobrazit jako pozadí. V případě navrhované aplikace tedy modré informační „i“.

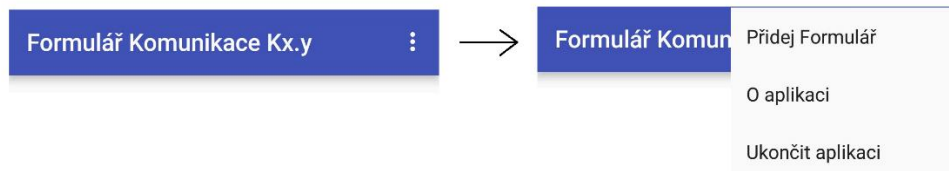
List View – V navrhované aplikaci je prvek použit jako seznam již zpracovaných a v databázi uložených formulářů. Jeho zobrazení je tedy dynamického charakteru. Zobrazuje nám čtyři údaje o formuláři, které slouží k jasné identifikaci požadovaného formuláře. Je to id, kód úseku, datum měření a jméno pracovníka, který tento formulář vyplnil. V našem případě se do tohoto List View načítá druhý XML soubor, obsahující jeden řádek, tak aby všechny řádky byly stejné.

Calendar View – Pro zadávání data, kdy bylo měření provedeno, slouží prvek ve formě kalendáře. Uživatel v kalendáři vidí dnešní datum a vybere požadované datum měření. Při nevybrání žádného data bude vyzván, aby tak učinil. Při zobrazení uložených dat vidí původně zadané datum, a pokud se rozhodne pro změnu, opět zvolením v kalendáři, změní původní datum za nové. Pokud datum bude souhlasit, nemusí uživatel již nic zadávat a v databázi zůstane uloženo datum původně zadané při vytváření formuláře.



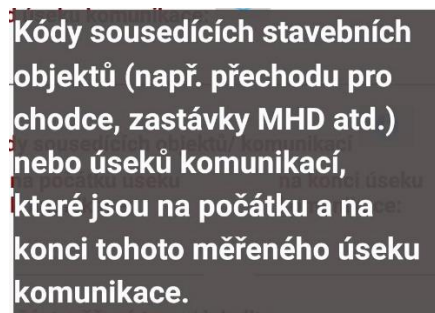
Obrázek 25: Ukázka prvků - Calendar View

Menu – Slouží k rychlejší navigaci uvnitř aplikace. Svými možnostmi se liší pro každou jednotlivou aktivitu v aplikaci. Přes Menu lze celou aplikaci i ukončit a opustit.



Obrázek 26: Ukázka prvků - Menu

Popup Window – Vyskakovací okno v navrhované aplikaci se používá pro kratší informativní údaje. Pro delší údaje je využití přechod na novou aktivitu.



Obrázek 27: Ukázka prvků - Popup Window

3.3.2 Metody pro práci s XML prvky

Inicializace proměnných – Hned na začátku je potřeba inicializovat proměnné, s kterými budeme dále pracovat. Takovýmto proměnným poté přiřadíme odpovídající prvek z XML návrhu.

Nastavení viditelnosti – Tato funkce slouží k zviditelnění, v některých případech nepotřebných, prvků. Jedná se zejména o prvky staničení, kde uživatel může zobrazit až 100

metrů, ale v některých případech může stačit i o mnoho méně. Tato funkce zajišťuje zviditelnění pouze těch metrů, o které opravdu stojí. Jednotlivé prvky může po stisknutí tlačítka přidat, čímž se šetří místo a zpřehledňuje obrazovka. Pokud vyčerpá maximální možnou velikost, zobrazí se zpráva s touto informací a zároveň tato funkce „zneviditelní“ tlačítko pro přidávání prvků. Dále je využito této funkce, když některý prvek je viditelný až po určité akci. Tato akce je využita u poruch povrch nebo Spinneru. Pokud uživatel zadá, že povrch je bezporuchový, není důvod, aby se zobrazovala četnost poruch. V takovém případě zůstane skryta. Pokud ovšem zaškrtně, že poruchy se vyskytují, zobrazí se i četnost poruch k vyplnění. U rozevíracího seznamu to funguje tak, že se pole zobrazí pouze v případě, nevyhovující nabízené možnosti a uživatel se rozhodne pro výběr vlastní.

Možnost stisknutí – Zajištění toho, že u některých výběracích bodů lze vybrat jen omezené množství odpovědí, nejčastěji jedna. Lze několika způsoby pomocí Radio Buttonů a využití slučování do skupin, pomocí rozevíracího seznamu nebo pomocí Checkboxů a následných podmínek v Java souboru. U prvních dvou možností se vždy jedná jen o jednu možnou odpověď. Pokud rozhodnutí padne na Checkboxy je možné, aby podmínka byla jakákoliv. U četnosti byly vybrány Checkboxy zejména z důvodu možnosti odkliknutí. Pokud uživatel zvolí nějakou četnost, ale rozhodne se, že vlastně stav je bezporuchový, může četnost odškrnout a vybrat stav bezporuchový. V případě Radio Buttonu by muselo být vedle zvláštní tlačítko na vynulování Radio Buttonu.

Verifikace – Tato funkce se sestává ze dvou bodů. Jednak z upozornění uživatele na nevyplnění požadovaných položek a za druhé na odmítnutí možnosti uložení takového stavu. Upozornění se děje pomocí výstražného vykřičníku a textu v daném poli a výstražné vyskakovací zprávy (Toastu). Také jsou ošetřeny znaky, které uživatel může zadat, aby nedošlo ke zbytečným chybám. Například v případě „délky úseku“ nemůže zadat znaménko mínus a jiný znak než číslici.

Kamera a Galerie – Důležitou usnadňující funkcí je využití vestavěného fotoaparátu na pořízení fotografie a možnost výběru mapového podkladu, v grafickém formátu typu JPG, PNG apod., ze zařízení. Použití fotoaparátu, stejně jako vstup do galerie, musíme aplikaci dovolit. Jak fotografii z fotoaparátu, tak obrázek z úložiště se uživateli zobrazí jako náhled v aplikaci, v příslušných oknech Image View. Zároveň v této funkci se řeší vložení cesty do příslušného řetězce String.

Pořízení fotografie probíhá tak, že po žádosti na aplikaci o pořízení fotografie aplikace nejdříve zkontroluje, zda je vyplněno pole kód úseku. Důvodem této podmínky je, aby bylo zaručeno, že ke každému formuláři je přiřazen odpovídající obrázek, protože název obrázku se tvoří následovně: „Kód úseku komunikace ve formátu Kx.y (kde x je zvolené číslo celé měřené

lokality a y je zvolené číslo tohoto měřeného úseku komunikace)“ + „_cam_image.jpg“. Kód úseku formuláře bude totiž unikátní a lehce rozpoznatelný oproti dalším kódům. Pokud kód nebude zadán, aplikace vyzve uživatele, aby ho nejdříve vyplnil a až potom bude moci pořídit fotografii. Pokud je kód zadán a uživatel podá žádost na pořízení fotografie, nastaví se pole pro vyplnění kódu úseku komunikace, tak aby již nešlo změnit. Uživatel přejde do prostředí fotoaparátu a může pořídit fotografii. Po pořízení fotografie aplikace zkontroluje, jestli se na interním úložišti nalézají adresář s názvem „MyDirectory“. Pokud ne, vytvoří ho, pokud ano, uloží do něj obrázek. Dále se do příslušného Stringu vloží adresa tohoto obrázku. Tento String se později uloží do databáze.

V případě vybrání obrázku z galerie jsou podmínky na akceptování žádosti o přístup do galerie totožné s podmínkami na žádost o použití fotoaparátu. Opět se zkontroluje vyplnění pole „kód úseku komunikace“ s tím rozdílem, že nyní se obrázek bude jmenovat jako „kód úseku komunikace“ + „_gall_image.jpg“. Obrázek se tentokrát musí ovšem zkopírovat z již existující adresy do požadovaného adresáře. Proto se ze všeho nejdříve určí původní adresa obrázku. Po jejím určení se soubor z této adresy zkopíruje na požadovanou adresu pod předem nastaveným jménem. Do příslušného Stringu se opět uloží nová adresa obrázku, která se poté vloží do databáze.

Lokalizace pomocí GPS – K lokalizaci polohy pomocí GPS musíme nejdříve aplikaci udělit potřebná oprávnění. Při požadavku na určení polohy se nejdříve zkontroluje, jestli je v zařízení zapnuto určování polohy. Pokud ne, je to uživateli, pomocí Toastu, oznámeno a dále je přesměrován do nastavení telefonu, kde může určování polohy zapnout. Po zapnutí je, opět pomocí Toastu, uživateli sděleno, že GPS byla zapnuta. Po zaměření, podá aplikace uživateli zprávu o zeměpisných souřadnicích. Tyto souřadnice následně uloží do příslušného Stringu. Určování souřadnic počátku i konce je obstaráno pomocí jedné funkce pouze s tím rozdílem, že po žádosti na určení počátku se do proměnné „GPS_Pozadavek“ uloží jiné číslo než při žádosti na určení konce. Tím se identifikujete, do kterého pole se mají údaje ukládat.

3.4 Postup vývoje aplikace

Při vývoji aplikace jsem se držel obecnými doporučeními pro vývoj softwaru. Principy vycházely z životního cyklu vytváření softwaru. Jednotlivé fáze toho cyklu jsou:

- **Co** – Specifikace, co má program dělat.
- **Jak** – Specifikace toho, jak toho má program docílit.
- **Vývoj** – Vývoj, naprogramování kódu, který implementuje bod „Jak“.
- **Použití** – Použití kódu, který provádí to, co bylo definováno v bodě „Co“. [2][3]

Při návrhu aplikace je stěžejním zdrojem papírový formulář. Ve svém návrhu se tedy nemusím zaobírat problémy, které byly vyřešeny při návrhu papírového formuláře. Těmito problémy myslím například, zda jsou informace o úseku postačující, nebo nejsou-li některé informace nadbytečné atd. Naopak беру papírový formulář jako šablonu či předlohu. Při rozboru tohoto formuláře jsem zjistil, že do své aplikace budu muset použít určitou škálu prvků. Nelze se spokojit pouze s textovým polem, do kterého uživatel vkládá údaje. Potřebné prvky jsou vypsány v kapitole: 3.3.1.

Po zjištění, jaké prvky jsou potřeba do aplikace implementovat, jsem zástupce jednotlivých prvků umístil do aplikace. Z počátku jsem schválně neumisťoval do aplikace všechny prvky, ale pouze jeden či dva příklady od každého typu. Je to z důvodu zrychlení a zpřehlednění práce. V takovémto případě jsem mohl vyřešit všechny problémy a požadavky pouze pro, z celkového hlediska, zanedbatelné množství prvků. Když se tedy vyskytla nějaká chyba či zádrhel, nemusel jsem přepisovat všech +/- 200 prvků. Ale pouze jejich zlomek.

První fáze byla tedy sběr požadavků a tvorba prvního návrhu sestávajícího se pouze z malé výseče celkového objemu prvků. Díky této strategii jsem mohl mnohem lépe, flexibilněji a rychleji reagovat na problémy, chyby a překlepy vzniklé během programování.

První věcí co jsem tvořil, bylo to, že jsem se rozhodl pro jednu obrazovku posouvající se pomocí vertikálního posuvníku. K tomu posloužil prvek Scroll View, jenž právě toto řeší a dovoluje. Je to tedy hierarchicky nejvýše postavený prvek. Do tohoto prvku jsem vložil Relative Layout, prvek, který dovoluje posouvat a vkládat ostatní prvky dle libosti. S tím, že v něm lze velmi dobře nastavit k čemu se má jaký komponent upnout a jeho odsazení. Relative Layout je tedy plátno, do kterého vkládáme všechny ostatní prvky. Pro zvýšení přehlednosti jsem nastavil odsazení od všech krajů na 16 dp. Tento Layout také zajistí, že aplikace je schopna mít dva prvky na témže řádku. Při volbě Linear Layoutu by byl problém v tom, že každý komponent by musel být umístěn na samostatném řádku (nebo sloupci), proto tedy volba Relative Layoutu. Nestane se tedy, že komponenty by byly umístěny hned u kraje, což by mohlo zhoršit orientaci, přehlednost a efekt.

Dále jsem vložil zástupce od každého prvku, vytvořil tabulku 4 x 5 a vyplnil ji požadovanými komponenty. Nejdříve jsem se zaměřil na funkčnost daných prvků, proto jsem řešil postupné zviditelnění tabulky po řádkách, a to tak, že jsem nechal první dva řádky tabulky viditelné. Ostatní jsem nastavil na parametr „gone“ s tím, že tlačítko „Přidej další staničení“ je připnuté k hraně Table Layoutu, proto po každém zvětšení tabulky odskočí dolů, aby bylo stále u hrany. V Java souboru jsem zajistil postupné zviditelňování řádků po jednom. V momentě, kdy se vyčerpají řádky, tlačítko už nemá využití, parametr „visibility“ se přepne na „gone“ a zároveň se spustí oznámení pomocí prvku Toast, které nám oznámí hlášku „Vyčerpána

maximální možná velikost staničení!“, přičemž staničení se zvedá o dva metry a jeho maximální velikost je 100 metrů.

Poté jsem vložil Image View výběru fotografie z galerie. Tedy prvek sloužící k výběru předpřipraveného mapového schématu. Velikost tohoto komponentu, a zároveň tedy náhledu, jsem zvolil 250 x 250 dp, je to z důvodu, aby takovýto obrázek byl dobře viditelný, ale zároveň nezabíral příliš velkou část obrazovky. V Java souboru bylo potřeba nejdříve vytvořit metodu, která otevře galerii po stisknutí požadovaného tlačítka. Zvolíme obrázek a ten se vloží do Image View, následně adresu vloží do Stringu. V zásadě stejný postup platí i u použití fotoaparátu. Jeho Image View má stejné vlastnosti. Pokud zařízení obsahuje fotoaparát, po stisknutí tlačítka se otevře, zaznamená se požadovaná fotografie, která se opět vloží jako náhled do Image View fotoaparátu a dále se adresa uloží do příslušného řetězce.

Jako další jsem řešil Radio Buttony. Údaje o vybrání jsem vyřešil jednoduchou podmínkou „if“. Zvolil jsem si jeden řetězec typu String, například String jménem „nerovnosti_povrchu“. A pomocí podmínky „if“ jsem nadefinoval, jaký text se má do ní ukládat. Pokud je zaškrtnuto „Rovný povrch bez nerovností“ obsahuje i proměnná „nerovnosti_povrchu“ tento text v okamžiku kdy uživatel zvolí jinou možnost, tato proměnná se přepíše právě na zvolenou možnost. Checkboxy jsou řešeny totožně s tím rozdílem, že obsahují sadu pravidel. Takže pokud uživatel zvolí, že „Povrch je bez poruch“ nastaví se ostatní možnosti na nemožné ke zvolení a „Četnost poruch“ zůstane skrytá a do příslušného řetězce se uloží nulová hodnota, pokud ovšem zvolí jinou možnost „Četnost poruch“ se objeví.

Dalším prvkem, který bylo třeba vyřešit, je Spinner. Největším jeho problémem bylo, aby při vybrání položky „Jiný – uveďte“ se zobrazil Edit Text a do příslušného Stringu se vložila data z tohoto Edit Textu. Na rozdíl od vybrání jakéhokoliv jiného prvku. V takovém případě se do příslušného Stringu vloží text vybraného prvku ze Spinneru a Edit Text zůstane skryt.

Technický prostředek, který velmi usnadní a zpřesní práci, je lokalizace polohy pomocí GPS. V aplikaci to funguje na principu získání souřadnic o zeměpisné délce a šířce po stisknutí tlačítka. Pokud uživatel nemá povolenou lokalizaci polohy, je přesměrován do nastavení, kde ji může zapnout. Data o geografických souřadnicích se vloží do Edit Textu.

Nejpočetněji zastoupeným prvkem v aplikaci je Edit Text. Tento prvek nepotřebuje žádná speciální pravidla. Funguje tak, že text, který do něj uživatel napíše, se vždy vloží do příslušného Stringu.

Dále byla potřeba vyřešit nápovědu. Ta se v aplikaci aktivuje stisknutím příslušného prvku Image Button. Nápověda je dvojího druhu, buď ve formě Popup Window, kdy se nápověda zobrazí v aktuální aktivitě pomocí vyskakovacího okna (takto se zobrazují kratší nápovědy),

nebo se po stisknutí požadovaného Image Buttonu spustí nová aktivita obsahující nápovědu, toto řešení jsem volil u delších nápověd.



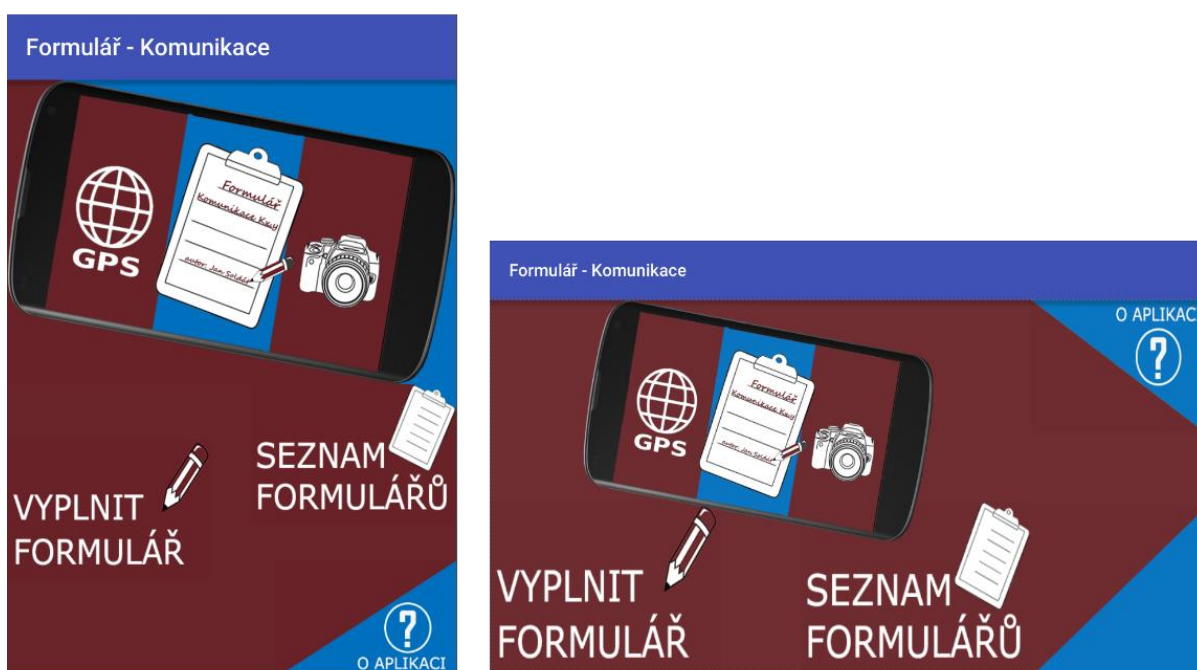
Obrázek 28.: Ukázka nápovědy k nerovnostem povrchu

Prvkem, který se objevuje, až na hlavní aktivitu, ve všech aktivitách, je menu. V navrhované aplikaci je použito tzv. Options Menu. Toto menu je umístěno v pravém horním rohu okna. Je reprezentováno pomocí třech vertikálních teček, které po jejich stisknutí otevřou požadované menu uzpůsobené pro každou aktivitu zvlášť, tak aby obsahovalo potřebné prvky. Zbytečné prvky, jako odkaz na současnou aktivitu, byly vyloučeny. V aplikaci se tedy nachází jak menu o jedné položce, v případě nápověd, kde je pouze možnost „Zpět“, tak i menu o čtyřech položkách v případě náhledu již vytvořeného formuláře. V této aktivitě je možnost přejít na aktivitu pro přidání nového formuláře, aktivitu se seznamem formulářů, aktivitu o aplikaci a ukončení aplikace.

Prvek Calendar View funguje tak, že po změně dne v kalendáři, se tento datum uloží do Stringu. Zadání je opět ošetřeno podmínkami.

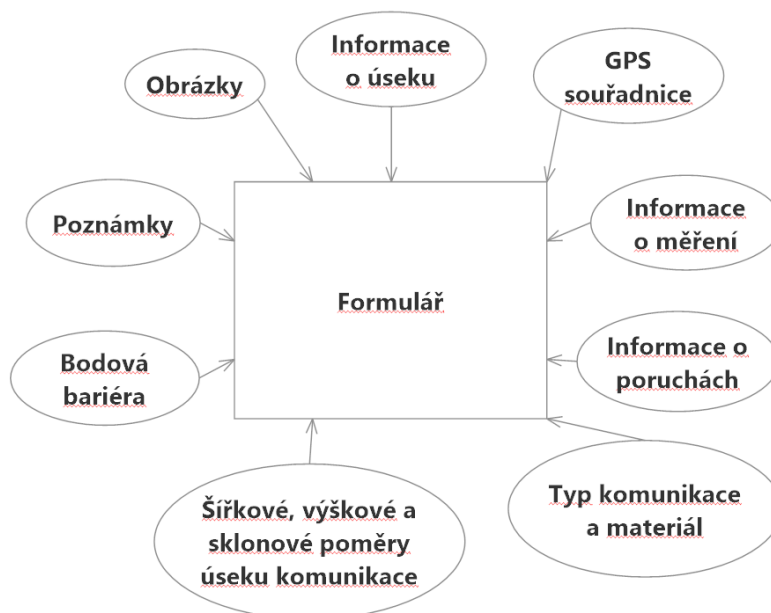
Pro spíše estetické účely slouží dva následující prvky. Je to obrazovka, která se zobrazuje při načítání aplikace, protože při startu aplikace nějakou dobu trvá (maximálně pár sekund a to především u prvního spuštění), než se aplikace spustí. V takovýto moment by uživatel viděl pouze bílou obrazovku, kterou jsem nahradil logem aplikace. Trvá pouze po dobu spouštění aplikace, tudíž uživatele nezdržuje a slouží spíše ke zpříjemnění spouštění.

Dalším estetickým prvkem je hlavní aktivita. Ta se skládá z loga aplikace vloženého do Image View a obrázků pro navigaci po aplikaci vložených taktéž v prvcích Image View. Uživatel se tedy v této aktivitě může rozhodnout, jestli vyplní nový formulář nebo se podívá na již vytvořený formulář, kde ho může aktualizovat nebo smazat či se chce podívat pouze na informace o aplikaci.



Obrázek 29: Ukázka úvodní stránky aplikace ve vertikální i horizontální poloze

3.4.1 Ukládání dat



Obrázek 30: E-R diagram

Jelikož nejdůležitější na celém procesu mapování je uložení dat, bylo nutné pracovníkem provádějící průzkum naměřená data uložit pro další práci. Android nabízí několik možností, jak data ukládat, a protože je žádoucí, aby data byla někde pevně uložena, v nějaké dané struktuře a nesmazala se při vypnutí nebo smazání aplikace, je nejvhodnější volbou použití SQLite databáze. Druhou zřejmě nejvhodnější volbou by byla MySQL databáze, data by tedy byla uložena na webovém serveru. Úskalí by mohlo nastat v tom, že v některých místech mapování nemusí být přístup k Internetu, nehledě na to, že organizace provádějící tyto průzkumy většinou nemají finance a není jisté, že by si mohly dovolit platit datové služby a data by se tedy v takovém případě neuložila. Pro ukládání těchto dat bylo nutné navrhnout a vytvořit databázi, která má jasně daný seznam prvků a jejich pořadí viz Obrázek 30. Ukládají se tam především informace do atributu s datovým typem text, proto všechny údaje získané z vyplnění formuláře jsem vložil do datového typu String, které se v databázi ukládají jako datový typ TEXT.

Formulář - Komunikace

Úsek komunikace: ⓘ
 Ulice na Florenci 25 -Taxi Fair Place

Kód úseku komunikace: ⓘ
 K1.1

Kódy sousedících objektů/ komunikací ⓘ

na počátku úseku komunikace:	na konci úseku komunikace:
není	K1.2

Součást měřené trasy/ lokality:
 Fakulta dopravní, Masarykovo nádraží

GPS souřadnice

počátek měřeného úseku:	konec měřeného úseku:
50,08939	50,0887100000
14,43753833333	0006
3334	14,43668999999
	9999

ZOBRAZ

Obrázek 31: Ukázka z aktivity vyplnění formuláře

Prvním prvkem v databázi je ID. ID je datového typu Integer a slouží jako primární klíč databáze. Nastaveno je tak, aby začínalo od čísla 1 a ne od nuly. Toto číslo je unikátní, každý záznam má své jasně dané a nezměnitelné id. Pokud se záznam z databáze smaže, tak ani nově přidané záznamy nepřevzou ID od smazaného formuláře. ID má nastavenou vlastnost tak, aby se automaticky zvyšovalo bez zásahu uživatele.

Tabulka_formulare_komunikace	
ID	INTEGER
Usek_komunikace	TEXT
Kod_useku_komunikace	TEXT
Kody_sousedicich_objektu_pocatek	TEXT
Kody_sousedicich_objektu_konec	TEXT
Soucast_merene_trasy	TEXT
GPS_souradnice_pocatek	TEXT

Obrázek 32: Ukázka struktury tabulky

Problémem v této aplikaci bylo uložení obrázků, tady se nabízely dvě možnosti. Bud uložení cesty k obrázku a obrázek vždy přikládat k databázi, nebo uložení jako Blobu. Já se původně rozhodl pro uložení do Blobu, ale po vyzkoušení aplikace se toto rozhodnutí ukázalo jako mylné, proto byl princip ukládání obrázku změněn a nyní aplikace ukládá pouze adresu

obrázku, místo celého obrázku převedeného do bytové řady. Toto rozhodnutí značně snížilo celkovou velikost databáze a odlehčilo aplikaci při zpracovávání dat.

Name	Type	Not	PK	AI	U	Default
ID	INTEGER	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1

Obrázek 33: Ukázka vlastností prvků databáze

Databáze se původně ukládala pouze do interní paměti telefonu, ke které nemá běžný uživatel, ani uživatel s vývojářskými právy přístup. Proto bylo nutné vymyslet, jak se k tomuto souboru dostat. Jednou z možností bylo složitě pomocí příkazového řádku, aplikace adb.exe a složitých příkazů tuto databázi vyexportovat, ale to by opět bylo krajně nepohodlné, složitě a neefektivní pro běžného uživatele. Proto bylo nutno napsat další Java soubor, který toto vyexportování provede za nás a soubor uloží do mnou zvolené složky v zařízení. Soubor nazvaný „DatabazeFormular001.db“ se tedy vyexportuje do adresáře „MyDirectory“.

Po zjištění, že všechny prvky se mi bez problémů uložily, jsem aplikaci rozšířil o zbývající komponenty, aby obsahovala vše důležité a podstatné.

3.4.2 Práce s uloženými daty

Práce s uloženými je dalším stěžejním bodem navrhované aplikace. Uživatel by měl mít možnost vidět seznam formulářů, které byly vloženy do aplikace, a poté si vybraný formulář zobrazit do předem definované aktivity. Tato aktivita je totožná s aktivitou pro ukládání dat, ovšem bez vysvětlivek a nápověd. V této aktivitě se předpokládá, že uživatel si chce pouze formulář zobrazit nebo upravit nepřesnosti, kterých se dopustil. Zobrazení dat funguje tak, že uživatel v seznamu vidí zástupce dat od všech vyplněných a uložených formulářů. Tato data byla zvolena tak, aby bylo na první pohled jasné, o jaký formulář se jedná a byl tak snadno identifikovatelný. Po vybrání formuláře se vždy spustí ta samá aktivita. Rozdíl je v datech, která naplňují dané pole. Tato data odpovídají vybranému formuláři. A tímto způsobem je zajištěna dynamičnost celého procesu. Je třeba získat dané položky z databáze a vložit je do příslušných polí. Edit Texty nejsou v tomto případě problém, protože většina dat je uložena v databázové třídě text. Problémy nastávají s Radio Buttony, Checkboxy a obrázky. První dva zmiňované je třeba ošetřit sadou pravidel s podmínkou if popřípadně switch. Obrázek se načte z příslušné adresy uložené v textovém řetězci String a zobrazí se v prvku Image View.

Formulář - Komunikace			
Seznam formulářů			
ID:	Kód úseku:	Jméno:	Den:
3	K1.1	Jan Soldát	12.7.2016

Obrázek 34: Ukázka seznamu formulářů

V této aktivitě můžeme záznam také aktualizovat. Všechna pole, kromě „kódu úseku komunikace“, lze přepsat nebo změnit. Fungují zde totožné funkce jako uložení záznamu do databáze. Prvek, který zajistí, aby se aktualizovala ta správná data, je „úsek komunikace“. Do řetězce typu String uloží původní obsah extrahovaný z databáze, takže i při změně pole můžeme aktualizovaná data přiřadit ke správnému záznamu v databázi. Při změně obrázku se soubor uložený v adresáři přepíše, nestane se tedy, že by byly v adresáři dva soubory stejného názvu.

Obrázek 35: Ukázka načtení formuláře a jeho menu

3.4.3 Vzhled navrhované aplikace

Co se týče vzhledu, nejdůležitější je, aby navrhovaná aplikace byla co nejvíce přehledná. K tomu slouží například kontrast textu a pozadí, není žádoucí, aby aplikace měla pozadí

i text v podobných odstínech. Je třeba předpokládat, že aplikace se bude používat za denního světla, které značně snižuje viditelnost displeje, tudíž barvy musí být dostatečně kontrastní. Aplikace by rozhodně měla mít také určitý design a ne být moc strohá, aby se v jejím prostředí uživatel cítil příjemně. Bylo proto potřeba nalézt kompromis mezi vzhledem a funkcí. Výsledkem je, že hlavní aktivita je více zaměřená na pěkný vzhled, zatímco aktivity kde se skutečně vyplňuje a pracuje, jsou zaměřeny spíše na funkčnost a přehlednost.

Dále písmo a ostatní prvky, použité v navrhované aplikaci, musí být dostatečně velké, aby uživatel neměl problém s přečtením nápisu či zvolením požadované možnosti. Zároveň však nesmí být příliš velké, pokud by v aplikaci byly prvky tak velké, že bychom bez posouvání obrazovky neviděli více než například dva prvky, je to problém, který velmi zpřehledňuje aplikaci, kazí z ní dojem a ve výsledku znemožňuje její používání, kvůli své nepraktičnosti.



Obrázek 36: Ukázka obrazovky nabídky „O aplikaci“

Aplikace obsahuje menu, aby bylo možné snadné přecházení mezi aktivitami. Všechny aktivity jsou, co se týče vzhledu, sjednocené a ucelené tak, aby používání aplikace bylo příjemné a vzhled nerozptyloval od hlavní náplně aplikace, funkčnosti. Vhodné je proto vše hned uživateli nezobrazovat, ale vyčkat, až bude uživatel daný prvek chtít zobrazit, viz staničení, speciální aktivity na nápovědu, vyskakovací okna nebo vyskakovací zprávy ve formě Toastu.

4 Zkouška aplikace v terénu

V srpnu 2016, u Fakulty dopravní na Florenci, proběhla zkouška aplikace v reálné situaci, přímo v terénu. Byla mapována pěší komunikace v úseku v ulici Na Florenci od budovy Fakulty dopravní směrem na Masarykovo nádraží, jak ukazuje Obrázek 37.



Obrázek 37: Měřený úsek [40]

Mapování probíhalo tak, že jsem si den před měřením připravil mapový podklad a zvýraznil v něm měřený úsek. Mapový podklad jsem poté uložil do mobilního telefonu. Pro potřeby mapování byl použit Laserový dálkoměr GLM 80 Professional.

V den měření jsem vyrazil na místo provádění průzkumu. Do aplikace jsem zapisoval naměřené údaje z mapování a pomocí laserového dálkoměru mapoval vzdálenosti a úhly. Nejvíce náročné je měření sklonů, v takovémto případě je vždy nutno laserový dálkoměr po dvou metrech umístit na komunikaci a údaje opsat do aplikace. Průzkum by s pomocí aplikace mohl provádět pouze jeden člověk, avšak při vyzkoušení takovéto situace během měření jsem došel k závěru, že by to bylo značně nepraktické a zpomalovalo by to práci, pokud by jeden člověk zastával všechny úkony. Proto mé doporučení k používání aplikace je, aby průzkum prováděli dva lidé. Jeden bude mít na starosti zapisování hodnot do aplikace a druhý bude měřit a hlásit úhly a šířku pomocí přístroje na měření úhlů a vzdáleností.

Během průzkumu panovaly dobré meteorologické podmínky. Bylo lehce zataženo, nebyla zima ani nepršelo. Z toho důvodu nebyl sebemenší problém s čitelností displeje, který měl nastaven výkon jasu na polovinu. Vše bylo dokonale čitelné, všechna pole byla snadno přiřaditelná a šla bez sebemenších problémů vyplnit. Nejasnosti by mohly vzniknout při vyplňování nerovností povrchu a velikosti poruch. Z tohoto důvodu byly do nápovědy k těmto prvkům přidány ilustrační obrázky, které uživateli usnadní rozhodování. Další chybou bylo, že u příčného sklonu v prvním staničení šla zadat pouze tečka namísto čárky. Tento problém byl

také odstraněn a u všech ostatních staničení byly požadované znaky znovu překontrolovány, tak aby se u nich tento problém neobjevoval.

Aplikace, i přes drobné a již odstraněné chyby, fungovala bezproblémově a zpříjemnila provádění průzkumu oproti papírové verzi. Průzkum probíhal zhruba 31 minut, přičemž časově nejnáročnější bylo zaznamenání staničení, které je velmi obsáhlé na počet prvků a je nutno každý jeden údaj změřit pomocí laserového dálkoměru. Formulář se zobrazuje v seznamu formulářů a všechny údaje se bezproblémově uložily a zpět načetly, jak uvádějí přiložené obrázky. Úbytek baterie i při zapnutých datech, zapnutém zjišťování polohy, zapnuté synchronizaci a s jasem displeje nastaveným zhruba na polovinu činil zhruba 1% u telefonu s baterií o kapacitě 4165 mAh. Při těchto podmínkách by mohlo mapování probíhat zhruba 52 hodin a 30 minut.

5 Závěr

Hlavní motivací pro vznik bakalářské práce bylo vytvoření elektronické verze formuláře pro mapování bezbariérovosti pomocí mobilního zařízení. Konkrétně formuláře pro mapování komunikací. Tato elektronická verze si klade za cíl usnadnit pracovníkům provádějícím mapování práci v terénu, tj. zjednodušit proces zapisování a získávání dat a omezit vznik chyb při vyplňování formuláře.

Prvním krokem byla analýza současné situace s operačními systémy. Po ní jsem dospěl k názoru, že nejvhodnějším systémem pro tvorbu aplikace je buď iOS, nebo Android. Android jsem vybral z důvodu většího procentuálního zastoupení mezi mobilními telefony a větší variability přístrojů. S tím souvisí také cenový rozptyl v nákladech na pořízení přístroje, který tento systém používá, protože organizace provádějící mapování bezbariérovosti mají v tomto případě větší možnost výběru vhodného a cenově dostupného zařízení. Dalším důvodem je přívětivé vývojářské prostředí a velké množství návodů a příkladů použití.

Při analýze technických prostředků jsem si uvědomil, které prostředky by mohly práci při sběru informací ulehčit a bylo by je tak vhodné využít. Jedná se samozřejmě o dotykový displej, na jehož podstatě stojí celá myšlenka navrhované aplikace. Fotoaparát a GPS pomohou při sběru informací a odstraní chybovost při provádění mapování. Naproti tomu jsem se rozhodl některé technické prostředky nevyužít. Mikrofon by dobře posloužil při zvukové poznámce, ale ve výsledku by zvuková poznámka mohla spíše práci přidělat, než ji ulehčit, musel by zde existovat někdo, kdo by poznámku přepsal do písemné podoby, což by vedlo k větší složitosti procesu.

Při rozboru papírových formulářů došlo k objasnění a ucelení směru, jakým se má aplikace vyvíjet, tj. vykryštalizovaly se prvky, které bude nutné v navrhované aplikaci použít a ukázalo se, které technické prostředky budou nejvhodnější k využití. Také bylo poukázáno na nedostatky papírového řešení, zejména na složitosti při několikanásobném vyplňování téhož formuláře, nejdříve rukou na místě provádění průzkumu a následně u přepsání do elektronické podoby. Při využití mobilní aplikace odpadne nutnost tisknutí formuláře, pořizování fotografií pomocí externího fotoaparátu. Při přechodu z papírové verze na verzi mobilní se také sníží chybovost při vyplňování formuláře. Aplikace nedovolí uživateli, aby zadal, na první pohled, nesmyslné údaje - například písmena do pole, kde smí být pouze čísla. Také zajistí, aby uživatel vyplnil požadované povinné údaje jako je jméno, datum měření apod.

Po analýze požadavků, jako je požadavek na interaktivnost nebo verifikaci, byly vyselektovány požadavky oprávněné a realizovatelné oproti těm spíše utopickým.

S ohledem na výše zmíněné informace, bylo prvním krokem vytvoření kostry požadovaných akčních prvků, zajištění jejich funkcí a požadovaných vlastností. Bylo potřeba udělat důležitá

rozhodnutí typu, jak pracovat s obrázky, které prvky ukázat hned a které skrýt, jak data ukládat a jak je zpět vyvolat, jakým způsobem vyřešit nápovědu, které prvky mají v aplikaci smysl a které svou složitostí je lepší nahradit jinou alternativou. Po vyřešení všech zásadních otázek, vytvoření kostry a zajištění, aby se data bez problému ukládala, načítala, aktualizovala a mazala, přišlo na řadu aplikování ostatních prvků na principy vytvořené právě pro tuto kostru. V momentě kdy aplikace obsahovala všechny prvky, bylo třeba vyřešit její celkový grafický design, aby aplikace byla přehledná a funkční. V grafickém programu jsem udělal logo aplikace, které by mělo vyjadřovat sloučení několika přístrojů do jednoho v podobě chytrého mobilního telefonu. Dále byla vytvořena úvodní stránka, ikonka zobrazující se na ploše zařízení, obrazovka, která se uživateli zobrazí při spuštění aplikace a celkové ucelení barev a písma v celé navrhované aplikaci.

Následovala kontrola aplikace jak s cvičnými daty, aby se otestovala správná funkčnost aplikace, tak generální zkouška přímo v terénu, kde se zadávala data reálná. Při těchto kontrolách bylo největším problémem ukládání obrázku, které muselo být následovně přepracováno a nyní se ukládá do databáze pouze cesta k obrázku namísto Blobu z první verze aplikace. Další problémy byly méně významné a jednalo se spíše o nepřesnosti nebo o vylepšení symbolického charakteru, jejichž vyřešení však značně zlepšilo funkčnosti celé aplikace.

Tato verze není konečná, počítá se s dalším rozvojem aplikace tak, aby v budoucnu byla mnohem rozsáhlejší a komplexnější. Plán je takový, že by aplikace měla obsahovat všechny dostupné druhy formulářů tj. více než 20 druhů. Měla by tak pomoci sjednotit mapování bezbariérovosti na území celé České republiky. Bude zde existovat provázanost mezi jednotlivými formuláři tak, aby se zajistilo pokrytí celé měřené oblasti. Také by zde mělo fungovat přihlašování a na základě přidělených funkcí oprávnění co s aplikací lze, či nelze provádět. Byla by zde tedy uživatelská kategorie uživatele, který by mohl pouze formuláře přidávat, prohlížet a vlastní formuláře upravovat a mazat a naproti tomu by byla uživatelská kategorie administrátora, který by mohl spravovat všechny formuláře. Vhodné by bylo propojení aplikace s mapovým podkladem od společnosti Google, uživatel by tedy nemusel předem připravovat mapový podklad a zároveň mohl nahlédnout do mapy, které oblasti jsou již zmapovány. Dále by se data měla po připojení k Wi-Fi síti synchronizovat a uložit na webové úložiště. Vize je taková, že by aplikace měla mít ještě lepší využití potenciálu chytrého mobilního zařízení a jeho technických prostředků a co nejvíce mapování bezbariérovosti usnadnit, zpříjemnit a zautomatizovat.

Seznam použitých zdrojů

Knihy

- [1] ST. LAURENT, ANDREW M., St. Laurent, Andrew M. Understanding open source and free software licensing. 1st ed. Sebastopol, CA: O'Reilly, c2004. ISBN 0596005814.
- [2] DONALDSON, Scott E. a Stanley G. SIEGEL. Successful software development. 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR, c2001. ISBN 0130868264.
- [3] ARLOW, Jim a Ila NEUSTADT. UML 2 a unifikovaný proces vývoje aplikací: objektově orientovaná analýza a návrh prakticky. 2., aktualiz. a dopl. vyd. Brno: Computer Press, 2007. ISBN 978-80-251-1503-9.
- [4] LACKO, Ľuboslav. Vývoj aplikací pro Android. Brno: Computer Press, 2015. ISBN 978-80-251-4347-6.

Internetové zdroje

- [5] Kiran Bala, Sumit Sharma, Gurpreet Kaur. A Study on Smartphone based Operating System. [online]. [cit. 2016-03-26]. Dostupné z:
<http://www.ijcaonline.org/archives/volume121/number1/21504-4351>
- [6] SteveJobs iPhone 2007 Presentation (Full HD). In: Youtube [online]. 13.05.2013 [cit. 2016-03-26]. Dostupné z:
<https://youtu.be/vN4U5FqrOdQ>. Kanál uživatele Jonathan Turetta
- [7] Apple Reinvents the Phone with iPhone [online]. [cit. 2016-03-26]. Dostupné z:
<http://www.apple.com/pr/library/2007/01/09Apple-Reinvents-the-Phone-with-iPhone.html>
- [8] Dotcom Infoway. Whitepaper on Mobile OS and efforts towards open standards [online]. [cit. 2016-03-27]. Dostupné z:
http://www.dotcominfoway.com/attachments/268_White-paper-on-Mobile-OS-and-efforts-on-Open-standards.pdf
- [9] Mobile/Tablet Operating System Market Share. [online]. [cit. 2016-03-27]. Dostupné z:
<https://www.netmarketshare.com/operating-system-market->

share.aspx?qprid=8&qpcustomd=1&qpstick=0&qpmr=1000&qptimeframe=Y&qpct=3&c
learaf=1

- [10] About the iOS Technologies. <https://developer.apple.com/>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html#//apple_ref/doc/uid/TP40007898-CH1-SW1
- [11] What is iOS? – Definition from Techopedia. <https://www.techopedia.com>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <https://www.techopedia.com/definition/25206/ios>
- [12] Gaurav Jindal, Mayank Jain. A Comparative Study of Mobile Phone's Operating Systems [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <http://gauravjindal.yolasite.com/resources/A%20Comparative%20Study%20of%20Mobile%20Phone%E2%80%99s%20Operating%20Systems.pdf>
- [13] Looking at the Android Operating System. <http://www.dummies.com/>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <http://www.dummies.com/how-to/content/looking-at-the-android-operating-system0.html>
- [14] What is open source?. <https://opensource.com>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <https://opensource.com/resources/what-open-source>
- [15] Amandeep Singh, Bathinda R.K Bansal, Ph.D, Neetu Jha. Open Source Software vs Proprietary Software. International Journal of Computer Applications. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <http://research.ijcaonline.org/volume114/number18/pxc3902132.pdf>
- [16] Licence – Open Source software – Root.cz. <http://www.root.cz/>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <http://www.root.cz/specialy/licence/open-source-software/#zdrojovy-kod>
- [17] Proprietary software is opposite of free software. <http://www.linfo.org/>. [online]. 16.4.2016 [cit. 2016-04-16]. Dostupné z: <http://www.linfo.org/proprietary.html>
- [18] Techbox: dotykové displeje – čím se liší rezistivní od kapacitního? | mobilenet.cz. <https://mobilenet.cz>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://mobilenet.cz/clanky/techbox-dotykovye-displeje—cim-se-lisi-rezistivni-od-kapacitniho-11566>

- [19] MFF / Informace pro veřejnost / Zprávičky / Víte proč ... dotykový displej reaguje?. <http://www.mff.cuni.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: http://www.mff.cuni.cz/verejnost/zpravicky/06_displej.htm
- [20] Dotykové technologie. <http://www.obcanskagramotnost.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: http://www.obcanskagramotnost.cz/documents/zpc_5_dotykove_technologie.pdf
- [21] Rezistivní displej. <http://www.mobilmania.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: http://www.mobilmania.cz/Files/Obrázky/art17/dotykové_displeje/obr3.jpg
- [22] Kapacitní displej. <http://zive.v.mfstatic.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: http://zive.v.mfstatic.cz/GetThumbNail.aspx?id_file=588612022&width=350&height=362&q=100
- [23] Techbox: váš telefon je prošpikovaný senzory | mobilenet.cz. <https://mobilenet.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://mobilenet.cz/clanky/techbox-vas-telefon-je-prospikovany-senzory-12496>
- [24] Vysvětlení jevu :: MEF. <http://fyzika.jreichl.com/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/419-vysvetleni-jevu>
- [25] L. Hájek, T. Hambálek. Gyroskopy. <http://fyzsem.fjfi.cvut.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <http://fyzsem.fjfi.cvut.cz/2010-2011/Zima10/proc/gyroskopy.pdf>
- [26] 1pair-lot-U8-IC-three-axis-gyroscope-ic-U18-Accelerometer-ic-For-iphone-5S-on-Logic.jpg. <http://g02.a.alicdn.com/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <http://g02.a.alicdn.com/kf/HTB1qMxjIFXXXb3XpXXq6xXFXXY/1pair-lot-U8-IC-three-axis-gyroscope-ic-U18-Accelerometer-ic-For-iphone-5S-on-Logic.jpg>
- [27] Jak funguje zaměření polohy pomocí GPS?. <http://www.svetandroida.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <http://www.svetandroida.cz/gps-princip-201503>
- [28] Techbox: jak funguje GPS, které vás dovede k vašemu cíli? | mobilenet.cz. <https://mobilenet.cz/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://mobilenet.cz/clanky/techbox-jak-funguje-gps-ktere-vas-dovede-k-vasemu-cili-11200>
- [29] VozejkMap – Aplikace pro Android ve službě Google Play. <https://play.google.com/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://play.google.com/store/apps/details?id=eu.bioport.vozejkmap&hl=cs>

- [30] Be My Eyes | Lend Your Eyes to the Blind. <http://www.bemyeyes.org/>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <http://www.bemyeyes.org/>
- [31] Parking Mobility – Aplikace pro Android ve službě Google Play. <https://play.google.com>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://play.google.com/store/apps/details?id=com.fortmobile.parkingmobility&hl=cs>
- [32] Forms – Aplikace pro Android ve službě Google Play. <https://play.google.com>. [online]. 20.4.2016 [cit. 2016-04-20]. Dostupné z: <https://play.google.com/store/apps/details?id=com.devicemagic.androidx.forms&hl=cs>
- [33] Fotografie reálné struktury MEMS mikrofonu. <http://automatizace.hw.cz/>. [online]. 20.5.2016 [cit. 2016-05-20]. Dostupné z: http://automatizace.hw.cz/files/images/image/MEMS_MICROPHONE_2.jpg
- [34] MEMS mikrofony – obecný popis struktury a funkce | Automatizace.HW.cz. <http://automatizace.hw.cz/>. [online]. 20.5.2016 [cit. 2016-05-20]. Dostupné z: <http://automatizace.hw.cz/mems-mikrofony-obecny-popis-struktury-a-funkce>
- [35] Techbox: vše o fotoaparátu v mobilu | mobilenet.cz. <https://mobilenet.cz/>. [online]. 20.5.2016 [cit. 2016-05-20]. Dostupné z: <https://mobilenet.cz/clanky/techbox-vse-o-fotoaparatu-v-mobilu-12397>
- [36] usporadani-barevnych-filtru-v-bayerove-masce. <https://mobilenet.cz/>. [online]. 20.5.2016 [cit. 2016-05-20]. Dostupné z: <https://mobilenet.cz/obrazek/usporadani-barevnych-filtru-v-bayerove-masce-118800>
- [37] prurez-snimace. <https://mobilenet.cz/>. [online]. 20.5.2016 [cit. 2016-05-20]. Dostupné z: <https://mobilenet.cz/obrazek/prurez-snimace-118801>
- [38] Čočka iPhone 6s. <http://www.appliste.cz>. [online]. 19.7.2016 [cit. 2016-07-19]. Dostupné z: <http://www.appliste.cz/wp-content/uploads/2015/12/2015-12-21-v-15.24.10.jpg>
- [39] More Resource Types. Android Developers. [online]. 26.7.2016 [cit. 2016-07-26]. Dostupné z: [\(http://developer.android.com/guide/topics/resources/more-resources.html#Dimension\)](https://developer.android.com/guide/topics/resources/more-resources.html#Dimension)
- [40] Mapy Google. Google.cz. [online]. 15.8.2016 [cit. 2016-08-15]. Dostupné z: <https://www.google.cz/maps>

- [41] MATOUŠEK, J. Parametry dopravní infrastruktury pro osoby s omezenou schopností pohybu. 2016. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta dopravní.
- [42] Vyhláška Ministerstva pro místní rozvoj ČR č. 398/2009 Sb. o obecných technických požadavcích zabezpečujících bezbariérové užívání staveb. In: Sbírka zákonů, Česká republika. 2009.

Seznam obrázků

Obrázek 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9]	8
Obrázek 2: Rezistivní displej [21]	12
Obrázek 3: Kapacitní displej [22]	13
Obrázek 4: Ukázka akcelometru a gyroskopu v prostředí chytrého telefonu [26]	14
Obrázek 5: Uspořádání barevných filtrů [36]	15
Obrázek 6: Průřez snímače [37]	15
Obrázek 7: Čočka iPhone 6s [38]	16
Obrázek 8: Fotografie reálné struktury MEMS mikrofonu	16
Obrázek 9: Ukázka jedné strany papírového formuláře [41]	19
Obrázek 10: Ukázka pole: Kód úseku komunikace	20
Obrázek 11: Blokové schéma technických prostředků	24
Obrázek 12: Vývojový diagram: Pořízení fotografie	25
Obrázek 13: Ukázka pořízení fotografie v prostředí aplikace	26
Obrázek 14: Vývojový diagram: Postup zjišťování GPS souřadnic	26
Obrázek 15: Vývojový diagram: Verifikace dat	28
Obrázek 16: Ukázka designového prostředí	29
Obrázek 17: Ukázka textového editoru	30
Obrázek 18: Ukázka prvků- Button	32
Obrázek 19: Ukázka prvků - Radio Button	32
Obrázek 20: Ukázka prvků - Checkbox	32
Obrázek 21: Ukázka prvků – Spinner	33
Obrázek 22: Ukázka prvků - Edit Text s atributem hint	33
Obrázek 23: Ukázka prvků - Image View	34
Obrázek 24: Ukázka prvků - Toast	34
Obrázek 25: Ukázka prvků - Calendar View	35
Obrázek 26: Ukázka prvků - Menu	35
Obrázek 27: Ukázka prvků - Popup Window	35
Obrázek 28.: Ukázka nápovědy k nerovnostem povrchu	40
Obrázek 29: Ukázka úvodní stránky aplikace ve vertikální i horizontální poloze	41
Obrázek 30: E-R diagram	42
Obrázek 31: Ukázka z aktivity vyplnění formuláře	43
Obrázek 32: Ukázka struktury tabulky	43
Obrázek 33: Ukázka vlastností prvků databáze	44
Obrázek 34: Ukázka seznamu formulářů	45

Obrázek 35: Ukázka načtení formuláře a jeho menu	45
Obrázek 36: Ukázka obrazovky nabídky „O aplikaci“	46
Obrázek 37: Měřený úsek [40]	47

Seznam tabulek

Tabulka 1: Procentuální rozložení zastoupení jednotlivých operačních systémů k únoru 2016 [9]	9
Tabulka 2: Výhody a nevýhody rezistivního displeje [18][20].....	12
Tabulka 3: Výhody a nevýhody kapacitního displeje. [18][20]	12

Seznam příloh

CD – ROM se zdrojovým kódem